

IBM Rational Developer for System z
Versión 9.1.1

*Guía de referencia de configuración de
host*



IBM Rational Developer for System z
Versión 9.1.1

*Guía de referencia de configuración de
host*



Nota

Antes de utilizar esta información, debe leer la información general que figura en el apartado “Avisos” en la página 241.

Novena edición (mayo de 2014)

Esta edición corresponde a IBM Rational Developer for System z Versión 9.1.1 (número de programa 5724-T07) y a todos los releases y modificaciones ulteriores hasta que se indique lo contrario en nuevas ediciones.

Puede pedir las publicaciones por teléfono o por fax. IBM Software Manufacturing Solutions acepta los pedidos de publicaciones entre las 8:30 de la mañana y las 7:00 de la tarde, hora estándar del este (EST). El número de teléfono es (800) 879-2755. El número de fax es (800) 445-9269. Los faxes deben enviarse a Attn: Publications, 3rd floor.

También puede pedir publicaciones a través de su representante de IBM o de la sucursal de IBM que presta servicio en su localidad. En la dirección que figura más abajo no hay publicaciones almacenadas.

IBM agradece sus comentarios. Puede enviar sus comentarios por correo a la siguiente dirección:

IBM Corporation
Attn: Information Development Department 53NA
Building 501 P.O. Box 12195
Research Triangle Park NC 27709-2195
Estados Unidos de América

Puede enviar sus comentarios por fax a: 1-800-227-5088 (EE.UU. y Canadá)

Cuando envía información a IBM, otorga a IBM un derecho no exclusivo a utilizar o distribuir la información del modo que IBM considere oportuno sin incurrir por ello en ninguna obligación para con usted.

Nota sobre los derechos restringidos de los usuarios del Gobierno de EE. UU. - El uso, la duplicación o la divulgación están sujetos a las restricciones establecidas en el contrato GSA ADP Schedule Contract con IBM Corp.

© Copyright IBM Corporation 2000, 2014.

Contenido

Figuras	vii
----------------	------------

Tablas	ix
---------------	-----------

Acerca de este documento	xi
---------------------------------	-----------

A quién va dirigido este documento	xii
Resumen de cambios	xii
Descripción del contenido del documento	xiv
¿Qué es Developer for System z?	xiv
Consideraciones relativas a la seguridad	xv
Consideraciones sobre TCP/IP	xv
Consideraciones sobre WLM	xv
Consideraciones acerca de los ajustes	xv
Consideraciones sobre el rendimiento	xv
Consideraciones sobre envío a cliente	xv
Consideraciones de CICSTS	xv
Consideraciones de salida de usuario	xvi
Personalizar el entorno TSO	xvi
Ejecutar varias instancias	xvi
Resolución de problemas de configuración	xvi
Configurar SSL y autenticación de X.509	xvi
Configurar TCP/IP	xvi

Guía de referencia de configuración de host de IBM Rational Developer for System z	1
---	----------

Capítulo 1. Comprender Developer for System z	3
--	----------

Visión general de los componentes	4
RSE como aplicación Java	5
Propietarios de tareas	7
Flujo de conexión	8
Depurador integrado	10
CARMA	11
Archivos de configuración de CARMA	11
CRASTART	12
Sometimiento por lotes	12
Propietario de bloqueo de conjunto de datos	13
Liberar un bloqueo	14
Estructura de directorios de z/OS UNIX	15
Privilegios de actualización para usuarios no administradores del sistema	17
Mandatos de seguridad de gran utilidad	17
Mandatos útiles de z/OS UNIX	17
Configuración de ejemplo	18

Capítulo 2. Consideraciones relativas a la seguridad	19
---	-----------

Métodos de autenticación	20
ID de usuario y contraseña	20
ID de usuario y contraseña para una sola vez	20
ID de usuario y frase de contraseña	20

Certificado X.509	20
Autenticación del supervisor de trabajos JES	21
Autenticación del gestor de depuración	21
Seguridad de conexión	21
Limitar la comunicación externa a puertos especificados	22
Cifrado de comunicaciones utilizando SSL o TLS	22
comprobación de puerto de entrada	23
Uso de PassTickets	23
Registro de auditoría	24
Control de auditoría	24
Procesamiento de auditoría	25
Datos de auditoría	25
Seguridad de JES	26
Acciones en trabajos - limitaciones de destino	26
Acciones en trabajos - limitaciones de ejecución	28
Acceso a los archivos de spool	29
Comunicación cifrada con SSL/TLS	30
Comunicación cifrada con el depurador integrado	31
Autenticación de cliente mediante certificados X.509	32
Validación de la autoridad certificadora (CA)	33
(Opcional) Consulta en una lista de certificados revocados (CRL)	33
Autenticación del software de seguridad	34
Autenticación del daemon RSE	35
Comprobación de puerto de entrada (POE)	36
Alterar las funciones de cliente	36
OFF.REMOTECOPY.MVS	37
Grupos de desarrollador Envío a cliente	37
Seguridad de archivo de registro	39
Permisos de la clase UNIXPRIV	41
permiso de perfil BPX.SUPERUSER	41
UID 0	42
Seguridad de depuración	42
Seguridad de CICSTS	42
Repositorio CRD	43
Transacciones CICS	43
Comunicación cifrada con SSL	43
Seguridad de SCLM	43
Información variada	43
Desecho de GATE	43
ACEE gestionado	44
Almacenamiento en memoria caché ACEE	44
Archivos de configuración de Developer for System z	44
Rastreo del daemon de bloqueo - FEJJCENFG	44
RSE - rsed.envvars	45
RSE - ssl.properties	46
RSE - pushtoclient.properties	46
Definiciones de seguridad	47
Requisitos y lista de comprobación	47
Activar los valores y las clases de seguridad	49
Definición de un segmento OMVS para usuarios de Developer for System z	50

Definir las tareas iniciadas de Developer for System z	50
Definición de RSE como un servidor z/OS UNIX seguro	51
Definir bibliotecas controladas por programa MVS para RSE	52
Definir el soporte de PassTicket para RSE	53
Definir permiso de acceso de archivos z/OS UNIX para RSE	54
Definir la protección de aplicaciones para el RSE	54
Definir archivos controlados por programa z/OS UNIX para el servidor	55
Definir la seguridad de mandatos JES	55
Definir acceso al depurador integrado	57
Definir los perfiles de conjunto de datos	57
Verificar los valores de seguridad	60

Capítulo 3. Consideraciones sobre TCP/IP 63

Puertos TCP/IP	63
Comunicación externa	64
Comunicación interna	64
Reserva de puerto TCP/IP	65
Puertos CARMA y TCP/IP	65
Consideraciones sobre LDAP	66
Alteración temporal del comportamiento de TCP/IP predeterminado	66
ACK retardado	66
Varias pilas (CINET)	66
CARMA y afinidad de pila	67
crastart*.conf	67
CRASUB*	67
VIPA dinámico distribuido	68
Restringir la selección de puerto	69
Configuración de ejemplo	71
Sistema SYS1 – perfil de TCP/IP	72
Sistema SYS2 – perfil de TCP/IP	72

Capítulo 4. Consideraciones sobre WLM. 73

Clasificación de carga de trabajo	73
Reglas de clasificación	74
Establecimiento de objetivos	75
Consideraciones para la selección de objetivos	76
STC	77
OMVS	77
JES	79
ASCH	79
CICS	80

Capítulo 5. Consideraciones acerca de los ajustes 81

Uso de recursos	81
Visión general	82
Recuento de espacios de direcciones	83
Recuento de procesos	86
Recuento de hebras	89
Uso temporal de recursos	94
Recuento de hebras	94
Uso de almacenamiento	98

Límite de tamaño de almacenamiento dinámico Java	98
Límite de tamaño del espacio de direcciones	99
Directrices de estimación de tamaño	100
Análisis del uso de almacenamiento de ejemplo	101
Uso de espacio del sistema de archivos de z/OS UNIX	105
Definiciones de recursos clave	108
/etc/rdz/rsed.envvars	108
SYS1.PARMLIB(BPXPRMxx)	109
definiciones de varios recursos	112
Tarjeta EXEC del servidor JCL	112
FEK.#CUST.PARMLIB(FEJJCENFG)	112
SYS1.PARMLIB(IEASYSxx)	113
SYS1.PARMLIB(IVTPRMxx)	113
SYS1.PARMLIB(ASCHPMxx)	113
Supervisión	114
Supervisión de RSE	114
Supervisión de z/OS UNIX	115
Supervisar la red	117
Supervisión de sistemas de archivos z/OS UNIX	117
Configuración de ejemplo	118
Recuento de agrupaciones de hebras	118
Determinar los límites mínimos	118
Definición de límites	119
Utilización de recursos de supervisor	120

Capítulo 6. Consideraciones sobre el rendimiento 123

Utilizar sistemas de archivos zFS	123
Evitar el uso de STEPLIB	123
Mejorar el acceso a las bibliotecas del sistema	123
Bibliotecas de tiempo de ejecución de Language Environment (LE)	124
Desarrollo de aplicaciones	124
Mejorar el rendimiento de la comprobación de seguridad	125
Gestión de cargas de trabajo	125
Almacenamiento dinámico Java de tamaño fijo	125
Opción -Xquickstart de Java	126
Compartimiento de clases entre las JVM	126
Habilitar el compartimiento de clases	127
Límites de tamaño de la memoria caché	127
Seguridad de memoria caché	127
SYS1.PARMLIB(BPXPRMxx)	127
Espacio de disco	128
Utilidades para la gestión de cachés	128

Capítulo 7. Consideraciones sobre envío a cliente 131

Introducción	131
Sistema primario	132
Metadatos Envío a cliente	133
Ubicación de metadatos	133
Seguridad de metadatos	133
Uso del espacio de metadatos	134
Control de configuración del cliente	134
Control de versión del cliente	135
Varios grupos de desarrollador	135
Activación	135

Concatenación de grupos	136
Enlace de espacio de trabajo	137
Ubicación de metadatos de grupo	138
Pasos de configuración	138
Selección de grupo basada en LDAP	140
Esquema de LDAP	141
Selección del servidor LDAP	142
Ubicación del servidor LDAP	142
Configuración de ejemplo	143
Adición del extremo de Envío a cliente a LDAP	143
Configuración de grupo LDAP inicial	144
Añadir desarrolladores a grupos LDAP.	144
pushtoclient.properties	145
rsed.envvars.	145
/var/rdz/pushtoclient/*install	145
Selección de grupo basada en SAF	145
Configuración de ejemplo	147
Definición de seguridad	148
pushtoclient.properties	148
rsed.envvars.	148
/var/rdz/pushtoclient/*install	148
Periodo de gracia para el rechazo de cambios	149
Proyectos basados en host	149

Capítulo 8. Consideraciones de CICSTS 151

RESTful versus Servicio Web	152
Comparación entre regiones de conexión primarias y no primarias	152
Registro de instalación de recursos CICS	153
Seguridad del Gestor de despliegue de aplicaciones	153
Seguridad del repositorio CRD	153
Seguridad de conducto	153
Seguridad de transacción	153
Comunicación cifrada con SSL.	155
Seguridad de recursos	155
Programa de utilidad administrativa	155
Notas de migración del programa de utilidad administrativo	159
Mensajes del programa de utilidad administrativo	160
Depuración de transacción CICS	162

Capítulo 9. Consideraciones de salida de usuario. 165

Características de salida de usuario	165
Activación de la salida de usuario	165
Escritura de una rutina de salida de usuario	165
Mensajes de consola	166
Ejecución con un ID de usuario variable	166
Script de shell de z/OS UNIX	166
Exec REXX de z/OS UNIX	167
Puntos de salida disponibles	168
audit.action	168
logon.action	168

Capítulo 10. Personalizar el entorno TSO 171

El servicio de mandatos TSO	171
---------------------------------------	-----

Métodos de acceso	171
Utilizar el método de acceso de Pasarela de cliente TSO/ISPF	172
ISPF.conf	172
Utilizar perfiles ISPF existente.	172
Utilizar un exec asignación	173
Utilizar varios ejecutables de asignación	173
Varios archivos ISPF.conf con varias configuraciones de Developer for System z	173

Capítulo 11. Ejecutar varias instancias 175

Configuración idéntica en todo un sysplex	175
Archivos de configuración diferentes con idéntico nivel de software	176
Sincronización automatizada	177
Todas las demás situaciones	178

Capítulo 12. Resolución de problemas de configuración 181

Anotar y configurar el análisis mediante FEKLOGS	181
Archivos de registro	182
Registro del gestor de depuración	184
Registro del supervisor de trabajos JES	184
Daemon RSE y registro de la agrupaciones de hebras.	184
Registro de usuario de RSE.	185
Registro de SCLM Developer Toolkit	186
Registro de CARMA	186
fekfivpc, registro de prueba IVP	187
Registro de prueba IVP de fekfivpi	187
Registro de prueba IVP de fekfivps	187
Registro de revisión de código.	187
Registro de cobertura de código	187
Archivos de vuelco	188
Vuelcos de MVS	188
Volcados de Java	188
Ubicaciones de volcados de z/OS UNIX	190
Rastrear	190
Rastreo del gestor de depuración.	190
Rastreo del supervisor de trabajos JES	190
Rastreo RSE	190
Rastreo de CARMA	191
Rastreo de información de retorno de errores	192
Bits de permiso de z/OS UNIX	193
Atributo del sistema de archivos SETUID	193
Autorización de control de programa	194
Autorización de APF	195
Bit de permanencia	195
Puertos TCP/IP reservados.	196
Tamaño del espacio de direcciones	198
Requisitos de JCL de inicio	198
Limitaciones establecidas en SYS1.PARMLIB(BPXPRMxx)	198
Limitaciones almacenadas en el perfil de seguridad	198
Limitaciones aplicadas por la rutinas de salida del sistema	198
Limitaciones para el direccionamiento de 64 bits	199
Información variada	199

Terminación anómala de espacio B37 de retorno de errores	199
Límites del sistema	199
Conexión rehusada	199
OutOfMemoryError	200
Emulador de conexión de host	200

Capítulo 13. Configurar SSL y autenticación de X.509 201

Elegir entre SSL o TLS como método de cifrado	202
Decida dónde desea almacenar los certificados y claves privadas	202
Crear un anillo de claves con RACF	203
Clonar la configuración RSE existente	205
Actualizar rsed.envvars para habilitar la coexistencia	205
Actualizar ssl.properties para habilitar la SSL	206
Activar la SSL creando un daemon RSE nuevo	206
Probar la conexión	207
(Opcional) Añadir soporte de autorización al cliente de X.509	210
(Opcional) Crear una base de datos de claves con gskkyman	210
(Opcional) Crear un almacén de claves con keytool	213

Capítulo 14. Configurar AT-TLS. 215

Configurar syslogd	216
Configuración AT-TLS en PROFILE.TCPIP	216
Tarea iniciada del agente de política	217
Configuración del agente de política	217
Política AT-TLS	218

Consideraciones TLS v1.2	219
Actualizaciones de seguridad de AT-TLS	220
Activación de la política AT-TLS	222

Capítulo 15. Configurar TCP/IP 225

Dependencia del nombre de host	225
¿Qué son los resolventes?	226
Qué es el orden de búsqueda de la información de configuración	226
Orden de búsqueda utilizado en el entorno z/OS UNIX	227
Archivos de configuración de resolvente base	227
Tablas de conversión	228
Tablas de hosts locales	228
Aplicación de esta información de configuración a Developer for System z	229
La dirección del host no se resuelve correctamente	231

Bibliografía 233

Publicaciones a las que se hace referencia	233
Publicaciones informativas	236

Glosario 237

Avisos 241

Licencia de copyright.	244
Reconocimientos de marcas registradas.	245

Índice. 247

Figuras

1.	Visión general de los componentes	4
2.	RSE como aplicación Java	5
3.	Propietarios de tareas	7
4.	Flujo de conexión	8
5.	Depurador integrado	10
6.	Flujo de CARMA	11
7.	Flujo de determinación de puesta en cola de conjunto de datos	13
8.	Estructura de directorios de z/OS UNIX	15
9.	Política AT-TLS para el gestor de depuración	32
10.	Puertos TCP/IP	63
11.	update.sh - soportar la configuración de DDVIPA con un cortafuegos	70
12.	Ejemplo de VIPA dinámico distribuido	71
13.	Clasificación de WLM	73
14.	Número máximo de espacios de direcciones	85
15.	Número de espacios de direcciones por cliente	86
16.	Número máximo de procesos	87
17.	Número de procesos de STCRSE	88
18.	Número de procesos por cliente.	89
19.	Número máximo de hebras de agrupación de hebras RSE (extractores de una sola hebra)	92
20.	Número máximo de hebras de agrupación de hebras RSE (extractores de varias hebras)	92
21.	Número máximo de hebras del Supervisor de trabajos JES	92
22.	Número máximo de hebras del gestor de depuración.	92
23.	Número máximo de hebras de agrupación de hebras RSE (extractores de una sola hebra)	97
24.	Número máximo de hebras de agrupación de hebras RSE (extractores de varias hebras)	97
25.	Número máximo de hebras del Supervisor de trabajos JES	97
26.	Número máximo de hebras del gestor de depuración.	97
27.	Uso de recursos con 5 inicios de sesión	102
28.	Uso de recursos con 5 inicios de sesión (continuación)	103
29.	Uso de recursos al editar un miembro PDS	104
30.	uso de espacio del sistema de archivos de z/OS UNIX	106
31.	Utilización de recursos de configuración de ejemplo	121
32.	Definición de esquema LDAP de ejemplo	141
33.	ADNJSPAU - programa de utilidad administrativo de CICSTS	157
34.	ADNJSPAU - programa de utilidad administrativa de CICSTS (parte 2 de 3)	158
35.	ADNJSPAU - Programa de utilidad administrativa de CICSTS (parte 3 de 3)	159
36.	RSEDSSL - Trabajos de usuario del daemon RSE para SSL	207
37.	Diálogo Importar certificado de host	208
38.	Diálogo Preferencias - SSL	209

Tablas

1. Mandatos de la consola del supervisor de trabajos JES	26	24. Uso de recursos específicos del usuario	82
2. Matriz de permisos de mandato LIMIT_COMMANDS	27	25. Recuento de espacios de direcciones	83
3. Perfiles JESSPOOL ampliados	27	26. Límites de espacios de direcciones	86
4. Matriz de autorización de consola LIMIT_CONSOLE	28	27. Recuento de procesos	86
5. matriz de permisos de examen de LIMIT_VIEW	29	28. Límites de procesos	89
6. Mecanismos de almacenamiento de certificados de SSL	30	29. Recuento de hebras	90
7. Información SAF para alterar las funciones de cliente	37	30. Límites de hebras	93
8. Información SAF de Envío a cliente	38	31. Recuento de hebras	94
9. Permisos relacionados UNIXPRIV de z/OS UNIX	41	32. Límites de hebras	98
10. Información SAF para funciones de depuración	42	33. Valores de referencia para uso de almacenamiento.	101
11. Variables de configuración de seguridad	47	34. Directivas de salidas de registro	107
12. Mandatos de operador del Supervisor de trabajos JES2	56	35. Directivas de salida temporales	108
13. Mandatos de operador del Supervisor de trabajos JES3	56	36. Matriz de soporte de grupo de Envío a cliente para *.enabled	135
14. Subsistemas de punto de entrada de WLM	74	37. Matriz de soporte de grupo de Envío a cliente para reject.*.updates	136
15. Calificadores de trabajo de WLM	75	38. Concatenaciones de grupo Envío a cliente	136
16. Cargas de trabajo WLM	76	39. Enlaces de grupo de configuración de espacio de trabajo.	137
17. Cargas de trabajo WLM - STC	77	40. Enlaces de grupo de producto de espacio de trabajo	137
18. Cargas de trabajo WLM - OMVS	78	41. Información LDAP de Envío a cliente	140
19. Cargas de trabajo WLM - JES	79	42. Información SAF de Envío a cliente	146
20. Cargas de trabajo WLM - ASCH	80	43. Variables de JAVA_DUMP_TDUMP_PATTERN.	189
21. Cargas de trabajo de WLM - CICS	80	44. Mecanismos de almacenamiento de certificados de SSL.	202
22. Uso de recursos comunes	82	45. Definiciones locales disponibles para el resolvente	231
23. Uso de recursos requisito específicos del usuario	82	46. Publicaciones a las que se hace referencia	233
		47. Sitios Web a los que se hace referencia	236
		48. Publicaciones informativas	236

Acerca de este documento

Este documento proporciona información básica sobre diferentes tareas de configuración de IBM® Rational Developer for System z y otros componentes y productos de z/OS (como WLM y CICS).

De aquí en adelante, en este manual se utilizarán los siguientes nombres:

- *IBM Rational Developer for System z* se denomina *Developer for System z*.
- *IBM Rational Developer for System z Depurador integrado* se denomina *Depurador integrado*.
- *Common Access Repository Manager* se denominará *CARMA*.
- *Software Configuration and Library Manager Developer Toolkit* se denominará *SCLM Developer Toolkit*, cuya abreviatura es *SCLMDT*.
- *z/OS UNIX System Services* se denomina *z/OS UNIX*.
- *Customer Information Control System Transaction Server* se denomina *CICSTS* y se abrevia como *CICS*.

Este documento forma parte de un conjunto de documentos que describen la configuración de host de Developer for System z. Cada uno de estos documentos está dirigido a un público específico. No es necesario que lea todos los documentos para completar la configuración de Developer for System z.

- En la *Guía de configuración de host de IBM Rational Developer for System z* (SC11-3660) se describen detalladamente todas las tareas y opciones (incluidas las que son opcionales) de planificación y configuración y se proporcionan escenarios alternativos.
- En la *Guía de referencia de configuración de host de IBM Rational Developer for System z* (SC11-7903) describe el diseño de Developer for System z y proporciona información previa para varias tareas de configuración de componentes de Developer for System z, z/OS y otros productos (tales como WLM y CICS) relacionados con Developer for System z.
- En la *Guía de inicio rápido de configuración de host de IBM Rational Developer for System z* (GI11-8628) se describe una configuración mínima de Developer for System z.
- En la *Guía del programa de utilidad de configuración de host de IBM Rational Developer for System z* (SC14-7282) se describe el programa de utilidad de configuración de host, una aplicación de panel ISPF que le guía por pasos de personalización opcionales básicos y comunes para Developer for System z.

La información de este documento se aplica a todos los paquetes de IBM Rational Developer for System z Versión 9.1.1.

Para obtener las versiones más actualizadas de este documento, consulte la Guía de referencia de configuración de host de *IBM Rational Developer for System z* (SC11-7903) que está disponible en <http://www-05.ibm.com/e-business/linkweb/publications/servlet/pbi.wss?CTY=US&FNC=SRX&PBL=SC14-7290>.

Para obtener las versiones más actualizadas de toda la documentación, incluyendo instrucciones de instalación, libros blancos, podcasts y guías de aprendizaje,

consulte la página de biblioteca del sitio web de IBM Rational Developer for System z (http://www-01.ibm.com/software/sw-library/en_US/products/Z964267S85716U24/).

A quién va dirigido este documento

Este documento está destinado a los programadores de sistemas que configuran y ajustan IBM Rational Developer for System z Versión 9.1.1.

Mientras que los pasos de configuración reales se describen en otra publicación, esta publicación proporciona una lista detallada de diversos temas relacionados, como por ejemplo, el ajuste, la configuración de seguridad y otros temas. Para utilizar esta documentación, debe estar familiarizado con z/OS UNIX System Services y con los sistemas de hospedaje MVS.

Resumen de cambios

En esta sección se resumen los cambios de la *Guía de referencia de configuración de host de IBM Rational Developer for System z Versión 9.1.1*, SC11-7903-08 (actualizada en diciembre de 2014).

Los cambios técnicos o las adiciones al texto y las ilustraciones se indican mediante una línea vertical situada a la izquierda del cambio.

Información nueva:

- Perfil de seguridad del depurador actualizados. Consulte “Seguridad de depuración” en la página 42.
- Se ha añadido información sobre el soporte para frases de contraseña. Consulte “Métodos de autenticación” en la página 20.

Este documento contiene información presentada anteriormente en la *Guía de referencia de configuración de host IBM Rational Developer for System z Versión 9.1.1*, SC11-7903-07.

Información nueva:

- Se ha añadido información sobre seguridad de archivos de registro. Consulte “Seguridad de archivo de registro” en la página 39.
- Se ha añadido información sobre el soporte de grupo para rechazar actualizaciones de envío a cliente. Consulte “Varios grupos de desarrollador” en la página 135.
- Se ha actualizado la información de uso de recursos. Consulte Capítulo 5, “Consideraciones acerca de los ajustes”, en la página 81.
- Se ha actualizado la información de archivos de registro y rastreo. Consulte Capítulo 12, “Resolución de problemas de configuración”, en la página 181.

Este documento contiene información presentada anteriormente en la *Guía de referencia de configuración de host de IBM Rational Developer for System z Versión 9.0.1*, SC11-7903-07.

Información nueva:

- Se ha añadido información sobre la configuración de AT-TLS. Consulte Capítulo 14, “Configurar AT-TLS”, en la página 215.

Este documento contiene información presentada anteriormente en la *IBM Rational Developer for System z Versión 9.0.1 Guía de referencia de configuración de host*, SC11-7903-07.

Información nueva:

- Se ha añadido información sobre nombres de archivos de registro con fecha y hora. Consulte “Archivos de registro” en la página 182.
- Se ha añadido información sobre sucesos auditables nuevos. Consulte Datos de auditoría.

Este documento incluye información ya presentada en la Guía de referencia de configuración de host de *IBM Rational Developer for System z Versión 9.0*, SC11-7903-04.

Información nueva:

- Se ha actualizado el uso del puerto TCP/IP. Consulte “Puertos TCP/IP” en la página 63.
- Se ha añadido un ejemplo para sincronizar automáticamente 2 daemons RSE. Consulte “Sincronización automatizada” en la página 177.
- Se ha añadido información sobre archivos de registro nuevos. Consulte “Archivos de registro” en la página 182.

Este documento contiene información presentada anteriormente en la *Guía de referencia de configuración de host de IBM Rational Developer for System z Versión 8.5.1*, SC11-7903-07.

Información nueva:

- Se ha añadido información sobre perfiles SAF para alterar las funciones de cliente. Consulte “Alterar las funciones de cliente” en la página 36.
- Se han actualizado los números de uso de recursos. Consulte Capítulo 5, “Consideraciones acerca de los ajustes”, en la página 81
- Valor predeterminado actualizado para número máximo de usuarios por agrupación de hebras. Consulte Capítulo 5, “Consideraciones acerca de los ajustes”, en la página 81.

Este documento contiene información presentada anteriormente en la *Guía de referencia de configuración de host de IBM Rational Developer for System z Versión 8.5*, SC11-7903-07.

Información nueva:

- Se ha actualizado la información de seguridad del Supervisor de trabajos JES. Consulte Capítulo 2, “Consideraciones relativas a la seguridad”, en la página 19.
- Se ha añadido información sobre las salidas de usuario. Consulte Capítulo 9, “Consideraciones de salida de usuario”, en la página 165.

Este documento contiene información presentada anteriormente en la *Guía de referencia de configuración de host IBM Rational Developer for System z Versión 8.0.3*, SC11-7903-07.

Información nueva:

- Estructura de directorios de z/OS UNIX nueva. Consulte “Estructura de directorios de z/OS UNIX” en la página 15.

- Se ha añadido información sobre control de cliente basado en host. Consulte Capítulo 7, “Consideraciones sobre envío a cliente”, en la página 131.
- Se ha añadido información de enviar a cliente relacionada con la seguridad. Consulte “Grupos de desarrollador Envío a cliente” en la página 37.
- Uso del documento de ACEE gestionados. Consulte “ACEE gestionado” en la página 44.
- Se ha añadido información sobre el proceso de registro de auditoría automatizado. Consulte “Procesamiento de auditoría” en la página 25.
- Se ha actualizado la información sobre las directivas relacionadas con la seguridad y la auditoría en los archivos de configuración. Consulte “Archivos de configuración de Developer for System z” en la página 44.
- Se ha añadido información de TCP/IP adicional. Consulte Capítulo 3, “Consideraciones sobre TCP/IP”, en la página 63.
- Se ha actualizado la información de Autoridad certificadora actualizada para la comunicación SSL. Consulte Capítulo 13, “Configurar SSL y autenticación de X.509”, en la página 201.
- Se ha actualizado el uso de recursos. Consulte “Uso de recursos” en la página 81.

Este documento contiene información presentada anteriormente en la *Guía de referencia de configuración de host de IBM Rational Developer for System z Versión 8.0.1*, SC11-7903-00.

Información nueva:

- Sección CARMA en Descripción de Developer for System z. Consulte “CARMA” en la página 11.
- Información general relacionada con TCP/IP. Consulte Capítulo 3, “Consideraciones sobre TCP/IP”, en la página 63.
- Resolución de la finalización anómala del espacio B37. Consulte “Terminación anómala de espacio B37 de retorno de errores” en la página 199.

Información eliminada:

- La información que antes se proporcionaba en la *Guía de configuración de host de IBM Rational Developer for System z versión 7.6.1* (SC11-3660-04) se ha dividido ahora en dos documentos: *Guía de configuración de host de IBM Rational Developer for System z* (SC11-3660) y *Guía de referencia de configuración de host de IBM Rational Developer for System z* (SC11-7903).
- La información relacionada con la configuración de APPC se ha pasado al libro blanco *Using APPC to provide TSO command services* (SC14-7291).
- Configurar INETD

Descripción del contenido del documento

En esta sección se resume la información presentada en este documento.

¿Qué es Developer for System z?

El host de Developer for System z está formado por varios componentes que interactúan para proporcionar al cliente acceso a los servicios y datos del host. Comprender el diseño de estos componentes puede ayudarle a tomar las decisiones de configuración correctas.

Consideraciones relativas a la seguridad

Developer for System z proporciona a los usuarios acceso al sistema central en una estación de trabajo que no es del sistema central. Algunos aspectos importantes de la configuración del producto son: validar las solicitudes de conexión, proporcionar una comunicación segura entre el host y la estación de trabajo, y autorizar y auditar la actividad.

Consideraciones sobre TCP/IP

Developer for System z utiliza TCP/IP para proporcionar a los usuarios acceso al sistema central en una estación de trabajo que no es del sistema central. También utiliza TCP/IP para establecer comunicación entre distintos componentes y otros productos.

Consideraciones sobre WLM

Al contrario que las aplicaciones z/OS tradicionales, Developer for System z no es una aplicación monolítica que se pueda identificar fácilmente para el Gestor de carga de trabajo (WLM). Developer for System z está formado por varios componentes que interactúan para proporcionar al cliente acceso a los servicios y datos del host. Algunos de estos servicios están activos en diferentes espacios de direcciones, lo que resulta en diferentes clasificaciones WLM.

Consideraciones acerca de los ajustes

RSE (Explorador de Sistemas remotos) es el núcleo de Developer for System z. Para gestionar las conexiones y cargas de trabajo de los clientes, RSE está formado por un espacio de direcciones de daemon, que controla los espacios de direcciones de agrupaciones de hebras. El daemon actúa como punto focal a efectos de conexión y gestión, mientras que las agrupaciones de hebras procesan las cargas de trabajo del cliente.

Ello hace que RSE sea el destino principal para ajustar la configuración de Developer for System z. Sin embargo, para mantener a cientos de usuarios, cada uno de los cuales utiliza 17 o más hebras, una cantidad determinada de almacenamiento y, posiblemente, uno o más espacios de direcciones es necesario configurar correctamente Developer for System z y z/OS.

Consideraciones sobre el rendimiento

z/OS es un sistema operativo sumamente personalizable, y los cambios de sistema (a veces pequeños) pueden afectar considerablemente al rendimiento global. En este capítulo se resaltan algunos de los cambios que se pueden hacer para mejorar el rendimiento de Developer for System z.

Consideraciones sobre envío a cliente

Enviar a cliente o el control de clientes basado en host, tiene soporte para la gestión central de lo siguiente:

- Archivos de configuración del cliente
- Versión del producto del cliente
- Definiciones del proyecto

Consideraciones de CICSTS

Este capítulo contiene información útil para un administrador de CICS Transaction Server.

Consideraciones de salida de usuario

Este capítulo le ayuda a mejorar Developer for System z escribiendo rutinas de salida.

Personalizar el entorno TSO

Este capítulo le ayuda a emular un procedimiento de inicio de sesión TSO añadiendo sentencias DD y conjuntos de datos al entorno TSO en Developer for System z.

Ejecutar varias instancias

En algunas ocasiones le interesará tener múltiples instancias de Developer for System z activas en el mismo sistema; por ejemplo, al probar una ampliación. Sin embargo, algunos recursos como los puertos TCP/IP no se pueden compartir, por lo que los valores predeterminados no siempre son aplicables. Utilice la información de este capítulo para planificar la coexistencia de distintas instancias de Developer for System z y después podrá usar esta guía de configuración para personalizarlas.

Resolución de problemas de configuración

Este capítulo se propone ayudarle a resolver algunos problemas comunes que pueden surgir durante la configuración de Developer for System z, y tiene las secciones siguientes:

- Anotar y configurar el análisis mediante FEKLOGS
- Archivos de registro
- Archivos de vuelco
- Rastrear
- Bits de permiso de z/OS UNIX
- Puertos TCP/IP reservados
- Tamaño del espacio de direcciones
- Transacción APPC y el servicio de mandatos TSO
- Información variada

Configurar SSL y autenticación de X.509

Esta sección se propone ayudarle a resolver algunos problemas comunes que pueden surgir al configurar la capa de sockets segura (SSL) o durante la comprobación o modificación de una configuración existente. Esta sección también facilita una configuración de ejemplo para admitir que los usuarios se autenticquen con un certificado X.509.

Configurar TCP/IP

Esta sección se propone ayudarle a resolver algunos problemas comunes que pueden surgir al configurar TCP/IP, o durante la tarea de comprobar o modificar una configuración existente.

Guía de referencia de configuración de host de IBM Rational Developer for System z

Capítulo 1. Comprender Developer for System z

El host de Developer for System z consta de varios componentes que interactúan para dar al cliente acceso a los servicios y datos del host. Comprender el diseño de estos componentes puede ayudarle a tomar las decisiones de configuración correctas.

En este capítulo se tratan estos temas:

- “Visión general de los componentes” en la página 4
- “RSE como aplicación Java” en la página 5
- “Propietarios de tareas” en la página 7
- “Flujo de conexión” en la página 8
- “Depurador integrado” en la página 10
- “CARMA” en la página 11
- “Propietario de bloqueo de conjunto de datos” en la página 13
- “Estructura de directorios de z/OS UNIX” en la página 15

Visión general de los componentes

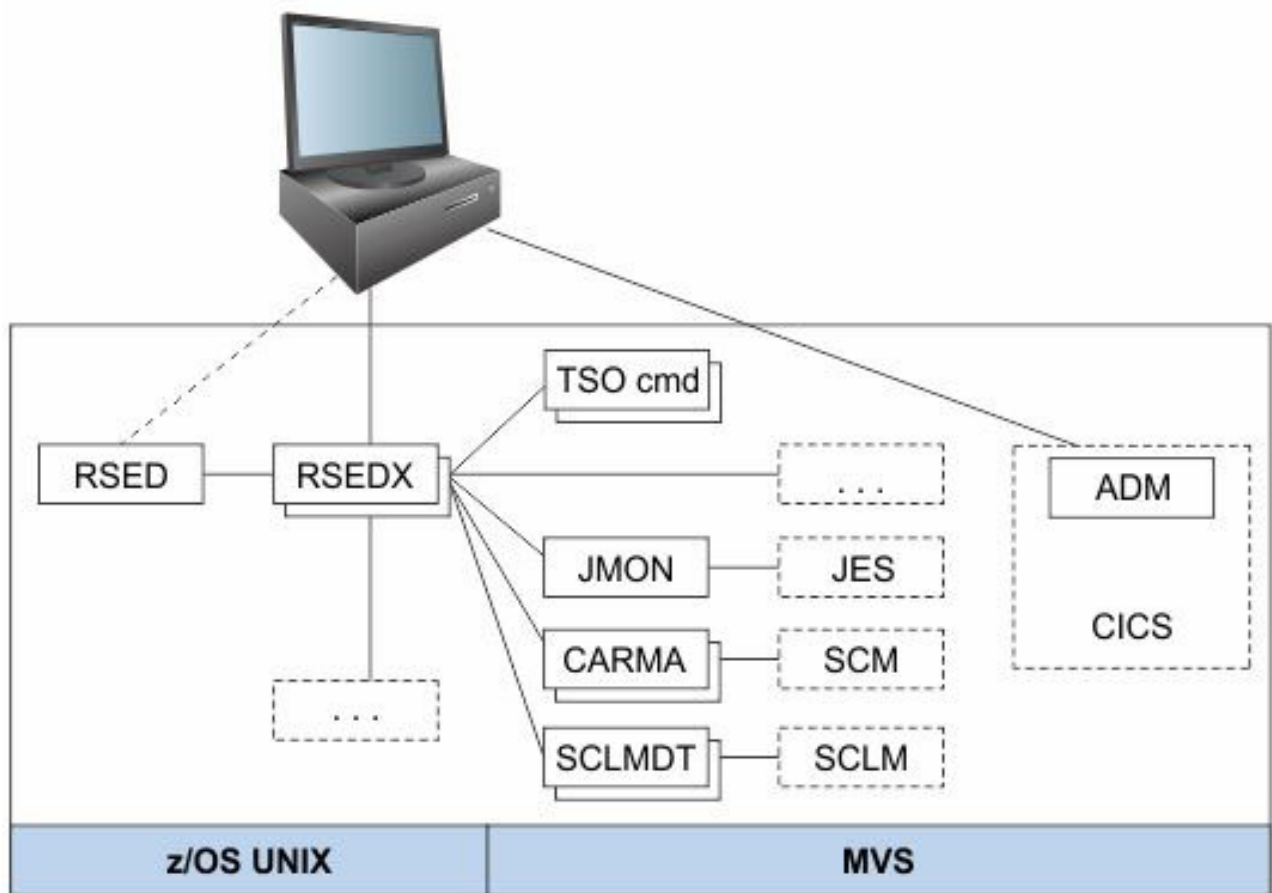


Figura 1. Visión general de los componentes

La Figura 1 muestra una visión general del diseño de Developer for System z en el sistema host.

- El Explorador de sistemas remotos (RSE) proporciona servicios del núcleo como los de conectar el cliente al host e iniciar otros servidores para servicios específicos. El RSE consta de dos entidades lógicas:
 - El daemon RSE (RSED), que gestiona la configuración de conexiones. El daemon RSE es responsable de la ejecución en modalidad de servidor único. Para ello, el daemon RSE crea uno o varios procesos hijo conocidos como agrupaciones de hebras RSE (RSEDx).
 - El servidor RSE, que maneja las solicitudes de clientes individuales. Un servidor RSE está activo como hebra dentro de una agrupación de hebras RSE.
- El Gestor de depuración (DBGMR) coordina la actividad del Depurador Integrado.
- El Servicio de mandatos TSO (TSO cmd) proporciona una interfaz de tipo por lotes para los mandatos TSO y ISPF.
- El Supervisor de trabajos JES (JMON) suministra todos los servicios relacionados con JES.

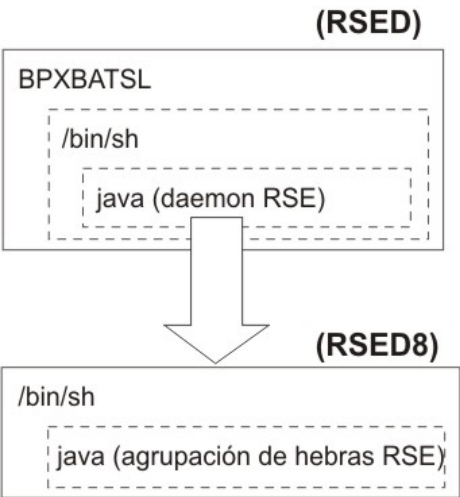
- El CARMA (Common Access Repository Manager) proporciona una interfaz para interactuar con SCM (Software Configuration Manager), tales como el CA Endeavor.
- SCLM Developer Toolkit (SCLMDT) proporciona una interfaz para mejorar e interactuar con SCLM.
- El Gestor de despliegue de aplicaciones (ADM) proporciona varios servicios relacionados con CICS.
- Hay más servicios disponibles, que pueden ser proporcionados por Developer for System z o por el software correquisito.

La descripción del párrafo y la lista anteriores muestra el rol central asignado al RSE. Salvo algunas excepciones, toda la comunicación de cliente va a través del RSE. Ello permite una configuración de la red de seguridad sencilla, ya que únicamente se utiliza un conjunto limitado de puertos para la comunicación cliente-host.

Para gestionar las conexiones y cargas de trabajo de los clientes, RSE está formado por un espacio de direcciones de daemon, que controla los espacios de direcciones de agrupaciones de hebras. El daemon actúa como punto focal a efectos de conexión y gestión, mientras que las agrupaciones de hebras procesan las cargas de trabajo del cliente. Basándose en los valores definidos en el archivo de configuración rsed.envvars y en la cantidad de conexiones de cliente reales, el daemon puede iniciar varios espacios de direcciones de agrupaciones de hebras.

RSE como aplicación Java

Procesos z/OS UNIX



Uso de almacenamiento Java

Sistema - compartido
Sistema - privado
Código (z/OS UNIX, Java, RSE)
Almacenamiento dinámico Java
No utilizado

NOMBRETRABAJO	Estado	PID	PPID	Mandato
RSED	FILE SYS KERNEL WAIT	50331904	1	BPXBATSL
RSED	WAITING FOR CHILD	67109114	50331904	/bin/sh...
RSED	FILE SYS KERNEL WAIT	50331949	67109114	Java...
RSED8	WAITING FOR CHILD	307	50331949	/bin/sh...
RSED8	FILE SYS KERNAL WAIT	308	307	java...

Figura 2. RSE como aplicación Java

La Figura 2 en la página 5 muestra una vista básica del uso de recursos (procesos y almacenamiento) por RSE.

RSE es una aplicación Java™, lo que significa que está activo en el entorno z/OS UNIX. Ello permite establecer puertos de forma sencilla a otras plataformas de host y una comunicación directa con el cliente de Developer for System z, que también es una aplicación Java (basada en la infraestructura de Eclipse). Por ello, tener un conocimiento básico de cómo funcionan z/OS UNIX y Java es de gran ayuda para comprender Developer for System z.

En z/OS UNIX un programa se ejecuta en un proceso, identificado por un PID (ID de proceso). Cada programa está activo en su propio proceso, de manera que el hecho de invocar otro programa crea un proceso nuevo. Se hace referencia al proceso que ha iniciado un proceso con un PPID (PID padre), y el proceso nuevo se denomina proceso hijo. El proceso hijo se puede ejecutar en el mismo espacio de direcciones, o bien se puede engendrar (crear) en un espacio de direcciones nuevo. Un proceso nuevo que se ejecuta en el mismo espacio de direcciones puede compararse con la ejecución de un mandato en TSO; mientras que engendrar uno en un espacio de direcciones nuevo es similar a someter un trabajo por lotes.

Tenga un proceso puede tener una sola o varias hebras. En una aplicación de varias hebras (como RSE), las distintas hebras compiten por los recursos del sistema, como si fueran espacios de direcciones separados (con menos sobrecarga).

Al correlacionar esta información de proceso al ejemplo de RSE de la Figura 2 en la página 5, obtenemos este flujo:

1. Cuando se inicia la tarea RSED, esta ejecuta BPXBATSL, que invoca z/OS UNIX y crea un entorno de shell – PID 50331904.
2. En este proceso se ejecuta el script de shell `rsed.sh shell`, que se ejecuta en un proceso independiente (`/bin/sh`) – PID 67109114.
3. El script de shell establece las variables de entorno definidas en `rsed.envvars` y ejecuta Java con los parámetros necesarios para iniciar el daemon RSE – PID 50331949.
4. El daemon RSE generará un shell nuevo en un proceso hijo (RSED8) – PID 307.
5. En este shell, se establecen las variables de entorno definidas en `rsed.envvars` y se ejecuta Java con los parámetros necesarios para iniciar la agrupación de hebras RSE – PID 308.

RSE es capaz de ejecutarse en modalidad de direccionamiento de 31 bits o de 64 bits, lo que resulta en diferentes límites de almacenamiento. En la modalidad de 31 bits, el almacenamiento disponible se limita a 2 GB, mientras que en la modalidad de 64 bits no hay límite, a menos que se especifique en `SYS1.PARMLIB`.

Las aplicaciones Java, tales como RSE, no asignan almacenamiento directamente, sino que utilizan los servicios de gestión de memorias de Java. Estos servicios, como la asignación de almacenamiento, la liberación de almacenamiento, y la recogida de basura, funcionan dentro de los límites del almacenamiento dinámico de Java. El tamaño mínimo y máximo del almacenamiento dinámico se define (ya sea implícita o explícitamente) durante el inicio de Java. Cuando se ejecuta en modalidad de 64 bits, Java intentará asignar el almacenamiento dinámico por encima de la barra de 2 GB, liberando el espacio por debajo de la barra.

Ello implica que obtener el máximo del tamaño de espacio de direcciones disponible es un acto de equilibrio que consiste en definir un tamaño de

almacenamiento dinámico grande y dejar suficiente espacio para que z/OS almacene una cantidad variable de bloques de control del sistema (que depende del número de hebras activas).

Propietarios de tareas

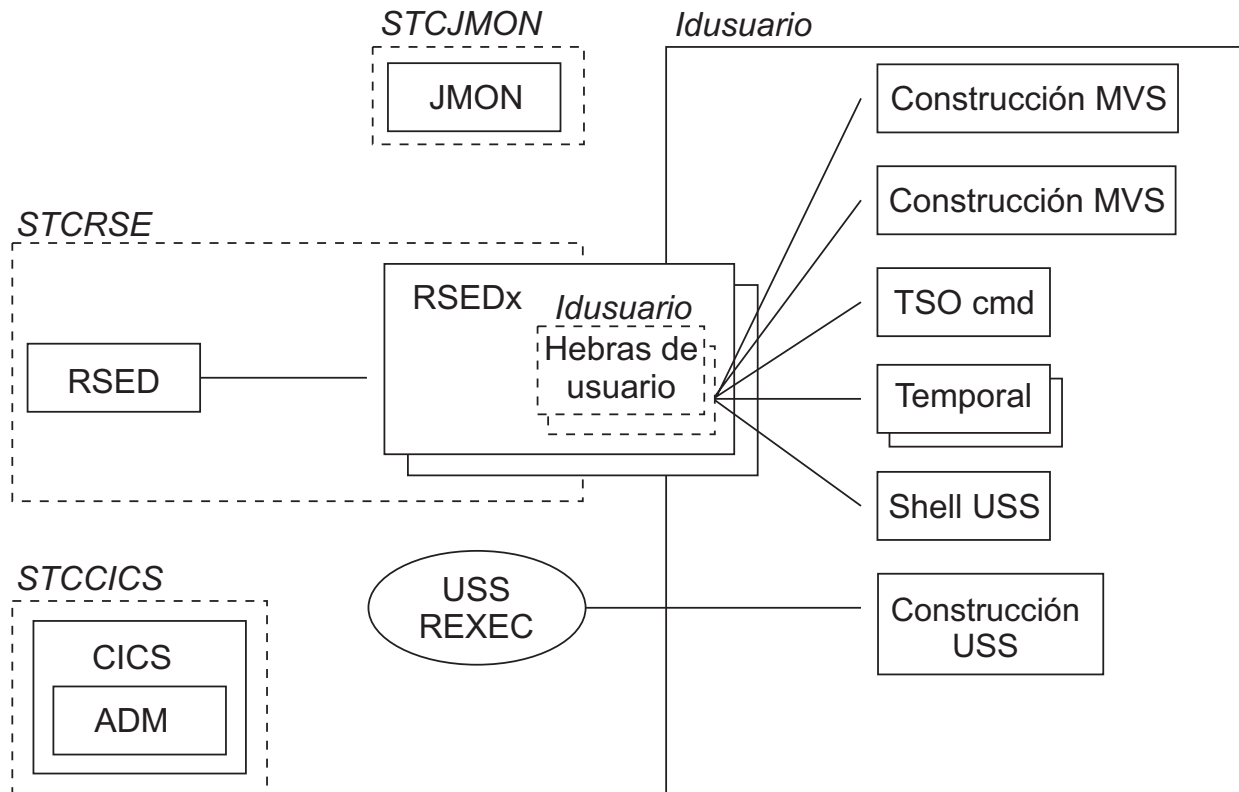


Figura 3. Propietarios de tareas

La Figura 3 muestra una visión general básica del propietario de las credenciales de seguridad utilizadas para varias tareas de Developer for System z.

La propiedad de una tarea se puede dividir en dos secciones. Las tareas iniciadas son propiedad del ID de usuario asignado a la tarea iniciada en el software de seguridad. El resto de tareas con las agrupaciones de hebras RSE (**RSEDx**) como excepción son propiedad del ID de usuario del cliente.

La Figura 3 muestra las tareas iniciadas de Developer for System z (**DBGMGR**, **JMON** y **RSED**) y las tareas iniciadas de ejemplo y los servicios del sistema con los que Developer for System z se comunica. Application Deployment Manager (**ADM**) está activo dentro de una región **CICS**. El código **USS REXEC** representa el servicio z/OS UNIX REXEC (o SSH).

El daemon **RSE** (**RSED**) crea uno o varios espacios de direcciones de agrupaciones de hebras **RSE** (**RSEDx**) para procesar las peticiones de cliente. Cada agrupación de hebras **RSE** soporta varios clientes y es propiedad del mismo usuario que el daemon **RSE**. Cada cliente tiene sus propias hebras dentro de una agrupación de hebras y estas hebras son propiedad del ID de usuario de cliente.

Según las acciones realizadas por el cliente, se pueden iniciar uno o varios espacios de direcciones, todos propiedad del ID de usuario cliente para realizar la acción solicitada. Estos espacios de direcciones pueden estar en un trabajo por lotes MVS, una transacción APPC o un proceso hijo z/OS UNIX. Tenga en cuenta que un proceso hijo z/OS UNIX está activo en un iniciador z/OS UNIX (BPXAS) y que el proceso hijo aparece como una tarea iniciada en JES.

La mayoría de las veces es una hebra de usuario en una agrupación de hebras la que desencadena la creación de estos espacios de direcciones, ya sea directamente o a través de servicios del sistema como por ejemplo ISPF. Sin embargo, el espacio de direcciones también lo puede crear un tercero. Por ejemplo, z/OS UNIX REXEC o SSH se invocan al iniciar construcciones en z/OS UNIX.

Los espacios de direcciones específicos del usuario terminan cuando finaliza la tarea o cuando caduca el temporizador de inactividad. Las tareas iniciadas permanecen activas. Los espacios de direcciones que aparecen en la lista de la Figura 3 en la página 7 permanecen en el sistema lo suficiente para ser visibles. Sin embargo, debe tener en cuenta que debido al diseño de z/OS UNIX, también hay varios espacios de direcciones temporales de vida breve.

Flujo de conexión

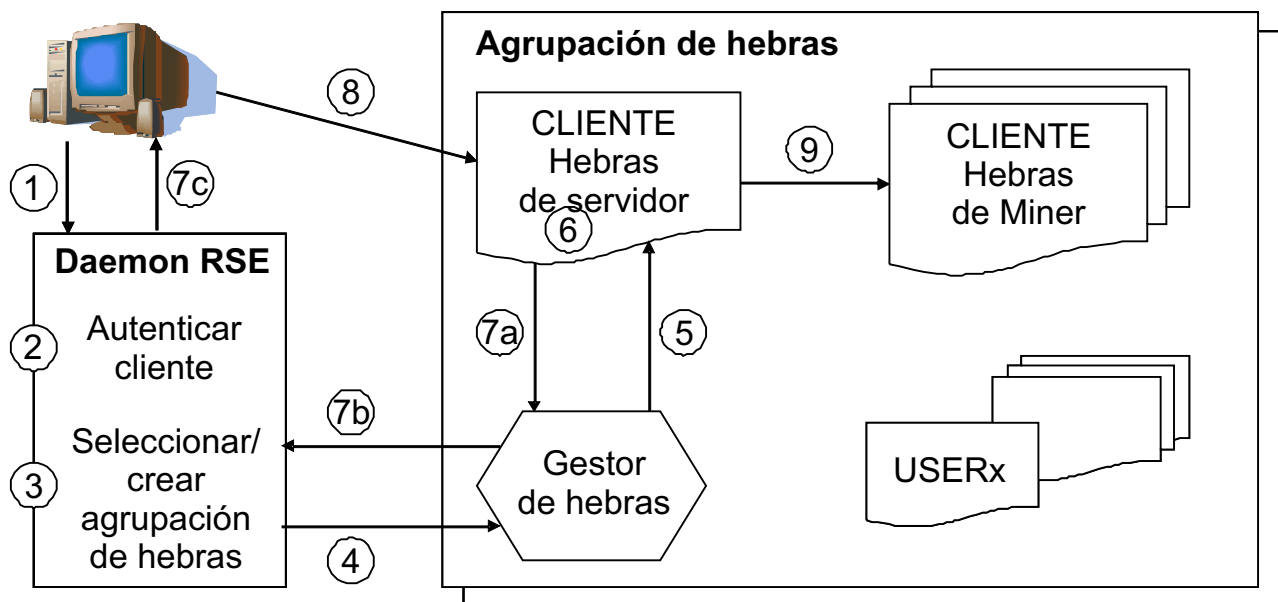


Figura 4. Flujo de conexión

La Figura 4 muestra una visión general esquemática de cómo un cliente se conecta al host mediante Developer for System z. También explica brevemente cómo se utilizan PassTickets.

1. El cliente inicia sesión en el daemon (puerto 4035).
2. El daemon RSE autentica el cliente mediante las credenciales presentadas por el éste.
3. El daemon RSE selecciona una agrupación de hebras existente o bien inicia una agrupación en caso de que todas estén completas.
4. El daemon RSE pasa el ID de usuario del cliente a la agrupación de hebras.

5. La agrupación de hebras crea una hebra de servidor RSE específica del cliente, utilizando el ID de usuario del cliente y un PassTicket para la autenticación.
6. La hebra de servidor de cliente se enlaza a un puerto para la futura comunicación con el cliente.
7. La hebra de servidor de cliente devuelve el número de puerto para que el cliente se conecte.
8. El cliente se desconecta del daemon RSE y se conecta al número de puerto proporcionado.
9. La hebra de servidor de cliente inicia otras hebras específicas del cliente (extractores), utilizando el ID de usuario del cliente y un PassTicket para la autenticación. Estas hebras proporcionan los servicios específicos del cliente que el cliente requiere.

La descripción anterior muestra el diseño orientado a hebras de RSE. En lugar de iniciar un espacio de direcciones por usuario, un único espacio de direcciones de agrupaciones de hebras proporciona servicio a varios usuarios. Dentro de la agrupación de hebras, cada extractor (un servicio específico del usuario) se activa en su propia hebra con el contexto de seguridad del usuario que tiene asignado, asegurando así una configuración segura. Este diseño acomoda un mayor número de usuarios con un uso limitado de recursos, pero conlleva que cada cliente utilice varias hebras (17 o más, según las tareas realizadas).

Desde un punto de vista de red, Developer for system z actúa de forma similar al FTP en modalidad pasiva. El cliente se conecta a un punto focal (daemon RSE), descarta la conexión y se reconecta a un número de puerto proporcionado por el punto focal. La siguiente lógica controla la selección del puerto que se utiliza para la segunda conexión:

1. Si el cliente ha especificado un número de puerto distinto de cero en la pestaña de propiedades de subsistema, el servidor RSE utilizará ese número de puerto para el enlace. Si este puerto no está disponible, la conexión falla.
2. Si se especifica `_RSE_PORTRANGE` en `rsed.envvars`, el servidor RSE se enlazará a un puerto de este rango. Si no hay ningún puerto disponible, la conexión falla. El servidor RSE no necesita el puerto exclusivamente para la duración de la conexión de cliente. Está sólo en el lapso de tiempo entre el enlace (servidor) y la conexión (cliente) que ningún otro servidor RSE puede enlazar al puerto. Esto significa que la mayoría de las conexiones utilizarán el primer puerto del rango, y el resto del rango será un almacenamiento intermedio en caso de varios inicios de sesión simultáneos.
3. Si no establece ninguna limitación, el servidor RSE se enlazará al puerto 0. El resultado es que TCP/IP elige el número de puerto.

El uso de PassTickets para todos los servicios de z/OS que requieren autenticación permite Developer for System z invocar estos servicios sin tener que almacenar la contraseña ni solicitarle al usuario continuamente que la introduzca. El uso de PassTickets para todos los servicios de z/OS también permite métodos de autenticación alternativos durante el inicio de sesión, como las contraseñas para una sola vez y los certificados X.509.

Depurador integrado

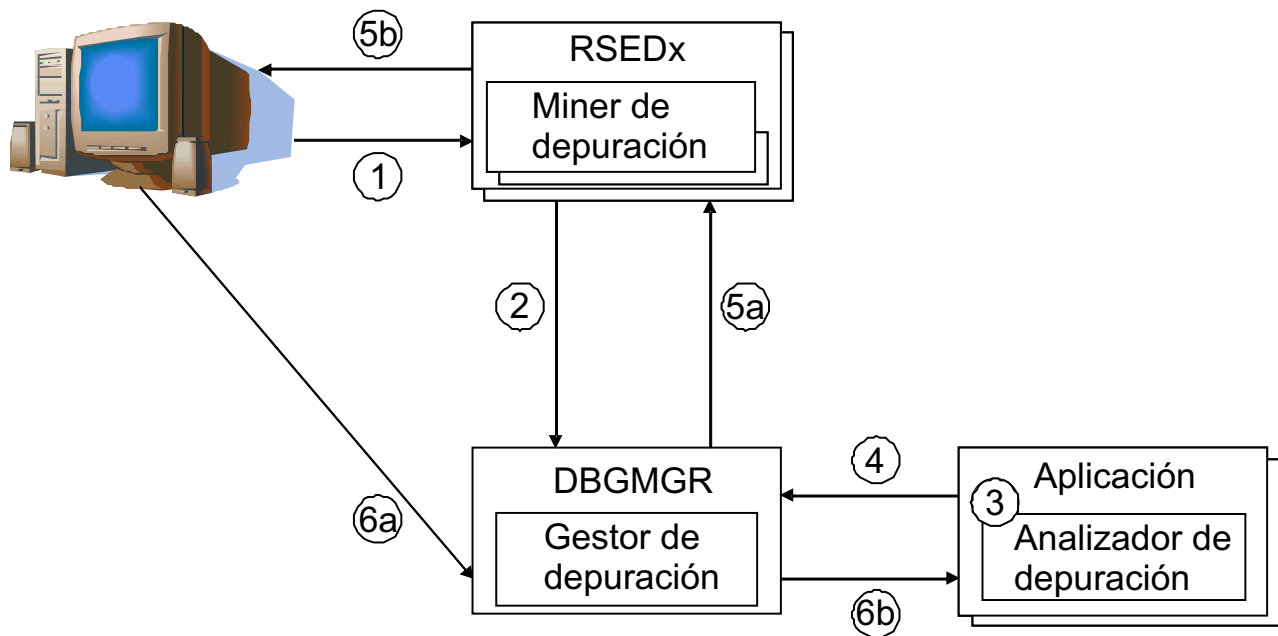


Figura 5. Depurador integrado

El depurador integrado se utiliza para depurar varias aplicaciones. La imagen 5 muestra una descripción general esquemática que ilustra cómo un cliente Developer for System z puede depurar una aplicación.

1. El cliente se conecta con el host, utilizando el inicio de sesión de host Developer for System z habitual.
2. Como parte del inicio de sesión, un extractor de depuración registrará al usuario con el gestor de depuración, que está activo en la tarea iniciada DBGMGR.
3. Cuando se inicia una aplicación con un indicador que se debe depurar, Language Environment (LE) invoca la sonda de depuración.
4. La sonda de depuración se registrará con el gestor de depuración.
5. Utilizando el extractor de depuración, el gestor de depuración notificará al cliente Developer for System z del usuario que recibirá esta sesión de depuración. Si el usuario no está registrado en estos momentos, la sesión de depuración pasa a estado inactivo, a la espera de que el usuario se registre con el gestor de depuración.

6. El motor de depuración del cliente se pone en contacto con el gestor de depuración, que, a su vez, se encarga de transmitir los datos entre el motor de depuración y la sonda de depuración.

CARMA

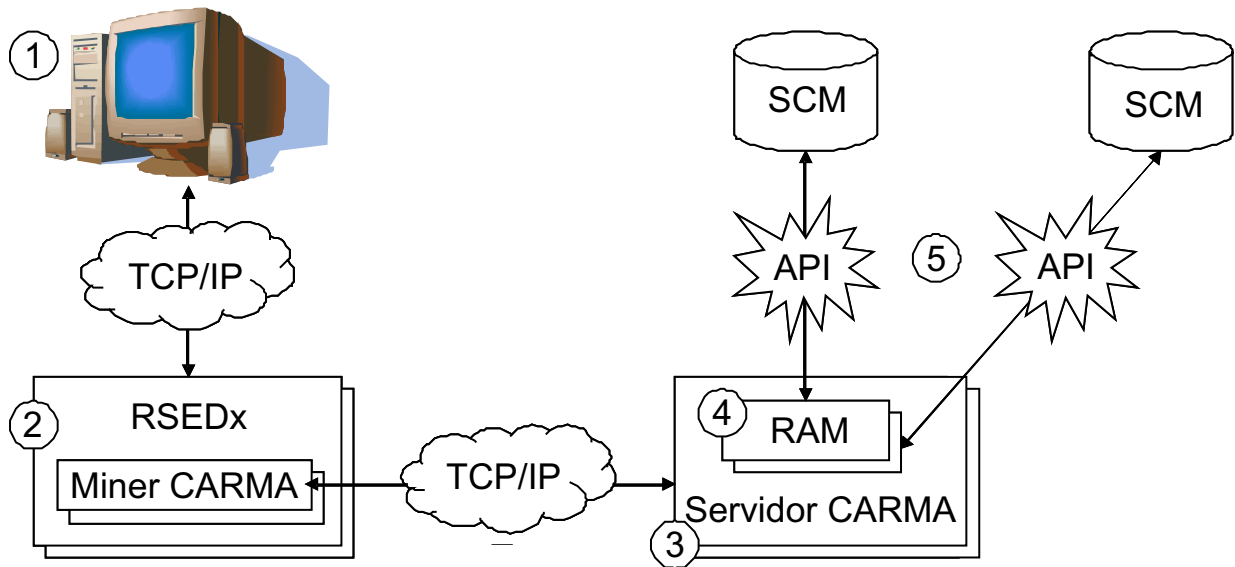


Figura 6. Flujo de CARMA

CARMA (Common Access Repository Manager) se utiliza para acceder a un SCM (Software Configuration Manager) basado en host, como por ejemplo, CA Endevor® SCM. La Figura 6 muestra una Visión general esquematizada de cómo un cliente de Developer for System z puede acceder a cualquier Software Configuration Manager (SCM) basado en host con soporte.

1. El cliente tiene un plug-in de Common Access Repository Manager (CARMA).
2. El plug-in CARMA se comunica con el extractor CARMA, activo como hebra específica de usuario en la agrupación de hebras RSE (RSEDx). Esta comunicación se realiza a través de la conexión RSE existente.
3. Cuando el cliente solicita acceso a un SCM, el extractor CARMA se enlazará a un puerto TCP/IP e iniciará un servidor CARMA específico de usuario con el número de puerto como argumento inicial. El servidor CARMA se conectará a este puerto y utilizará esta vía de acceso para la comunicación con el cliente. Tenga en cuenta que los SCM basados en host espera que los espacios de direcciones de usuario único accedan a sus servicios, lo que requiere que CARMA inicie un servidor de CARMA por usuario. No es posible crear un servidor único que dé soporte a varios usuarios.
4. El servidor CARMA cargará el Gestor de acceso a repositorios (RAM) que soporta el SCM solicitado.
5. El RAM se ocupa de los detalles técnicos de interactuar con el SCM específico y presenta una interfaz común al cliente.

Archivos de configuración de CARMA

Developer for System z permite utilizar varios métodos para iniciar un servidor CARMA. Cada método tiene ventajas e inconvenientes. Developer for System z

proporciona también varios Gestores de acceso a repositorio (RAM) que se pueden dividir en dos grupos, los RAM de producción y los RAM de ejemplo. Hay varias combinaciones de RAM y métodos de inicio de servidor como configuración preconfigurada.

Todos los métodos de inicio de servidor comparten un archivo de configuración común, `CRASRV.properties`, que (entre otras cosas) especifica qué método de inicio se utilizará.

CRASTART

El método "CRASTART" inicia el servidor CARMA como subtask dentro de RSE. Ofrece una configuración muy flexible mediante la utilización de un archivo de configuración independiente que define las asignaciones de conjunto de datos e invocaciones de programa necesarias para iniciar un servidor CARMA. Este método ofrece el mejor rendimiento y utiliza la menor cantidad de recursos, pero requiere que el módulo CRASTART se encuentre en LPA.

CRASTART utiliza las definiciones de `crastart*.conf` para crear un entorno válido para ejecutar mandatos TSO e ISPF por lotes. Developer for System z utiliza este entorno para ejecutar el servidor CARMA, CRASERV. Developer for System z proporciona varios archivos `crastart*.conf`, cada uno preconfigurado para un RAM específico.

Sometimiento por lotes

El método "sometimiento por lotes" inicia un servidor CARMA sometiendo un trabajo. Este es el método predeterminado utilizado en los archivos de configuración de ejemplo suministrados. La ventaja de este método es que puede accederse fácilmente a los registros de CARMA en la salida del trabajo. También permite utilizar JCL de servidor personalizado para cada desarrollador, cuyo mantenimiento realiza el propio desarrollador. Sin embargo, este método utiliza un iniciador de JES por cada desarrollador que inicia un servidor CARMA.

RSE invoca CLIST CRASUB*, que a su vez somete un JCL incorporado JCL para crear un entorno válido para ejecutar mandatos TSO e ISPF de proceso por lotes. Developer for System z utiliza este entorno para ejecutar el servidor CARMA, CRASERV. Developer for System z proporciona varios miembros CRASUB*, cada uno preconfigurado para un RAM específico.

Propietario de bloqueo de conjunto de datos

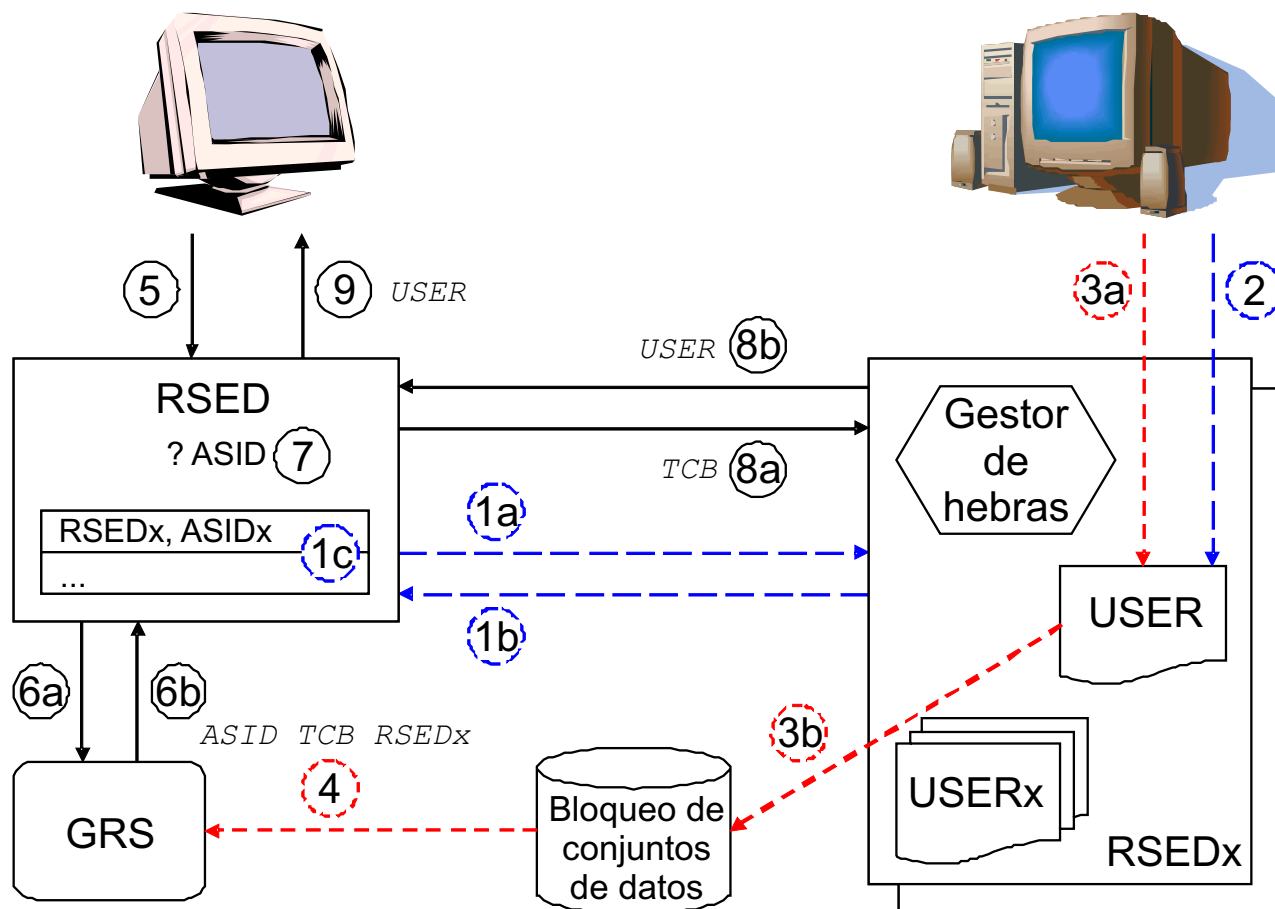


Figura 7. Flujo de determinación de puesta en cola de conjunto de datos

La Figura 7 muestra una visión general esquemática de cómo el daemon RSE determina qué cliente de Developer for System z es propietario de un bloqueo de conjunto de datos.

1. El daemon RSE (RSED) crea una agrupación de hebras (RSEDx). Para confirmar la finalización del inicio, la agrupación de hebras informa del Identificador de espacio de direcciones (ASID) al daemon RSE, que lo almacena en el bloque de control creado para hacer un seguimiento de esta agrupación de hebras.
2. El cliente inicia sesión, lo que crea una hebra de servidor RSE específica del usuario (USER) dentro de una agrupación de hebras (RSEDx). Cada hebra tiene un identificador de Bloque de control de tareas (TCB) exclusivo.
3. El cliente abre un conjunto de datos en edición, que especifica al servidor RSE que obtenga un bloqueo exclusivo (puesta en cola) en el conjunto de datos.
4. El sistema registra el ASID, TCB y el nombre de tarea (RSEDx) del solicitante como parte del proceso de puesta en cola. Esta información se almacena en las colas de serialización de recursos globales (GRS).
5. Un operador consulta el daemon RSE para conocer el estado de bloqueo del conjunto de datos.
6. El daemon RSE explora las colas de GRS para saber si el conjunto de datos está bloqueado y recupera el ASID, TCB y el nombre de tarea del propietario del bloqueo.

7. El ASID recuperado se compara con el ASID de diferentes agrupaciones de hebras.
8. El daemon RSE pide a la agrupación de hebras que posee el ASID, que determine qué usuario es propietario del TCB.
9. El ID de usuario del cliente relacionado se devuelve al solicitante cuando se encuentra una coincidencia. De lo contrario, el nombre de tarea recuperado de la GRS se devuelve.

Con una configuración de Developer for System z con un solo servidor, donde hay varios usuarios asignados a un único espacio de direcciones de agrupaciones de hebras, z/OS pierde la capacidad de rastrear quién es propietario de un bloqueo en un conjunto de datos o un miembro con el mandato del operador **DISPLAY GRS,RES=(*,dataset*)**. Los mandatos del sistema se detienen a nivel de espacio de dirección, que es la agrupación de hebras.

Para solucionar este problema, Developer for System z proporciona el mandato de operador **MODIFY rsed APPL=DISPLAY OWNER,DATASET=dataset**, tal como se describe en "Mandatos de operador" en la *Guía de configuración de host* SC11-3660 (SC23-7658). El mandato de operador puede resolver todos los bloqueos de miembros y conjunto de datos realizados por usuarios de RSE, así como los bloqueos realizados por otros productos, como por ejemplo ISPF.

Liberar un bloqueo

En circunstancias normales, un conjunto de datos o un miembro está bloqueado cuando un cliente lo abre en modalidad de edición, y este se libera cuando el cliente cierra la sesión de edición.

Algunas condiciones de error pueden provocar que este mecanismo no funcione tal como debe. En este caso, se puede cancelar el usuario que está manteniendo el bloqueo mediante el mandato de operador **modify cancel** de RSE, como se describe en "Mandatos de operador" en la publicación *Guía de configuración de host* (SC11-3660). Los bloqueos de conjuntos de datos activos de este usuario se liberan durante el proceso.

Estructura de directorios de z/OS UNIX

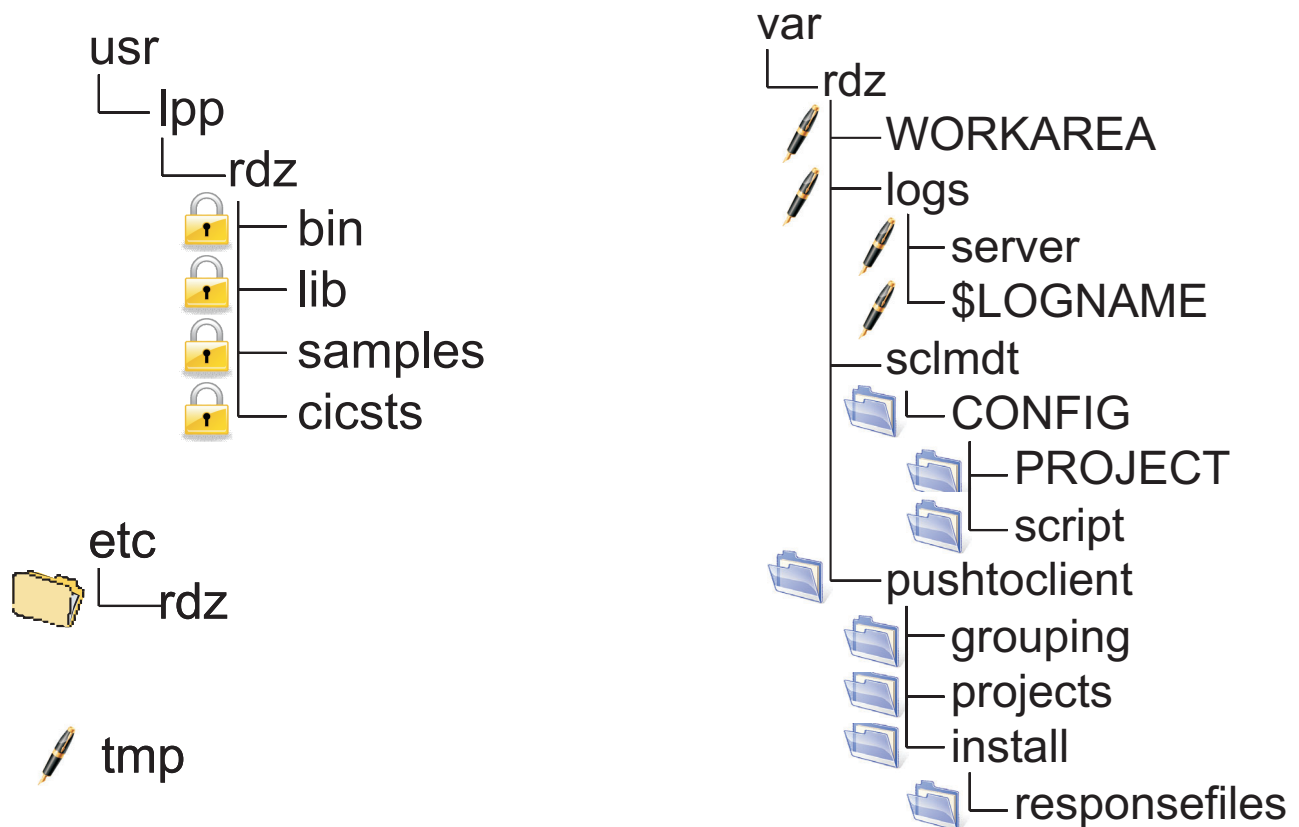


Figura 8. Estructura de directorios de z/OS UNIX

La Figura 8 muestra una visión general de los directorios de z/OS UNIX utilizados por Developer for System z. La lista siguiente describe cada directorio tocado por Developer for System z, cómo se puede cambiar la ubicación y quién mantiene los datos que contiene.

- /usr/lpp/rdz/ es la vía de acceso de raíz para el código de producto de Developer for System z. La ubicación real se especifica en la tarea iniciada RSED (variable HOME). SMP/E mantiene los archivos que contienen.
- /etc/rdz/ contiene los archivos de configuración relacionados con RSE y el extractor. La ubicación real se especifica en la tarea iniciada RSED (variable CNFG). El programador del sistema mantiene los archivos que contienen.
- La Pasarela de cliente TSO/ISPF de ISPF y varios extractores utilizan /tmp/ para almacenar datos temporales. Algunos IVP almacenan aquí su salida. Los archivos que contienen se mantienen mediante ISPF, los extractores y los IVP. La ubicación puede personalizarse con la variable TMPDIR en rsed.envvars. También es la ubicación predeterminada para los archivos de vuelco Java, que pueden personalizarse con la variable _CEE_DUMPTARG de rsed.envvars.

Nota: /tmp/ requiere la máscara de bit de permiso 777 para permitir a cada cliente crear archivos temporales.

- SCLMDT y la Pasarela de cliente TSO/ISPF de ISPF utilizan /var/rdz/WORKAREA/ para transferir datos entre z/OS UNIX y los espacios de direcciones basados en MVS. La ubicación real se especifica en rsed.envvars (variable CGI_ISPWORK). ISPF y SCLMDT mantienen los archivos que contienen.

Nota: /var/rdz/WORKAREA/ requiere la máscara de bit de permiso 777 para permitir a cada cliente crear archivos temporales.

- /var/rdz/logs/server/ contiene los registros del daemon RSE y los servidores de agrupaciones de hebras RSE. La ubicación real se especifica en rsed.envvars (variable daemon.log). RSE mantiene los archivos que contienen.
- /var/rdz/logs/\$LOGNAME/ contiene los registros específicos del usuario del servidor RSE y los extractores. La ubicación real se especifica en rsed.envvars (variables user.log y DSTORE_LOG_DIRECTORY). RSE y los extractores mantienen los archivos que contienen.

Nota: /var/rdz/logs/ requiere la máscara de bit de permiso 777 para permitir a cada cliente crear directorio \$LOGNAME y almacenar los archivos de registro específicos del cliente.

- /var/rdz/sclmdt/CONFIG/ contiene los archivos de configuración SCLMDT generales. La ubicación real se especifica en rsed.envvars (variable SCLMDT_CONF_HOME). El administrador de SCLM mantiene los archivos que contienen.
- /var/rdz/sclmdt/CONFIG/PROJECT/ contiene los archivos de configuración de proyectos SCLMDT. La ubicación real se especifica en rsed.envvars (variable SCLMDT_CONF_HOME). El administrador de SCLM mantiene los archivos que contienen.
- /var/rdz/sclmdt/CONFIG/script/ contiene los scripts relacionados con SCLMDT que pueden utilizar otros productos. La ubicación actual no se especifica en ningún sitio. El administrador de SCLM mantiene los archivos que contienen.
- /var/rdz/pushtoclient/ contiene los archivos de configuración del producto del cliente, la información de actualización de cliente y la información de proyectos basados en host que se pasan al cliente tras la conexión al host. La ubicación real se especifica en pushtoclient.properties (variable pushtoclient.folder). Los archivos que contiene los mantiene un administrador de cliente de Developer for System z.
- /var/rdz/pushtoclient/grouping/ contiene archivos de configuración de cliente específicos del grupo, información de actualización de producto de cliente e información de proyectos basados en host que se pasan al cliente tras la conexión al host. La ubicación real se especifica en pushtoclient.properties (variable pushtoclient.folder con la adición del sufijo /grouping). Los archivos que contiene los mantiene un administrador de cliente de Developer for System z.
- /var/rdz/pushtoclient/projects/ contiene los archivos de definición de proyectos basados en host. La ubicación real se especifica en el archivo /var/rdz/pushtoclient/keymapping.xml, de cuya creación y mantenimiento se ocupa el administrador del cliente Developer for System z. El gestor de proyectos o el desarrollador principal mantiene los archivos que contienen.
- /var/rdz/pushtoclient/install/ tiene los archivos de configuración utilizados para actualizar la versión del producto del cliente al conectarse al host. La ubicación real se especifica en el archivo /var/rdz/pushtoclient/keymapping.xml, de cuya creación y mantenimiento se ocupa el administrador del cliente Developer for System z. El administrador del cliente mantiene los archivos que contienen.
- /var/rdz/pushtoclient/install/responsefiles/ tiene los archivos de configuración utilizados para actualizar la versión del producto del cliente al conectarse al host. La ubicación real se especifica en el archivo /var/rdz/pushtoclient/keymapping.xml, de cuya creación y mantenimiento se

ocupa el administración del cliente Developer for System z. El administrador del cliente mantiene los archivos que contienen.

Privilegios de actualización para usuarios no administradores del sistema

Los datos de `/var/rdz/pushtoclient/` los mantienen los usuarios no administradores del sistema, que es posible que no dispongan de demasiados privilegios de actualización en z/OS UNIX. Por consiguiente, es importante comprender cómo z/OS UNIX establece los permisos de acceso durante la creación de archivos a fin de garantizar que la instalación funcione y sea segura.

Los estándares de UNIX indican que pueden establecerse permisos para tres tipos de usuario: propietario, grupo y otro. Pueden establecerse permisos de lectura, escritura y ejecución de forma individualizada para cada tipo.

z/OS UNIX establece el UID (ID de usuario) y el GID (ID de grupo) en los siguientes valores cuando se crea un archivo:

- El UID se establece en el UID efectivo de la hebra de creación.
- El GID se establece en el GID del directorio propietario. Si el perfil de seguridad `FILE.GROUPOWNER.SETGID` se define en la clase `UNIXPRIV`, se utiliza, de forma predeterminada, en el GID efectivo de la hebra de creación. Para obtener más detalles, consulte la publicación *UNIX System Services Planning* (GA22-7800).

Cada sitio puede establecer su propia máscara de permisos de acceso; no obstante, una máscara común concede permiso de lectura y escritura al tipo propietario y permiso de lectura a los tipos grupo y otro.

Los datos de `/var/rdz/pushtoclient/` se crean utilizando la máscara de permisos de acceso definida en la directiva `file.permission` de `pushtoclient.properties`. El valor predeterminado concede permiso de lectura y de escritura para los tipos propietario y grupo, y permiso de lectura para el tipo otro. Todos tienen permiso de ejecución. Los permisos de acceso finales deberían permitir el acceso de lectura y ejecución a todos, y el acceso de escritura para los administradores de clientes de Developer for System z que mantienen los datos.

Los datos de `/var/rdz/pushtoclient/projects/` se crean utilizando una máscara de permisos de acceso no específico. Los permisos de acceso finales deberían permitir acceso de lectura a todos, y permiso de escritura para los gestores de proyectos que mantienen los datos.

Mandatos de seguridad de gran utilidad

Para garantizar que un grupo de gestores de proyectos o administradores de clientes de Developer for System z pueden gestionar los datos de estos directorios, es posible que el administrador de seguridad deba crear un grupo con un segmento OMVS válido para los mismos. Este grupo es preferiblemente el grupo predeterminado para los ID de usuario implicados. Consulte la publicación *Security Server RACF Command Language Reference* (SA22-7687) para obtener más información sobre los siguientes mandatos RACF de ejemplo:

```
ADDGROUP RDZPROJ OMVS(GID(1200))
CONNECT IBMUSER GROUP(RDZPROJ)
ALTUSER IBMUSER DFLTGRP(RDZPROJ)
```

Mandatos útiles de z/OS UNIX

consulte la publicación *UNIX System Services Command Reference* (SA22-7802) para obtener más información sobre los siguientes mandatos de ejemplo de z/OS UNIX:

- Utilice el siguiente mandato **ls** de z/OS UNIX para visualizar todos los archivos de un directorio.

```
ls -lR /var/rdz/pushtoclient/
```

- Utilice el siguiente mandato **chown** de z/OS UNIX para cambiar el propietario de un directorio y todos sus archivos.

```
chown -R IBMUSER /var/rdz/pushtoclient/
```

- Utilice el mandato **chgrp** de z/OS UNIX siguiente para asignar el grupo al directorio y todos los archivos incluidos.

```
chgrp -R RDZPROJ /var/rdz/pushtoclient/
```

- Utilice el siguiente mandato **chmod** de z/OS UNIX para conceder al propietario y al grupo permiso de escritura para el directorio y todos sus archivos. El tipo Otro tiene el permiso de lectura. Todos tienen permiso de ejecución.

```
chmod -R 775 /var/rdz/pushtoclient/
```

Configuración de ejemplo

En el siguiente escenario, todos los gestores de proyectos de desarrollo, un equipo de tres personas, tienen asignada la tarea de ser administrador de un cliente de Developer for System z.

El administrador de seguridad ya ha asignado al equipo un grupo predeterminado (RDZPROJ) con el ID de grupo exclusivo (1200). Sus ID de usuario no tienen privilegios especiales (como UID 0) en z/OS UNIX. El administrador de seguridad no ha definido el perfil FILE.GROUPOWNER.SETGID, por lo que, al crear nuevos archivos, z/OS UNIX utilizará el ID de grupo del directorio. El programador de sistemas ha utilizado el ID de usuario IBMUSER (con el UID 0 y el grupo predeterminado SYS1) para crear el directorio /var/rdz/pushtoclient.

1. El programador de sistemas limita los permisos de escritura para /var/rdz/pushtoclient al propietario y al grupo:

```
# chmod 775 /var/rdz/pushtoclient
# ls -ld /var/rdz/pushtoclient
drwxrwxr-x  2 IBMUSER SYS1
/var/rdz/pushtoclient
```

Nota: El trabajo FEKSETUP utilizado durante la instalación personalizada se ocupa de este paso.

2. El programador del sistema establece el grupo RDZPROJ como propietario:

```
# chgrp RDZPROJ /var/rdz/pushtoclient
# ls -ld /var/rdz/pushtoclient
drwxrwxr-x  2 IBMUSER RDZPROJ
/var/rdz/pushtoclient
```

DE este modo concluye la configuración necesaria para limitar los permisos de escritura de /var/rdz/pushtoclient al programador del sistema (IBMUSER) y a los gestores de proyectos (RDZPROJ).

Capítulo 2. Consideraciones relativas a la seguridad

Developer for System z proporciona a los usuarios acceso al sistema central en una estación de trabajo que no es del sistema central. Algunos aspectos importantes de la configuración del producto son: validar las solicitudes de conexión, proporcionar una comunicación segura entre el host y la estación de trabajo, y autorizar y auditar la actividad.

Los mecanismos de seguridad utilizados por los servidores y servicios de Developer for System z se basan en que los conjuntos de datos y los sistemas de archivos en los que residen sean seguros. Esto implica que sólo los administradores del sistema que sean de confianza puedan actualizar las bibliotecas de programa y los archivos de configuración.

En este capítulo se tratan estos temas:

- “Métodos de autenticación” en la página 20
- “Seguridad de conexión” en la página 21
- “Uso de PassTickets” en la página 23
- “Registro de auditoría” en la página 24
- “Seguridad de JES” en la página 26
- “Comunicación cifrada con SSL/TLS” en la página 30
- “Autenticación de cliente mediante certificados X.509” en la página 32
- “Comprobación de puerto de entrada (POE)” en la página 36
- “Alterar las funciones de cliente” en la página 36
- “Grupos de desarrollador Envío a cliente” en la página 37
- “Seguridad de archivo de registro” en la página 39
- “Seguridad de depuración” en la página 42
- “Seguridad de CICSTS” en la página 42
- “Seguridad de SCLM” en la página 43
- “Información variada” en la página 43
- “Archivos de configuración de Developer for System z” en la página 44
- “Definiciones de seguridad” en la página 47

Nota: El Explorador de sistemas remotos (RSE), que proporciona servicios del núcleo como los de conectar el cliente al host, está formado por 2 entidades lógicas.

- El daemon RSE, que gestiona la configuración de conexiones y se inicia como tarea iniciada o como trabajo de usuario de larga ejecución.
- El servidor RSE, que maneja las solicitudes de clientes individuales y se inicia como una hebra en uno o varios procesos hijo del daemon RSE.

Consulte Capítulo 1, “Comprender Developer for System z”, en la página 3 para conocer los conceptos de diseño básicos de Developer for System z.

Métodos de autenticación

Developer for System z admite varias formas de autenticar un ID de usuario facilitado por un cliente durante la conexión.

- ID de usuario y contraseña
- ID de usuario y contraseña para una sola vez
- ID de usuario y frase de contraseña
- Certificado X.509

Nota: Los datos de autenticación facilitados por el cliente solamente se utilizan una vez, durante la configuración inicial de la conexión. Una vez se autentica un ID de usuario, este y los PassTickets autogenerados se utilizan para todas las acciones que requieren autenticación.

ID de usuario y contraseña

El cliente facilita un ID de usuario y una contraseña coincidente durante la conexión. El ID de usuario y la contraseña se utilizan para autenticar al usuario en el producto de seguridad.

ID de usuario y contraseña para una sola vez

Basándose en una única señal, un producto externo puede generar una contraseña para una sola vez. Las contraseñas para una sola vez mejoran la configuración de seguridad, ya que la señal exclusiva no se puede copiar y utilizar sin el conocimiento del usuario, y una contraseña interceptada no sirve para nada porque solamente es válida una vez.

El cliente facilita un ID de usuario y una contraseña para una sola vez durante la conexión que se utiliza para autenticar el ID de usuario con la salida de seguridad proporcionada por un programa externo. Se espera que esta salida de seguridad ignore los PassTickets utilizados para satisfacer las solicitudes de autenticación durante el proceso normal. Su software de seguridad debe procesar los PassTickets.

ID de usuario y frase de contraseña

El cliente facilita un ID de usuario y una frase de contraseña coincidente durante la conexión. El ID de usuario y la frase de contraseña se utilizan para autenticar al usuario en el producto de seguridad.

Certificado X.509

Un tercer puede proporcionar uno o varios certificados X.509 que se pueden utilizar en la autenticación de un usuario. Cuando están almacenados en dispositivos seguros, los certificados X.509 combinan una configuración segura con un uso sencillo para el usuario (no son necesarios ni ID de usuario ni contraseña).

Durante la conexión, el cliente facilita un certificado seleccionado y, opcionalmente, una extensión seleccionada, que se utiliza para autenticar el ID de usuario con su producto de seguridad.

Nota: Este método de autenticación solamente está soportado por el método de conexión del daemon RSE y la comunicación SSL (capa de sockets seguros) debe estar habilitada.

Autenticación del supervisor de trabajos JES

El daemon RSE (o REXEC/SSH) realiza la autenticación de clientes como parte de la solicitud de conexión del cliente. Una vez se autentica el usuario, se utilizan PassTickets autogenerados para las solicitudes de autenticación que se realicen en el futuro, incluido el inicio de sesión automático en el Supervisor de trabajos JES.

Para que el Supervisor de trabajos JES valide el ID de usuario y el PassTicket presentado por RSE, el Supervisor de trabajos JES debe poder evaluar el PassTicket. Ello implica:

- El módulo de carga FEJMON, ubicado, por omisión, en la biblioteca de carga FEK.SFEKAUTH, debe estar autorizado APF.
- Tanto RSE como el Supervisor de trabajos JES deben utilizar el mismo ID de aplicación (APPLID). Por omisión, ambos servidores utilizan FEKAPPL como APPLID, pero esto puede verse modificado por la directiva APPLID de rsed.envvars para RSE y de FEJCNFG para el Supervisor de trabajos JES.

Nota: Los clientes anteriores (versión 7.0 y anteriores) se comunican directamente con el supervisor de trabajos JES. Para estas conexiones, solamente está soportado el método de autenticación de ID de usuario y contraseña.

Autenticación del gestor de depuración

El daemon RSE (o REXEC/SSH) realiza la autenticación de clientes como parte de la solicitud de conexión del cliente. Una vez se autentica el usuario, se utilizan PassTickets autogenerados para las solicitudes de autenticación que se realicen en el futuro, incluido el inicio de sesión automático en el gestor de depuración.

Para que el gestor de depuración valide el ID de usuario y el PassTicket presentado por RSE, el gestor de depuración debe poder evaluar el PassTicket. Por tanto, el módulo de carga AQEZPCM, ubicado de forma predeterminada en la biblioteca de carga FEK.SFEKAUTH, debe estar autorizado por APF.

Cuando un motor de depuración basado en un cliente se conecta al gestor de depuración, debe presentar una señal de seguridad válida para la autenticación.

Seguridad de conexión

EL servidor RSE, que controla toda la comunicación entre el cliente y la mayoría de los servicios de Developer for System z, da soporte a varios niveles de seguridad de comunicaciones:

- La comunicación externa (cliente-host) puede limitarse a puertos especificados. Esta característica está inhabilitada por omisión.
- La comunicación externa (cliente-host) puede cifrarse mediante SSL o TLS. Esta característica está inhabilitada por omisión.
- Puede utilizarse la comprobación de puerto de entrada (POE) para permitir el acceso de host sólo a las direcciones TCP/IP de confianza. Esta característica está inhabilitada por omisión.

Algunos servicios de Developer for System z opcionales utilizan un vía de acceso para las comunicaciones externas (cliente-host) independiente:

- La comunicación del depurador integrado se puede cifrar utilizando TLS.
- La comunicación del gestor de despliegue de aplicaciones se puede cifrar utilizando SSL cuando se emplea la interfaz de Web Services.

Developer for System z depende de productos de terceros, como el servidor TN3270, para proporcionar algunos servicios. Consulte la documentación de producto correspondiente para conocer las opciones de seguridad de conexión.

Limitar la comunicación externa a puertos especificados

El programador del sistema puede especificar los puertos en los que el servidor RSE se puede comunicar con el cliente. Por omisión, se utiliza cualquier puerto disponible. Este rango de puertos no tiene conexión con el puerto del daemon RSE.

Para ayudarle a comprender la utilización de los puertos, se proporciona esta descripción corta del proceso de conexión del RSE:

1. El cliente se conecta al puerto del host 4035, el daemon RSE.
2. El puerto del daemon RSE crea una hebra de servidor RSE.
3. El servidor RSE abre un puerto de host para que el cliente se conecte. La selección de este puerto la puede configurar el usuario, ya sea en el cliente, en la pestaña de propiedades de subsistema (método no recomendado) o mediante la definición de `_RSE_PORTRANGE` en el archivo `rsed.envvars`.
4. El daemon RSE devuelve el número de puerto al cliente.
5. El cliente se conecta al puerto del host.

Nota:

- El proceso es similar para el método de conexión alternativo (opcional) mediante REXEC/SSH, que se describe en "(Opcional) Uso de REXEC (o SSH)", en *Guía de configuración de host* (SC11-3660).
- El puerto utilizado por el depurador integrado y el Gestor de despliegue de aplicaciones para comunicación externa está definido en la configuración del servicio.

Cifrado de comunicaciones utilizando SSL o TLS

Todas las corrientes de datos externas de Developer for System z que pasan a través de RSE pueden cifrarse mediante SSL (Capa de sockets seguros) o TLS (Seguridad de capa de transporte). El uso de comunicación cifrada está controlado por los valores del archivo de configuración `ssl.properties`, tal como se describe en la sección "Comunicación cifrada con SSL/TLS" en la página 30. La variable `DSTORE_SSL_ALGORITHM` de la directiva `_RSE_JAVA_OPTS` de `rsed.envvars` permite elegir entre SSL y su eventor TLS como método de cifrado, tal como se indica en la sección "Definición de parámetros de inicio de Java adicionales con `_RSE_JAVA_OPTS`" en la *Guía de configuración de host* (SC11-3660).

El motor del depurador integrado del cliente se conecta con el Gestor de depuración en el host. El uso de SS o TLS se controla mediante una directiva Application Transparent TLS (AT-TLS).

El Emulador de conexión de host del cliente se conecta a un servidor TN3270 del host. La utilización de SSL está controlada por TN3270, tal como se describe en la publicación *Communications Server IP Configuration Guide* (SC31-8775).

Las acciones remotas (basadas en host) de subproyectos z/OS UNIX utilizan un servidor REXEC o SSH en el host. La comunicación SSH siempre se cifra utilizando SSL.

El cliente Gestor de despliegue de aplicaciones utiliza el servicio Web de CICS TS de la interfaz RESTful para invocar los servicios de host del Gestor de despliegue

de aplicaciones. La utilización de SSL está controlada por CICS TS, tal como se describe en la publicación *RACF Security Guide for CICS TS*.

comprobación de puerto de entrada

Developer for System z da soporte a la comprobación de puerto de entrada (POE), que permite el acceso de host sólo a las direcciones TCP/IP de confianza. La utilización de POE está controlada por la definición de perfiles específicos del software de seguridad y por la directiva `enable.port.of.entry` del archivo `rzed.envvars`, como se describe en la sección “Comprobación de puerto de entrada (POE)” en la página 36.

Tenga en cuenta que la activación de POE influirá sobre otras aplicaciones TCPIP que den soporte a la comprobación de POE, como INETD.

Uso de PassTickets

Después de del inicio de sesión, se utilizan Pases (PassTickets) para establecer la seguridad de las hebras dentro del servidor RSE. Esta característica no puede inhabilitarse. Los PassTickets son contraseñas generadas por el sistema con un tiempo de vida aproximado de 10 minutos. Los PassTickets generados se basan en el algoritmo de cifrado DES, en el ID de usuario, en el ID de aplicación, en la indicación de fecha y hora, y en una clave secreta. Esta clave secreta es un número de 64 bits (16 caracteres hexadecimales) que deben definirse en el software de seguridad. Para seguridad adicional, el software de seguridad de z/OS utiliza PassTickets de forma predeterminada como contraseñas de un solo uso.

Para ayudarle a comprender la utilización de PassTicket, se proporciona esta breve descripción del proceso de seguridad del RSE:

1. El cliente se conecta al puerto del host 4035, el daemon RSE.
2. El daemon RSE autentica el cliente mediante las credenciales presentadas por el éste.
3. El daemon RSE crea un ID de cliente exclusivo (señal de seguridad) y una hebra de servidor RSE.
4. El servidor RSE genera un PassTicket y crea un entorno de seguridad para el cliente utilizando el PassTicket como contraseña.
5. El cliente se conecta al puerto de host devuelto por el daemon RSE.
6. El servidor RSE valida el cliente utilizando el ID de éste.
7. El servidor RSE utiliza un PassTicket generado como contraseña para todas las acciones futuras que la requieran.

Nota: Se utiliza un mecanismo parecido para configurar las conexiones seguras con el gestor de depuración.

La contraseña real del cliente ya no es necesaria después de la autenticación inicial porque los productos de seguridad compatibles con SAF pueden evaluar tanto los PassTickets como las contraseñas habituales. El servidor RSE genera y utiliza un PassTicket cada vez que es necesaria una contraseña, cuyo resultado es una contraseña válida (temporal) para el cliente.

El uso de PassTickets permite a RSE configurar un entorno de seguridad específico del usuario a voluntad, sin necesidad de almacenar todos los ID de usuario y las contraseñas en una tabla, cosa que podría poner en peligro esta información. También lo permite para métodos de autenticación de cliente que no utilizan contraseñas reutilizables, como los certificados X.509.

Los perfiles de seguridad de las clases de APPL y PTKTDATA son necesarios para poder utilizar los PassTickets. Estos perfiles son específicos de la aplicación y, por ello, no afectan a la configuración de su sistema actual.

El hecho de que los PassTickets sean específicos de la aplicación implica que tanto RSE como el Supervisor de trabajos JES deben utilizar el mismo ID de aplicación (APPLID). Por omisión, ambos servidores utilizan FEKAPPL como APPLID, pero esto puede verse modificado por la directiva APPLID de `rsed.envvars` para RSE y de `FEJJCNFG` para el Supervisor de trabajos JES.

No debe utilizar OMVSAPPL como ID de aplicación porque abrirá la clave secreta a la mayoría de aplicaciones z/OS UNIX. Tampoco debe utilizar el ID de aplicación MVS predeterminado, que es MVS seguido por el ID SMF del sistema, porque esto abrirá la clave secreta a la mayoría de aplicaciones MVS (incluyendo trabajos por lotes de usuarios).

La unidad más pequeña de indicación de fecha y hora del PassTicket es de 1 segundo. Esto implica que todos los PassTicket generados en un mismo segundo por la misma aplicación para el mismo ID de usuario serán idénticos. Esto, junto con el software de seguridad de z/OS gestionando los PassTicket como contraseñas de un solo uso, genera un problema para Developer for System z durante el inicio de sesión, ya que harán falta varios PassTickets en un mismo segundo. Por lo tanto, Developer for System z precisa de un distintivo en las definiciones de PassTicket que permita la reutilización de los PassTicket generados.

Atención: La solicitud de conexión del cliente fallará si los PassTickets no están configurados correctamente.

Registro de auditoría

Developer for System z da soporte a los registros de auditoría de acciones gestionadas por el daemon RSE. Los registros de auditoría se almacenan como archivos de texto en el directorio de registro del daemon, utilizando el formato CSV (valores separados por comas).

Control de auditoría

Varias opciones de `rsed.envvars` influyen sobre la función de auditoría, como se documenta en "Definición de parámetros de inicio de Java adicionales con `_RSE_JAVAOPTS`" en la publicación *Guía de configuración de host* (SC11-3660).

- La función de auditoría se habilita/inhabilita mediante la opción `enable.audit.log`.
- Los valores de auditoría predeterminados están controlados por las opciones `audit.*`.
- La ubicación de los archivos de registro de auditoría está controlada por la opción `daemon.log`. La vía de acceso completa a los registros de auditoría es `daemonlog/server`, donde `daemonlog` es el valor de la opción `daemon.log`.
- La página de códigos utilizada para grabar los registros de auditoría está controlada por la directiva `_RSE_HOST_CODEPAGE`, tal como se describe en el apartado "rsed.envvars, archivo de configuración RSE" de la *Guía de configuración de host* (SC11-3660).

Puede utilizarse el mandato de operador **modify switch** para pasar manualmente a un nuevo archivo de registro de auditoría, como se indica en "Mandatos de operador" en la publicación *Guía de configuración de host* (SC11-3660).

Se envía un mensaje de aviso a la consola cuando el sistema de archivos que contiene los archivos de registro de auditoría se está quedando sin espacio libre. Este mensaje de consola (FEK103E) se repite regularmente hasta que se ha resuelto el problema de falta de espacio.

Procesamiento de auditoría

Un archivo de registro de auditoría nuevo se inicia después de un tiempo predeterminado o cuando se emite el mandato de operador **modify switch**. El archivo de registro antiguo se guarda como `audit.log.aaaammdd.hhmmss`, donde `aaaammdd.hhmmss` es la fecha/indicación de fecha y hora de cierre de los registros. La fecha/indicación de fecha y hora del sistema asignada al archivo indica la creación del archivo de registro. La combinación de las dos fechas muestra el período de tiempo cubierto por este archivo de registro de auditoría.

Las directivas `audit.action*` en `rsed.envvars` le permiten especificar una salida de usuario (script de shell de z/OS UNIX, REXX de z/OS UNIX o programa de z/OS UNIX), que invocará RSE cuando se cierre una anotación de auditoría. Esta salida de usuario podrá así procesar los datos dentro de la anotación de auditoría.

Los archivos de anotación de auditoría tienen la máscara de bit de permiso 640 (-rw-r-----), si no la cambia la directiva `audit.log.mode` en `rsed.envvars`. Esto quiere decir que el propietario (uid de z/OS UNIX del daemon RSE) tiene acceso de grabación y lectura, y el grupo del propietario (predeterminado) tiene acceso de lectura. Todos los demás intentos de acceso se denegarán, a menos que los realice un superusuario (UID 0) o alguien con permiso suficiente sobre el perfil `SUPERUSER.FILESYS` de la clase de seguridad `UNIXPRIV`.

Datos de auditoría

Se anotan las siguientes acciones:

- Acceso al sistema (conexión, desconexión)
- Acceso al spool de JES (someter, visualizar, retener, liberar, cancelar, depurar)
- Acceso a conjuntos de datos (lectura, grabación, creación, supresión, red denominación, compresión, migración, rellamada)
- Acceso a archivos (lectura, grabación, creación, supresión, red denominación)
- Ejecución de los mandatos TSO y z/OS UNIX

Cada acción anotada se almacena (con una fecha/indicación de fecha y hora) utilizando el formato CSV (valores separados por comas), que puede leerse mediante una herramienta de análisis de datos o automatización. Por ejemplo:

```
aaaa/mm/dd hh:mm:ss.sss,userid,action,dataset_name[,returncode]
[,additional_information]]
```

El conjunto de datos y las estadísticas de miembro también se registran cuando se abre el archivo. Se añaden a la línea que registra la finalización de la acción `READ` y los campos se delimitan con `%n`. Por ejemplo:

```
aaaa/mm/dd hh:mm:ss.sss,userid,action,dataset_name,returncode,create%nmodyfy%n...
```

Los atributos siguientes se registran, por el orden de la lista:

- Fecha y hora de creación (mm/dd/aaaa hh:mm)

- Fecha y hora de última modificación (mm/dd/aaaa hh:mm:ss)
- Fecha y hora de último acceso (mm/dd/aaaa hh:mm:ss)
- Formato de registro (RECFM)
- Indicador de revisión SCLM (N = número de revisión establecido, D = número de revisión no establecido)
- Número de revisión SCLM
- Caracteres "Bad Hex" incluidos (Y = sí, N = no)

Nota: Los caracteres "Bad Hex" requieren servicios de correlación de Developer for System z porque no sobreviven a un viaje de ida y vuelta al cliente debido a las discrepancias de página de códigos.

- Longitud del registro lógico (LRECL)
- Tamaño del archivo
- Reservado para uso futuro
- Reservado para uso futuro
- ID de usuario
- Bloquear (poner en cola) propietario para este conjunto de datos o miembro
- Puntos de código host CR (retorno de carro), LF (salto de línea) y NL línea nueva) y sus caracteres de sustitución (sólo disponibles cuando se usa un cliente de la versión 8.0.3 o superior)

Seguridad de JES

Developer for System z permite a los clientes acceder al spool de JES por medio del Supervisor de trabajos JES. El servidor proporciona limitaciones básicas de acceso, que pueden ampliarse con las características estándar de protección de archivos de spool de su producto de seguridad. Las acciones del operador (Retener, Liberar, Cancelar y Depurar) en los archivos de spool se realizan por medio de la consola de EMCS, para la que deben configurarse permisos condicionales.

Acciones en trabajos - limitaciones de destino

El supervisor de trabajos JES no proporciona a los usuarios de Developer for System z acceso de operador pleno al spool JES. Sólo están disponibles los mandatos Retener, Liberar, Cancelar y Depurar, y, por omisión, sólo para los archivos de spool propiedad del usuario. Para emitir los mandatos, se selecciona la opción pertinente en la estructura de menús del cliente (no hay indicador de mandatos). El ámbito de los mandatos puede ampliarse utilizando perfiles de seguridad para definir para qué trabajos están disponibles los mandatos.

Parecido a la acción de SDSF SJ, el Supervisor de trabajos JES también soporta el mandato Mostrar JCL para recuperar el JCL que creó la salida del trabajo seleccionado y visualizarlo en una ventana de editor. El Supervisor de trabajos JES recupera el JCL de JES y lo convierte en una función útil para los casos en que no se puede ubicar el miembro de JCL fácilmente.

Tabla 1. Mandatos de la consola del supervisor de trabajos JES

Acción	JES2	JES3
Retener	\$Hx(idtrabajo) con x = {J, S o T}	*F,J=idtrabajo,H

Tabla 1. Mandatos de la consola del supervisor de trabajos JES (continuación)

Acción	JES2	JES3
Liberar	\$Ax(idtrabajo) con x = {J, S o T}	*F,J=idtrabajo,R
Cancelar	\$Cx(idtrabajo) con x = {J, S o T}	*F,J=jobid,C
Purgar	\$Cx(jobid),P con x = {J, S o T}	*F,J=idtrabajo,C
Mostrar JCL	no aplicable	no aplicable

Por omisión, los mandatos de JES disponibles listados en la Tabla 1 en la página 26 están limitados a los trabajos que son propiedad del usuario. Esto puede cambiarse mediante la directiva `LIMIT_COMMANDS`, como se describe en el apartado "FEJJCNFG, archivo de configuración del supervisor de trabajos JES" de la *Guía de configuración de host* (SC11-3660).

Tabla 2. Matriz de permisos de mandato `LIMIT_COMMANDS`

LIMIT_COMMANDS	Propietario del trabajo	
	Usuario	Otros
USERID (valor predeterminado)	Permitido	No permitido
LIMITED	Permitido	Permitido sólo si lo permiten explícitamente los perfiles de seguridad
NOLIMIT	Permitido	Permitido si lo permiten los perfiles de seguridad o cuando la clase JESSPOOL no está activa

JES utiliza la clase JESSPOOL para proteger los conjuntos de datos SYSIN/SYSOUT. Parecido a SDSF, el Supervisor de trabajos JES amplía la utilización de la clase JESSPOOL para proteger también los recursos de trabajo.

Si `LIMIT_COMMANDS` no es `USERID`, el Supervisor de trabajos JES solicitará el permiso al perfil relacionado con la clase JESSPOOL, tal como se muestra en la tabla siguiente:

Tabla 3. Perfiles JESSPOOL ampliados

Mandato	Perfil JESSPOOL	Acceso necesario
Retener	nodeid.userid.jobname.jobid	ALTER
Liberar	nodeid.userid.jobname.jobid	ALTER
Cancelar	nodeid.userid.jobname.jobid	ALTER
Purgar	nodeid.userid.jobname.jobid	ALTER
Mostrar JCL	nodeid.userid.jobname.jobid.JCL	READ

Utilice las siguientes sustituciones en la tabla anterior:

idnodo	ID del nodo NJE del subsistema JES destino
idusuario	ID de usuario local del propietario del trabajo
nombretabajo	Nombre del trabajo
idtrabajo	ID del trabajo JES

Si la clase JESSPOOL no está activa, se produce un comportamiento diferente para los valores LIMITED y NOLIMIT de LIMIT_COMMANDS, como se describe en la sección "tabla de matriz de permisos del mandato LIMIT_COMMANDS" en "FEJJCENFG, archivo de configuración del supervisor de trabajos JES" de la publicación *Guía de configuración de host* (SC11-3660). El comportamiento es idéntico si JESSPOOL está activa, ya que, por omisión, la clase deniega el permiso si un perfil no está definido.

Acciones en trabajos - limitaciones de ejecución

La segunda fase de la seguridad de mandatos de spool JES, una vez especificados los destinos permitidos, incluye los permisos necesarios para ejecutar realmente el mandato de operador. Las comprobaciones de seguridad de JES y z/OS aplican esta autorización de ejecución.

Tenga en cuenta que Mostrar JCL no es un mandato de operador igual que el resto de mandatos del supervisor de trabajos JES (Retener, Liberar, Cancelar y Depurar), de manera que las limitaciones en la lista siguiente no son de aplicación porque no hay ninguna comprobación de seguridad más.

El Supervisor de trabajos JES emite todos los mandatos de operador de JES solicitados por un usuario por medio de una consola de EMCS ampliada (EMCS), cuyo nombre está controlado por la directiva CONSOLE_NAME, tal como se describe en el apartado "FEJJCENFG, archivo de configuración del supervisor de trabajos JES" de la *Guía de configuración de host* (SC11-3660).

El Supervisor de trabajos JES permite definir cuánta autoridad se otorga a la consola EMCS con la directiva LIMIT_CONSOLE, tal como se documenta en "FEJJCENFG, archivo de configuración del supervisor de trabajos JES" in la *Guía de configuración de host* SC11-3660-07 (SC23-7658).

Tabla 4. Matriz de autorización de consola LIMIT_CONSOLE

LIMIT_CONSOLE	Perfil activo en la clase OPERCMDS	No hay ningún perfil activo en la clase OPERCMDS
LIMITED (predeterminado)	Permitido si está permitido por el perfil de seguridad	No permitido
NOLIMIT	Permitido si está permitido por el perfil de seguridad	Permitido

Esta configuración permite al administrador de seguridad definir permisos de ejecución de mandatos granulares mediante las clases OPERCMDS y CONSOLE.

- A fin de utilizar una consola de EMCS, el usuario debe disponer (como mínimo) de una autorización de LECTURA sobre el perfil MVS.MCSOPER.console-name de la clase OPERCMDS. Tenga en cuenta que si no se define ningún perfil, el sistema otorgará la petición de autorización.
- A fin de utilizar un mandato de operador de JES, el usuario debe disponer de la autorización suficiente sobre el perfil JES%.** (o más concreto) de la clase

OPERCMDS. Tenga en cuenta que si no se define ningún perfil o la clase OPERCMD no está activa, el mandato fallará a causa de JES si LIMIT_CONSOLE=LIMITED está definido en FEJJC�FG.

- El administrador de seguridad también puede requerir que un usuario utilice el supervisor de trabajos JES al ejecutar el mandato de operador especificando WHEN(CONSOLE(JMON)) en la definición PERMIT. La clase CONSOLE debe estar activa para esta configuración del trabajo. Tenga en cuenta que es suficiente con que la clase CONSOLE esté activa; las consolas de EMCS no comprueban los perfiles.

El software de seguridad impide la asunción de identidad del servidor Supervisor de trabajos JES creando una consola JMON desde una sesión TSO. Aunque la consola se puede crear, el punto de entrada es distinto (supervisor de trabajos JES versus TSO). Los mandatos JES emitidos desde esta consola fallarán la comprobación de seguridad, si la seguridad está configurada según se describe en esta publicación.

Tenga en cuenta que el Supervisor de trabajos JES no puede crear la consola cuando debe ejecutarse un mandato si el nombre de consola ya se está utilizando. Para evitarlo, el programador de sistemas puede establecer la directiva GEN_CONSOLE_NAME=ON en el archivo de configuración del supervisor de trabajos JES o bien el administrador de seguridad puede definir perfiles de seguridad para impedir que los usuarios de TSO creen una consola. Los siguientes mandatos RACF de ejemplo impiden que nadie (excepto aquellos que lo tienen permitido) cree una consola TSO o SDSF:

- RDEFINE TSOAUTH CONSOLE UACC(NONE)
- PERMIT CONSOLE CLASS(TSOAUTH) ACCESS(READ) ID(#idusuario)
- RDEFINE SDSF ISFCMD.ODSP.ULOG.* UACC(NONE)
- PERMIT ISFCMD.ODSP.ULOG.* CLASS(SDSF) ACCESS(READ) ID(#idusuario)

Nota: Aunque no posean autorización sobre estos mandatos de operador, los usuarios todavía pueden someter trabajos y leer la salida de los trabajos por medio del Supervisor de trabajos JES, en caso de que dispongan de la autorización suficiente sobre los perfiles posibles que protegen estos recursos (como los de las clases JESINPUT, JESJOBS y JESSPOOL).

Para obtener más información sobre la protección de los mandatos de operador, consulte el manual *Security Server RACF Security Administrator's Guide* (SA22-7683).

Acceso a los archivos de spool

Por omisión, el Supervisor de trabajos JES permite acceder a todos los archivos de spool. Esto puede cambiarse mediante la directiva LIMIT_VIEW, como se describe en la sección "FEJJC�FG, archivo de configuración del supervisor de trabajos JES" de la publicación *Guía de configuración de host* (SC11-3660).

Tabla 5. matriz de permisos de examen de LIMIT_VIEW

LIMIT_VIEW	Propietario del trabajo	
	Usuario	Otros
USERID	Permitido	No permitido
NOLIMIT (valor predeterminado)	Permitido	Permitido si lo permiten los perfiles de seguridad o cuando la clase JESSPOOL no está activa

Para limitar a los usuarios de forma que solo utilicen sus propios trabajos en el spool de JES, defina la sentencia "LIMIT_VIEW=USERID" en el archivo de configuración del supervisor de trabajos JES, FEJCNFG. Si los usuarios necesitan acceso a un rango de trabajos más amplio, pero no a todos, utilice las características de protección de archivo de spool estándar de su producto de seguridad, como la clase JESSPOOL.

Al definir protección adicional, tenga presente que el supervisor de trabajos JES utiliza SAPI (interfaz de programación de aplicaciones SYSOUT) para acceder al spool. Ello implica que el usuario necesita como mínimo el acceso de actualización (UPDATE) a los archivos de spool, incluso para la función de examen. Este requisito no es necesario si se ejecuta z/OS 1.7 (z/OS 1.8 para JES3) o superior. En este caso, el permiso de lectura (READ) es suficiente para la función de examen.

Para obtener más información sobre la protección del archivo spool de JES, consulte el manual *Security Server RACF Security Administrator's Guide* (SA22-7683).

Comunicación cifrada con SSL/TLS

La comunicación externa (cliente-host) con RSE puede cifrarse mediante SSL (Capa de sockets seguros) o TLS (Seguridad de la capa de transporte). Esta característica está inhabilita por omisión y está controlada por los valores de `ssl.properties`. Consulte la sección "(Opcional) `ssl.properties`, cifrado SSL de RSE" en la publicación *Guía de configuración de host* (SC11-3660).

El daemon RSE y el servidor RSE soportan distintos mecanismos para almacenar certificados debido a las diferencias de su arquitectura. Esto hace que las definiciones y certificados SSL sean necesarias tanto para el daemon RSE como para el servidor RSE. Se puede utilizar un certificado compartido si el daemon RSE y el servidor RSE utilizan el mismo método de gestión de certificados.

Tabla 6. Mecanismos de almacenamiento de certificados de SSL

Almacenamiento de certificados	Creado y gestionado por	Daemon RSE	servidor RSE
anillo de claves	producto de seguridad compatible con SAF	soportado	soportado
base de datos de claves	gskkyman de z/OS UNIX	soportado	/
almacén de claves	Keytool de Java	/	soportado

Nota: Los anillos de claves compatibles con SAF son el método preferido para gestionar certificados.

Los anillos de claves compatibles con SAF puede almacenar la clave privada del certificado en la base de datos de seguridad o mediante el ICSF (recurso de servicio criptográfico integrado), la interfaz al hardware criptográfico de System z.

Se recomienda el ICSF para el almacenamiento de claves privadas relacionadas con certificados digitales, ya que es una solución más segura que la gestión de claves privadas sin ICSF. ICSF asegura que las claves privadas se cifran con la clave maestra de ICSF y que el acceso a ellas está controlado por los recursos generales de las clases de seguridad CSFKEYS y CSFSERV. Además, el rendimiento operativo mejora porque ICSF utiliza el hardware Coprocesador criptográfico. Consulte

Cryptographic Services ICSF Administrator's Guide (SA22-7521) para obtener más detalles sobre ICSF y cómo controlar qué usuarios pueden utilizar los servicios y claves criptográficas.

El daemon RSE utiliza funciones de SSL del sistema para gestionar las comunicaciones cifradas con SSL. Ello implica que SYS1.SIEALNKE debe estar controlado por programa por su software de seguridad y disponible para RSE a través de la directiva LINKLIST o STEPLIB en rsed.envvars.

El ID de usuario de RSE (stcrse en los mandatos de ejemplo siguientes) necesita una autorización para acceder a su anillo de claves y a los certificados relacionados cuando se utilizan anillos de claves compatibles con SAF ya sea para el daemon RSE o para el servidor RSE.

- RDEFINE FACILITY IRR.DIGTCERT.LIST UACC(NONE)
- RDEFINE FACILITY IRR.DIGTCERT.LISTRING UACC(NONE)
- PERMIT IRR.DIGTCERT.LIST CLASS(FACILITY) ACCESS(READ) ID(stcrse)
- PERMIT IRR.DIGTCERT.LISTRING CLASS(FACILITY) ACCESS(READ) ID(stcrse)
- SETROPTS RACLIST(FACILITY) REFRESH

La variable DSTORE_SSL_ALGORITHM de la directiva _RSE_JAVAOPTS de rsed.envvars permite elegir entre el método de cifrado SSL y su eventor TLS, tal como se documenta en "Definir parámetros de inicio Java adicionales con _RSE_JAVAOPTS" en la *Guía de configuración de host SC11-3660 (SC23-7658)*.

Consulte Capítulo 13, "Configurar SSL y autenticación de X.509", en la página 201 para obtener más detalles sobre cómo activar SSL para Developer for System z.

Nota: El cliente y el host de Developer for System z deben tener acceso a protocolos de cifrado común (SSLv3 o TLS) y definiciones de paquete de cifrado para poder configurar la comunicación cifrada. Para obtener más información sobre las definiciones de paquete de cifrado Java que utilizan el cliente y el servidor RSE, consulte el sitio de información de seguridad de tecnología Java developerWorks (<http://www.ibm.com/developerworks/java/jdk/security/>). Para obtener más información sobre definiciones de paquete de cifrado System SSL que utiliza el daemon RSE, consulte la publicación *Cryptographic Services System SSL Programming (SC24-5901)*.

De forma predeterminada, el daemon RSE se basa en valores predeterminados System SSL para protocolos de cifrado soportados y definiciones de paquete de cifrado. Puede alterar estos valores predeterminados especificando las variables de entorno GSK_PROTOCOL_* y GSK_V3_CIPHER_SPECS* en rsed.envvars. Para obtener más información sobre estas variables de entorno, consulte la publicación *Cryptographic Services System SSL Programming (SC24-5901)*.

Comunicación cifrada con el depurador integrado

La comunicación externa (cliente-host) con el gestor de depuración opcional también puede cifrarse mediante SSL o TLS. Para cifrar siguiendo este método, cree una política AT-TLS para el puerto utilizado por el gestor de depuración para comunicaciones externas, 5335 de forma predeterminada. En la Figura 9 en la página 32 se proporciona una política de muestra. Consulte Capítulo 14, "Configurar AT-TLS", en la página 215 para obtener más información sobre la configuración de ATS-TLS (Application Transparent TLS).

```

TTLSSRule                                RDz_Debug_Manager
{
  LocalPortRange                        5335
  Dirección                            De entrada
  TLSGroupActionRef                    grp_Production
  TTLSEnvironmentActionRef            RDz_Debug_Manager
}
TTLSEnvironmentAction                    RDz_Debug_Manager
{
  HandshakeRole Server
  TLSKeyRingParms
  {
    Conjunto de claves dbgmgr.racf
# El gestor de depuración debe poseer el conjunto de claves
  }
}
TLSGroupAction                            grp_Production
{
  TTLSEnabled                          Activado
  Rastreo                              2
}

```

Figura 9. Política AT-TLS para el gestor de depuración

Nota: El método de comunicación que utiliza el motor de depuración en el clienteDeveloper for System z para hablar con el Gestor de depuración en el host está vinculado de forma predeterminada al método de comunicación que utiliza el clienteDeveloper for System z para hablar con el daemon RSE. Esto implica que si el cifrado está habilitado para RSE, se presupone que también está habilitado para el gestor de depuración. Sin embargo, hay escenarios alternativos para otras configuraciones.

Autenticación de cliente mediante certificados X.509

El daemon RSE admite que los usuarios se autenticuen con un certificado X.509. El uso de una comunicación cifrada con SSL es un requisito previo para utilizar esta función, dado que es una extensión de la autenticación de host con un certificado utilizado en SSL.

El daemon RSE inicia el proceso de autenticación de cliente validando el certificado de cliente. Algunos de los aspectos clave que se comprueban son las fechas de validez del certificado y la fiabilidad de la autoridad certificadora (CA) utilizada para la firma del certificado. Opcionalmente, también se puede consultar una Lista de certificados revocados (CLR) (externa).

Una vez el daemon RSE ha validado el certificado, este es procesado para su autenticación. El certificado pasa a su producto de seguridad para que lo autentique, a menos que la directiva de `rsed.envvars enable.certificate.mapping` tenga un valor `false`, en cuyo caso el daemon RSE se encargará de la autenticación.

Si se realiza con éxito, el proceso de autenticación determinará el ID de usuario que deberá utilizarse para esta sesión; posteriormente, el daemon RSE lo probará para asegurar que es adecuado para el sistema host donde se está ejecutando el daemon RSE.

La última comprobación (que realizar para todos los mecanismos de autenticación, no sólo para los certificados X.509) verifica que el ID de usuario puede utilizar Developer for System z.

Si está familiarizado con las clasificaciones de seguridad SSL utilizadas por TCP/IP, la combinación de estos pasos de validación coinciden con las especificaciones del “Nivel 3 de autenticación de cliente” (el nivel más alto disponible).

Validación de la autoridad certificadora (CA)

Parte del proceso de validación del certificado incluye la comprobación de que el certificado ha sido firmado por una autoridad certificadora (CA) de confianza. Para ello, el daemon RSE debe tener acceso a un certificado que identifique a la CA.

Al utilizar la base de datos de claves **gskkyman** para su conexión SSL, debe añadirse a la base de datos de claves el certificado de CA.

Al utilizar un anillo de claves de SAF (método recomendado), debe añadir el certificado de CA a su base de datos de seguridad como certificado CERTAUTH con el atributo TRUST o HIGHTRUST, tal como se muestra en este mandato RACF de ejemplo:

- `RACDCERT CERTAUTH ADD(dsn) HIGHTRUST WITHLABEL('label')`

Tenga en cuenta que la mayor parte de productos de seguridad ya tienen los certificados de conocidas CA disponibles en sus bases de datos con estado NOTRUST. Utilice estos mandatos RACF de ejemplo para enumerar los certificados de CA existentes y marque uno de ellos como De confianza en base a la etiqueta que tiene asignada.

- `RACDCERT CERTAUTH LIST`
- `RACDCERT CERTAUTH ALTER(LABEL('HighTrust CA')) HIGHTRUST`

Nota: El estado HIGHTRUST es necesario si confía en que RACF autentique al usuario en base a la extensión HostIdMappings del certificado. Para obtener más información, consulte “Autenticación del software de seguridad” en la página 34.

Una vez el certificado de CA está añadido a su base de datos de seguridad, debe conectarse al anillo de claves del RSE, tal como se muestra en el mandato RACF de ejemplo:

- `RACDCERT ID(stcrse) CONNECT(CERTAUTH LABEL('HighTrust CA'))
RING(rdzssl.racf))`

Consulte el manual *Security Server RACF Command Language Reference* (SA22-7687) para obtener más información sobre el mandato **RACDCERT**.

Atención: si confía más en el daemon RSE que en su software de seguridad para autenticar un usuario, debe tener cuidado de no mezclar las CA con los estados TRUST y HIGHTRUST en su anillo de claves de SAF ni en su base de datos de claves de **gskkyman**. El daemon RSE no los puede diferenciar, por lo que los certificados firmados por una CA con estado TRUST serán válidos para la autenticación del ID de usuario.

(Opcional) Consulta en una lista de certificados revocados (CRL)

Si lo desea, puede indicar al daemon RSE que compruebe una o varias Listas de certificados revocados (CRL) para hacer más seguro el proceso de validación. Para hacerlo, añada variables de entorno relacionadas con la CRL a `rsed.envvars`.

- `GSK_CRL_SECURITY_LEVEL`

- GSK_LDAP_SERVER
- GSK_LDAP_PORT
- GSK_LDAP_USER
- GSK_LDAP_PASSWORD

Consulte el manual *Cryptographic Services System Secure Sockets Layer Programming* (SC24-5901) para obtener más información sobre estas y otras variables de entorno utilizadas por SSL del sistema de z/OS.

Nota: Tenga cuidado al especificar otras variables de entorno de SSL del sistema de z/OS (GSK_*) en `rsed.envvars`, ya que pueden cambiar la forma en que el daemon RSE maneja las conexiones SSL y la autenticación de certificados.

Autenticación del software de seguridad

RACF lleva a cabo varias comprobaciones para autenticar un certificado y devolver el ID de usuario asociado. Tenga en cuenta que puede que otros productos de seguridad lo hagan de manera distinta. Consulte la documentación de su producto de seguridad para obtener información adicional sobre la función `initACEE` utilizada para llevar a cabo la autenticación (modalidad de consulta).

1. RACF comprueba si el certificado está definido en la clase `DIGTCERT`. De ser así, RACF devuelve el ID de usuario que estaba asociado a este certificado cuando se añadió a la base de datos RACF.

Los certificados se definen en RACF mediante el mandato `RACDCERT`, como en el ejemplo siguiente:

```
RACDCERT ID(userid) ADD(dsn) TRUST WITHLABEL('label')
```

2. En caso que el certificado no esté definido, RACF comprueba si hay un filtro de nombre de certificado coincidente en las clases `DIGTNMAP` o `DIGTCRIT`. De ser así, devuelve el ID de usuario asociado al filtro coincidente más específico.

Nota: Se recomienda no utilizar filtros de nombre para los certificados utilizados por Developer for System z, ya que estos filtros correlacionan todos los certificados con un único ID de usuario. El resultado es que todos sus usuarios de Developer for System z iniciarán sesión con el mismo ID de usuario.

3. Si no hay ningún filtro de nombre coincidente, RACF ubica la extensión de certificado de `HostIdMappings` y extrae el par de nombre de host e ID de usuario incluido. En caso de que se encuentre y se valide, RACF devuelve el ID de usuario definido dentro de la extensión de `HostIdMappings`.

El par de ID de usuario y nombre de host es válido si todas estas condiciones son true:

- El certificado de CA utilizado para firmar este certificado está marcado como `HIGHTRUST` en la clase `DIGTCERT`.
- El ID de usuario almacenado en la extensión tiene una longitud válida (de 1 a 8 caracteres).
- El ID de usuario asignado al daemon RSE tiene (como mínimo) autorización de `LECTURA` sobre el perfil `IRR.HOST.hostname` en la clase de `SERVAUTH`, donde `hostname` es el nombre de host almacenado en la extensión. Este suele ser un nombre de dominio como, por ejemplo, `CDFMVS08.RALEIGH.IBM.COM`.

La definición de la extensión de `HostIdMappings` en sintaxis `ASN.1` es:

```

id-ce-hostIdMappings OBJECT IDENTIFIER ::= { 1 3 18 0 2 18 1 }
HostIdMappings ::= SET OF HostIdMapping
HostIdMapping ::= SEQUENCE {
    hostName          IMPLICIT[1] IA5String,
    subjectId         IMPLICIT[2] IA5String,
    proofOfIdPossession IdProof OPTIONAL
}
IdProof ::= SEQUENCE {
    secret            OCTET STRING,
    encryptionAlgorithm OBJECT IDENTIFIER
}

```

Nota: Una extensión de HostIdMappings no se respeta si el ID de usuario destino ha sido creado una vez iniciado el período de validez del certificado que incluye la extensión de HostIdMappings. Por ello, si está creando ID de usuario concretamente para certificados con extensiones de HostIdMappings, asegúrese de que crea el ID de usuario antes de que se envíen las peticiones del certificado.

Consulte la publicación *Security Server RACF Security Administrator's Guide* (SA22-7683) para obtener más información sobre los certificados X.509, sobre cómo los gestiona RACF y sobre cómo definir filtros de nombres de certificado. Consulte el manual *Security Server RACF Command Language Reference* (SA22-7687) para obtener más información sobre el mandato **RACDCERT**.

Autenticación del daemon RSE

Developer for System z puede llevar a cabo la autenticación básica de certificados X.509 sin basarse en su producto de seguridad. La autenticación llevada a cabo por el daemon RSE requiere que se definan un ID de usuario y un nombre de host en una extensión de certificado, y solamente se activa si la directiva `enable.certificate.mapping` de `rsed.envvars` tiene el valor de `FALSE`.

Esta función debe utilizarse cuando su producto de seguridad no admita la autenticación de un usuario en base a un certificado X.509, o bien si en el caso que su certificado fallara la(s) prueba(s) realizadas por el producto de seguridad (por ejemplo, si el certificado tiene un identificador erróneo para la extensión de HostIdMappings y no hay ninguna definición ni filtro de nombre en DIGTCERT).

El cliente consultará al usuario qué identificador de extensión (OID) utilizar, que es, de forma predeterminada, el OID de HostIdMappings {1 3 18 0 2 18 1}.

El daemon RSE le extraerá el ID de usuario y el nombre de host utilizando el formato de la ampliación de HostIdMappings. Este formato se describe en "Autenticación del software de seguridad" en la página 34.

El par de ID de usuario y nombre de host es válido si todas estas condiciones son `true`:

- El ID de usuario almacenado en la extensión tiene una longitud válida (de 1 a 8 caracteres).
- El ID de usuario asignado al daemon RSE tiene (como mínimo) autorización de LECTURA sobre el perfil IRR.HOST.hostname en la clase de SERVAUTH, donde hostname es el nombre de host almacenado en la extensión. Este suele ser un nombre de dominio como, por ejemplo, CDFMVS08.RALEIGH.IBM.COM.

Atención: Es decisión del administrador de seguridad asegurar que todas las CA reconocidas por el daemon RSE sean de confianza total, dado que el daemon RSE no puede comprobar si la persona que ha firmado el certificado de cliente es de confianza total o es simplemente de confianza. Consulte “Validación de la autoridad certificadora (CA)” en la página 33 para obtener más información sobre los certificados de CA accesibles.

Comprobación de puerto de entrada (POE)

Developer for System z da soporte a la comprobación de puerto de entrada (POE), que permite el acceso de host sólo a las direcciones TCP/IP de confianza. Esta característica está inhabilitada por omisión y requiere la definición del perfil de seguridad BPX.POE, como se muestra en los mandatos RACF de ejemplo que figuran a continuación:

- RDEFINE FACILITY BPX.POE UACC(NONE)
- PERMIT BPX.POE CLASS(FACILITY) ACCESS(READ) ID(STCRSE)
- SETROPTS RACLIST(FACILITY) REFRESH

Nota:

- El RSE debe configurarse para utilizar POE descomentando la opción “enable.port.of.entry=true” del archivo `rsed.envvars`, como se describe en el apartado “Definición de parámetros de inicio de Java adicionales con `_RSE_JAVAOPTS`” de la *Guía de configuración de host* (SC11-3660).
- El ID de usuario de RSE de STCRSE requiere el `UID(0)` si el perfil no está definido y la comprobación de POE está habilitada en `rsed.envvars`.
- El hecho de definir BPX.POE influirá sobre otras aplicaciones TCP/IP que den soporte a la comprobación de POE, como INETD.
- Deben configurarse zonas de seguridad (perfiles de EZB.NETACCESS.**, que son rangos de direcciones IP) en la clase SERVAUTH para poder utilizar todas las posibilidades de la comprobación de POE.

Consulte la publicación *Communications Server IP Configuration Guide* (SC31-8775) para obtener más información acerca del control de acceso a la red mediante comprobaciones de POE.

Alterar las funciones de cliente

Los clientes de Developer for System z versión 8.5.1 y superior pueden comprobar la autorización de acceso a perfiles de seguridad SAF y, en función del resultado, habilitar o inhabilitar la función relacionada para el usuario.

Developer for System z verifica los permisos de acceso a los perfiles listados en la Tabla 7 en la página 37 para determinar qué opciones deben estar habilitadas o inhabilitadas para el usuario.

Tabla 7. Información SAF para alterar las funciones de cliente

Perfil FACILITY	Longitud fija	Acceso necesario	Resultado
FEK.USR.OFF.REMOTECOPY.MVS.sysname	27	READ	El cliente inhabilita copiar y las funciones relacionadas para los conjuntos de datos MVS

Nota: Developer for System z presupone que un usuario no tiene autorización de acceso cuando su software de seguridad indique que no puede determinar si un usuario tiene o no autorización de acceso a un perfil. Un ejemplo sería cuando el perfil no está definido.

El valor de sysname coincide con el nombre de sistema del sistema de destino.

La columna "Longitud fija" indica la longitud de la parte fija del perfil de seguridad relacionado.

De forma predeterminada, Developer for System z espera que los perfiles FEK.* estén en la clase de seguridad FACILITY. Tenga en cuenta que los perfiles en la clase FACILITY tienen 39 caracteres como máximo. Si la suma de la longitud de la parte fija de perfil (FEK.USR.<key>) y la longitud de la parte de perfil específica del sitio (sysname) sobrepasa este número, puede colocar los perfiles en otra clase e instar a Developer for System z a que utilice esta clase en su lugar. Para ello, active _RSE_FEK_SAF_CLASS en rsed.envvars y proporcione el nombre de clase que quiera.

Las siguientes definiciones de seguridad de muestra permiten la acción REMOTECOPY.MVS para todos los usuarios en CDFMVS08, excepto aquellos del grupo RESTRICT:

```
RDEFINE FACILITY (FEK.USR.OFF.REMOTECOPY.MVS.CDFMVS08) -
  UACC(NONE) DATA('RATIONAL DEVELOPER FOR SYSTEM Z - CLIENT CONTROL')
PERMIT FEK.USR.OFF.REMOTECOPY.MVS.CDFMVS08 CLASS(FACILITY) -
  ID(RESTRICT) ACCESS(READ)
SETROPTS RACLIST(FACILITY) REFRESH
```

OFF.REMOTECOPY.MVS

Cuando los usuarios tienen acceso de lectura (READ) al perfil FEK.USR.OFF.REMOTECOPY.MVS.sysname, su versión de cliente de Developer for System z 8.5.1 y superior inhabilitarán las acciones arrastrar, copiar, guardar como y trabajar sin conexión para los conjuntos de datos MVS. El resultado es que los usuarios pueden acceder a los conjuntos de datos en este sistema, pero los usuarios no pueden crear una copia local de un conjunto de datos en su estación de trabajo. Esto ayuda a prevenir la exposición de información confidencial si la estación de trabajo local se pierde o es robada.

Grupos de desarrollador Envío a cliente

Los clientes de Developer for System z versión 8.0.1 y posteriores pueden tomar la información de actualización y de los archivos de configuración del cliente desde el host cuando se conectan, asegurando que todos los cliente tienen valores comunes y que están actualizados.

Desde la versión 8.0.3, el administrador del cliente puede crear varios conjuntos de configuraciones de cliente y varios escenarios de actualización del cliente para ajustar las necesidades de distintos grupos de desarrolladores. Esto permite a los usuarios recibir una configuración personalizada basada en un criterio como la pertenencia de un grupo LDAP o permiso para un perfil de seguridad.

Cuando utilice definiciones en su base de datos de seguridad como mecanismo de selección (el valor de SAF se especifica para las directivas en `pushtoclient.properties`), Developer for System z comprueba los permisos de acceso a los perfiles listados en la Tabla 8 para determinar a qué grupo de desarrolladores pertenece el usuario y si el usuario tiene permiso para rechazar actualizaciones.

Tabla 8. Información SAF de Envío a cliente

Perfil FACILITY	Longitud fija	Acceso necesario	Resultado
FEK.PTC.CONFIG.ENABLED. sysname.devgroup	23	READ	El cliente acepta actualizaciones de configuración para el grupo especificado
FEK.PTC.PRODUCT. ENABLED.sysname.devgroup	24	READ	El cliente acepta actualizaciones de producto para el grupo especificado
FEK.PTC.REJECT.CONFIG. UPDATES.sysname[.devgroup]	30	READ	El usuario puede rechazar actualizaciones de configuración
FEK.PTC.REJECT.PRODUCT. UPDATES.sysname[.devgroup]	31	READ	El usuario puede rechazar actualizaciones de producto

Nota: Developer for System z presupone que un usuario no tiene autorización de acceso cuando su software de seguridad indique que no puede determinar si un usuario tiene o no autorización de acceso a un perfil. Un ejemplo sería cuando el perfil no está definido.

El valor de `devgroup` coincide con el nombre de grupo asignado a un grupo específico de desarrolladores. Tenga en cuenta que el nombre de grupo es visible en clientes de Developer for System z.

El valor de `sysname` coincide con el nombre de sistema del sistema de destino.

La columna "Longitud fija" indica la longitud de la parte fija del perfil de seguridad relacionado.

De forma predeterminada, Developer for System z espera que los perfiles FEK.* estén en la clase de seguridad FACILITY. Tenga en cuenta que los perfiles en la clase FACILITY tienen 39 caracteres como máximo. Si la suma de la longitud de la parte fija del perfil (FEK.PTC.<key>) y la longitud de la parte del perfil específica del sitio (`sysname` o `sysname.devgroup`) sobrepasa este número, puede colocar los

perfiles en otra clase e indicar a Developer for System z que utilice esta clase en su lugar. Para ello, active `_RSE_FEK_SAF_CLASS` en `rsed.envvars` y proporcione el nombre de clase que quiera.

Tenga en cuenta que el administrador del cliente debe estar en la lista de acceso de perfiles `FEK.PTC.*.ENABLED.*` para definir y gestionar los metadatos Envío a cliente relacionados. Esto implica que los perfiles se deben definir con (al menos) el administrador de cliente en la lista de acceso para poder implementar el Envío a cliente con soporte para grupo.

Para obtener más información sobre la habilitación de varios soportes de grupo, consulte "(Opcional) `pushtoclient.properties`, control del cliente basado en host" en la publicación *Guía de configuración de host SC11-3660 (SC23-7658)*. Para obtener más información sobre conceptos e implementación del Envío a cliente, consulte Capítulo 7, "Consideraciones sobre envío a cliente", en la página 131.

Seguridad de archivo de registro

Creación de registro

Los directorios de registro y los archivos de registro que ha creado Developer for System z tienen, de forma predeterminada, permisos de acceso seguros donde sólo el propietario tiene acceso (lectura y escritura). Para los registros de servidor (y auditoría) el propietario es el ID de usuario de tarea iniciada RSED. Para registros de usuario el propietario es el ID de usuario proporcionado por el usuario final durante el inicio de sesión. La directiva `log.file.mode` en `rsed.envvars` se puede utilizar para establecer permisos de acceso diferentes. Tenga en cuenta que los permisos de acceso para los archivos de auditoría están controlados por separado y se establecen con la directiva `audit.log.mode` en `rsed.envvars`.

Antes de escribir en un directorio de registro, Developer for System z validará la propiedad del archivo y fallará la escritura si un usuario diferente es propiedad del archivo. Este comportamiento es nuevo en la versión 9.1.0 y es posible que necesite modificar una estructura de archivos de registro existente. La directiva `log.secure.mode` en `rsed.envvars` se puede utilizar para inhabilitar la comprobación de propiedad.

El JCL de ejemplo `FEKPBITS` se puede utilizar para convertir los permisos de acceso y de propiedad de una infraestructura de archivo de registro existente. `FEKPBITS` se encuentra en `FEK.#CUST.JCL`, a menos que haya especificado otra ubicación cuando ha personalizado y sometido el trabajo `FEK.SFEKSAMP (FEKSETUP)`. Para obtener más información, consulte "Configuración de personalización" en la *Guía de configuración del host (SC11-3660)*.

Recopilación de registros – requisitos para el peticionario

La tarea iniciada RSED da soporte al mandato de operador **MODIFY LOGS** para recopilar registros de host de Developer for System z e información de configuración. Los datos recopilados se sitúan en el archivo `z/OS UNIX, $TMPDIR/feklogs%sysname.%jobname`, donde `$TMPDIR` es el valor de la directiva `TMPDIR` en `rsed.envvars` (`/tmp` predeterminado), `%sysname` es el nombre del sistema `z/OS` y `%jobname` es el nombre de la tarea iniciada RSED.

Developer for System z consultará el producto de seguridad para permisos de acceso a perfiles `FEK.CMD.LOGS.**` para determinar si el peticionario tiene permiso para recopilar los registros especificados. De forma predeterminada, el peticionario

es el ID de usuario de la tarea iniciada RSED, a menos que se especifique la opción OWNER. Sólo el peticionario tiene acceso al archivo que contiene los datos recopilados.

Perfil FACILITY	Longitud fija	Acceso necesario	Resultado
FEK.CMD.LOGS.AUDIT.jobname	19	READ	El peticionario puede recopilar registros de auditoría de nombre de trabajo
FEK,CMD.LOGS.SERVER.jobname	20	READ	El peticionario puede recopilar registros de servidor de nombre de trabajo
FEK,CMD.LOGS.USER.userid	18	READ	El peticionario puede recopilar registros de usuario de ID de usuario
FEK,CMD.LOGS.OWNER.userid	19	READ	El peticionario cambia del ID de usuario de la tarea iniciada RSED al ID de usuario

Nota: Developer for System z presupone que un usuario tiene autorización de acceso cuando su software de seguridad indique que no puede determinar si un usuario tiene o no autorización de acceso a un perfil. Un ejemplo sería cuando el perfil no está definido.

El valor jobname coincide con el nombre de la tarea iniciada RSED. El valor userid coincide con un ID de usuario válido.

La columna "Longitud fija" indica la longitud de la parte fija del perfil de seguridad relacionado.

De forma predeterminada, Developer for System z espera que los perfiles FEK.* estén en la clase de seguridad FACILITY. Tenga en cuenta que los perfiles en la clase FACILITY tienen 39 caracteres como máximo. Si la suma de la longitud de la parte fija del perfil (FEK.CMD.LOGS.<key>) y la longitud de la parte del perfil específica del sitio (jobname o userid) sobrepasa este número, puede colocar los perfiles en otra clase e indicar a Developer for System z que utilice esta clase en su lugar. Para ello, active _RSE_FEK_SAF_CLASS en rsed.envvars y proporcione el nombre de clase que quiera.

Las violaciones de acceso se notifican mediante el mensaje de consola FEK302E.

Las definiciones de seguridad de ejemplo siguientes permiten a todo el mundo recopilar registros de host, pero sólo el grupo SYSPROG puede recopilar datos de auditoría:

```
RDEFINE FACILITY (FEK.CMD.LOGS.** ) UACC(READ) -
  DATA('RATIONAL DEVELOPER FOR SYSTEM Z - LOGS OPERATOR COMMAND')
RDEFINE FACILITY (FEK.CMD.LOGS.AUDIT.** ) UACC(NONE) -
  DATA('RATIONAL DEVELOPER FOR SYSTEM Z - LOGS OPERATOR COMMAND')
PERMIT FEK.CMD.LOGS.AUDIT.** CLASS(FACILITY) -
  ID(SYSPROG) ACCESS(READ)
SETOPTS RACLIST(FACILITY) REFRESH
```

Recopilación de registros – requisitos para el peticionario

El mandato de operador **MODIFY LOGS** utiliza el ID de usuario de la tarea iniciada RSED para recopilar registros de host e información de configuración y, de forma predeterminada, los archivos de registro de usuario se crean con permisos de acceso de archivos (el único que tiene acceso es el propietario). Para poder recopilar archivos de registro de usuario seguros, se debe otorgar permiso al ID de usuario de tarea iniciada RSED para leerlos.

El argumento **OWNER** del mandato del operador **MODIFY LOGS** hace que el ID de usuario especificado se convierta en el propietario de los datos recopilados. Para poder cambiar la propiedad, el ID de usuario de tarea iniciada RSED debe tener permiso para utilizar el servicio **CHOWN** de z/OS UNIX.

Hay tres formas de que estos permisos se suministren al ID de usuario de la tarea iniciada RSED. En orden de preferencia, son

- Acceso para seleccionar perfiles en la clase **UNIXPRIV**. Este método se utiliza en el trabajo de ejemplo **FEKRACF**.
- Acceso al perfil **BPX.SUPERUSER** en la clase **FACILITY**
- **UID 0**

Permisos de la clase **UNIXPRIV**

La clase **UNIXPRIV** contiene perfiles que permiten al administrador de seguridad otorgar selectivamente permisos z/OS UNIX relacionados especiales, en lugar de otorgar todos los permisos z/OS UNIX relacionados con el procedimiento de superusuario.

*Tabla 9. Permisos relacionados **UNIXPRIV** de z/OS UNIX*

Perfil	Permiso	Resultado
SUPERUSER.FILESYS	READ	Se permite al usuario leer cualquier archivo o directorio.
SUPERUSER.FILESYS.ACLOVERRIDE	READ	Sólo se requiere permiso si ACLOVERRIDE ya está definido. Permite al usuario leer cualquier archivo o directorio, independientemente de las definiciones ACL.
SUPERUSER.FILESYS.CHOWN	READ	Se permite al usuario cambiar el propietario de cualquier archivo o directorio.

Nota: Cuando el perfil **SUPERUSER.FILESYS.ACLOVERRIDE** está definido, los permisos de acceso definidos en ACL (access Control List) tienen prioridad sobre los permisos otorgados a través de **SUPERUSER.FILESYS**. El ID de usuario de la tarea iniciada RSED necesitará permiso de acceso **READ** al perfil **SUPERUSER.FILESYS.ACLOVERRIDE** para eludir definiciones ACL.

permiso de perfil **BPX.SUPERUSER**

Cuando el ID de usuario de tarea iniciada RSED tiene permiso **READ** al perfil **BPX.SUPERUSER** en la clase **FACILITY**, puede crear temporalmente un superusuario z/OS UNIX, para quien los permisos de archivos z/OS UNIX no cuentan.

UID 0

Cuando el ID de usuario de tarea iniciada RSED tiene UID 0 especificado en su segmento OMVS, es un superusuario z/OS UNIX, para quien no cuentan los permisos de acceso de archivos z/OS UNIX. No obstante, este procedimiento no es aconsejable puesto que UID 0 es probablemente un UID compartido y es aconsejable dar al ID de usuario de tarea iniciada RSED un UID exclusivo debido a otros permisos otorgados al ID. (Por ejemplo, los administradores de z/OS UNIX requieren UID 0 para determinadas tareas de gestión del sistema.)

Seguridad de depuración

El Depurador integrado opcional requiere que los usuarios dispongan de suficientes permisos de acceso a los perfiles de seguridad especificados. Si el usuario no dispone del permiso necesario, la sesión de depuración no se iniciará.

Developer for System z verifica el acceso a los perfiles listados en la Tabla 10 para determinar que permisos de depuración se han concedido.

Tabla 10. Información SAF para funciones de depuración

Perfil FACILITY	Acceso necesario	Resultado
AQE.AUTHDEBUG.STDPGM	READ	El usuario puede depurar aplicaciones sobre el estado del problema
AQE.AUTHDEBUG.AUTHPGM	READ	El usuario puede depurar aplicaciones sobre el estado del problema y aplicaciones autorizadas

Nota:

- Developer for System z presupone que un usuario no tiene autorización de acceso cuando su software de seguridad indique que no puede determinar si un usuario tiene o no autorización de acceso a un perfil. Un ejemplo sería cuando el perfil no está definido.
- Las versiones de Developer for System z anteriores a la versión 9.1.1 comprobaban los permisos UPDATE en el perfil AQE.AUTHDEBUG.WRITEBUFFER para permitir la depuración de transacciones CICS sólo de lectura. Este perfil ha dejado de utilizarse y se puede eliminar si el sistema host sólo dispone de Developer for System z versión 9.1.1 o una superior.

Los siguientes ejemplos de definiciones de seguridad permiten a todos los usuarios del grupo RDZDEBUG depurar aplicaciones sobre el estado del problema:

```
RDEFINE FACILITY (AQE.AUTHDEBUG.STDPGM) -  
  UACC(NONE) DATA('RATIONAL DEVELOPER FOR SYSTEM Z – DEBUG PROBLEM-STATE')  
PERMIT AQE.AUTHDEBUG.STDPGM CLASS(FACILITY) -  
  ID(RDZDEBUG) ACCESS(READ)  
SETROPTS RACLIST(FACILITY) REFRESH
```

Seguridad de CICSTS

El depurador integrado opcional puede depurar transacciones CICS. Consulte “Depuración de transacción CICS” en la página 162 para obtener más detalles.

Developer for System z permite, mediante el Gestor de despliegue de aplicaciones, que los administradores de CICS controlen qué definiciones de recursos CICS puede editar el desarrollador, sus valores predeterminados y la visualización de una definición de recurso CICS por medio del servidor CRD (definiciones de recursos CICS). Consulte la sección Capítulo 8, “Consideraciones de CICSTS”, en la página 151 para obtener más información acerca de las definiciones de recursos CICS TS necesarias.

Repositorio CRD

El conjunto de datos VSAM del repositorio del servidor CRD contiene todas las definiciones de recurso predeterminadas y, por tanto, debe protegerse contra actualizaciones, pero los desarrolladores deben poder leer los valores almacenados en él.

Transacciones CICS

Developer for System z suministra varias transacciones que el servidor CRD utiliza al definir y consultar recursos CICS. Cuando se conecta la transacción, la comprobación de la seguridad de recursos CICS, si está habilitada, se asegura de que el ID de usuario tiene autorización para ejecutar el ID de transacción.

Comunicación cifrada con SSL

El cliente Gestor de despliegue de aplicaciones utiliza la interfaz RESTful o los servicios Web de CICS TS para invocar el servidor CRD. El uso de SSL para esta comunicación está controlado por la definición CICS TS TCIPSERVICE, según se indica en la documentación de *RACF Security Guide for CICS TS*.

Seguridad de SCLM

El servicio SCLM Developer Toolkit ofrece funciones de seguridad opcionales para las funciones de construcción, promoción y despliegue.

Si el administrador de SCLM habilita la seguridad para una función, se realizan llamadas SAF para comprobar la autorización para ejecutar la función protegida con el ID de usuario del llamante o uno subordinado.

Consulte la *Guía del administrador de SCLM Developer Toolkit*, SC11-3815 (SC23-9801), para obtener más información acerca de las definiciones de seguridad de SCLM necesarias.

Información variada

Desecho de GATE

La primera vez que un espacio de direcciones insta a RACF a acceder a una clase de recurso que no está en RACLIST (almacenada en memoria), como por ejemplo la clases DATASET, RACF recuperará y almacenará todos los perfiles genéricos relacionados en el espacio de direcciones del usuario, en una lista conocida como GATE (Entrada de tabla de anclas genéricas). Hasta z/OS 1.12, RACF mantiene cuatro anclase genéricas para cada espacio de direcciones y cuatro para cada TCB de MVS que tiene su propio ACEE. Cuando se han utilizado las cuatro, RACF sustituye la referida menos recientemente cuando entra una nueva.

Si los usuarios acceden frecuentemente a más de cuatro calificadores de alto nivel de conjuntos de datos, las agrupaciones de hebras de RSE (que sirven a varios

usuarios que utilizan hebras con ACEEs específicas de usuario) pueden experimentar la operación de desecho de GATE ya que RACF debe rotar las entradas nuevas a través de las ranuras de ancla disponibles.

En z/OS 1.12, RACF introdujo la opción **GENERICANCHOR** del mandato **SET** que permite aumentar el tamaño de la tabla. Esto se puede establecer para todo el sistema o para cada nombre de trabajo.

ACEE gestionado

Developer for System z utiliza servicios del kernel de z/OS UNIX, como `pthread_security_np()` y `__passwd()`, que utilizan el servicio de seguridad `InitACEE`, que tiene como resultado bloques de control de seguridad "gestionados por ACEE". Su producto de seguridad pone un ACEE (Accessor Environment Element) gestionado en memoria caché, y dicho producto no tendrá en cuenta determinados cambios (como cambios de contraseña fuera de Developer for System z) hasta que transcurra el tiempo de espera de la memoria caché. (La caducidad puede tardar varios minutos).

Actualice la memoria caché del ACEE gestionado tras los cambios de seguridad para asegurarse de que Developer for System z utiliza los datos nuevos.

Almacenamiento en memoria caché ACEE

RACF puede ahorrar entornos ACEE (Accessor Environment Elements) utilizando el recurso VLF (Virtual Lookaside Facility) y recuperarlos para utilizarlos más tarde. Developer for System z solicita al software de seguridad que cree varios entornos de seguridad (ACEE) para el mismo usuario (uno para cada hebra específica de usuario en la agrupación de hebras RSE) y, por consiguiente, se puede beneficiar del almacenamiento en memoria caché ACEE.

Para obtener mas información sobre el almacenamiento en memoria caché ACEE, consulte "ACEEs and VLF considerations" en la publicación *Security Server RACF System Programmer's Guide (SA22-7681)*.

Archivos de configuración de Developer for System z

Existen varios archivos de configuración de Developer for System z, cuyas directivas afectan a la configuración de seguridad y auditoría. En base a la información de este capítulo, el administrador de seguridad y el programados de sistemas pueden decidir cuáles deberían ser los valores para las directivas siguientes.

Rastreo del daemon de bloqueo - FEJJCNFG

- `LIMIT_COMMANDS={USERID | LIMITED | NOLIMIT}`

Definir en qué trabajos pueden realizarse las acciones (excluidas las acciones examinar y someter). Para obtener más información, consulte "Acciones en trabajos - limitaciones de destino" en la página 26.

- `LIMIT_CONSOLE={LIMITED | NOLIMIT}`

Definir el nivel de autorización de la consola EMCS utilizada para ejecutar acciones. Para obtener más información, consulte "Acciones en trabajos - limitaciones de destino" en la página 26.

- `LIMIT_VIEW={USERID | NOLIMIT}`

Definir qué archivos de spool pueden examinarse. Para obtener más información, consulte "Acceso a los archivos de spool" en la página 29.

- `LOOPBACK_ONLY={ON | OFF}`
Definir si se puede acceder al Supervisor de trabajos JES desde fuera de este sistema z/OS. Para obtener más información, consulte el apartado *FEJJC�FG, archivo de configuración del supervisor de trabajos JES* del capítulo *Personalización básica* de la *Guía de configuración de host* SC11-3660 (SC23-7658).
- `APPLID={FEKAPPL | *}`
ID de aplicación utilizado para la creación/validación de PassTicket. Para obtener más información, consulte “Uso de PassTickets” en la página 23.

Nota: Puede obtener detalles sobre estas y otras directivas FEJJC�FG en el apartado “FEJJC�FG, archivo de configuración del supervisor de trabajos JES” de la *Guía de configuración de host* (SC11-3660).

RSE - rsed.envvars

- `_RSE_FEK_SAF_CLASS={FACILITY | *}`
Clase de seguridad que contiene perfiles FEK.**. Para obtener más información, consulte “Grupos de desarrollador Envío a cliente” en la página 37 and “Alterar las funciones de cliente” en la página 36.
- `(_RSE_JAVAOPTS) -DDENY_PASSWORD_SAVE={true | false}`
Denegar a los usuarios que guarden la contraseña del host en el cliente. Para obtener más información, consulte la sección “Definición de parámetros de inicio de Java adicionales con _RSE_JAVAOPTS” de la publicación *Guía de configuración de host* (SC11-3660).
- `(_RSE_JAVAOPTS) -DDSTORE_IDLE_SHUTDOWN_TIMEOUT=value`
Temporizador para desconectar a los clientes desocupados. Para obtener más información, consulte la sección “Definición de parámetros de inicio de Java adicionales con _RSE_JAVAOPTS” de la publicación *Guía de configuración de host* (SC11-3660).
- `(_RSE_JAVAOPTS) -DAPPLID={FEKAPPL | *}`
ID de aplicación utilizado para la creación/validación de PassTicket. Para obtener más información, consulte “Uso de PassTickets” en la página 23.
- `(_RSE_JAVAOPTS) -Denable.port.of.entry={true | false}`
Habilitar la comprobación de puerto de entrada. Para obtener más información, consulte “Comprobación de puerto de entrada (POE)” en la página 36.
- `(_RSE_JAVAOPTS) -DDSTORE_SSL_ALGORITHM={TLSv1.2 | SSL}`
Seleccionar SSL o TLS como método de cifrado de comunicaciones. Para obtener más información, consulte “Comunicación cifrada con SSL/TLS” en la página 30.
- `(_RSE_JAVAOPTS) -Denable.certificate.mapping={true | false}`
Utilizar su producto de seguridad para autenticar a los usuarios con un certificado X.509. Para obtener más información, consulte “Autenticación de cliente mediante certificados X.509” en la página 32.
- `GSK_CRL_SECURITY_LEVEL={LOW | MEDIUM | HIGH}`
`GSK_LDAP_SERVER=*`
`GSK_LDAP_PORT={389 | *}`
`GSK_LDAP_USER=*`
`GSK_LDAP_PASSWORD=*`
Comprobaciones de seguridad adicionales para autenticación de X.509. Para obtener más información, consulte “(Opcional) Consulta en una lista de certificados revocados (CRL)” en la página 33.
- `(_RSE_JAVAOPTS) -Dlog.file.mode={RW.N.N | * }`
Máscara de permisos de acceso de los directorios y archivos de registro de host.

- (RSE_JAVAOPTS) -Dlog.secure.mode={true | false }
Comprobaciones de seguridad adicionales (como la propiedad) para directorios y archivos de registro de host.
- (RSE_JAVAOPTS) -Ddaemon.log={/var/rdz/logs | *}
Vía de acceso que lleva a los archivos de registro de auditoría. Para obtener más información, consulte “Registro de auditoría” en la página 24.
- (RSE_JAVAOPTS) -Daudit.log.mode={RW.R.N | * }
Máscara de permisos de acceso de los archivos de registro de auditoría. Para obtener más información, consulte “Registro de auditoría” en la página 24.
- (RSE_JAVAOPTS) -Daudit.action=<script de shell>
(RSE_JAVAOPTS)
-Daudit.action.id=<ID de usuario>
Salida de usuario basada en z/OS UNIX que procesa registros de auditoría. Para obtener más información, consulte “Registro de auditoría” en la página 24.

Nota: Puede obtener detalles sobre estas y otras directivas `rsed.envvars` en el apartado “rsed.envvars, archivo de configuración RSE” de la *Guía de configuración de host* (SC11-3660).

RSE - ssl.properties

- `daemon_keydb_file={SAF key ring name | gskkyman key database name}`
Ubicación del certificado del daemon RSE. Para obtener más información, consulte “Comunicación cifrada con SSL/TLS” en la página 30.
- `daemon_key_label=certificate label`
Nombre del certificado del daemon RSE. Para obtener más información, consulte “Comunicación cifrada con SSL/TLS” en la página 30.
- `server_keystore_file={SAF key ring name | Java key store name}`
Ubicación del certificado del servidor RSE. Para obtener más información, consulte “Comunicación cifrada con SSL/TLS” en la página 30.
- `server_keystore_label=certificate label`
Nombre del certificado del servidor RSE. Para obtener más información, consulte “Comunicación cifrada con SSL/TLS” en la página 30.
- `server_keystore_type={JKS | JCECARACFKS | JCECCARACFKS}`
Tipo de almacén de claves utilizado (almacén de claves Java o anillo de claves de SAF). Para obtener más información, consulte “Comunicación cifrada con SSL/TLS” en la página 30.

Nota: Puede obtener detalles sobre estas y otras directivas `ssl.properties` en el apartado “(Opcional) ssl.properties, cifrado SSL de RSE” de la *Guía de configuración de host* (SC11-3660).

RSE - pushtoclient.properties

- `config.enabled={true | false | SAF | LDAP}`
`reject.config.updates={true | false | SAF | LDAP}`
Control basado en host de archivos de configuración de cliente de Developer for System z. Para obtener más información, consulte Capítulo 7, “Consideraciones sobre envío a cliente”, en la página 131.
- `product.enabled={true | false | SAF | LDAP}`
`reject.product.updates={true | false | SAF | LDAP}`

Control basado en host de actualizaciones del producto del cliente de Developer for System z. Para obtener más información, consulte Capítulo 7, "Consideraciones sobre envío a cliente", en la página 131.

Nota: Encontrará detalles sobre éstas y otras directivas `pushtoclient.properties` en "(Opcional) `pushtoclient.properties`, Control de cliente basado en host" en la publicación *Guía de configuración de host* (SC11-3660).

Definiciones de seguridad

Personalice y someta el miembro de ejemplo FEKRACF, que contiene mandatos de ejemplo RACF y z/OS UNIX para crear las definiciones básicas de seguridad para Developer for System z.

FEKRACF se encuentra en FEK.#CUST.JCL, a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP(FEKSETUP). Para obtener más detalles, consulte "Configuración de personalización" en la *Guía de configuración de host de IBM Rational Developer for System z*.

Consulte la publicación *RACF Command Language Reference* (SA22-7687), para obtener más información sobre los mandatos RACF.

Nota:

- Para los sitios que utilizan CA ACF2™ para z/OS, consulte la página de producto en el sitio de soporte de CA (<https://support.ca.com>) y compruebe el Documento de conocimiento de Developer for System z, TEC492389 relacionado. Este Documento de conocimiento contiene detalles sobre los mandatos de seguridad necesarios para configurar correctamente Developer for System z.
- Para los sitios que utilizan CA Top Secret® para z/OS, consulte la página de producto en el sitio de soporte de CA (<https://support.ca.com>) y compruebe el Documento de conocimiento de Developer for System z, TEC492091 relacionado. Este Documento de conocimiento contiene detalles sobre los mandatos de seguridad necesarios para configurar correctamente Developer for System z.

Las siguientes sesiones describen los pasos necesarios, la configuración opcional y las posibles alternativas.

Requisitos y lista de comprobación

Para completar la configuración de seguridad, el administrador de seguridad necesita conocer los valores enumerados en la Tabla 11. Estos valores se han definido durante los pasos anteriores de la instalación y personalización de Developer for System z.

Tabla 11. Variables de configuración de seguridad

Descripción	<ul style="list-style-type: none">• Valor predeterminado• Dónde encontrar la respuesta	Valor
Calificador de alto nivel de producto de Developer for System z	<ul style="list-style-type: none">• FEK• Instalación de SMP/E	

Tabla 11. Variables de configuración de seguridad (continuación)

Descripción	<ul style="list-style-type: none"> • Valor predeterminado • Dónde encontrar la respuesta 	Valor
Calificador de alto nivel de personalización de Developer for System z	<ul style="list-style-type: none"> • FEK.#CUST • FEK.SFEKSAMP(FEKSETUP), según se describe en "Configuración de personalización" en la <i>Guía de configuración de host de IBM Rational Developer for System z</i>. 	
Nombre de tarea iniciada del depurador integrado	<ul style="list-style-type: none"> • DBGMGR • FEK.#CUST.PROCLIB(DBGMGR), según se describe en "Cambios de PROCLIB", en la <i>Guía de configuración de host de IBM Rational Developer for System z</i> 	
Nombre de tarea iniciada del Supervisor de trabajos JES	<ul style="list-style-type: none"> • JMON • FEK.#CUST.PROCLIB(JMON), según se describe en "Cambios de PROCLIB", en la <i>Guía de configuración de host de IBM Rational Developer for System z</i> 	
Nombre de tarea iniciada del daemon RSE	<ul style="list-style-type: none"> • RSED • FEK.#CUST.PROCLIB(RSED), según se describe en "Cambios de PROCLIB", en la <i>Guía de configuración de host de IBM Rational Developer for System z</i>. 	
ID de aplicación	<ul style="list-style-type: none"> • FEKAPPL • /etc/rdz/rsed.envvars, según se describe en "Definir parámetros de inicio Java adicionales con _RSE_JAVAOPTS", en la <i>Guía de configuración de host de IBM Rational Developer for System z</i> 	

La lista que sigue es una visión general de las acciones necesarias para completar la configuración de seguridad básica de Developer for System z. Tal como se describe en las secciones siguientes, se pueden utilizar distintos métodos para cumplir estos requisitos en función del nivel de seguridad necesario. Para obtener información sobre la configuración de seguridad de servicios opcionales de Developer for System z, consulte las secciones anteriores.

- “Activar los valores y las clases de seguridad” en la página 49

- “Definición de un segmento OMVS para usuarios de Developer for System z” en la página 50
- “Definir las tareas iniciadas de Developer for System z” en la página 50
- “Definición de RSE como un servidor z/OS UNIX seguro” en la página 51
- “Definir bibliotecas controladas por programa MVS para RSE” en la página 52
- “Definir el soporte de PassTicket para RSE” en la página 53
- “Definir la protección de aplicaciones para el RSE” en la página 54
- “Definir permiso de acceso de archivos z/OS UNIX para RSE” en la página 54
- “Definir la seguridad de mandatos JES” en la página 55
- “Definir acceso al depurador integrado” en la página 57
- “Definir los perfiles de conjunto de datos” en la página 57
- “Verificar los valores de seguridad” en la página 60

Activar los valores y las clases de seguridad

Developer for System z utiliza diversos mecanismos de seguridad para garantizar un entorno de sistema host seguro y controlado para el cliente. Para ello, deben estar activos varias clases y valores de seguridad, como se muestra en los siguientes mandatos de RACF de muestra:

- Visualizar valores actuales
 - SETROPTS LIST
- Activar clase de recurso para z/OS UNIX, perfiles de certificados digitales y depurador integrado
 - SETROPTS GENERIC(FACILITY)
 - SETROPTS CLASSACT(FACILITY) RACLIST(FACILITY)
- Activar definiciones de tareas iniciadas
 - SETROPTS GENERIC(STARTED)
 - RDEFINE STARTED ** STDATA(USER(=MEMBER) GROUP(STCGROUP) TRACE(YES))
 - SETROPTS CLASSACT(STARTED) RACLIST(STARTED)
- Activar seguridad de consola para el Supervisor de trabajos JES
 - SETROPTS GENERIC(CONSOLE)
 - SETROPTS CLASSACT(CONSOLE) RACLIST(CONSOLE)
- Activar protección de mandatos de operador para el Supervisor de trabajos JES
 - SETROPTS GENERIC(OPERCMDS)
 - SETROPTS CLASSACT(OPERCMDS) RACLIST(OPERCMDS)
- Activar permiso de acceso de archivos z/OS UNIX para RSE
 - o SETROPTS GENERIC(UNIXPRIV)
 - o SETROPTS CLASSACT(UNIXPRIV) RACLIST(UNIXPRIV)
- Activar protección de aplicaciones para RSE
 - SETROPTS GENERIC(APPL)
 - SETROPTS CLASSACT(APPL) RACLIST(APPL)
- Activar el inicio de sesión seguro mediante PassTickets para el RSE
 - SETROPTS GENERIC(PTKTDATA)
 - SETROPTS CLASSACT(PTKTDATA) RACLIST(PTKTDATA)
- Activar control de programa para garantizar que el RSE sólo pueda cargar código de confianza
 - RDEFINE PROGRAM ** ADDMEM('SYS1.COMDLIB'//NOPADCHK) UACC(READ)

- SETROPTS WHEN(PROGRAM)

Nota: No cree el perfil ** si ya tiene un perfil * en la clase PROGRAM. Oscurece y complica la vía de acceso de búsqueda utilizada por el software de seguridad. En este caso, debe fusionar las definiciones * existentes y las definiciones ** nuevas. Utilice el perfil **, tal como se describe en la publicación *Security Server RACF Security Administrator's Guide* (SA22-7683).

Atención: Algunos productos, por ejemplo FTP, deben estar controlados por programa si "WHEN PROGRAM" está activo. Debe someter a prueba este control de programa antes de activarlo en un sistema de producción.

- (Opcional) Activar el soporte del puerto de entrada (POE) ampliado y el HostIdMappings X.509
 - SETROPTS GENERIC(SERVAUTH)
 - SETROPTS CLASSACT(SERVAUTH) RACLIST(SERVAUTH)

Definición de un segmento OMVS para usuarios de Developer for System z

Debe definirse un segmento OMVS de RACF o equivalente que especifique un ID de usuario (UID) de z/OS UNIX válido que no sea cero, un directorio inicial y un mandato de shell para cada usuario de Developer for System z. Su grupo predeterminado también requiere un segmento OMVS con un ID de grupo.

Al utilizar el depurador integrado opcional, el ID de usuario cuya aplicación se está depurando está activo y el grupo predeterminado correspondiente también requiere un segmento OMVS de RACF o equivalente activo.

En los mandatos RACF de ejemplo que figuran a continuación, sustituya los espacios reservados #idusuario, #identificador-usuario, #nombre-grupo e #identificador-grupo por los valores reales:

- ALTUSER #idusuario
OMVS(UID(#identif.-usuario) HOME(/u/#idusuario) PROGRAM(/bin/sh) NOASSIZEMAX)
- ALTGROUP #nombre-grupo OMVS(GID(#identificador-grupo))

Definir las tareas iniciadas de Developer for System z

Los siguientes mandatos RACF de ejemplo crean las tareas iniciadas DBGMR, JMON y RSED, con ID de usuario protegido (STCDBM, STCJMON y STCRSE) y el grupo STCGROUP asignado a los mismos. Sustituya los espacios reservados #id-grupo e #id-usuario-* por identificadores de OMVS válidos.

- ADDGROUP STCGROUP OMVS(AUTOUID)
DATA('GROUP WITH OMVS SEGMENT FOR STARTED TASKS')
- ADDUSER STCDBM DFLTGRP(STCGROUP) NOPASSWORD NAME('RDZ - DEBUG MANAGER')
OMVS(AUTOUID HOME(/tmp) PROGRAM(/bin/sh))
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
- ADDUSER STCJMON DFLTGRP(STCGROUP) NOPASSWORD NAME('RDZ - JES JOBMONITOR')
OMVS(AUTOUID HOME(/tmp) PROGRAM(/bin/sh))
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
- ADDUSER STCRSE DFLTGRP(STCGROUP) NOPASSWORD NAME('RDZ - RSE DAEMON')
OMVS(AUTOUID HOME(/tmp) PROGRAM(/bin/sh) ASSIZEMAX(2147483647))
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
- RDEFINE STARTED DBGMR.* DATA('RDZ - DEBUG MANAGER')
STDATA(USER(STCDBM) GROUP(STCGROUP) TRUSTED(NO))
- RDEFINE STARTED JMON.* DATA('RDZ - JES JOBMONITOR')
STDATA(USER(STCJMON) GROUP(STCGROUP) TRUSTED(NO))

- RDEFINE STARTED RSED.* DATA('RDZ - RSE DAEMON')
STDATA(USER(STCRSE) GROUP(STCGROUP) TRUSTED(NO))
- SETROPTS RACLIST(STARTED) REFRESH

Nota:

- Asegúrese de que los IDs de usuario de las tareas iniciadas están protegidos especificando la palabra clave NOPASSWORD.
- Asegúrese de que el servidor RSE tenga un uid OMVS exclusivo debido a los privilegios relacionados con z/OS UNIX otorgados a este uid.
- El daemon RSE requiere un tamaño de espacio de direcciones grande (2GB) para funcionar adecuadamente. Establezca este valor en la variable ASSIZEMAX del segmento OMVS para el ID de usuario STCRSE. El establecimiento de este valor asegura que el daemon RSE consigue el tamaño de región necesario, independientemente de los cambios realizados en MAXASSIZE de SYS1.PARMLIB(BPXPRMxx).
- RSE también requiere un gran número de hebras para funcionar adecuadamente. Puede establecer este límite en la variable THREADSMAX del segmento OMVS para el ID de usuario STCRSE. El establecimiento de este límite asegura que el RSE consigue el límite de hebras necesario, independientemente de los cambios realizados en MAXTHREADS o MAXTHREADTASKS de SYS1.PARMLIB(BPXPRMxx). Para determinar el valor correcto del límite de hebra, consulte "Consideraciones acerca de los ajustes" en la publicación *Guía de referencia de configuración de host* (SC11-7903).
- El ID de usuario STCJMON es otro buen candidato para establecer THREADSMAX en el segmento OMVS, ya que el supervisor de trabajos JES utilice una hebra por cada conexión de cliente.
- La tarea iniciada de depurador integrado (DBGMR) solo se utiliza en la función de depurador integrado opcional.

Considere la posibilidad de que el ID de usuario STCRSE sea restringido. Los usuarios con el atributo RESTRICTED no pueden acceder a recursos protegidos (MVS) a los que no tienen autorización de acceso específica.

ALTUSER STCRSE RESTRICTED

Para asegurarse de que los usuarios restringidos no obtengan acceso a los recursos del sistema de archivos de z/OS UNIX mediante los "otros" bits de permiso, defina el perfil RESTRICTED.FILESYS.ACCESS en la clase UNIXPRIV con UACC(NONE). Para obtener más información sobre cómo restringir IDs de usuario, consulte *Security Server RACF Security Administrator's Guide* (SA22-7683).

Atención: Si utiliza IDs de usuario restringidos, añada explícitamente el permiso para acceder a un recurso utilizando los mandatos **PERMIT** de TSO o **setfacl** de z/OS UNIX. Los recursos incluyen los recursos en los que la documentación de Developer for System z utiliza UACC, como por ejemplo el perfil ** de la clase PROGRAM, o los basados en convenciones comunes de z/OS UNIX, como por ejemplo que todos los usuarios tengan permiso de lectura y ejecución sobre las bibliotecas de Java. Pruebe el acceso antes de activarlo en un sistema de producción.

Definición de RSE como un servidor z/OS UNIX seguro

RSE requiere acceso de actualización (UPDATE) al perfil BPX.SERVER para crear o suprimir el entorno de seguridad de la hebra del cliente. Si este perfil no está definido, RSE requiere el UID(0). Este paso es necesario para que los clientes se puedan conectar.

El depurador integrado requiere acceso de actualización (UPDATE) al perfil BPX.SERVER para crear o suprimir el entorno de seguridad de la hebra del cliente. Si este perfil no se ha definido, se requiere UID(0) para el ID de usuario de tarea iniciada STCDBM. Este permiso solo es necesario cuando se utiliza la función de depurador integrado opcional.

- RDEFINE FACILITY BPX.SERVER UACC(NONE)
- PERMIT BPX.SERVER CLASS(FACILITY) ACCESS(UPDATE) ID(STCRSE)
- PERMIT BPX.SERVER CLASS(FACILITY) ACCESS(UPDATE) ID(STCDBM)
- SETROPTS RACLIST(FACILITY) REFRESH

Atención: Definir el perfil BPX.SERVER hace que z/OS UNIX como un todo cambie de la seguridad a nivel de UNIX a la seguridad a nivel de z/OS UNIX, la cual es más segura. Este conmutador puede afectar a otras operaciones y aplicaciones de z/OS UNIX. Pruebe la seguridad antes de activarlo en un sistema de producción. Para obtener más información sobre los diferentes niveles de seguridad, consulte *UNIX System Services Planning* (GA22-7800).

Definir bibliotecas controladas por programa MVS para RSE

Los servidores con autorización sobre BPX.SERVER deben ejecutarse en un entorno limpio controlado por programa. Este requisito implica que todos los programas a los que llama RSE también deben estar controlados por programa. Para las bibliotecas de carga MVS, el control de programa se gestiona mediante el software de seguridad. Este paso es necesario para que los clientes se puedan conectar.

RSE utiliza el sistema (SYS1.LINKLIB), el tiempo de ejecución de Language Environment (CEE.SCEERUN*) y la biblioteca de carga de la Pasarela de cliente TSO/ISPF (ISP.SISPLOAD) de ISPF.

- RALTER PROGRAM ** UACC(READ) ADDMEM('SYS1.LINKLIB'//NOPADCHK)
- RALTER PROGRAM ** UACC(READ) ADDMEM('SYS1.CSSLIB'//NOPADCHK)
- RALTER PROGRAM ** UACC(READ) ADDMEM('CEE.SCEERUN'//NOPADCHK)
- RALTER PROGRAM ** UACC(READ) ADDMEM('CEE.SCEERUN2'//NOPADCHK)
- RALTER PROGRAM ** UACC(READ) ADDMEM('ISP.SISPLOAD'//NOPADCHK)
- SETROPTS WHEN(PROGRAM) REFRESH

Nota: No utilice el perfil ** si ya tiene un perfil * en la clase PROGRAM. El perfil oscurece y complica la vía de acceso de búsqueda utilizada por el software de seguridad. En este caso, debe fusionar las definiciones * existentes y las definiciones ** nuevas. Utilice el perfil **, tal como se describe en la publicación *Security Server RACF Security Administrator's Guide* (SA22-7683).

Las siguientes bibliotecas adicionales prerequisite deben estar controladas por programa para dar soporte a la utilización de servicios opcionales. Esta lista no incluye los conjuntos de datos específicos de un producto con el que interactúa Developer for System z, como IBM File Manager.

- Biblioteca de tiempo de ejecución REXX alternativa, para SCLM Developer Toolkit
 - REXX.*.SEAGALT
- Biblioteca de carga del sistema, para cifrado SSL
 - SYS1.SIEALNKE
- Biblioteca de Developer for System z, para Integrated Debugger

- FEK.SFEKAUTH

Nota: Las bibliotecas diseñadas para colocación en LPA también requieren autorizaciones de control de programa si se accede a ellas por medio de LINKLIST o STEPLIB. Esta publicación documenta la utilización de las siguientes bibliotecas de LPA:

- ISPF, para pasarela de cliente TSO/ISPF de ISPF
 - ISP.SISPLPA
- Biblioteca de tiempo de ejecución REXX, para SCLM Developer Toolkit
 - REXX.*.SEAGLPA
- Developer for System z, para CARMA
 - FEK.SFEKLPA

Definir el soporte de PassTicket para RSE

La contraseña del cliente u otras formas de identificación, como un certificado X.509 sólo se utiliza para verificar la identidad durante la conexión. Después de eso, se utilizan Pases (PassTickets) para mantener la seguridad de las hebras. Este paso es necesario para que los clientes se puedan conectar.

Los PassTickets son contraseñas generadas por el sistema con un tiempo de vida aproximado de 10 minutos. Las PassTickets generadas se basan en una clave secreta. Esta clave es un número de 64 bits (16 caracteres hexadecimales). En los mandatos RACF de ejemplo siguientes, sustituya el espacio reservado key16 por una serie hexadecimal de 16 caracteres proporcionada por el usuario cuyos caracteres estén en los rangos 0-9 y A-F.

- RDEFINE PTKTDATA FEKAPPL UACC(NONE) SSIGNON(KEYMASKED(key16))
APPLDATA('NO REPLAY PROTECTION – DO NOT CHANGE')
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
- RDEFINE PTKTDATA IRRPTAUTH.FEKAPPL.* UACC(NONE)
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
- PERMIT IRRPTAUTH.FEKAPPL.* CLASS(PTKTDATA) ACCESS(UPDATE) ID(STCRSE)
- SETROPTS RACLIST(PTKTDATA) REFRESH

RSE soporta el uso de un ID de aplicación distinto de FEKAPPL. Elimine el comentario y personalice la opción "APPLID=FEKAPPL" en rsed.envvars para activar esto, según lo documentado en "Definir parámetros de inicio Java adicionales con _RSE_JAVAOPTS" en la *Guía de configuración de host de IBM Rational Developer for System z*. Las definiciones de clase PTKTDATA deben coincidir con el ID de aplicación real utilizado por RSE.

No debe utilizar OMVSAPPL como ID de aplicación porque abrirá la clave secreta a la mayoría de aplicaciones z/OS UNIX. Tampoco debe utilizar el ID de aplicación MVS predeterminado, que es MVS seguido por el ID SMF del sistema, porque esto abrirá la clave secreta a la mayoría de las aplicaciones MVS, incluidos trabajos por lotes de usuarios.

Nota:

- Si la clase PTKTDATA ya está definida, verifique que lo está como clase genérica antes de crear los perfiles enumerados a continuación. El soporte para los caracteres genéricos de la clase PTKTDATA es nuevo del release 1.7 de z/OS, con la introducción de una interfaz de Java a PassTickets.
- Sustituya el comodín (*) de la definición IRRPTAUTH.FEKAPPL.* con una máscara de ID de usuario válida para limitar los ID de usuario para los que RSE puede generar un PassTicket.

- Dependiendo de sus valores RACF, es posible que el usuario que ha definido un perfil aparezca también en la lista de acceso para el perfil en cuestión. Elimine este permiso para los perfiles PTKTDATA.
- RSE y el supervisor de trabajos JES deben tener el mismo ID de aplicación para que el supervisor de trabajos JES pueda evaluar los PassTickets presentados por RSE. Para el Supervisor de trabajos JES, el ID de aplicación se establece en el archivo de configuración FEJJCNGF con la directiva APPLID.
- Si el sistema tiene un producto criptográfico instalado y disponible, puede cifrar la clave de la aplicación de inicio de sesión seguro para obtener más protección. Para ello, utilice la palabra clave KEYENCRYPTED, en lugar de KEYMASKED. Para obtener más información, consulte *Security Server RACF Security Administrator's Guide* (SA22-7683).

Atención: La solicitud de conexión del cliente falla si PassTickets no está configurado correctamente.

Definir permiso de acceso de archivos z/OS UNIX para RSE

El mandato **MODIFY LOGS** del operador utiliza el ID de usuario de tarea iniciada RSED para recopilar registros de host e información de instalación. Y de forma predeterminada, los archivos de registro de usuario se crean con permisos de acceso de archivos (el único que tiene acceso es el propietario). Para poder recopilar archivos de registro de usuario seguros, se debe otorgar permiso al ID de usuario de tarea iniciada RSED para leerlos.

El argumento OWNER del mandato del operador **MODIFY LOGS** hace que el ID de usuario especificado se convierta en el propietario de los datos recopilados. Para poder cambiar la propiedad, el ID de usuario de tarea iniciada RSED debe tener permiso para utilizar el servicio CHOWN de z/OS UNIX.

- RDEFINE UNIXPRIV SUPERUSER.FILESYS UACC(NONE) DATA('OVERRIDE UNIX FILE ACCESS RESTRICTIONS')
- RDEFINE UNIXPRIV SUPERUSER.FILESYS.CHOWN UACC(NONE) DATA('OVERRIDE UNIX CHANGE OWNER RESTRICTIONS')
- PERMIT SUPERUSER.FILESYS CLASS(UNIXPRIV) ACCESS(READ) ID(STCRSE)
- PERMIT SUPERUSER.FILESYS.CHOWN CLASS(UNIXPRIV) ACCESS(READ) ID(STCRSE)
- SETROPTS RACLIST(UNIXPRIV) REFRESH

Observe que cuando está definido el perfil SUPERUSER.FILESYS.ACLOVERRIDE, los permisos de acceso definidos en ACL (access Control List) tienen prioridad sobre los permisos otorgados a través de SUPERUSER.FILESYS. El ID de usuario de la tarea iniciada RSED necesitará permiso de acceso READ al perfil SUPERUSER.FILESYS.ACLOVERRIDE para eludir las definiciones ACL.

Definir la protección de aplicaciones para el RSE

Durante el inicio de sesión de clientes, el daemon RSE verifica que un usuario pueda utilizar la aplicación.

- RDEFINE APPL FEKAPPL UACC(READ) DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
- SETROPTS RACLIST(APPL) REFRESH

Nota:

- Tal como se describe más detalladamente en “Definir el soporte de PassTicket para RSE” en la página 53, RSE soporta el uso de un ID de aplicación distinto de FEKAPPL. La definición de clase APPL debe coincidir con el ID de aplicación real que utiliza RSE.
- La solicitud de conexión de cliente es satisfactoria si el ID de aplicación no está definido en la clase APPL.
- La solicitud de conexión del cliente sólo fallará si el ID de aplicación está definido y el usuario no tiene acceso de lectura (READ) al perfil.

Definir archivos controlados por programa z/OS UNIX para el servidor

Los servidores con autorización sobre BPX.SERVER deben ejecutarse en un entorno limpio controlado por programa. Este requisito implica que todos los programas a los que llama RSE también deben estar controlados por programa. Para archivos z/OS UNIX, el control del programa viene gestionado por el mandato **extattr**. Para ejecutar este mandato, necesita acceso de lectura (READ) a BPX.FILEATTR.PROGCTL en la clase FACILITY o tener el UID(0).

El servidor RSE utiliza la biblioteca compartida de Java de RACF, (/usr/lib/libIRRRacf*.so).

- `extattr +p /usr/lib/libIRRRacf*.so`

Nota:

- A partir de z/OS 1.9, /usr/lib/libIRRRacf*.so se instala en modalidad controlada por programa durante la instalación de RACF SMP/E.
- A partir de z/OS 1.10, /usr/lib/libIRRRacf*.so forma parte de SAF, que se proporciona con el producto base z/OS, por lo que también está disponible para los clientes no RACF.
- La configuración puede ser diferente si utiliza un producto distinto de RACF. Para obtener más información, consulte la documentación de su producto de seguridad.
- La instalación SMP/E de Developer for System z establece el bit de control de programa para los programas internos de RSE.
- Utilice el mandato **ls -Eog** z/OS UNIX para visualizar el estado actual del bit de control de programa. El archivo está controlado por programa si la letra **p** se visualiza en la segunda serie.

```
$ ls -Eog /usr/lib/libIRRRacf*.so
-rwxr-xr-x  aps-  2      69632 Oct  5  2007 /usr/lib/libIRRRacf.so
-rwxr-xr-x  aps-  2      69632 Oct  5  2007 /usr/lib/libIRRRacf64.so
```

Definir la seguridad de mandatos JES

El Supervisor de trabajos JES emite todos los mandatos de operador de JES solicitados por un usuario por medio de una consola de EMCS ampliada (EMCS), cuyo nombre está controlado por la directiva `CONSOLE_NAME`, tal como se describe en el apartado "FEJJCNFG, archivo de configuración del supervisor de trabajos de JES" en *IBM Rational Developer for System z - Guía de configuración de host*.

Los mandatos RACF de ejemplo siguientes proporcionan a los usuarios de Developer for System z acceso condicional a un conjunto limitado de mandatos JES, que son Retener, Liberar, Cancelar y Depurar. Los usuarios sólo tienen permiso de ejecución si emiten los mandatos a través del Supervisor de trabajos JES. Sustituya el espacio reservado `#console` con el nombre de la consola.

- RDEFINE OPERCMDS MVS.MCSOPER.#console UACC(READ)
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
- RDEFINE OPERCMDS JES%.** UACC(NONE)
- PERMIT JES%.** CLASS(OPERCMDS) ACCESS(UPDATE) WHEN(CONSOLE(JMON)) ID(*)
- SETROPTS RACLIST(OPERCMDS) REFRESH

Nota:

- El uso de la consola está permitido si no está definido el perfil MVS.MCSOPER.#console.
- La clase CONSOLE debe estar activa para que WHEN(CONSOLE(JMON)) funcione, pero no hay ninguna comprobación real de perfiles en la consola CONSOLE para las consolas de EMCS.
- No sustituya JMON con el nombre real de la consola en la cláusula WHEN(CONSOLE(JMON)). La palabra clave JMON representa la aplicación de punto de entrada, no el nombre de la consola.

Atención: El hecho de definir mandatos JES con el acceso universal NONE en su software de seguridad puede afectar a otras operaciones y aplicaciones. Pruebe la seguridad antes de activarlo en un sistema de producción.

La Tabla 12 y la Tabla 13 muestran los mandatos de operador emitidos para JES2 y JES3 y los perfiles de seguridad específicos que pueden utilizarse para protegerlos.

Tabla 12. Mandatos de operador del Supervisor de trabajos JES2

Acción	Mandato	Perfil OPERCMDS	Acceso necesario
Retener	\$Hx(jobid) con x = {J, S o T}	jesname.MODIFYHOLD.BAT jesname.MODIFYHOLD.STC jesname.MODIFYHOLD.TSU	UPDATE
Liberar	\$Ax(jobid) con x = {J, S o T}	jesname.MODIFYRELEASE.BAT jesname.MODIFYRELEASE.STC jesname.MODIFYRELEASE.TSU	UPDATE
Cancelar	\$Cx(jobid) con x = {J, S o T}	jesname.CANCEL.BAT jesname.CANCEL.STC jesname.CANCEL.TSU	UPDATE
Purgar	\$Cx(jobid),P con x = {J, S o T}	jesname.CANCEL.BAT jesname.CANCEL.STC jesname.CANCEL.TSU	UPDATE

Tabla 13. Mandatos de operador del Supervisor de trabajos JES3

Acción	Mandato	Perfil OPERCMDS	Acceso necesario
Retener	*F,J=idtrabajo,H	jesname.MODIFY.JOB	UPDATE
Liberar	*F,J=idtrabajo,R	jesname.MODIFY.JOB	UPDATE
Cancelar	*F,J=idtrabajo,C	jesname.MODIFY.JOB	UPDATE
Purgar	*F,J=idtrabajo,C	jesname.MODIFY.JOB	UPDATE

Nota:

- Los mandatos del operador JES Retener, Liberar, Cancelar y Depurar, y el mandato Mostrar JCL sólo pueden ejecutarse en los archivos de spool propiedad del ID de usuario cliente, a menos que se especifique LIMIT_COMMANDS= con el

valor LIMITED, o se especifique NOLIMIT en el archivo de configuración del Supervisor de trabajos JES. Para obtener más información, consulte la sección "Acciones en trabajos - limitaciones de destino" de la publicación *Guía de referencia de configuración de host* (SC11-7903).

- Los usuarios pueden examinar cualquier archivos de spool, a menos que se haya definido LIMIT_VIEW=USERID en el archivo de configuración del Supervisor de trabajos JES. Para obtener más información, consulte la sección "Acceso a los archivos de spool" de la publicación *Guía de referencia de configuración de host* (SC11-7903).
- Incluso aunque no posean autorización sobre estos mandatos de operador, los usuarios todavía pueden someter trabajos y leer la salida de los trabajos por medio del Supervisor de trabajos JES, en caso de que dispongan de la autorización suficiente sobre los perfiles posibles que protegen estos recursos, como los de las clases JESINPUT, JESJOBS y JESSPOOL.

El software de seguridad impide la asunción de identidad del servidor Supervisor de trabajos JES creando una consola JMON desde una sesión TSO. Aunque la consola se puede crear, el punto de entrada es distinto, supervisor de trabajos JES versus TSO. Los mandatos JES emitidos desde esta consola fallarán la comprobación de seguridad, si la seguridad está configurada según se describe en esta publicación.

Definir acceso al depurador integrado

Los usuarios requieren acceso READ a uno de los perfiles AQE.AUTHDEBUG.* listados para poder utilizar el depurador integrado para depurar programas de estado del problema. Los usuarios con permiso para el perfil AQE.AUTHDEBUG.AUTHPGM también tienen permiso para depurar programas autorizados APF. Sustituya el espacio reservado #apf con ID de usuario o nombres de grupo RACF válidos para dichos usuarios que pueden depurar programas autorizados.

- RDEFINE FACILITY AQE.AUTHDEBUG.STDPGM UACC(NONE)
- PERMIT AQE.AUTHDEBUG.STDPGM CLASS(FACILITY) ACCESS(READ) ID(*)
- RDEFINE FACILITY AQE.AUTHDEBUG.AUTHPGM UACC(NONE)
- PERMIT AQE.AUTHDEBUG.AUTHPGM CLASS(FACILITY) ACCESS(READ) ID(#apf)
- SETROPTS RACLIST(FACILITY) REFRESH

Nota: Las versiones de Developer for System z anteriores a la versión 9.1.1 han utilizado otro perfil de clase FACILITY, AQE.AUTHDEBUG.WRITEBUFFER, que ya no se utiliza. Puede eliminarse si el sistema host solo tiene Developer for System z versión 9.1.1 o posterior.

Definir los perfiles de conjunto de datos

El acceso de lectura (READ) para los usuarios y de modificación (ALTER) para los programadores de sistemas es suficiente para la mayoría de conjuntos de datos de Developer for System z. Sustituya el espacio reservado #progsis por identificadores de usuario o nombres de grupo de RACF válidos. Solicite al programador del sistema que ha instalado y configurado el producto los nombres de conjunto de datos correctos. FEK es el calificador de alto nivel predeterminado utilizado durante la instalación y FEK.#CUST es el calificador de alto nivel predeterminado para los conjuntos de datos creados durante el proceso de personalización.

- ADDGROUP (FEK) OWNER(IBMUSER) SUPGROUP(SYS1)
DATA('RATIONAL DEVELOPER FOR SYSTEM Z - HLQ STUB')
- ADDSD 'FEK.*.***' UACC(READ)
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
- PERMIT 'FEK.*.***' CLASS(DATASET) ACCESS(ALTER) ID(#progsis)

- SETROPTS GENERIC(DATASET) REFRESH

Nota:

- Proteja FEK.SFEKAUTH contra actualizaciones porque este conjunto de datos está autorizado por APF. Lo mismo puede decirse de FEK.SFEKLOAD y FEK.SFEKLPA, pero en este caso debido a que estos conjuntos de datos están controlados por programa.
- En los mandatos de ejemplo se esta publicación y en el trabajo FEKRACF se presupone que EGN (Denominación genérica mejorada) está activa. Cuando EGN está activa, se puede utilizar el calificador ** para representar cualquier número de calificadores en la clase DATASET. Sustituya ** por * si EGN no está activa en el sistema. Para obtener más información sobre EGN, consulte *Security Server RACF Security Administrator's Guide* (SA22-7683).

Algunos de los componentes opcionales de Developer for System z requieren perfiles de conjunto de datos de seguridad adicionales. Sustituya los espacios reservados #progsis, #desarrollador-ram y #admcics por identificadores de usuario o nombres de grupo de RACF válidos:

- Si se utiliza la conversión de nombres largos/abreviados de SCLM Developer Toolkit, los usuarios necesitarán acceso de actualización (UPDATE) al VSAM de correlación, FEK.#CUST.LSTRANS.FILE.
 - ADDSD 'FEK.#CUST.LSTRANS.*.*' UACC(UPDATE)
 - DATA('RATIONAL DEVELOPER FOR SYSTEM Z - SCLMDT')
 - PERMIT 'FEK.#CUST.LSTRANS.*.*' CLASS(DATASET) ACCESS(ALTER) ID(#progsis)
 - SETROPTS GENERIC(DATASET) REFRESH
- Los desarrolladores de RAM (Repository Access Manager) de CARMA requieren acceso de actualización (UPDATE) a los VSAM de CARMA, FEK.#CUST.CRA*.
 - ADDSD 'FEK.#CUST.CRA*.*' UACC(READ)
 - DATA('RATIONAL DEVELOPER FOR SYSTEM Z - CARMA')
 - PERMIT 'FEK.#CUST.CRA*.*' CLASS(DATASET) ACCESS(ALTER) ID(#progsis)
 - PERMIT 'FEK.#CUST.CRA*.*' CLASS(DATASET) ACCESS(UPDATE) ID(#desarr.-ram)
 - SETROPTS GENERIC(DATASET) REFRESH
- Si se utiliza el servidor de Definición de recurso CICS (CRD) del Gestor de despliegue de aplicaciones, los administradores de CICS necesitan acceso UPDATE al VSAM de repositorio de CRD.
 - ADDSD 'FEK.#CUST.ADNREP*.*' UACC(READ)
 - DATA('RATIONAL DEVELOPER FOR SYSTEM Z - ADN')
 - PERMIT 'FEK.#CUST.ADNREP*.*' CLASS(DATASET) ACCESS(ALTER) ID(#progsis)
 - PERMIT 'FEK.#CUST.ADNREP*.*' CLASS(DATASET) ACCESS(UPDATE) ID(#admcics)
 - SETROPTS GENERIC(DATASET) REFRESH
- Si se define el repositorio de manifiestos del Gestor de despliegue de aplicaciones, todos los usuarios de CICS Transaction Server necesitan acceso de actualización (UPDATE) al VSAM del repositorio de manifiestos.
 - ADDSD 'FEK.#CUST.ADNMAN*.*' UACC(UPDATE)
 - DATA('RATIONAL DEVELOPER FOR SYSTEM Z - ADN')
 - PERMIT 'FEK.#CUST.ADNMAN*.*' CLASS(DATASET) ACCESS(ALTER) ID(#progsis)
 - SETROPTS GENERIC(DATASET) REFRESH

Utilice los siguientes mandatos RACF de ejemplo para establecer una configuración más segura, en la que el acceso de lectura (READ) también esté controlado.

- protección de conjunto de datos uacc(none)
 - ADDGROUP (FEK)
 - DATA('RATIONAL DEVELOPER FOR SYSTEM Z - HLQ STUB')
 - OWNER(IBMUSER) SUPGROUP(SYS1)"
 - ADDSD 'FEK.*.*' UACC(NONE)
 - DATA('RATIONAL DEVELOPER FOR SYSTEM Z')

```

- ADDSD 'FEK.SFEKAUTH' UACC(NONE)
  DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
- ADDSD 'FEK.SFEKLOAD' UACC(NONE)
  DATA('RATIONAL DEVELOPER FOR SYSTEM Z')

  ADDSD 'FEK.SFEKLMOD' UACC(NONE)
  DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
-
  ADDSD 'FEK.SFEKPROC' UACC(NONE)
  DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
- ADDSD 'FEK.#CUST.PARMLIB' UACC(NONE)
  DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
- ADDSD 'FEK.#CUST.CNTL' UACC(NONE)
  DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
-
  ADDSD 'FEK.#CUST.SQL' UACC(NONE)
  DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
- ADDSD 'FEK.#CUST.LSTRANS*.**' UACC(NONE)
  DATA('RATIONAL DEVELOPER FOR SYSTEM Z - SCLMDT')
- ADDSD 'FEK.#CUST.CRA*.**' UACC(NONE)
  DATA('RATIONAL DEVELOPER FOR SYSTEM Z - CARMA')
- ADDSD 'FEK.#CUST.ADNREP*.**' UACC(NONE)
  DATA('RATIONAL DEVELOPER FOR SYSTEM Z - ADN')
- ADDSD 'FEK.#CUST.ADNMAN*.**' UACC(NONE)
  DATA('RATIONAL DEVELOPER FOR SYSTEM Z - ADN')
• Permitir al programador del sistema gestionar todas las bibliotecas
- PERMIT 'FEK.*.**' CLASS(DATASET) ACCESS(ALTER) ID(#progsis)
- PERMIT 'FEK.SFEKAUTH' CLASS(DATASET) ACCESS(ALTER) ID(#progsis)
- PERMIT 'FEK.SFEKLOAD' CLASS(DATASET) ACCESS(ALTER) ID(#progsis)
- PERMIT 'FEK.SFEKLMOD' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
- PERMIT 'FEK.SFEKPROC' CLASS(DATASET) ACCESS(ALTER) ID(#progsis)
- PERMIT 'FEK.#CUST.PARMLIB' CLASS(DATASET) ACCESS(ALTER) ID(#progsis)
- PERMIT 'FEK.#CUST.CNTL' CLASS(DATASET) ACCESS(ALTER) ID(#progsis)
-
  PERMIT 'FEK.#CUST.SQL' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
- PERMIT 'FEK.#CUST.LSTRANS*.**' CLASS(DATASET) ACCESS(ALTER) ID(#progsis)
- PERMIT 'FEK.#CUST.CRA*.**' CLASS(DATASET) ACCESS(ALTER) ID(#progsis)
- PERMIT 'FEK.#CUST.ADNREP*.**' CLASS(DATASET) ACCESS(ALTER) ID(#progsis)
- PERMIT 'FEK.#CUST.ADNMAN*.**' CLASS(DATASET) ACCESS(ALTER) ID(#progsis)
• Permitir a los clientes acceder a las librerías de carga y las bibliotecas de exec
- PERMIT 'FEK.SFEKAUTH' CLASS(DATASET) ACCESS(READ) ID(*)
- PERMIT 'FEK.SFEKLOAD' CLASS(DATASET) ACCESS(READ) ID(*)
- PERMIT 'FEK.SFEKPROC' CLASS(DATASET) ACCESS(READ) ID(*)
- PERMIT 'FEK.#CUST.CNTL' CLASS(DATASET) ACCESS(READ) ID(*)
-
  PERMIT 'FEK.#CUST.SQL' CLASS(DATASET) ACCESS(READ) ID(*)

```

Nota: No son necesarios permisos para FEK.SFEKLPA, ya que todos los usuarios pueden acceder a todos los códigos que residen en LPA.

- Permitir que Integrated Debugger acceda a la biblioteca de carga.
 - PERMIT 'FEK.SFEKAUTH' CLASS(DATASET) ACCESS(READ) ID(STCDBM)
- Permitir al Supervisor de trabajos JES acceder a la biblioteca de carga y de parámetros
 - PERMIT 'FEK.SFEKAUTH' CLASS(DATASET) ACCESS(READ) ID(STCJMON)
 - PERMIT 'FEK.#CUST.PARMLIB' CLASS(DATASET) ACCESS(READ) ID(STCJMON)

- (Opcional) Permitir a los clientes actualizar el VSAM de conversión de nombres largos/abreviados para SCLMDT
 - PERMIT 'FEK.#CUST.LSTRANS.*.*' CLASS(DATASET) ACCESS(UPDATE) ID(*)
- (Opcional) Permitir a los desarrolladores de RAM actualizar los VSAM de CARMA para CARMA
 - PERMIT 'FEK.#CUST.CRA*.*' CLASS(DATASET) ACCESS(UPDATE) ID(#desarr.-ram)
- (Opcional) Permitir a los usuarios de CICS leer el VSAM del repositorio de CRD para el Gestor de despliegue de aplicaciones
 - PERMIT 'FEK.#CUST.ADNREP*.*' CLASS(DATASET) ACCESS(READ) ID(*)
- (Opcional) Permitir a los administradores de CICS actualizar el VSAM del repositorio de CRD para el Gestor de despliegue de aplicaciones
 - PERMIT 'FEK.#CUST.ADNREP*.*' CLASS(DATASET) ACCESS(UPDATE) ID(#admcics)
- (Opcional) Permitir a los usuarios de CICS actualizar el VSAM del repositorio de manifiestos para el Gestor de despliegue de aplicaciones
 - PERMIT 'FEK.#CUST.ADNMAN*.*' CLASS(DATASET) ACCESS(UPDATE) ID(*)
- (Opcional) Permitir al servidor TS CICS acceder a la biblioteca de carga para bidireccional y el Gestor de despliegue de aplicaciones
 - PERMIT 'FEK.SFEKLOAD' CLASS(DATASET) ACCESS(READ) ID(#cicsts)
- (Opcional) Permitir que el servidor CICS TS, las regiones IMS y los trabajos por lotes de MVS accedan a la biblioteca carga para mensajes de IRZ
 - PERMIT 'FEK.SFEKLMOD' CLASS(DATASET) ACCESS(READ) ID(#cicsts)
 - PERMIT 'FEK.SFEKLMOD' CLASS(DATASET) ACCESS(READ) ID(#ims)
 - PERMIT 'FEK.SFEKLMOD' CLASS(DATASET) ACCESS(READ) ID(#batch)
- Activar perfiles de seguridad
 - SETROPTS GENERIC(DATASET) REFRESH

Al controlar el acceso de lectura (READ) a los conjuntos de datos del sistema, debe otorgar a los servidores y usuarios de Developer for System z el permiso para leer (READ) los conjuntos de datos siguientes:

- CEE.SCEERUN
- CEE.SCEERUN2
- CBC.SCLBDLL
- ISP.SISPLoad
- ISP.SISPLPA
- SYS1.LINKLIB
- SYS1.SIEALNKE
- SYS1.SIEAMIGE
- REXX.V1R4M0.SEAGLPA

Nota: Si utiliza la biblioteca alternativa para el paquete de producto REXX, el nombre predeterminado de la biblioteca de tiempo de ejecución de REXX es REXX.*.SEAGALT en lugar de REXX.*.SEAGLPA, como se utilizaba en el ejemplo anterior.

Verificar los valores de seguridad

Utilice los siguientes mandatos de ejemplo para visualizar los resultados de las personalizaciones relacionadas con la seguridad.

- Valores y clases de seguridad
 - SETROPTS LIST
- Segmento OMVS para usuarios
 - LISTUSER #userid NORACF OMVS

- LISTGRP #group-name NORACF OMVS
- Tareas iniciadas
 - LISTGRP STCGROUP OMVS
 - LISTUSER STCDBM OMVS
 - LISTUSER STCJMON OMVS
 - LISTUSER STCRSE OMVS
 - RLIST STARTED DBGMR.* ALL STDATA
 - RLIST STARTED JMON.* ALL STDATA
 - RLIST STARTED RSED.* ALL STDATA
- RSE como servidor z/OS UNIX seguro
 - RLIST FACILITY BPX.SERVER ALL
- Bibliotecas controladas por programa MVS para RSE
 - RLIST PROGRAM ** ALL
- Soporte de PassTicket para RSE
 - RLIST PTKDATA FEKAPPL ALL SSIGNON
 - RLIST PTKDATA IRRPTAUTH.FEKAPPL.* ALL
- Protección de aplicaciones para el RSE
 - RLIST APPL FEKAPPL ALL
- Permiso de acceso de archivos z/OS UNIX para RSE
 - RLIST UNIXPRIV SUPERUSER.FILESYS ALL
 - RLIST UNIXPRIV SUPERUSER.FILESYS.CHOWN ALL
- Seguridad de mandatos JES
 - RLIST CONSOLE JMON ALL
 - RLIST OPERCMDS MVS.MCSOPER.JMON ALL
 - RLIST OPERCMDS JES%,** ALL
- Acceso al depurador integrado
 - RLIST FACILITY AQE.** ALL
- Perfiles de conjunto de datos
 - LISTGRP FEK
 - LISTDSD PREFIX(FEK) ALL

Opcionalmente, pueden existir perfiles que determinen el comportamiento Developer for System z para un usuario específico. Estos perfiles coinciden con el filtro FEK.** y se encuentra de forma predeterminada en la clase FACILITY. Consulte la directiva `_RSE_FEK_SAF_CLASS` en `rsed.envvars`. Puede utilizar el mandato **SEARCH** para listar los nombres de perfil. Utilice el mandato **RLIST** para mostrar los detalles de un perfil.

- SEARCH CLASS(FACILITY) FILTER(FEK.**)
- RLIST FACILITY #profile-name ALL

Capítulo 3. Consideraciones sobre TCP/IP

Developer for System z utiliza TCP/IP para proporcionar a los usuarios acceso al sistema central en una estación de trabajo que no es del sistema central. También utiliza TCP/IP para establecer comunicación entre distintos componentes y otros productos.

Tenga en cuenta que la mayoría de funciones de Developer for System z están basadas en z/OS UNIX, por tanto, TCP/IP utilizará la orden de búsqueda de z/OS UNIX para buscar sus archivos de configuración. Consulte Capítulo 15, “Configurar TCP/IP”, en la página 225 para obtener más información.

En este capítulo se tratan estos temas:

- “Puertos TCP/IP”
- “Alteración temporal del comportamiento de TCP/IP predeterminado” en la página 66
- “Varias pilas (CINET)” en la página 66
- “VIPA dinámico distribuido” en la página 68

Puertos TCP/IP

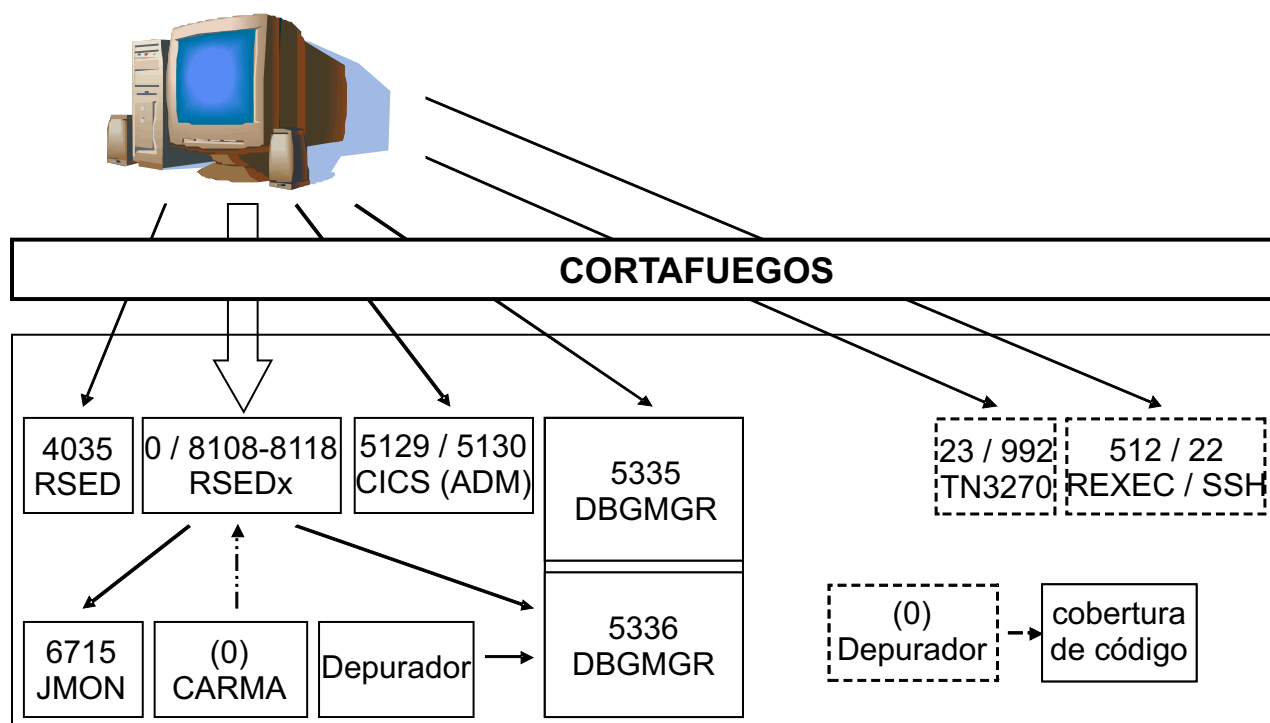


Figura 10. Puertos TCP/IP

La Figura 10 muestra los puertos de TCP/IP que Developer for System z puede utilizar. Las flechas muestran qué parte realiza el enlace (parte de la punta de la flecha) y qué parte realiza la conexión.

Comunicación externa

Defina los puertos siguientes para el cortafuegos que protege el host z/OS, ya que se utilizan para la comunicación cliente-host (utilizar el protocolo tcp):

- Daemon RSE para la configuración de la comunicación cliente-host, puerto predeterminado 4035. El puerto puede establecerse en el archivo de configuración `rsed.envvars`. La comunicación en este puerto puede cifrarse mediante SSL o TLS.
- Servidor RSE para la comunicación entre cliente y host. Por omisión, se utiliza cualquier puerto disponible, pero puede limitarse a un rango especificado con la definición de `_RSE_PORTRANGE` de `rsed.envvars`. El rango de puertos predeterminado para `_RSE_PORTRANGE` es 8108-8118 (11 puertos). La comunicación en este puerto puede cifrarse mediante SSL o TLS.
- (opcional) Gestor de depuración para servicios de depurador integrado, puerto predeterminado 5335. El puerto se puede establecer en el JCL de la tarea iniciada DBGMGR. La comunicación en este puerto puede cifrarse mediante SSL o TLS.
- (opcional) Servicio INETD para acciones remotas (basadas en host) en subproyectos z/OS UNIX:
 - REXEC (versión z/OS UNIX), puerto predeterminado 512.
 - SSH (versión z/OS UNIX), puerto predeterminado 22. La comunicación en este puerto está cifrada mediante SSL.
- (opcional) Servicio Telnet TN3270 para el Emulador de conexión de host, puerto predeterminado 23. La comunicación en este puerto puede cifrarse mediante SSL o TLS (puerto predeterminado 992). El puerto predeterminado que se asigna al servicio Telnet TN3270 depende de si el usuario elige el uso del cifrado.
- (opcional) Una de las dos o ambas interfaces de aplicación CICSTS para el Gestor de despliegue de aplicaciones:
 - Interfaz RESTful, puerto predeterminado 5130. El puerto puede establecerse en el CSD de CICS.
 - Interfaz de servicios Web, puerto predeterminado 5129. El puerto puede establecerse en el CSD de CICS. La comunicación en este puerto puede estar cifrada mediante SSL.

Nota: En general, el cliente especifica qué dirección TCP/IP se debe utilizar para establecer conexión con el host. Sin embargo, para asegurarse de que las sesiones de depuración se comuniquen con el host correspondiente, el gestor de depuración dicta al cliente las direcciones TCP/IP que se deben utilizar.

Comunicación interna

Varios servicios de host de Developer for System z se ejecutan en hebras o espacios de direcciones separados y utilizando sockets TCP/IP como mecanismo de comunicación, mediante la dirección de bucle de retorno del sistema. Todos estos servicios utilizan RSE para comunicarse con el cliente, confinando con ello su corriente de datos solamente al host. Para algunos servicios se utilizará cualquier puerto disponible, mientras que para otros el programador del sistema puede elegir el puerto o rango de puertos que se utilizará:

- Supervisor de trabajos JES para servicios relacionados con JES, puerto predeterminado 6715. El puerto se puede establecer en el miembro de configuración de `FEJJCNFG` y se repite en el archivo de configuración de `rsed.envvars`.
- (opcional) La comunicación de CARMA utiliza de forma predeterminada un puerto efímero pero se puede establecer un rango de puertos en el archivo de configuración de `CRASRV.properties`.

- (opcional) Gestor de depuración para servicios relacionados con la depuración, puerto predeterminado 5336. El puerto se puede establecer en el JCL de la tarea iniciada DBGMR.
- La cobertura de código basada en host, que es un trabajo por lotes, asigna un puerto efímero para permitir que IBM Debug Tool for z/OS establezca comunicación y proporcione los datos necesarios para el informe de cobertura de código.

Reserva de puerto TCP/IP

Si utiliza la sentencia PORT o PORTRANGE en PROFILE.TCPIP para reservar los puertos utilizados por Developer for System z, tenga en cuenta que las hebras activas en una agrupación de hebras de RSE realizarán varios enlaces. El nombre de trabajo de la agrupación de hebras de RSE es RSEDx, donde RSED es el nombre de la tarea iniciada RSE, y x, un número aleatorio de un dígito, por lo que es necesario utilizar comodines en la definición.

```
PORT      4035      TCP RSED ; Developer for System z - daemon de RSE
PORT      6715      TCP JMON ; Developer for System z - supervisor de trabajos JES
PORT      5335      TCP DBGMR ; Developer for System z - Integrado
integrado
PORT      5336      TCP DBGMR ; Developer for System z - Integrado
integrado
PORTRange 8108 11   TCP RSED* ; Developer for System z - _RSE_PORTRANGE
;PORTRange 5227 100 TCP RSED* ; Developer for System z - CARMA
```

Puertos CARMA y TCP/IP

CARMA (Common Access Repository Manager) se utiliza para acceder a un CSM (Software Configuration Manager) basado en host, como por ejemplo, CA Endeavor® SCM. En la mayoría de los casos, como con el daemon RSE, un servidor se enlaza a un puerto y escucha las solicitudes de conexión. Sin embargo, CARMA utiliza otro procedimiento, dado que el servidor CARMA no está activo cuando el cliente inicia la solicitud de conexión.

Cuando el cliente envía una solicitud de conexión, el extractor de CARMA, que está activo como hebra de usuario en una agrupación de hebras RSE, solicitará un puerto efímero o buscará un puerto libre dentro del rango especificado en el archivo de configuración CRASRV.properties y se enlaza a dicho puerto. El extractor inicia entonces el servidor CARMA y pasa el número de puerto, de manera que el puerto sepa a qué puerto conectarse. Cuando el servidor está conectado, el cliente puede mandar solicitudes al servidor y recibir los resultados.

Desde una perspectiva de TCP/IP, RSE (a través del extractor de CARMA) es el servidor que se enlaza al puerto, y el servidor CARMA es el cliente que se conecta.

Si utiliza la sentencia PORT o PORTRANGE en PROFILE.TCPIP para reservar el rango de puertos utilizados por CARMA, debe tener en cuenta que el extractor CARMA está activo en una agrupación de hebras RSE. El nombre de trabajo de la agrupación de hebras de RSE es RSEDx, donde RSED es el nombre de la tarea iniciada RSE, y x, un número aleatorio de un dígito, por lo que es necesario utilizar comodines en la definición.

```
PORTRange 5227 100 RSED* ; Developer for System z - CARMA
```

Nota: El IVP de CARMA, fekfivpc, fallará si reserva los puertos de CARMA para el uso de espacios de direcciones de RSE. Esto es de prever porque el IVP se ejecuta en el espacio de direcciones de la persona que ejecuta el IVP, no en el espacio de direcciones de RSE y TCP/IP no ejecutará correctamente la solicitud de enlace.

Consideraciones sobre LDAP

El servidor RSE se puede configurar para consultar uno o más servidores LDAP desde varios servicios de Developer for System z:

- Consultar grupos LDAP para soporte de grupos de desarrolladores múltiples de Envío a cliente.
- Consultar una o más listas de revocación de certificados (CRL) para autenticación X.509.

Tenga en cuenta que las medidas de seguridad de TCP/IP, como cortafuegos podrían impedir la conexión del servidor RSE (basado en host) al servidor LDAP. Utilice la información siguiente para asegurarse de que se pueda establecer contacto con el servidor LDAP:

- Las direcciones TCP/IP o nombres de DNS del servidor LDAP se muestran en la lista en las variables *_LDAP_SERVER de rsed.envvars.
- Los números de puerto del servidor LDAP se muestran en la lista en las variables *_LDAP_PORT en rsed.envvars.
- LDAP utiliza el protocolo TCP.
- El servidor RSE basado en host se pone en contacto con el servidor LDAP.
- El servidor RSE está activo en un espacio de direcciones RSEDx, donde RSED es el nombre de la tarea iniciada por RSE y x es un número aleatorio de un dígito, por ejemplo, RSED8.

Alteración temporal del comportamiento de TCP/IP predeterminado

ACK retardado

El ACK retardado retrasa el acuse de recibo (ACK) del paquete TCP en hasta unos 200ms. Este retardo aumenta la oportunidad de que el ACK se pueda enviar junto con la repuesta al paquete recibido, reduciendo así el tráfico de red. No obstante, si el remitente está esperando el ACK antes de enviar un paquete nuevo (por ejemplo, debido a la implementación del algoritmo de Nagle) y no hay respuesta al paquete recién enviado (por ejemplo, porque es parte de una transferencia de archivo), la comunicación se retrasa innecesariamente.

Developer for System z le permite inhabilitar la función de ACK retardada. En el host, esto se hace con la directiva `DSTORE_TCP_NO_DELAY` en `rsed.envvars`, según se indica en la documentación de la *Guía de configuración del host* SC11-3660 (SC23-7658).

Varias pilas (CINET)

z/OS Communication Server permite tener varias pilas TCP/IP activas simultáneamente en un mismo sistema. Esto se conoce como configuración CINET.

Si Developer for System z no está activo en la pila predeterminada, es posible que las funciones de Developer for System z seleccionadas fallen. Utilizar la afinidad de pila es una forma segura de resolver este problema. La afinidad de pila indica a Developer for System z que utilice únicamente una pila TCP/IP específica (en lugar de todas las pilas TCP/IP disponibles, que es el valor predeterminado para las tareas iniciadas).

Anule el comentario y personalice la directiva `_BPXK_SETIBMOPT_TRANSPORT` en el archivo de configuración `rsed.envvars` para establecer la afinidad de pila para la

tarea iniciada RSED. Consulte la sección correspondiente del "Capítulo 2 Personalización básica" de la publicación *Guía de configuración de host* (SC11-3660) para obtener más detalles sobre cómo personalizar estos archivos de configuración.

CARMA y afinidad de pila

CARMA (Common Access Repository Manager) se utiliza para acceder a un CSM (Software Configuration Manager) basado en host, como por ejemplo, CA Endevor® SCM. Para ello, CARMA inicia un servidor específico de usuario, que necesita una configuración adicional para aplicar la afinidad de pila.

De forma parecida a las tareas iniciadas por Developer for System z, la afinidad de pila para un servidor CARMA se establece con la variable `_BPXK_SETIBMOPT_TRANSPORT`, que debe pasarse a LE (Language Environment). Esto puede hacerse ajustando el mandato de arranque en el archivo de configuración `crastart*.conf` o `CRASUB*` activo.

Nota:

- El nombre exacto del archivo de configuración que contiene el mandato de arranque depende de distintas opciones seleccionadas por el programador de sistemas que ha configurado CARMA. Para obtener más información sobre este asunto, consulte el "Capítulo 3. (Opcional) Common Access Repository Manager (CARMA)" de la publicación *Guía de configuración de host* (SC11-3660).
- `_BPXK_SETIBMOPT_TRANSPORT` especifica el nombre de la pila TCP/IP que debe utilizarse, como se define en la sentencia `TCPIPJOBNAME` de la sentencia `TCPIP.DATA` relacionada.
- El hecho de codificar una sentencia `SYSTCPD DD` no establece la afinidad de pila solicitada.
- De forma predeterminada, CARMA no utiliza las pilas TCP/IP normales. CARMA utiliza la dirección de bucle de retorno para la comunicación entre el extractor CARMA y el servidor CARMA. Esto mejora la seguridad (solo los procesos locales tienen acceso a la dirección de bucle de retorno) y es probable que evite tener que añadir afinidad de pila a la comunicación con CARMA.

crastart*.conf

Sustituya la siguiente parte:

```
... PARM(&CRAPRM1. &CRAPRM2.)
```

por esta (donde `TCPIP` representa la pila TCP/IP deseada):

```
... PARM(ENVAR("_BPXK_SETIBMOPT_TRANSPORT=TCPIP") / &CRAPRM1. &CRAPRM2.)
```

Nota: `CRASTART` no da soporte a continuaciones de línea, pero no existe ningún límite en cuanto a la longitud de línea aceptada.

CRASUB*

Sustituya la siguiente parte:

```
... PARM(&PORT &TIMEOUT)
```

por esta (donde `TCPIP` representa la pila TCP/IP deseada):

```
... PARM(ENVAR("_BPXK_SETIBMOPT_TRANSPORT=TCPIP") / &PORT &TIMEOUT)
```

Nota: el sometimiento de trabajo limita la longitud de línea a 80 caracteres. Puede dividir una línea más larga en un espacio en blanco () y utilizar un signo más (+) al final de la primera línea para concatenar las dos líneas.

VIPA dinámico distribuido

DVIPA (Direccionamiento IP virtual dinámico) distribuido permite ejecutar simultáneamente configuraciones de Developer for System z idénticas en diferentes sistemas en su sysplex y hacer que TCP/IP, con la ayuda opcional de WLM, distribuya las conexiones de cliente entre esos sistemas.

Hay varias formas de configurar un DVIPA distribuido, pero Developer for System z impone algunas restricciones sobre estas opciones.

- El daemon de RSE posee el puerto que está definido para DVIPA distribuido pero el trabajo real tiene lugar en el servidor RSE que está activo como una hebra en otro espacio de direcciones. Por lo tanto no puede utilizar el método de distribución de SERVERWLM para equilibrar la carga entre sus sistemas porque los consejos de WLM se basarán en estadísticas del daemon RSE, no del servidor RSE.
- El cliente sólo conoce la dirección DVIPA utilizada por el Distribuidor de Sysplex para el daemon RSE. El Distribuidor de Sysplex pasará la petición de conexión a uno de los daemons de RSE disponibles que a su vez iniciarán una hebra de servidor RSE que enlazará con un puerto de ese sistema. Cuando el cliente se conecta con este puerto, vuelve a utilizar la dirección de DVIPA, no la dirección del sistema real por lo que debe asegurarse de que el Distribuidor Sysplex redirija la conexión nueva al sistema correcto.

Por lo tanto, Developer for System z requiere la definición de SYSPLEXPORTS en la sentencia VIPADISTRIBUTE para asegurarse de que los puertos utilizados por las hebras del servidor RSE sean exclusivos dentro del sysplex.

Nota:

- La utilización de SYSPLEXPORTS implica que la estructura de EZBEPOR se debe definir en su recurso de acoplamiento.
- La utilización de SYSPLEXPORTS implica que TCP/IP seleccionará un puerto efímero para la conexión secundaria. Esto implica que no puede reservar puertos para estas conexiones en el perfil TCP/IP con las directivas PORT y PORTRANGE. Tampoco puede utilizar _RSE_PORTRANGE en rsed.envvars para limitar los puertos utilizados por Developer for System z. Developer for System z proporciona una solución temporal para esta restricción ya que esto complica la configuración del cortafuegos.

También hay algunas restricciones en Developer for System z cuando se utiliza DVIPA:

- La directiva enable.dDVIPA en rsed.envvars debe estar habilitada.
- Para asegurarse de que el cliente Developer for System z no interferirá con la selección de puerto correcta que haga TCP/IP, debe habilitar la directiva deny.nonzero.port en rsed.envvars.
- Todos los servidores Developer for System z deben tener una configuración idéntica. Debe compartir /usr/lpp/rdz y /etc/rdz entre todos los sistemas participantes. También debe compartir /var/rdz/projects, /var/rdz/pushtoclient y /var/rdz/sc1mdt, si se utilizan estos. Tenga en cuenta que /var/rdz/WORKAREA y /var/rdz/logs deben ser exclusivos para cada sistema.
- Consulte Capítulo 11, “Ejecutar varias instancias”, en la página 175 para saber qué componentes de Developer for System z se deben compartir y qué componentes deben ser exclusivos por sistema.

El Supervisor de trabajos JES, CARMA y otros servidores Developer for System z sólo interactúan con el RSE local y por lo tanto no necesitan una configuración DVIPA.

El depurador integrado interactúa con el RSE local y no requiere la configuración de DVIPA. Para asegurarse de que las sesiones de depuración se comuniquen con el host correspondiente, el gestor de depuración dicta al cliente las direcciones TCP/IP que se deben utilizar y, por tanto, no requiere una configuración de DVIPA.

Los DVIPA distribuidos se definen mediante las palabras clave VIPADefine y VIPABackup del bloque VIPADynamic en su perfil TCP/IP. La palabra clave VIPADIStribute añade las definiciones de Sysplex Distributor. DVIPA distribuido necesita que todas las pilas participantes conozcan la existencia de sysplex, lo que se hace a través de las palabras clave SYSPLEXRouting y DYNAMICXCF del bloque IPCONFIG del perfil TCP/IP. Consulte la publicación *Communications Server: IP Configuration Reference* (SC31-8776) para obtener más detalles sobre estas directivas.

Consulte las publicaciones *MVS Setting Up a Sysplex* (SA22-7625) y *Communication Server: SNA Network Implementation Guide* (SC31-8777) para obtener más información sobre la configuración de la estructura EZBEPORIS en su recurso de acoplamiento.

Restringir la selección de puerto

La utilización de SYSPLEXPORTS implica que TCP/IP seleccionará un puerto efímero para la conexión secundaria. Un puerto efímero es cualquier puerto libre y no reservado de ninguna forma. El uso de un puerto efímero choca con los procedimientos recomendados de cortafuegos para limitar los puertos abiertos para comunicación ya que no se sabe qué puerto se utilizará.

Puede saltarse este problema forzando a Developer for System z a que utilice puertos conocidos para la conexión secundaria definiendo un _RSE_PORTRANGE exclusivo por sistema y asegurándose de que los rangos de puertos usados se reservan para el uso de Developer for System z en todos los sistemas. Debe tener en cuenta que este salto requiere TCP/IP APAR PM63379.

Para asegurarse de que TCP/IP dirigirá la conexión secundaria al sistema correcto, Developer for System z debe utilizar un rango de puertos exclusivo en cada sistema. Esto implica que no puede utilizar una configuración compartida e idéntica para los sistemas ya que _RSE_PORTRANGE en rsed.envvars debe ser exclusivo. Consulte "Archivos de configuración diferentes con idéntico nivel de software" en la página 176 in Capítulo 11, "Ejecutar varias instancias", en la página 175 para obtener información sobre cómo configurar varios servidores con archivos de configuración diferentes mientras se utiliza el mismo código. Debe utilizar una copia maestra de rsed.envvars y un script para ajustarlo y copiarlo en una configuración específica del sistema para asegurarse de que el archivo permanece idéntico entre los sistemas distintos.

1. Configure Developer for System z en SYS1 como si fuera una configuración de un sólo sistema, pero asegúrese de que /usr/lpp/rdz y /etc/rdz estén ubicados en un sistema de archivos compartido. Todas las partes basadas en MVS también se deben compartir con SYS2.
2. Utilice /etc/rdz/rsed.envvars como la copia maestra y añada una referencia a /etc/rdz al final del archivo para que las copias específicas de sistema puedan recoger los archivos de configuración restantes.

```

$ oedit /etc/rdz/rsed.envvars
-> añada lo siguiente al FINAL:
# -- NECESARIO PARA ENCONTRAR LOS ARCHIVOS DE CONFIGURACIÓN RESTANTES
CFG_BASE=/etc/rdz
CLASSPATH=.:$CFG_BASE:$CLASSPATH
# --

```

3. Cree /etc/rdz/update.sh, un script de shell que copiará el rsed.envvars maestro y ajustará _RSE_PORTRANGE

```

$ oedit /etc/rdz/update.sh
$ chmod 755 /etc/rdz/update.sh

```

```

#!/bin/sh
# Materiales bajo licencia - Propiedad de IBM
# 5724-T07 Copyright IBM Corp. 2012
# clonar rsed.envvars y establecer PORTRANGE para utilizarlo con RDz y DDVIPA

file=rsed.envvars          #; echo file $file
sys=${1:-$(sysvar SYSNAME)} #; echo sys $sys
dir=$(dirname $0)          #; echo dir $dir
# si sysname tiene un car. especial, anteponer \ (p. ej. SYS\1)
case "$sys" in
    "SYS1") range=8108-8118;;
    "SYS2") range=8119-8129;;
esac                        #; echo range $range
echo "setting port range $range for $sys using $dir/$file"

if test ! $range ; then
    echo ERROR: no port range defined for $sys ; exit 12 ; fi
if test ! -e $dir/$file ; then
    echo ERROR: file $dir/$file does not exist ; exit 12 ; fi
if test ! -d $dir/$sys ; then
    echo ERROR: directory $dir/$sys does not exist ; exit 12 ; fi

mv $dir/$sys/$file $dir/$sys/prev.$file 2>/dev/null
sed="/_RSE_PORTRANGE/s/.*/_RSE_PORTRANGE=$range/"
sed "$sed" $dir/$file > $dir/$sys/$file

if test ! -s $dir/$sys/$file ; then
    echo ERROR creating $dir/$sys/$file, restoring backup
    mv $dir/$sys/prev.$file $dir/$sys/$file ; exit 8 ; fi

```

Figura 11. update.sh - soportar la configuración de DDVIPA con un cortafuegos

4. Cree los directorios /etc/rdz/SYS1 y /etc/rdz/SYS2 y ejecute /etc/rdz/update.sh para llenarlos.

```

$ mkdir /etc/rdz/SYS1 /etc/rdz/SYS2
$ /etc/rdz/update.sh SYS1
setting port range 8108-8118 for SYS1 using
/etc/rdz/rsed.envvars
$ /etc/rdz/update.sh SYS2
setting port range 8119-8129 for SYS2 using
/etc/rdz/rsed.envvars

```
5. Asegúrese de que la tarea iniciada RSED apunta a /etc/rdz/&SYSNAME.

```

//      CNFG='/etc/rdz/&SYSNAME.'

```

A continuación, debe asegurarse de que los rangos de puertos definidos están reservados para Developer for System z en todos los sistemas, en el sysplex para asegurarse de que el número de puerto sigue siendo exclusivo dentro del sysplex. Utilice la sentencia PORT o PORTRANGE en PROFILE.TCPIP para reservar todos los rangos en cada sistema. El nombre de trabajo de la agrupación de hebras de RSE

es RSEDx, donde RSED es el nombre de la tarea iniciada RSE, y x, un número aleatorio de un dígito, por lo que es necesario utilizar comodines en la definición.

```
PORTRange 8108 22 RSED*           ; 8108-8129 - Developer for System z
                                   ; - conexión secundaria
```

Tal como se explica en “Flujo de conexión” en la página 8, el rango de puertos de _RSE_PORTRANGE puede ser pequeño. El servidor RSE no necesita el puerto exclusivamente para la duración de la conexión de cliente. Está sólo en el lapso de tiempo entre el enlace (servidor) y la conexión (cliente) que ningún otro servidor RSE puede enlazar al puerto. Esto significa que la mayoría de las conexiones utilizarán el primer puerto del rango, y el resto del rango será un almacenamiento intermedio en caso de varios inicios de sesión simultáneos.

Configuración de ejemplo

En el ejemplo siguiente hay dos sistemas z/OS, SYS1 y SYS2 que forman parte de un sysplex. System SYS1 está definido como el sistema que normalmente alberga el Sysplex Distributor del DVIPA distribuido de Developer for System z.

Después de definir el DVIPA distribuido, Developer for System z se puede iniciar en los sistemas para permitir el equilibrado de carga de conexiones de cliente entre los sistemas. El Supervisor de trabajos JES solo interactúa con el RSE local y por tanto no requiere una configuración de a DVIPA. Los clientes se conectarán al puerto 4035 en la dirección IP 10.10.10.1.

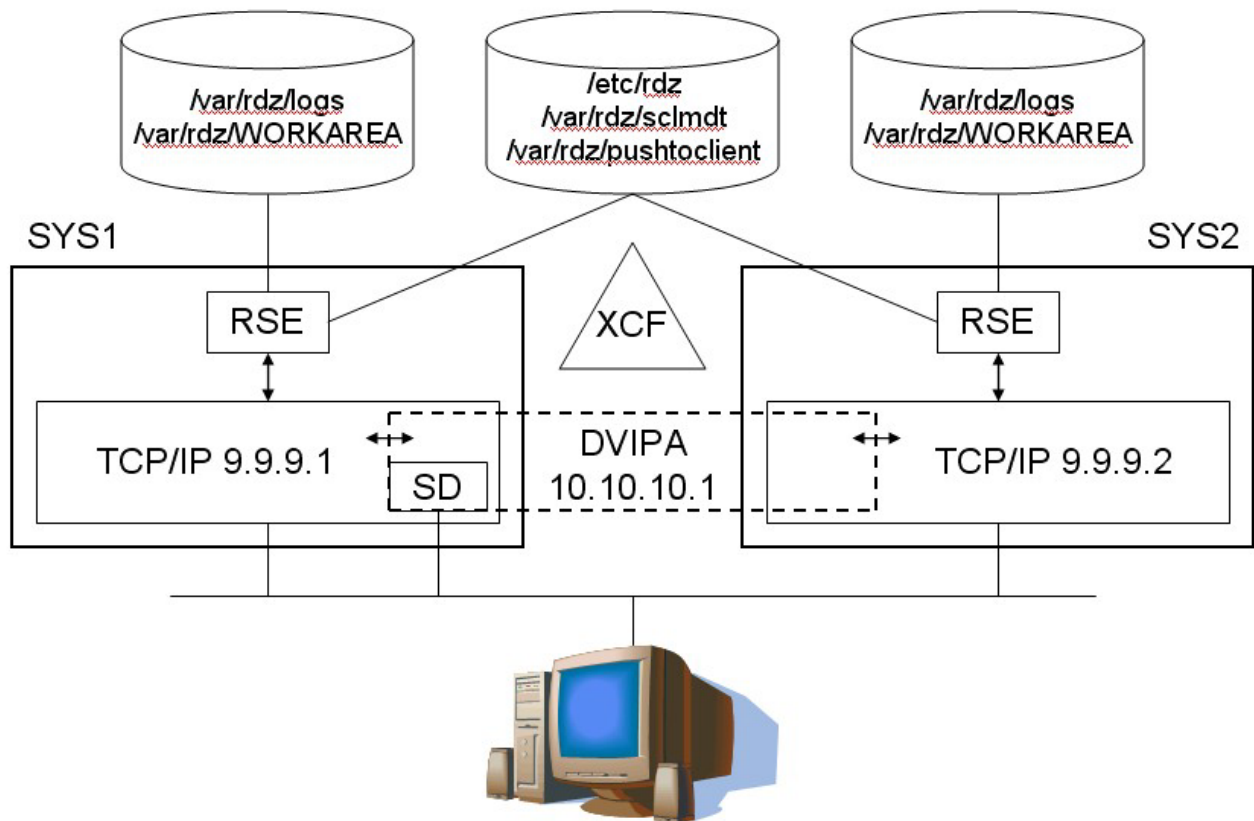


Figura 12. Ejemplo de VIPA dinámico distribuido

Sistema SYS1 – perfil de TCP/IP

```
IPCONFIG
  SYSPLEXRouting
; SYSPLEXROUTING es necesario ya que esta pila necesita comunicación sysplex
  DYNAMICXCF 9.9.9.1 255.255.255.0 1
; DYNAMICXCF define el dispositivo/enlace con la dirección inicial 9.9.9.1
; según sea necesario
  IGNORERedirect

VIPADYNAMIC
  VIPADEFINE 255.255.255.0 10.10.10.1
; VIPADEFINE define 10.10.10.1 como DVIPA principal en SYS1 para RDz
  VIPADISTRIBUTE DEFINE
; VIPADISTRIBUTE convierte 10.10.10.1 en un DVIPA distribuido, debe coincidir
; con SYS2
  SYSPLEXPORTS ; Prerrequisito de RDz
  DISTMETHOD BASEWLM ; BASEWLM o ROUNDROBIN
  10.10.10.1 ; Dirección DVIPA utilizada por clientes RDz
  PORT 4035 ; Puerto utilizado por clientes RDz
  DESTIP 9.9.9.1 9.9.9.2 ; RDz activo en SYS1 y SYS2
ENDVIPADYNAMIC
```

Sistema SYS2 – perfil de TCP/IP

```
IPCONFIG
  SYSPLEXRouting
; SYSPLEXROUTING es necesario ya que esta pila necesita comunicación sysplex
  DYNAMICXCF 9.9.9.2 255.255.255.0 1
; DYNAMICXCF define el dispositivo/enlace con la dirección inicial 9.9.9.2
; según sea necesario
  IGNORERedirect

VIPADYNAMIC
  VIPABACKUP 255.255.255.0 10.10.10.1
; VIPABACKUP define 10.10.10.1 como DVIPA de seguridad en SYS2 para RDz
  VIPADISTRIBUTE DEFINE
; VIPADISTRIBUTE convierte 10.10.10.1 en un DVIPA distribuido, debe coincidir
; con SYS1
  SYSPLEXPORTS ; Prerrequisito de RDz
  DISTMETHOD BASEWLM ; BASEWLM o ROUNDROBIN
  10.10.10.1 ; Dirección DVIPA utilizada por clientes RDz
  PORT 4035 ; Puerto utilizado por clientes RDz
  DESTIP 9.9.9.1 9.9.9.2 ; RDz activo en SYS1 y SYS2
ENDVIPADYNAMIC
```

Capítulo 4. Consideraciones sobre WLM

Al contrario que las aplicaciones z/OS tradicionales, Developer for System z no es una aplicación monolítica que se pueda identificar fácilmente para el Gestor de carga de trabajo (WLM). Developer for System z está formado por varios componentes que interactúan para proporcionar al cliente acceso a los servicios y datos del host. Tal como se describe en Capítulo 1, “Comprender Developer for System z”, en la página 3, algunos de estos servicios están activos en diferentes espacios de direcciones, lo que resulta en diferentes clasificaciones WLM.

En este capítulo se tratan estos temas:

- “Clasificación de carga de trabajo”
- “Establecimiento de objetivos” en la página 75

Clasificación de carga de trabajo

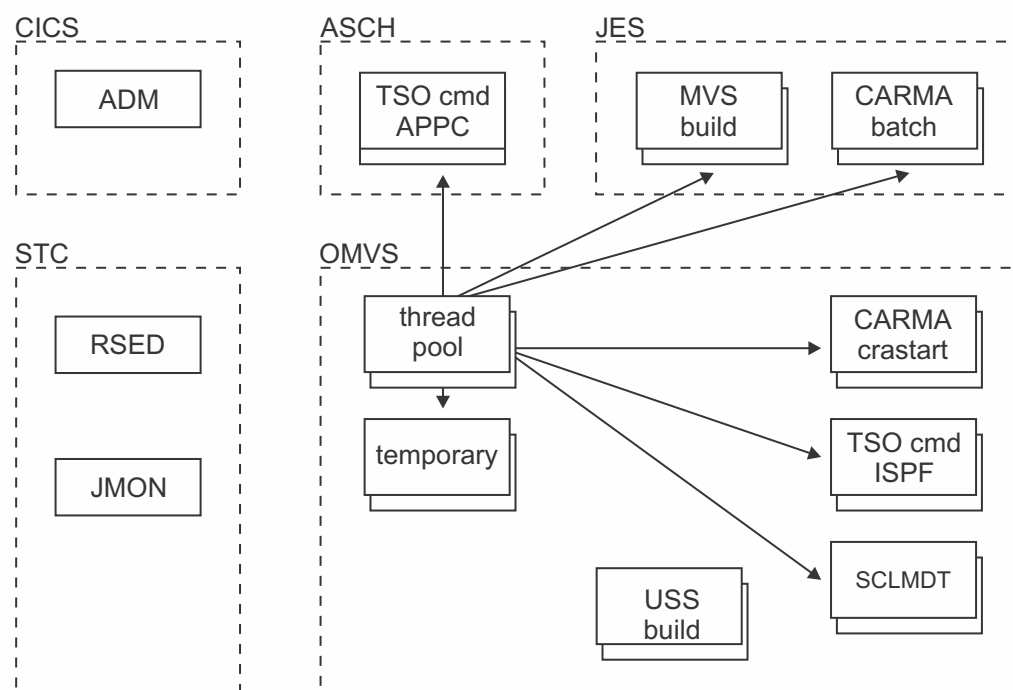


Figura 13. Clasificación de WLM

La Figura 13 muestra una visión general básica de los subsistemas a través de los cuales se presentan las cargas de trabajo de Developer for System z a WLM.

El Gestor de despliegue de aplicaciones (ADM) está activo en una región CICS y por lo tanto seguirá las reglas de clasificación CICS en WLM.

El daemon RSE (RSED), el Gestor de depuración (DBGMGR) y el Supervisor de trabajos JES (JMON) son tareas iniciadas por Developer for System z (o trabajos por lotes de larga ejecución), cada uno con su espacio de direcciones individual.

Tal como se explica en “RSE como aplicación Java” en la página 5, el daemon RSE genera un proceso hijo para cada servidor de agrupaciones de hebras RSE (que soporta un número variable de clientes). Cada agrupación de hebras está activo en un espacio de direcciones aparte (utilizando un iniciador z/OS UNIX, BPXAS). Como se trata de procesos generados, se clasifican mediante las reglas de clasificación WLM OMVS, no las reglas de clasificación de tareas iniciadas.

Los clientes activos en una agrupación de hebras pueden crear muchos otros espacios de direcciones, según las acciones realizadas por los usuarios. Dependiendo de la configuración de Developer for System z, algunas cargas de trabajo, como por ejemplo el servicio de mandatos TSO (TSO cmd) o CARMA, se pueden ejecutar en subsistemas diferentes.

Los espacios de direcciones que aparecen en la lista de la Figura 13 en la página 73 siguen permaneciendo en el sistema durante tiempo suficiente para ser visibles pero debe tener en cuenta que debido al diseño de z/OS UNIX, también hay varios espacios de direcciones temporales de vida breve. Estos espacios de direcciones temporales están activos en el subsistema OMVS.

Tenga en cuenta que mientras que las agrupaciones de hebras de RSE utilizan el mismo ID de usuario y un nombre de trabajo parecido al del daemon RSE, todos los espacios de direcciones iniciados por una agrupación de hebras son propiedad del IDE de usuario del cliente que solicita la acción. El ID de usuario cliente también se utiliza como (es parte del) nombre de trabajo para todos los espacios de direcciones basados en OMVS declarados por la agrupación de hebras.

Otros servicios han creado más espacios de direcciones que Developer for System z utiliza, como por ejemplo Gestor de archivos (FMNCAS) o z/OS UNIX REXEC (construcción USS).

Reglas de clasificación

WLM utiliza reglas de clasificación para correlacionar el trabajo que entra en el sistema con una clase de servicio. Esta clasificación se basa en calificadores de trabajo. El primer calificador (obligatorio) es el tipo de subsistema que recibe la petición de trabajo. La Tabla 14 enumera los tipos de subsistema que pueden recibir cargas de trabajo de Developer for System z.

Tabla 14. Subsistemas de punto de entrada de WLM

Tipo de subsistema	Descripción de trabajo
ASCH	Las peticiones de trabajo incluyen todos los programas de transacción APPC planificados por el planificador de transacciones APPC/MVS proporcionados por IBM, ASCH.
CICS	Las peticiones de trabajo incluyen todas las transacciones procesadas por CICS.
JES	Las solicitudes de trabajo incluyen todos los trabajos iniciados por JES2 o JES3.
OMVS	Las peticiones de trabajo incluyen el trabajo procesado en espacios de direcciones hijo bifurcados de z/OS UNIX System Services.
STC	Las peticiones de trabajo incluyen todo el trabajo iniciado por los mandatos START y MOUNT. STC también incluye espacios de direcciones de componentes del sistema.

En la Tabla 15 se enumeran calificadores adicionales que se pueden utilizar para asignar una carga de trabajo a una clase de servicio específica. Consulte la planificación de MVS: Workload Management (SA22-7602) para obtener más detalles sobre los calificadores de trabajo de la lista.

Tabla 15. Calificadores de trabajo de WLM

		ASCH	CICS	JES	OMVS	STC
AI	Información de contabilidad	x		x	x	x
LU	Nombre de LU (*)		x			
PF	Realizar (*)			x		x
PRI	Prioridad			x		
SE	Nombre de entorno de planificación			x		
SSC	Nombre de recogida de subsistema			x		
SI	Instancia de subsistema (*)		x	x		
SPM	Parámetro de subsistema					x
PX	Nombre de Sysplex	x	x	x	x	x
SY	Nombre de sistema (*)	x			x	x
TC	Clase de transacción/trabajo (*)	x		x		
TN	Nombre de transacción/trabajo (*)	x	x	x	x	x
UI	ID de usuario (*)	x	x	x	x	x

Nota: Para los calificadores marcados con (*), puede especificar grupos de clasificación añadiendo una G a la abreviación de tipo. Por ejemplo, un grupo de nombres de transacción sería TNG.

Establecimiento de objetivos

Tal como se explica en “Clasificación de carga de trabajo” en la página 73, Developer for System z crea varios tipos de cargas de trabajo en el sistema. Estas diferentes tareas se comunican entre sí, lo que implica que el tiempo transcurrido real se vuelve importante para evitar los problemas de tiempo de espera para las conexiones entre las tareas. Como resultado, las tareas de Developer for System z deben colocarse en clases de servicio de alto rendimiento o en clases de servicio de rendimiento moderado con una alta prioridad.

Por lo tanto, es recomendable revisar y posiblemente actualizar sus objetivos de WLM. Esto es especialmente cierto para las tiendas MVS tradicionales sin experiencia con cargas de trabajo OMVS para las que el tiempo es muy importante.

Nota:

- La información de objetivo de esta sección se mantiene deliberadamente a un nivel descriptivo porque los objetivos de rendimiento reales son muy específicos del sitio.
- Para ayudarle a comprender el impacto de una tarea específica en el sistema, se utilizan términos como utilización de recursos mínima, moderada y sustancial. Todos ellos son relativos a la utilización de recursos total de Developer for System z, no de todo el sistema.

La Tabla 16 enumera los espacios de direcciones utilizados por Developer for System z. z/OS UNIX sustituirá "x" en la columna "Nombre de tarea" por un número aleatorio de 1 dígito.

Tabla 16. Cargas de trabajo WLM

Descripción	Nombre de tarea	Carga de trabajo
Gestor de depuración	DBGMGR	STC
Supervisor de trabajos JES	JMON	STC
Daemon RSE	RSED	STC
Agrupación de hebras RSE	RSEDx	OMVS
Pasarela de cliente ISPF (Servicio de mandatos TSO y SCLMDT)	<IDusuario>x	OMVS
Servicio de mandatos TSO (APPC)	FEKFRSRV	ASCH
CARMA (por lotes)	CRA<puerto>	JES
CARMA (crastart)	<IDusuario>x	OMVS
CARMA (Pasarela de cliente ISPF)	<IDusuario> e <IDusuario>x	OMVS
Construcción de MVS (trabajo por lotes)	*	JES
Construcción de z/OS UNIX (mandatos de shell)	<IDusuario>x	OMVS
Shell de z/OS UNIX	<IDusuario>	OMVS
Gestor de despliegue de aplicaciones	CICSTS	CICS

Consideraciones para la selección de objetivos

Las consideraciones de WLM generales siguientes le pueden ayudar a definir adecuadamente las definiciones de objetivos correctas para Developer for System z:

- Debe basar los objetivos en lo que se puede conseguir realmente, no en lo que desea que ocurra. Si establece los objetivos más arriba de lo necesario, WLM mueve recursos de trabajos menos importantes a trabajos más importantes que realmente no necesitan los recursos.
- Limite la cantidad de trabajo asignada a las clases de servicio SYSTEM y SYSSTC, porque estas clases tienen una prioridad de despacho más alta que cualquier clase gestionada por WLM. Utilice estas clases para el trabajo que tenga más importancia pero que utilice menos CPU.
- El trabajo que queda entre las reglas de clasificación termina en la clase SYSOTHER, que tiene un objetivo discrecional. Un objetivo discrecional indica a WLM que haga lo mejor que pueda cuando el sistema tenga recursos de sobra.

Cuándo utilizar los objetivos de tiempo de respuesta:

- Debe haber una cadencia de llegada de tareas constante (al menos 10 tareas en 20 minutos) para que WLM gestione adecuadamente un objetivo de tiempo de respuesta.
- Utilice los objetivos de tiempo de respuesta sólo para cargas de trabajo bien controladas porque una sola transacción larga tiene un impacto grande en el tiempo de respuesta promedio y puede hacer que WLM reaccione de manera exagerada.

Cuándo utilizar métodos de velocidad:

- Normalmente no puede conseguir un método de velocidad mayor del 90% por varias razones. Por ejemplo, todos los espacios de direcciones SYSTEM y SYSSTC tienen una prioridad de despacho mayor que cualquier objetivo de tipo de velocidad.
- WLM utiliza un número mínimo de ejemplos (utilizar y retardar) sobre los que basar sus decisiones de objetivo de velocidad. Así, cuanto menos trabajo se esté ejecutando en una clase de servicio, más se tardará en recoger el número necesario de ejemplos y en ajustar la política de despacho.
- Vuelva a evaluar los métodos de velocidad cuando cambie el hardware. Además, pasar a menos y más rápidos procesadores necesita cambios en los objetivos de velocidad.

STC

Todas las tareas iniciadas por Developer for System z, daemon de RSE y Supervisor de trabajos JES están dando servicio en tiempo real a peticiones de cliente.

Tabla 17. Cargas de trabajo WLM - STC

Descripción	Nombre de tarea	Carga de trabajo
Supervisor de trabajos JES	JMON	STC
Gestor de depuración	DBGMGR	STC
Daemon RSE	RSED	STC

- Supervisor de trabajos JES
El Supervisor de trabajos JES proporciona todos los servicios relacionados con JES como por ejemplo el sometimiento de trabajos, el examen de archivos en spool y la ejecución de mandatos de operador JES. Debe especificar un objetivo de velocidad de un periodo y alto rendimiento porque la tarea no comunica las transacciones individuales a WLM. La utilización de recursos depende mucho de las acciones de usuario y por lo tanto fluctuará, pero se espera que sea entre mínima y moderada.
- Gestor de depuración
El Gestor de depuración proporciona servicios para conectar programas que se depuran a clientes que los depuran. Debe especificar un objetivo de velocidad de un periodo y alto rendimiento porque la tarea no comunica las transacciones individuales a WLM. La utilización de recursos depende mucho de las acciones de usuario y por lo tanto fluctuará, pero se espera que sea mínima.
- Daemon RSE
El daemon RSE maneja el inicio de sesión y la autenticación de clientes y gestiona las diferentes agrupaciones de hebras de RSE. Debe especificar un objetivo de velocidad de un periodo y alto rendimiento porque la tarea no comunica las transacciones individuales a WLM. Se espera que la utilización de recursos sea moderada, con un pico al principio del día de trabajo.

OMVS

Las cargas de trabajo de OMVS pueden dividirse en dos grupos, agrupaciones de hebras RSE y todo lo demás. Esto es porque todas las cargas de trabajo excepto las agrupaciones de hebras RSE utilizan el ID de usuario de cliente como base para el nombre del espacio de direcciones. (z/OS UNIX sustituirá "x" en la columna "Nombre de tarea" por un número aleatorio de 1 dígito.)

Tabla 18. Cargas de trabajo WLM - OMVS

Descripción	Nombre de tarea	Carga de trabajo
Agrupación de hebras RSE	RSEDx	OMVS
Pasarela de cliente ISPF (Servicio de mandatos TSO y SCLMDT)	<IDusuario>x	OMVS
CARMA (crastart)	<IDusuario>x	OMVS
CARMA (Pasarela de cliente ISPF)	<IDusuario> e <IDusuario>x	OMVS
Construcción de z/OS UNIX (mandatos de shell)	<IDusuario>x	OMVS
Shell de z/OS UNIX	<IDusuario>	OMVS

- Agrupación de hebras RSE

Una agrupación de hebras RSE es como el corazón y el cerebro Developer for System z. Casi todos los datos fluyen por aquí, y los extractores (hebras específicas de usuario) de dentro de la agrupación de hebras controlan las acciones para la mayoría de las otras tareas relacionadas de Developer for System z. Debe especificar un objetivo de velocidad de un periodo y alto rendimiento porque la tarea no comunica las transacciones individuales a WLM. La utilización de recursos depende mucho de las acciones de usuario y por lo tanto fluctuará, pero se espera que sea sustancial.

Las cargas de trabajo restantes finalizarán todas en la misma clase de servicio debido a un convenio de denominación de espacios de direcciones común. Debe especificar un objetivo de varios periodos para esta clase de servicio. Los primeros periodos deberían ser objetivos de tiempo de respuesta percentil de alto rendimiento, mientras que el último periodo debería tener un objetivo de velocidad de rendimiento moderado. Algunas cargas de trabajo como por ejemplo la Pasarela de cliente ISPF informarán de transacciones individuales a WLM, mientras que otras no lo harán.

- Pasarela de cliente ISPF

La Pasarela de cliente ISPF es un servicio ISPF invocado por Developer for System z para ejecutar mandatos TSO y ISPF no interactivos. Esto incluye mandatos explícitos emitidos por el cliente así como mandatos implícitos emitidos por Developer for System z, como por ejemplo obtener una lista de miembros de PDS. La utilización de recursos depende mucho de las acciones de usuario y por lo tanto fluctuará, pero se espera que sea mínima.

- CARMA

CARMA es un servidor Developer for System z opcional utilizado para interactuar con Gestores de configuraciones de software (SCM), como por ejemplo CA Endevor® SCM. Developer for System z permite diferentes métodos de inicio para un servidor CARMA, algunos de los cuales se convierten en una carga de trabajo OMVS. La utilización de recursos depende mucho de las acciones de usuario y por lo tanto fluctuará, pero se espera que sea mínima.

- Construcción de z/OS UNIX

Cuando un cliente inicia una construcción para un proyecto z/OS UNIX, z/OS UNIX REXEC (o SSH) iniciará una tarea que ejecuta un número de mandatos de shell z/OS UNIX para realizar la construcción. La utilización de recursos depende mucho de las acciones de usuario y por lo tanto fluctuará, pero se espera que sea entre moderada y sustancial, dependiendo del tamaño del proyecto.

- Shell de z/OS UNIX

Esta carga de trabajo procesa mandatos shell z/OS UNIX emitidos por el cliente. La utilización de recursos depende mucho de las acciones de usuario y por lo tanto fluctuará, pero se espera que sea mínima.

JES

Los procesos por lotes gestionados por JES los utiliza Developer for System z de varias maneras. La utilización más común es para construcciones MVS, donde un trabajo se somete y supervisa para determinar cuándo finaliza. Pero Developer for System z también podría iniciar un servidor CARMA por lotes y comunicarse con el mediante TCP/IP.

Tabla 19. Cargas de trabajo WLM - JES

Descripción	Nombre de tarea	Carga de trabajo
CARMA (por lotes)	CRA<puerto>	JES
Construcción de MVS (trabajo por lotes)	*	JES

- CARMA

CARMA es un servidor Developer for System z opcional utilizado para interactuar con Gestores de configuraciones de software (SCM), como por ejemplo CA Endevor® SCM. Developer for System z permite diferentes métodos de inicio para un servidor CARMA, algunos de los cuales se convierten en una carga de trabajo JES. Debe especificar un objetivo de velocidad de un periodo y alto rendimiento porque la tarea no comunica las transacciones individuales a WLM. La utilización de recursos depende mucho de las acciones de usuario y por lo tanto fluctuará, pero se espera que sea mínima.

- Construcción de MVS

Cuando un cliente inicia una construcción para un proyecto MVS, Developer for System z iniciará un trabajo por lotes para realizar la construcción. La utilización de recursos depende mucho de las acciones de usuario y por lo tanto fluctuará, pero se espera que sea entre moderada y sustancial, dependiendo del tamaño del proyecto. En función de sus circunstancias locales, puede ser aconsejable seguir diferentes estrategias de objetivo de rendimiento moderado.

- Debe especificar un objetivo de varios periodos con un periodo de tiempo de respuesta percentil y un periodo de velocidad final. En este casos, sus desarrolladores deben utilizar principalmente el mismo procedimiento de construcción y archivos de entrada de tamaños parecidos para crear trabajos con tiempos de respuesta uniformes. Debe haber también una cadencia de llegada de trabajos constante (al menos 10 trabajos en 20 minutos) para que WLM gestione adecuadamente un objetivo de tiempo de respuesta.
- Un objetivo de velocidad se ajusta mejor a los trabajos por lotes porque este objetivo puede manejar tiempos de ejecución y cadencias de llegada muy variables.

ASCH

En las versiones actuales Developer for System z, la Pasarela de cliente ISPF se utiliza para ejecutar mandatos TSO e ISPF no interactivos. Por razones históricas, Developer for System z también soporta la ejecución de estos mandatos a través de una transacción APPC. Debe tener en cuenta que el método APPC está en desuso.

Tabla 20. Cargas de trabajo WLM - ASCH

Descripción	Nombre de tarea	Carga de trabajo
Servicio de mandatos TSO (APPC)	FEKFRSRV	ASCH

- Servicio de mandatos TSO

El servicio de mandatos TSO puede iniciarse como una transacción APPC por parte de Developer for System z para ejecutar mandatos TSO e ISPF no interactivos. Esto incluye mandatos explícitos emitidos por el cliente así como mandatos implícitos emitidos por Developer for System z, como por ejemplo obtener una lista de miembros de PDS. Debe especificar un objetivo de varios periodos para esta clase de servicio. Para los primeros periodos debe especificar objetivos de tiempo de respuesta percentil de alto rendimiento. Para el último periodo debe especificar un objetivo de velocidad de rendimiento moderado. La utilización de recursos depende mucho de las acciones de usuario y por lo tanto fluctuará, pero se espera que sea mínima.

CICS

El Gestor de despliegue de aplicaciones es un servidor de Developer for System z opcional que está activo dentro de una región de CICS Transaction Server.

Tabla 21. Cargas de trabajo de WLM - CICS

Descripción	Nombre de tarea	Carga de trabajo
Gestor de despliegue de aplicaciones	CICSTS	CICS

- Gestor de despliegue de aplicaciones

El servidor Gestor de despliegue de aplicaciones que está activo dentro de una región CICSTS, permite descargar de forma segura las tareas de gestión CICSTS seleccionadas para los desarrolladores. La utilización de recursos depende mucho de las acciones de usuario y por lo tanto fluctuará, pero se espera que sea mínima. El tipo de clase de servicio que utilice depende del resto de transacciones activas en esta región CICS y por lo tanto no se trata en detalle.

WLM soporta varios tipos de gestión que puede utilizar para CICS:

- Gestionar CICS hacia un objetivo de región

El objeto se establece en una clase de servicio que gestiona los espacios de dirección CICS. Sólo puede utilizar un objetivo de velocidad de ejecución para esta clase de servicio. WLM utiliza las reglas de clasificación de JES o STC para los espacios de dirección pero no utiliza las reglas de clasificación de subsistemas CICS para transacciones.

- Gestionar CICS hacia un objetivo de tiempo de respuesta

Se puede establecer un objetivo de tiempo de respuesta en una clase de servicio asignada a una sola transacción o a un grupo de transacciones. WLM utiliza las reglas de clasificación JES o STC para los espacios de dirección y las reglas de clasificación de subsistema CICS para transacciones.

Capítulo 5. Consideraciones acerca de los ajustes

Tal como se explica en Capítulo 1, “Comprender Developer for System z”, en la página 3, RSE (Explorador de Sistemas remotos) es el núcleo de Developer for System z. Para gestionar las conexiones y cargas de trabajo de los clientes, RSE está formado por un espacio de direcciones de daemon, que controla los espacios de direcciones de agrupaciones de hebras. El daemon actúa como punto focal a efectos de conexión y gestión, mientras que las agrupaciones de hebras procesan las cargas de trabajo del cliente.

Ello hace que RSE sea el destino principal para ajustar la configuración de Developer for System z. Sin embargo, para mantener a cientos de usuarios, cada uno de los cuales utiliza 17 o más hebras, una cantidad determinada de almacenamiento y, posiblemente, 1 o más espacios de direcciones es necesario configurar correctamente Developer for System z y z/OS.

En este capítulo se tratan estos temas:

- “Uso de recursos”
- “Uso de almacenamiento” en la página 98
- “Uso de espacio del sistema de archivos de z/OS UNIX” en la página 105
- “Definiciones de recursos clave” en la página 108
- “definiciones de varios recursos” en la página 112
- “Supervisión” en la página 114
- “Configuración de ejemplo” en la página 118

Uso de recursos

Utilice la información de esta sección para estimar el uso normal y máximo de recursos por parte de Developer for System z, de manera que pueda planificar acorde la configuración del sistema.

Al utilizar los números y las fórmulas presentadas en esta sección para definir los valores para los límites del sistema, tenga en cuenta que está trabajando con estimaciones bastante precisas. Deje un margen suficiente al establecer los límites del sistema para permitir el uso de recursos por las tareas temporales o por otras tareas, o por usuarios que se conecten varias veces al host simultáneamente. (Por ejemplo, a través de RSE y TN3270).

Nota:

- El ámbito de la información está limitado a los servicios a los que se accede a través de RSE proporcionados por el propio Developer for System z. Por ejemplo, el uso de recursos de TN3270 no está documentado (no se accede a través de RSE), como tampoco lo está el uso de recursos de los programas llamados durante construcciones remotas (basadas en host) de proyectos de MVS o z/OS UNIX (no proporcionados por Developer for System z).
- Añadir extensiones externas a Developer for System z puede aumentar los contadores de uso de recursos.
- Todos los servicios tienen tareas de “mantenimiento” cortas, que utilizan recursos durante su ejecución, y que pueden ejecutarse secuencialmente o paralelamente. Los recursos utilizados por estas tareas no están documentados.

- El uso de recursos específicos del usuario del software requisito, como la Pasarela de cliente ISPF, se documenta cuando es útil.
- Los números que se presentan aquí pueden cambiar sin notificación previa.

Visión general

Las tablas siguientes proporcionan una visión general del número de espacios de dirección, procesos y hebras utilizados por Developer for System z. Encontrará más detalles de los números presentados aquí en las secciones siguientes:

- “Recuento de espacios de direcciones” en la página 83
- “Recuento de procesos” en la página 86
- “Recuento de hebras” en la página 89

La Tabla 22 proporciona una visión general de los recursos clave utilizados por las tareas iniciadas de Developer for System z. Estos recursos se asignan únicamente una vez. Todos los clientes de Developer for System z los comparten.

Tabla 22. Uso de recursos comunes

Tarea iniciada	Espacios de direcciones	Procesos	Hebras
JMON	1	1	3
DBGMGR	1	1	4
RSED	1	3	16
RSEDx	(a) 1 + 2	1 + 3	1 + 14

Nota: (a) Existe un espacio de direcciones con autorización APF y como mínimo 1 agrupación de hebras RSE, que consta de dos espacios de direcciones. Consulte “Recuento de espacios de direcciones” en la página 83 para determinar el número real de espacios de direcciones de agrupaciones de hebras RSE.

La Tabla 23 proporciona una visión general de los recursos clave utilizados por el software de seguridad. Estos recursos se asignan para cada cliente de Developer for System z que invoque la función relacionada.

Tabla 23. Uso de recursos requisito específicos del usuario

Software requisito	Espacios de direcciones	Procesos	Hebras
Pasarela de cliente ISPF	1	2	4
APPC	1	1	2

La Tabla 24 proporciona una visión general de los recursos de clave utilizados por cada cliente de Developer for System z al ejecutar la función especificado. Los valores no numéricos, como ISPF, son una referencia al valor correspondiente de la Tabla 23.

Tabla 24. Uso de recursos específicos del usuario

Acción de usuario	Espacios de direcciones	Procesos	Hebras		
	ID usuario	ID usuario	ID usar.	RSEDx	JMON
Inicio de sesión	-	-	-	17	1

Tabla 24. Uso de recursos específicos del usuario (continuación)

Acción de usuario	Espacios de direcciones	Procesos	Hebras		
	ID usuario	ID usuario	ID usar.	RSEDx	JMON
Temporizador para tiempo de espera desocupado	-	-	-	1	-
Buscar	-	-	-	1	-
Ampliar PDS(E)	ISPF	ISPF	ISPF	-	-
Abrir conjunto de datos	ISPF	ISPF	ISPF	1	-
Mandato TSO	ISPF	ISPF	ISPF	-	-
Shell de z/OS UNIX	1	1	1	6	-
Construcción de MVS	1	-	-	-	-
Construcción de z/OS UNIX	3	3	3	-	-
CARMA (por lotes)	1	1	2	1	-
CARMA (crastart)	1	1	2	1	-
CARMA (crastart con rastreo)	3	1+1+2	1+1+1+2	2	-
CARMA (ispf)	4	4	7	5	-
SCLMDT	ISPF	ISPF	ISPF	-	-

Nota: ISPF se puede sustituir por APPC, excepto por SCLM Developer Toolkit.

Recuento de espacios de direcciones

La Tabla 25 lista los espacios de direcciones utilizados por Developer for System z, donde “u” en la columna “Recuento” indica que la cantidad se debe multiplicar por el número de usuarios activos que utilizan la función simultáneamente. z/OS UNIX sustituirá “x” de la columna “Nombre de tarea” por un número de 1 dígito aleatorio.

Tabla 25. Recuento de espacios de direcciones

Recuento	Descripción	Nombre de tarea	Compartido	Finaliza tras
1	Supervisor de trabajos JES	JMON	Sí	Nunca
1	Gestor de depuración	DBGMGR	Sí	Nunca
1	Daemon RSE	RSED	Sí	Nunca
1	Autorización APF daemon RSE	RSEDx	Sí	Nunca

Tabla 25. Recuento de espacios de direcciones (continuación)

Recuento	Descripción	Nombre de tarea	Compartido	Finaliza tras
(a)	Agrupación de hebras RSE	RSEDx	Sí	Nunca
(a)	Autorización APF de agrupación de hebras RSE	RSEDx	Sí	Nunca
1u	Pasarela de cliente ISPF (Servicio de mandatos TSO y SCLMDT)	<IDusuario>x	No	15 minutos o fin de sesión de usuario
1u	Servicio de mandatos TSO (APPC)	FEKFRSRV	No	60 minutos o fin de sesión de usuario
1u	CARMA (por lotes)	CRA<puerto>	No	7 minutos o fin de sesión de usuario
1u	CARMA (crastart)	<IDusuario>x	No	7 minutos o fin de sesión de usuario
3u	CARMA (crastart con rastreo) (c)	<IDusuario> e <IDusuario>x	No	7 minutos o fin de sesión de usuario
4u	CARMA (ispf, en desuso)	(1)<IDusuario> o (3)<IDusuario>x	No	7 minutos o fin de sesión de usuario
(b)	Uso simultáneo de la Pasarela de cliente ISPF por 1 usuario	<IDusuario>x	No	Compleción de la tarea
1u	Construcción de MVS (trabajo por lotes)	*	No	Compleción de la tarea
3u	Construcción de z/OS UNIX (mandatos de shell)	<IDusuario>x	No	Compleción de la tarea
1u	Shell de z/OS UNIX	<IDusuario>	No	Fin de sesión de usuario

Nota:

- (a) Hay, como mínimo, un espacio de direcciones de agrupaciones de hebras RSE activo. EL número real depende de:
 - La directiva `minimum.threadpool.process` de `rsed.envvars`. El valor predeterminado es 1.
 - El número de usuarios a los que una agrupación de hebras puede proporcionar servicios. El objetivo de los valores predeterminados es de 30 usuarios por agrupación de hebras.

Nota: Si la directiva `single.logon` está activa, habrá como mínimo dos agrupaciones de hebras iniciadas, aun cuando el valor de `minimum.threadpool.process` sea 1. El valor predeterminado para `single.logon` en `rsed.envvars` es `active`.

- (b) Developer for System z tiene varias hebras activas por usuario. En caso de que el espacio de direcciones de la Pasarela de cliente ISPF no haya terminado de servir la solicitud de una hebra cuando otra hebra manda una solicitud, ISPF iniciará una Pasarela de cliente nueva para procesar la nueva solicitud. Este espacio de direcciones finaliza tras la compleción de tareas.
- (c) El rastreo de inicio de crastart de CARMA está controlador por el nivel de depuración activa de RSE para `rsecomm.log`.
- SCLMDT requiere un espacio de direcciones de Pasarela de cliente ISPF. SCLMDT comparte el espacio de direcciones con el servicio de mandatos TSO.
- La mayor parte de acciones relacionadas con conjuntos de datos de MVS utilizan el servicio de mandatos TSO, que puede estar activo en la Pasarela de cliente ISPF o en una transacción APPC, respectivamente.

Utilice la fórmula de la Figura 14 para estimar el número máximo de espacios de direcciones utilizado por Developer for System z.

$$3 + 2 * A + N * (x + y + z) + (2 + N * 0.01)$$

Figura 14. Número máximo de espacios de direcciones

Donde

- “3” iguala el número de espacios de direcciones del servidor activo permanente.
- “A” representa el número de espacios de direcciones de agrupaciones de hebras RSE.
- “N” representa el número máximo de usuarios simultáneos.
- “x” es uno de los siguientes valores, dependiendo de las opciones de configuración seleccionadas.

X	SCLMDT	TSO a través de la Pasarela de cliente	TSO a través de APPC
1	No	No	Sí
1	No	Sí	No
1	Sí	Sí	No

- “y” es uno de los siguientes valores, dependiendo de las opciones de configuración seleccionadas.

Y	
0	No CARMA
1	CARMA (por lotes)
1	CARMA (crastart)
3	CARMA (crastart con rastreo)
4	CARMA (ispf, en desuso)

- “z” es 0 de forma predeterminada, pero puede aumentar dependiendo de las acciones de usuario:
 - Añada 1 cuando se realice una construcción de MVS. Estos espacios de direcciones finalizan cuando se completa la tarea de construcción relacionada (un trabajo por lotes).
 - Añada 3 cuando se realice una construcción de z/OS UNIX. Tenga en cuenta que el número real puede ser superior, dependiendo de las necesidades de los programas invocados. Estos espacios de direcciones finalizan cuando se completa la tarea de construcción relacionada.
- “2 + N*0.01” añade un almacenamiento intermedio para los espacios de direcciones temporales. El tamaño del almacenamiento intermedio necesario puede ser distinto en su sitio.

Utilice la fórmula de la Figura 15 en la página 86 para estimar el número máximo de espacios de direcciones utilizados por un cliente de Developer for System z (sin contar los espacios de direcciones temporales no documentados).

$$x + y + z$$

Figura 15. Número de espacios de direcciones por cliente

Donde

- "x" depende de las opciones de configuración seleccionadas y se documenta para que la fórmula calcule el número máximo de espacios de direcciones (Figura 14 en la página 85).
- "y" depende de las opciones de configuración seleccionadas y se documenta para que la fórmula calcule el número máximo de espacios de direcciones (Figura 14 en la página 85).
- "z" es 0 de forma predeterminada, pero puede aumentar dependiendo de las acciones de usuario, ya que se documenta para que la fórmula calcule el número máximo de espacios de direcciones (Figura 14 en la página 85).

Las definiciones de la Tabla 26 pueden limitar el número real de espacios de direcciones.

Tabla 26. Límites de espacios de direcciones

Ubicación	Límite	Recursos afectados
rsed.envvars	maximum.threadpool.process	Limita el número de agrupaciones de hebras RSE
IEASYMxx	MAXUSER	Limita el número de espacios de direcciones
ASCHPMxx	MAX	Limita el número de iniciadores APPC para el servicio de mandatos TSO (APPC)

Recuento de procesos

La Tabla 27 lista el número de procesos por espacio de direcciones utilizado por Developer for System z. "u" en la columna "Hebras" indica que la cantidad se debe multiplicar por el número de usuarios activos que utilizan la función simultáneamente.

Tabla 27. Recuento de procesos

Procesos	Espacios de direcciones	Descripción	ID de usuario
1	1	Supervisor de trabajos JES	STCJMON
1	1	Gestor de depuración	STCDBM
3	1	Daemon RSE	STCRSE
1	1	Autorización APF daemon de RSE	STCRSE
2	(a)	Agrupación de hebras RSE	STCRSE
1	(a)	Autorización APF de agrupación de hebras RSE	STCRSE
2	(b)	Pasarela de cliente ISPF (Servicio de mandatos TSO y SCLMDT)	<IDusuario>
2	(a)	Agrupación de hebras RSE	STCRSE
1	1u	Servicio de mandatos TSO (APPC)	<IDusuario>
1	1u	CARMA (por lotes)	<IDusuario>
1	1u	CARMA (crastart)	<IDusuario>

Tabla 27. Recuento de procesos (continuación)

Procesos	Espacios de direcciones	Descripción	ID de usuario
1+1+2	3u	CARMA (crastart con rastreo) (c)	<IDusuario>
1	1u	CARMA (ispf, en desuso)	<IDusuario>
1	3u	Construcción de z/OS UNIX (mandatos de shell)	<IDusuario>
1	1u	Shell de z/OS UNIX	<IDusuario>
(5)	(u)	SCLM Developer Toolkit	<IDusuario>

Nota:

- (a) Hay, como mínimo, un espacio de direcciones de agrupaciones de hebras RSE activo. Consulte “Recuento de espacios de direcciones” en la página 83 para determinar el número real de espacios de direcciones de agrupaciones de hebras RSE.
- El daemon RSE y todas las agrupaciones de hebras RSE utilizan el mismo ID de usuario.
- (b) En los casos normales, y cuando se utilizan las opciones de configuración predeterminadas, hay 1 Pasarela de cliente ISPF activa por usuario. El número real puede variar, tal como se describe en “Recuento de espacios de direcciones” en la página 83.
- (c) El rastreo de inicio de CRASTART de CARMA está controlador por el nivel de depuración activa de RSE para rsecomm.log.
- SCLMDT requiere un espacio de direcciones de Pasarela de cliente ISPF. SCLMDT comparte el espacio de direcciones con el servicio de mandatos TSO.
- (u) Los procesos SCLMDT se ejecutan en el espacio de direcciones de la Pasarela de cliente ISPF, por lo que no tienen un valor para el recuento de espacios de direcciones.
- Los procesos SCLMDT son temporales y finalizan cuando se completan las tareas, pero puede haber varios procesos activos simultáneamente para un único usuario. La Tabla 27 en la página 86 enumera el número máximo de procesos SCLMDT simultáneos.
- La mayor parte de acciones relacionadas con conjuntos de datos de MVS utilizan el servicio de mandatos TSO, que puede estar activo en la Pasarela de cliente ISPF o en una transacción APPC, respectivamente.
- Una construcción de z/OS UNIX utiliza tres procesos en total, y cada uno de ellos se ejecuta en su propio espacio de direcciones.
- Todos los procesos enumerados permanecen activos hasta que el espacio de direcciones relacionado finaliza, a menos que se indique lo contrario.

Utilice la fórmula de la Figura 16 para estimar el número máximo de procesos utilizado por Developer for System z.

$$6 + 3 * A + N * (x + y + z) + (10 + N * 0.05)$$

Figura 16. Número máximo de procesos

Donde

- "6" equivale al número de procesos utilizados por los espacios de direcciones del servidor activo.
- "A" representa el número de espacios de direcciones de agrupaciones de hebras RSE.
- "N" representa el número máximo de usuarios simultáneos.
- "x" es uno de los siguientes valores, dependiendo de las opciones de configuración seleccionadas.

X	SCLMDT	TSO a través de la Pasarela de cliente	TSO a través de APPC
1	No	No	Sí
2	No	Sí	No
7	Sí	Sí	No

- "y" es uno de los siguientes valores, dependiendo de las opciones de configuración seleccionadas.

Y	
0	No CARMA
1	CARMA (por lotes)
1	CARMA (crastart)
4	CARMA (crastart con rastreo)
4	CARMA (ispf, en desuso)

- "z" es 0 de forma predeterminada, pero puede aumentar dependiendo de las acciones de usuario:
 - Añada 1 cuando se abra un shell de z/OS UNIX. Este proceso permanece activo hasta que el usuario finaliza la sesión.
 - Añada 3 cuando se realice una construcción de z/OS UNIX. Tenga en cuenta que el número real puede ser superior, dependiendo de las necesidades de los programas invocados. Estos procesos finalizan cuando se completa la tarea de construcción relacionada.
- "10 + N*0.05" añade un almacenamiento intermedio para los procesos temporales. El tamaño del almacenamiento intermedio necesario puede ser distinto en su sitio.

Utilice la fórmula de la Figura 17 para estimar el número máximo de procesos que utiliza STCRSE, el ID de usuario de tareas iniciadas RSED (sin contar los procesos temporales no documentados).

$$4 + 3 * A$$

Figura 17. Número de procesos de STCRSE

Donde

- "4" equivale al número de procesos que utilizan el daemon de RSE y los espacios de direcciones autorizados APF de RSE.
- "A" representa el número de espacios de direcciones de agrupación de hebras de RSE.

Utilice la fórmula de la Figura 18 para estimar el número máximo de procesos utilizados por un cliente de Developer for System z (sin contar los procesos temporales no documentados).

$$(x + y + z) + 5*s$$

Figura 18. Número de procesos por cliente

Donde

- "x" depende de las opciones de configuración seleccionadas y se documenta para que la fórmula calcule el número máximo de procesos (Figura 16 en la página 87).
- "y" depende de las opciones de configuración seleccionadas y se documenta para que la fórmula calcule el número máximo de procesos (Figura 16 en la página 87).
- "z" es 0 de forma predeterminada, pero puede aumentar dependiendo de las acciones de usuario, ya que se documenta para que la fórmula calcule el número máximo de procesos (Figura 16 en la página 87).
- "s" es 1 cuando se utiliza SCLM Developer Toolkit; de lo contrario, el valor es 0.

Las definiciones de la Tabla 28 pueden limitar el número real de procesos.

Tabla 28. Límites de procesos

Ubicación	Límite	Recursos afectados
BPXPRMxx	MAXPROCSYS	Limita el número total de procesos
BPXPRMxx	MAXPROCUSER	Limita el número de procesos por UID de z/OS UNIX
Segmento OMVS	PROCUSERMAX	Limita el número de procesos para un ID de usuario

Nota:

- El daemon RSE y las agrupaciones de hebras RSE utilizan el mismo ID de usuario. Dado que el daemon RSE inicia una agrupación de hebras nueva cada vez que es necesaria, el número de procesos para este ID de usuario puede aumentar. Así, debe establecerse MAXPROCUSER para acomodar este aumento, que se puede formular como "3 + 2*A".
- El límite MAXPROCUSER es por ID de usuario de z/OS UNIX exclusivo (UID). Multiplique el recuento estimado de procesos por usuario por el número de clientes activos simultáneamente, en caso de que los usuarios compartan el mismo UID.
- El límite PROCUSERMAX es exclusivo del ID de usuario y está definido en su software de seguridad, en el segmento OMVS del ID de usuario.

Recuento de hebras

La Tabla 29 en la página 90 enumera el número de hebras utilizado por las funciones de Developer for System z seleccionadas. "u" de la columna "Hebras" indica que la cantidad debe multiplicarse por el número de usuarios activos simultáneamente que están utilizando la función. El recuento de hebras se enumera por proceso, puesto que los límites están establecidos a este nivel.

- RSEDx: Estas hebras se crean en la agrupación de hebras RSE, que pueden compartir varios clientes. Todas las hebras que finalizan en la misma agrupación de hebras pueden añadirse conjuntamente para obtener el recuento total.
- Activas: Estas hebras son parte del proceso que realiza en efecto la función solicitada. Cada proceso es una unidad autónoma, de manera que no es necesario sumar los recuentos de hebras, aunque estén asignados a un mismo ID de usuario; a menos que se especifique lo contrario.
- Programa de arranque: Los procesos del programa de arranque son necesarios para iniciar el proceso real. Cada uno tiene 1 hebra, y puede haber varios programas de arranque consecutivos. No es necesario sumar los recuentos de hebras.

Tabla 29. Recuento de hebras

Hebras			ID de usuario	Descripción
RSEDx	Activa	Prog. arranque		
-	(f) 3 + 1u	-	STCJMON	Supervisor de trabajos JES
-	4	-	STCDBM	Gestor de depuración
-	14	2	STCRSE	Daemon RSE
-	1	-	STCRSE	Autorización APF daemon RSE
(a,g) 12 + 8u	-	(a) 1	STCRSE	Agrupación de hebras RSE con extractores de una sola hebra
(a,g) 12 + 19u	-	(a) 1	STCRSE	Agrupación de hebras RSE, con extractores de varias hebras
-	(a) 1	-	STCRSE	Autorización APF de agrupación de hebras RSE
-	(b) 4u	(b) 1u	<IDusuario>	Pasarela de cliente ISPF (Servicio de mandatos TSO y SCLMDT)
-	2u	-	<IDusuario>	Servicio de mandatos TSO (APPC)
1u	2u	-	STCRSE e <IDusuario>	CARMA (por lotes)
1u	2u	-	STCRSE e <IDusuario>	CARMA (crastart)
2u	(1+1+1+1)u	1u	STCRSE e <IDusuario>	CARMA (crastart con rastreo) (h)
5u	4u	3u	STCRSE e <IDusuario>	CARMA (ispf, en desuso)
-	(c) 1u	2u	<IDusuario>	Construcción de z/OS UNIX (mandatos de shell)

Tabla 29. Recuento de hebras (continuación)

Hebras			ID de usuario	Descripción
6u	1u	-	STCRSE e <IDusuario>	Shell de z/OS UNIX
(d) 1	-	-	STCRSE	Descargar
(e) 1	-	-	STCRSE	Buscar
-	(5)	-	<IDusuario>	SCLM Developer Toolkit
1u	-	-	STCRSE	Temporizador para tiempo de espera desocupado

Nota:

- (a) Hay, como mínimo, un espacio de direcciones de agrupaciones de hebras RSE activo. Consulte “Recuento de espacios de direcciones” en la página 83 para determinar el número real de espacios de direcciones de agrupaciones de hebras RSE.
- (b) En los casos normales, y cuando se utilizan las opciones de configuración predeterminadas, hay 1 Pasarela de cliente ISPF activa por usuario. El número real puede variar, tal como se describe en “Recuento de espacios de direcciones” en la página 83.
- SCLMDT requiere un espacio de direcciones de Pasarela de cliente ISPF. SCLMDT comparte el espacio de direcciones con el servicio de mandatos TSO.
- Dependiendo de la acción seleccionada, SCLMDT puede utilizar varios procesos de una única hebra que finalizan cuando se completan las tareas. La Tabla 29 en la página 90 enumera el número máximo de hebras SCLMDT simultáneos.
- La mayor parte de acciones relacionadas con conjuntos de datos de MVS utilizan el servicio de mandatos TSO, que puede estar activo en la Pasarela de cliente ISPF o en una transacción APPC, respectivamente.
- (c) Una construcción de z/OS UNIX invoca los distintos programas de utilidad de construcción, que pueden ser de varias hebras. La Tabla 29 en la página 90 enumera el número mínimo de hebras de construcción de z/OS UNIX simultáneas.
- (d) Cada descarga de datos de host utilizará una hebra diferente. Esta hebra acabará cuando los datos se hayan transferido al cliente.
- (e) Cada búsqueda remota utilizará una hebra diferente. Esta hebra acabará cuando los resultado se hayan transferido al cliente.
- Todas las hebras enumeradas permanecen activas hasta que el proceso relacionado finaliza, a menos que se indique lo contrario.
- El recuento de hebras normal para el código con autorización APF de RSE es 1. Sin embargo, durante el inicio temporalmente hay 13 o más hebras simultáneamente activas.
- (f) Un solo usuario puede tener varias hebras activas en el Supervisor de trabajos JES para permitir el proceso simultáneo de varias solicitudes.
- (g) Los extractores específicos de usuario se pueden iniciar de dos maneras; todos los extractores para un solo usuario pueden compartir una única hebra (modalidad de una sola hebra) o cada extractor utiliza una hebra dedicada (modalidad de varias hebras). La agrupación de todos los extractores para un usuario en una sola hebra reduce el uso de hebras en la agrupación de hebras,

pero puede provocar retrasos en el proceso del mandato cuando un usuario realiza varias tareas. El método de inicio está controlado por la directiva `DSTORE_USE_THREADED_MINERS` en `rsed.envvars`. El ejemplo `rsed.envvars` utiliza la modalidad de varias hebras.

- (h) El rastreo de inicio de `CRASTART` de `CARMA` está controlador por el nivel de depuración activa de RSE para `rsecomm.log`.

Utilice la fórmula de la Figura 19 para calcular el número máximo de hebras que utiliza la agrupación de hebras RSE en una configuración de extractores de una sola hebra. Utilice la fórmula de la Figura 20 para calcular el número máximo de que utiliza la agrupación de hebras RSE en una configuración de extractores de varias hebras. Utilice la fórmula de la Figura 21 para estimar el número máximo de hebras utilizado por el Supervisor de trabajos JES. Utilice la fórmula de la Figura 22 para calcular el número máximo de hebras que utiliza el gestor de depuración.

$$12 + N * (8 + x + y + z) + (20 + N * 0.1)$$

Figura 19. Número máximo de hebras de agrupación de hebras RSE (extractores de una sola hebra)

$$12 + N * (19 + x + y + z) + (20 + N * 0.1)$$

Figura 20. Número máximo de hebras de agrupación de hebras RSE (extractores de varias hebras)

$$3 + N + (20 + N * 0.1)$$

Figura 21. Número máximo de hebras del Supervisor de trabajos JES

$$4$$

Figura 22. Número máximo de hebras del gestor de depuración

Donde

- "N" representa el número máximo de usuarios simultáneos en esta agrupación de hebras o Supervisor de trabajos JES. El objetivo de los valores predeterminados es de 30 usuarios por agrupación de hebras.
- "x" es uno de los siguientes valores, dependiendo de las opciones de configuración seleccionadas.

X	SCLMDT	TSO a través de la Pasarela de cliente	TSO a través de APPC	Tiempo de espera
0	No	No	Sí	No
0	No	Sí	No	No
0	Sí	Sí	No	No
1	No	No	Sí	Sí
1	No	Sí	No	Sí

X	SCLMDT	TSO a través de la Pasarela de cliente	TSO a través de APPC	Tiempo de espera
1	Sí	Sí	No	Sí

- “y” es uno de los siguientes valores, dependiendo de las opciones de configuración seleccionadas.

Y	
0	No CARMA
1	CARMA (por lotes)
1	CARMA (crastart)
2	CARMA (crastart con rastreo)
5	CARMA (ispf, en desuso)

- “z” es 0 de forma predeterminada, pero puede aumentar dependiendo de las acciones de usuario:
 - Añada 6 cuando se abra un shell de z/OS UNIX. Estas hebras permanecen activas hasta que el usuario finaliza la sesión.
- “20 + N*0.1” añade un almacenamiento intermedio para las hebras temporales. El tamaño del almacenamiento intermedio necesario puede ser distinto en su sitio. Varias descargas y búsquedas simultáneas son dos ejemplos que podrían necesitar que aumente el tamaño del almacenamiento intermedio.

Las definiciones de la Tabla 30 pueden limitar el número real de hebras en un proceso, que es de suma importancia para las agrupaciones de hebras RSE.

Tabla 30. Límites de hebras

Ubicación	Límite	Recursos afectados
Segmento OMVS	THREADSMAX	Limita el número de hebras para un ID de usuario
BPXPRMxx	MAXTHREADS	Limita el número de hebras en un proceso
BPXPRMxx	MAXTHREADTASKS	Limita el número de tareas de MVS en un proceso.
BPXPRMxx	MAXASSIZE	Limita el tamaño de espacio de direcciones y, con ello, el almacenamiento disponible para los bloques de control relacionados con las hebras.
rsed.envvars	Xmx	Establece el tamaño del almacenamiento dinámico Java máximo. Este almacenamiento está reservado, por lo que no está disponible para los bloques de control relacionados con las hebras.
rsed.envvars	maximum.clients	Limita el número de clientes (y sus hebras) de una agrupación de hebras RSE.
rsed.envvars	maximum.threads	Limita el número de hebras de cliente en una Agrupación de hebras RSE.
FEJCNFG	MAX_THREADS	Limita el número de hebras en el Supervisor de trabajos JES.

Nota:

- El límite THREADSMAX es exclusivo de cada ID de usuario y está definido en su software de seguridad, en el segmento OMVS del ID de usuario.

- El valor para `maximum.threads` en `rsed.envvars` debe ser inferior al valor de `MAXTHREADS` y `MAXTHREADTASKS` en `BPXPRMxx`, y `THREADSMAX` en el segmento OMVS del ID de usuario de tarea iniciada RSED.
- El mandato de operador **DISPLAY PROCESS,CPU**, que muestra las hebras activas en una agrupación de hebras, está limitado para que sólo muestre las primeras 4000 hebras.

Uso temporal de recursos

El uso de recursos que se documenta en las secciones anteriores es permanente durante el lapso de vida de Developer for System z o semipermanente para determinadas tareas específicas del usuario.

Sin embargo, Developer for System z utilizará temporalmente recursos adicionales para tareas de mantenimiento y para cumplir las siguientes solicitudes:

- El proceso de un evento de archivo de auditoría (directiva `audit.action` en `rsed.envvars`) utiliza una hebra adicional, un proceso adicional y posiblemente (si se establece `audit.action.id`) un espacio de direcciones adicional.
- El proceso de un evento de inicio de sesión (directiva `logon.action` en `rsed.envvars`) utiliza una hebra adicional, un proceso adicional y posiblemente (si se establece `logon.action.id`) un espacio de direcciones adicional.
- El mandato de operador IVP PASSTICKET utilizará dos hebras adicionales.
- El mandato de operador IVP DAEMON utilizará una hebra adicional, un proceso adicional y un espacio de direcciones adicional.
- El mandato de operador IVP ISPF utilizará una hebra adicional, un proceso adicional y un espacio de direcciones adicional, además de los recursos utilizados por la Pasarela de cliente ISPF.

Recuento de hebras

La Tabla 29 en la página 90 enumera el número de hebras utilizado por las funciones de Developer for System z seleccionadas. "u" de la columna "Hebras" indica que la cantidad debe multiplicarse por el número de usuarios activos simultáneamente que están utilizando la función. El recuento de hebras se enumera por proceso, puesto que los límites están establecidos a este nivel.

- RSEDx: Estas hebras se crean en la agrupación de hebras RSE, que pueden compartir varios clientes. Todas las hebras que finalizan en la misma agrupación de hebras pueden añadirse conjuntamente para obtener el recuento total.
- Activas: Estas hebras son parte del proceso que realiza en efecto la función solicitada. Cada proceso es una unidad autónoma, de manera que no es necesario sumar los recuentos de hebras, aunque estén asignados a un mismo ID de usuario; a menos que se especifique lo contrario.
- Programa de arranque: Los procesos del programa de arranque son necesarios para iniciar el proceso real. Cada uno tiene 1 hebra, y puede haber varios programas de arranque consecutivos. No es necesario sumar los recuentos de hebras.

Tabla 31. Recuento de hebras

Hebras			ID de usuario	Descripción
RSEDx	Activa	Prog. arranque		
-	(f) 3 + 1u	-	STCJMON	Supervisor de trabajos JES

Tabla 31. Recuento de hebras (continuación)

Hebras			ID de usuario	Descripción
-	4	-	STCDBM	Gestor de depuración
-	14	2	STCRSE	Daemon RSE
-	1	-	STCRSE	Autorización APF daemon RSE
(a,g) 12 + 8u	-	(a) 1	STCRSE	Agrupación de hebras RSE con extractores de una sola hebra
(a,g) 12 + 19u	-	(a) 1	STCRSE	Agrupación de hebras RSE, con extractores de varias hebras
-	(a) 1	-	STCRSE	Autorización APF de agrupación de hebras RSE
-	(b) 4u	(b) 1u	<IDusuario>	Pasarela de cliente ISPF (Servicio de mandatos TSO y SCLMDT)
-	2u	-	<IDusuario>	Servicio de mandatos TSO (APPC)
1u	2u	-	STCRSE e <IDusuario>	CARMA (por lotes)
1u	2u	-	STCRSE e <IDusuario>	CARMA (crastart)
2u	(1+1+1+1)u	1u	STCRSE e <IDusuario>	CARMA (crastart con rastreo) (h)
5u	4u	3u	STCRSE e <IDusuario>	CARMA (ispf, en desuso)
-	(c) 1u	2u	<IDusuario>	Construcción de z/OS UNIX (mandatos de shell)
6u	1u	-	STCRSE e <IDusuario>	Shell de z/OS UNIX
(d) 1	-	-	STCRSE	Descargar
(e) 1	-	-	STCRSE	Buscar
-	(5)	-	<IDusuario>	SCLM Developer Toolkit
1u	-	-	STCRSE	Temporizador para tiempo de espera desocupado

Nota:

- (a) Hay, como mínimo, un espacio de direcciones de agrupaciones de hebras RSE activo. Consulte “Recuento de espacios de direcciones” en la página 83 para determinar el número real de espacios de direcciones de agrupaciones de hebras RSE.
- (b) En los casos normales, y cuando se utilizan las opciones de configuración predeterminadas, hay 1 Pasarela de cliente ISPF activa por usuario. El número real puede variar, tal como se describe en “Recuento de espacios de direcciones” en la página 83.
- SCLMDT requiere un espacio de direcciones de Pasarela de cliente ISPF. SCLMDT comparte el espacio de direcciones con el servicio de mandatos TSO.
- Dependiendo de la acción seleccionada, SCLMDT puede utilizar varios procesos de una única hebra que finalizan cuando se completan las tareas. La Tabla 29 en la página 90 enumera el número máximo de hebras SCLMDT simultáneos.
- La mayor parte de acciones relacionadas con conjuntos de datos de MVS utilizan el servicio de mandatos TSO, que puede estar activo en la Pasarela de cliente ISPF o en una transacción APPC, respectivamente.
- (c) Una construcción de z/OS UNIX invoca los distintos programas de utilidad de construcción, que pueden ser de varias hebras. La Tabla 29 en la página 90 enumera el número mínimo de hebras de construcción de z/OS UNIX simultáneas.
- (d) Cada descarga de datos de host utilizará una hebra diferente. Esta hebra acabará cuando los datos se hayan transferido al cliente.
- (e) Cada búsqueda remota utilizará una hebra diferente. Esta hebra acabará cuando los resultado se hayan transferido al cliente.
- Todas las hebras enumeradas permanecen activas hasta que el proceso relacionado finaliza, a menos que se indique lo contrario.
- El recuento de hebras normal para el código con autorización APF de RSE es 1. Sin embargo, durante el inicio temporalmente hay 13 o más hebras simultáneamente activas.
- (f) Un solo usuario puede tener varias hebras activas en el Supervisor de trabajos JES para permitir el proceso simultáneo de varias solicitudes.
- (g) Los extractores específicos de usuario se pueden iniciar de dos maneras; todos los extractores para un solo usuario pueden compartir una única hebra (modalidad de una sola hebra) o cada extractor utiliza una hebra dedicada (modalidad de varias hebras). La agrupación de todos los extractores para un usuario en una sola hebra reduce el uso de hebras en la agrupación de hebras, pero puede provocar retrasos en el proceso del mandato cuando un usuario realiza varias tareas. El método de inicio está controlado por la directiva `DSTORE_USE_THREADED_MINERS` en `rsed.envvars`. El ejemplo `rsed.envvars` utiliza la modalidad de varias hebras.
- (h) El rastreo de inicio de CRASTART de CARMA está controlador por el nivel de depuración activa de RSE para `rsecomm.log`.

Utilice la fórmula de la Figura 19 en la página 92 para calcular el número máximo de hebras que utiliza la agrupación de hebras RSE en una configuración de extractores de una sola hebra. Utilice la fórmula de la Figura 20 en la página 92 para calcular el número máximo de que utiliza la agrupación de hebras RSE en una configuración de extractores de varias hebras. Utilice la fórmula de la Figura 21 en la página 92 para estimar el número máximo de hebras utilizado por el Supervisor de trabajos JES. Utilice la fórmula de la Figura 22 en la página 92 para calcular el número máximo de hebras que utiliza el gestor de depuración.

$$12 + N*(8 + x + y + z) + (20 + N*0.1)$$

Figura 23. Número máximo de hebras de agrupación de hebras RSE (extractores de una sola hebra)

$$12 + N*(19 + x + y + z) + (20 + N*0.1)$$

Figura 24. Número máximo de hebras de agrupación de hebras RSE (extractores de varias hebras)

$$3 + N + (20 + N*0.1)$$

Figura 25. Número máximo de hebras del Supervisor de trabajos JES

4

Figura 26. Número máximo de hebras del gestor de depuración

Donde

- "N" representa el número máximo de usuarios simultáneos en esta agrupación de hebras o Supervisor de trabajos JES. El objetivo de los valores predeterminados es de 30 usuarios por agrupación de hebras.
- "x" es uno de los siguientes valores, dependiendo de las opciones de configuración seleccionadas.

X	SCLMDT	TSO a través de la Pasarela de cliente	TSO a través de APPC	Tiempo de espera
0	No	No	Sí	No
0	No	Sí	No	No
0	Sí	Sí	No	No
1	No	No	Sí	Sí
1	No	Sí	No	Sí
1	Sí	Sí	No	Sí

- "y" es uno de los siguientes valores, dependiendo de las opciones de configuración seleccionadas.

Y	
0	No CARMA
1	CARMA (por lotes)
1	CARMA (crastart)
2	CARMA (crastart con rastreo)
5	CARMA (ispf, en desuso)

- "z" es 0 de forma predeterminada, pero puede aumentar dependiendo de las acciones de usuario:

- Añada 6 cuando se abra un shell de z/OS UNIX. Estas hebras permanecen activas hasta que el usuario finaliza la sesión.
- "20 + N*0.1" añade un almacenamiento intermedio para las hebras temporales. El tamaño del almacenamiento intermedio necesario puede ser distinto en su sitio. Varias descargas y búsquedas simultáneas son dos ejemplos que podrían necesitar que aumente el tamaño del almacenamiento intermedio.

Las definiciones de la Tabla 30 en la página 93 pueden limitar el número real de hebras en un proceso, que es de suma importancia para las agrupaciones de hebras RSE.

Tabla 32. Límites de hebras

Ubicación	Límite	Recursos afectados
Segmento OMVS	THREADSMAX	Limita el número de hebras para un ID de usuario
BPXPRMxx	MAXTHREADS	Limita el número de hebras en un proceso
BPXPRMxx	MAXTHREADTASKS	Limita el número de tareas de MVS en un proceso.
BPXPRMxx	MAXASSIZE	Limita el tamaño de espacio de direcciones y, con ello, el almacenamiento disponible para los bloques de control relacionados con las hebras.
rsed.envvars	Xmx	Establece el tamaño del almacenamiento dinámico Java máximo. Este almacenamiento está reservado, por lo que no está disponible para los bloques de control relacionados con las hebras.
rsed.envvars	maximum.clients	Limita el número de clientes (y sus hebras) de una agrupación de hebras RSE.
rsed.envvars	maximum.threads	Limita el número de hebras de cliente en una Agrupación de hebras RSE.
FEJCNFG	MAX_THREADS	Limita el número de hebras en el Supervisor de trabajos JES.

Nota:

- El límite THREADSMAX es exclusivo de cada ID de usuario y está definido en su software de seguridad, en el segmento OMVS del ID de usuario.
- El valor para maximum.threads en rsed.envvars debe ser inferior al valor de MAXTHREADS y MAXTHREADTASKS en BPXPRMxx, y THREADSMAX en el segmento OMVS del ID de usuario de tarea iniciada RSED.
- El mandato de operador **DISPLAY PROCESS,CPU**, que muestra las hebras activas en una agrupación de hebras, está limitado para que sólo muestre las primeras 4000 hebras.

Uso de almacenamiento

RSE es una aplicación Java, que implica que la planificación de uso de almacenamiento (memoria) para Developer for System z debe tener en cuenta dos límites de asignación de almacenamiento, el almacenamiento dinámico Java y el tamaño de Espacio de direcciones.

Límite de tamaño de almacenamiento dinámico Java

Java ofrece varios servicios para facilitar los esfuerzos de codificación para las aplicaciones Java. Uno de estos servicios es la gestión de almacenamiento.

La gestión de almacenamiento de Java asigna bloques grandes de almacenamiento, y los utiliza para las solicitudes de almacenamiento de la aplicación. Este almacenamiento gestionado por Java se denomina almacenamiento dinámico Java. La recogida de basura periódica (desfragmentación) reclama el espacio no utilizado del almacenamiento dinámico y reduce su tamaño. Tenga en cuenta que para guardar ciclos de CPU, la recogida de basura tiende a esperar hasta que el almacenamiento ocupado sea realmente necesario, dejando así almacenamiento que ya no se utiliza asignado (y convirtiéndose en paginado) durante más tiempo que el absolutamente necesario.

El tamaño máximo de almacenamiento dinámico Java se define en `rsed.envvars` con la directiva `Xmx`. Si no se especifica esta directiva, Java utiliza un tamaño predeterminado de 512 MB. Debe especificar un valor de 256 MB o superior. Cuando se ejecuta en modalidad de 64 bits, Java intentará asignar el almacenamiento dinámico por encima de la barra de 2 GB, liberando el espacio por debajo de la barra.

Cada agrupación de hebras RSE (que proporciona servicio a las acciones de cliente) es una aplicación Java individual, por lo que tiene un almacenamiento dinámico Java personal. Tenga en cuenta que todas las agrupaciones de hebras utilizan el mismo archivo de configuración `rsed.envvars`, por lo que tienen el mismo límite de tamaño de almacenamiento dinámico Java.

El uso de la agrupación de hebras del almacenamiento intermedio Java depende sobre todo de las acciones realizadas por los clientes conectados. Es necesario supervisar regularmente el uso del almacenamiento dinámico para establecer el límite de tamaño de almacenamiento dinámico óptimo. Utilice el mandato de operador **modify display process** para supervisar el uso del almacenamiento dinámico Java por parte de las agrupaciones de hebras RSE.

Límite de tamaño del espacio de direcciones

Todas las aplicaciones z/OS, incluidas las aplicaciones Java, están activas dentro de un espacio de direcciones, por lo que están limitadas por las limitaciones de tamaño del espacio de direcciones.

El tamaño de espacio de direcciones deseado se especifica durante el inicio, por ejemplo con el parámetro `REGION` en `JCL`. Sin embargo, los valores del sistema pueden limitar el tamaño de espacio de direcciones real. Consulte “Tamaño del espacio de direcciones” en la página 198 para obtener más información sobre estos límites.

- `MAXASSIZE` en `SYS1.PARMLIB(BPXPRMxx)`
- `ASSIZEMAX` en el segmento `OMVS` del ID de usuario asignado a la tarea iniciada
- salidas del sistema `IEFUSI` y `IEALIMIT`
- `MEMLIMIT` en `SYS1.PARMLIB(SMFPRMxx)` para la modalidad de direccionamiento de 64 bits

Las agrupaciones de hebras RSE heredan los límites de tamaño del espacio de direcciones del daemon RSE. El tamaño del espacio de direcciones debe ser suficiente para albergar el almacenamiento dinámico Java, el propio Java, las áreas de almacenamiento comunes y todos los bloques de control que el sistema crea para soportar la actividad de las agrupaciones de hebras, por ejemplo, un TCB (bloque de control de tareas) por hebra. Tenga en cuenta que parte del uso de este almacenamiento está por debajo de la línea de 16 MB. Cuando se ejecuta en

modalidad de 64 bits, Java intentará asignar el almacenamiento dinámico por encima de la barra de 2 GB, liberando espacio por debajo de la barra.

Debe supervisar el tamaño del espacio de direcciones real antes de cambiar ningún valor que tenga una influencia sobre el mismo, como cambiar el tamaño del almacenamiento dinámico Java o la cantidad de usuarios soportados por una única agrupación de hebras. Utilice el software de supervisión del sistema habitual para rastrear el uso del almacenamiento real por Developer for system z. Si no dispone de una herramienta de supervisión para ello, se puede reunir la información básica con herramientas como la vista SDSF DA o TASID (una herramienta de información del sistema tal cual disponible en el sitio Web de ISPF "Soporte y descargas").

Directrices de estimación de tamaño

Tal como se ha indicado anteriormente, el uso real que Developer for system z hace del almacenamiento está muy condicionado por la actividad del usuario. Algunas acciones utilizan una cantidad fija de almacenamiento (por ejemplo, el inicio de sesión), mientras que otras son variables (por ejemplo, enumerar conjuntos de datos con un calificador de alto nivel especificado).

- Utilice un espacio de direcciones de 2 GB para que RSE tenga espacio para el almacenamiento dinámico Java y todos los bloques de control del sistema.
- Cuando se ejecuta en modalidad de 64 bits, asegúrese de que el almacenamiento por encima de la barra de 2 GB está realmente disponible para RSE.
- Consulte "Tamaño del espacio de direcciones" en la página 198 para aprender más sobre dónde se pueden establecer los límites del tamaño de espacio de direcciones.
- El objetivo de la configuración de `rsed.envvars` de ejemplo es de 30 usuarios por agrupación de hebras.
 - `maximum.clients=30`
 - `maximum.threads=520` ($10+17*30 = 520$, de modo que 520 permite 30 clientes)
- La configuración de `rsed.envvars` de ejemplo permite al almacenamiento dinámico Java crecer hasta 512 MB. Ello permite, con 30 clientes, que cada cliente pueda utilizar una media de 17 MB ($30*17 = 510$).

Tenga en cuenta que RSE muestra el almacenamiento dinámico Java actual y el límite de tamaño del espacio de direcciones durante el inicio en el mensaje de consola FEK004I.

Utilice cualquiera de los siguientes casos de ejemplo si la supervisión muestra que el almacenamiento dinámico Java actual es insuficiente para la carga de trabajo real:

- Aumente el tamaño máximo de almacenamiento dinámico Java con la directiva `Xmx` de `rsed.envvars`. Antes de hacerlo, asegúrese de que hay espacio en el espacio de direcciones para el aumento de tamaño.
- Reduzca el número máximo de clientes por agrupación de hebras con la directiva `maximum.clients` de `rsed.envvars`. RSE seguirá soportando el mismo número de clientes, pero los clientes se distribuirán entre más agrupaciones de hebras.

Como referencia, la Tabla 33 en la página 101 muestra valores utilizados por clientes de Developer for System z reales en valores clave `rsed.envvars` que tienen impacto en el uso de almacenamiento.

Tabla 33. Valores de referencia para uso de almacenamiento

xmx (almacenamiento dinámico de Java máximo)	maximum.clients	Tipo de desarrollo primario
512M	30	PL/I
512M	10	COBOL
384M	12	COBOL
800M (64 bits)	20	No especificado

Análisis del uso de almacenamiento de ejemplo

Las pantallas de las siguientes figuras muestran algunos número del uso de almacenamiento de ejemplo para una configuración de Developer for System z predeterminada con estas modificaciones.

- single.logon se inhabilita para impedir que RSE cree como mínimo dos espacios de direcciones de agrupaciones de hebras
- El tamaño máximo de almacenamiento dinámico Java se establece en 10 MB, dado que un máximo pequeño resultará en un percentil de uso mayor y los límites de tamaño del almacenamiento dinámico se alcanzarán antes.

Tamaño máx. almacenamiento dinámico=10MB y Tamaño AS privado=1,959MB

inicio

BPXM023I (STCRSE)
ProcessId(268) Uso de memoria(7%) Clientes(0)

Jobname	Tiempo Cpu	Almacenamiento	EXCP
JMON	0.01	2740	72
RSED	4.47	32.8M	15910
RSED8	1.15	27.4M	12612

inicio de sesión 1

BPXM023I (STCRSE)
ProcessId(268) Uso de memoria(13%) Clientes(1)

Jobname	Tiempo Cpu	Almacenamiento	EXCP
JMON	0.01	2864	81
RSED	4.55	32.8M	15980
RSED8	3.72	55.9M	24128

inicio de sesión 2

BPXM023I (STCRSE)
ProcessId(268) Uso de memoria(23%) Clientes(2)

Jobname	Tiempo Cpu	Almacenamiento	EXCP
JMON	0.02	2944	86
RSED	4.58	32.9M	16027
RSED8	4.20	57.8M	25205

inicio de sesión 3

BPXM023I (STCRSE)
ProcessId(268) Uso de memoria(37%) Clientes(3)

Jobname	Tiempo Cpu	Almacenamiento	EXCP
JMON	0.02	3020	91
RSED	4.60	32.9M	16076
RSED8	4.51	59.6M	26327

inicio de sesión 4

BPXM023I (STCRSE)
ProcessId(268) Uso de memoria(41%) Clientes(4)

Jobname	Tiempo Cpu	Almacenamiento	EXCP
JMON	0.02	3108	96
RSED	4.61	32.9M	16125
RSED8	4.77	62.3M	27404

Figura 27. Uso de recursos con 5 inicios de sesión

inició de sesión 5

```
BPXM023I (STCRSE)
ProcessId(268      ) Uso de memoria(41%) Clientes(4)
ProcessId(33554706) Uso de memoria(13%) Clientes(1)
```

Jobname	Tiempo Cpu	Almacenamiento	EXCP
JMON	0.03	3184	101
RSED	4.64	32.9M	16229
RSED8	4.78	62.4M	27413
RSED9	4.60	56.6M	24065

Figura 28. Uso de recursos con 5 inicios de sesión (continuación)

La Figura 27 en la página 102 y la Figura 28 muestran un caso de ejemplo en el que 5 clientes inician sesión en un daemon RSE con un almacenamiento dinámico Java de 10 MB.

- Una agrupación de hebras (RSED8) está en estado latente en el inicio y utiliza alrededor de 27 MB, de los cuales 0.7 MB están en el almacenamiento intermedio Java (7% de 10 MB).
- La agrupación de hebras pasa a estar activa cuando se conecta el primer cliente, y utiliza otros 27 MB más 2 MB por cada cliente que se conecta.
- Parte de estos 2MB por conexión estará en el almacenamiento intermedio Java, tal como muestra el aumento del uso de almacenamiento dinámico.
- Sin embargo, no hay ningún patrón real sobre el uso de almacenamiento dinámico, puesto que depende de mecanismos Java que estiman el almacenamiento necesario y asignan más del necesario. La recogida de basura intermitente libera almacenamiento, haciendo que las tendencias sean todavía más difíciles de detectar.
- Mecanismos internos que limitan el número de conexiones por agrupación de hebras para asegurar que haya el tamaño de almacenamiento dinámico suficiente para el resultado de hebras activas en la quinta conexión en una agrupación de hebras nueva (RSED9). Estas redes de seguridad interna no se suelen invocar cuando se utiliza una configuración adecuada, ya que antes se alcanzarían otros límites (probablemente el `maximum.clients` de `rsed.envvars`).

Tamaño máx. almacenamiento dinámico=10MB y Tamaño AS privado=1,959MB

inicio

BPXM023I (STCRSE)
ProcessId(212) Uso de memoria(7%) Clientes(0)

Jobname	Tiempo Cpu	Almacenamiento	EXCP
JMON	0.01	2736	71
RSED	4.35	32.9M	15117
RSED8	1.43	27.4M	12609

inicio de sesión

BPXM023I (STCRSE)
ProcessId(212) Uso de memoria(13%) Clientes(1)

Jobname	Tiempo Cpu	Almacenamiento	EXCP
JMON	0.01	2864	80
RSED	4.48	33.0M	15187
RSED8	3.53	53.9M	24125

ampliar árbol de MVS grande (195 conjuntos de datos)

BPXM023I (STCRSE)
ProcessId(212) Uso de memoria(13%) Clientes(1)

Jobname	Tiempo Cpu	Almacenamiento	EXCP
JMON	0.01	2864	80
RSED	4.58	33.1M	16094
RSED8	4.28	56.1M	24740

ampliar PDS pequeño (21 miembros)

BPXM023I (STCRSE)
ProcessId(212) Uso de memoria(13%) Clientes(1)

Jobname	Tiempo Cpu	Almacenamiento	EXCP
IBMUSER2	0.22	2644	870
JMON	0.01	2864	80
RSED	4.61	33.1M	16108
RSED8	4.40	56.2M	24937

abrir miembro de tamaño medio (86 líneas)

BPXM023I (STCRSE)
ProcessId(212) Uso de memoria(13%) Clientes(1)

Jobname	Tiempo Cpu	Almacenamiento	EXCP
IBMUSER2	0.22	2644	870
JMON	0.01	2864	80
RSED	4.61	33.1M	16108
RSED8	8.12	62.7M	27044

Figura 29. Uso de recursos al editar un miembro PDS

La Figura 29 muestra un caso de ejemplo en el que 1 cliente inicia sesión en un daemon RSE con un almacenamiento dinámico Java de 10 MB y edita un miembro PDS.

- La búsqueda de catálogos que resulta en 195 nombres de conjuntos de datos ha utilizado aproximadamente 2MB de almacenamiento, todo ello por la actividad del sistema, ya que el uso de almacenamiento dinámico Java no aumenta.

- Abrir un PDS de 21 miembros apenas utiliza memoria en la agrupación de hebras, pero la pantalla muestra que se ha invocado el Servicio de mandatos TSO. Hay un espacio de direcciones nuevo activo (IBMUUSER2), que utiliza el tamaño de región asignado a este ID de usuario en TSO. Este espacio de direcciones permanece activo durante un período de tiempo especificado, de manera que el servicio de mandatos TSO lo puede volver a utilizar para futuras peticiones.
- Abrir un miembro muestra números parecidos a ampliar un calificador de alto nivel. El uso de almacenamiento dinámico Java permanece igual, pero hay un aumento del almacenamiento de 6.5 MB como consecuencia de la actividad del sistema.

Uso de espacio del sistema de archivos de z/OS UNIX

La mayoría de datos relacionados con Developer for System z que no se escriben una una sentencia DD terminan en un archivo z/OS UNIX. El programador del sistema controla qué datos se escriben y a dónde van. Sin embargo, no controla la cantidad de datos que se escriben.

Los datos pueden agruparse en estas categorías:

- Análisis de problemas (archivos de registro y archivos de vuelco del sistema). Se documentan más detalles en la Capítulo 12, “Resolución de problemas de configuración”, en la página 181
- Auditoría, tal como se documenta en “Registro de auditoría” en la página 24
- Metadatos de Envío a cliente, como se documenta en “Metadatos Envío a cliente” en la página 133.
- Datos temporales

Tal como se documenta en Capítulo 12, “Resolución de problemas de configuración”, en la página 181, Developer for System z escribe los registros del host relacionadas con RSE en los siguientes directorios de z/OS UNIX:

- /var/rdz/logs/server para registros de tareas iniciadas RSE
- /var/rdz/logs/\$LOGNAME para registros de usuario

De forma predeterminada, en los registros sólo se escriben los mensajes de error y de aviso. De manera que, si todo sale según se prevé, estos directorios deberían contener únicamente archivos vacíos o prácticamente vacíos (sin contar los registros de auditoría).

Puede habilitar los registros de mensajes informativos, preferiblemente bajo la dirección del centro de soporte de IBM, cosa que aumenta significativamente el tamaño de los archivos de registro.

```

inicio

$ ls -l /var/rdz/logs/server
total 144
-rw-rw-rw- 1 STCRSE STCGRP 33642 10 Jul 12:10 rsedaemon.log
-rw-rw-rw- 1 STCRSE STCGRP 1442 10 Jul 12:10 rseserver.log

inicio de sesión

$ ls -l /var/rdz/logs/server
total 144
-rw----- 1 STCRSE STCGRP 36655 Jul 10 12:11 rsedaemon.log
-rw----- 1 STCRSE STCGRP 1893 Jul 10 12:11 rseserver.log
$ ls -l /var/rdz/logs/IBMUSER
total 160
-rw----- 1 IBMUSER SYS1 3459 Jul 10 12:11 ffs.log
-rw----- 1 IBMUSER SYS1 0 Jul 10 12:11 ffsget.log
-rw----- 1 IBMUSER SYS1 0 Jul 10 12:11 ffsput.log
-rw----- 1 IBMUSER SYS1 303 Jul 10 12:11 ffslock.log
-rw----- 1 IBMUSER SYS1 7266 Jul 10 12:11 rsecomm.log

fin de sesión

$ ls -l /var/rdz/logs/server
total 80
-rw----- 1 STCRSE STCGRP 36655 Jul 10 12:11 rsedaemon.log
-rw----- 1 STCRSE STCGRP 2208 Jul 10 12:11 rseserver.log
$ ls -l /var/rdz/logs/IBMUSER
total 296
-rw----- 1 IBMUSER SYS1 6393 Jul 10 12:11 ffs.log
-rw----- 1 IBMUSER SYS1 0 Jul 10 12:11 ffsget.log
-rw----- 1 IBMUSER SYS1 0 Jul 10 12:11 ffsput.log
-rw----- 1 IBMUSER SYS1 609 Jul 10 12:11 ffslock.log
-rw----- 1 IBMUSER SYS1 45157 Jul 10 12:11 rsecomm.log

detener

$ ls -l /var/rdz/logs/server
total 80
-rw----- 1 STCRSE STCGRP 36655 Jul 10 12:11 rsedaemon.log
-rw----- 1 STCRSE STCGRP 2490 Jul 10 12:12 rseserver.log

```

Figura 30. uso de espacio del sistema de archivos de z/OS UNIX

La Figura 30 muestra el uso de espacio mínimo del sistema de archivos de z/OS UNIX al utilizar el nivel de depuración 2 (mensajes informativos).

- Los registros de tareas iniciadas utilizan 34 KB tras el inicio y aumentan paulatinamente a medida que los usuarios inician sesión, finalizan sesión, o bien se emiten mandatos de operador.
- Un directorio de registro de cliente utiliza 11 KB tras el inicio de sesión y aumenta bastante cuando el usuario empieza a trabajar (no se muestra en el ejemplo).
- Finalizar la sesión añade otros 40 KB a los registros de usuario, incrementando el número hasta 51 KB.

A excepción de los registros de auditoría, los archivos de registro se sobrescriben cada vez que se reinicia (para la tarea iniciada RSE) o se finaliza la sesión (para un cliente), manteniendo el tamaño total adecuado. Los registros de auditoría se eliminan después de que caduque el intervalo especificado en `audit.retention.period`. La directiva `keep.last.log` de `rsed.envvars` cambia esto ligeramente, ya que puede hacer que RSE mantenga una copia de los registros anteriores. Las copias antiguas se eliminan siempre. Si la directiva `keep.all.logs` en `rsed.envvars` está habilitada, todos los registros tienen una indicación de fecha

y hora que se añade al nombre y los archivos se eliminan después de que caduque el intervalo especificado en `log.retention.period`.

Se envía un mensaje de aviso a la consola cuando el sistema de archivos que contiene los archivos de registro se está quedando sin espacio disponible. Este mensaje de consola (FEK103E) se repite regularmente hasta que se ha resuelto el problema de falta de espacio. Cuando el sistema de archivos se queda sin espacio, RSE intentará suprimir los archivos de registro existentes para liberar espacio. Los registros de auditoría no están afectados por este proceso.

Las definiciones de la Tabla 34 controlan qué datos se graban en los directorios de registro y dónde ubican los directorios.

Tabla 34. Directivas de salidas de registro

Ubicación	Directiva	Función
resecomm.properties	debug_level	Establecer el nivel de detalle de registro predeterminado
resecomm.properties	USER	Habilite nivel_depuración 2 para usuarios especificados.
rsed.envvars	keep.all.logs	Conserva una copia de los registros previos antes del inicio/inicio de sesión.
rsed.envvars	keep.last.log	Conserva una copia de los registros previos antes del inicio/inicio de sesión.
rsed.envvars	enable.audit.log	Mantener un rastreo de auditoría de las acciones de clientes.
rsed.envvars	enable.standard.log	Escribir las secuencias stdout y stderr de la agrupación (o agrupaciones) de hebras en un archivo de registro.
rsed.envvars	DSTORE_TRACING_ON	Habilitar registro de acciones de DataStore.
rsed.envvars	DSTORE_MEMLOGGING_ON	Habilitar registro de uso de memoria de DataStore.
Mandato de operador	modify resecommlog <nivel>	Cambiar dinámicamente el nivel de detalle de registro de resecomm.log
Mandato de operador	modify rsedaemonlog <nivel>	Cambiar dinámicamente el nivel de detalle de registro de rsedaemon.log
Mandato de operador	modify rseserverlog <nivel>	Cambiar dinámicamente el nivel de detalle de registro de rseserver.log
Mandato de operador	modify rsestandardlog {on off}	Cambiar dinámicamente la actualización de std*.log
Mandato de operador	modify trace {on off} USER=userid	Habilite nivel_depuración 2 para usuarios especificados.
Mandato de operador	modify trace {on off} SERVER=pid	Habilite nivel_depuración 2 para usuarios especificados.
Mandato de operador	modify trace clear	Inhabilite la configuración de rastreo

Tabla 34. Directivas de salidas de registro (continuación)

Ubicación	Directiva	Función
Mandato de operador	modify logs	Recopilar registros de host e información de configuración
rsed.envvars	daemon.log	Vía de acceso inicial para la tarea iniciada RSE y los registros de auditoría.
rsed.envvars	user.log	Vía de acceso inicial de los registros de usuario.
rsed.envvars	CGI_ISPWORK	Vía de acceso de inicio para los registros de pasarela de cliente ISPF
rsed.envvars	TMPDIR	Directorio para registros IVP y mandato de operador modify logs
rsed.envvars	_CEE_DMPTARG	Directorio para vuelcos Java

Developer for System z junto con el software requisito, como la Pasarela de cliente ISPF, también escribe datos temporales en /tmp y /var/rdz/WORKAREA. La cantidad de datos escritos aquí como resultado de las acciones de usuario no es predecible, de manera que debe tener mucho espacio libre en los sistemas de archivos que contienen estos directorios.

Developer for System z siempre intenta limpiar estos archivos temporales, pero se puede realizar la limpieza manual en cualquier momento, tal como se describe en el apartado "(Opcional) Borrado de WORKAREA y /tmp" de la *Guía de configuración de host* (SC11-3660), se puede realizar prácticamente en cualquier momento.

Las definiciones de la Tabla 35 controlan la ubicación de los directorios de datos temporales.

Tabla 35. Directivas de salida temporales

Ubicación	Directiva	Función
rsed.envvars	CGI_ISPWORK	Vía de acceso de inicio para datos temporales.
rsed.envvars	TMPDIR	Directorio para datos temporales.

Definiciones de recursos clave

/etc/rdz/rsed.envvars

RSE, Java y z/OS UNIX utilizan las variables de entorno definidas en rsed.envvars. El archivo de ejemplo que viene con Developer for System z tiene como destino instalaciones pequeñas o de tamaño medio que no requieren los componentes opcionales de Developer for System z. En el apartado "rsed.envvars, archivo de configuración RSE" de la *Guía de configuración de host* (SC11-3660) se describe cada variable definida en el archivo de ejemplo, en el que las siguientes variables precisan de atención especial:

_RSE_JAVA_OPTS="_RSE_JAVA_OPTS -Xms128m -Xmx512m"

Establecer el tamaño inicial (Xms) y máximo (Xmx) de la memoria dinámica. Los valores predeterminados son 128M y 512M respectivamente. Cámbielo para

aplicar los valores de tamaño de almacenamiento dinámico deseados. Si esta directiva tiene caracteres de comentario, se utilizarán los valores predeterminados de Java, que son 4M y 512M respectivamente.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Dmaximum.clients=30"

Número máximo de clientes a los que proporciona servicios una agrupación de hebras. El valor predeterminado es 30. Descomente y personalice este valor para limitar el número de clientes por agrupación de hebras. Tenga en cuenta que puede que otros límites impidan que RSE llegue a este límite.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Dmaximum.threads=520"

Cantidad máxima de hebras activas en una agrupación de hebras para permitir clientes nuevos. El valor predeterminado es 520. Descomente y personalice este valor para limitar el número de clientes por agrupación de hebras según el número de hebras que se estén utilizando. Tenga en cuenta que cada conexión de cliente utiliza varias hebras (17 o más) y que otros límites pueden impedir que RSE llegue a este límite.

Nota: Este valor debe ser inferior al valor de MAXTHREADS y MAXTHREADTASKS en SYS1.PARMLIB(BPXPRMxx).

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Dminimum.threadpool.process=1"

Número mínimo de agrupaciones de hebras activas. El valor predeterminado es 1. Descomente y personalice este valor para iniciar como mínimo el número de procesos de agrupaciones de hebras indicado. Los procesos de agrupaciones de hebras se utilizan para el equilibrio de carga de las hebras del servidor RSE. Se inician más procesos nuevos cuando estos son necesarios. Iniciar procesos nuevos ayuda a evitar los retrasos de conexión pero utiliza más recursos durante momentos desocupados.

Nota: Si la directiva single.logon está activa, habrá como mínimo dos agrupaciones de hebras iniciadas, aun cuando el valor de minimum.threadpool.process sea 1. El valor predeterminado para single.logon en rsed.envvars es active.

#_RSE_JAVAOPTS="\$_RSE_JAVAOPTS -Dmaximum.threadpool.process=100"

Número máximo de agrupaciones de hebras activas. El valor predeterminado es 100. Descomente y personalice este valor para limitar el número de procesos de procesos de agrupaciones de hebras. Los procesos de agrupaciones de hebras se utilizan para el equilibrio de carga de las hebras del servidor RSE, por lo que, al limitarlos, se limitará la cantidad de conexiones de cliente activas.

SYS1.PARMLIB(BPXPRMxx)

RSE es una aplicación Java, lo que significa que está activo en el entorno z/OS UNIX. Ello hace que BPXPRMxx se convierta en un miembro parmlib crucial, ya que contiene los parámetros que controlan el entorno z/OS UNIX y los sistemas de archivos. BPXPRMxx se describe en el manual *MVS Initialization and Tuning Reference* (SA22-7592). Se conoce que las siguientes directivas afectan a Developer for System z:

MAXPROCSYS(nnnnn)

Especifica el número máximo de procesos que el sistema permite.

Rango del valor: nnnnn es un valor decimal del 5 al 32767.

Valor predeterminado: 900

MAXPROCUSER(nnnnn)

Especifica el número máximo de procesos que un solo ID de usuario de z/OS UNIX puede tener activos simultáneamente, independientemente de cómo se crearon los procesos.

Rango del valor: nnnnn es un valor decimal del 3 al 32767.

Valor predeterminado: 25

Nota:

- Todos los procesos RSE utilizan el mismo ID de usuario de z/OS UNIX (el del usuario asignado al daemon RSE), ya que todos los clientes se ejecutan como hebras dentro de los procesos RSE.
- Este valor también se puede establecer con la variable PROCUSERMAX en el segmento de perfil de seguridad OMVS del usuario asignado a la tarea iniciada RSED.

MAXTHREADS(nnnnnn)

Especifica el número máximo de hebras pthread_created, incluyendo las que están en ejecución, en cola y de las que se ha salido pero que no se han desconectado, que un único proceso puede tener activas simultáneamente. Especificar un valor de 0 impide que las aplicaciones puedan utilizar pthread_create.

Rango del valor: nnnnnn es un valor decimal del 0 al 100000.

Valor predeterminado: 200

Nota:

- Cada cliente utiliza, como mínimo, 17 hebras en el proceso de agrupación de hebras RSE, y hay varios clientes activos dentro del proceso.
- Este valor también se puede establecer con la variable THREADSMAX en el segmento de perfil de seguridad OMVS del usuario asignado a la tarea iniciada RSED. Cuando se establece, el valor THREADSMAX se utiliza tanto para MAXTHREADS como para MAXTHREADTASKS.

MAXTHREADTASKS(nnnnn)

Especifica el número máximo de tareas de MVS que un único proceso puede tener activas simultáneamente para las hebras pthread_created.

Rango del valor: nnnnn es un valor decimal del 0 al 32768.

Valor predeterminado: 1000

Nota:

- Cada hebra activa tiene una tarea de MVS (TCB, bloque de control de tareas).
- Cada tarea simultánea de MVS necesita almacenamiento adicional, y parte de este deberá estar por debajo de la línea de 16 MB.
- Cada cliente utiliza, como mínimo, 17 hebras en el proceso de agrupación de hebras RSE, y hay varios clientes activos dentro del proceso.
- Este valor también se puede establecer con la variable THREADSMAX en el segmento de perfil de seguridad OMVS del usuario asignado a la tarea iniciada RSED. Cuando se establece, el valor THREADSMAX se utiliza tanto para MAXTHREADS como para MAXTHREADTASKS.

MAXUIDS(nnnnn)

Especifica el número máximo de ID de usuario de z/OS UNIX (UID) que pueden funcionar simultáneamente.

Rango del valor: nnnnn es un valor decimal del 1 al 32767.

Valor predeterminado: 200

MAXASSIZE(nnnnn)

Especifica los valores de recurso RLIMIT_AS que se establecerán como valores iniciales para los procesos nuevos. RLIMIT_AS indica el tamaño de región del espacio de direcciones.

Rango del valor: nnnnn es un valor decimal de 10485760 (10 Megabytes) a 2147483647 (2 Gigabytes).

Valor predeterminado: 209715200 (200 Megabytes)

Nota:

- Este valor se debe establecer como 2G.
- Este valor también se puede establecer con la variable ASSIZEMAX en el segmento de perfil de seguridad OMVS del usuario asignado a la tarea iniciada RSED.

MAXFILEPROC(nnnnnn)

Especifica el número máximo de descriptores para archivos, sockets, directorios, y cualquier otro objeto del sistema de archivos que un único proceso puede tener activos o asignados simultáneamente.

Rango del valor: nnnnnn es un valor decimal del 3 al 524287.

Valor predeterminado: 64000

Nota:

- Una agrupación de hebras tiene todas las hebras de cliente en un único proceso.
- Este valor también se puede establecer con la variable FILEPROCMAX en el segmento de perfil de seguridad OMVS del usuario asignado a la tarea iniciada RSED.

MAXMAPAREA(nnnnn)

Especifica la cantidad máxima de espacio de almacenamiento de espacios de datos (en páginas) que se puede asignar para correlaciones de memoria de archivos de z/OS UNIX. El almacenamiento no se asigna hasta que la correlación de memoria no está activa.

Rango del valor: nnnnn es un valor decimal del 1 al 16777216.

Valor predeterminado: 40960

Nota: Este valor también se puede establecer con la variable MMAPAREAMAX en el segmento de perfil de seguridad OMVS del usuario asignado a la tarea iniciada RSED.

Utilice el mandato de operador **SETOMVS** o **SET OMVS** para aumentar o disminuir dinámicamente (hasta la próxima IPL) el valor de cualquiera de las variables BPXPRMxx anteriores. Para realizar un cambio permanente, edite el miembro BPXPRMxx que se utilizará para las IPL. Consulte la publicación *MVS System Commands* (SA22-7627) para obtener más información sobre estos mandatos de operador.

Las definiciones siguientes son subparámetros de la sentencia NETWORK.

MAXSOCKETS (nnnnnnnn)

Especifica el número máximo de sockets soportados por este sistema de archivos para esta familia de direcciones. Este es un parámetro opcional.

Rango del valor: nnnnnnnn es un valor decimal del 0 al 16777215.

Valor predeterminado: 100

INADDRANYCOUNT (nnnn)

Especifica el número de puertos que el sistema reserva para utilizar con el puerto PORT 0, enlaces INADDR_ANY, empezando por el número de puerto especificado en el parámetro INADDRANYPORT. Este valor solo se necesita para CINET (varias pilas TCP/IP).

Rango del valor: nnnn es un valor decimal del 1 al 4000.

Valor predeterminado: si no se especifica ni INADDRANYPORT

ni INADDRANYCOUNT,

el valor predeterminado para

INADDRANYCOUNT es 1000.

De lo contrario, no se reservará ningún puerto (0).

definiciones de varios recursos

Tarjeta EXEC del servidor JCL

Se recomienda añadir las siguientes definiciones a la tarjeta EXEC del JCL de los servidores de Developer for System z.

REGION=0M

Se recomienda REGION=0M para las tareas iniciadas del daemon RSE y el Supervisor de trabajos de JES, RSED y JMON respectivamente. Con ello, el tamaño del espacio de direcciones está únicamente limitado por el almacenamiento privado disponible, o por las salidas del sistema IEFUSI o IEALIMIT. Tenga en cuenta que IBM recomienda encarecidamente no utilizar estas salidas para los espacios de direcciones de z/OS UNIX, como el daemon RSE.

TIME=NOLIMIT

Se recomienda utilizar TIME=NOLIMIT para todos los servidores Developer for System z. Ello es debido a que el tiempo de la CPU de todos los clientes de Developer for System z se acumula en los espacios de direcciones del servidor.

FEK.#CUST.PARMLIB(FEJJCNFG)

El Supervisor de trabajos JES utiliza las variables de entorno definidas en FEJJCNFG. El archivo de ejemplo que viene con Developer for System z está destinado a instalaciones pequeñas-medias. En el apartado "FEJJCNFG, archivo de configuración del supervisor de trabajos JES" de la *Guía de configuración de host* (SC11-3660) se describe cada variable definida en el archivo de ejemplo, en el que las siguientes variables precisan de atención especial:

MAX_THREADS

Número máximo de usuarios que pueden utilizar un supervisor de trabajos JES en un momento dado. El valor predeterminado es 200. El valor máximo es 2147483647. Si aumenta este número, es posible que también deba aumentar el tamaño del espacio de direcciones del supervisor de trabajos JES.

SYS1.PARMLIB(IEASYSxx)

IEASYSxx contiene parámetros del sistema y se describe en el manual *MVS Initialization and Tuning Reference* (SA22-7592). Se conoce que las siguientes directivas afectan a Developer for System z:

MAXUSER=nnnnn

Este parámetro especifica un valor que, en la mayoría de los casos, el sistema utiliza para limitar los trabajos y tareas iniciadas que se puede ejecutar simultáneamente durante una IPL determinada.

Rango del valor: nnnnn es un valor decimal del 0 al 32767. Observe que la suma de los valores especificados para los parámetros de sistema MAXUSER, RSVSTRT, y RSVNONR no puede superar 32767.

Valor predeterminado: 255

SYS1.PARMLIB(IVTPRMxx)

IVTPRMxx establece los parámetros para el Communication Storage Manager (CSM) y se describe en el manual *MVS Initialization and Tuning Reference* (SA22-7592). Se conoce que las siguientes directivas afectan a Developer for System z:

FIXED MAX(maxfix)

Define la cantidad máxima de almacenamiento dedicado a almacenamientos intermedios de CSM fijos.

Rango del valor: maxfix es un valor de 1024K a 2048M.

Valor predeterminado: 100M

ECSA MAX(maxecsa)

Define la cantidad máxima de almacenamiento dedicado a almacenamientos intermedios de CSM ECSA.

Rango del valor: maxecsa es un valor de 1024K a 2048M.

Valor predeterminado: 100M

SYS1.PARMLIB(ASCHPMxx)

El miembro parmlib de ASCHPMxx contiene información de planificación para el planificador de transacciones ASCH y se describe en el manual *MVS Initialization and Tuning Reference* (SA22-7592). Se conoce que las siguientes directivas afectan a Developer for System z:

MAX(nnnnn)

Un parámetro opcional de la definición CLASSADD que especifica el número máximo de iniciadores de transacciones APPC permitidos para una clase concreta de iniciadores de transacciones. Una vez se alcanza este límite, no se crean espacios de direcciones nuevos y las peticiones entrantes se dejan en cola hasta que los espacios de direcciones existentes del indicador queden disponibles. El valor no debe superar el número máximo de espacios de direcciones permitidos por la instalación, y debe tener también en cuenta los productos del sistema que también necesitarán espacios de direcciones.

Rango del valor: nnnnn es un valor decimal del 1 al 64000.

Valor predeterminado: 1

Nota: Si utiliza la APPC para iniciar el Servicio de mandatos TSO, la clase de transacción utilizada deberá tener suficientes iniciadores de transacción para permitir un iniciador para cada usuario simultáneo de Developer for System z.

Supervisión

Dado que las cargas de trabajo pueden cambiar la necesidad de los recursos del sistema, es necesario supervisar el sistema regularmente para medir el uso de recursos, de manera que se pueda ajustar Rational Developer for System z y las configuraciones del sistema en respuesta a los requisitos de los usuarios. Los siguientes mandatos se pueden utilizar como ayuda en este proceso de supervisión.

Supervisión de RSE

Las agrupaciones de hebras RSE son el punto focal para la actividad de usuarios en Developer for System z y, por ello, requieren supervisión para que su uso sea el óptimo. Se puede consultar el daemon RSE para obtener información que no se puede reunir con las herramientas de supervisión de sistemas normales.

- Utilice las herramientas de supervisión del sistema normales, como RMF, para reunir datos específicos del espacio de direcciones, por ejemplo, el almacenamiento real utilizado y el tiempo de la CPU. Si no dispone de una herramienta de supervisión para ello, se puede reunir la información básica con herramientas como la vista SDSF DA o TASID (una herramienta de información del sistema tal cual disponible en el sitio Web de ISPF “Soporte y descargas”).
- Durante el inicio, el daemon RSE informa del tamaño del espacio de direcciones disponible y del tamaño del almacenamiento dinámico Java con el mensaje de consola FEK004I.

```
FEK004I Daemon Rse: Tamaño máx. almacenamiento dinámico=65MB y
          Tamaño AS privado=1,959MB
```

- El mandato de operador **MODIFY RSED,APPL=DISPLAY PROCESS** muestra los procesos de agrupaciones de hebras RSE. El campo “Uso de memoria” muestra qué cantidad del almacenamiento dinámico Java definido se está utilizando realmente. Consulte el apartado “Mandatos de operador” de la publicación *Guía de configuración de host* (SC11-3660) para obtener más información sobre este mandato.

```
f rsed,appl=d p
BPXM023I (STCRSE)
ProcessId(16777456) Uso de memoria(33%) Clientes(4) Orden(1)
```

Cuando se utiliza la opción **DETAIL** del mandato de modificación **DISPLAY PROCESS**, se proporciona más información:

```
f rsed,appl=d p,detail
BPXM023I (STCRSE)
ProcessId(33555087) ASId(002E) JobName(RSED8) Order(1)
PROCESS LIMITS:    CURRENT  HIGHWATER  LIMIT
JAVA HEAP USAGE(%) 10       56         100
CLIENTS             0       25         30
MAXFILEPROC         83      103        64000
MAXPROCUSER         97      99         200
MAXTHREADS           9       14        1500
MAXTHREADTASKS       9       14        1500
```

La opción de CPU del mandato de modificación de **DISPLAY PROCESS** mostrará el uso de CPU acumulado (en milisegundos) de cada hebra de una agrupación de hebras:

```
f rsed,appl=d p,cpu
BPXM023I (STCRSE)
ProcessId(33555087) ASId(002E) JobName(RSED8) Order(1)
```

USERID	THREAD-ID	TCB@	ACC_TIME	TAG
STCRSE	0EDE540000000000	005E6B60	822	1/ThreadPoolProcess
STCRSE	0EDE870000000001	005E69C8	001	
STCRSE	0EDE980000000002	005E6518	1814	
STCRSE	0EDEBA0000000003	005E66B0	2305	
STCRSE	0EDECB0000000004	005E62F8	001	
STCRSE	0EDED00000000005	005E60D8	001	
STCRSE	0EDF860000000006	005C2BF8	628	6/ThreadPoolMonitor\$Memory UsageMonitor
STCRSE	0EDF970000000007	005C2D90	003	7/ThreadPoolMonitor
IBMUSER	0EE2C700000000024	005C08B0	050	38/JESMiner
IBMUSER	0EE2B600000000026	005C0690	004	40/FAMiner
IBMUSER	0EE30B00000000027	005C0250	002	41/LuceneMiner
IBMUSER	0EE31C00000000028	005C0030	002	42/CDTParserMiner
IBMUSER	0EE32D00000000029	005BDE00	002	43/MVSLuceneMiner
IBMUSER	0EE33E0000000002A	005BDBE0	002	44/CDTMVSParserMiner

- Cuando finaliza un proceso de agrupación de hebras RSE, muestra detalles con las estadísticas de uso de recursos, como si se emitiera el mandato de modificación **DISPLAY PROCESS,DETAIL** sólo para dicho proceso de agrupación de hebras RSE. La marca de límite superior muestra el uso de recursos simultáneos máximo mientras dura el proceso de agrupación de hebras RSE, permitiendo al ajustador del sistema determinar si los recursos asignados a RSE están asignados por exceso o por defecto.

Supervisión de z/OS UNIX

La mayoría de límites de z/OS UNIX que afectan a Developer for System z se pueden visualizar con mandatos de operador. Algunos mandatos muestran incluso el uso simultáneo y la marca de límite superior para un límite concreto. Consulte la publicación *MVS System Commands* (SA22-7627) para obtener más información sobre estos mandatos.

- La directiva LIMMSG(ALL) de SYS1.PARMLIB(BPXPRMxx) hace que z/OS UNIX muestre los mensajes de consola (BPXI040I) cuando se está a punto de alcanzar alguno de los límites de parmlib. El valor predeterminado de LIMMSG es NONE, que inhabilita la función. Utilice el mandato de operador **SETOMVS LIMMSG=ALL** para activar dinámicamente esta función (hasta la próxima IPL). Consulte el manual *MVS Initialization and Tuning Reference* (SA22-7592) para obtener más información sobre esta directiva.
- El mandato de operador **DISPLAY OMVS,OPTIONS** muestra los valores actuales de las directivas z/OS UNIX que se pueden establecer dinámicamente.

```
d omvs,o
BPX0043I 13.10.16 DISPLAY OMVS 066
OMVS      000D ETC/INIT WAIT  OMVS=(M7)
CURRENT UNIX CONFIGURATION SETTINGS:
MAXPROCSYS      =      256      MAXPROCUSER      =      16
MAXFILEPROC     =      256      MAXFILESIZE       = NOLIMIT
MAXCPUPTIME     =      1000     MAXUIDS        =      200
MAXPTY          =      256
MAXMMAPAREA     =      256      MAXASSIZE         = 209715200
MAXTHREADS      =      200      MAXTHREADTASKS    =      1000
MAXCORESIZE     =      4194304  MAXSHAREPAGES  =      4096
IPCMSGQBYTES    = 2147483647    IPCMSGQMNUM    =      10000
IPCMSGNIDS      =      500      IPCSEMNIDS      =      500
IPCSEMNOPS      =      25       IPCSEMNSEMS    =      1000
IPCshmPAGES     =      25600    IPCshmNIDS     =      500
IPCshmNSEGS     =      500      IPCshmSPAGES   = 262144
SUPERUSER       = BPXROOT      FORKCOPY         = COW
STEPLIBLIST     =
USERIDALIASTABLE=
SERV_LINKLIB    = POSIX.DYN SERV.LOADLIB  BPXLK1
SERV_LPALIB     = POSIX.DYN SERV.LOADLIB  BPXLK1
PRIORITYPG VALUES: NONE
```

```

PRIORITYGOAL VALUES: NONE
MAXQUEUEDSIGs = 1000    SHRLIBRGNSIZE = 67108864
SHRLIBMAXPAGES = 4096    VERSION = /
SYSCALL COUNTS = NO     TTYGROUP = TTY
SYSPLEX = NO           BRML SERVER = N/A
LIMMSG = NONE          AUTOCVT = OFF
RESOLVER PROC = DEFAULT
AUTHPGMLIST = NONE
SWA = BELOW

```

- El mandato de operador **DISPLAY OMVS,LIMITS** muestra información sobre los límites de parmlib de los Servicios de sistemas z/OS UNIX actuales, las marcas de nivel superior y el uso del sistema actual.

```

d omvs,l
BPX0051I 14.05.52 DISPLAY OMVS 904
OMVS 0042 ACTIVE OMVS=(69)
SYSTEM WIDE LIMITS: LIMMSG=SYSTEM

```

	CURRENT USAGE	HIGHWATER USAGE	SYSTEM LIMIT
MAXPROCSYS	1	4	256
MAXUIDS	0	0	200
MAXPTYs	0	0	256
MAXMMAPAREA	0	0	256
MAXSHAREPAGES	0	10	4096
IPCMSGNIDS	0	0	500
IPCSEMNIDS	0	0	500
IPCSHMNIDS	0	0	500
IPCSHMSPAGES	0	0	262144 *
IPCMSGQBYTES	---	0	262144
IPCMSGQMNUM	---	0	10000
IPCSHMMPAGES	---	0	256
SHRLIBRGNSIZE	0	0	67108864
SHRLIBMAXPAGES	0	0	4096

El mandato muestra las marcas de nivel superior y el uso actual de un proceso individual cuando se especifica también la palabra clave PID=processid.

```

d,omvs,l,pid=16777456
BPX0051I 14.06.28 DISPLAY OMVS 645
OMVS 000E ACTIVE OMVS=(76)
USER  JOBNAME  ASID      PID      PPID STATE  START  CT_SECS
STCRSE RSED8    007E    16777456  67109106 HF---- 20.00.56 113.914
LATCHWAITPID= 0 CMD=java -Ddaemon.log=/var/rdz/logs -
PROCESS LIMITS: LIMMSG=NONE

```

	CURRENT USAGE	HIGHWATER USAGE	PROCESS LIMIT
MAXFILEPROC	83	103	256
MAXFILESIZE	---	---	NOLIMIT
MAXPROCUSER	97	99	200
MAXQUEUEDSIGs	0	1	1000
MAXTHREADS	9	14	200
MAXTHREADTASKS	9	14	1000
IPCSHMNSEGS	0	0	500
MAXCORESIZE	---	---	4194304
MAXMEMLIMIT	0	0	16383P

- El mandato de operador **DISPLAY OMVS,PFS** muestra información sobre cada sistema de archivos físico que forma actualmente parte de la configuración de z/OS UNIX, que incluye pilas TCP/IP.

```

d omvs,p
BPX0046I 14.35.38 DISPLAY OMVS 092
OMVS 000E ACTIVE OMVS=(33)
PFS CONFIGURATION INFORMATION

```

PFS TYPE	DESCRIPTION	ENTRY	MAXSOCK	OPNSOCK	HIGHUSED
TCP	SOCKETS AF_INET	EZBPFINI	50000	244	8146
UDS	SOCKETS AF_UNIX	BPXTUINT	64	6	10

```

ZFS          LOCAL FILE SYSTEM      IOEFSCM
          14:32.00 RECYCLING
HFS          LOCAL FILE SYSTEM      GFUAINIT
BPXFTCLN     CLEANUP DAEMON         BPXFTCLN
BPXFTSYN     SYNC DAEMON             BPXFTSYN
BPXFPINT     PIPE                    BPXFPINT
BPXFCSIN     CHAR SPECIAL            BPXFCSIN
NFS          REMOTE FILE SYSTEM      GFSCINIT
PFS NAME     DESCRIPTION             ENTRY   STATUS   FLAGS
TCP41        SOCKETS                  EZBPFINI ACT     CD
TCP42        SOCKETS                  EZBPFINI ACT
TCP43        SOCKETS                  EZBPFINI INACT  SD
TCP44        SOCKETS                  EZBPFINI INACT
PFS PARM INFORMATION
HFS          SYNCDEFAULT(60) FIXED(50) VIRTUAL(100)
          CURRENT VALUES: FIXED(55) VIRTUAL(100)
NFS          biod(6)

```

- El mandato de operador **DISPLAY OMVS,PID=processid** muestra la información de hebras de un proceso específico.

```

d omvs,pid=16777456
BPX0040I 15.30.01 DISPLAY OMVS 637
OMVS      000E ACTIVE                OMVS=(76)
USER      JOBNAME  ASID              PID      PPID STATE   START   CT_SECS
STCRSE    RSED8    007E    16777456    67109106 HF---- 20.00.56 113.914
LATCHWAITPID=      0 CMD=java -Ddaemon.log=/var/rdz/logs -
THREAD_ID   TCB0     PRI_JOB  USERNAME  ACC_TIME SC  STATE
0E08A00000000000 005E6DF0 OMVS          .927 RCV  FU
0E08F00000000000 005E6C58          .001 PTX  JYNV
0E09300000000000 005E6AC0          7.368 PTX  JYNV
0E0CB00000000000 005C2CF0 OMVS          1.872 SEL  JFNV
0E1920000000003CE 005A0B70 OMVS    IBMUSER  14.088 POL  JFNV
0E18D0000000003CF 005A1938    IBMUSER   .581 SND  JYNV

```

Supervisar la red

Cuando se admite que un número grande de clientes se conecten al host, no sólo Developer for System z debe ser capaz de manejar la carga de trabajo, sino que la infraestructura de red también debe serlo. La gestión de redes es un tema amplio y bien documentado que no está dentro del ámbito de la documentación de Developer for System z. Por ello, únicamente se facilitan los siguientes puntos:

- El mandato de operador **DISPLAY NET,CSM** le permite supervisar el uso del almacenamiento gestionado por el Communications Storage Manager (CSM). Puede utilizar este mandato para determinar qué cantidad de almacenamiento de CSM se está utilizando para ECSA y para agrupaciones de almacenamiento de espacios de datos, tal como se documenta en *Communications Server SNA Operations* (SC31-8779).

Supervisión de sistemas de archivos z/OS UNIX

Developer for System z utiliza sistemas de archivos de z/OS UNIX para almacenar varios tipos de datos, como archivos de registro y temporales. Utilice el mandato de z/OS UNIX **df** para ver cuántos descriptores de archivos quedan disponibles, y cuánto espacio libre queda hasta que se crea la siguiente extensión del conjunto de datos HFS o zFS subyacente.

```

$ df
Montado en      Filesystem          Disp/Total      Archivos  Estado
/tmp            (OMVS.TMP)         1393432/1396800 4294967248 Disponible
/u/ibmuser      (OMVS.U.IBMUSER)   1248/1728        4294967281 Disponible
/usr/lpp/rdz    (OMVS.LPP.FEK)     3062/43200       4294967147 Disponible
/var            (OMVS.VAR)         27264/31680      4294967054 Disponible

```

Configuración de ejemplo

La siguiente configuración de ejemplo muestra la configuración necesaria para soportar estos requisitos:

- 500 conexiones de cliente simultáneas
- 300 construcciones de MVS simultáneas (trabajo por lotes)
- 200 conexiones de CARMA simultáneas (utilizando el método de inicio CRASTART)
- 3 horas de tiempo de espera de inactividad
- no permitir el uso de z/OS UNIX
- SCLM Developer Toolkit no se utiliza
- Previsión de un uso medio de almacenamiento dinámico Java de 20 MB
- Los usuarios tienen UID de z/OS UNIX exclusivos
- Las agrupaciones de hebra operan en modalidad de extractor de varias hebras

Recuento de agrupaciones de hebras

De forma predeterminada, Developer for System z intenta añadir 30 usuarios a una única agrupación de hebras. Sin embargo, nuestros requisitos indican que el tiempo de espera de inactividad estará activo. La Tabla 29 en la página 90 muestra que esto añadirá una hebra por cada cliente conectado. Esta hebra es una hebra temporizadora, por lo que está activa constantemente. Esto impedirá que RSE coloque a 30 usuarios en una única agrupación de hebras, dado que $10 + 30 \times (17 + 1) = 550$ y `maximum.threads` se ha establecido en 520 de forma predeterminada.

Podríamos aumentar `maximum.threads`, pero debido al requisito de tener una media de 20 MB de almacenamiento dinámico Java por usuario, decidimos reducir `maximum.clients` a 25 ($10 + 25 \times 18 = 460$). Con esto, sigue dentro del tamaño de almacenamiento dinámico Java máximo de 512 MB predeterminado ($20 \times 25 = 500$).

Con 25 clientes por agrupación de hebras y la necesidad de admitir 500 conexiones, sabemos que necesitaremos 20 espacios de direcciones de agrupaciones de hebras.

Determinar los límites mínimos

Utilizando las fórmulas mostradas anteriormente en este capítulo y los criterios indicados al principio de esta sección, podemos determinar el uso de los recursos que se deben acomodar.

- Recuento de espacios de direcciones - máximo
$$3 + 2 \times A + N \times (x + y + z) + (2 + N \times 0.01)$$
$$3 + 2 \times 20 + 500 \times 1 + 200 \times 1 + 300 \times 1 + (2 + 500 \times 0.01) = 1050$$
- Recuento de espacios de direcciones - por usuario
$$x + y + z$$
$$1 + 1 + 1 = 3$$
- Recuento de procesos - máximo
$$6 + 3 \times A + N \times (x + y + z) + (10 + N \times 0.05)$$
$$6 + 3 \times 20 + 500 \times 2 + 200 \times 1 + 300 \times 0 + (10 + 500 \times 0.05) = 1591$$
- Recuento de procesos - STCRSE
$$4 + 3 \times A$$
$$4 + 3 \times 20 = 64$$

- Recuento de procesos - por usuario
 $(x + y + z) + 5*s$
 $(2 + 1 + 0) + 5*0 = 3$
- Recuento de hebras - agrupación de hebras RSE
 $12 + N*(19 + x + y + z) + (20 + N*0.1)$
 $12 + 25*(19 + 1 + 4 + 0) + (20 + 25*0.1) = 635$
- Recuento de hebras - Supervisor de trabajos JES
 $3 + N + (20 + N*0.1)$
 $3 + 500 + (20 + 500*0.1) = 573$
- Recuento de hebras – Gestor de depuración
4
4
- ID de usuario
 $500 + 3 = 503$
Los 3 ID de usuario adicionales son para STCJMON, STCDBM y STCRSE, los ID de usuario de tarea iniciada de Developer for System z.

Definición de límites

Ahora que conocemos los números del uso de recursos, podemos personalizar las directivas de limitación con los valores adecuados.

- /etc/rdz/rsed.envvars
 - Xmx512m

no cambia
 - Dmaximum.clients=25
 - Dmaximum.threads=520

no cambia
 - Dminimum.threadpool.process=10
Este cambio es opcional; RSE iniciará nuevas agrupaciones de hebras según sea necesario
 - DDSTORE_USE_THREADED_MINERS=true
 - DHIDE_ZOS_UNIX=true
 - DDSTORE_IDLE_SHUTDOWN_TIMEOUT=10800000
- FEK.#CUST.PARMLIB(FEJJCNFG)
 - MAX_THREADS=573
- SYS1.PARMLIB(BPXPRMxx)
 - MAXPROCSYS(2500)

mínimo de 1591, almacenamiento intermedio extra añadido para otras tareas que las de Developer for System z
 - MAXPROCUSER(100)

mínimo de 64, almacenamiento intermedio extra añadido en caso de que las agrupaciones de hebra de RSE den soporte a menos de los 25 clientes previstos

- MAXTHREADS(1500)

debe ser, como mínimo, 573 (para el Supervisor de trabajos JES) si THREADSMAX en el segmento OMVS del ID de usuario STCRSE se utiliza para establecer el límite para RSE (mínimo 635)

- MAXTHREADTASKS(1500)

debe ser, como mínimo, 573 (para el Supervisor de trabajos JES) si THREADSMAX en el segmento OMVS del ID de usuario STCRSE se utiliza para establecer el límite para RSE (mínimo 635)

- MAXUIDS(700)

mínimo de 503, almacenamiento intermedio extra añadido para tareas que no sean las de Developer for System z

- MAXASSIZE(209715200)

no cambia (valor predeterminado del sistema 200 MB), utilizamos ASSIZEMAX en el segmento OMVS del ID de usuario STCRSE

- SYS1.PARMLIB(IEASYSxx)

- MAXUSER=2000

mínimo de 1050, almacenamiento intermedio extra añadido para otras tareas que las de Developer for System z

- Segmento OMVS del ID de usuario STCRSE

- ASSIZEMAX(2147483647)

2 GB

Utilización de recursos de supervisor

Después de activar los límites del sistema según se explica en “Definición de límites” en la página 119, podemos empezar a supervisar la utilización de recursos por parte de Developer for System z para ver si es necesario ajustar varias variables. La Figura 31 en la página 121 muestra el uso de recursos después de que 499 usuarios hayan iniciado sesión. (El ejemplo de la figura muestra sólo el inicio de sesión. No se han indicado acciones de usuario en el ejemplo.)

```

F RSED,APPL=D P
BPXM023I (STCRSE)
ProcessId(83886168) Memory Usage(17%) Clients(25) Order(1)
ProcessId(91 ) Memory Usage(17%) Clients(25) Order(2)
ProcessId(122 ) Memory Usage(17%) Clients(25) Order(3)
ProcessId(16777348) Memory Usage(17%) Clients(25) Order(4)
ProcessId(16777358) Memory Usage(17%) Clients(25) Order(5)
ProcessId(16777368) Memory Usage(17%) Clients(25) Order(6)
ProcessId(16777378) Memory Usage(17%) Clients(25) Order(7)
ProcessId(16777388) Memory Usage(17%) Clients(25) Order(8)
ProcessId(16777398) Memory Usage(17%) Clients(25) Order(9)
ProcessId(33554622) Memory Usage(17%) Clients(25) Order(10)
ProcessId(16777416) Memory Usage(17%) Clients(25) Order(11)
ProcessId(16777426) Memory Usage(17%) Clients(25) Order(12)
ProcessId(16777436) Memory Usage(9%) Clients(25) Order(13)
ProcessId(16777446) Memory Usage(17%) Clients(25) Order(14)
ProcessId(16777456) Memory Usage(17%) Clients(25) Order(15)
ProcessId(16777466) Memory Usage(17%) Clients(25) Order(16)
ProcessId(16777476) Memory Usage(17%) Clients(25) Order(17)
ProcessId(16777487) Memory Usage(17%) Clients(25) Order(18)
ProcessId(16777497) Memory Usage(17%) Clients(25) Order(19)
ProcessId(16777507) Memory Usage(16%) Clients(24) Order(20)

```

```

F RSED,APPL=D P,D
BPXM023I (STCRSE)
ProcessId(83886168) ASId(0022) JobName(RSED857 ) Order(1)
PROCESS LIMITS:      CURRENT  HIGHWATER    LIMIT
  JAVA HEAP USAGE(%)    17        17        100
    CLIENTS              25        25         25
  MAXFILEPROC           365       366      64000
  MAXPROCUSER           64        64        100
  MAXTHREADS            362       363      1500
  MAXTHREADTASKS        363       363      1500

```

TASID			
Jobname	Cpu time	Storage	EXCP
-----	-----	-----	-----
JMON	0.00	1780	73
RSED	5.88	95.2M	41958
RSED1	8.26	190.1M	58669
RSED1	8.17	187.0M	58605
RSED2	8.06	185.3M	58653
RSED2	8.19	183.1M	60209
RSED3	8.12	189.1M	58650
RSED3	8.03	186.7M	58590
RSED4	8.15	188.2M	58646
RSED4	5.50	182.5M	58585
RSED5	7.72	184.4M	58631
RSED5	7.82	184.1M	58576
RSED6	7.14	184.1M	58622
RSED6	6.27	186.9M	58583
RSED7	5.17	185.1M	58804
RSED7	6.57	185.2M	58621
RSED7	5.86	182.8M	58565
RSED8	0.36	1560	2459
RSED8	7.94	184.1M	58615
RSED8	7.45	181.8M	58548
RSED9	8.16	190.6M	58802
RSED9	7.62	183.8M	58610
RSED9	7.36	177.7M	57478

Figura 31. Utilización de recursos de configuración de ejemplo

Capítulo 6. Consideraciones sobre el rendimiento

z/OS es un sistema operativo sumamente personalizable, y los cambios de sistema (a veces pequeños) pueden afectar considerablemente al rendimiento global. En este capítulo se resaltan algunos de los cambios que se pueden hacer para mejorar el rendimiento de Developer for System z.

Consulte las publicaciones *MVS Initialization and Tuning Guide* (SA22-7591) y *UNIX System Services Planning* (GA22-7800) para obtener más información acerca del ajuste del sistema.

Utilizar sistemas de archivos zFS

zFS (sistema de archivos de zSeries) y HFS (sistema de archivos jerárquico) son sistemas de archivos UNIX que pueden utilizarse en un entorno z/OS UNIX. Sin embargo, zFS proporciona las siguientes ventajas y características:

- Aumento del rendimiento en muchos entornos de cliente al acceder a archivos con un tamaño cercano a 8K que se actualizan con frecuencia. El rendimiento del acceso a archivos de menor tamaño es equivalente al del HFS.
- Clonación solo de lectura de un sistema de archivos en el mismo conjunto de datos. El sistema de archivos clonado se puede poner a disposición de los usuarios para proporcionar una copia puntual solo de lectura de un sistema de archivos. Esta es una característica opcional que solo está disponible en un entorno que no sea sysplex.
- zFS es el sistema de archivos estratégico de z/OS UNIX. La funcionalidad del HFS se ha estabilizado y las mejoras realizadas en el sistema de archivos solo serán para zFS.

Consulte la publicación *UNIX System Services Planning* (GA22-7800) para obtener más información acerca de zFS.

Evitar el uso de STEPLIB

Cada proceso z/OS UNIX que tenga una STEPLIB que se propague de padre a hijo o a través de un exec consumirá unos 200 bytes de ECSA (área de almacenamiento común ampliada). Si no se define ninguna variable de entorno STEPLIB, o si se define como STEPLIB=CURRENT, z/OS UNIX propaga todas las asignaciones de TASKLIB, STEPLIB y JOBLIB actualmente activas durante una función fork(), spawn() o exec().

Developer for System z tiene el valor predeterminado STEPLIB=NONE codificado en `rshed.envvars`, como se describe en la sección dedicada al archivo de configuración `rshed.envvars`. Por los motivos mencionados más arriba, no debe cambiar esta directiva, y sí debería colocar los conjuntos de datos tomados como objetivo en LINKLIST o en LPA (área de módulos residentes).

Mejorar el acceso a las bibliotecas del sistema

Algunas bibliotecas del sistema y algunos módulos de carga se utilizan intensivamente en z/OS UNIX y en las actividades de desarrollo de aplicaciones. El hecho de mejorar el acceso a ellas (por ejemplo, añadirlas al área de módulos residentes, LPA) puede mejorar el rendimiento del sistema. En el manual *MVS*

Initialization and Tuning Reference (SA22-7592) hallará más información sobre cómo cambiar los miembros SYS1.PARMLIB descritos a continuación:

Bibliotecas de tiempo de ejecución de Language Environment (LE)

Los programas C (incluida la shell de z/OS UNIX), cuando se ejecutan, suelen utilizar rutinas de la biblioteca de tiempo de ejecución de Language Environment (LE). Como promedio, unos 4 MB de la biblioteca de tiempo de ejecución se cargan en memoria para cada espacio de direcciones que se ejecute en un programa habilitado para LE, y se copian en cada bifurcación.

CEE.SCEELPA

El conjunto de datos CEE.SCEELPA contiene un subconjunto de rutinas de tiempo de ejecución de LE, que se utilizan muy a menudo en z/OS UNIX. Debe añadir este conjunto de datos a SYS1.PARMLIB(LPALSTxx) para obtener el máximo rendimiento. Así, los módulos se leen del disco una sola vez y se colocan en una ubicación compartida.

Nota: Añada la siguiente sentencia a SYS1.PARMLIB(PROGxx) si prefiere añadir los módulos de carga a la LPA (área de módulos residentes) dinámica:

```
LPA ADD MASK(*) DSN(CEE.SCEELPA)
```

Conviene asimismo colocar las bibliotecas de tiempo de ejecución de LE CEE.SCEERUN y CEE.SCEERUN2 en LINKLIST, añadiendo los conjuntos de datos a SYS1.PARMLIB(LNKSTxx) o a SYS1.PARMLIB(PROGxx). Ello elimina la actividad adicional que supone utilizar la STEPLIB de z/OS UNIX, y se reduce la entrada/salida debido a la gestión por parte de LLA y VLF, o de productos similares.

Nota: Añada la biblioteca de clases DLL de C/C++ CBC.SCLBDLL también a LINKLIST, por los mismos motivos.

Si decide que no quiere colocar estas bibliotecas en LINKLIST, debe configurar la sentencia STEPLIB pertinente en rsed.envvars, como se describe en la sección dedicada al archivo de configuración rsed.envvars. Aunque este método siempre utiliza almacenamiento virtual adicional, podrá mejorar el rendimiento definiendo las bibliotecas de tiempo de ejecución de LE en LLA o en un producto similar. Esto reduce la E/S necesaria para cargar los módulos.

Desarrollo de aplicaciones

En los sistemas cuya actividad principal es el desarrollo de aplicaciones, el rendimiento también podrá mejorar si coloca el editor de enlaces en la LPA dinámica, añadiendo las líneas siguientes a SYS1.PARMLIB(PROGxx):

```
LPA ADD MODNAME(CEEBINIT,CEEBLIBM,CEEEV003,EDCV) DSN(CEE.SCEERUN)
LPA ADD MODNAME(IEFIB600,IEFXB603) DSN(SYS1.LINKLIB)
```

En el caso del desarrollo C/C++, también puede añadir el conjunto de datos de compilador CBC.SCCNCP a SYS1.PARMLIB(LPALSTxx).

Las sentencias anteriores son ejemplos de posibles candidatos a la LPA, pero las necesidades en su local puede ser distintas. Consulte la publicación *Language Environment Customization* (SA22-7564) para obtener más información acerca de la colocación de otros módulos de carga de LE en la LPA dinámica. Consulte la

publicación *UNIX System Services Planning* (GA22-7800) para obtener más información acerca de la colocación de módulos de carga de compilador C/C++ en la LPA dinámica.

Mejorar el rendimiento de la comprobación de seguridad

Para mejorar el rendimiento de la comprobación de seguridad que se realiza para z/OS UNIX, defina el perfil BPX.SAFFASTPATH en la clase FACILITY del software de seguridad. Así se reduce la actividad adicional que supone realizar comprobaciones de seguridad en z/OS UNIX para una gran variedad de operaciones. Entre ellas está la comprobación de acceso a los archivos de inclusión, la comprobación de acceso a IPC y la comprobación de ser propietario del proceso. Para obtener más información sobre este perfil, consulte *UNIX System Services Planning* (GA22-7800).

Nota: Los usuarios no necesitan tener autorización sobre el perfil BPX.SAFFASTPATH.

Gestión de cargas de trabajo

Cada local tiene sus necesidades específicas, y puede personalizar el sistema operativo z/OS para poder sacar el mayor partido de los recursos disponibles y responder a dichas necesidades. Con la gestión de cargas de trabajo (WLM), se definen objetivos de rendimiento y se asigna una importancia de negocio a cada objetivo. Los objetivos se definen para el trabajo en términos de negocio, y el sistema decide la cantidad de recursos (por ejemplo, la cantidad de CPU y de almacenamiento) que hay que dar al trabajo para responder a su objetivo.

El rendimiento de Developer for System z se puede equilibrar estableciendo los objetivos correctos para sus procesos. A continuación figuran algunas directrices generales:

- Al utilizarlo, asigne la transacción APPC a un grupo de rendimiento TSO.
- Asigne un grupo de rendimiento de tareas iniciadas (SYSSTC) a los espacios de direcciones del servidor de Developer for System z: Supervisor de trabajos JES (JMON), Daemon RSE (RSED) y Agrupación de hebras RSE (RSEDx).

Consulte la publicación *MVS Planning Workload Management* (SA22-7602) para obtener más información sobre este tema.

Almacenamiento dinámico Java de tamaño fijo

Con una memoria dinámica de tamaño fijo, no se producen ampliaciones ni contracciones de la memoria dinámica, lo que puede aumentar notablemente el rendimiento en algunas situaciones. Sin embargo, el hecho de utilizar una memoria dinámica de tamaño fijo no suele ser una buena idea, porque retarda el inicio de la recogida de basura hasta que la memoria dinámica esté llena, y en ese momento pasará a ser una tarea importante. También aumenta el riesgo de fragmentación, lo que exige una compactación de la memoria dinámica. Por lo tanto, solo debe utilizar memorias dinámicas de tamaño fijo después de haberlas probado debidamente o cuando así lo indique el centro de soporte de IBM. Consulte la publicación *Java Diagnostics Guide* (SC34-6650) para obtener más información acerca de los tamaños del almacenamiento dinámico y la recogida de basura.

Los tamaños inicial y máximo del almacenamiento dinámico de una z/OS Java Virtual Machine (JVM) se pueden establecer con las opciones de línea de mandatos -Xms (inicial) y -Xmx (máximo) de Java.

En Developer for System z, las opciones de línea de mandatos Java se definen en la directiva `_RSE_JAVAOPTS` del archivo `rzed.envvars`, como se describe en el apartado "Definición de parámetros de inicio de Java adicionales con `_RSE_JAVAOPTS`" de la *Guía de configuración de host* (SC11-3660).

```
#_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Xms128m -Xmx128m"
```

Opción -Xquickstart de Java

Nota: La opción `-Xquickstart` de Java sólo resulta de utilidad si utiliza el método de inicio alternativo REXEC/SSH para el servidor RSE. Este método se describe en el apartado "(Opcional) Uso de REXEC (o SSH)" de la publicación *Guía de configuración de host* (SC11-3660).

La opción `-Xquickstart` puede utilizarse para mejorar el tiempo de inicio de algunas aplicaciones Java. `-Xquickstart` hace que el compilador JIT (Just In Time) se ejecute con un subconjunto de optimizaciones; es decir, una compilación rápida. Esta compilación rápida permite mejorar el tiempo de inicio.

La opción `-Xquickstart` es adecuada para aplicaciones de corta ejecución, especialmente para aquellas en las que el tiempo de ejecución no está concentrado en un pequeño número de métodos. `-Xquickstart` puede degradar el rendimiento si se utiliza en aplicaciones de larga ejecución que contienen métodos dinámicos.

Para habilitar la opción `-Xquickstart` para el servidor RSE, añada la directiva siguiente al final de `rzed.envvars`:

```
_RSE_JAVAOPTS="$_RSE_JAVAOPTS -Xquickstart"
```

Compartimiento de clases entre las JVM

La máquina virtual Java (JVM) de IBM versión 5 y posteriores permite compartir clases de aplicación y programa de arranque entre las JVM almacenándolas en una memoria caché de memoria compartida. El hecho de compartir clases reduce el consumo global de memoria virtual cuando hay más de una JVM que comparte una caché. El hecho de compartir clases también reduce el tiempo de arranque de una JVM después de haberse creado la caché.

La caché de clases compartidas es independiente de las JVM activas y persiste más allá del tiempo de vida de la JVM que creó la caché. Dado que la caché de clases compartidas persiste más allá del tiempo de vida de las JVM, la caché se actualiza dinámicamente para reflejar las modificaciones que se hayan podido hacer en los JAR o en las clases del sistema de archivos.

La actividad adicional que supone crear y poblar una caché nueva es mínima. El coste en tiempo del arranque de una sola JVM se suele situar entre 0 y el 5% más de tiempo si se compara con un sistema que no utilice clases compartidas, y depende de la cantidad de clases cargadas. La mejora del tiempo de arranque de JVM con una caché poblada se suele situar entre el 10% y el 40% menos de tiempo si se compara con un sistema que no utilice clases compartidas, y depende del sistema operativo y del número de clases que se carguen. Si hay múltiples JVM en ejecución concurrente, el tiempo de arranque global mejorará.

Consulte la publicación *Java SDK and Runtime Environment User Guide* para obtener más información acerca del compartimiento de clases.

Habilitar el compartimiento de clases

Para habilitar el compartimiento de clases para el servidor RSE, añade la directiva siguiente al final de `rsed.envvars`. La primera sentencia define una caché llamada RSE con acceso de grupo, y permite que el servidor RSE se inicie incluso si falla la prestación de compartir clases. La segunda sentencia es opcional y establece que el tamaño de la caché sea igual a 6 megabytes (el valor predeterminado del sistema es de 16 MB). La tercera sentencia añade los parámetros de compartimiento de clases a las opciones de inicio de Java.

```
_RSE_CLASS_OPTS=-Xshareclasses:name=RSE,groupAccess,nonFatal
# RSE_CLASS_OPTS="$ _RSE_CLASS_OPTS -Xscmx6m
_RSE_JAVA_OPTS="$ _RSE_JAVA_OPTS $ _RSE_CLASS_OPTS"
```

Nota: Como se ha mencionado en la sección “Seguridad de memoria caché”, todos los usuarios que utilizan la clase compartida deben tener el mismo ID de grupo primario (GID). Esto significa que los usuarios deben tener definido el mismo grupo predeterminado en el software de seguridad, o que los distintos grupos predeterminados tengan el mismo GID en el correspondiente segmento OMVS.

Límites de tamaño de la memoria caché

El tamaño máximo de la caché teórica compartida es 2 GB. El tamaño de caché que se puede especificar está limitado por la cantidad de memoria física y de espacio de intercambio físico disponible en el sistema. Dado que el espacio de direcciones virtuales de un proceso se comparte entre la memoria caché de clases compartidas y el almacenamiento dinámico de Java, el aumento del tamaño máximo del almacenamiento dinámico de Java reducirá el tamaño de la memoria caché de clases compartidas que puede crear.

Seguridad de memoria caché

El acceso a la caché de clases compartidas está limitado por los permisos del sistema operativo y por los permisos de la seguridad Java.

Por defecto, las cachés de clases se crean con seguridad a nivel de usuario, por lo que el usuario que ha creado la caché es el único que puede acceder a ella. En z/OS UNIX, existe una opción, `groupAccess`, que da acceso a todos los usuarios del grupo primario del usuario que creó la caché. Sin embargo, sea cual sea el nivel de acceso que se emplee, el único que puede destruir una caché es el usuario que la ha creado o un usuario root (UID 0).

Consulte *Java SDK and Runtime Environment User Guide* para obtener más información sobre las opciones de seguridad adicionales usando un `SecurityManager` de Java.

SYS1.PARMLIB(BPXPRMxx)

Algunos de los valores de `SYS1.PARMLIB(BPXPRMxx)` afectan al rendimiento de las clases compartidas. Si se emplean valores incorrectos, las clases compartidas podrían dejar de funcionar. Estos valores también podrían afectar al rendimiento. Para obtener más información acerca de las implicaciones sobre el rendimiento y la utilización de estos parámetros, consulte las publicaciones *MVS Initialization and Tuning Reference* (SA22-7592) y *UNIX System Services Planning* (GA22-7800). Los parámetros más significativos de `BPXPRMxx` que afectan a la operación de las clases compartidas son los siguientes:

- `MAXSHAREPAGES`, `IPCSHMSPAGES`, `IPCSHMMPAGES` y `IPCSHMNSEGS`

Estos valores afectan a la cantidad de páginas de memoria compartida disponibles para la JVM. El tamaño de páginas compartidas en el caso de un

servicio de sistema z/OS UNIX de 31 bits se ha fijado en 4 KB. Las clases compartidas intentan crear una caché de 16 MB por defecto. Por tanto, establezca IPCSHMPAGES en un valor superior a 4096.

Si establece un tamaño de caché utilizando -Xscmx, la JVM redondeará el valor por exceso al megabyte más cercano. Debe tenerlo en cuenta cuando establezca IPCSHMPAGES en su sistema.

- IPCSEMNIDS y IPCSEMNSEMS

Estos valores afectan a la cantidad de semáforos disponibles en los procesos UNIX. Las clases compartidas utilizan semáforos IPC para la comunicación entre máquinas virtuales Java (JVM).

Espacio de disco

La caché de clases compartidas necesita espacio en disco en el que almacenar información de identificación sobre las cachés que existen en el sistema. Esta información se almacena en /tmp/javasharedresources. Si el directorio de información de identificación se suprime, la JVM no puede identificar las clases compartidas en el sistema y debe volver a crear la caché.

Utilidades para la gestión de cachés

El mandato de línea Java -Xshareclasses puede tener varias opciones, algunas de las cuales son utilidades para la gestión de cachés. Tres de ellas se muestran en el ejemplo siguiente (\$ es el indicador de z/OS UNIX). Consulte la publicación *Java SDK and Runtime Environment User Guide* para obtener una visión general completa de las opciones de línea de mandatos soportadas.

```
$ java -Xshareclasses:listAllCaches
Caché compartida      OS shmid      en uso      Hora de última desconexión
RSE                   401412        0           Lun Jun 18 17:23:16 2007
```

No se ha podido crear la máquina virtual Java (JVM).

```
$ java -Xshareclasses:name=RSE,printStats
```

Estadísticas actuales de la caché "RSE":

```
dirección base      = 0x0F300058
dirección final     = 0x0F8FFFF8
puntero de asignación = 0x0F4D2E28
```

```
tamaño de caché     = 6291368
bytes libres        = 4355696
bytes de ROMClass   = 1912272
bytes de metadatos  = 23400
% de metadatos usados = 1%
```

```
nº de ROMClasses    = 475
nº de vías de acceso de clases = 4
nº de URL            = 0
nº de símbolos      = 0
nº de clases obsoletas = 0
% de clases obsoletas = 0%
```

La caché está 30% llena

No se ha podido crear la máquina virtual Java (JVM).

```
$ java -Xshareclasses:name=RSE,destroy
JVMSHRC010I La caché compartida "RSE" se ha destruido
No se ha podido crear la máquina virtual Java (JVM).
```

Nota:

- Los programas de utilidad de memoria caché realizan la operación necesaria en la memoria caché especificada sin iniciar la JVM, por lo que el mensaje "No se ha podido crear la máquina virtual Java." es normal.
- Una memoria caché sólo puede destruirse si todas las JVM que la utilizan se han cerrado y el usuario que emite el mandato tiene autorización suficiente.

Capítulo 7. Consideraciones sobre envío a cliente

Enviar a cliente o el control de clientes basado en host, tiene soporte para la gestión central de lo siguiente:

- Archivos de configuración del cliente
- Versión del producto del cliente
- Definiciones del proyecto

En este capítulo se tratan estos temas:

- "Introducción"
- "Sistema primario" en la página 132
- "Metadatos Envío a cliente" en la página 133
- "Control de configuración del cliente" en la página 134
- "Control de versión del cliente" en la página 135
- "Varios grupos de desarrollador" en la página 135
- "Selección de grupo basada en LDAP" en la página 140
- "Selección de grupo basada en SAF" en la página 145
- "Proyectos basados en host" en la página 149

Introducción

Los clientes de Developer for System z versión 8.0.1 y posteriores pueden tomar la información de actualización de producto y de los archivos de configuración del cliente desde el host cuando se conectan, asegurando que todos los cliente tienen valores comunes y que están actualizados.

Desde la versión 8.0.3, el administrador del cliente puede crear varios conjuntos de configuraciones de cliente y varios escenarios de actualización del cliente para ajustar las necesidades de distintos grupos de desarrolladores. Esto permite a los usuarios recibir una configuración personalizada basada en un criterio como la pertenencia de un grupo LDAP o permiso para un perfil de seguridad.

Los proyectos de z/OS se pueden definir individualmente a través de la perspectiva Proyectos de z/OS en el cliente; los proyectos de z/OS también se pueden definir de forma centralizada en el host y luego propagarse al cliente en base a usuarios individuales. Estos "proyectos basados en host" tienen el mismo aspecto y funcionan exactamente igual que proyectos definidos en el cliente, salvo porque el cliente no puede modificar la estructura, miembros y propiedades, y sólo se puede acceder a ellos cuando se está conectado al host.

`pushtoclient.properties` indica al cliente si estas funciones están habilitadas, y dónde se almacenan los datos relacionados. Para obtener más información, consulte "(Opcional) `pushtoclient.properties`, Control de cliente basado en host" en la *Guía de configuración del host* SC11-3660 (SC23-7658).

Por lo general, los sistemas z/OS, estaciones de trabajo de desarrollador y proyectos de desarrollo se gestionan mediante distintos grupos de personas. El diseño Envío a cliente sigue este principio y asigna obligaciones específicas a cada grupo:

- El programador del sistema z/OS controla la ubicación de los metadatos de Envío al cliente, los aspectos de seguridad básicos y si el Envío a cliente está activo.
- El administrador del cliente mantiene el contenido de los metadatos de envío a cliente utilizando el cliente de Developer for System z para crear una o más configuraciones cliente y utilizando IBM Installation Manager para crear archivos de respuestas que se utilizan para actualizar el cliente de Developer for System z.
- Un gestor de proyecto de desarrollo define un proyecto y le asigna desarrolladores individuales.

Para obtener detalles sobre cómo el administrador del cliente y el gestor del proyecto de desarrollo pueden realizar las tareas que tienen asignadas, consulte el Information Center Developer for System z Information Center (http://www-01.ibm.com/support/knowledgecenter/SSQ2R2_9.1.0/com.ibm.etools.getstart.wsentdev.doc/kc_version_welcome_rdz.html).

Cuando habilite el soporte de control de versión o configuración para varios grupos de desarrolladores, hay un equipo adicional implicado en la gestión del Envío a cliente. El equipo que será depende de la opción elegida para identificar los grupos a los que pertenece un desarrollador:

- Un administrador de LDAP mantiene definiciones de grupo que colocan a cada desarrollador en uno o más grupos de LDAP de FEK.PTC.*, o bien en ninguno.
- Un administrador de seguridad mantiene listas de acceso a perfiles de seguridad FEK.PTC.*. Un desarrollador puede estar autorizado para ninguno, uno o más perfiles. desarrollador.

Sistema primario

Envío a cliente está diseñado para almacenar datos específicos de sistema por cada sistema, mientras mantiene datos comunes (globales) en un único sistema (el sistema primario) para reducir el esfuerzo de gestión. El sistema primario está identificado por la directiva `primary.system` en `pushtoclient.properties`. El valor predeterminado es `false`.

Asegúrese de que tiene un sistema (y solamente uno) definido como sistema primario. Los administradores del cliente Developer for System z no pueden exportar datos de configuración global a menos que el sistema de destino sea un sistema primario. Los clientes de Developer for System z podrían mostrar un comportamiento errático al conectarse a varios sistemas primarios con configuraciones no sincronizadas.

La regla "sólo uno" no se aplica cuando varios sistemas comparten la configuración de Developer for System z (`/etc/rdz`) y los metadatos de Envío a cliente (`/var/rdz/pushtoclient`). Como la configuración está compartida, todos los sistemas implicados se indentifican como el sistema primario. Pero en tanto que todos los sistemas también comparten los metadatos, esta duplicación no es un problema.

Metadatos Envío a cliente

Ubicación de metadatos

La directiva `pushtoclient.folder` en `pushtoclient.properties` identifica el directorio base en el que se almacenan los metadatos de Envío a cliente. El valor predeterminado es `/var/rdz/pushtoclient`.

El directorio base tiene el archivo de configuración de Envío a cliente raíz, `keymapping.xml`. El resto de metadatos está en subdirectorios.

La mayoría de los subdirectorios se crean dinámicamente cuando el administrador de cliente exporta la configuración de estación de trabajo de Envío a cliente. Estos subdirectorios agrupan los metadatos por asunto, como correlaciones y preferencias. A medida que hay más componentes de cliente de Developer for System z disponibles para ser gestionados por el Envío a cliente, se crean más subdirectorios de forma dinámica. Consulte el asistente de exportación en el cliente de Developer for System z (**Archivo > Exportar... > Rational Developer for System z > Archivos de configuración**) para ver qué se almacena en estos subdirectorios.

Algunos subdirectorios se crea durante la personalización del host inicial. Estos subdirectorios tienen datos mantenidos manualmente por el administrador de cliente o el gestor del proyecto de desarrollo.

- `/var/rdz/pushtoclient/projects/` contiene los archivos de definición de proyectos basados en host. La ubicación real se especifica en el archivo `/var/rdz/pushtoclient/keymapping.xml`, de cuya creación y mantenimiento se ocupa el administrador del cliente Developer for System z. El gestor de proyectos o el desarrollador principal mantiene los archivos que contienen.
- `/var/rdz/pushtoclient/install/` tiene los archivos de configuración utilizados para actualizar la versión del producto del cliente al conectarse al host. La ubicación real se especifica en el archivo `/var/rdz/pushtoclient/keymapping.xml`, de cuya creación y mantenimiento se ocupa el administrador del cliente Developer for System z. El administrador del cliente mantiene los archivos que contienen.
- `/var/rdz/pushtoclient/install/responsefiles/` tiene los archivos de configuración utilizados para actualizar la versión del producto del cliente al conectarse al host. La ubicación real se especifica en el archivo `/var/rdz/pushtoclient/keymapping.xml`, de cuya creación y mantenimiento se ocupa el administrador del cliente Developer for System z. El administrador del cliente mantiene los archivos que contienen.

Para obtener más información sobre la creación de estos subdirectorios, consulte "Configuración de personalización" en el capítulo "Personalización básica" de la *Guía de configuración de host SC11-3660 (SC23-7658)*.

Seguridad de metadatos

De forma predeterminada (consulte la directiva `file.permission` en `pushtoclient.properties`), todos los archivos y directorios creados en el directorio base reciben la máscara de bits de permiso 775 (`rwrxwrx-x`), lo que permite al propietario y al grupo predeterminado del propietario acceso de lectura y grabación a la estructura de directorios y los archivos que contiene. El resto sólo tiene acceso de lectura a la estructura de directorios y los archivos que contiene.

Es importante establecer el UID de propietario (ID de usuario) y GID (ID de grupo) para estos directorios antes de iniciar la configuración del Envío a cliente.

Los mandatos RACF del ejemplo siguiente crean un grupo nuevo (RDZADMIN), le asignan un GUID exclusivo (2) y, hacen que sea el grupo predeterminado para el ID de usuario RDZADM1, que también recibe un UID exclusivo (6).

```
ADDGROUP RDZADMIN OWNER(IBMUSER) SUPGROUP(SYS1) -  
    DATA('RATIONAL DEVELOPER FOR SYSTEM Z - CLIENT ADMIN')  
ALTGROUP RDZADMIN OMVS(GID(2))  
CONNECT RDZADM1 GROUP(RDZADMIN) AUTH(USE)  
ALTUSER RDZADM1 DFLTGRP(RDZADMIN) OMVS(UID(6))
```

El mandato de ejemplo siguiente **chown** de z/OS UNIX cambia el propietario y el grupo de /var/rdz/pushtoclient y todo lo que incluye a RDZADM1 y RDZADMIN respectivamente. El mandato lo debería ejecutar un superusuario (UID 0) para evitar problemas de permiso.

```
chown -R rdzadm1:rdzadmin /var/rdz/pushtoclient
```

El mandato de ejemplo **chmod** siguiente de z/OS UNIX cambia la máscara de bits de permiso de /var/rdz/pushtoclient y todo lo que contiene a 775. Ejecútelos para asegurarse de que cualquier adición manual al directorio siga la lógica utilizada por Developer for System z. El mandato lo debería ejecutar un superusuario (UID 0) para evitar problemas de permiso.

```
chmod -R 775 /var/rdz/pushtoclient
```

Para obtener más información sobre los mandatos RACF de ejemplo, consulte *Security Server RACF Command Language Reference* (SA22-7687). Para obtener más información sobre los mandatos de z/OS UNIX de ejemplo, consulte *UNIX System Services Command Reference* (SA22-7802). Para obtener información adicional, consulte “Estructura de directorios de z/OS UNIX” en la página 15.

Uso del espacio de metadatos

Los metadatos de Enviar al cliente utilizan una cantidad de espacio en disco razonablemente pequeña en z/OS UNIX, porque el grueso de los metadatos está en archivos XML codificados en UTF-8. Tenga en cuenta el código de producto utilizado para los escenarios de actualización del cliente se puede almacenar en cualquier lugar de la red; no tiene que estar almacenado en z/OS UNIX, ya que los metadatos de Enviar al cliente relacionados (denominados archivos de respuestas), indican al cliente la ubicación correcta.

Control de configuración del cliente

Cuando un cliente de Developer for System z (versión 8.0.1. o posterior) se conecta al host, lee las definiciones en pushtoclient.properties. Si la directiva config.enabled está habilitada, el cliente compara su configuración actual con las definiciones en los metadatos de Envío al cliente. Si se encuentran diferencias, el cliente inicia un asistente que recupera los datos necesarios y activa la configuración según determine Envío al cliente.

La directiva reject.config.updates en pushtoclient.properties controla si un usuario tiene permiso para rechazar las actualizaciones de configuración que Envío al cliente está a punto de entregar.

Un cliente de Developer for System z (versión 8.0.1 y posteriores) tiene un asistente, que el administrador del cliente puede utilizar para exportar la configuración actual que, por otro lado, podrán importar todos los clientes de Developer for System z por medio de Envío al cliente. Tenga en cuenta que esta función está disponible en todos los clientes, por lo que debe asegurarse de que

sólo los administradores del cliente tengan permiso de escritura para los directorios de z/OS UNIX en los que se encuentran los metadatos del Envío al cliente (/var/rdz/pushtoclient).

Para habilitar el soporte de grupos, tanto el cliente como el host deben tener la versión 8.0.3 o posterior, según se indica en la documentación “Varios grupos de desarrollador”.

Control de versión del cliente

Cuando un cliente de Developer for System z (versión 8.0.1. o posterior) se conecta al host, lee las definiciones en `pushtoclient.properties`. Si la directiva `product.enabled` está habilitada, el cliente compara su versión del producto actual con las definiciones en los metadatos de Envío al cliente. Si se encuentran diferencias, el cliente inicia un asistente que recupera los datos necesarios y activa la configuración según determine Envío al cliente.

La directiva `reject.product.updates` en `pushtoclient.properties` controla si un usuario tiene permiso para rechazar las actualizaciones del producto que Envío al cliente está a punto de entregar.

Para habilitar el soporte de grupos, tanto el cliente como el host deben tener la versión 8.0.3 o posterior, según se indica en la documentación “Varios grupos de desarrollador”.

Varios grupos de desarrollador

Desde la versión 8.0.3, el administrador del cliente puede crear varios conjuntos de configuraciones de cliente y varios escenarios de actualización del cliente para ajustar las necesidades de distintos grupos de desarrolladores. Esto permite a los usuarios recibir una configuración personalizada basada en un criterio como la pertenencia de un grupo LDAP o permiso para un perfil de seguridad.

Activación

El soporte para varios grupos de desarrolladores, cada uno de ellos con su propios requisitos de actualización de cliente y configuración de cliente, se habilita mediante la asignación del valor desado a las directivas relacionadas (`config.enabled` y `product.enabled`) en `pushtoclient.properties`, según se indica en la Tabla 36.

Tabla 36. Matriz de soporte de grupo de Envío a cliente para `*.enabled`

<code>*.enabled value</code>	Función habilitada	Soporte para varios grupos
Falso	No	No
Verdadero	Sí	No
LDAP	Sí	Sí, basado en pertenencia a grupos LDAP FEK.PTC.*.ENABLED.sysname.devgroup
SAF	Sí	Sí, basado en permiso a perfiles de seguridad FEK.PTC.*.ENABLED.sysname.devgroup

Tenga en cuenta que cuando la función está habilitada (esto incluye el valor TRUE), los desarrolladores siempre son parte de un grupo predeterminado. Un desarrollador puede no ser parte de ningún grupo adicional, o bien ser parte de uno o varios grupos.

Se puede hacer que el rechazo de las actualizaciones también sea condicional, según se muestra en la Tabla 37.

Tabla 37. Matriz de soporte de grupo de Envío a cliente para reject..updates*

Valor reject.*.updates	Función habilitada
Falso	No
Verdadero	Sí
LDAP	Depende de la pertenencia a grupo LDAP FEK.PTC.REJECT.*.UPDATES.sysname.**
SAF	Depende del permiso al perfil de seguridad FEK.PTC.REJECT.*.UPDATES.sysname.**

Tenga en cuenta que las directivas de pushtoclient.properties funcionan de forma independiente entre ellas. Puede asignar cualquier valor con soporte a cualquier directiva. No hay requisito alguno por el que los valores deban ser iguales.

Consulte “Selección de grupo basada en LDAP” en la página 140 y “Selección de grupo basada en SAF” en la página 145 para obtener detalles sobre la configuración necesaria para la función respectiva. Para obtener más información sobre la habilitación de varios soportes de grupo, consulte "(Opcional) pushtoclient.properties, control del cliente basado en host" en la publicación *Guía de configuración de host* SC11-3660 (SC23-7658).

Concatenación de grupos

Cuando la función *.enabled está habilitada (esto incluye el valor TRUE) en pushtoclient.properties, los desarrolladores siempre son parte de un grupo predeterminado para la función relacionada. Un desarrollador puede no ser parte de ningún grupo adicional, o bien ser parte de uno o varios grupos.

Para limitar la complejidad de la aplicación de cambios definidos en varios grupos, Developer for System z limita las definiciones a utilizar, en base a una selección realizada por el usuario.

Tabla 38. Concatenaciones de grupo Envío a cliente

Grupos adicionales	Definiciones usadas
Ninguno	Valor predeterminado
Uno	Predeterminada o (predeterminada + grupo)
Varios	Predeterminada o (predeterminada + 1 grupo)

Developer for System z utiliza la lógica siguiente en la construcción y aplicación del conjunto de cambios:

1. Tomar las actualizaciones, si las hubiera, especificadas en las definiciones predeterminadas.
2. Tomar las actualizaciones especificadas en la definición de grupo seleccionada, si la hubiera, cambiando las predeterminadas si ya estuvieran ahí.

3. Aplicar las actualizaciones en el cliente.

Nota: Las actualizaciones pueden ser acciones de supresión, adición o preformato.

Enlace de espacio de trabajo

Aunque un desarrollador puede ser parte de varios grupos de forma simultánea, el espacio de trabajo del desarrollador no puede. El espacio de trabajo activo debe estar enlazado a un grupo de configuración específico (que puede ser el grupo predeterminado) para recibir actualizaciones de producto o de configuración. Una vez realizado el enlace, no se puede deshacer. Se debe crear un espacio de trabajo nuevo si hace falta un nuevo enlace de grupo.

Cuando un espacio de trabajo que no tenga enlace de grupo de configuración se conecta al host y `config.enabled` indica que la función Envío al cliente está activa, Developer for System z consulta todos los grupos de configuración para determinar a qué grupos pertenece el usuario y solicita al usuario seleccionar un grupo. Tras sucesivas conexiones, sólo se consulta al grupo seleccionado para ver si la pertenencia al grupo aún es válida.

Tabla 39. Enlaces de grupo de configuración de espacio de trabajo

<code>config.enabled</code>	Espacio de trabajo vinculado a este grupo de actualización configuración
Falso	Ninguno
Verdadero	Valor predeterminado
LDAP	Predeterminado o grupo (tras solicitud)
SAF	Predeterminado o grupo (tras solicitud)

Cuando un espacio de trabajo que no tenga enlace de grupo se conecta al host y `product.enabled` indica que la función Envío al cliente está activa, Developer for System z consulta todos los grupos para determinar a qué grupos pertenece el usuario, y solicita al usuario seleccionar un grupo. Tras sucesivas conexiones, sólo se consulta al grupo seleccionado para ver si la pertenencia al grupo aún es válida.

Tabla 40. Enlaces de grupo de producto de espacio de trabajo

<code>product.enabled</code>	Espacio de trabajo vinculado a este grupo de actualización de grupo
Falso	Ninguno
Verdadero	Valor predeterminado
LDAP	Predeterminado o grupo (tras solicitud)
SAF	Predeterminado o grupo (tras solicitud)

Las directivas `reject.*.updates` pueden trabajar con y sin definiciones de grupos. Si se utilizan grupos para `reject.*.updates`, se utiliza el enlace de grupo de la directiva relacionada `*.enabled`. Cuando hay una actualización, Developer for System z determina si el usuario tiene permiso para rechazarla, y actúa en consecuencia.

El soporte de grupo para las directivas `reject.*.updates` es nuevo en la versión 9.1.0 y requiere que el host y el cliente de Developer for System z estén en la versión 9.1.0 o posterior. El soporte cambia el modo en que se procesan las palabras clave LDAP y SAF.

Antes de la versión 9.1.0, estar en la lista de acceso para FEK.PTC.REJECT.*.UPDATES.sysname era suficiente para rechazar una actualización, independientemente del enlace de grupo de espacio de trabajo. Desde la versión 9.1.0, FEK.PTC.REJECT.*.UPDATES.sysname sólo se utiliza para rechazar actualizaciones por espacios de trabajo vinculados al grupo predeterminado. Los espacios de trabajo vinculados a un grupo requieren que esté en la lista de acceso para FEK.PTC.REJECT.*.UPDATES.sysname.groupname para rechazar actualizaciones.

Ubicación de metadatos de grupo

Según se indica en la documentación en “Ubicación de metadatos” en la página 133, todos los metadatos del Envío a cliente se almacenan en una estructura de directorios en la parte superior de /var/rdz/pushtoclient/ cuando se utiliza una configuración sin soporte para grupos. Se mantiene el mismo diseño de datos cuando el soporte de grupos está activado, pero con pequeñas diferencias de interpretación sobre el directorio base, /var/rdz/pushtoclient/:

- Los datos existentes en /var/rdz/pushtoclient/ se interpretan como los datos para el grupo predeterminado. La exportación al grupo predeterminado crea o actualiza los metadatos en /var/rdz/pushtoclient/. Esta interpretación asegura la compatibilidad con los clientes de versiones 8.0.1 y 8.0.2, que están habilitados para el Envío a cliente, pero no tienen soporte para grupos múltiples.
- La exportación a un grupo crea o actualiza los metadatos en /var/rdz/pushtoclient/grouping/<devgroup>/, como si estuvieran en el directorio base en vez de en /var/rdz/pushtoclient/. El valor de <devgroup> coincide con el nombre de grupo asignado a un grupo específico de desarrolladores.

La personalización inicial del producto crea el directorio grouping/ en /var/rdz/pushtoclient/. El administrador del cliente es responsable de añadir los directorios <devgroup>/ a /var/rdz/pushtoclient/grouping/.

Tenga en cuenta que durante la personalización inicial de producto, se crean los directorios projects/, install/ e install/responsefiles/ en /var/rdz/pushtoclient/. El administrador de cliente debe repetir las acciones make-directory (crear directorio) en /var/rdz/pushtoclient/grouping/<devgroup>/ si se necesitaran escenarios de actualización de productos específicos de grupos o proyectos basados en host específicos de grupos.

La secuencia de mandatos z/OS UNIX de ejemplo siguiente crea los subdirectorios con la máscara de bits de permisos correcta. Los mandatos los debe ejecutar el administrador del cliente para evitar problemas de propiedad.

```
saved_umask=$(umask)
umask 0000
cd /var/rdz/pushtoclient/grouping/
mkdir -m775 <devgroup>
cd <devgroup>
mkdir -m775 install
mkdir -m775 install/responsefiles
mkdir -m775 projects
umask $saved_umask
```

Para obtener más información sobre los mandatos de z/OS UNIX de ejemplo, consulte *UNIX System Services Command Reference* (SA22-7802).

Pasos de configuración

La configuración del soporte para varios grupos de desarrolladores precisa de cierta coordinación entre el programador del sistema z/OS, el administrador de

cliente y el administrador que gestiona los criterios de selección (administrador de LDAP o de seguridad). En la descripción siguiente del flujo de trabajo, el administrador de seguridad gestiona los criterios de selección:

1. El administrador de cliente pide al administrador de seguridad información sobre la configuración de agrupación de desarrolladores existente. La reutilización de la configuración existente acelera y simplifica la configuración del Envío a cliente.
2. El administrador de cliente determina cómo quiere estructurar el soporte de grupos múltiples, y quién debe ser parte de dichos grupos de Envío a cliente.

Nota:

- Siempre hay una configuración predeterminada establecida y un escenario de actualización de producto predeterminado.
 - Los conjuntos de cambios de Envío a cliente pueden incluir acciones de supresión, adición o preformato.
 - Los conjuntos de cambios de Envío a cliente pueden estar vacíos.
 - Un desarrollador puede no ser parte de ningún grupo de Envío a cliente, o bien ser parte de uno o varios grupos.
 - El administrador de cliente debe ser miembro de cada grupo de Envío a cliente.
3. El administrador de cliente y el administrador de seguridad deben acordar los nombres de los grupos de Envío a cliente que deben utilizarse.
 4. El administrador del cliente crea el directorio
`/var/rdz/pushtoclient/grouping/<devgroup>`

para cada grupo de Envío a cliente.

Nota: Los bits de permiso para este directorio deben ser 775 (drwxrwxr-x).

5. El administrador de seguridad realiza la configuración inicial necesaria para definir los perfiles de selección de Envío a cliente y añade los grupos de Envío a cliente a las listas de acceso.

Nota:

- Las estructuras de los criterios de selección deben estar definidas con al menos el administrador de cliente en la lista de acceso antes de que éste pueda crear los metadatos de Envío a cliente relacionados.
 - Para la configuración inicial, únicamente el administrador del cliente debe estar en la lista de acceso para un grupo de Envío a cliente. Esto es para evitar que los clientes de Developer for system z reciban configuraciones que estén en construcción.
6. El programador del sistema z/OS activa el soporte para grupos múltiples ajustando `pushtoclient.properties`.

Nota: Las directivas `*.enabled` deben estar habilitadas para que el administrador de cliente pueda crear los metadatos de Envío a cliente relacionados.

7. El administrador de cliente crea los espacios de trabajo para cada grupo y los exporta al host utilizando los nombres de grupo respectivos. El administrador de cliente también crea los archivos de respuestas necesarios para crear los escenarios de actualización del producto específico del grupo.
8. El administrador de seguridad añade los desarrolladores a los grupos de Envío a cliente, activando el envío a cliente para los desarrolladores.

Selección de grupo basada en LDAP

Aunque LDAP (Lightweight Directory Access Protocol) es el nombre de un protocolo basado en TCP/IP, se suele utilizar para describir un conjunto de servicios de directorio distribuidos. Como una base de datos, un directorio es un conjunto estructurado de registros. Developer for System z puede utilizar un servidor LDAP como base de datos jerárquica sencilla, en la que los grupos tienen uno o más miembros.

Cuando utilice definiciones en su servidor LDAP como mecanismo de selección (el valor de LDAP se especifica para las directivas en `pushtoclient.properties`), Developer for System z comprueba la pertenencia de los nombres de grupos listados en la Tabla 41 para determinar a qué grupo de desarrolladores pertenece el usuario y si el usuario tiene permiso para rechazar actualizaciones.

Tabla 41. Información LDAP de Envío a cliente

Nombre de grupo (cn=)	Resultado
FEK.PTC.CONFIG.ENABLED.sysname.devgroup	El cliente acepta actualizaciones de configuración para el grupo especificado
FEK.PTC.PRODUCT.ENABLED.sysname.devgroup	El cliente acepta actualizaciones de producto para el grupo especificado
FEK.PTC.REJECT.CONFIG.UPDATES.sysname	El usuario puede rechazar actualizaciones de configuración cuando el espacio de trabajo está enlazado al grupo predeterminado
FEK.PTC.REJECT.CONFIG.UPDATES.sysname.devgroup	El usuario puede rechazar actualizaciones de configuración cuando el espacio de trabajo está enlazado al grupo especificado
FEK.PTC.REJECT.PRODUCT.UPDATES.sysname	El usuario puede rechazar actualizaciones de producto cuando el espacio de trabajo está enlazado al grupo predeterminado
FEK.PTC.REJECT.PRODUCT.UPDATES.sysname.devgroup	El usuario puede rechazar actualizaciones de producto cuando el espacio de trabajo está enlazado al grupo especificado

El valor de `devgroup` coincide con el nombre de grupo asignado a un grupo específico de desarrolladores. Tenga en cuenta que el nombre de grupo es visible en clientes de Developer for System z.

El valor de `sysname` coincide con el nombre de sistema del sistema de destino.

Un usuario puede seleccionar enlazar un espacio de trabajo al grupo predeterminado para actualizaciones de configuración si `config.enabled` en `pushtoclient.properties` está establecido en SAF o LDAP. Si `config.enabled` está establecido en `TRUE`, el espacio de trabajo se enlaza automáticamente al grupo predeterminado.

Un usuario puede seleccionar enlazar un espacio de trabajo al grupo predeterminado para actualizaciones de producto si `product.enabled` en `pushtoclient.properties` está establecido en SAF o LDAP. Si `product.enabled` está establecido en `TRUE`, el espacio de trabajo está enlazado automáticamente al grupo predeterminado.

El soporte de grupo para las directivas `reject.*.updates` es nuevo en la versión 9.1.0 y cambia el modo en que se procesan las palabras clave LDAP y SAF.

Esquema de LDAP

El esquema de LDAP debe cumplir las reglas siguientes:

1. Cada grupo de Envío al cliente debes estar definido como grupo en el esquema.
2. Cada usuario debe estar definido como usuario en el esquema.
3. Una entrada de grupo tiene las referencias a entradas de usuario que pertenecen a su propio grupo.

La Figura 32 es un ejemplo de definición de LDAP para un grupo y usuario, expresados en formato LDIF.

Nota: El formato de intercambio de datos LDAP (LDIF) es un formato de texto estándar para la representación de objetos LDAP y actualizaciones LDAP. Los archivos que contienen registros LDIF se utilizan para transferir datos entre los servidores de directorios o como entrada por los programas de utilidad de LDAP.

```
# Definición de grupo
dn: cn=FEK.PTC.CONFIG.ENABLED.CDFMVS08.GROUPA,o=PTC,c=DeveloperForZ
objectClass: groupOfUniqueNames
objectClass: top
cn: FEK.PTC.CONFIG.ENABLED.CDFMVS08.GROUPA
description: Project A
uniqueMember: uid=mborn,ou=Users,dc=example,dc=com

# Definición de usuario
dn: uid=mborn,ou=Users,dc=example,dc=com
objectClass: organizationalPerson
objectClass: person
objectClass: inetOrgPerson
objectClass: uidObject
objectClass: top
cn: May Born
sn: Born
uid: mborn
facsimiletelephonenumber: +1 800 982 6883
givenname: May
mail: mborn@example.com
ou: Users
```

Figura 32. Definición de esquema LDAP de ejemplo

Selección del servidor LDAP

Hay disponible una amplia selección de servidores LDAP tanto gratuitos como comerciales. Un ejemplo es IBM Tivoli Directory Server (<http://www-01.ibm.com/software/tivoli/products/directory-server/>). También hay una amplia selección de herramientas basadas en la GUI y en la línea de mandatos para gestionar un servidor LDAP.

Tal y como se ha mencionado en “Esquema de LDAP” en la página 141, los usuarios deben estar definidos en el servidor LDAP. Para reducir el esfuerzo de gestión, lo mejor es colocar el esquema del Envío a cliente en un servidor LDAP que ya tenga acceso a todas las definiciones de usuarios. Por ejemplo, puede utilizar el IBM Tivoli Directory Server activo en z/OS usando una base de datos SDBM (que es una envoltura para su base de datos de seguridad).

Según las políticas del sitio, el esquema del Envío a cliente en el servidor LDAP podría estar gestionado por el administrador de cliente. Esto reduciría las necesidades de colaboración, así como posibles retrasos y errores de comunicación.

Una ventaja de la gestión de LDAP por parte del administrador de cliente es que el esquema del Envío a cliente no tiene nada confidencial ni relacionado con la seguridad. Cuando las definiciones de usuario están disponibles para el servidor LDAP a través de otros esquemas, los objetos LDAP de Developer for System z determinan qué opciones tiene un desarrollador en la selección del diseño del espacio de trabajo y las actualizaciones automáticas de producto del cliente Developer for System z.

Ubicación del servidor LDAP

Cualquier servidor de bases de datos que tenga soporte para el protocolo LDAP se puede utilizar para albergar el esquema de Envío a cliente de Developer for System z. Por lo tanto, Developer for System z le permite especificar la información necesaria para conectar al servidor LDAP. También puede especificar un sufijo que haga que la base de datos sea exclusiva dentro del servidor LDAP.

Directiva rsed.envvars	Valor predeterminado
_RSE_LDAP_SERVER	Sistema de hosts locales
_RSE_LDAP_PORT	389
_RSE_LDAP_PTC_GROUP_SUFFIX	"O=PTC,C=DeveloperForZ"

Tenga en cuenta que las medidas de seguridad de TCP/IP, como cortafuegos podrían impedir la conexión del servidor RSE (basado en host) al servidor LDAP. Póngase en contacto con el administrador de TCP/IP y proporcione la información siguiente para asegurarse de que se pueda establecer contacto con el servidor LDAP:

- Nombre DNS o dirección TCP/IP del servidor LDAP
- Número de puerto del servidor LDAP
- LDAP utiliza el protocolo TCP
- El servidor RSE basado en host se pone en contacto con el servidor LDAP
- El servidor RSE está activo en un espacio de direcciones de RSEDx, donde RSED es el nombre de tarea iniciada RSE y x es un número aleatorio de un dígito

Configuración de ejemplo

Supongamos que una empresa tiene Developer for System z activo en el sistema CDFMVS08. IBM Tivoli Directory Server, también activo en CDFMVS08, se usa como servidor LDAP. El servidor LDAP está configurado según se describe en “Adición del extremo de Envío a cliente a LDAP”.

Los usuarios siguientes utilizan Developer for System z:

- Desarrolladores que trabajan en aplicaciones bancarias, con ID de usuario BNK010 -> BNK014
- Desarrolladores que trabajan en aplicaciones de seguros, con ID de usuario INS010 -> INS014
- Un administrador del cliente Developer for System z, con ID de usuario RDZADM1

Cada grupo de desarrolladores precisa de los archivos de configuración de cliente específicos, y todos los desarrolladores están sujetos al mismo control de versiones del cliente. Al contrario que los administradores del cliente, los desarrolladores no tienen permiso para rechazar ninguno de los cambios que presente Envío a cliente.

El administrador de cliente y el administrador de LDAP acuerdan usar los nombres de grupo BANKING e INSURANCE para las actualizaciones de configuración.

Adición del extremo de Envío a cliente a LDAP

En este ejemplo, las actualizaciones se realizan al IBM Tivoli Directory Server en z/OS, que actualmente utiliza sólo una base de datos SDBM (envoltura de base de datos de seguridad) añadiendo una base de datos LDBM (archivos z/OS UNIX) para albergar el esquema del Envío a cliente.

1. Añadir la sección del extremo LDBM al archivo de configuración LDAP.

```
# nombre de archivo ds.conf
# reiniciar la tarea GLDSRV iniciada para activar los cambios

# global section
adminDN "cn=LDAP admin"
adminPW password
listen ldap://:389
schemaPath /etc/ldap

# SDBM back-end section (RACF)
database SDBM GLDBSD31/GLDBSD64
suffix "cn=RACF,o=IBM,c=US"

# LDBM back-end section (z/OS UNIX files)
database LDBM GLDBLD31/GLDBLD64 LDBM-RDZ
suffix "o=PTC,c=DeveloperForZ"
databaseDirectory /var/ldap/ldbm/rdz
```

2. Detenga e inicie la tarea LDAP iniciada, GRDSRV, para activar los cambios de configuración.
3. Cree el directorio /var/ldap/ldbm/rdz.
mkdir -p /var/ldap/ldbm/rdz
4. Actualice el esquema LDAP para añadir el extremo LDBM.

```
ldapmodify -D "cn=LDAP admin" -w password -f
/usr/lpp/ldap/etc/schema.user.ldif

ldapmodify -D "cn=LDAP admin" -w password -f
/usr/lpp/ldap/etc/schema.IBM.ldif
```

5. Añadir la entrada root al extremo LDBM.

```
ldapadd -D "cn=LDAP admin" -w password -f
/u/ibmuser/ptc_root.ldif
```

donde /u/ibmuser/ptc_root.ldif tiene lo siguiente:

```
dn: o=PTC,c=DeveloperForZ
objectclass: top
objectclass: organization
o: PTC
```

Configuración de grupo LDAP inicial

Añada los distintos objetos de grupo LDAP al esquema, y haga que el administrador de cliente sea parte de cada uno de ellos. La definición de usuario para el ID de usuario RDZADM1 se obtiene del esquema RACF.

```
ldapadd -D "cn=LDAP admin" -w password -f /u/ibmuser/ptc_setup.ldif
```

donde /u/ibmuser/ptc_setup.ldif tiene lo siguiente:

```
# configuración de estación de trabajo de bancos
dn: cn=FEK.PTC.CONFIG.ENABLED.CDFMVS08.BANKING,o=PTC,c=DeveloperForZ
objectclass: groupOfUniqueNames
cn: FEK.PTC.CONFIG.ENABLED.CDFMVS08.BANKING
description: Developer for System z push-to-client
# proporcionar acceso de administrador al cliente
uniqueMember: racfID=RDZADM1,profileType=user,cn=RACF,o=IBM,c=US

# configuración de estación de trabajo de seguros
dn: cn=FEK.PTC.CONFIG.ENABLED.CDFMVS08.INSURANCE,o=PTC,c=DeveloperForZ
objectclass: groupOfUniqueNames
cn: FEK.PTC.CONFIG.ENABLED.CDFMVS08.INSURANCE
description: Developer for System z push-to-client
# proporcionar acceso de administrador al cliente
uniqueMember: racfID=RDZADM1,profileType=user,cn=RACF,o=IBM,c=US

# rechazar actualizaciones de configuración
dn: cn=FEK.PTC.REJECT.CONFIG.UPDATES.CDFMVS08,o=PTC,c=DeveloperForZ
objectclass: groupOfUniqueNames
cn: FEK.PTC.REJECT.CONFIG.UPDATES.CDFMVS08
description: Developer for System z push-to-client
# proporcionar acceso de administrador al cliente
uniqueMember: racfID=RDZADM1,profileType=user,cn=RACF,o=IBM,c=US

# rechazar actualizaciones de producto
dn: cn=FEK.PTC.REJECT.PRODUCT.UPDATES.CDFMVS08,o=PTC,c=DeveloperForZ
objectclass: groupOfUniqueNames
cn: FEK.PTC.REJECT.PRODUCT.UPDATES.CDFMVS08
description: Developer for System z push-to-client
# proporcionar acceso de administrador al cliente
uniqueMember: racfID=RDZADM1,profileType=user,cn=RACF,o=IBM,c=US
```

Añadir desarrolladores a grupos LDAP

Añadir los desarrolladores a los objetos del grupo LDAP. Las definiciones de usuario para los ID de usuario se toman del esquema de RACF.

```
ldapmodify -D "cn=LDAP admin" -w password -f /u/ibmuser/ptc_add.ldif
```

donde /u/ibmuser/ptc_add.ldif tiene lo siguiente:

```
# configuración de estación de trabajo de bancos
dn: cn=FEK.PTC.CONFIG.ENABLED.CDFMVS08.BANKING,o=PTC,c=DeveloperForZ
changeType: modify
add: uniqueMember
uniqueMember: racfID=BNK010,profileType=user,cn=RACF,o=IBM,c=US
uniqueMember: racfID=BNK011,profileType=user,cn=RACF,o=IBM,c=US
uniqueMember: racfID=BNK012,profileType=user,cn=RACF,o=IBM,c=US
```

```
uniqueMember: racfID=BNK013,profileType=user,cn=RACF,o=IBM,c=US
uniqueMember: racfID=BNK014,profileType=user,cn=RACF,o=IBM,c=US

# configuración de estación de trabajo de seguros
dn: cn=FEK.PTC.CONFIG.ENABLED.CDFMVS08.INSURANCE,o=PTC,c=DeveloperForZ
changeType: modify
add: uniqueMember
uniqueMember: racfID=INS010,profileType=user,cn=RACF,o=IBM,c=US
uniqueMember: racfID=INS011,profileType=user,cn=RACF,o=IBM,c=US
uniqueMember: racfID=INS012,profileType=user,cn=RACF,o=IBM,c=US
uniqueMember: racfID=INS013,profileType=user,cn=RACF,o=IBM,c=US
uniqueMember: racfID=INS014,profileType=user,cn=RACF,o=IBM,c=US
```

pushtoclient.properties

```
# BANKING e INSURANCE tienen distintas necesidades de configuración
config.enabled=LDAP
# todos reciben actualizaciones del producto
product.enabled=TRUE
# sólo RDZADMIN puede rechazar actualizaciones de configuración
reject.config.updates=LDAP
# sólo RDZADMIN puede rechazar actualizaciones del producto
reject.product.updates=LDAP
```

rsed.envvars

No hacen falta actualizaciones porque se han usado los valores predeterminados:

- `_RSE_LDAP_SERVER=CDFMVS08.RALEIGH.IBM.COM`
- `_RSE_LDAP_PORT=389`
- `_RSE_LDAP_PTC_GROUP_SUFFIX="o=PTC,c=DeveloperForZ"`

/var/rdz/pushtoclient/*install

Cuando se exporta la configuración de estación de trabajo para los grupos BANKING e INSURANCE, el asistente de exportación crea los directorios `/var/rdz/pushtoclient/grouping/<devgroup>/`, y la estructura de directorios subsiguiente.

- `/var/rdz/pushtoclient/grouping/BANKING/*`
- `/var/rdz/pushtoclient/grouping/INSURANCE/*`

Como no hay escenarios de actualización de productos individualizados, el administrador del cliente no necesita crear o actualizar los subdirectorios `install/` e `install/responsefiles/` en `/var/rdz/pushtoclient/grouping/<devgroup>/`.

El administrador del cliente debe crear los archivos de respuestas necesarios para las actualizaciones del producto en el directorio de grupo predeterminado, `/var/rdz/pushtoclient/install/responsefiles/`.

Selección de grupo basada en SAF

SAF (Servicio de acceso a seguridad, por su siglas en inglés) es una interfaz para acceder a cualquier producto de seguridad de z/OS. Developer for System z puede utilizar esta interfaz para consultar su producto de seguridad y recuperar la información relacionada de Envío a cliente.

Cuando utilice definiciones en su base de datos de seguridad como mecanismo de selección (el valor de SAF se especifica para las directivas en `pushtoclient.properties`), Developer for System z comprueba los permisos de acceso a los perfiles listados en la Tabla 42 en la página 146 para determinar a qué grupo de desarrolladores pertenece el usuario y si el usuario tiene permiso para rechazar actualizaciones.

Tabla 42. Información SAF de Envío a cliente

Perfil FACILITY	Longitud fija	Acceso necesario	Resultado
FEK.PTC.CONFIG.ENABLED. sysname.devgroup	23	READ	El cliente acepta actualizaciones de configuración para el grupo especificado
FEK.PTC.PRODUCT.ENABLED. sysname.devgroup	24	READ	El cliente acepta actualizaciones de producto para el grupo especificado
FEK.PTC.REJECT.CONFIG. UPDATES.sysname	30	READ	El usuario puede rechazar actualizaciones de configuración cuando el espacio de trabajo está enlazado al grupo predeterminado
FEK.PTC.REJECT.CONFIG. UPDATES.sysname.devgroup	30	READ	El usuario puede rechazar actualizaciones de configuración cuando el espacio de trabajo está enlazado al grupo especificado
FEK.PTC.REJECT.PRODUCT. UPDATES.sysname	31	READ	El usuario puede rechazar actualizaciones de producto cuando el espacio de trabajo está enlazado al grupo predeterminado
FEK.PTC.REJECT.PRODUCT. UPDATES.sysname.devgroup	31	READ	El usuario puede rechazar actualizaciones de producto cuando el espacio de trabajo está enlazado al grupo especificado

Nota: Developer for System z presupone que un usuario no tiene autorización de acceso cuando su software de seguridad indique que no puede determinar si un usuario tiene o no autorización de acceso a un perfil. Un ejemplo sería cuando el perfil no está definido.

El valor de `devgroup` coincide con el nombre de grupo asignado a un grupo específico de desarrolladores. Tenga en cuenta que el nombre de grupo es visible en clientes de Developer for System z.

El valor de `sysname` coincide con el nombre de sistema del sistema de destino.

Un usuario puede seleccionar enlazar un espacio de trabajo al grupo predeterminado para actualizaciones de configuración si `config.enabled` en `pushtoclient.properties` está establecido en SAF o LDAP. Si `config.enabled` está establecido en TRUE, el espacio de trabajo se enlaza automáticamente al grupo predeterminado.

Un usuario puede seleccionar enlazar un espacio de trabajo al grupo predeterminado para actualizaciones de producto si `product.enabled` en `pushtoclient.properties` está establecido en SAF o LDAP. Si `product.enabled` está establecido en TRUE, el espacio de trabajo se enlaza automáticamente al grupo predeterminado.

La columna "Longitud fija" indica la longitud de la parte fija del perfil de seguridad relacionado.

De forma predeterminada, Developer for System z espera que los perfiles FEK.* estén en la clase de seguridad FACILITY. Tenga en cuenta que los perfiles en la clase FACILITY tienen 39 caracteres como máximo. Si la suma de la longitud de la parte del perfil fija (FEK.PTC.<key>.) y la longitud de la parte del perfil específica del sitio (`sysname` o `sysname.devgroup`) sobrepasa este número, puede colocar los perfiles en otra clase e indicar a Developer for System z que utilice esta clase en su lugar. Para ello, active `_RSE_FEK_SAF_CLASS` en `rsed.envvars` y proporcione el nombre de clase que quiera.

Configuración de ejemplo

Supongamos que una empresa tiene Developer for System z activo en el sistema CDFMVS08. La base de datos de seguridad de RACF está compartida entre varios sistemas y los grupos siguientes están definidos en la base de datos de seguridad.

- DEVBANK : desarrolladores que trabaja en aplicaciones bancarias
- DEVINSUR : desarrolladores que trabajan aplicaciones de seguros
- RDZADMIN : administradores del cliente Developer for System z

Cada grupo de desarrolladores precisa de los archivos de configuración de cliente específicos, y todos los desarrolladores están sujetos al mismo control de versiones del cliente. Al contrario que los administradores del cliente, los desarrolladores no tienen permiso para rechazar ninguno de los cambios que presente Envío a cliente. La regla de rechazo es válida para todos los sistemas, en preparación para ampliación futura.

El cliente y administrador de seguridad acuerdan usar los nombres de grupo de Envío a cliente BANKING e INSURANCE para las actualizaciones de configuración.

Definición de seguridad

Los perfiles están definidos en la clase XFACILIT porque el nombre de perfil más largo, FEK.PTC.REJECT.PRODUCT.UPDATES.CDFMVS08.DEVINSUR, tiene 48 caracteres de longitud, que son más de 39 caracteres soportados por la clase FACILITY.

```
# permitir que RDZADMIN y DEVBANK seleccionen el grupo de Envío a cliente BANKING
RDEFINE XFACILIT (FEK.PTC.CONFIG.ENABLED.CDFMVS08.BANKING) -
  UACC(NONE) DATA('RATIONAL DEVELOPER FOR SYSTEM Z - PUSH-TO-CLIENT')
PERMIT FEK.PTC.CONFIG.ENABLED.CDFMVS08.BANKING CLASS(XFACILIT) -
  ID(RDZADMIN DEVBANK) ACCESS(READ)

# permitir que RDZADMIN y DEVINSUR seleccionen el grupo de Envío a cliente INSURANCE
RDEFINE XFACILIT (FEK.PTC.CONFIG.ENABLED.CDFMVS08.INSURANCE) -
  UACC(NONE) DATA('RATIONAL DEVELOPER FOR SYSTEM Z - PUSH-TO-CLIENT')
PERMIT FEK.PTC.CONFIG.ENABLED.CDFMVS08.INSURANCE CLASS(XFACILIT) -
  ID(RDZADMIN DEVINSUR) ACCESS(READ)

# RDZADMIN puede rechazar actualizaciones de configuración en cualquier
sistema y para cualquier grupo
RDEFINE XFACILIT (FEK.PTC.REJECT.CONFIG.UPDATES.***) -
  UACC(NONE) DATA('RATIONAL DEVELOPER FOR SYSTEM Z - PUSH-TO-CLIENT')
PERMIT FEK.PTC.REJECT.CONFIG.UPDATES.** CLASS(XFACILIT) -
  ID(RDZADMIN) ACCESS(READ)

# RDZADMIN puede rechazar actualizaciones de producto en cualquier
sistema y para cualquier grupo
RDEFINE XFACILIT (FEK.PTC.REJECT.PRODUCT.UPDATES.***) -
  UACC(NONE) DATA('RATIONAL DEVELOPER FOR SYSTEM Z - PUSH-TO-CLIENT')
PERMIT FEK.PTC.REJECT.PRODUCT.UPDATES.** CLASS(XFACILIT) -
  ID(RDZADMIN) ACCESS(READ)

# activar cambios
SETROPTS RACLIST(XFACILIT) REFRESH
```

pushtoclient.properties

```
# BANKING e INSURANCE tienen distintas necesidades de configuración
config.enabled=SAF
# todos reciben actualizaciones del producto
product.enabled=TRUE
# sólo RDZADMIN puede rechazar actualizaciones de configuración
reject.config.updates=SAF
# sólo RDZADMIN puede rechazar actualizaciones del producto
reject.product.updates=SAF
```

rsed.envvars

```
_RSE_FEK_SAF_CLASS=XFACILIT
```

/var/rdz/pushtoclient/*install

Cuando se exporta la configuración de estación de trabajo para los grupos BANKING e INSURANCE, el asistente de exportación crea los directorios /var/rdz/pushtoclient/grouping/<devgroup>/, y la estructura de directorios subsiguiente.

- /var/rdz/pushtoclient/grouping/BANKING/*
- /var/rdz/pushtoclient/grouping/INSURANCE/*

Como no hay escenarios de actualización de productos individualizados, el administrador del cliente no necesita crear o actualizar los subdirectorios install/ e install/responsefiles/ en /var/rdz/pushtoclient/grouping/<devgroup>/.

El administrador del cliente debe crear los archivos de respuestas necesarios para las actualizaciones del producto en el directorio de grupo predeterminado, /var/rdz/pushtoclient/install/responsefiles/.

Periodo de gracia para el rechazo de cambios

Supongamos que mientras la configuración de ejemplo está activa, hay disponible un Developer for System zfix-pack con mejoras importantes, pero la planificación del proyecto bancario es tal que los distintos desarrolladores podrían estar muy preocupados sobre cambios en sus estaciones de trabajo en ese preciso momento del proyecto.

Para resolver el problema, el administrador de seguridad puede proporcionar a todos los desarrolladores de DEVBANK un periodo de gracia en el que pueden elegir si retrasar (rechazar) la actualización.

La configuración de un periodo de gracia es un proceso muy sencillo:

```
# inicio del periodo de gracia
PERMIT FEK.PTC.REJECT.PRODUCT.UPDATES.** CLASS(XFACILIT) -
    ID(DEVBANK) ACCESS(READ)

# activar cambios
SETROPTS RACLIST(FACILITY) REFRESH
```

Al final del periodo de gracia, la autoridad adicional se puede eliminar de nuevo:

```
# fin del periodo de gracia
PERMIT FEK.PTC.REJECT.PRODUCT.UPDATES.** CLASS(XFACILIT) -
    ID(DEVBANK) DELETE

# activar cambios
SETROPTS RACLIST(FACILITY) REFRESH
```

Nota: El administrador de seguridad podría haber creado también un perfil FEK.PTC.REJECT.PRODUCT.UPDATES.*.DEVBANK con UACC(READ). Esto permitiría a todos los desarrolladores que enlazan su espacio trabajo al grupo DEVBANK rechazar actualizaciones de producto. El permiso de rechazo no está otorgado a desarrolladores que enlazan su espacio de trabajo al grupo predeterminado, aunque sean un miembro del grupo DEVBANK, puesto que esto está controlado por el perfil FEK.PTC.REJECT.PRODUCT.UPDATES.*.

Proyectos basados en host

Los proyectos de z/OS se pueden definir individualmente a través de la perspectiva Proyectos de z/OS en el cliente; los proyectos de z/OS también se pueden definir de forma centralizada en el host y luego propagarse al cliente en base a usuarios individuales. Estos "proyectos basados en host" tienen el mismo aspecto y funcionan exactamente igual que proyectos definidos en el cliente, salvo porque el cliente no puede modificar la estructura, miembros y propiedades, y sólo se puede acceder a ellos cuando se está conectado al host.

El directorio base para proyectos basados en host está definido (por el administrador del cliente) en /var/rdz/pushtoclient/keymapping.xml, y es /var/rdz/pushtoclient/projects de forma predeterminada.

Para configurar proyectos basados en host, el gestor de proyectos o desarrollador encargado necesita definir los tipos de archivos de configuración siguientes. Todos los archivos son archivos XML codificados en UTF-8.

- Los archivos de instancia de proyecto son específicos para un único ID de usuario, y hacen referencia a archivos de definición de proyectos reutilizables. Cada usuario que trabaja con proyectos basados en host necesita un

subdirectorio, `/var/rdz/pushtoclient/projects/<IDusuario>/`, que contiene un archivo de instancia de proyecto (*.hbpin) para cada proyecto a descargar.

- Los archivos de definición de proyectos definen la estructura y contenido del proyecto, y pueden ser reutilizados por varios usuarios. Los archivos de definición de proyectos (*.hbppd) muestran una lista de los subproyectos que se encuentran en el proyecto y están ubicados en el directorio de la definición del proyecto raíz o uno de sus subdirectorios.
- Los archivos de definición de subproyecto definen la estructura y contenidos del subproyecto y pueden ser reutilizados por varios usuarios. Los archivos de definición de subproyecto (*.hbpsd) definen el conjunto de recursos necesarios para construir un único módulo de carga y se encuentran en el directorio de definición de proyecto raíz o en uno de sus subdirectorios.
- Los archivos de propiedades de subproyecto son archivos de propiedades con soporte para sustitución de variable y que varios subproyectos pueden reutilizar. Los archivos de propiedades de subproyecto (*.hbppr) tienen soporte para la sustitución de variable, para permitir la compartición de los archivos de propiedades entre varios usuarios, y se encuentran en el directorio de definición de proyecto raíz o en uno de sus subdirectorios.

Los proyectos basados en host también se pueden elegir para participar en la configuración de grupo múltiple tratada en “Varios grupos de desarrollador” en la página 135. Esta capacidad de elección quiere decir que los proyectos basados en host también se pueden definir en `/var/rdz/pushtoclient/grouping/<devgroup>/projects/`.

Cuando un espacio de trabajo está enlazado a un grupo específico, y hay definiciones de proyecto para un usuario en este grupo y en el grupo predeterminado, el usuario recibe las definiciones de proyectos tanto desde el grupo predeterminado como desde el grupo específico.

Capítulo 8. Consideraciones de CICSTS

Tradicionalmente, el papel de definir recursos en CICS ha sido competencia del administrador de CICS. Ha habido cierta renuencia en permitir que el desarrollador de aplicaciones definiera recursos CICS por diversas razones:

- La mayoría de las definiciones de recurso CICS tienen muchos parámetros que, debido a su complejidad, interrelación con otras definiciones de recurso y estándares comerciales, requieren conocimientos de administración de CICS para definirlos correctamente. Las definiciones incorrectas pueden provocar resultados inesperados que pueden influir sobre toda la región CICS.
- La mayoría de establecimientos comerciales ofrecen entornos de prueba y desarrollo CICS que deben estar disponibles para el uso compartido de varios grupos de aplicaciones y desarrolladores. Muchos establecimientos comerciales cuentan con el Acuerdo de nivel de servicio para estos entornos. El cumplimiento de estos acuerdos requiere un estricto control de los entornos.

Developer for System z soluciona estos problemas permitiendo que los administradores de CICS controlen los valores predeterminados de la definición de recursos de CICS y que controlen las propiedades de visualización de un parámetro de definición de recursos de CICS mediante el servidor Definición de recursos de CICS (CRD) que forma parte del Gestor de despliegue de aplicaciones.

Por ejemplo, el administrador de CICS puede suministrar determinados parámetros de definición de recurso CICS que el desarrollador de aplicaciones no puede actualizar. Otros parámetros de definición de recurso CICS pueden ser actualizables, con o sin valores predeterminados suministrados, o el parámetro de definición de recurso CICS puede ocultarse para evitar una complejidad innecesaria.

Una vez que el desarrollador de aplicaciones está satisfecho con las definiciones de recurso CICS, éstas pueden instalarse de inmediato en el entorno de prueba CICS en ejecución, o pueden exportarse las definiciones en un manifiesto para que un administrador de CICS las edite y apruebe. El administrador de CICS puede utilizar el programa de utilidad administrativo (programa de utilidad de proceso por lotes) o la herramienta de Proceso de manifiestos para implementar cambios de definición de recurso.

Nota: La herramienta de Proceso de manifiestos es un plugin de IBM CICS Explorer.

Consulte el apartado "(Opcional) Gestor de despliegue de aplicaciones" de la *Guía de configuración de host* (SC11-3660) para obtener más información acerca de las tareas necesarias para configurar el Gestor de despliegue de aplicaciones en el sistema host.

La personalización del Gestor de despliegue de aplicaciones añade los siguientes servicios a Developer for System z:

- (en el cliente) CICS Explorer de IBM proporciona una infraestructura basada en Eclipse para poder visualizar y manipular recursos CICS, además de permitir una mayor integración entre las herramientas CICS
- (en el cliente) El editor CRD (definiciones de recursos CICS)

- (en el host) El servidor CDR (definiciones de recursos CICS), que se ejecuta como una aplicación CICS

El servidor CRD (definición de recurso CICS) del Gestor de despliegue de aplicaciones consta del propio servidor CRD, un repositorio CRD, las definiciones de recursos CICS asociadas y, al utilizar la interfaz de servicios Web, archivos de enlace de servicio Web y un manejador de mensajes de conducto de ejemplo. El servidor CRD debe ejecutarse en una WOR (Web Owning Region), a la que se hace referencia en la documentación de Developer for System z como región de conexión primaria CICS.

Consulte el Information Center de Developer for System z (http://www-01.ibm.com/support/knowledgecenter/SSQ2R2_9.1.0/com.ibm.etools.getstart.wsentdev.doc/kc_version_welcome_rdz.html) para obtener más información sobre los servicios del Gestor de despliegue de aplicaciones disponibles en el release actual de Developer for System z.

RESTful versus Servicio Web

CICS Transaction Server proporciona en la versión 4.1 y superior soporte para una interfaz HTTP diseñada utilizando los principios de la transferencia de estado representativo (RESTful). Esta interfaz RESTful es ahora la interfaz CICSTS estratégica para las aplicaciones de cliente. La interfaz de servicio web anterior se ha estabilizado, y las mejoras serán únicamente para la interfaz RESTful.

El Gestor de despliegue de aplicaciones sigue esta sentencia de dirección y necesita el servidor CRD de RESTful para todos los servicios que son nuevos Developer for System versión 7.6 o superior.

Las interfaces RESTful y de servicio Web puede estar activas simultáneamente en una única región CICS, si lo desea. En este caso, habrá dos servidores CRD activos en la región. Ambos servidores compartirán el mismo repositorio CRD. Tenga en cuenta que CICS emitirá algunos avisos sobre definiciones duplicadas cuando se defina la segunda interfaz en la región.

Comparación entre regiones de conexión primarias y no primarias

Un entorno de prueba CICS puede constar de varias regiones MRO (Opción multiregión) conectadas. Con el tiempo, se han utilizado denominaciones no oficiales para categorizar estas regiones. Son designaciones típicas TOR (región propietaria de terminal, WOR (región propietaria Web), AOR (región propietaria de aplicación) y DOR (región propietaria de datos).

Una WOR (región propietaria Web) se utiliza para implementar el soporte de servicios Web de CICS, y el servidor CRD (definición de recurso CICS) del Gestor de despliegue de aplicaciones debe ejecutarse en esta región. Esta región se conoce en el gestor de despliegue de aplicaciones como la región de conexión primaria CICS. El cliente CRD implementa una conexión de servicio Web con la región de conexión primaria CICS.

Las regiones de conexión no primaria CICS son todas las demás regiones a las que el servidor CRD puede dar servicio. Este servicio incluye la visualización de recursos mediante IBM CICS Explorer y la definición de recursos mediante el editor de definiciones de recurso CICS.

Si se utiliza BAS (Business Application Services) de CICSplex SM para gestionar las definiciones de recursos CICS de la región de conexión primaria CICS, el servidor CRD puede dar servicio a todas las demás regiones CICS gestionadas por BAS.

Las regiones CICS no gestionadas por BAS requieren cambios adicionales para que el servidor CRD pueda darles servicio.

Registro de instalación de recursos CICS

Las acciones realizadas por el servidor CRD en los recursos CICS se anotan en la cola TD CSDL de CICS, que generalmente señala hacia a la DD MSGUSR de la región CICS.

Si se utiliza CICSplex SM Business Application Services (BAS) para gestionar sus definiciones de recursos de CICS, la directiva de CICSplex SM EYUPARM BASLOGMSG se debe establecer en (YES) para que se cree la anotación cronológica.

Seguridad del Gestor de despliegue de aplicaciones

Seguridad del repositorio CRD

El conjunto de datos VSAM del repositorio del servidor CRD contiene todas las definiciones de recurso predeterminadas y, por tanto, debe protegerse contra actualizaciones, pero los desarrolladores deben poder leer los valores almacenados en él. Consulte “Definir los perfiles de conjunto de datos” en la página 57 para obtener mandatos RACF de ejemplo para proteger el repositorio de CRD.

Seguridad de conducto

Cuando CICS recibe un mensaje SOAP a través de la interfaz de servicios Web, un conducto lo procesa. Un conducto es un conjunto de manejadores de mensajes que se ejecutan por orden. CICS lee el archivo de configuración del conducto para determinar qué manejadores de mensajes deben invocarse en el conducto. Un manejador de mensajes es un programa en el que pueden realizarse procesos especiales de solicitudes y respuestas de servicios Web.

El Gestor de despliegue de aplicaciones suministra un archivo de configuración de conducto de ejemplo que especifica la invocación de un manejador de mensajes y un programa de proceso de cabeceras SOAP.

El manejador de mensajes de conducto (ADNTMSGH) se utiliza para la seguridad, mediante el proceso del ID de usuario y la contraseña de la cabecera SOAP. El archivo de configuración de conducto de ejemplo hace referencia a ADNTMSGH, que, por lo tanto, se debe colocar en la concatenación RPL de CICS.

Seguridad de transacción

CPIH es el ID de transacción predeterminado bajo el que se ejecutará una aplicación invocada por un conducto. Generalmente, CPIH se establece para un nivel de autorización mínimo.

Developer for System z suministra varias transacciones que el servidor CRD utiliza al definir y consultar recursos CICS. Estos ID de transacción se establecen mediante el servidor CRD, dependiendo de la operación solicitada. Consulte el apartado "(Opcional) Gestor de despliegue de aplicaciones" de la publicación *Guía de configuración de host* (SC11-3660) para obtener más información sobre cómo personalizar los ID de transacción.

Transacción	Descripción
ADMS	Para las solicitudes de la herramienta de Proceso de manifiestos para cambiar recursos CICS. Normalmente, está destinado a los administradores de CICS. Esta transacción requiere un alto nivel de autorización.
ADMI	Para las peticiones que definen, instalan o desinstalan recursos CICS. Esta transacción puede requerir un nivel de autorización medio, dependiendo de las políticas establecidas en la ubicación.
ADMR	Para todas las demás peticiones que recuperan información de recursos o de entorno de CICS. Esta transacción puede requerir un nivel de autorización mínimo, dependiendo de las políticas establecidas en la ubicación.

Algunas de las solicitudes de definiciones de recurso realizadas por las transacciones del servidor CRD, o todas ellas, deben protegerse. Como mínimo, los mandatos de actualización (actualizar parámetros de servicio Web predeterminados, parámetros de descriptor predeterminados y enlace de nombre de archivo a nombre de conjunto de datos) deben protegerse para impedir nadie, excepto los administradores de CICS, emita estos mandatos utilizados para establecer valores predeterminados de recursos globales.

Cuando se conecta la transacción, la comprobación de la seguridad de recursos CICS, si está habilitada, se asegura de que el ID de usuario tiene autorización para ejecutar el ID de transacción.

La comprobación de recursos está controlada por la opción RESSEC en la transacción que se ejecuta, por el parámetro de inicialización del sistema RESSEC y, para el servidor CRD, por el parámetro de inicialización del sistema XPCT.

La comprobación de recursos sólo tiene lugar si el parámetro de inicialización del sistema XPCT tiene un valor que no es NO y la opción RESSEC de la definición de TRANSACTION es YES o el valor del parámetro de inicialización del sistema RESSEC es ALWAYS.

Los siguientes mandatos de RACF ofrecen un ejemplo de cómo pueden protegerse las transacciones del servidor CRD. Consulte la publicación *RACF Security Guide for CICSTS* para obtener más información acerca de la definición de la seguridad de CICS.

- RALTER GCICSTRN SYSADM UACC(NONE) ADDMEM(ADMS)
- PERMIT SYSADM CLASS(GCICSTRN) ID(#admindcics)
- RALTER GCICSTRN DEVELOPER UACC(NONE) ADDMEM(ADMI)
- PERMIT DEVELOPER CLASS(GCICSTRN) ID(#desarrolladorcics)
-

RALTER GCICSTRN ALLUSER UACC(READ) ADDMEM(ADMR)

•

SETROPTS RACLIST(TCICSTRN) REFRESH

Comunicación cifrada con SSL

El cifrado SSL de la secuencia de datos está soportada cuando el cliente Gestor de despliegue de aplicaciones utiliza la interfaz de servicios Web para invocar el servidor CRD. La utilización de SSL para esta comunicación está controlada por la palabra clave SSL(YES) en la definición de CICSTS TCPIP SERVICE, tal como se documenta en *RACF Security Guide for CICSTS*.

Seguridad de recursos

CICSTS proporciona la capacidad de proteger recursos, además de los mandatos para manipularlos. Algunas acciones del Gestor de despliegue de aplicaciones pueden fallar si la seguridad está activada pero no está completamente configurada (por ejemplo, otorgar permisos para la manipulación de tipos de recursos nuevos).

Cuando falle una función en el Gestor de despliegue de aplicaciones, examine los registros de CICS para buscar mensajes similares al siguiente, y realice la acción correctiva, tal como se documenta en *RACF Security Guide for CICSTS*.

```
DFHXS1111 %date %time %applid %tranid Security violation by user
%userid at netname %portname for resource %resource in class
%classname. Los códigos SAF son (X'safresp',X'safreas'). Los códigos ESM son
(X'esmpresp',X'esmpreas').
```

Programa de utilidad administrativa

Developer for System z suministra el programa de utilidad administrativo que permite a los administradores de CICS proporcionar los valores predeterminados para las definiciones de recursos CICS. Estos valores predeterminados pueden ser de sólo lectura o pueden ser editables para los desarrolladores de aplicaciones.

El programa de utilidad administrativo ofrece las funciones siguientes:

- Nombre CICSplex para los entornos de prueba gestionados por CICSplex
- Nombre del grupo intermedio de CICSplex SM
- Valor de la norma de exportación de manifiestos
- Permisos de visualización y valores predeterminados de atributo de recurso CICS
- Enlace lógico a físico CICS utilizado para las definiciones de conjunto de datos VSAM

El programa de utilidad administrativo se invoca mediante el trabajo de ejemplo ADNJSPAU del conjunto de datos FEK.#CUST.JCL. La utilización de este programa de utilidad requiere acceso de actualización (UPDATE) al repositorio de CRD.

ADNJSPAU se encuentra en FEK.#CUST.JCL, a menos que el programador de sistemas z/OS haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP(FEKSETUP). Consulte a sección "Configuración de personalización" de la publicación *Guía de configuración de host* (SC11-3660) para obtener más detalles.

Nota: Antes de ejecutar el trabajo ADNJSPAU, debe cerrarse el repositorio CRD de CICS. El repositorio puede abrirse de nuevo una vez finalizado el trabajo. Por

ejemplo, después de iniciar la sesión en CICS, especifique los mandatos siguientes para cerrar y abrir el archivo, respectivamente:

- CEMT S FILE(ADNREPF0) CLOSED
- CEMT S FILE(ADNREPF0) OPEN

Se utilizan sentencias de control de entrada para actualizar el repositorio del CRD para un entorno de prueba CICS, para las que se aplican las siguientes normas generales de sintaxis:

- Un asterisco en la posición 1 indica una línea de comentario.
- Un mandato DEFINE debe empezar en la posición 1, seguido por un sólo espacio, seguido de una palabra clave válida, como por ejemplo TRANSACTION.
- Una palabra clave debe ir seguida inmediatamente de un valor de palabra clave. No se permiten espacios intercalados. La única excepción la constituyen las palabras clave de permiso de visualización UPDATE, PROTECT y HIDDEN, que no tienen valores.
- Los valores de palabra clave se especifican entre paréntesis.
- Una palabra clave y su valor deben especificarse en una sola línea.

Las siguientes definiciones de ejemplo siguen la estructura de los mandatos DFHCSDUP, tal como se define en *CICS Resource Definition Guide for CICSTS*. La única diferencia es la inserción de las siguientes palabras clave de permiso de visualización utilizadas para agrupar los valores de atributo en tres conjuntos de permisos:

UPDATE	Un desarrollador de aplicaciones podrá actualizar los atributos situados a continuación de esta palabra clave mediante Developer for System z. Este es también el valor predeterminado para los atributos omitidos.
PROTECT	Los atributos situados a continuación de esta palabra clave se visualizarán, pero el desarrollador de aplicaciones no podrá actualizarlos mediante Developer for System z.
HIDDEN	Los atributos situados a continuación de esta palabra clave no se visualizarán, y el desarrollador de aplicaciones no podrá actualizarlos mediante Developer for System z.

Consulte el siguiente ejemplo de código de ADNJSPAU.

```

//ADNJSPAU JOB <PARÁMETROS DEL TRABAJO>
//*
//ADNSPAU EXEC PGM=ADNSPAU,REGION=1M
//STEPLIB DD DISP=SHR,DSN=FEK.SFEKLOAD
//ADMREP DD DISP=OLD,DSN=FEK.#CUST.ADNREPF0
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
*
* Parámetros de CICSplex SM
*
DEFINE CPSMNAME( )
*DEFINE STAGINGGROUPNAME(ADMSTAGE)
*
* Regla de exportación de manifiestos
*
DEFINE MANIFESTEXPORTRULE(installOnly)
*
* Valores predeterminados de definición de recurso CICS
* Los atributos omitidos toman por omisión el valor UPDATE.
*
* Atributos predeterminados de DB2TRAN
*
DEFINE DB2TRAN()
    UPDATE DESCRIPTION()
    ENTRY()
    TRANSID()
*
* Atributos predeterminados de DOCTEMPLATE
*
DEFINE DOCTEMPLATE()
    UPDATE DESCRIPTION()
    TEMPLATENAME()
    FILE() TSQUEUE() TDQUEUE() PROGRAM() EXITPGM()
    DDNAME(DFHHTML) MEMBERNAME()
    HFSFILE()
    APPENDCRLF(YES) TYPE(EBCDIC)
*
* Atributos predeterminados de File
*
DEFINE FILE()
    UPDATE DESCRIPTION()
    RECORDSIZE() KEYLENGTH()
    RECORDFORMAT(V) ADD(NO)
    BROWSE(NO) DELETE(NO) READ(YES) UPDATE(NO)
    REMOTESYSTEM() REMOTENAME()
    PROTECT DSNAME() RLSACCESS(NO) LSRPOOLID(1) STRINGS(1)
    STATUS(ENABLED) OPENTIME(FIRSTREF)
    DISPOSITION(SHARE) DATABUFFERS(2) INDEXBUFFERS(1)
    TABLE(NO) MAXNUMRECS(NOLIMIT)
    READINTEG(UNCOMMITTED) DSNSHARING(ALLREQS)
    UPDATEMODEL(LOCKING) LOAD(NO)
    JNLREAD(NONE) JOURNAL(NO)
    JNLSYNCREAD(NO) JNLUPDATE(NO)
    JNLADD(NONE) JNLSYNCSWRITE(YES)
    RECOVERY(NONE) FWDRECOVLOG(NO)
    BACKUPTYPE(STATIC)
    PASSWORD() NSRGROUP()
    CFDTPOOL() TABLENAME()

```

Figura 33. ADNJSPAU - programa de utilidad administrativo de CICSTS

```

*
*   Atributos predeterminados de Mapset
*
DEFINE MAPSET()
    UPDATE DESCRIPTION()
    PROTECT RESIDENT(NO) STATUS(ENABLED)
        USAGE(NORMAL) USELPACOPY(NO)
** Atributos predeterminados de Processtype
*
DEFINE PROCESSTYPE()
    UPDATE DESCRIPTION()
        FILE(BTS)
    PROTECT STATUS(ENABLED)
        AUDITLOG() AUDITLEVEL(OFF)
*
*   Atributos predeterminados de Program
*
DEFINE PROGRAM()
    UPDATE DESCRIPTION()
        CEDF(YES) LANGUAGE(LE370)
        REMOTESYSTEM() REMOTENAME() TRANSID()
    PROTECT API(CICSAPI) CONCURRENCY(QUASIRENT)
        DATALOCATION(ANY) DYNAMIC(NO)
        EXECKEY(USER) EXECUTIONSET(FULLAPI)
        RELOAD(NO) RESIDENT(NO)
        STATUS(ENABLED) USAGE(NORMAL) USELPACOPY(NO)
        HIDDEN JVM(NO) JVMCLASS() JVMPROFILE(DFHJVMPR)
*
*   Atributos predeterminados de TDQueue
*
DEFINE TDQUEUE()
    UPDATE DESCRIPTION()
        TYPE(INTRA)
* Parámetros extra-partición
        DDNAME() DSNAME()
        REMOTENAME() REMOTESYSTEM() REMOTELength(1)
        RECORDSIZE() BLOCKSIZE(0) RECORDFORMAT(UNDEFINED)
        BLOCKFORMAT() PRINTCONTROL() DISPOSITION(SHR)
* Parámetros intra-partición
        FACILITYID() TRANSID() TRIGERRLEVEL(1)
        USERID()
* Parámetros indirectos
        INDIRECTNAME()
        PROTECT WAIT(YES) WAITACTION(REJECT)
* Parámetros extra-partición
        DATABUFFERS(1)
        SYSOUTCLASS() ERROROPTION(IGNORE)
        OPENTIME(INITIAL) REWIND(LEAVE) TYPEFILE(INPUT)
* Parámetros intra-partición
        ATIFACILITY(TERMINAL) RECOVSTATUS(NO)

```

Figura 34. ADNJSPAU - programa de utilidad administrativa de CICSTS (parte 2 de 3)

```

*
*   Atributos predeterminados de Transaction
*
DEFINE TRANSACTION()
    UPDATE DESCRIPTION()
        PROGRAM()
        TWASIZE(0)
        REMOTESYSTEM() REMOTENAME() LOCALQ(NO)
    PROTECT PARTITIONSET() PROFILE(DFHCICST)
        DYNAMIC(NO) ROUTABLE(NO)
        ISOLATE(YES) STATUS(ENABLED)
        RUNAWAY(SYSTEM) STORAGEECLEAR(NO)
        SHUTDOWN(DISABLED)
        TASKDATAKEY(USER) TASKDATALOC(ANY)
        BREXIT() PRIORITY(1) TRANCLASS(DFHTCL00)
        DTIMOUT(NO) RESTART(NO) SPURGE(NO) TPURGE(NO)
        DUMP(YES) TRACE(YES) CONFDATA(NO)
        OTSTIMEOUT(NO) WAIT(YES) WAITTIME(00,00,00)
        ACTION(BACKOUT) INDOUBT(BACKOUT)
        RESSEC(NO) CMDSEC(NO)
        TRPROF()
        ALIAS() TASKREQ()
        XTRANID() TPNAME() XTPNAME()

*
*   Atributos de URDIMAP
*
DEFINE URIMAP()
    UPDATE USAGE(CLIENT)
        DESCRIPTION()
        PATH(/required/path)
        TCPIPSERVICE()
        TRANSACTION()
        PROGRAM()
    PROTECT ANALYZER(NOANALYZER)
        ATOMSERVICE()
        CERTIFICATE()
        CHARACTERSET()
        CIPHERS()
        CONVERTER()
        HFSFILE()
        HOST(host.mycompany.com)
        HOSTCODEPAGE()
        LOCATION()
        MEDIATYPE()
        PIPELINE()
        PORT(NO)
        REDIRECTTYPE(NONE)
        SCHEME(HTTP)
        STATUS(ENABLED)
        TEMPLATENAME()
        USERID()
        WEBSERVICE()

*
*   Enlace de nombre de archivo opcional a nombre de conjunto de datos VSAM
*
*DEFINE DSBINDING() DSNAME()
/*

```

Figura 35. ADNJSPAU - Programa de utilidad administrativa de CICSTS (parte 3 de 3)

Notas de migración del programa de utilidad administrativo

Developer for System z versión 7.6.1 añadió soporte de URIMAP al programa de utilidad administrativo. Para poder utilizar el soporte URIMAP, el conjunto de

datos de repositorio CRD VSAM debe estar asignado con un tamaño de registro máximo de 3000. Hasta Developer for System z Versión 7.6.1, el trabajo de asignación de repositorio CRD de ejemplo utiliza un tamaño de registro máximo de 2000.

Siga estos pasos para habilitar el soporte de URIMAP si está utilizando un repositorio CRD más antiguo:

1. Cree una copia de seguridad del repositorio CRD existente, FEK.#CUST.ADNREPF0.
2. Suprima el repositorio CRD existente.
3. Personalice y someta el trabajo FEK.SFEKSAMP(ADNVCRD) para asignar e inicializar un repositorio CRD nuevo. Consulte la documentación del miembro para obtener instrucciones de personalización.
4. Personalice y someta el trabajo FEK.SFEKSAMP(ADNJSPAU) para utilizar el programa de utilidad administrativo para llenar el repositorio CRD nuevo.

Nota:

- No es necesario migrar el repositorio CRD existente porque el programa de utilidad administrativo sustituye todo el contenido del repositorio CRD cada vez que se ejecuta.
- No hay problemas de compatibilidad de versión con el repositorio CRD. Todo el código de cliente y de host de Developer for System z soportado funcionará con el tamaño de registro máximo para cada caso. Pero el soporte de URIMAP estará inhabilitado si el tamaño de registro máximo no es 3000.

Mensajes del programa de utilidad administrativo

El programa de utilidad administrativo emite los mensajes siguientes para la DD SYSPRINT. Los mensajes CRAZ1803E, CRAZ1891E, CRAZ1892E y CRAZ1893E contienen códigos de estado de archivo, retorno VSAM, función VSAM e información de retorno VSAM. Los códigos de retorno, función e información de retorno de VSAM se describen en la publicación *DFSMS Macro Instructions for Data Sets* (SC26-7408). Los códigos de estado de archivo se describen en la publicación *Enterprise COBOL for z/OS Language Reference* (SC27-1408).

CRAZ1800I

se ha completado correctamente en línea <último número de línea de la sentencia de control>

Descripción: El programa de utilidad administrativo del programador del sistema se ha completado satisfactoriamente.

Respuesta del usuario: Ninguna.

CRAZ1801W

se ha completado con avisos en la línea <último número de línea de la sentencia de control>

Descripción: El programa de utilidad administrativo del programador del sistema se ha completado con uno o varios avisos encontrados al procesar las sentencias de control.

Respuesta del usuario: Compruebe si hay otros mensajes de aviso.

CRAZ1802E

se ha detectado un error en la línea <número de línea>

Descripción: El programa de utilidad administrativo del programador del sistema ha encontrado un error grave.

Respuesta del usuario: Compruebe si hay otros mensajes de aviso.

CRAZ1803E

**Error de apertura de repositorio, status=<código de estado de archivo>
RC=<código de retorno de VSAM> FC=<código de función de VSAM>
FB=<códigos de información de retorno de VSAM>**

Descripción: El programa de utilidad administrativo del programador del sistema ha encontrado un error grave al abrir el repositorio de CRD.

Respuesta del usuario: Compruebe los códigos de estado, de retorno, de función y de información de retorno de VSAM.

CRAZ1804E

Registro de entrada no reconocido en la línea <número de línea>

Descripción: El programa de utilidad administrativo del programador del sistema ha encontrado una sentencia de control de entrada no reconocida.

Respuesta del usuario: Compruebe que un mandato **DEFINE** vaya seguido de un solo espacio y de la palabra clave CPSMNAME, STAGINGGROUPNAME, MANIFESTEXPORTRULE, DSBINDING, DB2TRAN, DOCTEMPLATE, FILE, MAPSET, PROCESSTYPE, PROGRAM, TDQUEUE o TRANSACTION.

CRAZ1805E

Procesando la palabra clave <palabra clave> en la línea <número de línea>

Descripción: El programa de utilidad administrativo del programador del sistema está procesando la sentencia de control de entrada de la palabra clave **DEFINE**.

Respuesta del usuario: Ninguna.

CRAZ1806E

Regla de exportación de manifiesto no válida en la línea <número de línea>

Descripción: El programa de utilidad administrativo del programador del sistema ha encontrado una regla de exportación de manifiestos no válida.

Respuesta del usuario: Compruebe que el valor de la palabra clave **MANIFESTEXPORTRULE** sea "installOnly", "exportOnly" o "both".

CRAZ1807E

Falta la palabra clave DSNAME en la línea <número de línea>

Descripción: El programa de utilidad administrativo del programador del sistema estaba procesando una sentencia de control **DEFINE DSBINDING** a la que le falta la palabra clave **DSNAME**.

Respuesta del usuario: Compruebe que la sentencia de control **DEFINE DSBINDING** contenga la palabra clave **DSNAME**.

CRAZ1808E

Valor de palabra clave no válido para la palabra clave <palabra clave> en la línea <número de línea>

Descripción: El programa de utilidad administrativo del programador del sistema estaba procesando una sentencia de control **DEFINE** y ha encontrado un valor no válido para la palabra clave indicada.

Respuesta del usuario: Compruebe que la longitud y el valor de la palabra clave indicada sean correctos.

CRAZ1890W

Error de sintaxis de palabra clave en la línea <número de línea>

Descripción: El programa de utilidad administrativo del programador del sistema estaba procesando una sentencia de control DEFINE y ha encontrado un error de sintaxis en una palabra clave o en un valor de palabra clave.

Respuesta del usuario: Compruebe que el valor de palabra clave se haya especificado entre paréntesis y se encuentre inmediatamente a continuación de la palabra clave. La palabra clave y su valor deben encontrarse en la misma línea.

CRAZ1891W

Error de escritura de clave duplicada de repositorio, status=<código de estado de archivo> RC=<código de retorno de VSAM> FC=<código de función de VSAM> FB=<códigos de información de retorno de VSAM>

Descripción: El programa de utilidad administrativo del programador del sistema ha encontrado un error de clave duplicada al grabar en el repositorio de CRD.

Respuesta del usuario: Compruebe los códigos de estado, de retorno, de función y de información de retorno de VSAM.

CRAZ1892W

Error de grabación de repositorio, status=<código de estado de archivo> RC=<código de retorno de VSAM> FC=<código de función de VSAM> FB=<códigos de información de retorno de VSAM>

Descripción: El programa de utilidad administrativo del programador del sistema ha encontrado un error grave al grabar en el repositorio de CRD.

Respuesta del usuario: Compruebe los códigos de estado, de retorno, de función y de información de retorno de VSAM.

CRAZ1893W

Error de lectura de repositorio, status=<código de estado de archivo> RC=<código de retorno de VSAM> FC=<código de función de VSAM> FB=<códigos de información de retorno de VSAM>

Descripción: El programa de utilidad administrativo del programador del sistema ha encontrado un error grave al leer el repositorio de CRD.

Respuesta del usuario: Compruebe los códigos de estado, de retorno, de función y de información de retorno de VSAM.

Depuración de transacción CICS

Para depurar transacciones CICS, el depurador integrado requiere las actualizaciones de CICS siguientes:

- Actualizaciones de los parámetros de inicialización del sistema CICS (SIT)
 - Especifique DEBUGT00L=YES.
 - Especifique TCP/IP=YES.
 - Especifique LLACOPY=YES si depende de LINKLIST para captar un módulo de carga desde la concatenación DFHRPL DD.
 - Especifique RENTPGM=NOPROTECT si no permite a los usuarios utilizar la SVC del depurador integrado (que es necesaria para depurar las transacciones cargadas en la memoria de solo lectura).
- Actualizaciones JCL de CICS:
 - Especifique REGION=0M en la sentencia EXEC de la región.

- Defina la biblioteca de carga FEK.SFEKAUTH en la sentencia DFHRPL DD de la región. Si se especifica el parámetro SIT LLACOPY=YES, la biblioteca puede encontrarse también en LINKLIST.
- Defina la biblioteca de carga SYS1.MIGLIB en la sentencia DFHRPL DD de la región. Si se especifica el parámetro SIT LLACOPY=YES, la biblioteca puede encontrarse también en LINKLIST.
- Para z/OS 1.13 y superiores, defina la biblioteca de carga SYS1.SIEAMIGE en la sentencia DFHRPL DD de la región. Si se especifica el parámetro SIT LLACOPY=YES, la biblioteca puede encontrarse también en LINKLIST.

Nota:

- El ID de usuario de región CICS necesita permisos UPDATE para el perfil CSVLLA.dataset de la clase FACILITY para que el parámetro SIT LLACOPY=YES funcione correctamente.
- Para depurar programas escritos en COBOL v4, el depurador integrado tiene que acceder a un conjunto de datos de lista (PDS o PDS/E). El nombre de conjunto de datos puede proporcionarse mediante una variable de entorno AQE_DBG_V4LIST, o un DD AQEV4LIST. Si ninguno de los dos elementos está presente, el depurador integrado creará el nombre de conjunto de datos sustituyendo el último calificador del conjunto de datos del ejecutable (por ejemplo .LOAD) por .LISTING. Pregúntele a los desarrolladores qué método puede utilizarse en su caso.
- Actualizaciones CSD de CICS:
Defina el depurador en un región de CICS, tal como se documenta en el trabajo de actualización CSD de ejemplo de AQECSD. AQECSD se encuentra en FEK.#CUST.JCL, a menos que el programador de sistemas z/OS haya especificado otra ubicación al personalizar y enviar el trabajo FEK.SFEKSAMP(FEKSETUP). Para obtener más información, consulte "Configuración de personalización" en la *Guía de configuración del host* (SC11-3660).

Para depurar las transacciones de CICS cargadas en la memoria de solo lectura, el depurador integrado requiere las siguientes actualizaciones de sistema:

- Llamada al supervisor (SVC) de depurador integrado definido en el sistema. Consulte "Cambios de PARMLIB" en la *Guía de configuración del host* (SC11-3660) para obtener más información.
- SVC requiere que los usuarios tengan acceso a un perfil de seguridad si se utiliza en un entorno con estado de problemas (no autorizado). Consulte "Seguridad de depuración" en la página 42 para obtener más detalles.

Nota:

- Sólo puede estar activo un depurador basado en Language Environment (LE) en una región CICS determinada. Una indicación clara de un depurador basado en LE es que proporciona un módulo de carga CEEVDBG o un alias que debe estar disponible para la aplicación.
- El Depurador integrado utiliza CICS CADP para proporcionar opciones de tiempo de ejecución TEST en transacciones CICS. Para obtener más información sobre CADP, consulte la documentación de CICSTS.

Capítulo 9. Consideraciones de salida de usuario

Este capítulo le ayuda a mejorar Developer for System z escribiendo rutinas de salida.

Developer for System z proporciona puntos de salida para seleccionar eventos de Developer for System z. Un punto de salida es un punto específico en el proceso de una función, donde la función invoca una rutina de salida si es que existe una. Puede escribir una rutina de salida para realizar proceso adicional.

Tenga en cuenta que al contrario que en el caso de los puntos de salida tradicionales, los puntos de salida de Developer for System z no permite cambiar el comportamiento de la función. La rutina de salida, si existe, se invoca asíncronamente, una vez completada la función. El proceso de Developer for System z no espera a que la rutina de salida finalice ni comprueba el estado de completitud.

Características de salida de usuario

Activación de la salida de usuario

Las salidas de usuario se activan con las variables `_RSE_JAVAOPTS` `<punto_salida>.action` variables en `rsed.envvars`, donde `<punto_salida>` representa una palabra clave que identifica un punto de salida específico, tal como se indica en “Puntos de salida disponibles” en la página 168.

```
#_RSE_JAVAOPTS="$_RSE_JAVAOPTS -D<punto_salida>.action=<salida_usuario>"
```

De forma predeterminada, todos los puntos de salida están inhabilitados. Quite el comentario y especifique el nombre de vía de acceso completo de la rutina de salida para habilitar el punto de salida.

```
#_RSE_JAVAOPTS="$_RSE_JAVAOPTS -D<punto_salida>.action.id=<IDusuario>"
```

De forma predeterminada, el ID de usuario asignado al daemon RSE se utiliza para ejecutar la rutina de salida proporcionada. Quite el comentario y especifique un ID de usuario para utilizar el ID especificado para ejecutar la salida de usuario. No hay necesidad de especificar una contraseña porque RSE generará un PassTicket para utilizarlo como contraseña cuando conmuta al ID de usuario especificado.

Escritura de una rutina de salida de usuario

Las rutinas de salida de usuario se invocan como mandato shell de z/OS UNIX con uno o varios argumentos. Esto implica que la rutina de salida que desarrolla debe poder ejecutarse desde la línea de mandatos de z/OS UNIX. Las técnicas de codificación comunes incluyen el script shell de z/OS UNIX y el exec REXX de z/OS UNIX, pero también es posible utilizar código compilado, como por ejemplo C/C++.

Consulte la guía *UNIX System Services User's Guide* (SA22-7801) para obtener más información sobre los scripts de shell de z/OS UNIX. Consulte *Using REXX and*

z/OS UNIX System Services (SA22-7806) para obtener más información sobre las extensiones específicas de UNIX de z/OS para el lenguaje REXX.

La rutina de salida la ejecutará un ID de usuario con permisos especiales (como por ejemplo el ID de usuario de tarea iniciada RSE que puede generar PassTickets). Es por lo tanto importante que limite la autoridad de actualización a la rutina de salida para evitar abusos. Los mandatos de ejemplo de z/OS UNIX siguientes limitan la autorización de escritura al propietario mientras que cualquiera puede leer y ejecutar el script.

```
$ chmod 755 process_logon.sh
$ ls -l process_logon.sh
-rwxr-xr-x  1 IBMUSER  SYS1          2228 Feb 28 23:44 process_logon.sh
```

Las definiciones de `rsed.envvars` están disponibles en la rutina de salida de usuario como variables de entorno.

RSE invoca la rutina de salida de usuario con una sola serie de argumento. La serie de argumento puede ser un solo valor o una sola serie que contiene varias palabras claves y valores delimitados. Consulte “Puntos de salida disponibles” en la página 168 para obtener más detalles.

Mensajes de consola

Developer for System z utiliza el ID de mensaje de consola FEK910I para visualizar datos relacionados con las salidas de usuario.

La invocación de la rutina de salida está marcada con el mensaje de consola siguiente:

```
FEK910I
<EXIT_POINT> EXIT: invoking <punto_salida> processing exit
in thread <ID_hebra>
```

Todos los datos escritos en stdout (mandato **echo** en un script de shell, mandato **say** en un exec REXX) se enviarán a la consola:

```
FEK910I <EXIT_POINT> EXIT: <mensaje>
```

La terminación de la rutina de salida está marcada con el mensaje de consola siguiente:

```
FEK910I <EXIT_POINT> EXIT: completed <punto_salida> processing exit
in thread <ID_hebra>
```

Ejecución con un ID de usuario variable

Developer for System z permite ejecutar una rutina de salida con el ID de usuario de tarea o un ID de usuario especificado. Sin embargo, es posible que ejecute algunas acciones en la rutina de salida mediante otro ID de usuario, como por ejemplo el ID de usuario de cliente en la rutina de salida de inicio de sesión. Esto se puede conseguir mediante los servicios de z/OS UNIX estándar, tal como se muestra en los ejemplos siguientes.

Script de shell de z/OS UNIX

Tal como se indica en *UNIX System Services Command Reference* (SA22-7802), z/OS UNIX ofrece el mandato **su** para utilizar los privilegios de un superusuario o de otro usuario. Hay unas pocas cosas que se deben tener en cuenta cuando se utiliza el mandato **su**.

- El ID de usuario que ejecuta el mandato **su** debe tener permiso READ sobre el perfil BPX.SRV.<IDusuario>, en la clase SURROGAT del producto de seguridad para poder conmutar al ID de usuario identificado por <IDusuario> sin especificar una contraseña.
- El mandato **su** inicia un shell nuevo de modo que los mandatos restantes del script de shell no se ejecutarán hasta que salga el shell iniciado por el mandato **su**. Para organizar la ejecución de mandatos en el shell nuevo iniciado por el mandato **su**, puede utilizar el mandato **echo** para crear el mandato deseado y el mandato carácter de mandato pipe para incluirlo en el shell nuevo, tal como se muestra en el ejemplo siguiente. Tenga en cuenta que las reglas de creación de scripts de shell estándar se aplican para la evasión de caracteres especiales.

```
#!/bin/sh
myID=ibmuser
echo a $(id)
echo 'echo b $(id)' | su -s $myID
echo "echo c \$(id)" | su -s $myID
cat /u/ibmuser/iefbr14
echo "submit /u/ibmuser/iefbr14" | su -s $myID
```

Esta salida de inicio de sesión de ejemplo, ejecutada por el ID de usuario de tarea iniciada, generará los mensajes de consola siguiente:

```
+FEK910I LOGON EXIT: invoking logon processing exit in thread 411
+FEK910I LOGON EXIT: a uid=8(STCRSE) gid=1(STCGRP)
+FEK910I LOGON EXIT: b uid=1(IBMUSER) gid=0(SYS1)
+FEK910I LOGON EXIT: c uid=1(IBMUSER) gid=0(SYS1)
+FEK910I LOGON EXIT: //IEFBR14 JOB CLASS=A,MSGCLASS=A,MSGLEVEL=(1,1)
+FEK910I LOGON EXIT: //IEFBR14 EXEC PGM=IEFBR14
$HASP100 IEFBR14 ON INTRDR FROM STC03919
IBMUSER
IRR010I USERID IBMUSER IS ASSIGNED TO THIS JOB.
+FEK910I LOGON EXIT: JOB JOB03926 submitted from path '/u/ibmuser/iefbr14'
ICH70001I IBMUSER LAST ACCESS AT 00:46:13 ON MONDAY, MARCH 19, 2012
$HASP373 IEFBR14 STARTED - INIT 2 - CLASS A - SYS CD08
IEF403I IEFBR14 - STARTED - TIME=00.46.14
+FEK910I LOGON EXIT: completed logon processing exit in thread 411
IEFBR14 IEFBR14 IEFBR14 0000
IEF404I IEFBR14 - ENDED - TIME=00.46.14
$HASP395 IEFBR14 ENDED
$HASP309 INIT 2 INACTIVE ***** C=BA
```

Exec REXX de z/OS UNIX

Tal como se indica en *Using REXX and z/OS UNIX System Services* (SA22-7806), z/OS UNIX ofrece el mandato **seteuid** SYSCALL para establecer el UID efectivo del proceso actual. Hay unas pocas cosas que se deben tener en cuenta cuando se utiliza el mandato **seteuid**.

- El mandato **seteuid** utiliza el UID de z/OS UNIX, no el ID de usuario de MVS. Primero debe determinar el UID del ID de usuario de destino, lo que se puede realizar con el mandato **getpwnam** SYSCALL.
- El ID de usuario que ejecuta el mandato **seteuid** debe tener permiso READ sobre el perfil BPX.SRV.<IDusuario>, en la clase SURROGAT del producto de seguridad para poder conmutar al ID de usuario identificado por <IDusuario> sin especificar una contraseña. Tenga en cuenta que cuando varios ID de usuario comparten el mismo UID, no hay forma de determinar qué IDs de usuario se comprobarán.

```
/* rexx */
myID='ibmuser'
say userid()
address SYSCALL 'getpwnam' myID 'pw.'
```

```
say pw.1 pw.2 pw.3 pw.4 pw.5
address SYSCALL 'seteuid' pw.2 /* PW_UID = 2 */
say retval errno errnojr
say userid()
```

Esta salida de inicio de sesión de ejemplo, ejecutada por el ID de usuario de tarea iniciada, generará los mensajes de consola siguiente:

```
+FEK910I LOGON EXIT: invoking logon processing exit in thread 515
+FEK910I LOGON EXIT: STCRSE
+FEK910I LOGON EXIT: IBMUSER 1 0 / /bin/sh
+FEK910I LOGON EXIT: 0 0 0
+FEK910I LOGON EXIT: IBMUSER
+FEK910I LOGON EXIT: completed logon processing exit in thread 515
```

Puntos de salida disponibles

Los puntos de salida siguientes los proporciona Developer for System z:

- "audit.action"
- "logon.action"

audit.action

- **Temporización:**

La salida de usuario se invoca cuando se cierra el archivo de registro de auditoría activo. (La auditoría continúa ya que RSE ha cambiado a un archivo de registro de auditoría.)

- **Argumentos de invocación (1):**

- <audit_log>: nombre de vía de acceso completo del archivo de registro de auditoría cerrado

- **Ejemplo:**

```
/usr/lpp/rdz/samples/process_audit.rex
```

Este z/OS UNIX REXX exec de ejemplo construye un trabajo de proceso por lotes que procesará el registro de auditoría cerrado.

logon.action

- **Temporización:**

La salida de usuario se invoca cuando un usuario ha completado el proceso de inicio de sesión.

- **Argumentos de invocación (6):**

- -i <IDusuario>: ID de usuario de cliente, las mayúsculas y minúsculas son según indica el cliente
- -u <vía_acceso_registro_usuario>: directorio en el que se mantienen los registros de usuario de este cliente
- -s <vía_acceso_registro_servidor>: directorio en el que se mantienen los registros de servidor
- -c <vía_acceso_configuración>: directorio en el que se mantienen los archivos de configuración
- -b <vía_acceso_binarios>: directorio en el que se instala Developer for System z
- -p <puerto>: puerto del daemon RSE

- **Ejemplo:**

```
/usr/lpp/rdz/samples/process_logon.sh
```

Este script de shell z/OS UNIX de ejemplo escribe un mensaje de inicio de sesión en la consola.

Capítulo 10. Personalizar el entorno TSO

Este capítulo está destinado a servir de ayuda para emular un procedimiento de inicio de sesión TSO añadiendo sentencias DD y conjuntos de datos al entorno TSO en Developer for System z.

El servicio de mandatos TSO

El servicio de mandatos TSO es el componente de Developer for System z que ejecuta mandatos TSO e ISPF (por lotes) y devuelve el resultado al cliente solicitante. Estos mandatos puede solicitarlos implícitamente el producto o explícitamente el usuario.

Los miembros de ejemplo suministrados con Developer for System z crean un entorno TSO/ISPF mínimo. Si los desarrolladores de su establecimiento necesitan acceso a bibliotecas personalizadas o de terceros, el programador del sistema z/OS debe añadir las sentencias DD y bibliotecas necesarias al entorno de servicio de mandatos TSO. Aunque la implementación es diferente en Developer for System z, la lógica subyacente es idéntica al procedimiento de inicio de sesión de TSO.

Nota: El servicio de mandatos TSO es una herramienta de línea de mandatos no interactiva, por lo que los mandatos o procedimientos que solicitan datos o visualizan paneles de ISPF no funcionarán. Para ejecutarlos, será necesario un emulador 3270, como el Emulador de conexión de host que forma parte del cliente Developer for System z.

Métodos de acceso

Desde la versión 7.1, Developer for System z ofrece una opción relativa a cómo acceder al servicio de mandatos TSO.

- Pasarela de cliente TSO/ISPF de ISPF, que requiere un nivel de servicio mínimo de ISPF. Este es el método predeterminado utilizado en los ejemplos suministrados.
- Una transacción APPC (como en los releases anteriores a la versión 7.1). Este método está en desuso.

Nota:

- El servicio de Pasarela de cliente TSO/ISPF de ISPF sustituye a la función de SCLM Developer Toolkit utilizada en la versión 7.1.
- La utilización de APPC por parte de Developer for System z está en desuso. La información relacionada con APPC se ha eliminado de esta publicación. Para obtener más información, consulte el libro blanco *Using APPC to provide TSO command services* (SC14-7291), disponible en la biblioteca de Developer for System z, <http://www-01.ibm.com/support/docview.wss?uid=swg27038517>.

Compruebe `rsed.envvars` para determinar el método de acceso utilizado para hosts de la versión 7.1 y posteriores. Si se han utilizado los valores predeterminados durante el proceso de configuración, `rsed.envvars` reside en `/etc/rdz/`.

- Si la sentencia `_RSE_JAVAOPTS="_RSE_JAVAOPTS -DTSO_SERVER=APPC"` no está presente (o está comentada), se utiliza el servicio de Pasarela de cliente TSO/ISPF de ISPF.

- Si la sentencia `_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DTSO_SERVER=APPC"` está presente (y no está comentada), se utiliza APPC.

Utilizar el método de acceso de Pasarela de cliente TSO/ISPF

ISPF.conf

El archivo de configuración ISPF.conf (por omisión ubicado en `/etc/rdz/`) define el entorno TSO/ISPF utilizado por Developer for System z. Sólo existe un archivo de configuración ISPF.conf activo, que utilizan todos los usuarios de Developer for System z.

La sección principal del archivo de configuración define los nombres de DD y las concatenaciones de conjuntos de datos relacionados, como en el ejemplo siguiente:

```
sysproc=ISP.SISPCLIB,FEK.SFEKPROC
ispmllib=ISP.SISPMENU
isptllib=ISP.SISPTENU
isppllib=ISP.SISPPENU
ispplib=ISP.SISPSLIB
ispllib=ISP.SISpload
myDD=HLQ1.LLQ1,HLQ2.LLQ2
```

- Cada definición de DD utiliza exactamente una línea (la multilínea no está soportada) y no hay límites de longitud de línea.
- Las definiciones no son sensibles a las mayúsculas y minúsculas.
- Las líneas de comentarios empiezan por un asterisco (*).
- Los nombres de DD van seguidos de un signo igual (=), que a su vez va seguido de la concatenación de conjuntos de datos. Los diversos nombres de conjuntos de datos están separados por una coma (,).
- Las búsquedas en las concatenaciones de conjuntos de datos se realizan en el orden de la lista.
- Los conjuntos de datos deben estar totalmente calificados, sin entrecomillarse (') y sin utilizar variables.
- Todos los conjuntos de datos se asignan con `DISP=SHR`.
- Pueden añadirse nombres de DD nuevos a voluntad, pero deben ajustarse a las normas (JCL) para los nombres de DD y no pueden entrar en conflicto con otros parámetros de configuración de ISPF.conf. Asimismo, `ISPPROF` se asigna dinámicamente (`DISP=NEW,DELETE`) por medio del servicio de Pasarela de cliente TSO/ISPF.

Utilizar perfiles ISPF existente

Por omisión, La Pasarela de cliente TSO/ISPF crea un perfil ISPF temporal para el servicio de mandatos TSO. Sin embargo, puede indicar a la Pasarela de cliente z TSO/ISPF que utilice una copia de un perfil ISPF existente. La clave es aquí la sentencia `_RSE_ISPF_OPTS` de `rsed.envvars`.

```
#_RSE_ISPF_OPTS="$_RSE_ISPF_OPTS&ISPPROF=&SYSUID..ISPPROF"
```

Descomente la sentencia (elimine el carácter de almohadilla (#) inicial) y especifique el nombre totalmente calificado del conjunto de datos del perfil ISPF existente para utilizar este recurso.

Pueden utilizarse las variables siguientes en el nombre del conjunto de datos:

- `&SYSUID`. para sustituir el ID de usuario del desarrollador
- `&SYSPREF`. para sustituir el prefijo TSO del desarrollador

- &SYSDNAME. para sustituir el nombre del sistema tal como se especifica en el miembro parmlib IEASYSxx

Nota:

- Si el nombre del conjunto de datos pasado en "ISPPROF" no es válido, se utiliza en su lugar un perfil ISPF temporal vacío.
- El perfil ISPF (tanto el temporal como el copiado) se suprime al final de la sesión. Los cambios efectuados en el perfil no se fusionan con el perfil ISPF existente.

Utilizar un exec asignación

La sentencia `allocjob` de `ISPF.conf` (que está comentada por omisión) señala a un exec que puede utilizarse para suministrar más asignaciones de conjunto de datos por ID de usuario.

```
*allocjob = ISP.SISPSAMP(ISPZISP2)
```

Descomente la sentencia (elimine el carácter de asterisco (*) inicial) y especifique la referencia totalmente calificada al exec de asignación para utilizar este recurso.

- El exec se ejecuta después de la asignación de `ISPPROF` y de los DD definidos en `ISPF.conf`, pero antes de inicializar ISPF. Asegúrese del que el exec de asignación no deshace estas definiciones.
- Se pasa 1 parámetro al exec: el ID de usuario del llamante.
- En la biblioteca de ejemplo `FEK.#CUST.CNTL` se proporciona un exec de ejemplo `CRAISPRX`, a menos que haya especificado otra ubicación al personalizar y someter el trabajo `FEK.SFEKSAMP(FEKSETUP)`. Consulte a sección "Configuración de personalización" de la publicación *Guía de configuración de host* (SC11-3660) para obtener más detalles.

Nota: Dado que el exec se llama antes de inicializar ISPF, no puede utilizarse `VPUT` ni `VGET`. Sin embargo, puede crear su propia implementación de estas funciones utilizando un archivo PDS(E) o VSAM.

Utilizar varios ejecutables de asignación

Aunque `ISPF.conf` sólo da soporte a la llamada a un exec de asignación, no hay límite para que ese exec llame a otro exec. Y el ID de usuario del cliente que se pasa como parámetro abre la posibilidad de llamar a ejecutables de asignación personalizados. Por ejemplo, puede comprobar si el miembro `USERID'.EXEC(ALLOC)'` existe y ejecutarlo.

Una variante elaborada de este tema permite utilizar los procedimientos de inicio de sesión TSO existentes, del siguiente modo:

- Lea un archivo de configuración específico del usuario, como `USERID'.FEKPROF'`.
- Observe qué procedimiento de inicio de sesión se menciona en el archivo.
- Lea el procedimiento mencionado en `SYS1.PROCLIB` y analícelo para encontrar las sentencias DD y las asignaciones de conjunto de datos que contiene.
- Asigne el conjunto de datos de modo similar al procedimiento de inicio de sesión real.

Varios archivos ISPF.conf con varias configuraciones de Developer for System z

Si los escenarios de exec de asignación descritos en las secciones anteriores no pueden manejar sus necesidades específicas, puede crear instancias diferentes del

servidor de comunicaciones RSE de Developer for System z, cada una de ellas con su propio archivo ISPF.conf. El principal inconveniente del método descrito más abajo es que los usuarios de Developer for System z deben conectarse a servidores diferentes del mismo host para obtener el entorno TSO deseado.

Nota: Para crear una segunda instancia del servidor RSE sólo es necesario duplicar y actualizar los archivos de configuración, el JCL de inicio y las definiciones de tareas iniciadas. No es necesario realizar una nueva instalación del producto, ni tampoco duplicar ningún código.

```
$ cd /etc/rdz
$ mkdir /etc/rdz/tso2
$ cp rsed.envvars /etc/rdz/tso2
$ cp ISPF.conf /etc/rdz/tso2
$ ls /etc/rdz/tso2
ISPF.conf          rsed.envvars
$ oedit /etc/rdz/tso2/rsed.envvars
-> cambiar: _RSE_RSED_PORT=4037
-> cambiar: CGI_ISPCONF=/etc/rdz/tso2
-> cambiar: -Ddaemon.log=/var/rdz/logs/tso2
-> cambiar: -Duser.log=/var/rdz/logs/tso2
-> añadir al FINAL:
# -- NECESARIO PARA ENCONTRAR LOS ARCHIVOS DE CONFIGURACIÓN RESTANTES
CFG_BASE=/etc/rdz
CLASSPATH=.:$CFG_BASE:$CLASSPATH
# --
$ oedit /etc/rdz/tso2/ISPF.conf
-> cambiar: cambiar según las necesidades
```

Los mandatos del ejemplo anterior copian los archivos de configuración de Developer for System z que necesitan cambios en un directorio tso2 de reciente creación. La variable CGI_ISPCONF en rsed.envvars se debe actualizar para definir el nuevo directorio de inicio de ISPF.conf, y daemon.log y user.log se deben actualizar para definir una nueva ubicación de registro (que se crea automáticamente si no existe). La actualización de _RSE_RSED_PORT asegura que tanto el daemon RSE existente como el nuevo utilicen números de puerto exclusivos. La actualización de CLASSPATH garantiza que RSE pueda encontrar los archivos de configuración que no se han copiado en tso2. El propio archivo ISPF.conf puede actualizarse para ajustarlo a las necesidades. Tenga en cuenta que el área de trabajo de ISPF (variable CGI_ISPWORK de rsed.envvars) puede compartirse entre ambas instancias.

La tarea restante consiste en crear una tarea iniciada para RSE que utilice un número de puerto nuevo y los nuevos archivos de configuración /etc/rdz/tso2. Tenga en cuenta que si no se cambia _RSE_RSED_PORT en rsed.envvars, la tarea nueva iniciada debe especificar un puerto nuevo como argumento de inicio.

Consulte la *Guía de configuración de host de IBM Rational Developer for System z* (SC11-3660) para obtener más información sobre las acciones mostradas anteriormente en este apartado.

Capítulo 11. Ejecutar varias instancias

En algunas ocasiones le interesará tener múltiples instancias de Developer for System z activas en el mismo sistema; por ejemplo, al probar una ampliación. Sin embargo, algunos recursos como los puertos TCP/IP no se pueden compartir, por lo que los valores predeterminados no siempre son aplicables. Utilice la información de esta sección para planificar la coexistencia de distintas instancias de Developer for System z y después podrá usar esta guía de configuración para personalizarlas.

Aunque es posible compartir algunos componentes de Developer for System z entre dos (o más) instancias, es mejor que NO lo haga, a menos que los correspondientes niveles de software sean idénticos y que los únicos cambios estén en los miembros de configuración. Developer for System z deja suficiente margen de personalización para crear múltiples instancias que no se solapen, y le aconsejamos encarecidamente que utilice estas características.

Nota:

- FEK y /usr/lpp/rdz son el calificador de alto nivel y la vía de acceso que se emplean durante la instalación del producto. FEK.#CUST, /etc/rdz y /var/rdz son las ubicaciones predeterminadas que se utilizan durante la personalización del producto (consulte el apartado "Configuración de personalización" de la publicación *Guía de configuración de host* (SC11-3660) para obtener más información)..
- Debe instalar Developer for System z en un sistema de archivos privado (HFS o zFS) para desplegar fácilmente los componentes z/OS UNIX del producto.
- Si no puede utilizar un sistema de archivos privado, debe utilizar una herramienta de archivado como el mandato tar de z/OS UNIX tar para transportar los directorios de z/OS UNIX de un sistema a otro. Eso permite conservar los atributos (como por ejemplo el control de programa) de los archivos y directorios de Developer for System z.

Consulte la publicación *UNIX System Services Command Reference* (SA22-7802) para obtener más información acerca de los siguientes mandatos de ejemplo para archivar y restaurar el directorio de instalación de Developer for System z.

- Archivar: `cd /SYS1/usr/lpp/rdz; tar -cSf /u/userid/rdz.tar`
- Restaurar: `cd /SYS2/usr/lpp/rdz; tar -xSf /u/userid/rdz.tar`

Configuración idéntica en todo un sysplex

Los archivos de configuración de Developer for System z (y el código) se pueden compartir entre los distintos sistemas de un sysplex (cada sistema ejecuta su propia copia idéntica de Developer for System z, si se siguen algunas directrices. Tenga en cuenta que esta información es para instancias de Developer for System z autónomas. Hay más reglas de configuración de TCP/IP aplicables cuando utiliza VIPA dinámico distribuido para agrupar varios servidores (cada uno en un sistema distinto) en un servidor virtual, como se documenta en "VIPA dinámico distribuido" en la página 68.

- Los archivos de registro deben terminar en ubicaciones exclusivas para evitar que un sistema sobrescriba la información de otro. Direccionando los registros de z/OS UNIX a ubicaciones concretas con las directivas `daemon.log` y `user.log` de `rsed.envvars`, puede compartir los archivos de configuración si monta un

sistema de archivos de z/OS UNIX específico del sistema en la vía de acceso especificada. De esta manera, todos los registros se escriben en el mismo sitio lógico, pero terminan en ubicaciones físicas distintas.

- Los directorios del tipo configuración, como `/etc/rdz/` y `/var/rdz/pushtoclient/`, se pueden compartir en sysplex, ya que Developer for System z los utiliza en modalidad de sólo lectura.
- Los directorios de datos temporales, como `/tmp/` y `/var/rdz/WORKAREA/`, deberían ser exclusivos para cada sistema, ya que los nombres de archivos temporales no tienen en cuenta sysplex.
- Si comparte el código, debe compartir también los archivos de configuración para asegurar que no tienen algunos sistemas fuera de la sincronización una vez aplicado el mantenimiento.
- Si comparte un archivo de configuración `/etc/rdz/pushtoclient.properties` activo, también debe compartir el directorio de metadatos relacionado, `/var/rdz/pushtoclient/`.

Archivos de configuración diferentes con idéntico nivel de software

En un conjunto de circunstancias limitado, puede compartir todo salvo (algunos de) los componentes personalizables. Por ejemplo, proporcionar acceso no SSL para la utilización en el local y comunicación codificada por SSL para la utilización fuera del local.

Atención: La configuración compartida NO se puede utilizar de manera segura para probar el mantenimiento, un avance tecnológico o un nuevo release.

Para configurar otra instancia de una instalación activa de Developer for System z, rehaga los pasos de personalización para los componentes que sean distintos, utilizando diferentes conjuntos de datos, directorios y puertos para evitar que se solape con la instalación actual.

En el ejemplo SSL mencionado anteriormente, la configuración del daemon RSE actual se puede clonar y, después, la configuración clonada se puede actualizar. A continuación, el JCL de inicio del daemon RSE puede clonarse y personalizarse con un nuevo puerto TCP/IP y la ubicación de los archivos de configuración actualizados. Las personalizaciones de MVS (supervisor de trabajos JES, etc.) se pueden compartir entre las instancias SSL y las no SSL. Ello daría como resultado las siguientes acciones:

```
$ cd /etc/rdz
$ mkdir /etc/rdz/ssl
$ cp rsed.envvars /etc/rdz/ssl
$ cp ssl.properties /etc/rdz/ssl
$ ls /etc/rdz/ssl/
rsed.envvars    ssl.properties
$ oedit /etc/rdz/ssl/rsed.envvars
-> cambiar: _RSE_RSED_PORT=4047
-> cambiar: -Ddaemon.log=/var/rdz/logs/ssl
-> cambiar: -Duser.log=/var/rdz/logs/ssl
-> añadir al FINAL:
# -- NECESARIO PARA ENCONTRAR LOS ARCHIVOS DE CONFIGURACIÓN RESTANTES
CFG_BASE=/etc/rdz
CLASSPATH=.:$CFG_BASE:$CLASSPATH
# --
$ oedit /etc/rdz/ssl/ssl.properties
-> cambiar: cambiar según las necesidades
```

Los mandatos del ejemplo precedente copian los archivos de configuración de Developer for System z que necesitan cambios en un directorio `ssl` de reciente creación. Las variables `daemon.log` y `user.log` de `rsed.envvars` deben actualizarse para definir una ubicación de registro nueva (que se crea automáticamente en caso de que no exista). La actualización de `CLASSPATH` garantiza que RSE pueda encontrar los archivos de configuración que no se han copiado en `ssl`. El propio archivo `ssl.properties` puede actualizarse para ajustarlo a las necesidades.

La tarea restante consiste en crear una tarea iniciada para RSE que utilice un número de puerto nuevo y los nuevos archivos de configuración `/etc/rdz/ssl`.

Consulte las secciones relacionadas en la *IBM Rational Developer for System z Guía de configuración de host* (SC11-3660) para obtener más información sobre las acciones mostradas anteriormente en esta sección.

Nota: Cuando se utiliza esta técnica para crear clones dependientes, sabemos que `ssl.properties` debe estar siempre clonado al directorio dependiente, aunque no cambie. `rsed.envvars` también debe copiarse y como mínimo debe cambiarse la directiva `_RSE_RSED_PORT`.

Sincronización automatizada

En el ejemplo de SSL mencionado anteriormente, los cambios entre el daemon RSE no habilitado por SSL y habilitado por SSL son mínimos, lo que permite automatizar el proceso de mantenimiento de la sincronización de los archivos `rsed.envvars` correspondientes. Esto simplifica el lanzamiento del servicio porque sólo es necesario mantener un archivo `rsed.envvars`.

El ejemplo siguiente añade el número de puerto RSED a los nombres de directorio de registro y actualiza la variable `CLASSPATH` de modo que los clones encontrarán los archivos de configuración restantes. A continuación, el ejemplo mejora la tarea iniciada JCL iniciada del daemon de RSE habilitado por SSL para clonar el archivo `rsed.envvars` del daemon de RSE no SSL al iniciar, actualizando el número de puerto en los procesos. Puesto que el número de puerto está incorporado en el nombre del directorio, es diferente para ambos daemons.

1. Prepare el `rsed.envvars` maestro.

```
$ oedit /etc/rdz/rsed.envvars
-> change: -Ddaemon.log=/var/rdz/logs/$RSE_RSED_PORT
-> change: -Duser.log=/var/rdz/logs/$RSE_RSED_PORT
-> añadir al FINAL:
# -- NECESARIO PARA LOS CLONES PARA ENCONTRAR LOS ARCHIVOS DE
CONFIGURACIÓN RESTANTES
CFG_BASE=/etc/rdz
CLASSPATH=.:$CFG_BASE:$CLASSPATH
# --
```

2. Prepare los otros archivos de configuración (que no son archivos `rsed.envvars`) que difieren entre el master (no SSL) y el clon (SSL).

```
$ mkdir /etc/rdz/ssl
$ cp /etc/rdz/ssl.properties /etc/rdz/etc/rdz/ssl
$ oedit /etc/rdz/ssl/ssl.properties
-> cambiar: cambiar según las necesidades
```

3. Crear una tarea iniciada RSED que clonará el archivo `rsed.envvars` base y alterará el puerto del daemon RSE (4035 -> 4034).

```
/**
/** RSE DAEMON - SSL
/**
/**RSED      PROC IVP=,                * 'IVP' para hacer una prueba de IVP
/**          HOME='/usr/lpp/rdz',
```

```

//          CNFG='/etc/rdz/ssl'
// *
//          SET SED='"/RSED_PORT/s/4035/4034/'
//          SET FILE='rsed.envvars'
// *
// * copy /etc/rdz/rsed.envvars to /etc/rdz/ssl/rsed.envvars
// * and alter RSED_PORT
// *
//CLONE     EXEC PGM=BPXBATCH,REGION=0M,COND=(4,LT),
//  PARM='SH cd &CNFG;sed &SED ../&FILE>&FILE'
//STDOUT    DD SYSOUT=*
//STDERR    DD SYSOUT=*
// *
// * iniciar RSED con el archivo rsed.envvars recién creado
// *
//RSED      EXEC PGM=BPXBATSL,REGION=0M,TIME=NOLIMIT,COND=(4,LT),
//  PARM='PGM &HOME./bin/rsed.sh &IVP -C&CNFG'
//STDOUT    DD SYSOUT=*
//STDERR    DD SYSOUT=*
//          PEND
// *

```

Todas las demás situaciones

Cuando hay cambios de código implicados (de mantenimiento, avances tecnológicos, nuevo release), o cuando los cambios son bastante complejos, debe hacer otra instalación de Developer for System z. En este apartado se describen los posibles puntos de conflicto entre distintas instalaciones.

La siguiente lista es una visión general resumida de los elementos que deben ser distintos (o conviene que lo sean) entre las instancias de Developer for System z:

- SMP/E CSI
- Bibliotecas de instalación
- Puerto TCP/IP del supervisor de trabajos JSE, así como su archivo de configuración FEJJCNF
- JCL de inicio del supervisor de trabajos JES
- Nombre de la transacción APPC
- Archivos de configuración de RSE, rsed.envvars, *.properties y *.conf
- Puerto TCP/IP de RSE
- JCL de arranque de RSE

A continuación se proporciona una visión general más detallada:

- SMP/E CSI
 1. Instale cada instancia de Developer for System z en un CSI separado. SMP/E impedirá una segunda instalación del mismo FMID en un CSI, pero aceptará la instalación de otro FMID. Si el segundo FMID es una versión más reciente, suprimirá la versión existente del producto. Si el segundo FMID es una versión más antigua, la instalación fallará debido a los nombres de componentes duplicados.
- Bibliotecas de instalación
 1. Instale cada instancia de Developer for System z en conjuntos de datos y directorios separados. Tenga presente que solo puede cambiar la vía de acceso z/OS UNIX prefijando el valor predeterminado de IBM, que es /usr/lpp/rdz. Un ejemplo válido podría ser /service/usr/lpp/rdz.
 2. El trabajo de configuración de personalización FEK.SFEKSAMP(FEKSETUP) crea los conjuntos de datos y directorios utilizados para almacenar los archivos de

configuración. Dado que los archivos de configuración deben ser exclusivos, y para evitar la sobrescritura de las personalizaciones existentes, debe utilizar nombres exclusivos de conjunto de datos y directorio al someter este trabajo.

- Componentes obligatorios

1. El archivo de configuración del supervisor de trabajos JES, FEK.#CUST.PARMLIB(FEJJCNFG), contiene el número de puerto TCP/IP del supervisor de trabajos JES y, por lo tanto, no se puede compartir. El propio miembro se puede red denominar (si también se actualiza el JCL), lo que le permite colocar todas las versiones personalizadas de este miembro en el mismo conjunto de datos, si no va a hacer las actualizaciones en el conjunto de datos de instalación.
2. El JCL de arranque del supervisor de trabajos JES, FEK.#CUST.PROCLIB(JMON), hace referencia a FEJJCNFG y, por lo tanto, tampoco se puede compartir. Tras red denominar el miembro (y la tarjeta JOB si la ha iniciado como trabajo de usuario), puede colocar todos los JCL en el mismo conjunto de datos.
3. El archivo de configuración de RSE, /etc/rdz/rsed.envvars, contiene referencias a la vía de instalación y, opcionalmente, a la ubicación de los registros del servidor, que tiene que ser exclusivo. El nombre del archivo es obligatorio, por lo tanto, no podrá conservar distintas copias en el mismo directorio.
4. El archivo de configuración ISPF.conf tiene una referencia a FEK.SFEKPROC. Es específico del nivel de software, por lo que debe crear un archivo ISPF.conf por cada instancia.
5. Todos los demás archivos de configuración basados en z/OS UNIX (como *.properties) deben residir en el mismo directorio que rsed.envvars y, por tanto no se pueden compartir, ya que rsed.envvars debe estar en una ubicación no compartida.
6. El JCL de inicio de RSE FEK.#CUST.PROCLIB(RSED) no puede compartirse, ya que define el número de puerto TCP/IP y contiene una referencia a los directorios de instalación y configuración, que deben ser exclusivos. Tras red denominar el miembro (y la tarjeta JOB si la ha iniciado como trabajo de usuario), puede colocar todos los JCL en el mismo conjunto de datos.

- Componentes opcionales

1. Los puertos TCP/IP de REXEC y SSH se pueden compartir sin restricciones.
2. La transacción APPC tiene una referencia a FEK.SFEKPROC(FEKFRSRV), el servidor de mandatos TSO. Este es específico del nivel de software, por lo que debe crear una transacción APPC por cada instancia. Tenga presente que, dado que el nombre de la transacción APPC cambia, hay que definir la variable _FEKFSCMD_TP_NAME_ en rsed.envvars.
3. Algunos procedimientos ELAXF* tienen una referencia a FEK.SFEKLOAD o FEK.SFEKAUTH, las bibliotecas de carga de Developer for System z. Consulte la nota sobre JCLLIB en el apartado "Procedimientos de construcción remota ELAXF*" de la *Guía de configuración de host* (SC11-3660) para encontrar una posible manera de poner diferentes conjuntos a disposición de los usuarios.
4. El soporte bidireccional en las regiones CICS se basa en un miembro de la biblioteca de carga y, por lo tanto, no se puede compartir entre releases. Sin embargo, si el nombre del módulo de carga es idéntico para todas las instancias, podrá compartir la versión más reciente entre las instancias, incluso en los distintos releases. La compatibilidad hacia atrás no está disponible si el módulo de carga ha cambiado de nombre.
5. Los módulos de carga del Gestor de despliegue de aplicaciones incluidos en las regiones CICS son compatibles hacia atrás y, por lo tanto, la versión más reciente se puede compartir en los distintos releases.

6. El VSAM del CRD del Gestor de despliegue de aplicación es compatible hacia atrás y, por lo tanto, la versión más reciente se puede compartir en los distintos releases.
7. Las definiciones de recursos CICS del Gestor de despliegue de aplicación son compatibles hacia atrás y, por lo tanto, la versión más reciente se puede compartir en los distintos releases.
8. Los VSAM de CARMA podrían cambiar entre niveles de software, por lo que no conviene compartirlos.
9. La tarea iniciada del Gestor de depuración es compatible con versiones anteriores y, por consiguiente, la versión más reciente se puede compartir en los distintos releases.

Capítulo 12. Resolución de problemas de configuración

Este capítulo se propone ayudarle a resolver algunos problemas comunes que pueden surgir durante la configuración de Developer for System z, y tiene las secciones siguientes:

- “Anotar y configurar el análisis mediante FEKLOGS”
- “Archivos de registro” en la página 182
- “Archivos de vuelco” en la página 188
- “Rastrear” en la página 190
- “Bits de permiso de z/OS UNIX” en la página 193
- “Puertos TCP/IP reservados” en la página 196
- “Tamaño del espacio de direcciones” en la página 198
- “Información variada” en la página 199

La publicación *Developer for System z Messages and Codes* (SC14-7497) documenta los mensajes y los códigos de retorno generados por los componentes de Developer for System z. *Developer for System z Answers to common host configuration and maintenance issues* (SC14-7373) describe varios casos de problemas y sus soluciones.

Encontrará más información en la sección de soporte del sitio Web de Developer for System z (<http://www-03.ibm.com/software/products/us/en/developerforsystemz/>), donde hay fichas técnicas que le aportarán la información más reciente de nuestro equipo de soporte.

En la sección de biblioteca del sitio Web (<http://www-01.ibm.com/support/docview.wss?uid=swg27038517>) también hallará la versión más reciente de la documentación de Developer for System z los libros blancos.

El Information Center de Developer for System z (http://www-01.ibm.com/support/knowledgecenter/SSQ2R2_9.1.0/com.ibm.etools.getstart.wsentdev.doc/kc_version_welcome_rdz.html) documenta el cliente Developer for System z y cómo interactúa con el host (desde una perspectiva del cliente).

También encontrará información valiosa en la biblioteca Internet de z/OS, disponible en <http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/>.

Infórmenos si opina que a Developer for System z le falta alguna función. Puede abrir una Solicitud de mejora (RFE) en

<https://www.ibm.com/developerworks/support/rational/rfe/>

Anotar y configurar el análisis mediante FEKLOGS

La tarea iniciada RSED da soporte al mandato de operador **MODIFY LOGS** para recopilar registros de host de Developer for System z e información de configuración. Los datos recopilados se sitúan en el archivo z/OS UNIX, \$TMPDIR/feklogs%sysname.%jobname, donde \$TMPDIR es el valor de la directiva TMPDIR en rsed.envvars (/tmp predeterminado), %sysname es el nombre del sistema z/OS y %jobname es el nombre de la tarea iniciada RSED.

De forma predeterminada, sólo se recopilan registros del servidor. Las opciones de mandato le permiten recopilar diferentes registros:

USER	Recopilar archivos de registro para el ID de usuario especificado
AUDIT	Recopilar registros de auditoría
NOSERVER	No recopilar registros del servidor

Developer for System z consultará el producto de seguridad para permisos de acceso a perfilesFEK.CMD.LOGS.** para determinar si el peticionario tiene permiso para recopilar los registros especificados. De forma predeterminada, el peticionario es el ID de usuario de la tarea iniciada RSED, a menos que se especifique la opción OWNER. Sólo el peticionario tiene acceso al archivo que contiene los datos recopilados.

Para recopilar datos antes de que se pueda iniciar la tarea iniciada RSED, Developer for System z proporciona un trabajo de ejemplo, FEKLOGS, que recopila todos los archivos de registro z/OS UNIX así como información de instalación y configuración de Developer for System z.

El trabajo de ejemplo FEKLOGS se encuentra en FEK.#CUST.JCL, a menos que haya especificado otra ubicación al personalizar y someter el trabajo FEK.SFEKSAMP(FEKSETUP). Consulte a sección "Configuración de personalización" de la publicación *Guía de configuración de host* (SC11-3660) para obtener más detalles.

La personalización de FEKLOGS se describe en el JCL. La personalización abarca la provisión de algunas variables clave.

Nota: Los clientes de SDSF pueden utilizar el mandato de línea XDC en SDSF para guardar la salida de trabajo en un conjunto de datos, que a su vez se puede entregar al centro de soporte de IBM. Tenga en cuenta que el conjunto de datos de salida tiene que asignarse como VB 2051 (el valor predeterminado en SDSF es VB 240) para evitar el truncamiento de registros.

Archivos de registro

Developer for System z crea archivos de registro que le ayudarán a usted y al centro de soporte de IBM a identificar y resolver problemas. La lista que sigue es una visión general de los archivos de registro que se pueden crear en su sistema host z/OS. Junto a estos archivos de registro específicos del producto, no olvide consultar SYSLOG por si hay mensajes relacionados.

Los registros basados en MVS se pueden localizar mediante la sentencia DD pertinente. Los archivos de registro basados en z/OS UNIX se encuentran en los siguientes directorios:

- userlog/\$LOGNAME/

Los archivos de registro específicos del usuario están ubicados en userlog/\$LOGNAME/, donde userlog es el valor combinado de la directivas user.log y DSTORE_LOG_DIRECTORY de rsed.envvars, y \$LOGNAME es el ID de usuario de inicio de sesión (en mayúsculas). Si la directiva user.log no tiene caracteres de comentario o no está presente, se utiliza la vía de acceso inicial del usuario. La vía de acceso inicial se define en el segmento de seguridad OMVS del ID de usuario. Si la directiva DSTORE_LOG_DIRECTORY no tiene caracteres de comentario o no está presente, se añade .eclipse/RSE/ al valor user.log.

- .dstoreMemLogging - Registro de utilización de memoria de almacén de datos
- .dstoreTrace - Registro de acciones de almacén de datos

- `.dstoreHashmap.*` - Instantánea de la correlación hash de DataStore activa
- `.dstoreStackTrace.*` - Instantánea de las hebras de DataStore activas y de donde se invocaron
- `ffs.log` - Registro del servidor FFS (Foreign File System), que ejecuta funciones nativas de MVS
- `ffsget.log` - Registro del lector de archivos, que lee un conjunto de datos secuencial (SDS) o un miembro PDS
- `ffsput.log` - Registro del transcriptor de archivos, que escribe un conjunto de datos secuencial (SDS) o un miembro PDS
- `ffslock.log` - El registro del gestor de bloqueo, que bloquea/desbloquea un conjunto de datos secuencial o un miembro PDS
- `rsecomm.log` - Registro del servidor RSE, que maneja mandatos del cliente y el registro de comunicación de todos los servicios basados en RSE (puede contener un rastreo de la pila de excepciones Java)

Nota:

- El directorio `.eclipse` y los archivos de registro `.dstore*` empiezan por un punto (`.`), lo que hace que estén ocultos. Utilice el mandato de z/OS UNIX `ls -lA` para listar los archivos y directorios ocultos. Cuando se utiliza el cliente Developer for System z, seleccione la página de preferencias **Ventana > Preferencias... > Sistemas remotos > Archivos** y habilite "Mostrar archivos ocultos".
- `daemon-home/server/`
El daemon RSE y los archivos de registro específicos de la agrupación de hebras RSE se encuentran en `daemon-home/server`, donde `daemon-home` es el valor de la directiva `daemon.log` en `rsed.envvars`. Si la directiva `daemon.log` no tiene caracteres de comentario o no está presente, se utiliza el directorio inicial del ID de usuario asignado a la tarea iniciada RSED. El directorio inicial se define en el segmento de seguridad OMVS del ID de usuario.
 - `rsedaemon.log` - Registro del daemon RSE
 - `rseserver.log` - Registro de las agrupaciones de hebras RSE
 - `audit.log` - Seguimiento de auditoría de RSE
 - `serverlogs.count` - Contador para anotar las secuencias de agrupaciones de hebras RSE
 - `stderr.*.log` - Secuencia de error estándar de agrupaciones de hebras RSE
 - `stdout.*.log` - Secuencia de salida estándar de agrupaciones de hebras RSE
- `/tmp`
Los archivos de registro específicos de IVP (Installation Verification Program) se encuentran en el directorio al que hace referencia la variable `TMPDIR`, si esta variable está definida en `rsed.envvars`. Si la variable no está definida, los archivos se crean en `/tmp`. El mandato de operador **MODIFY LOGS** para la tarea iniciada RSED también crea su salida en este directorio.
 - `fekfivpi.log` - Registro de la prueba IVP de `fekfivpi`
 - `fekfivps.log` - Registro de la prueba IVP de `fekfivps`
 - `fekfivpc.log` - Registro de comunicaciones de la prueba IVP de `fekfivpc`
 - `feklogs.*` - Salida del mandato de operador **MODIFY LOGS**

Nota: Existen mandatos de operador disponibles para controlar la cantidad de datos grabados en algunos de los archivos de registro mencionados. Consulte la sección "Mandatos de operador" de la publicación *Guía de configuración de host* (SC11-3660) para obtener más información.

Registro del gestor de depuración

- **DD SYSPRINT**

El registro de rastreo y el registro de operaciones normales. El valor predeterminado del JCL de ejemplo FEK.#CUST.PROCLIB(DBGMGR) es SYSOUT=*.

Registro del supervisor de trabajos JES

- **DD SYSOUT**

Registro de las operaciones normales. El valor predeterminado del JCL de ejemplo FEK.#CUST.PROCLIB(JMON) es SYSOUT=*.

- **DD SYSPRINT**

Registro de rastreo. El valor predeterminado del JCL de ejemplo FEK.#CUST.PROCLIB(JMON) es SYSOUT=*. El rastreo se activa con el parámetro -TV; hallará más detalles en: “Rastreo del supervisor de trabajos JES” en la página 190.

Daemon RSE y registro de la agrupaciones de hebras

- **STDOUT DD**

Los datos redirigidos de stdout, la salida estándar de Java del daemon RSE. El valor predeterminado del JCL de ejemplo FEK.#CUST.PROCLIB(RSED) es SYSOUT=*.

- **STDERR DD**

Los datos redirigidos de stderr, la salida de error estándar de Java del daemon RSE. El valor predeterminado del JCL de ejemplo FEK.#CUST.PROCLIB(RSED) es SYSOUT=*.

- **daemon-home**

Los archivos de registro específicos del daemon RSE y de la agrupación de hebras RSE se encuentran en daemon-home, donde daemon-home es el valor de la directiva daemon.log de rsed.envvars. Si la directiva daemon.log no tiene caracteres de comentario o no está presente, se utiliza el directorio inicial del ID de usuario asignado a la tarea iniciada RSED. El directorio inicial se define en el segmento de seguridad OMVS del ID de usuario.

- rsedaemon.log - Registro del daemon RSE
- rseserver.log - Registro de las agrupaciones de hebras RSE
- audit.log - Seguimiento de auditoría de RSE
- serverlogs.count - Contador para anotar las secuencias de agrupaciones de hebras RSE
- stderr.*.log - Secuencia de error estándar de agrupaciones de hebras RSE
- stdout.*.log - Secuencia de salida estándar de agrupaciones de hebras RSE

Nota:

- serverlogs.count, stderr.*.log y stdout.*.log solamente se crean si la directiva enable.standard.log de rsed.envvars está activa, o si la función está activada dinámicamente con el mandato de operador **modify rstandardlog on**.
- El * de stderr.*.log y stdout.*.log es 1 de forma predeterminada. Sin embargo, puede haber varias agrupaciones de hebras RSE, en cuyo caso este número aumentará por cada nueva agrupación de hebras RSE a fin de asegurar que los nombres de los archivos sean exclusivos.
- Existen mandatos de operador disponibles para controlar la cantidad de datos grabados en algunos de los archivos de registro mencionados. Consulte la

sección "Mandatos de operador" de la publicación *Guía de configuración de host* (SC11-3660) para obtener más información.

- Los archivos `rse*.log` también pueden existir con una extensión `".last"` en lugar de `".log"` si se especifica `keep.last.log=true` en `rsed.envvars`. De forma predeterminada, no se crean los archivos de registro `".last"`.
- Los archivos `rse*.log` tendrán un nombre ampliado si se especifica `keep.all.logs=true` en `rsed.envvars`. De forma predeterminada se utiliza el nombre ampliado. A continuación se muestra un nombre ampliado de muestra, en el que RSED representa el nombre de espacio de direcciones del daemon RSE y `yyyymmddhhmmss` es una indicación de fecha y hora (año, mes, día, hora, minuto, segundo): `rseserver.RSED#yyyymmddhhmmss.log`

Registro de usuario de RSE

- **userlog/\$LOGNAME/**

Los componentes relacionados con RSE crean varios archivos de registro. Todos están ubicados en `userlog/$LOGNAME/`, donde `userlog` es el valor combinado de la directivas `user.log` y `DSTORE_LOG_DIRECTORY` de `rsed.envvars`, y `$LOGNAME` es el ID de usuario de inicio de sesión (en mayúsculas). Si la directiva `user.log` no tiene caracteres de comentario o no está presente, se utiliza la vía de acceso inicial del usuario. La vía de acceso inicial se define en el segmento de seguridad OMVS del ID de usuario. Si la directiva `DSTORE_LOG_DIRECTORY` no tiene caracteres de comentario o no está presente, se añade `.eclipse/RSE/` al valor `user.log`.

- `.dstoreMemLogging` - Registro de utilización de memoria de almacén de datos
- `.dstoreTrace` - Registro de acciones de almacén de datos
- `.dstoreHashmap.*` - Instantánea de la correlación hash de DataStore activa
- `.dstoreStackTrace.*` - Instantánea de las hebras de DataStore activas y de donde se invocaron
- `ffs.log` - Registro del servidor FFS (Foreign File System), que ejecuta funciones nativas de MVS
- `ffsget.log` - Registro del lector de archivos, que lee un conjunto de datos secuencial (SDS) o un miembro PDS
- `ffsput.log` - Registro del transcriptor de archivos, que escribe un conjunto de datos secuencial (SDS) o un miembro PDS
- `ffslock.log` - El registro del gestor de bloqueo, que bloquea o desbloquea un conjunto de datos secuenciales o un miembro PDS
- `rsecomm.log` - Registro del servidor RSE, que maneja mandatos del cliente y los registros de comunicación de todos los servicios basados en RSE (puede contener un rastreo de la pila de excepciones Java)

Nota:

- El directorio `.eclipse` y los archivos de registro `.dstore*` empiezan por un punto (`.`), lo que hace que estén ocultos. Utilice el mandato de `z/OS UNIX ls -lA` para listar los archivos y directorios ocultos. Cuando se utiliza el cliente Developer for System z, seleccione la página de preferencias **Ventana > Preferencias... > Sistemas remotos > Archivos** y habilite "Mostrar archivos ocultos".
- La creación de los archivos de registro `.dstore*` está controlada por las opciones de inicio `-DDSTORE_*` Java, como se describe en el apartado "Definición de parámetros de inicio de Java adicionales con `_RSE_JVAOPTS`" de la publicación *Guía de configuración de host* (SC11-3660).

- Los archivos de registro `.dstore*` se crean en UTF8. Utilice el mandato de z/OS UNIX `iconv -f UTF8 -t IBM-1047 .dstore*` para visualizarlos en EBCDIC (al utilizar la página de códigos IBM-1047).
- Al contrario que todos los archivos `*.log`, los archivos de registro `.dstore*` no se eliminan automáticamente cuando se vuelve a conectar el cliente. La eliminación de estos archivos es una acción manual.
- Existen mandatos de operador disponibles para controlar la cantidad de datos grabados en algunos de los archivos de registro mencionados. Consulte la sección "Mandatos de operador" de la publicación *Guía de configuración de host* (SC11-3660) para obtener más información.
- Los archivos `ffs*.log` y `rsecomm.log` también pueden existir con una extensión `".last"` en lugar de una extensión `".log"` si `keep.last.log=true` se ha especificado en `rsed.envvars`. De forma predeterminada, no se crean los archivos de registro `".last"`.
- Los archivos `ffs*.log` y `rsecomm.log` tendrán un nombre ampliado si se ha especificado `keep.all.logs=true` en `rsed.envvars`. De forma predeterminada se utiliza el nombre ampliado. A continuación se muestra un nombre ampliado de muestra, en el que `RSEDx` representa el nombre de espacio de direcciones de la agrupación de hebras en la que el usuario está activo y `yyyymmddhhmmss` es una indicación de fecha y hora (año, mes, día, hora, minuto, segundo):
`ffs.RSEDx#yyyymmddhhmmss.log`

Registro de SCLM Developer Toolkit

- **userlog/\$LOGNAME/rsecomm.log**
Registro de comunicación de SCLM Developer Toolkit, donde `userlog` es el valor combinado de la directivas `user.log` y `DSTORE_LOG_DIRECTORY` de `rsed.envvars`, y `$LOGNAME` es el ID de usuario de inicio de sesión (en mayúsculas). Si la directiva `user.log` no tiene caracteres de comentario o no está presente, se utiliza la vía de acceso inicial del usuario. La vía de acceso inicial se define en el segmento de seguridad OMVS del ID de usuario. Si la directiva `DSTORE_LOG_DIRECTORY` no tiene caracteres de comentario o no está presente, se añade `.eclipse/RSE/` al valor `user.log`.

Registro de CARMA

- **Trabajo de servidor de CARMA**
Al abrir una conexión con CARMA mediante la interfaz de proceso por lotes, `FEK.#CUST.SYSPROC(CRASUBMT)` iniciará un trabajo servidor (cuyo propietario será el ID del usuario) llamado `CRApuerto`, siendo `puerto` el número de puerto TCP/IP que se utiliza.
- **CARMALOG DD**
Si se especifica la sentencia DD `CARMALOG` en el método de inicio de CARMA elegido, los registros de CARMA se redirigen a esta sentencia DD en el trabajo servidor; de lo contrario, van a `SYSPRINT`.
- **DD SYSPRINT**
El `SYSPRINT` DD del trabajo servidor contiene los registros de CARMA, si la sentencia DD `CARMALOG` no está definida.
- **SYSTSPRT DD**
El `SYSTSPRT` DD del trabajo del servidor contiene los mensajes del sistema (TSO) para el inicio del servidor CARMA.
- **userlog/\$LOGNAME/rsecomm.log**

Registro de comunicación de CARMA, donde `userlog` es el valor combinado de la directivas `user.log` y `DSTORE_LOG_DIRECTORY` de `rsed.envvars`, y `$LOGNAME` es el ID de usuario de inicio de sesión (en mayúsculas). Si la directiva `user.log` no tiene caracteres de comentario o no está presente, se utiliza la vía de acceso inicial del usuario. La vía de acceso inicial se define en el segmento de seguridad OMVS del ID de usuario. Si la directiva `DSTORE_LOG_DIRECTORY` no tiene caracteres de comentario o no está presente, se añade `.eclipse/RSE/` al valor `user.log`.

fekfivpc, registro de prueba IVP

- **/tmp/fekfivpc.log**

El mandato `fekfivpc` (prueba IVP relacionada con CARMA) creará el archivo `fekfivpc.log` para documentar la comunicación entre RSE y CARMA. El registro se creará en el directorio al que hace referencia la variable `TMPDIR`, si esta variable está definida en `rsed.envvars`. Si la variable no está definida, el archivo se creará en `/tmp`.

Registro de prueba IVP de fekfivpi

- **/tmp/fekfivpi.log**

Salida del mandato `fekfivpi -file` (prueba IVP relacionada con la pasarela de cliente TSO/ISPF). El registro se creará en el directorio al que hace referencia la variable `TMPDIR`, si esta variable está definida en `rsed.envvars`. Si la variable no está definida, el archivo se creará en `/tmp`.

Registro de prueba IVP de fekfivps

- **/tmp/fekfivps.log**

Salida del mandato `fekfivps -file` (prueba IVP relacionada con SCLMDT). El registro se creará en el directorio al que hace referencia la variable `TMPDIR`, si esta variable está definida en `rsed.envvars`. Si la variable no está definida, el archivo se creará en `/tmp`.

Registro de revisión de código

- **SYSTSPRT DD**

El SYSTSPRT DD del paso que invoca el procedimiento de revisión de código alberga los mensajes del componente frontal que controla el proceso de análisis de código.

- **WORKSPCE DD**

El WORKSPCE DD del paso que invoca el procedimiento de revisión de código alberga los mensajes de registro de espacio de trabajo de Eclipse del proceso de análisis de código.

- **ERRMSGs DD**

El ERRMSGs DD del paso que invoca el procedimiento de revisión de código alberga la salida `stderr` del proceso de análisis de código.

Registro de cobertura de código

- **SYSTSPRT DD**

El SYSTSPRT DD del paso que invoca el procedimiento de revisión de código alberga los mensajes del componente frontal que controla el proceso de análisis de código.

- **WORKSPCE DD**

El WORKSPACE DD del paso que invoca el procedimiento de revisión de código alberga los mensajes de registro de espacio de trabajo de Eclipse del proceso de análisis de código.

- ERRMSG DD

El ERRMSG DD del paso que invoca el procedimiento de revisión de código alberga la salida stderr del proceso de análisis de código.

Archivos de vuelco

Cuando un producto se interrumpe de forma anómala, se crea un vuelco de almacenamiento para ayudar a determinar el problema. La disponibilidad y la ubicación de los vuelcos depende en gran medida de los valores específicos del local. Los vuelcos podrían no crearse, o bien se podrían crear en distintas ubicaciones de las mencionadas en las secciones siguientes.

Vuelcos de MVS

Si el programa se ejecuta en MVS, compruebe los archivos de vuelco del sistema y compruebe también el JCL de las siguientes sentencias DD (en función del producto):

- SYSABEND
- SYSMDUMP
- SYSUDUMP
- CEEDUMP
- SYSPRINT
- SYSOUT

Consulte las publicaciones *MVS JCL Reference* (SA22-7597) y *Language Environment Debugging Guide* (GA22-7560) para obtener más información acerca de estas sentencias DD.

Volcados de Java

En z/OS UNIX, la mayoría de vuelcos de Developer for System z están controlados por la máquina virtual Java (JVM).

La JVM crea por defecto un conjunto de agentes de vuelco durante su inicialización (SYSTDUMP y JAVADUMP). Puede alterar temporalmente este conjunto de agentes de vuelco utilizando la variable de entorno `JAVA_DUMP_OPTS`, y aún puede alterar adicionalmente el conjunto utilizando `-Xdump` en la línea de mandatos. Las opciones de línea de mandatos de la JVM están definidas en la directiva `_RSE_JAVA_OPTS` de `rsed.envvars`. No cambie ninguno de los valores, a menos que se lo indique el centro de soporte de IBM.

Nota: La opción `-Xdump:what` de la línea de mandatos permite determinar qué agentes de vuelco existen al realizarse el inicio.

Los tipos de vuelco que se pueden producir son los siguientes:

SYSTDUMP

Vuelco de transacciones Java. Es un vuelco de almacenamiento sin formatear generado por z/OS.

El vuelco se escribe en un conjunto de datos MVS secuencial, utilizando un nombre predeterminado con el formato %uid.JVM.TDUMP.%job.D%y%m%d.T%H%M%S, o tal como viene determinado por el valor de la variable de entorno JAVA_DUMP_TDUMP_PATTERN.

Nota: JAVA_DUMP_TDUMP_PATTERN permite el uso de variables, que se convierten en un valor real en el momento del volcado de la transacción.

Tabla 43. Variables de JAVA_DUMP_TDUMP_PATTERN

Variable	Uso
%uid	ID de usuario
%job	Nombre de trabajo
%y	Año (2 dígitos)
%m	Mes (2 dígitos)
%d	Día (2 dígitos)
%H	Hora (2 dígitos)
%M	Minuto (2 dígitos)
%S	Segundo (2 dígitos)

CEEDUMP

Vuelco de Language Environment (LE). Vuelco del sistema, resumido y formateado, que muestra rastreo de la pila para cada hebra que esté en el proceso de la JVM, junto con información de registro y un vuelco de almacenamiento corto para cada registro.

El vuelco se escribe en un archivo de z/OS UNIX llamado CEEDUMP.aaaamdd.hhmmss.pid, donde aaaamdd es la fecha actual, hhmmss es la hora actual y pid es el ID del proceso actual. Las posibles ubicaciones de este archivo se describen en: "Ubicaciones de volcados de z/OS UNIX" en la página 190.

HEAPDUMP

Vuelco formateado (en forma de lista) de los objetos que se encuentran en la memoria dinámica Java.

El vuelco se escribe en un archivo de z/OS UNIX llamado HEAPDUMP.aaaamdd.hhmmss.pid.TXT, donde aaaamdd es la fecha actual, hhmmss es la hora actual y pid es el ID del proceso actual. Las posibles ubicaciones de este archivo se describen en: "Ubicaciones de volcados de z/OS UNIX" en la página 190.

Tenga en cuenta que Developer for System z proporciona un mandato de operador para desencadenar este vuelco. Consulte el capítulo "Mandatos de operador" en la publicación *Guía de configuración del host SC11-3660* (SC23-7658) para obtener más detalles.

JAVADUMP

Análisis formateado de la JVM. Contiene información de diagnóstico relacionada con la JVM y la aplicación Java, como el entorno de la aplicación, las hebras, la pila nativa, los bloqueos y la memoria.

El vuelco se escribe en un archivo de z/OS UNIX llamado JAVADUMP.aaaamdd.hhmmss.pid.TXT, donde aaaamdd es la fecha actual, hhmmss es la hora actual y pid es el ID del proceso actual. Las posibles ubicaciones de este archivo se describen en: "Ubicaciones de volcados de z/OS UNIX" en la página 190.

Tenga en cuenta que Developer for System z proporciona un mandato de operador para desencadenar este vuelco. Consulte el capítulo "Mandatos de operador" en la publicación *Guía de configuración del host SC11-3660* (SC23-7658) para obtener más detalles.

Consulte las publicaciones *Java Diagnostic Guide* (SC34-6358), para obtener más información acerca de los vuelcos de JVM, y *Language Environment Debugging Guide* (GA22-7560) para obtener información específica acerca de LE.

Ubicaciones de volcados de z/OS UNIX

La JVM comprueba cada una de las siguientes ubicaciones para ver si tienen permiso de escritura y existencia, y almacena los archivos CEEDUMP, HEAPDUMP y JAVADUMP en la primera ubicación disponible. Tenga en cuenta que debe tener suficiente espacio en disco libre para que el archivo de vuelco se escriba correctamente.

1. El directorio de la variable de entorno `_CEE_DMPTARG`, si se encuentra. Esta variable se establece en el valor `/tmp` en el archivo `rsed.envvars`. Se puede cambiar a `/dev/null` para no tener que crear los archivos de vuelco.
2. El directorio de trabajo actual, si no es el directorio raíz (`/`) y si se puede escribir en él.
3. El directorio de la variable de entorno `TMPDIR` (una variable de entorno que indica la ubicación de un directorio temporal si no es `/tmp`), si se encuentra.
4. El directorio `/tmp`.
5. Si el vuelco no se puede almacenar en ninguna de las ubicaciones mencionadas anteriormente, se enviará a `stderr`.

Rastrear

Rastreo del gestor de depuración

El rastreo del gestor de depuración está controlado por el operador del sistema, tal como se describe en "Mandatos de operador" en *Guía de configuración de host* (SC11-3660).

- Iniciar la tarea inicia DBGMGR con el parámetro `PRM=DEBUG` activa el rastreo.
- El mandato de operador **modify loglevel** le permite seleccionar el nivel de detalle deseado para los mensajes de registro.

Rastreo del supervisor de trabajos JES

El rastreo del Supervisor de trabajos JES está controlado por el operador del sistema, como se describe en la sección "Mandatos de operador" de la publicación *Guía de configuración de host* (SC11-3660).

- Al iniciar la tarea iniciada JMON con el parámetro `PRM=-TV`, se activa la modalidad verbosa (rastreo).
- Los mandatos del operador **modify trace** y **modify message** permiten seleccionar el nivel de detalle deseado para los mensajes de registro.

Rastreo RSE

Los componentes relacionados con RSE crean varios archivos de registro. La mayoría están ubicados en `userlog/$LOGNAME/`, donde `userlog` es el valor combinado de la directivas `user.log` y `DSTORE_LOG_DIRECTORY` de `rsed.envvars`, y `$LOGNAME` es el ID de usuario de inicio de sesión (en mayúsculas). Si la directiva `user.log` no tiene caracteres de comentario o no está presente, se utiliza la vía de

acceso inicial del usuario. La vía de acceso inicial se define en el segmento de seguridad OMVS del ID de usuario. Si la directiva `DSTORE_LOG_DIRECTORY` no tiene caracteres de comentario o no está presente, se añade `.eclipse/RSE/` al valor `user.log`.

La cantidad de datos escritos en `ffs*.log`, `lock.log` y `rsecomm.log` se controla mediante el mandato del operador **modify rsecommlog** o estableciendo `debug_level` en `rsecomm.properties`. Para obtener más detalles, consulte la sección "Mandatos de operador" de la publicación *Guía de configuración de host* (SC11-3660) y la sección "(Opcional) Rastreo RSE" de la publicación *Guía de configuración de host* (SC11-3660).

La creación de los archivos de registro `.dstore*` está controlada por las opciones de inicio `-DDSTORE_*` Java, como se describe en el apartado "Definición de parámetros de inicio de Java adicionales con `_RSE_JAVAOPTS`" de la publicación *Guía de configuración de host* (SC11-3660).

Nota:

- El directorio `.eclipse` y los archivos de registro `.dstore*` empiezan por un punto (`.`), lo que hace que estén ocultos. Utilice el mandato de z/OS UNIX `ls -lA` para listar los archivos y directorios ocultos. Cuando se utiliza el cliente Developer for System z, seleccione la página de preferencias **Ventana > Preferencias... > Sistemas remotos > Archivos** y habilite "Mostrar archivos ocultos".
- Los archivos de registro `.dstore*` se crean en UTF8. Utilice el mandato `iconv -f UTF8IBM-1047 .dstore*` de z/OS UNIX para visualizarlos en EBCDIC (al utilizar la página de códigos IBM-1047).
- Al contrario que todos los archivos `*.log`, los archivos de registro `.dstore*` no se eliminan automáticamente cuando se vuelve a conectar el cliente. La eliminación de estos archivos es una acción manual.

Los archivos de registro específicos del daemon RSE y de la agrupación de hebras RSE se encuentran en `daemon-home`, donde `daemon-home` es el valor de la directiva `daemon.log` de `rsed.envvars`. Si la directiva `daemon.log` no tiene caracteres de comentario o no está presente, se utiliza el directorio inicial del ID de usuario asignado a la tarea iniciada RSED. El directorio inicial se define en el segmento de seguridad OMVS del ID de usuario.

La cantidad de datos escritos en `rsedaemon.log` y `rseserver.log` se controla mediante los mandatos del operador **modify rsedaemonlog** y **modify rseserverlog** o estableciendo `debug_level` en `rsecomm.properties`. Para obtener más detalles, consulte la sección "Mandatos de operador" de la publicación *Guía de configuración de host* (SC11-3660) y la sección "(Opcional) Rastreo RSE" de la publicación *Guía de configuración de host* (SC11-3660).

`serverlogs.count`, `stderr*.log` y `stdout*.log` solamente se crean si la directiva `enable.standard.log` de `rsed.envvars` está activa, o si la función está activada dinámicamente con el mandato de operador **modify rsestandardlog on**.

Rastreo de CARMA

El usuario puede controlar la cantidad de información de rastreo generada por un servidor CARMA estableciendo el Nivel de rastreo en la pestaña de propiedades de la conexión CARMA en el cliente. Las opciones de nivel de rastreo son:

- Inhabilitar registro

- Registro de error
- Registro de aviso
- Registro informativas
- Registro de depuración

El valor predeterminado es el siguiente:

Registro de error

Para obtener más información sobre la ubicación de los archivos de registro, consulte: “Archivos de registro” en la página 182.

El programador del sistema z/OS puede controlar la cantidad de información de rastreo que genera el método de inicio CRASTART de CARMA estableciendo el valor crastart.syslog en CRASRV.properties y estableciendo el nivel de depuración para rsecomm.log en rsecomm.properties o con un mandato de operador.

Rastreo de información de retorno de errores

El siguiente procedimiento permite reunir la información necesaria para diagnosticar problemas de información de retorno de errores con procedimientos de construcción remotos. Este rastreo afectará negativamente al rendimiento y solo se debe activar cuando así lo indica el centro de soporte de IBM. En este apartado, todas las referencias a hlq se refieren al calificador de alto nivel empleado durante la instalación de Developer for System z. El valor predeterminado de la instalación es FEK, pero quizá no sea válido para su local.

1. Haga una copia de seguridad del procedimiento de compilación ELAXFC0C activo. Este procedimiento viene de forma predeterminada en el conjunto de datos hlq.SFEKSAMP, pero es posible que se haya copiado en una ubicación distinta, como SYS1.PROCLIB, según se describe en la sección "Procedimientos de construcción remota ELAXF*" de la publicación *Guía de configuración de host* (SC11-3660).
2. Cambie el procedimiento ELAXFC0C activo para que incluya la serie 'MAXTRACE' en la opción de compilación EXIT(ADEXIT(ELAXMGUX)).

```
//COBOL EXEC PGM=IGYCRCTL,REGION=2048K,
//*      PARM=('EXIT(ADEXIT(ELAXMGUX))',
//      PARM=('EXIT(ADEXIT('MAXTRACE',ELAXMGUX))',
//      'ADATA',
//      'LIB',
//      'TEST(NONE,SYM,SEP)',
//      'LIST',
//      'FLAG(I,I)'&CICS &DB2 &COMP)
```

Nota: Tendrá que duplicar los apóstrofes en torno a MAXTRACE. Ahora, la opción es: EXIT(ADEXIT('MAXTRACE',ELAXMGUX)).

3. Realice una comprobación de sintaxis remota en el programa COBOL del que desea obtener el rastreo detallado.
4. El componente SYSOUT de la salida de JES empezará enumerando los nombres de los conjuntos de datos de SIDEFILE1, SIDEFILE2, SIDEFILE3 y SIDEFILE4.

```
ABOUT TOO OPEN SIDEFILE1 - NAME = 'uid.DT021207.TT110823.M0000045.C0000000'
SUCCESSFUL OPEN SIDEFILE1 - NAME = 'uid.DT021207.TT110823.M0000045.C0000000'
ABOUT TOO OPEN SIDEFILE2 - NAME = 'uid.DT021207.TT110823.M0000111.C0000001'
SUCCESSFUL OPEN SIDEFILE2 - NAME = 'uid.DT021207.TT110823.M0000111.C0000001'
ABOUT TOO OPEN SIDEFILE3 - NAME = 'uid.DT021207.TT110823.M0000174.C0000002'
SUCCESSFUL OPEN SIDEFILE3 - NAME = 'uid.DT021207.TT110823.M0000174.C0000002'
ABOUT TOO OPEN SIDEFILE4 - NAME = 'uid.DT021207.TT110823.M0000236.C0000003'
SUCCESSFUL OPEN SIDEFILE4 - NAME = 'uid.DT021207.TT110823.M0000236.C0000003'
```

Nota: En función de los valores que tenga, SIDEFILE1 y SIDEFILE2 podrían señalar hacia una sentencia DD (SUCCESSFUL OPEN SIDEFILE1 - NAME = DD:WSEDSF1). Consulte el componente JESJCL de la salida (situado antes del componente SYSOUT) para obtener el nombre del conjunto de datos real.

```
22 //COBOL.WSEDSF1 DD DISP=MOD,  
    // DSN=uid.ERRCOB.member.SF.Z682746.XML  
23 //COBOL.WSEDSF2 DD DISP=MOD,  
    // DSN=uid.ERRCOB.member.SF.Z682747.XML
```

5. Copie estos cuatro conjuntos de datos en el PC, creando por ejemplo un proyecto COBOL local en Developer for System z y añadiendo los conjuntos de datos SIDEFILE1->4.
6. Copie los registros de trabajo JES completos en el PC, abriendo por ejemplo la salida de trabajo en Developer for System z y guardándola en el proyecto local, seleccionado **Archivo > Guardar como...**
7. Restaure el procedimiento ELAXFC0C a su estado original, ya sea deshaciendo el cambio (elimine la serie "MAXTRACE" de las opciones de compilación) o restaurando la copia de seguridad.
8. Envíe los archivos recogidos (SIDEFILE1->4 y registro de trabajo) al centro de soporte de IBM.

Bits de permiso de z/OS UNIX

Developer for System z requiere que el sistema de archivos de z/OS UNIX y algunos archivos de z/OS UNIX tengan establecidos determinados bits de permiso.

Atributo del sistema de archivos SETUID

Explorador de Sistemas remotos (RSE) es el componente de Developer for System z que proporciona servicios del núcleo como por ejemplo la conexión del cliente con el host. Debe permitírsele realizar tareas tales como crear el entorno de seguridad del usuario.

El sistema de archivos (HFS o zFS) en el que se instala Developer for System z debe estar montado con el bit de permiso SETUID (este el valor predeterminado del sistema). El hecho de montar el sistema de archivos con el parámetro NOSETUID impedirá que Developer for System z cree el entorno de seguridad del usuario y la solicitud de conexión fallará. Otros indicadores de este problema de configuración son:

- el mensaje de consola "FEK999E El módulo, fekfomvs debe estar marcado como autorizado por APF2
- El IVP de PassTicket falla con "ICH409I 282-010 ABEND DURING RACHECK PROCESSING"

Se pueden esperar errores similares (como por ejemplo los mensajes BPXP014I y BPXP015I) si los sistemas de archivos que albergan binarios Java o z/OS UNIX se montan con el parámetro NOSETUID.

Utilice el mandato **ISHELL** de TSO para listar el estado actual del bit SETUID. En el panel de ISHELL, seleccione **Sistemas de archivos > 1. Tabla de montaje...** para listar los sistemas de archivos montados. El mandato abreviado **a** mostrará los atributos del sistema de archivos seleccionado, donde el campo "Ignorar SETUID" debe ser 0.

Autorización de control de programa

Explorador de Sistemas remotos (RSE) es el componente de Developer for System z que proporciona servicios del núcleo como por ejemplo la conexión del cliente con el host. Se debe ejecutar en modalidad controlada por programa para poder realizar tareas como las de pasar al ID de usuario del cliente.

El bit de control de programa de z/OS UNIX se establece durante la instalación de SMP/E cuando es necesario, excepto para la interfaz de Java del producto de seguridad, tal como se documenta en la sección Capítulo 2, “Consideraciones relativas a la seguridad”, en la página 19. Este bit de permiso puede perderse si no lo ha conservado durante una copia manual de los directorios de Developer for System z.

Los siguientes archivos de Developer for System z deben estar controlados por programa:

- /usr/lpp/rdz/bin/
 - fekfdivp
 - fekfomvs
 - fekfrivp
- /usr/lpp/rdz/lib/
 - fekfdir.dll
 - libfekdcore.so
 - libfekfmain.so
- /usr/lpp/rdz/lib/icuc/
 - libicudata.dll
 - libicudata50.1.dll
 - libicudata50.dll
 - libicudata64.50.1.dll
 - libicudata64.50.dll
 - libicudata64.dll
 - libicuuc.dll
 - libicuuc50.1.dll
 - libicuuc50.dll
 - libicuuc64.50.1.dll
 - libicuuc64.50.dll
 - libicuuc64.dll

Utilice el mandato **ls -E** de z/OS UNIX para listar los atributos ampliados, en los que el bit de control de programa está marcado con la letra **p**, según se muestra en el ejemplo siguiente (\$ es el indicador de mandatos de z/OS UNIX):

```
$ cd /usr/lpp/rdz
$ ls -E lib/fekf*
-rwxr-xr-x -ps- 2 user      group      94208 Jul  8 12:31 lib/fekfdir.dll
```

Utilice el mandato **extattr +p** de z/OS UNIX para establecer el bit de control de programa manualmente, como se muestra en el ejemplo siguiente (\$ y # son los indicadores de mandatos de z/OS UNIX):

```
$ cd /usr/lpp/rdz
$ su
# extattr +p lib/fekf*
```



```
# exit
$ ls -E lib/fekf*
-rwxr-xr-x  -ps-  2 user      group      94208 Jul  8 12:31 lib/fekfdir.dll
```

Nota: Para poder utilizar el mandato **extattr +p**, debe tener como mínimo acceso de lectura (READ) al perfil BPX.FILEATTR.PROGCTL en la clase FACILITY del software de seguridad, o ser un superusuario (UID 0) si este perfil no está definido. Para obtener más información, consulte la publicación *UNIX System Services Planning* (GA22-7800).

Autorización de APF

Explorador de Sistemas remotos (RSE) es el componente de Developer for System z que proporciona servicios del núcleo como por ejemplo la conexión del cliente con el host. Se debe ejecutar con autorización de APF para poder realizar tareas como por ejemplo visualizar la utilización de recursos de proceso detallados.

El bit de z/OS UNIX APF se establece durante la instalación de SMP/E, cuando sea necesario. Este bit de permiso puede perderse si no lo ha conservado durante una copia manual de los directorios de Developer for System z.

Los siguientes archivos de Developer for System z deben tener autorización de APF:

- /usr/lpp/rdz/bin/
 - CRSTART
 - fekfomvs
 - fekfriwp

Utilice el mandato **ls -E** de z/OS UNIX para listar los atributos ampliados, en los que el bit de APF está marcado con la letra a, según se muestra en el ejemplo siguiente: (\$ es el indicador de mandatos de z/OS UNIX):

```
$ cd /usr/lpp/rdz
$ ls -E bin/fekfriwp
-rwxr-xr-x  aps-  2 user      group      114688 Sep 17 06:41 bin/fekfriwp
```

Utilice el mandato **extattr +a** de z/OS UNIX para establecer el bit de APF manualmente, como se muestra en el ejemplo siguiente (\$ y # son las solicitudes de z/OS UNIX):

```
$ cd /usr/lpp/rdz
$ su
# extattr +a bin/fekfriwp
# exit
$ ls -E bin/fekfriwp
-rwxr-xr-x  aps-  2 user      group      114688 Sep 17 06:41 bin/fekfriwp
```

Nota: Para poder utilizar el mandato **extattr +a**, debe tener como mínimo acceso de lectura (READ) al perfil BPX.FILEATTR.APF en la clase FACILITY del software de seguridad, o ser un superusuario (UID 0) si este perfil no está definido. Para obtener más información, consulte la publicación *UNIX System Services Planning* (GA22-7800).

Bit de permanencia

Algunos de los servicios opcionales de Developer for System z requieren que los módulos de carga de MVS estén disponibles para z/OS UNIX. Esta operación se realiza creando un apéndice (un archivo ficticio) en z/OS UNIX con el bit de

"permanencia" activado. Al ejecutar el apéndice, z/OS UNIX buscará un módulo de carga de MVS con el mismo nombre y ejecutará el módulo de carga en su lugar.

El bit de permanencia de z/OS UNIX se establece durante la instalación de SMP/E, cuando es necesario. Estos bits de permiso pueden perderse si no los ha conservado durante una copia manual de los directorios de Developer for System z.

Los siguientes archivos de Developer for System z deben tener el bit de permanencia activado:

- /usr/lpp/rdz/bin/
 - AZUTSTRN
 - CRASTART

Utilice el mandato **ls -l** de z/OS UNIX para listar los atributos ampliados, en los que el bit de control de programa está marcado con la letra **t**, según se muestra en el ejemplo siguiente (\$ es el indicador de mandatos de z/OS UNIX):

```
$ cd /usr/lpp/rdz
$ ls -l bin/CRA*
-rwxr-xr-t  2 user      group          71 Jul  8 12:31 bin/CRASTART
```

Utilice el mandato **chmod +t** de z/OS UNIX para establecer el bit de permanencia manualmente, como se muestra en el ejemplo siguiente (\$ y # son indicadores de mandatos de z/OS UNIX):

```
$ cd /usr/lpp/rdz
$ su
# chmod +t bin/CRA*
# exit
$ ls -l bin/CRA*
-rwxr-xr-t  2 user      group          71 Jul  8 12:31 bin/CRASTART
```

Nota: Para poder utilizar el mandato **chmod**, debe tener como mínimo acceso de lectura (READ) al perfil SUPERUSER.FILESYS.CHANGEPERMS en la clase UNIXPRIV del software de seguridad, o ser un superusuario (UID 0) si este perfil no está definido. Para obtener más información, consulte la publicación *UNIX System Services Planning* (GA22-7800).

Puertos TCP/IP reservados

Con el mandato **netstat** (TSO o z/OS UNIX) puede obtener una visión general de los puertos que se utilizan en este momento. Los datos de salida de este mandato se parecerán a los del ejemplo siguiente. Los puertos utilizados son el último número (a continuación de "..") en la columna "Socket local". Como estos puertos ya se están utilizando, no se pueden utilizar para la configuración de Developer for System z.

IPv4

MVS TCP/IP	NETSTAT	CS VxRy	TCP/IP	Nombre	TCP/IP:	TCP/IP	16:36:42
ID us.	Conexión	Socket Local		Socket	Foráneo		Estado
-----	----	-----		-----			-----
BPX0INIT	00000014	0.0.0.0..10007		0.0.0.0..0			Escucha
INETD4	0000004D	0.0.0.0..512		0.0.0.0..0			Escucha
RSED	0000004B	0.0.0.0..4035		0.0.0.0..0			Escucha
JMON	00000038	0.0.0.0..6715		0.0.0.0..0			Escucha

IPv6

```

MVS TCP/IP NETSTAT CS VxRy      Nombre TCPIP: TCPIP      12:46:25
ID usuario  Conexión  Estado
-----
BPX0INIT    00000018  Escucha
Socket local:  0.0.0.0..10007
Socket foráneo: 0.0.0.0..0
INETD4      00000046  Escucha
Socket local:  0.0.0.0..512
Socket foráneo: 0.0.0.0..0
RSED        0000004B  Escucha
Socket local:  0.0.0.0..4035
Socket foráneo: 0.0.0.0..0
JMON        00000037  Escucha
Socket local:  0.0.0.0..6715
Socket foráneo: 0.0.0.0..0

```

Otra posible limitación son los puertos TCP/IP reservados. Hay dos lugares comunes en los que se reservan puertos TCP/IP:

- **PROFILE.TCPIP**

Este es el conjunto de datos al que hace referencia la sentencia DD PROFILE de la tarea iniciada TCP/IP, que a menudo se llama SYS1.TCPPARMS(TCPPROF).

- PORT: Reserva un puerto para los nombres de trabajo especificados.
- PORTRANGE: Reserva un rango de puertos para los nombres de trabajo especificados.

Consulte la publicación *Communications Server: IP Configuration Guide* (SC31-8775) para obtener más información acerca de estas sentencias.

- **SYS1.PARMLIB(BPXPRMxx)**

- INADDRANYPORT: Especifica el número de puerto inicial del rango de números de puerto que el sistema reserva para utilizar con los enlaces PORT 0, INADDR_ANY. Este valor solo se necesita para CINET (varias pilas TCP/IP activas en un único host).
- INADDRANYCOUNT: Especifica el número de puertos que el sistema reserva, empezando por el número de puerto especificado en el parámetro INADDRANYPORT. Este valor solo se necesita para CINET (varias pilas TCP/IP activas en un único host).

Consulte las publicaciones *UNIX System Services Planning* (GA22-7800) y *MVS Initialization and Tuning Reference* (SA22-7592) para obtener más información sobre estas sentencias.

Para obtener una lista de estos puertos reservados, se puede utilizar el mandato **netstat portl** (TSO o z/OS UNIX), que crea una salida parecida a la del ejemplo siguiente:

```

MVS TCP/IP NETSTAT CS VxRy      Nombre TCPIP: TCPIP      17:08:32
NºPto Prot Usuario  Dstivos  Rango      Dirección IP
-----
00007 TCP  MISCSERV DA
00009 TCP  MISCSERV DA
00019 TCP  MISCSERV DA
00020 TCP  OMVS     D
00021 TCP  FTPD1    DA
00025 TCP  SMTP     DA
00053 TCP  NAMESRV  DA
00080 TCP  OMVS     DA
03500 TCP  OMVS     DAR      03500-03519
03501 TCP  OMVS     DAR      03500-03519

```

Consulte la publicación *Communications Server: IP System Administrator's Commands* (SC31-8781) para obtener más información acerca del mandato **NETSTAT**.

Nota: El mandato **NETSTAT** solo muestra la información definida en **PROFILE.TCPIP**, que debe solapar las definiciones de **BPXPRMxx**. En caso de duda o problemas, compruebe el miembro **parmlib BPXPRMxx** para verificar los puertos que se reservan aquí.

Tamaño del espacio de direcciones

Para el daemon RSE, que es un proceso z/OS UNIX Java se necesita una región de gran tamaño para efectuar sus funciones. Por lo tanto, es importante establecer límites de almacenamiento grandes para los espacios de direcciones de OMVS.

Requisitos de JCL de inicio

El daemon RSE se inicia mediante JCL utilizando **BPXBATSL**, cuyo tamaño de región debe ser 0.

Limitaciones establecidas en **SYS1.PARMLIB(BPXPRMxx)**

Establezca que **MAXASSIZE** en **SYS1.PARMLIB(BPXPRMxx)**, que define el tamaño de región (proceso) de espacio de direcciones OMVS predeterminado, en 2G. Es el tamaño máximo permitido. Este es un límite a escala del sistema y, por ello, está activo para todos los espacios de direcciones z/OS UNIX. Si no desea este límite, puede establecer el límite únicamente para Developer for System z en el software de seguridad.

Este valor se puede comprobar y establecer dinámicamente (hasta la próxima IPL) con los siguientes mandatos de consola, como se describe en el manual *MVS System Commands* (GC28-1781):

1. **DISPLAY OMVS,0**
2. **SETOMVS MAXASSIZE=2G**

Limitaciones almacenadas en el perfil de seguridad

Compruebe **ASSIZEMAX**, en el segmento OMVS del ID de usuario del daemon, y establézcalo en 2147483647 o, preferiblemente, en **NONE** para que utilice el valor **SYS1.PARMLIB(BPXPRMxx)**.

Con RACF, este valor se puede comprobar y establecer con los siguientes mandatos TSO, como se describe en el manual *Security Server RACF Command Language Reference* (SA22-7687):

1. **LISTUSER userid NORACF OMVS**
2. **ALTUSER userid OMVS(NOASSIZEMAX)**

Limitaciones aplicadas por la rutinas de salida del sistema

Asegúrese de que no permite que las rutinas de salida **IEFUSI** o **IEALIMIT** del sistema controlen los tamaños de las regiones del espacio de direcciones de OMVS. Una manera posible de lograrlo es escribiendo **SUBSYS(OMVS,NOEXITS)** en el código de **SYS1.PARMLIB(SMFPRMxx)**.

Los valores de **SYS1.PARMLIB(SMFPRMxx)** se pueden comprobar y activar con los siguientes mandatos de consola, como se describe en el manual *MVS System Commands* (GC28-1781):

1. **DISPLAY SMF,0**
2. **SET SMF=xx**

Limitaciones para el direccionamiento de 64 bits

La palabra clave **MEMLIMIT** en **SYS1.PARMLIB(SMFPRMxx)** limita el almacenamiento virtual que una tarea de 64 bits puede asignar por encima de la barra de 2GB. Al contrario que el parámetro **REGION** en **JCL**, **MEMLIMIT=0M** significa que el proceso no puede utilizar almacenamiento virtual por encima de la barra.

Si no se especifica **MEMLIMIT** en **SMFPRMxx**, el valor predeterminado es **0M**, con lo que las tareas están limitadas a los 2GB (31 bits) por debajo de la barra. El valor predeterminado cambió en **z/OS 1.10** a **2G**, permitiendo que las tareas de 64 bits utilicen hasta 4GB (los 2GB por debajo de la barra y los 2GB por encima de la barra otorgados por **MEMLIMIT**).

Los valores de **SYS1.PARMLIB(SMFPRMxx)** se pueden comprobar y activar con los siguientes mandatos de consola, como se describe en el manual *MVS System Commands* (GC28-1781):

1. **DISPLAY SMF,0**
2. **SET SMF=xx**

MEMLIMIT también se puede especificar como parámetro en una tarjeta **EXEC** en **JCL**. Si no se especifica ningún parámetro **MEMLIMIT**, el valor predeterminado es el valor definido por **SMF**, excepto cuando se especifica **REGION=0M**, en cuyo caso el valor predeterminado es **NOLIMIT**.

Información variada

Terminación anómala de espacio B37 de retorno de errores

Cuando un usuario selecciona el retorno de errores durante una acción de compilación, **Developer for System z** crea varios conjuntos de datos temporales. Si uno de estos conjuntos de datos se queda sin espacio, los trabajos de compilación finalizan con una terminación anómala de espacio **B37-04**.

Ajuste la asignación de espacio en **FEK.SFEKPROC(FEKFERRF)** si los usuarios se encuentran con este problema. El valor predeterminado es **SPACE(200,40) TRACKS**.

Límites del sistema

SYS1.PARMLIB(BPXPRMxx) define muchas limitaciones relacionadas con **z/OS UNIX**, que se podrían alcanzar cuando hay varios clientes **Developer for System z** activos. La mayoría de los valores **BPXPRMxx** se pueden cambiar dinámicamente con los mandatos de consola **SETOMVS** y **SET OMVS**.

Utilice el mandato de consola **SETOMVS LIMMSG=ALL** para que **z/OS UNIX** muestre los mensajes de consola (**BPXI040I**) cuando se está a punto de alcanzar alguno de los límites de **BPXPRMxx**.

Conexión rehusada

Cada conexión **RSE** inicia varios procesos que están permanentemente activos. Se pueden rehusar nuevas conexiones debido al límite establecido en **SYS1.PARMLIB(BPXPRMxx)** sobre la cantidad de procesos, especialmente cuando los usuarios comparten un mismo **UID** (como cuando se utiliza el segmento **OMVS** predeterminado).

- El límite por cada **UID** se establece mediante la palabra clave **MAXPROCUSER**, y su valor predeterminado es 25.

- El límite a escala del sistema se establece mediante la palabra clave MAXPROCSYS, y su valor predeterminado es 200.

Otra fuente de conexiones rehusadas es el límite de la cantidad de espacios de direcciones z/OS activas y de usuarios de z/OS UNIX activos.

- La cantidad máxima de IDs de espacios de direcciones (ASID) se define en SYS1.PARMLIB(IEASYSxx) con la palabra clave MAXUSER, y su valor predeterminado es 255.
- La cantidad máxima de ID de usuarios (UID) de z/OS UNIX se define en SYS1.PARMLIB(BPXPRMxx) con la palabra clave MAXUIDS, y su valor predeterminado es 200.

OutOfMemoryError

Una agrupación de hebras RSE puede fallar con la anotación de un mensaje OutOfMemoryError. Este error está relacionado con el tamaño de almacenamiento dinámico Java y puede producirse si los usuarios activos en esta agrupación de hebras utilizan más recursos de los previstos. A continuación se proporcionan las causas comunes de este error:

- Expansión de filtros de conjuntos de datos grandes en el Explorador de sistemas remotos
- Apertura de PDS(E) con una gran cantidad de miembros
- Apertura de miembros o archivos secuenciales grandes

Para solucionar este problema, puede realizar las acciones siguiente:

- Aumentar la directiva -Xmx en rsed.envvars, ya que controla el tamaño máximo de almacenamiento dinámico Java. Tenga en cuenta que el almacenamiento dinámico Java debe caber dentro de los límites de espacios de direcciones.
- Disminuir la directiva -Dmaximum.clients en rsed.envvars, ya que controla cuántos usuarios se pueden colocar en una única agrupación de hebras (y de este modo compartir un único almacenamiento dinámico Java).

Emulador de conexión de host

- El emulador de Host Connect utiliza telnet TN3270 (no el servidor RSE) para establecer conexión con el host.
- Cuando se utiliza telnet seguro (SSL) y se trabaja con certificados no firmados por una CA conocida, cada cliente debe añadir el certificado de la CA a su lista de CA de confianza del emulador de conexión de host.
- La opción NOSNAEXT de TELNETPARMS de TCP/IP podría ser necesaria para inhabilitar las extensiones funcionales de SNA. Si se especifica NOSNAEXT, el servidor telnet TN3270 no negocia las funciones de resolución de contiendas y detección de SNA.

Capítulo 13. Configurar SSL y autenticación de X.509

Esta sección se propone ayudarle a resolver algunos problemas comunes que pueden surgir al configurar la capa de sockets segura (SSL) o durante la comprobación o modificación de una configuración existente. Esta sección también facilita una configuración de ejemplo para admitir que los usuarios se autenticuen con un certificado X.509.

Que una comunicación sea segura implica asegurarse de que su interlocutor sea la persona que afirma ser y transmitir información de tal manera que a las otras personas les resulte difícil interceptar los datos y leerlos. SSL proporciona capacidad para ello en una red TCP/IP. Funciona utilizando certificados digitales para identificarse a sí mismo, y un protocolo de claves públicas para cifrar la comunicación. Consulte la publicación *Security Server RACF Security Administrator's Guide* (SA22-7683) para obtener más información acerca de los certificados digitales y el protocolo de claves públicas utilizado por SSL.

Las acciones necesarias para configurar las comunicaciones SSL para Developer for System z variarán según el local, en función de las necesidades exactas, del método de comunicación RSE empleado y de lo que ya esté disponible en el local.

En esta sección clonaremos las definiciones actuales de RSE para poder tener una segunda conexión del daemon RSE que utilice SSL. También crearemos nuestros propios certificados de seguridad para que los utilicen diferentes componentes de la conexión RSE.

- “Elegir entre SSL o TLS como método de cifrado” en la página 202
- “Decida dónde desea almacenar los certificados y claves privadas” en la página 202
- “Crear un anillo de claves con RACF” en la página 203
- “Clonar la configuración RSE existente” en la página 205
- “Actualizar rsed.envvars para habilitar la coexistencia” en la página 205
- “Actualizar ssl.properties para habilitar la SSL” en la página 206
- “Activar la SSL creando un daemon RSE nuevo” en la página 206
- “Probar la conexión” en la página 207
- “(Opcional) Añadir soporte de autorización al cliente de X.509” en la página 210
- “(Opcional) Crear una base de datos de claves con gskkyman” en la página 210
- “(Opcional) Crear un almacén de claves con keytool” en la página 213

A lo largo de esta sección se utiliza un convenio de denominación uniforme:

- Certificado: rdzrse
- Almacenamiento de claves y certificados: rdzssl.*
- Contraseña: rsessl
- ID de usuario del daemon : stcrse

En algunas tareas que se describen en las secciones siguientes, se espera que esté activo en z/OS UNIX. Para ello, emita el mandato TSO **OMVS**. Utilice el mandato **exit** para volver a TSO.

Elegir entre SSL o TLS como método de cifrado

La variable `DSTORE_SSL_ALGORITHM` de la directiva `_RSE_JAVA_OPTS` de `rsed.envvars` permite elegir entre el método de cifrado SSL y su eventor TLS (seguridad de la capa de transporte), tal como se documenta en "Definir parámetros de inicio Java adicionales con `_RSE_JAVA_OPTS`" en la *Guía de configuración de host* SC11-3660 (SC23-7658).

Decida dónde desea almacenar los certificados y claves privadas

Los certificados de identidad y las claves de cifrado/descifrado que SSL emplea se almacenan en un archivo de claves. Existen distintas implementaciones de este archivo de claves, en función del tipo de aplicación.

sin embargo, todas las implementaciones siguen el mismo principio. Un mandato genera un par de claves (una clave pública y una clave privada asociada). Este envuelve luego la clave pública en un certificado X.509 autofirmado, que se almacena como cadena de certificados de un solo elemento. Esta cadena de certificados y la clave privada se almacenen como una entrada (identificado por un alias) en un archivo de claves.

El daemon RSE es una aplicación SSL del sistema y utiliza un archivo de base de datos de claves. Esta base de datos de claves puede ser un archivo físico creado por `gskkyman` o un anillo de claves gestionado por el software de seguridad compatible con SAF (por ejemplo, por RACF). El servidor RSE (que se inicia mediante el daemon) es una aplicación SSL Java y utiliza un archivo de almacén de claves creado por `keytool` o bien un anillo de claves gestionado por su software de seguridad.

Tabla 44. Mecanismos de almacenamiento de certificados de SSL

Almacenamiento de certificados	Creado y gestionado por	Daemon RSE	servidor RSE
anillo de claves	producto de seguridad compatible con SAF	soportado	soportado
base de datos de claves	<code>gskkyman</code> de z/OS UNIX	soportado	/
almacén de claves	<code>Keytool</code> de Java	/	soportado

Para establecer la conexión por medio de SSL, necesitamos tanto el almacén de claves como la base de datos de claves, ya sea como un archivo de z/OS UNIX o como un anillo de claves compatible con SAF:

- almacén de claves (RACF o `keytool`)
- base de datos de claves (RACF o `gskkyman`)

Nota:

- Los anillos de claves compatibles con SAF son el método preferido para gestionar certificados.
- Se puede utilizar un certificado compartido si el daemon RSE y el servidor RSE utilizan el mismo método de gestión de certificados.
- El daemon RSE debe ejecutarse controlado por programa. Utilizar el SSL del sistema implica que `SYS1.SIEALNKE` debe estar controlado por programa por el software de seguridad.

- Para poder ejecutar una aplicación SSL del sistema (conexión del daemon), SYS1.SIEALNKE debe estar en LINKLIST o en STEPLIB. Si prefiere el método de STEPLIB, añada la siguiente sentencia al final de rsed.envvars.

STEPLIB=\$STEPLIB:SYS1.SIEALNKE

Sin embargo, tenga en cuenta lo siguiente:

- El hecho de utilizar STEPLIB en z/OS UNIX afecta negativamente al rendimiento.
- Si una biblioteca de STEPLIB tiene autorización APF, todas deben tener autorización. Las bibliotecas pierden su autorización APF si se mezclan con bibliotecas sin autorización en STEPLIB.
- SSL del sistema utiliza el recurso de servicio criptográfico integrado (ICSF) si está disponible. ICSF proporciona soporte criptográfico por hardware, que se utilizará en lugar de los algoritmos de software de SSL del sistema. Consulte la publicación *System SSL Programming* (SC24-5901) para obtener más información.

Para obtener información sobre RACF y certificados digitales, consulte *Security Server RACF - Guía del administrador de seguridad* (SA22-7683). La documentación relativa a gskkyman se encuentra en la publicación *System SSL Programming* (SC24-5901) y la documentación de keytool está disponible en el sitio <http://java.sun.com/j2se/1.5.0/docs/tooldocs/solaris/keytool.html>.

Crear un anillo de claves con RACF

No realice este paso si utiliza gskkyman para crear la base de datos de claves del daemon RSE y keytool para crear el almacén de claves del servidor RSE.

El mandato **RACDCERT** instala y mantiene claves privadas y certificados en RACF. RACF permite gestionar múltiples claves privadas y certificados en forma de grupo. Estos grupos se llaman anillos de claves.

Los certificados pueden ser autofirmados o estar firmados por una autoridad certificadora (CA). Un certificado firmado por una CA implica que la CA garantiza que el propietario del certificado es quien dice ser. El proceso de firma añade las credenciales de la CA (otro certificado) al certificado, constituyendo una cadena de certificados de varios elementos.

Al utilizar un certificado firmado por una CA, puede evitar las preguntas de validación de la confianza realizadas por el cliente de Developer for System z, si para el cliente la CA ya es de confianza.

Para obtener detalles sobre el mandato **RACDCERT**, consulte *Security Server RACF Command Language Reference* (SA22-7687).

```
# permita al daemon RSE acceder a los certificados
RDEFINE FACILITY IRR.DIGTCERT.LIST UACC(NONE)
RDEFINE FACILITY IRR.DIGTCERT.LISTRING UACC(NONE)
PERMIT IRR.DIGTCERT.LIST CLASS(FACILITY) ACCESS(READ) ID(stcrse)
PERMIT IRR.DIGTCERT.LISTRING CLASS(FACILITY) ACCESS(READ) ID(stcrse)
```

```
# renueve para que los cambios sean visibles
SETROPTS RACLIST(FACILITY) REFRESH
```

```
# cree un certificado autofirmado
RACDCERT ID(stcrse) GENCERT SUBJECTSDN(CN('rdz rse ssl') +
OU('rdz') O('IBM') L('Raleigh') SP('NC') C('US')) +
NOTAFTER(2017-05-21) WITHLABEL('rdzrse') KEYUSAGE(HANDSHAKE)
```

```
# (opcional) pasos adicionales necesarios para utilizar un certificado con firma
```

```

# 1. cree una solicitud de firma para el certificado autofirmado
RACDCERT ID(stcrse) GENREQ (LABEL('rdzrse')) DSN(dsn)
# 2. envíe la solicitud de firma a la CA de su elección
# 3. compruebe si las credenciales de la CA (también un certificado)
    ya se conocen
RACDCERT CERTAUTH LIST
# 4. marque el certificado de autoridad emisora de certificados como fiable
RACDCERT CERTAUTH ALTER(LABEL('CA cert')) TRUST
#     o añada el certificado de autoridad emisora de certificados a la base
#     de datos
RACDCERT CERTAUTH ADD(dsn) WITHLABEL('CA cert') TRUST
# 5. añada el certificado firmado a la base de datos;
#     de esta forma se sustituye el autofirmado
RACDCERT ID(stcrse) ADD(dsn) WITHLABEL('rdzrse') TRUST
#     NO suprima el certificado autofirmado antes de sustituirlo.
#     Si lo hace, perderá la clave privada que acompaña al certificado,
#     lo que hace que el certificado no sirva para nada.

RACDCERT ID(stcrse) ADDRING(rdzssl.racf)
RACDCERT ID(stcrse) CONNECT(LABEL('rdzrse') RING(rdzssl.racf) +
    DEFAULT USAGE(PERSONAL))

# paso adicional necesario para utilizar un certificado firmado
# 6. añada el certificado de autoridad emisora de certificados al conjunto de
    claves
RACDCERT ID(stcrse) CONNECT(CERTAUTH LABEL('CA cert') +
    RING(rdzssl.racf))
# renueve para que los cambios sean visibles
SETROPTS RACLIST(DIGTCERT) REFRESH

```

En el ejemplo anterior se empieza por crear los perfiles necesarios y permitir que el ID de usuario STCRSE tenga acceso a los anillos de claves y a los certificados propiedad de ese ID de usuario. El ID de usuario que se utilice debe coincidir con el ID de usuario utilizado para ejecutar para el daemon RSE SSL. El próximo paso consiste en crear un certificado autofirmado con la etiqueta rdzrse . No se necesita ninguna contraseña. Luego, este certificado se añade a un anillo de claves recién creado (rdzssl.racf). Igual que con el certificado, tampoco se necesita una contraseña para el anillo de claves. También se enumeran los pasos necesarios para utilizar un certificado con firma.

Tenga en cuenta que el certificado de CA utilizado para firmar su certificado puede, por otro lado, ser firmado por otro certificado de CA, de mayor nivel. Si eso sucediera, el certificado de CA también se debe añadir al conjunto de claves. Este proceso se repite hasta que el certificado de CA de mayor nivel es un certificado de CA raíz, que siempre es un certificado autofirmado.

El resultado puede verificarse con las opciones list y listing:

```

RACDCERT ID(stcrse) LIST
Información de certificado digital para el usuario STCRSE:

Etiqueta: rdzrse
ID de certificado: 2QjW10Xi0sXZ1aaEqZmihUBA
Estado: TRUST
Fecha inicial: 2007/05/24 00:00:00
Fecha final: 2017/05/21 23:59:59
Número de serie:
    >00<
Nombre del emisor:
    >CN=my CA.OU=rdz.0=IBM.L=Raleigh.SP=NC.C=US<
Nombre del sujeto:
    >CN=rdz rse ssl.OU=rdz.0=IBM.L=Raleigh.SP=NC.C=US<
Tipo de clave privada: Non-ICSF
Tamaño de clave privada: 1024

```

```
Asociaciones de anillo:
  Propietario de anillo: STCRSE
  Anillo:
    >rdzssl.racf<

RACDCERT ID(stcrse) LISTRING(rdzssl.racf)
Digital ring information for user STCRSE:
```

```
Ring:
  >rdzssl.racf<
Certificate Label Name      Cert Owner      USAGE      DEFAULT
-----
rdzrse                      ID(STCRSE)      PERSONAL    YES
CA cert                     CERTAUTH        CERTAUTH    NO
```

Clonar la configuración RSE existente

En este paso se crea una nueva instancia de los archivos de configuración de RSE para que la configuración de SSL pueda ejecutarse en paralelo con las existentes. En los mandatos que siguen se presupone que los archivos de configuración se encuentran en `/etc/rdz/`, que es la ubicación predeterminada utilizada en la sección "Configuración de personalización" de la publicación *Guía de configuración de host* (SC11-3660).

```
$ cd /etc/rdz
$ mkdir ssl
$ cp rsed.envvars ssl
$ cp ssl.properties ssl
$ ls ssl
rsed.envvars    ssl.properties
```

Los mandatos z/OS UNIX que figuran en el ejemplo anterior crean un subdirectorio llamado `ssl` y lo llenan con los archivos de configuración que requieren cambios. Podemos compartir el resto de archivos de configuración, el directorio de instalación y los componentes de MVS, puesto que no son específicos de la SSL.

Reutilizando la mayor parte de los archivos de configuración existente, podemos centrarnos en los cambios que son realmente necesarios para la configuración de la SSL y evitar tener que volver a realizar la configuración de RSE completa. (Por ejemplo, podemos ahorrarnos definir una nueva ubicación para `ISPF.conf`.)

Actualizar `rsed.envvars` para habilitar la coexistencia

Hasta el momento, las definiciones son una copia exacta de la configuración actual, lo que implica que los registros del daemon RSE nuevo se superponen a los archivos de registro del servidor actual. RSE también necesita saber dónde encontrar los archivos de configuración que no se han copiado en el directorio `ssl`. Ambas emisiones pueden direccionarse por cambios sin importancia a `rsed.envvars`.

```
$ oedit /etc/rdz/ssl/rsed.envvars
-> cambiar: _RSE_RSED_PORT=4047
-> cambiar: -Ddaemon.log=/var/rdz/logs/ssl
-> cambiar: -Duser.log=/var/rdz/logs/ssl
-> añadir al FINAL:
# -- NECESARIO PARA ENCONTRAR LOS ARCHIVOS DE CONFIGURACIÓN RESTANTES
CFG_BASE=/etc/rdz
CLASSPATH=.:$CFG_BASE:$CLASSPATH
# --
```

Los cambios del ejemplo anterior definen una nueva ubicación de registro (que el daemon RSE creará en caso que la ubicación de registro no exista). Los cambios también actualizan la CLASSPATH de manera que los procesos RSE de la SSL buscarán los archivos de configuración primero en el directorio actual (/etc/rdz/ssl) y luego en el directorio original (/etc/rdz).

Actualizar ssl.properties para habilitar la SSL

Actualizando ssl.properties el RSE sigue instrucciones para empezar a utilizar la comunicación cifrada de la SSL.

```
$ oedit /etc/rdz/ssl/ssl.properties
-> cambiar: enable_ssl=true
-> descomentar y cambiar: daemon_keydb_file=rdzssl.racf
-> descomentar y cambiar: daemon_key_label=rdzrse
-> descomentar y cambiar: server_keystore_file=rdzssl.racf
-> descomentar y cambiar: server_keystore_label=rdzrse
-> descomentar y cambiar: server_keystore_type=JCERACFKS
```

Los cambios del ejemplo siguiente habilitan la SSL e indican al daemon RSE y al servidor RSE que el certificado (compartido) está almacenado bajo la etiqueta rdzrse en el anillo de claves rdzssl.racf. La palabra clave JCERACFKS indica al servidor RSE que se está utilizando un anillo de claves compatible con SAF como almacén de claves.

Tenga en cuenta que el SSL del sistema (usado por el daemon) siempre utiliza ICSF, la interfaz al hardware de cifrado de System z, si está disponible. Para poder compartir las definiciones de daemon con el servidor cuando se utiliza ICSF, server_keystore_type JCECCARACFKS debe especificarse. Aquí, el conjunto de claves compatible con SAF también se utiliza como almacén de claves para las claves públicas, pero la clave privada se almacena en ICSF. Según se muestra en la documentación *Cryptographic Services ICSF Administrator's Guide* (SA22-7521), ICSF utiliza los perfiles en las clases de seguridad CSFKEYS y CSFSERV para controlar quién puede utilizar servicios y claves criptográficas.

Activar la SSL creando un daemon RSE nuevo

Como se ha indicado anteriormente, vamos a crear una segunda conexión que utilizará SSL, lo que implica crear un daemon RSE. El daemon RSE puede ser una tarea iniciada o un trabajo de usuario. Utilizaremos el método del trabajo de usuario para la configuración inicial (prueba). En las instrucciones que siguen se presupone que el JCL de ejemplo se encuentra en FEK.#CUST.PROCLIB(RSED), que es la ubicación predeterminada utilizada en la sección "Configuración de personalización" de la publicación *Guía de configuración de host* (SC11-3660):

1. Cree un miembro FEK.#CUST.PROCLIB(RSEDSSL) y cópielo en el JCL de ejemplo FEK.#CUST.PROCLIB(RSED).
2. Personalice RSEDSSL añadiendo una tarjeta de trabajo al principio y una sentencia exec al final. Proporcione también la ubicación de los archivos de configuración relacionados con SSL (/etc/rdz/ssl), como se muestra en ejemplo de código siguiente. Observe que aplicamos la utilización del ID de usuario STCRSE, ya que a este ID de usuario se le ha otorgado autorización de acceso a los certificados y los anillos de claves en el paso anterior.

```

//RSEDSSL JOB CLASS=A,MSGCLASS=A,MSGLEVEL=(1,1),USER=STCRSE
//*
/* RSE DAEMON - SSL
/*
//RSED      PROC TMPDIR=,
//          PORT=,
//          IVP=,                * 'IVP' para hacer una prueba de IVP
//          CNFG='/etc/rdz/ssl',
//          HOME='/usr/lpp/rdz'
/*
//RSED      EXEC PGM=BPXBATSL,REGION=0M,TIME=NOLIMIT,
// PARM='PGM &HOME./bin/rsed.sh &IVP -C&CNFG -P&PORT -T&TMPDIR'
//STDOUT    DD SYSOUT=*
//STDERR    DD SYSOUT=*
//          PEND
/*
//RSED      EXEC RSED
/*

```

Figura 36. RSEDSSL - Trabajos de usuario del daemon RSE para SSL

Nota: El ID de usuario asignado al trabajo RSEDSSL debe tener las mismas autorizaciones que el daemon RSE original. El perfil FACILITY BPX.SERVER y el perfil de PTKTDATAIRRPAUTH.FEKAPPL.* son aquí elementos de claves.

Probar la conexión

La configuración del host de SSL está completa y el daemon RSE para SSL puede iniciarse sometiéndolo al trabajo FEK.#CUST.PROCLIB(RSEDSSL) creado anteriormente.

Ahora, puede probarse la configuración nueva conectándose con el cliente de Developer for System z. Dado que hemos creado una nueva configuración (clonando la existente) para que SSL la utilice, hay que definir una nueva conexión en el cliente utilizando el puerto 4047 para el daemon RSE.

Al conectarse, el host y el cliente se iniciarán con algún establecimiento de enlace para poder configurar una vía de acceso segura. Parte del establecimiento de enlace es el intercambio de certificados. El cliente Developer for System z, si no reconoce el certificado del host o la CA que lo ha firmado, el cliente de Developer for System z preguntará al usuario si el certificado es de confianza.

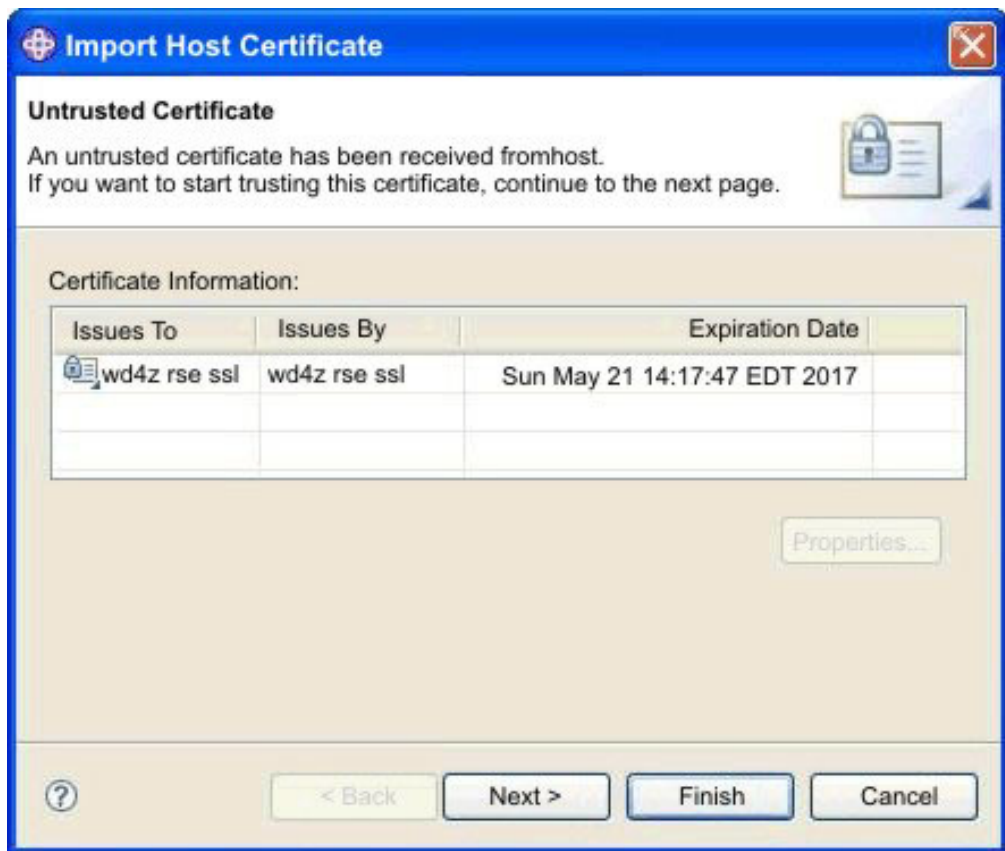


Figura 37. Diálogo Importar certificado de host

Con el botón Finalizar, el usuario puede aceptar este certificado como de confianza, después de lo cual continuará la inicialización de la conexión.

Nota: El daemon RSE y el servidor RSE pueden utilizar dos ubicaciones de certificado diferentes, generando dos certificados distintos y, por consiguiente, dos confirmaciones.

Una vez reconocido el certificado ante el cliente, este diálogo ya no vuelve a aparecer. La lista de certificados de confianza se puede gestionar seleccionando **Ventana > Preferencias... > Sistemas Remotos > SSL**, con lo cual aparece el diálogo:

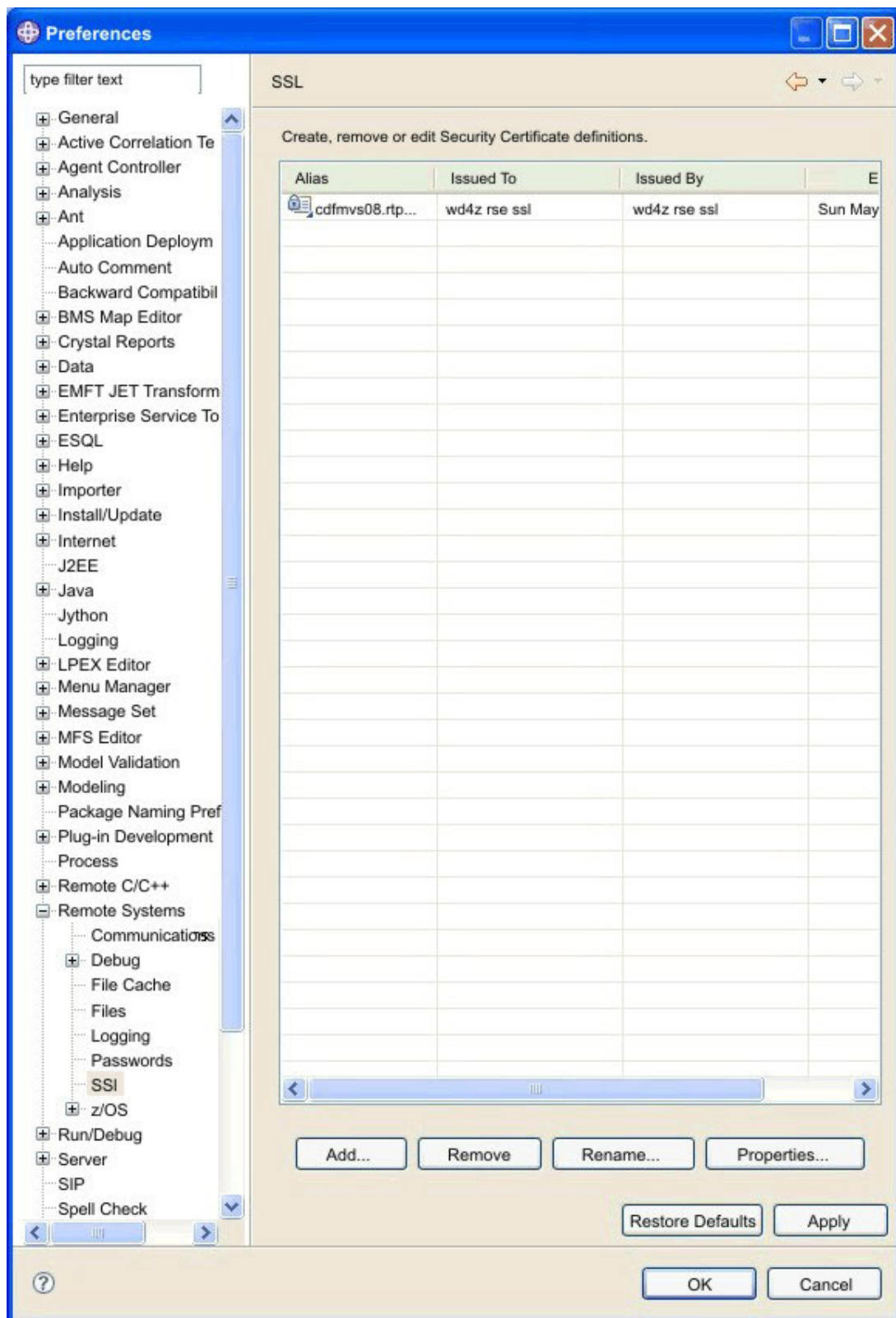


Figura 38. Diálogo Preferencias - SSL

Si la comunicación SSL falla, el cliente devolverá un mensaje de error. Hay más información en los distintos archivos de registro del usuario y del servidor, como

se describe en: “Daemon RSE y registro de la agrupaciones de hebras” en la página 184 y “Registro de usuario de RSE” en la página 185.

(Opcional) Añadir soporte de autorización al cliente de X.509

El daemon RSE admite que los usuarios se autenticuen con un certificado X.509. El uso de una comunicación cifrada con SSL es un requisito previo para utilizar esta función, dado que es una extensión de la autenticación de host con un certificado utilizado en SSL.

Hay varias formas de realizar la autorización de certificados para un usuario, tal como se describe en “Autenticación de cliente mediante certificados X.509” en la página 32. Los siguientes pasos describen la configuración necesaria para soportar el método por el cual su software de seguridad autentica el certificado mediante la ampliación de certificado HostIdMappings.

1. Cambie el certificado que identifica a la autoridad certificadora (CA) utilizada para la firma del certificado del cliente por un certificado de CA de confianza total. Aunque el estado TRUST basta para la validación de un certificado, se realiza el cambio a HIGHTRUST porque se utiliza para la parte de autenticación de certificados del proceso de inicio de sesión.
`RACDCERT CERTAUTH ALTER(LABEL('HighTrust CA')) HIGHTRUST`
2. Añada el certificado de CA al anillo de claves, `rdzssl.racf`, de manera que esté disponible para validar los certificados de clientes.

```
RACDCERT ID(stcrse) CONNECT(CERTAUTH LABEL('HighTrust CA') +  
RING(rdzssl.racf))
```

Esta acción concluye la configuración del software de seguridad para el certificado de CA.

3. Defina un recurso (formato `IRR.HOST.hostname`) en la clase `SERVAUTH` para el nombre de host, `CDFMVS08.RALEIGH.IBM.COM`, definido en la ampliación de `HostIdMappings` de su certificado de cliente.

```
RDEFINE SERVAUTH IRR.HOST.CDFMVS08.RALEIGH.IBM.COM UACC(NONE)
```

4. Otorgue al ID de usuario de la tarea iniciada RSE, `STCRSE`, acceso a este recurso con autorización de `LECTURA`.

```
PERMIT IRR.HOST.CDFMVS08.RALEIGH.IBM.COM CLASS(SERVAUTH) +  
ACCESS(READ) ID(stcrse)
```

5. Active los cambios en la clase `SERVAUTH`. Utilice el primer mandato si la clase `SERVAUTH` no está todavía activa. Utilice el segundo para renovar una configuración activa.

```
SETOPTS CLASSACT(SERVAUTH) RACLIST(SERVAUTH)  
o bien  
SETOPTS RACLIST(SERVAUTH) REFRESH
```

Esta acción concluye la configuración del software de seguridad para ampliación de `HostIdMappings`.

6. Reinicie la tarea iniciada RSE para empezar a aceptar los inicios de sesión de clientes mediante certificados X.509.

(Opcional) Crear una base de datos de claves con gskkyman

No realice este paso si utiliza un anillo de claves compatible con SAF para la base de datos de claves del daemon RSE.

gskkyman es un programa dirigido por menú y basado en la shell z/OS UNIX que crea, puebla y gestiona un archivo z/OS UNIX que contiene claves privadas, peticiones de certificado y certificados. Este archivo z/OS UNIX se llama base de datos de claves.

Nota: Las siguientes sentencias podrían ser necesarias para configurar el entorno de cara a gskkyman. Consulte la publicación *System SSL Programming* (SC24-5901) para obtener más información acerca de este tema.

```
PATH=$PATH:/usr/lpp/gskssl/bin
export NLSPATH=/usr/lpp/gskssl/lib/nls/msg/En_US.IBM-1047/%N:$NLSPATH
export STEPLIB=$STEPLIB:SYS1.SIEALNKE
```

```
$ cd /etc/rdz/ssl
$ gskkyman          Menú de base de datos
```

1 - Crear base de datos nueva

Entre el número de opción: **1**

Especifique el nombre de la base de datos de claves (pulse Intro para volver al menú): **rdzssl.kdb**

Entre la contraseña de la base de datos (pulse Intro para volver al menú): **rsessl**

Vuelva a entrar la contraseña de la base de datos: **rsessl**

Entre el tiempo de caducidad de la contraseña en días (pulse Intro si no caduca):

Entre la longitud de registro de la base de datos (pulse Intro para utilizar 2500):

Se ha creado la base de datos de claves /etc/rdz/ssl/rdzssl.kdb.

Pulse Intro para continuar.

Menú de gestión de claves

6 - Crear un certificado autofirmado

Entre el número de opción (pulse Intro para volver al menú anterior): **6**

Tipo de certificado

5 - Certificado de usuario o servidor con clave RSA de 1024 bits

Seleccione el tipo de certificado (pulse Intro para volver al menú): **5**

Entre la etiqueta (pulse Intro para volver al menú): **rdzrse**

Entre el nombre de sujeto del certificado

Nombre común (necesario): **rdz rse ssl**

Unidad de organización (OU, opcional): **rdz**

Organización (necesario): **IBM**

Ciudad/Localidad (opcional): **Raleigh**

Estado/Provincia (opcional): **NC**

País/Región (2 caracteres - necesario): **US**

Entre el número de días durante los que el certificado será válido (valor predeterminado, 365): **3650**

Entre 1 para especificar nombres de sujetos alternativos o 0 para continuar: **0**

Espere por favor

El certificado se ha creado.

Pulse Intro para continuar.

Menú de gestión de claves

0 - Salir del programa

Entre el número de opción (pulse Intro para volver al menú anterior): **0**

```
$ ls -l rdzssl.*
```

```
total 152
```

```

-rw----- 1 IBMUSER SYS1      35080 May 24 14:24 rdzssl.kdb
-rw----- 1 IBMUSER SYS1      80 May 24 14:24 rdzssl.rdb
$ chmod 644 rdzssl.*
$ ls -l rdzssl.*
-rw-r--r-- 1 IBMUSER SYS1      35080 May 24 14:24 rdzssl.kdb
-rw-r--r-- 1 IBMUSER SYS1      80 May 24 14:24 rdzssl.rdb

```

En el ejemplo anterior se empieza por crear una base de datos de claves llamada `rdzssl.kdb` con la contraseña `rsessl`. Una vez creada la base de datos, ésta se llena creando un certificado autofirmado, válido durante 10 años (sin contar los días bisiestos). El certificado se almacena bajo la etiqueta `rdzrse` y que tiene la misma contraseña (`rsessl`) que la que se empleó para la base de datos de claves (este es un requisito de RSE).

`gskkyman` asigna la base de datos de claves con una máscara de bit de permiso 600 (muy seguro, el único que tiene acceso es el propietario). Los permisos se tienen que establecer para que sean menos restrictivos, a menos que el daemon utilice el mismo ID de usuario que el creador de la base de datos de claves. 644 (el propietario tiene acceso de lectura/escritura y los demás tienen acceso de lectura) es una máscara que se puede usar para el mandato `chmod`.

El resultado se puede verificar seleccionando la opción **Mostrar información de certificado**, en el submenú **Gestionar claves y certificados**, del siguiente modo:

```
$ gskkyman
```

```
Menú de base de datos
```

```
2 - Abrir base de datos
```

```
Entre el número de opción: 2
```

```
Especifique el nombre de la base de datos de claves (pulse Intro para volver al menú): rdzssl.kdb
```

```
Entre la contraseña de la base de datos (pulse Intro para volver al menú): rsessl
```

```
Menú de gestión de claves
```

```
1 - Gestionar claves y certificados
```

```
Entre el número de opción (pulse Intro para volver al menú anterior): 1
```

```
Lista de claves y certificados
```

```
1 - rdzrse
```

```
Entre el número de etiqueta (Intro para volver al menú de selección, p para lista anterior): 1
```

```
Menú de claves y certificados
```

```
1 - Mostrar información de certificado
```

```
Entre el número de opción (pulse Intro para volver al menú anterior): 1
```

```
Información de certificado
```

```
Etiqueta: rdzrse
ID de registro: 14
ID de registro del emisor: 14
De confianza: Sí
Versión: 3
Número de serie: 45356379000ac997
Nombre del emisor: rdz rse ssl
rdz
```

```

IBM
Raleigh
NC
US
Nombre de sujeto: rdz rse ssl
rdz
IBM
Raleigh
NC
US
Fecha de efectividad: 2007/05/24
Fecha de caducidad: 2017/05/21
Algoritmo de clave pública: rsaEncryption
Tamaño de clave pública: 1024
Algoritmo de signature: sha1WithRsaEncryption
ID exclusivo del emisor: Ninguno
ID exclusivo del sujeto: Ninguno
Número de extensiones: 3

```

Entre 1 para visualizar las extensiones, entre 0 para volver al menú: 0

Menú de claves y certificados

0 - Salir del programa

Entre el número de opción (pulse Intro para volver al menú anterior): 0

El siguiente ejemplo de `ssl.properties` muestra que las directivas de `daemon_*` difieren del ejemplo de anillo de claves de SAF anterior.

```

$ oedit /etc/rdz/ssl/ssl.properties
-> cambiar: enable_ssl=true
-> descomentar y cambiar: daemon_keydb_file=rdzssl.kdb
-> descomentar y cambiar: daemon_keydb_password=rsessl
-> descomentar y cambiar: daemon_key_label=rdzrse
-> descomentar y cambiar: server_keystore_file=rdzssl.racf
-> descomentar y cambiar: server_keystore_label=rdzrse
-> descomentar y cambiar: server_keystore_type=JCERACFKS

```

Los cambios anteriores habilitan SSL e indican al daemon RSE que el certificado está almacenado bajo la etiqueta `rdzrse` en la base de datos de claves `rdzssl.kdb` con la contraseña `rsessl`. El servidor RSE sigue utilizando un anillo de claves compatible con SAF.

(Opcional) Crear un almacén de claves con keytool

No realice este paso si utiliza un anillo de claves compatible con SAF para el almacén de claves del servidor RSE.

"`keytool -genkey`" genera un par de claves privadas y un certificado autofirmado coincidente, que está almacenado como una entrada (identificado por un alias) en un archivo de almacén de claves (nuevo).

Nota: Hay que incluir Java en los directorios de búsqueda de mandatos. Para poder ejecutar `keytool`, podría ser necesaria la siguiente sentencia, donde `/usr/lpp/java/J5.0` es el directorio en el que está instalado Java:

```
PATH=$PATH:/usr/lpp/java/J5.0/bin
```

Toda la información se puede pasar como un parámetro, pero debido a las limitaciones de longitud que tiene la línea de mandatos, se necesita algo de interactividad, del siguiente modo:

```

$ cd /etc/rdz/ssl
$ keytool -genkey -alias rdzrse -validity 3650 -keystore rdzssl.jks -storepass
rsessl -keypass rsessl
¿Cuál es su nombre y su apellido?
[Desconocido]: rdz rse ssl
¿Cuál es el nombre de su unidad de organización (OU)?
[Desconocido]: rdz
¿Cuál es el nombre de su organización?
[Desconocido]: IBM
¿Cuál es el nombre de su ciudad o localidad?
[Desconocido]: Raleigh
¿Cuál es el nombre de su estado o provincia?
[Desconocido]: NC
¿Cuál es el código de dos letras de esta unidad?
[Desconocido]: US
¿Es CN=rdz rse ssl, OU=rdz, O=IBM, L=Raleigh, ST=NC, C=US correcto? (escriba "yes"
o "no")
[no]: yes
$ ls -l rdzssl.*
-rw-r--r--  1 IBMUSER  SYS1          1224 May 24 14:17 rdzssl.jks

```

El certificado autofirmado creado en el ejemplo anterior es válido durante 10 años (sin contar los días bisiestos). Se almacena en /etc/rdz/ssl/rdzssl.jks utilizando el alias rdzrse. Su contraseña (rsessl) es idéntica a la contraseña del almacén de claves, que es un requisito para RSE.

El resultado se puede verificar con la opción -list, del siguiente modo:

```

$ keytool -list -alias rdzrse -keystore rdzssl.jks -storepass rsessl -v
Nombre de alias: rdzrse
Fecha de creación: May 24, 2007
Tipo de entrada: keyEntry
Longitud de la cadena de certificados: 1
Certificado 1:
Propietario: CN=rdz rse ssl, OU=rdz, O=IBM, L=Raleigh, ST=NC, C=US
Emisor: CN=rdz rse ssl, OU=rdz, O=IBM, L=Raleigh, ST=NC, C=US
Número de serie: 46562b2b
Válido desde: 5/24/07 2:17 PM hasta: 5/21/17 2:17 PM
Huellas digitales del certificado
    MD5:  9D:6D:F1:97:1E:AD:5D:B1:F7:14:16:4D:9B:1D:28:80
    SHA1: B5:E2:31:F5:B0:E8:9D:01:AD:2D:E6:82:4A:E0:B1:5E:12:CB:10:1C

```

El siguiente ejemplo de ssl.properties muestra que las directivas de server_* difieren del ejemplo de anillo de claves de SAF anterior.

```

$ oedit /etc/rdz/ssl/ssl.properties
-> cambiar: enable_ssl=true
-> descomentar y cambiar: daemon_keydb_file=rdzssl.racf
-> descomentar y cambiar: daemon_key_label=rdzrse
-> descomentar y cambiar: server_keystore_file=rdzssl.jks
-> descomentar y cambiar: server_keystore_password=rsessl
-> descomentar y cambiar: server_keystore_label=rdzrse
-> descomentar y cambiar (opcional): server_keystore_type=JKS

```

Los cambios anteriores habilitan la SSL e indican al servidor RSE que el certificado está almacenado bajo la etiqueta rdzrse en el almacén de claves rdzssl.jks con la contraseña rsessl. El daemon RSE sigue utilizando un anillo de claves compatible con SAF.

Capítulo 14. Configurar AT-TLS

Esta sección se propone ayudarle a resolver algunos problemas comunes que pueden surgir al configurar el protocolo AT-TLS (Application Transparent Transport Layer Security), o durante la comprobación o modificación de una configuración existente.

El protocolo de seguridad de la capa de transporte (TLS) definido en RFC 2246 proporciona privacidad en las comunicaciones realizadas a través de internet. Al igual que se predecesor SSL (capa de sockets seguros), el protocolo permite que las aplicaciones de servidor y cliente se comuniquen de una forma diseñada para evitar escuchas no autorizadas, manipulaciones indebidas y falsificaciones de mensajes. AT-TLS (Application Transparent Transport Layer Security) consolida la implementación de TLS para aplicaciones basadas en z/OS en una ubicación, permitiendo a todas las aplicaciones soportar el cifrado basado en TLS sin conocer el protocolo TLS. Consulte la publicación *Communications Server IP Configuration Guide* (SC31-8775) para obtener más información sobre AT-TLS.

El depurador integrado en IBM Rational Developer for System z confía en AT-TLS para las comunicaciones cifradas con el cliente, porque la sesión de depuración no fluye por el mismo sitio que otra comunicación entre el host y el cliente Developer for System z.

Las acciones necesarias para configurar AT-TLS varían de un sitio a otro, dependiendo de las necesidades exactas y de lo que ya esté disponible en ese momento.

La información en esta sección muestra cómo se configura el agente de política del protocolo TCP/IP que gestiona el protocolo AT-TLS y definir una política para utilizar con el depurador integrado de Developer for System z en un sistema z/OS 1.13 con soporte para TLS v1.2.

1. “Configurar syslogd” en la página 216
2. “Configuración AT-TLS en PROFILE.TCPIP” en la página 216
3. “Tarea iniciada del agente de política” en la página 217
4. “Configuración del agente de política” en la página 217
5. “Política AT-TLS” en la página 218
6. “Actualizaciones de seguridad de AT-TLS” en la página 220
7. “Activación de la política AT-TLS” en la página 222

A lo largo de esta sección se utiliza un convenio de denominación uniforme:

- Puerto de gestor de depuración para comunicación externa: 5335
- ID de usuario del gestor de depuración: stcdbm
- ID de usuario del agente de política: pagent
- Certificado: dbgmgr
- Almacenamiento de certificados y claves: dbgmgr.racf

En algunas tareas que se describen en las secciones siguientes, se espera que esté activo en z/OS UNIX. Para ello, emita el mandato TSO **OMVS**. Utilice el mandato **oedit** para editar archivos en z/OS UNIX. Utilice el mandato **exit** para volver a TSO.

Configurar syslogd

La documentación de TCP/IP recomienda escribir mensajes del agente de política en el syslog de z/OS UNIX en lugar de utilizar el archivo de registro predeterminado. AT-TLS siempre escribe mensajes en el syslog de z/OS UNIX.

Para ello, el daemon syslog de z/OS UNIX, `syslogd`, tiene que estar configurado y activo. También necesitará un mecanismo para controlar el tamaño de los archivos de registro creados por `syslogd`.

Las siguientes actualizaciones del archivo de configuración de muestra se pueden utilizar para configurar e iniciar `syslogd`, con un simple mecanismo de gestión de archivos de registro (borre los registros existentes cuando se inicie z/OS UNIX y cree otros nuevos con el inicio de `syslogd`).

- `/etc/services`
`syslog 514/udp`
- `/etc/syslog.conf`

```
# /etc/syslog.conf - control output of syslogd
# 1. todos los archivos se imprimirán en /tmp/syslog.auth.log
auth.*          /tmp/syslog.auth.log
# 2. todos los mensajes de error se imprimirán en /tmp/syslog.error.log
*.err           /tmp/syslog.error.log
# 3. todos los mensajes anteriores y de depuración se imprimirán en
                /tmp/syslog.debug.log
*.debug         /tmp/syslog.debug.log
# Los archivos nombrados tienen que existir antes de que se inicie el
# daemon syslog,
# a no ser que se utilice la opción de inicio -c
```
- `/etc/rc`

```
# Inicie el daemon SYSLOGD para el registro
# (limpie registros antiguos)
sed -n '/^#/!s/.* \(.*)/\1p' /etc/syslog.conf | xargs -i rm {}
# (cree registros nuevos y añada el ID de usuario del remitente del mensaje)
_BPX_JOBNAME='SYSLOGD' /usr/sbin/syslogd -cuf /etc/syslog.conf &
sleep 5
```

Configuración AT-TLS en PROFILE.TCPIP

El soporte AT-TLS se activa mediante el parámetro TTLS en la sentencia `TCPCONFIG` del conjunto de datos `PROFILE.TCPIP`. AT-TLS se gestiona mediante el agente de política, que tiene que estar activo para poder imponer la política AT-TLS. Como el agente de política tiene que esperar a que TCP/IP esté activo, la sentencia `AUTOSTART` de `PROFILE.TCPIP` es un buen lugar para activar el inicio de este servidor.

Estos requisitos tienen como resultado cambios en `PROFILE.TCPIP`, que suele denominarse `TCPIP.TCPPARMS(TCPPROF)`.

```
TCPCONFIG TTLS          ; Required for AT-TLS
AUTOLOG
  PAGENT                ; POLICY AGENT, required for AT-TLS
ENDAUTOLOG
```

Tarea iniciada del agente de política

Como mencionamos antes, AT-TLS se gestiona mediante el agente de política, que puede iniciarse como una tarea iniciada. Utilice el siguiente JCL para crear SYS1.PROCLIB(PAGENT), utilizando el archivo de configuración predeterminado y la ubicación de registro recomendada (SYSLOGD). Más adelante se tratan las definiciones necesarias para su software de seguridad.

```
//PAGENT   PROC PRM='-L SYSLOGD'                                * '' or '-L SYSLOGD'
//*
//* TCP/IP POLICY AGENT
//*
//* default cfg file: /etc/pagent.conf      (PARM) (envar)
//* default log file: /tmp/pagent.log      (-C)  (PAGENT_CONFIG_FILE)
//* default log size: 300,3 (3x 300KB files) (PAGENT_LOG_FILE_CONTROL)
//*
//PAGENT   EXEC PGM=PAGENT,REGION=0M,TIME=NOLIMIT,
//          PARM='ENVAR("TZ=EST5DST")/&PRM'
//SYSPRINT DD SYSOUT=*
//SYSOUT   DD SYSOUT=*
//*
```

Configuración del agente de política

El agente de política impone las políticas relacionadas con TCP/IP creadas por el administrador TCP/IP. Gestiona políticas para AT-TLS, denominadas TTLS, pero también para otros servicios como IPsec. El agente de política utiliza un archivo de configuración para saber qué políticas hay que imponer y dónde se encuentran. El archivo de configuración predeterminado es /etc/pagent.conf, pero puede especificarse una ubicación diferente en la tarea iniciada JCL del agente de política.

```
#
# Información de configuración del agente de política TCP/IP.
#
TTLSConfig /etc/pagent.ttls.conf
# Especifica la vía de acceso de un archivo de política TTLS que retiene sentencias
# específicas de la pila.
#
#TcpImage TCPIP /etc/pagent.conf
# Si no se especifica ninguna sentencia TcpImage, se instalarán todas las políticas
# en la pila TCP/IP predeterminada.
#
#LogLevel 31
# La suma de los siguientes valores que representan niveles de registro:
# LOGL_SYSERR      1
# LOGL_OBJERR      2
# LOGL_PROTERR     4
# LOGL_WARNING     8
# LOGL_EVENT      16
# LOGL_ACTION      32
# LOGL_INFO        64
# LOGL_ACNTING     128
# LOGL_TRACE       256
# El nivel de registro 31 es el nivel de registro predeterminado.
#
#Página de códigos IBM-1047
# Especifique la página de códigos EBCDIC que se va a utilizar para leer todos
# los archivos de configuración y los archivos de definición de políticas.
# IBM-1047 es la página de códigos predeterminada.
```

Este archivo de configuración de muestra especifica dónde puede encontrar el agente de política la política TTLS. Utiliza los valores predeterminados del agente de política para otras sentencias.

Política AT-TLS

Una política TTLS describe las reglas AT-TLS deseadas. Tal y como se define en el archivo de configuración del agente de política, la política TTLS se encuentra en /etc/pagent.ttls.conf. Más adelante se tratan las definiciones necesarias para su software de seguridad.

Este ejemplo muestra un política de dos reglas bastante simple que activa el soporte de SSL v3, TLS v1, TLS v1.1 y TLS v1.2 para las vías de acceso de comunicación soportadas por el gestor de depuración, Probe-Client y el depurador integrado de Developer for System z. Tal y como se define en el archivo de configuración del agente de política, la política TTLS se encuentra en /etc/pagent.ttls.conf.

```
##
## TCP/IP Policy Agent AT-TLS configuration information.
##
##-----
TTLSRule                                RDz_Debug_Manager
{
  LocalPortRange                        5335
  Dirección                            De entrada
  TTLSGroupActionRef                    grp_Production
  TTLSEnvironmentActionRef              act_RDz_Debug_Manager
}
##-----
TTLSEnvironmentAction                    act_RDz_Debug_Manager
{
  HandshakeRole Server
  TTLSKeyRingParms
  {
    Conjunto de claves dbgmgr.racf
  }
  # El gestor de depuración debe poseer el conjunto de claves
  TTLSEnvironmentAdvancedParms
  {
    ## TLSV1.2 solo para z/OS 2.1 y superior
    # TLSV1.2 On                        # SSLv3, TLSv1 & TLSv1.1
    # están activados de forma predeterminada
  }
}
##-----
TTLSRule                                RDz_Debug_Probe-Client
{
  RemotePortRange                       8001
  Direction                             Outbound
  TTLSGroupActionRef                    grp_Production
  TTLSEnvironmentActionRef              act_RDz_Debug_Probe-Client
}
##-----
TTLSEnvironmentAction                    act_RDz_Debug_Probe-Client
{
  HandshakeRole                         Client
  TTLSKeyRingParms
  {
    Keyring *AUTH/*                     # virtual key ring holding CA certificates
  }
  TTLSEnvironmentAdvancedParms
  {
    ## TLSV1.2 solo para z/OS 2.1 y superior
    # TLSV1.2 On                        # SSLv3, TLSv1 & TLSv1.1 are on by default
  }
}
##-----
TTLSGroupAction                          grp_Production
```

```

{
  TTLS-enabled          Activado
## TLSv1.2zOS1.13 solo para z/OS 1.13
  TTLSGroupAdvancedParmsRef TLSv1.2zOS1.13
  Trace                 3      # Log Errors to syslogd & IP joblog
#Trace                 254    # Log everything to syslogd
}
##-----
TTLSGroupAdvancedParms TLSv1.2zOS1.13
{
  Envfile /etc/pagent.ttls.TLS1.2zOS1.13.env
}

```

Una política TTLS tiene en cuenta gran cantidad de filtros para especificar cuando se aplica una regla.

El gestor de depuración es un servidor que escucha en el puerto 5335 conexiones de entrada del motor de depuración. Esta información se almacena en la regla RDz_Debug_Manager.

Como SSL y TLS requieren el uso de un certificado de servidor, especifique que el gestor de políticas tiene que utilizar los certificados del conjunto de claves dbgmgr.racf, propiedad del Id de usuario de tarea iniciada del gestor de depuración. De forma predeterminada, el soporte de TLS v1.2 está inhabilitado, por lo que esta política lo habilita explícitamente.

Cuando se inicia el analizador de depuración con la opción de Language Environment (LE) TEST(,,,TCPIP&&ipaddress%8001:*), se le solicita que no utilice el gestor de depuración, sino que se ponga en contacto con el cliente de Developer for System z directamente en el puerto 8001. Esto implica, desde una perspectiva TCP/IP, que el analizador de depuración basado en host es un cliente que se pone en contacto con un servidor (el Debug UI) en el cliente Developer for System z. Esta información se captura en la regla RDz_Debug_Probe-Client.

como el host es un cliente TCP/IP, el gestor de políticas necesitará validar el certificado de servidor presentado por el Debug UI. En lugar de utilizar un conjunto de claves con nombre para todos los usuarios que pueden necesitar una sesión de depuración cifrada, utilizamos en conjunto de claves virtual CERTAUTH de RACF (*AUTH*/*). Este conjunto de claves virtual contiene los certificados públicos de las entidades emisoras de certificados y puede utilizarse si el Debug UI presenta un certificado de servidor firmado por una de las entidades emisoras de certificados de confianza.

Tenga en cuenta que para políticas más complejas, debe utilizar el Asistente de configuración de IBM para z/OS Communications Server. Se trata de una herramienta basada en GUI que proporciona una interfaz guiada para configurar funciones de red basadas en políticas TCP/IP y está disponible como tarea en IBM z/OS Management Facility (z/OSMF) y como aplicación autónoma de la estación de trabajo.

Consideraciones TLS v1.2

El soporte de TLS v1.2 está disponible a partir de z/OS 2.1, y está inhabilitado de forma predeterminada. Esta política muestra el mandato (TLSV1.2 0n) para habilitarlo de forma explícita, pero lo tiene descomentado porque el sistema de destino está utilizando z/OS 1.13.

Al aplicar los dos siguientes APAR, se añade el soporte TLS v1.2 a z/OS 1.13:

- Sistema SSL APAR OA39422
- Servidor de comunicaciones (AT-TLS) APAR PM62905

z/OS 1.13 System SSL, que utiliza AT-TLS para implementar una comunicación TLS encriptada, requiere parámetros adicionales para el soporte TLS v1.2. Estos parámetros se suministran mediante la política AT-TLS utilizando un archivo con variables de entorno de System SSL, /etc/pagent.ttls.TLS1.2zOS1.13.env.

```
#
# Añada soporte TLSv1.2 a AT-TLS
# es necesario z/OS 1.13 con OA39422 y PM62905
#
GSK_RENEGOTIATION=ALL
GSK_PROTOCOL_TLSV1_2=ON
```

Actualizaciones de seguridad de AT-TLS

Hay varias actualizaciones necesarias para su configuración de seguridad para que AT-TLS funcione correctamente. Esta sección incluye mandatos RACF de ejemplo para realizar la configuración necesaria.

Como ya se ha mencionado en “Tarea iniciada del agente de política” en la página 217, utilice una tarea iniciada para ejecutar el agente de política. Para esto hace falta un ID de usuario de tarea iniciada y un perfil en la clase STARTED.

```
# defina el ID de usuario de tarea iniciada
# El permiso BPX.DAEMON es necesario para un ID de usuario no cero
ADDUSER PAGENT DFLTGRP(SYS1) OMVS(UID(0) SHARED HOME('/')) +
  NAME('TCP/IP POLICY AGENT') NOPASSWORD

# defina la tarea iniciada
RDEFINE STARTED PAGENT.* STDATA(USER(PAGENT) GROUP(SYS1)) +
  DATA('TCP/IP POLICY AGENT')

# renueve para que los cambios sean visibles
SETROPTS RACLIST(STARTED) REFRESH
```

Defina un perfil llamado MVS.SERVGR.PAGENT en la clase OPERCMDS y conceda al ID de usuario PAGENT acceso de CONTROL al mismo. El perfil restringe quién puede iniciar el agente de política. Si el perfil no se ha definido y se impide acceder a él mediante un perfil genérico, PAGENT no podrá iniciar el agente de política, que impedirá la inicialización de la pila TCP/IP.

```
# restrinja el inicio del agente de política
RDEFINE OPERCMDS MVS.SERVGR.PAGENT UACC(NONE) +
  DATA('restrict startup of policy agent')
PERMIT MVS.SERVGR.PAGENT CLASS(OPERCMDS) ACCESS(CONTROL) ID(PAGENT)

# renueve para que los cambios sean visibles
SETROPTS RACLIST(OPERCMDS) REFRESH
```

Tal y como se menciona en “Configuración AT-TLS en PROFILE.TCPIP” en la página 216, el agente de política se inicia una vez inicializado TCP/IP. Esto significa que hay una ventana (pequeña) en la que las aplicaciones pueden utilizar la pila TCP/IP sin que se imponga la política TTLS. Defina el perfil EZB.INITSTACK.** en la clase SERVAUTH para impedir el acceso a la pila durante esta ventana de tiempo, excepto para las aplicaciones con acceso de lectura al perfil. Tiene que permitir un conjunto limitado de aplicaciones administrativas para el perfil para garantizar la inicialización completa de la pila, tal y como se explica en “TCP/IP stack initialization access control” *Communications Server IP Configuration Guide* (SC31-8775).

```
# bloquee el acceso de pila entre la pila y la disponibilidad AT-TLS
# SETROPTS GENERIC(SERVAUTH)
# SETROPTS CLASSACT(SERVAUTH) RACLIST(FACILITY)
# RDEFINE SERVAUTH EZB.INITSTACK.** UACC(NONE)
# Agente de política
# PERMIT EZB.INITSTACK.** CLASS(SERVAUTH) ACCESS(READ) ID(PAGENT)
# Daemon OMPROUTE
# PERMIT EZB.INITSTACK.** CLASS(SERVAUTH) ACCESS(READ) ID(OMPROUTE)
# Agente SNMP y subagentes
# PERMIT EZB.INITSTACK.** CLASS(SERVAUTH) ACCESS(READ) ID(OSNMPD)
# PERMIT EZB.INITSTACK.** CLASS(SERVAUTH) ACCESS(READ) ID(IOBSNMP)
# Daemon NAME
# PERMIT EZB.INITSTACK.** CLASS(SERVAUTH) ACCESS(READ) ID(NAMED)

# renueve para que los cambios sean visibles
SETROPTS RACLIST(SERVAUTH) REFRESH
```

(Opcional) El mandato de z/OS UNIX **pasearch** muestra definiciones de política activas. Defina el perfil EZB.PAGENT.** en la clase SERVAUTH para restringir el acceso al mandato **pasearch**.

```
# restrinja el acceso al mandato pasearch
# RDEFINE SERVAUTH EZB.PAGENT.** UACC(NONE) +
# DATA('restrict access to pasearch command')
# PERMIT EZB.PAGENT.** CLASS(SERVAUTH) ACCESS(READ) ID(tcpadmin)

# renueve para que los cambios sean visibles
# SETROPTS RACLIST(SERVAUTH) REFRESH
```

Tal y como se menciona en “Política AT-TLS” en la página 218, el gestor de depuración necesita un certificado para que AT-TLS pueda configurar una comunicación encriptada SSL o TLS en nombre del gestor de depuración. Estos mandatos de ejemplo crean un certificado nuevo denominado dbgmgr, que se almacena en un conjunto de claves RACF llamado dbgmgr.racf. Tanto el certificado como el conjunto de claves son propiedad de STCDBM, el ID de usuario de tarea iniciada del gestor de depuración.

```
# permita al gestor de depuración que acceda al certificado
#RDEFINE FACILITY IRR.DIGTCERT.LIST UACC(NONE)
#RDEFINE FACILITY IRR.DIGTCERT.LISTRING UACC(NONE)
# PERMIT IRR.DIGTCERT.LIST CLASS(FACILITY) ACCESS(READ) ID(stcdbm)
# PERMIT IRR.DIGTCERT.LISTRING CLASS(FACILITY) ACCESS(READ) ID(stcdbm)

# renueve para que los cambios sean visibles
SETROPTS RACLIST(FACILITY) REFRESH

# cree el certificado autofirmado
RACDCERT ID(stcdbm) GENCERT SUBJECTSDN(CN('RDz Debug Manager') +
OU('RTP labs') O('IBM') L('Raleigh') SP('NC') C('US')) +
NOTAFTER(2015-12-31) KEYUSAGE(HANDSHAKE) WITHLABEL('dbgmgr')

# (opcional) pasos adicionales necesarios para usar un certificado con firma
# 1. cree una solicitud de firma para el certificado autofirmado
RACDCERT ID(stcdbm) GENREQ (LABEL('dbgmgr')) DSN(dsn)
# 2. envíe la solicitud de firma a la CA de su elección
# 3. compruebe si las credenciales de la CA (también un certificado)
# ya se conocen
RACDCERT CERTAUTH LIST
# 4. marque el certificado de autoridad emisora de certificados como fiable
RACDCERT CERTAUTH ALTER(LABEL('CA cert')) TRUST
# o añada el certificado de autoridad emisora de certificados a la base
# de datos
RACDCERT CERTAUTH ADD(dsn) WITHLABEL('CA cert') TRUST
# 5. añada el certificado firmado a la base de datos;
# de esta forma se sustituye el autofirmado
RACDCERT ID(stcdbm) ADD(dsn) WITHLABEL('dbgmgr') TRUST
```

```
# NO suprima el certificado autofirmado antes de sustituirlo.
# Si lo hace, perderá la clave privada que acompaña al certificado,
# lo que hace que el certificado no sirva para nada.

# cree un conjunto de claves
RACDCERT ID(stcdbm) ADDRING(dbgmgr.racf)

# añada el certificado al conjunto de claves
RACDCERT ID(stcdbm) CONNECT(LABEL('dbgmgr') +
RING(dbgmgr.racf) USAGE(PERSONAL) DEFAULT)

# paso adicional necesario para utilizar un certificado firmado
# 6. añada el certificado de autoridad emisora de certificados al conjunto de
claves
RACDCERT ID(stcdbm) CONNECT(CERTAUTH LABEL('CA cert') +
RING(dbgmgr.racf))

# renueve para que los cambios sean visibles
SETROPTS RACLIST(DIGTCERT) REFRESH
```

La política AT-TLS también documenta el uso del conjunto de claves virtual CERTAUTH para la validación del certificado de servidor presentado por el Debug UI en un escenario Probe-Client. Esto implica que el certificado CA que utiliza Debug UI es considerado de confianza por el host de z/OS.

```
# compruebe si las credenciales de la CA (también un certificado) ya se conocen
RACDCERT CERTAUTH LIST
# marque el certificado de autoridad emisora de certificados como fiable
RACDCERT CERTAUTH ALTER(LABEL('CA cert')) TRUST
# o añada el certificado de autoridad emisora de certificados a la base
# de datos
RACDCERT CERTAUTH ADD(dsn) WITHLABEL('CA cert') TRUST

# renueve para que los cambios sean visibles
SETROPTS RACLIST(DIGTCERT) REFRESH
```

Utilice los siguientes mandatos para verificar la configuración:

```
# verifique la configuración de la tarea iniciada
LISTGRP SYS1 OMVS
LISTUSER PAGENT OMVS
RLIST STARTED PAGENT.* ALL STDATA

# verifique el permiso de inicio dl agente de política
RLIST OPERCMDS MVS.SERVMMGR.PAGENT ALL

# verifique la protección de initstack
RLIST SERVAUTH EZB.INITSTACK.** ALL

# verifique la protección de pasearch
RLIST SERVAUTH EZB.PAGENT.** ALL

# verifique la configuración del certificado
RACDCERT CERTAUTH LIST(LABEL('CA cert'))
RACDCERT ID(stcdbm) LIST(LABEL('dbgmgr'))
RACDCERT ID(stcdbm) LISTRING(dbgmgr.racf)
```

Activación de la política AT-TLS

Se ha completado la configuración de AT-TLS y la política se activará en la siguiente IPL del sistema. Siga estos pasos para empezar a utilizar la política sin una IPL:

1. Active el soporte AT-TLS en la pila TCP/IP.

Cree un archivo obey TCP/IP, por ejemplo, TCPIP.TCPPARMS(OBEY), con el siguiente contenido:

```
TCPCONFIG TTLS
```

Actívelo con este mandato de operador:

```
V TCPIP,,OBEY,TCPIP.TCPPARMS(OBEY)
```

Compruebe el resultado buscando este mensaje de la consola:

```
EZZ4249I stackname INSTALLED TTLS POLICY HAS NO RULES
```

2. Inicie el agente de política.

Emita el mandato de operador:

```
S PAGENT
```

Verifique el resultado buscando el mensaje de la consola:

```
EZD1586I PAGENT HAS INSTALLED ALL LOCAL POLICIES FOR stackname
```

3. Reinicie el gestor de depuración para interrumpir todas las sesiones activa no encriptadas.

Emita los mandatos de operador:

```
P DBGMR
```

```
S DBBMGR
```

Capítulo 15. Configurar TCP/IP

Esta sección se propone ayudarle a resolver algunos problemas comunes que pueden surgir al configurar TCP/IP, o durante la tarea de comprobar o modificar una configuración existente.

Consulte las publicaciones *Communications Server: IP Configuration Guide* (SC31-8775) y *Communications Server: IP Configuration Reference* (SC31-8776) para obtener más información acerca de la configuración de TCP/IP.

Dependencia del nombre de host

Al utilizar APPC para el servicio de mandatos TSO, Developer for System z depende de que TCP/IP tenga el nombre de host correcto cuando se inicializa. Ello implica que los distintos archivos de configuración de TCP/IP y del resolvente estén configurados correctamente.

Puede probar la configuración de TCP/IP con IVP (programa de verificación de la instalación) fekfivpt. El mandato debe devolver datos de salida parecidos a los de este ejemplo (\$ es el indicador de z/OS UNIX):

```
$ fekfivpt
```

```
Wed Jul  2 13:11:54 EDT 2008
uid=1(USERID) gid=0(GROUP)
using /etc/rdz/rsed.envvars
```

```
-----
configuración del resolvente TCP/IP (orden de búsqueda de z/OS UNIX):
-----
```

```
Inicialización de rastreo de resolvente completada -> 2008/07/02 13:11:54.745964
```

```
Valores de resolvente res_init:
```

```
Conjunto de datos Tcp/Ip global      = Ninguno
Conjunto de datos Tcp/Ip predeterminado = Ninguno
Conjunto de datos Tcp/Ip local       = /etc/resolv.conf
Tabla de conversión                   = Predeterminada
IDusuario/NombreTrabajo               = USERID
API llamante                         = LE C Sockets
Modalidad llamante                    = EBCDIC
(L) DataSetPrefix = TCPIP
(L) HostName      = CDFMVS08
(L) TcpIpJobName  = TCPIP
(L) DomainOrigin  = RALEIGH.IBM.COM
(L) NameServer    = 9.42.206.2
                  9.42.206.3
(L) NsPortAddr    = 53          (L) ResolverTimeout    = 10
(L) ResolveVia    = UDP         (L) ResolverUdpRetries = 1
(*) Options NDots = 1
(*) SockNoTestStor
(*) AlwaysWto     = NO          (L) MessageCase       = MIXED
(*) LookUp        = DNS LOCAL
```

```
res_init Satisfactoria
```

```
res_init Iniciada: 2008/07/02 13:11:54.755363
```

```
res_init Finalizada: 2008/07/02 13:11:54.755371
```

```
*****
```

```
MVS TCP/IP NETSTAT CS V1R9      Nombre TCPIP: TCPIP      13:11:54
```

```
Tcpip iniciado a las 01:28:36 el 06/23/2008 con IPv6 habilitado
```

```
-----
```

dirección IP del host:

```
-----  
hostName=CDFMVS08  
hostAddr=9.42.112.75  
bindAddr=9.42.112.75  
localAddr=9.42.112.75
```

Satisfactorio, coincidencia de direcciones

¿Qué son los resolventes?

El resolvente funciona en nombre de los programas como un cliente que accede a los servidores de nombres para obtener una resolución de nombre en dirección o una resolución de dirección en nombre. Para resolver la consulta del programa peticionario, el resolvente puede acceder a los servidores de nombres disponibles, utilizar definiciones locales (por ejemplo, `/etc/resolv.conf`, `/etc/hosts`, `/etc/ipnodes`, `HOSTS.SITEINFO`, `HOSTS.ADDRINFO` o `ETC.IPNODES`), o utilizar una combinación de ambos.

El resolvente, al iniciarse su espacio de direcciones, lee un conjunto de datos de instalación opcional del resolvente hacia el que señala la tarjeta DD SETUP en el procedimiento del JCL del resolvente. Si no se proporciona información de instalación, el resolvente utiliza el orden de búsqueda nativo de MVS o z/OS UNIX aplicable sin ninguna información de `GLOBALTCPIPDATA`, `DEFAULTTCPIPDATA`, `GLOBALIPNODES`, `DEFAULTIPNODES` o `COMMONSEARCH`.

Qué es el orden de búsqueda de la información de configuración

Es importante comprender el orden de búsqueda de archivos de configuración que las funciones de TCP/IP utilizan, y conviene saber cuándo se puede alterar temporalmente el orden de búsqueda predeterminado con las variables de entorno, con el JCL o con otras variables que proporcione. Este conocimiento le permite acomodar sus estándares de denominación de los conjuntos de datos y archivos de HFS locales, y también le resultará de utilidad conocer el conjunto de datos o archivo de HFS de configuración al diagnosticar problemas.

Otro punto importante a tener en cuenta es que cuando se aplica un orden de búsqueda para cualquier archivo de configuración, la búsqueda finaliza con el primer archivo que se encuentre. Por lo tanto, es posible obtener resultados inesperados si coloca información de configuración en un archivo que nunca se va a encontrar, ya sea porque existen otros archivos antes según el orden de la búsqueda, o porque el archivo no está incluido en el orden de búsqueda elegido por la aplicación.

Al buscar archivos de configuración, puede indicar explícitamente a TCP/IP dónde está la mayoría de los archivos de configuración, utilizando para ello sentencias DD en los procedimientos del JCL o estableciendo variables de entorno. Por otro lado, puede dejar que sea TCP/IP el que determine dinámicamente la ubicación de los archivos de configuración, basándose en el orden de búsqueda documentado en la publicación *Communications Server: IP Configuration Guide* (SC31-8775).

El componente de configuración de la pila de TCP/IP utiliza `TCPIP.DATA` durante la inicialización de la pila de TCP/IP para determinar el nombre de host (`HOSTNAME`) de la pila. Para obtener este valor, se utiliza el orden de búsqueda del entorno z/OS UNIX.

Nota: Utilice el recurso de resolvente de rastreo para determinar qué valores de TCPIP.DATA utiliza el resolvente y dónde se han leído. Para obtener información sobre cómo iniciar dinámicamente el rastreo, consulte la publicación *Communications Server: IP Diagnosis Guide* (GC31-8782). Una vez que el rastreo esté activo, emita un mandato TSO **NETSTAT HOME** y un mandato **netstat -h** de la shell de z/OS UNIX para mostrar los valores. Si se emite un mandato PING de un nombre de host desde TSO y desde la shell z/OS UNIX, también se muestra la actividad de los servidores DNS que podrían estar configurados.

Orden de búsqueda utilizado en el entorno z/OS UNIX

El archivo o tabla concreto que se busca puede ser un conjunto de datos MVS o un archivo HFS, en función de los valores de configuración del resolvente y de la presencia de determinados archivos en el sistema.

Archivos de configuración de resolvente base

El archivo de configuración de resolvente base contiene sentencias TCPIP.DATA. Además de las directivas del resolvente, se le hace referencia para determinar, entre otras cosas, el prefijo de conjunto de datos (valor de la sentencia DATASETPREFIX) que hay que utilizar al intentar acceder a algunos de los archivos de configuración especificados en esta sección.

El orden de búsqueda empleado para acceder al archivo de configuración resolvente base es el siguiente:

1. **GLOBALTCPIPDATA**

Si está definido, se utiliza el valor de la sentencia de configuración GLOBALTCPIPDATA del resolvente (vea también: “¿Qué son los resolventes?” en la página 226). La búsqueda continúa hasta encontrar un archivo de configuración adicional. La búsqueda finaliza con el próximo archivo encontrado.

2. El valor de la variable de entorno **RESOLVER_CONFIG**

Se utiliza el valor de la variable de entorno. Esta búsqueda fallará si el archivo no existe o si se ha asignado de manera exclusiva en otra parte.

3. **/etc/resolv.conf**

4. Tarjeta **DD //SYSTCPD**

Se utiliza el conjunto de datos asignado a la DD de nombre SYSTCPD. En el entorno z/OS UNIX, un proceso hijo no tiene acceso a la DD SYSTCPD. Ello se debe a que la asignación de SYSTCPD no se hereda del proceso padre por las llamadas a fork() o a la función exec.

5. **userid.TCPIP.DATA**

userid es el ID de usuario asociado al entorno de seguridad actual (espacio de direcciones, tarea o hebra).

6. **jobname.TCPIP.DATA**

jobname es el nombre especificado en la sentencia JCL de JOB para los trabajos por lotes o el nombre de un procedimiento iniciado.

7. **SYS1.TCPPARMS(TCPDATA)**

8. **DEFAULTTCPIPDATA**

Si está definido, se utiliza el valor de la sentencia de configuración DEFAULTTCPIPDATA del resolvente (vea también: “¿Qué son los resolventes?” en la página 226).

9. **TCPIP.TCPIP.DATA**

Tablas de conversión

A las tablas de conversión (de EBCDIC a ASCII y de ASCII a EBCDIC) se les hace referencia para determinar los conjuntos de datos de conversión que hay que utilizar. El orden de búsqueda empleado para acceder a este archivo de configuración es el siguiente. La búsqueda finaliza en el primer archivo encontrado:

1. El valor de la variable de entorno **X_XLATE** El valor de variable de entorno es el nombre de la tabla de conversión producida por el mandato TSO **CONVXLAT**.
2. **userid.STANDARD.TCPXLBIN**
userid es el ID de usuario asociado al entorno de seguridad actual (espacio de direcciones o tarea/hebra).
3. **jobname.STANDARD.TCPXLBIN**
jobname es el nombre especificado en la sentencia JCL de JOB para los trabajos por lotes o el nombre de un procedimiento iniciado.
4. **hlq.STANDARD.TCPXLBIN**
hlq representa el valor de la sentencia DATASETPREFIX especificada en el archivo de configuración de resolvente base (si se da con él); en caso contrario, hlq es TCPIP por defecto.
5. Si no se encuentra ninguna tabla, el resolvente emplea una tabla predeterminada codificada por programa, idéntica a la tabla que figura en el miembro de conjunto de datos SEZATCPX(STANDARD).

Tablas de hosts locales

Por defecto, en primer lugar el resolvente intenta utilizar los servidores de nombres de dominio que estén configurados para las peticiones de resolución. Si la petición de resolución no se puede satisfacer, se emplean las tablas de hosts locales. El comportamiento del resolvente se controla mediante las sentencias TCPIP.DATA.

Las sentencias TCPIP.DATA del resolvente definen si hay que utilizar los servidores de nombres de dominio y cómo hay que hacerlo. La sentencia LOOKUP TCPIP.DATA también puede servir para controlar cómo se utilizan los servidores de nombres de dominio y las tablas de hosts locales. Para obtener más información sobre las sentencias TCPIP.DATA, consulte la publicación *Communications Server: IP Configuration Reference* (SC31-8776).

El resolvente emplea el orden de búsqueda exclusivo de Ipv4 para obtener información de nombres de locales incondicionalmente para las llamadas a la API getnetbyname. El orden de búsqueda exclusivo de Ipv4 para obtener información de nombres de locales es el siguiente. La búsqueda finaliza en el primer archivo encontrado:

1. El valor de la variable de entorno **X_SITE**
El valor de la variable de entorno es el nombre del archivo de información HOSTS.SITEINFO creado por el mandato TSO **MAKESITE**.
2. El valor de la variable de entorno **X_ADDR**
El valor de la variable de entorno es el nombre del archivo de información HOSTS.ADDRINFO creado por el mandato TSO **MAKESITE**.
3. **/etc/hosts**
4. **userid.HOSTS.SITEINFO**
userid es el ID de usuario asociado al entorno de seguridad actual (espacio de direcciones o tarea/hebra).

5. jobname.HOSTS.SITEINFO

jobname es el nombre especificado en la sentencia JCL de JOB para los trabajos por lotes o el nombre de un procedimiento iniciado.

6. hlq.HOSTS.SITEINFO

hlq representa el valor de la sentencia DATASETPREFIX especificada en el archivo de configuración de resolvente base (si se da con él); en caso contrario, hlq es TCPIP por defecto.

Aplicación de esta información de configuración a Developer for System z

Tal como se ha indicado antes, Developer for System z depende de que TCP/IP tenga el nombre de host correcto en el momento de la inicialización cuando se está utilizando APPC. Ello implica que los distintos archivos de configuración de TCP/IP y del resolvente estén configurados correctamente.

El ejemplo siguiente se centra en algunas tareas de configuración de TCP/IP y del resolvente. Tenga en cuenta que no se trata de describir una configuración completa de TCP/IP o del resolvente, sino tan solo de resaltar algunos aspectos clave que podrían ser válidos para su local:

1. En el siguiente JCL, puede ver que TCP/IP empleará SYS1.TCPPARMS(TCPDATA) para determinar el nombre de host de la pila.

```
//TCPIP    PROC  PARMS='CTRACE(CTIEZB00)',PROF=TCPPROF,DATA=TCPDATA
//*
//* RED TCP/IP
//*
//TCPIP    EXEC  PGM=EZBTCPIP,REGION=0M,TIME=1440,PARM=&PARMS
//PROFILE  DD   DISP=SHR,DSN=SYS1.TCPPARMS(&PROF)
//SYSTCPD  DD   DISP=SHR,DSN=SYS1.TCPPARMS(&DATA)
//SYSPRINT DD   SYSOUT=*,DCB=(RECFM=VB,LRECL=132,BLKSIZE=136)
//ALGPRINT DD   SYSOUT=*,DCB=(RECFM=VB,LRECL=132,BLKSIZE=136)
//CFGPRINT DD   SYSOUT=*,DCB=(RECFM=VB,LRECL=132,BLKSIZE=136)
//SYSOUT   DD   SYSOUT=*,DCB=(RECFM=VB,LRECL=132,BLKSIZE=136)
//CEEDUMP  DD   SYSOUT=*,DCB=(RECFM=VB,LRECL=132,BLKSIZE=136)
//SYSERROR DD   SYSOUT=*
```

2. SYS1.TCPPARMS(TCPDATA) nos indica que queremos que el nombre del sistema sea el nombre de host y que no utilizamos un servidor de nombres de dominio (DNS); todos los nombres se resolverán por medio de la búsqueda en la tabla de locales.

```
; HOSTNAME especifica el nombre de host TCP de este sistema. Si no
; se especifica, el HOSTNAME predeterminado será el nombre de nodo especificado
; en el miembro IEFSSNxx PARMLIB.
;
; HOSTNAME
;
; DOMAINORIGIN especifica el origen de dominio que se añadirá
; a los nombres de host que se pasen al resolvente. Si un nombre de host
; contiene puntos, el DOMAINORIGIN no se añadirá al final del
; nombre de host.
;
DOMAINORIGIN  RALEIGH.IBM.COM
;
; NSINTERADDR especifica la dirección IP del servidor de nombres.
; LOOPBACK (14.0.0.0) especifica el servidor de nombres local. Si
; no se va a emplear uno, no codifique una sentencia NSINTERADDR.
; (Descomente la siguiente línea NSINTERADDR). Hará que todos los nombres
; se resuelvan por medio de la búsqueda de la tabla de locales.
;
; NSINTERADDR  14.0.0.0
```



```

;
; TRACE RESOLVER provocará un rastreo completo de todas las consultas y
; respuestas del servidor de nombres o de las tablas de locales, que se
; van a grabar en la consola de usuario. Este mandato solo tiene fines
; de depuración.
;
; TRACE RESOLVER

```

3. En el JCL del resolvente vemos que no se utilizar la sentencia DD SETUP. Como ya se mencionó en: “¿Qué son los resolventes?” en la página 226, esto quiere decir que no se empleará la variable GLOBALTCPIPDATA ni tampoco otras variables.

```

//RESOLVER PROC PARMS='CTRACE(CTIRES00)'
//*
/* IP NAME RESOLVER – START WITH SUB=MSTR
/*
//RESOLVER EXEC PGM=EZBREINI,REGION=0M,TIME=1440,PARM=&PARMS
/*SETUP DD DISP=SHR,DSN=USER.PROCLIB(RESSETUP),FREE=CLOSE

```

4. Si damos por sentado que la variable de entorno RESOLVER_CONFIG no está establecida, podemos ver en la Tabla 45 en la página 231 que el resolvente intentará utilizar /etc/resolv.conf como archivo de configuración base.

```

TCPIPJOBNAME TCPIP
DomainOrigin RALEIGH.IBM.COM
HostName CDFMVS08

```

Como ya se ha mencionado en: “Orden de búsqueda utilizado en el entorno z/OS UNIX” en la página 227, el archivo de configuración base contiene sentencias TCPIP.DATA. Si el nombre del sistema es CDFMVS08 (TCPDATA indicaba que se utiliza el nombre del sistema como nombre de host), podemos ver que /etc/resolv.conf está en sincronización con SYS1.TCPPARMS(TCPDATA). No hay definiciones de DNS y, por lo tanto, se utilizará la búsqueda de la tabla de locales.

5. La Tabla 45 en la página 231 también nos indica que si no tenemos nada que hacer, se utiliza la tabla de conversión ASCII-EBCDIC predeterminada.
6. Suponiendo que no se utiliza el mandato TSO **MAKESITE** (puede crear las variables X_SITE y X_ADDR), /etc/hosts será la tabla de locales empleada para la búsqueda de nombres.

```

# Resolvente /etc/hosts file cdfmvs08
9.42.112.75 cdfmvs08 # Host CDFMVS08
9.42.112.75 cdfmvs08.raleigh.ibm.com # Host CDFMVS08
127.0.0.1 localhost

```

El contenido mínimo de este archivo es información sobre el sistema actual. En el ejemplo anterior, tanto cdfmvs08 como cdfmvs08.raleigh.ibm.com se definen como un nombre válido para la dirección IP del sistema z/OS.

Si utiliza un servidor de nombres de dominio (DNS), el DNS contendría la información de /etc/hosts, y /etc/resolv.conf y SYS1.TCPPARMS(TCPDATA) tendrían sentencias que identificarían el DNS ante su sistema.

Para evitar confusiones, debe mantener los archivos de configuración de TCP/IP y del resolvente sincronizados entre sí.

Tabla 45. Definiciones locales disponibles para el resolvente

Descripción de tipo de archivo	Interfaces API afectadas	Archivos candidatos
Archivos de configuración de resolvente base	Todas las API	<ol style="list-style-type: none"> 1. GLOBALTCPIPDATA 2. Variable de entorno RESOLVER_CONFIG 3. /etc/resolv.conf 4. SYSTCPD DD-name 5. userid.TCPIP.DATA 6. jobname.TCPIP.DATA 7. SYS1.TCPPARMS(TCPDATA) 8. DEFAULTTCPIPDATA 9. TCPIP.TCPIP.DATA
Tablas de conversión	Todas las API	<ol style="list-style-type: none"> 1. Variable de entorno X_XLATE 2. userid.STANDARD.TCPXLBIN 3. jobname.STANDARD.TCPXLBIN 4. hlq.STANDARD.TCPXLBIN 5. Tabla de conversión proporcionada por el resolvente, miembro STANDARD de SEZATCPX
Tablas de hosts locales	endhostent endnetent getaddrinfo gethostbyaddr gethostbyname gethostent GetHostNumber GetHostResol GetHostString getnameinfo getnetbyaddr getnetbyname getnetent IsLocalHost Resolve sethostent setnetent	IPv4 <ol style="list-style-type: none"> 1. Variable de entorno X_SITE 2. Variable de entorno X_ADDR 3. /etc/hosts 4. userid.HOSTS.xxxxINFO 5. jobname.HOSTS.xxxxINFO 6. hlq.HOSTS.xxxxINFO IPv6 <ol style="list-style-type: none"> 1. GLOBALIPNODES 2. Variable de entorno RESOLVER_IPNODES 3. userid.ETC.IPNODES 4. jobname.ETC.IPNODES 5. hlq.ETC.IPNODES 6. DEFAULTIPNODES 7. /etc/ipnodes

Nota: La Tabla 45 es una copia parcial de una tabla de la publicación *Communications Server: IP Configuration Guide* (SC31-8775). En ese manual encontrará la tabla completa.

La dirección del host no se resuelve correctamente

Cuando encuentre problemas relacionados con que el Resolvente de TCP/IP no pueda resolver la dirección de host correctamente, probablemente se deba a un archivo de configuración resolvente faltante o incompleto. Una indicación clara de este problema es el mensaje siguiente en `lock.log`:

```
clientip(0.0.0.0) <> callerip(<dirección IP host>)
```

Para verificarlo, ejecute el IVIP de TCP/IP fekfivpt, como se describe en la sección "Verificación de la instalación" de la publicación *Guía de configuración de host* (SC11-3660). La sección de configuración del resolvente de la salida será como la del ejemplo siguiente:

Inicialización de rastreo de resolvente completada -> 2008/07/02 13:11:54.745964

```
Valores de resolvente res_init:
Conjunto de datos Tcp/Ip global      = Ninguno
Conjunto de datos Tcp/Ip predeterminado = Ninguno
Conjunto de datos Tcp/Ip local      = /etc/resolv.conf
Tabla de conversión                  = Predeterminada
IDusuario/NombreTrabajo              = USERID
API llamante                        = LE C Sockets
Modalidad llamante                   = EBCDIC
```

Asegúrese de que las definiciones del archivo (o del conjunto de datos) a las que hace referencia "Conjunto de datos Tcp/Ip local" sean correctas.

Este campo estará en blanco si no utiliza un nombre predeterminado para el archivo resolvente de IP (utilizando el orden de búsqueda de z/OS UNIX). Si es así, añada la sentencia siguiente a `rsed.envvars`, donde `<archivo resolvente>` o `<conjunto datos resolvente>` representa el nombre del archivo resolvente de IP:

```
RESOLVER_CONFIG=<archivo resolvente>
```

o bien

```
RESOLVER_CONFIG='<conjunto datos resolvente>'
```

Bibliografía

Publicaciones a las que se hace referencia

Las publicaciones a las que se hace referencia en este documento son:

Tabla 46. Publicaciones a las que se hace referencia

Título de la publicación	Número de pedido	Referencia	Sitio Web de referencia
Program Directory for IBM Rational Developer for System z	GI11-8298	Developer for System z	http://www-01.ibm.com/support/docview.wss?uid=swg27038517
Program Directory for IBM Rational Developer for System z Host Utilities	GC43-0676	Developer for System z	http://www-01.ibm.com/support/docview.wss?uid=swg27038517
Requisitos previos de IBM Rational Developer for System z	SC43-0674 (SC23-7659)	Developer for System z	http://www-01.ibm.com/support/docview.wss?uid=swg27038517
Guía de inicio rápido de configuración de host de IBM Rational Developer for System z	GI11-8628	Developer for System z	http://www-01.ibm.com/support/docview.wss?uid=swg27038517
Guía de configuración de host IBM Rational Developer for System z	SC11-3660	Developer for System z	http://www-01.ibm.com/support/docview.wss?uid=swg27038517
Guía de referencia de configuración de host de IBM Rational Developer for System z	SC11-7903 (SC14-7290)	Developer for System z	http://www-01.ibm.com/support/docview.wss?uid=swg27038517
Guía del programa de utilidad de configuración de host de IBM Rational Developer for System z	SC11-7871 (SC14-7282)	Developer for System z	http://www-01.ibm.com/support/docview.wss?uid=swg27038517
IBM Rational Developer for System z Messages and Codes	SC14-7497	Developer for System z	http://www-01.ibm.com/support/docview.wss?uid=swg27038517
IBM Rational Developer for System z Answers to common host configuration and maintenance issues	SC14-7373	Developer for System z	http://www-01.ibm.com/support/docview.wss?uid=swg27038517
IBM Rational Developer for System z Common Access Repository Manager Developer's Guide	SC23-7660	Developer for System z	http://www-01.ibm.com/support/docview.wss?uid=swg27038517
Requisitos previos de IBM Rational Developer for System z	SC43-0674 (SC23-7659)	Developer for System z	http://www.ibm.com/software/rational/products/developer/systemz/library/index.html
Guía de inicio rápido de configuración de host de IBM Rational Developer for System z	GI11-8628	Developer for System z	http://www.ibm.com/software/rational/products/developer/systemz/library/index.html

Tabla 46. Publicaciones a las que se hace referencia (continuación)

Título de la publicación	Número de pedido	Referencia	Sitio Web de referencia
Guía del administrador de SCLM Developer Toolkit	SC11-3815-00 (SC23-9801)	Developer for System z	http://www-01.ibm.com/support/docview.wss?uid=swg27038517
Using APPC to provide TSO command services	SC14-7291	Libro blanco	http://www-01.ibm.com/support/docview.wss?uid=swg27038517
Using ISPF Client Gateway to provide CARMA services	SC14-7292	Libro blanco	http://www-01.ibm.com/support/docview.wss?uid=swg27038517
Communications Server IP Configuration Guide	SC31-8775	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
Communications Server IP Configuration Reference	SC31-8776	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
Communications Server IP Diagnosis Guide	GC31-8782	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
Communications Server IP System Administrator's Commands	SC31-8781	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
Communications Server SNA Network Implementation Guide	SC31-8777	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
Communications Server SNA Operations	SC31-8779	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
Cryptographic Services System SSL Programming	SC24-5901	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
DFSMS Macro Instructions for Data Sets	SC26-7408	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
DFSMS Using data sets	SC26-7410	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
Language Environment Customization	SA22-7564	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
Language Environment Debugging Guide	GA22-7560	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
MVS Diagnóstico: Ayudas de servicio y herramientas	GA22-7589	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
MVS Initialization and Tuning Guide	SA22-7591	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
MVS Initialization and Tuning Reference	SA22-7592	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
MVS JCL Reference	SA22-7597	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
MVS Planning APPC/MVS Management	SA22-7599	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
MVS Planning Workload Management	SA22-7602	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
MVS System Commands	SA22-7627	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
Security Server RACF Command Language Reference	SA22-7687	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/

Tabla 46. Publicaciones a las que se hace referencia (continuación)

Título de la publicación	Número de pedido	Referencia	Sitio Web de referencia
Security Server RACF Security Administrator's Guide	SA22-7683	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
TSO/E Customization	SA22-7783	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
TSO/E REXX Reference	SA22-7790	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
UNIX System Services Command Reference	SA22-7802	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
UNIX System Services Planning	GA22-7800	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
UNIX System Services User's Guide	SA22-7801	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
Using REXX and z/OS UNIX System Services	SA22-7806	z/OS 1.13	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
Java™ Diagnostic Guide	SC34-6650	Java 6.0	http://www.ibm.com/developerworks/java/jdk/diagnosis/
Java SDK and Runtime Environment User Guide	/	Java 6.0	http://www-03.ibm.com/servers/eserver/zseries/software/java/
Resource Definition Guide	SC34-6430	CICSTS 3.1	http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html
Resource Definition Guide	SC34-6815	CICSTS 3.2	http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html
Resource Definition Guide	SC34-7000	CICSTS 4.1	https://publib.boulder.ibm.com/infocenter/cicsts/v4r1/index.jsp?topic=/com.ibm.cics.ts.home.doc/library/library_html.html
Resource Definition Guide	SC34-7181	CICSTS 4.2	https://publib.boulder.ibm.com/infocenter/cicsts/v4r2/index.jsp?topic=/com.ibm.cics.ts.home.doc/library/library_html.html
RACF Security Guide	SC34-6454	CICSTS 3.1	http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html
RACF Security Guide	SC34-6835	CICSTS 3.2	http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html
RACF Security Guide	SC34-7003	CICSTS 4.1	https://publib.boulder.ibm.com/infocenter/cicsts/v4r1/index.jsp?topic=/com.ibm.cics.ts.home.doc/library/library_html.html
RACF Security Guide	SC34-7179	CICSTS 4.2	https://publib.boulder.ibm.com/infocenter/cicsts/v4r2/index.jsp?topic=/com.ibm.cics.ts.home.doc/library/library_html.html
Language Reference	SC27-1408	Enterprise COBOL para z/OS	http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html

En este documento se hace referencia a los siguientes sitios Web:

Tabla 47. Sitios Web a los que se hace referencia

Descripción	Sitio Web de referencia
Developer for System z IBM Knowledge Center	http://www-01.ibm.com/support/knowledgecenter/SSQ2R2_9.1.0/com.ibm.etools.getstart.wsentdev.doc/kc_version_welcome_rdz.html
Biblioteca de Developer for System z	http://www-01.ibm.com/support/docview.wss?uid=swg27038517
Página inicial de Developer for System z	http://www-03.ibm.com/software/products/en/developerforsystemz/
Servicio recomendado de Developer for System z	http://www-01.ibm.com/support/docview.wss?rs=2294&context=SS2QJ2&uid=swg27006335
Solicitud de mejora de Developer for System z	https://www.ibm.com/developerworks/support/rational/rfe/
Biblioteca internet de z/OS	http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/
CICSTS IBM Knowledge Center	https://publib.boulder.ibm.com/infocenter/cicsts/v4r1/index.jsp
IBM Tivoli Directory Server	http://www-01.ibm.com/software/tivoli/products/directory-server/
Plug-ins de herramientas de determinación de problemas	http://www-01.ibm.com/software/awdtools/deployment/pdtplugins/
Información de seguridad de Java	http://www.ibm.com/developerworks/java/jdk/security/
Descargar Apache Ant	http://ant.apache.org/
Documentación de keytool de Java	http://java.sun.com/j2se/1.5.0/docs/tooldocs/solaris/keytool.html
Página inicial de soporte de CA	https://support.ca.com/

Publicaciones informativas

Las publicaciones siguientes pueden serle de utilidad para entender aspectos de configuración de los componentes del sistema host obligatorios:

Tabla 48. Publicaciones informativas

Título de la publicación	Número de pedido	Referencia	Sitio web de referencia
ABCs of z/OS System Programming Volume 9 (z/OS UNIX)	SG24-6989	Redbook	http://www.redbooks.ibm.com/
System Programmer's Guide to: Workload Manager	SG24-6472	Redbook	http://www.redbooks.ibm.com/
TCPIP Implementation Volume 1: Base Functions, Connectivity, and Routing	SG24-7532	Redbook	http://www.redbooks.ibm.com/
TCPIP Implementation Volume 3: High Availability, Scalability, and Performance	SG24-7534	Redbook	http://www.redbooks.ibm.com/
TCP/IP Implementation Volume 4: Security and Policy-Based Networking	SG24-7535	Redbook	http://www.redbooks.ibm.com/
Tivoli Directory Server for z/OS	SG24-7849	Redbook	http://www.redbooks.ibm.com/

Glosario

Acción de bloquear

Bloquea un miembro.

Archivo de respuestas

1. Archivo que contiene un conjunto de respuestas predefinidas a preguntas formuladas por un programa y que se utiliza en lugar de entrar dichos valores de uno en uno.
2. Archivo ASCII que se puede personalizar con los datos de instalación y configuración que automatizan una instalación. Durante una instalación interactiva, es necesario entrar los datos de instalación y configuración, pero si se utiliza un archivo de respuestas, la instalación puede continuar sin ningún tipo de intervención.

atributo bidireccional

Tipo de texto, orientación del texto, intercambio numérico e intercambio simétrico.

Base de datos

Conjunto de elementos de datos interrelacionados o independientes, almacenados conjuntamente para servir a una o más aplicaciones.

Biblioteca de carga

Biblioteca que contiene módulos de carga.

Bidireccional (bi-di)

Relativo a scripts como el árabe o el hebreo que generalmente van de derecha a izquierda, excepto los números, que van de izquierda a derecha. Esta definición pertenece al glosario de la Localization Industry Standards Association (LISA).

Compilar

1. En los lenguajes Integrated Language Environment (ILE), convertir las sentencias fuente en módulos que luego se pueden enlazar en programas o programas de servicio.
2. Convertir todo o parte de un programa expresado en un lenguaje de alto nivel en un programa informático expresado en un lenguaje intermedio, ensamblador o lenguaje de máquina.

Conjunto de datos

Unidad principal de almacenamiento y recuperación de datos, que consiste en una colección de datos en una de varias disposiciones prescritas y descritas por la información de control a la que tiene acceso el sistema.

Contenedor

1. En CoOperative Development Environment/400, objeto del sistema que contiene y organiza archivos fuente. Son ejemplos de contenedor una biblioteca de i5/OS o un conjunto de datos particionado MVS.
2. En Java EE, entidad que proporciona a los componentes servicios de gestión del ciclo de vida, de seguridad, de despliegue y de tiempo de ejecución. (Sun) Cada tipo de contenedor (EJB, Web, JSP, servlet, applet y cliente de aplicaciones) también proporciona servicios específicos del componente.
3. En los Servicios BRM, objeto físico utilizado para almacenar y mover medios, como una caja, un estuche o un bastidor.
4. En un servidor de cintas virtual (VTS), receptáculo en el que es posible almacenar uno o más volúmenes lógicos exportados (LVOL). Volumen apilado que contiene uno o más LVOL y que reside fuera de una biblioteca VTS y que se considera como contenedor de dichos volúmenes.
5. Ubicación de almacenamiento físico de los datos. Por ejemplo, un archivo, un directorio o un dispositivo.
6. Columna o fila que se utiliza para disponer el diseño de un portlet o de otro contenedor en una página.
7. Elemento de la interfaz de usuario que contiene objetos. En el gestor de carpetas, objeto que puede contener otras carpetas o documentos.

Depurar

Detectar, diagnosticar y eliminar errores en programas.

Desinstalación silenciosa

Proceso de desinstalación que no envía mensajes a la consola, sino que almacena los mensajes y los errores en archivos de registro después de haberse invocado el mandato de desinstalación.

Estante lateral

Biblioteca que publica las funciones de un programa DLL. Los nombre de entradas y de módulos se almacenan en la biblioteca una vez compilado el código fuente.

ID de acción

Identificador numérico de una acción, entre 0 y 999

Instalación silenciosa

Instalación que no envía mensajes a la consola, sino que almacena los mensajes y los errores en archivos de registro. Además, en la instalación silenciosa se pueden utilizar archivos de respuestas como entrada de datos.

Instancia de repositorio

Proyecto o componentes que existe en un SCM.

Interactive System Productivity Facility (ISPF)

Programa IBM bajo licencia que funciona como editor de pantalla completa y gestor de diálogos. Si se utiliza para escribir programas de aplicaciones, proporciona una manera de generar paneles de pantallas estándar y diálogos interactivos entre el programador de aplicaciones y el usuario del terminal. ISPF consta de cuatro componentes principales: DM, PDF, SCLM, y C/S. El componente DM es el gestor de diálogos, que proporciona servicios a los diálogos y usuarios finales. El componente PDF es el recurso de desarrollo de programas, que proporciona servicios para ayudar al desarrollador de diálogos o de aplicaciones. El componente SCLM es el gestor de bibliotecas de configuraciones de software, que proporciona servicios a los desarrolladores de aplicaciones para gestionar sus bibliotecas de entorno de aplicaciones. El componente C/S es el cliente/servidor, que permite ejecutar ISPF en una estación de trabajo programable, para visualizar los paneles utilizando la función de visualización del sistema operativo de la estación de trabajo, y para integrar herramientas y

datos de la estación de trabajo con las herramientas y datos del host.

Intérprete

Programa que convierte y ejecuta cada instrucción de un lenguaje de programación de alto nivel antes de convertir y ejecutar la siguiente instrucción.

Isomórfico

Cada elemento compuesto (en otras palabras, un elemento que contiene otros elementos) del documento de instancia XML que comienza desde la raíz tiene un y solo un elemento de grupo COBOL correspondiente cuya profundidad de anidación es idéntica a la profundidad de anidación de su equivalente XML. Cada elemento no compuesto (en otras palabras, un elemento que no contiene otros elementos) en el documento de instancia XML que comienza desde la parte superior tiene un y solo un elemento COBOL correspondiente cuya profundidad de anidación es idéntica a la profundidad de anidación de su equivalente XML y cuya dirección de memoria en tiempo de ejecución puede identificarse de forma inequívoca.

Lista de tareas

Lista de procedimientos que se pueden ejecutar mediante un único flujo de control.

Memoria intermedia de error

Parte del almacenamiento utilizado para contener temporalmente la información de salida de errores.

No isomórfico

Correlación simple de elementos COBOL y elementos XML que pertenecen a documentos XML y a grupos COBOL que no son idénticos en la forma (no isomórficos). También se puede crear una correlación no isomorfa entre elementos no isomórficos de estructuras isomórficas.

Nombre de shell

Nombre de la interfaz de shell.

Pasarela

1. Componente middleware entre Internet y los entornos de intranet durante las invocaciones de servicios Web.

2. Software que proporciona servicios entre los puntos finales y el resto del entorno Tivoli.
3. Componente del protocolo de voz por Internet, que proporciona un puente entre VoIP y los entornos de circuitos conmutados.
4. Dispositivo o programa utilizado para conectar redes o sistemas con diferentes arquitecturas de red. Los sistemas pueden tener distintas características, como distintos protocolos de comunicaciones, distinta arquitectura de red o distintas políticas de seguridad, en cuyo caso la pasarela adquiere un rol de conversión, así como un rol de conexión.

Perspectiva

Grupo de vistas que muestran los diversos aspectos de los recursos del entorno de trabajo. El usuario del entorno de trabajo puede pasar de una perspectiva a otra, en función de la tarea que esté realizando, y personalizar la disposición de las vistas y editores dentro de la perspectiva.

Perspectiva

Proporciona una interfaz para gestionar sistemas remotos utilizando convenciones similares a ISPF.

Petición de construcción

Petición procedente del cliente para realizar una transacción de construcción.

RAM Gestor de acceso a repositorios

Repositorio

1. Área de almacenamiento para los datos. Cada repositorio tiene un nombre y un tipo de elemento de negocio asociado. Por defecto, el nombre será el mismo que el nombre del elemento de negocio. Por ejemplo, un repositorio de facturas se llamará Facturas. Hay dos tipos de repositorios de información: local (específico del proceso) y global (reutilizable).
2. Conjunto de datos VSAM en el que se almacenan los estados de los procesos BTS. Cuando un proceso no se está ejecutando bajo el control de BTS, su estado (y los estados de sus

actividades subordinadas) se conservan escribiéndose en un conjunto de datos de repositorio. Los estados de todos los procesos de un tipo de proceso en particular (y los de sus instancias de actividad) se almacenan en el mismo conjunto de datos del repositorio. Es posible escribir registros de varios tipos de proceso en el mismo repositorio.

3. Área de almacenamiento persistente del código fuente y de otros recursos de las aplicaciones. En un entorno de programación en equipo, el repositorio compartido permite que varios usuarios accedan a los recursos de la aplicación.
4. Recopilación de información acerca de los gestores de cola que son miembros de un clúster. Esta información incluye nombres de gestores de colas, sus ubicaciones, sus canales, qué colas hospedan, etc.

Script de shell

Archivo que contiene mandatos que la shell puede interpretar. El usuario escribe el nombre del archivo de script en el indicador de mandatos de la shell para hacer que la shell ejecute los mandatos del script.

Sección de enlace

Sección de la división de datos de una unidad activada (un programa al que se llame o un método invocado) que describe los elementos de datos disponibles de la unidad que lo activa (un programa o un método). A estos elementos de datos les puede hacer referencia la unidad activada y la unidad que activa.

Servidor de aplicaciones

1. Programa que maneja todas las operaciones de aplicación entre los sistemas basados en navegador y las aplicaciones o bases de datos de negocio de fondo de una organización. Hay una clase especial de servidores de aplicación basados en Java que cumplen el estándar Java EE. El código Java EE puede portarse fácilmente entre estos servidores de aplicaciones. Puede soportar JSP y

servlets para contenido Web dinámico y EJB para transacciones y acceso a bases de datos.

2. Destino de una petición procedente de una aplicación remota. En el entorno DB2, la función de servidor de aplicaciones la proporciona el servicio de datos distribuidos, y sirve para acceder a datos de DB2 desde aplicaciones remotas.
3. En una red distribuida, programa servidor que proporciona el entorno de ejecución de un programa de aplicación.
4. Destino de una petición procedente de un peticionario de aplicación. El sistema de gestión de bases de datos (DBMS) en el sitio del servidor de aplicaciones proporciona los datos solicitados.
5. Software que gestiona la comunicación con el cliente que solicita un activo y consultas del gestor de contenido.

Sesión de depuración

Actividades de depuración que tienen lugar entre el momento en que un desarrollador inicia un depurador y el momento en que el desarrollador sale de él.

Shell Interfaz de software entre los usuarios y el sistema operativo, que interpreta mandatos e interacciones del usuario y los comunica al sistema operativo. Cada sistema puede tener varias capas de shells para los diversos niveles de interacción de los usuarios.

Sistema de archivos remoto

Sistema de archivos que reside en un servidor o sistema operativo independiente.

Sistema remoto

Cualquier otro sistema en la red con el que puede comunicarse su sistema.

Transacción de construcción

Trabajo iniciado en MVS para realizar construcciones después haberse recibido una petición de construcción procedente del cliente.

URL Localizador uniforme de recursos.

Vista Consola de salida

Visualiza la salida de un proceso y permite proporcionar entrada de teclado para un proceso.

Vista de definición de datos

Contiene una representación local de bases de datos y de sus objetos y proporciona características para manipular estos objetos y exportarlos a una base de datos remota

Vista Navegador

Vista jerárquica de los recursos que hay en el entorno de trabajo.

Vista repositorios

Muestra la ubicación de los repositorios CVS que se han añadido al entorno de trabajo.

Vista Salida

Muestra los mensajes, parámetros y resultados relacionados con los objetos con los que se esté trabajando.

Vista Servidores

Visualiza una lista de todos los servidores, así como las configuraciones asociadas a ellos.

Avisos

© Copyright IBM Corporation 1992, 2013.

Derechos restringidos de los usuarios del Gobierno de EE. UU. - El uso, la reproducción o la divulgación están sujetos a las restricciones establecidas en el contrato GSA ADP Schedule Contract con IBM Corp.

Esta información se ha desarrollado para productos y servicios ofrecidos en los Estados Unidos de América.

Es posible que IBM no ofrezca en otros países los productos, servicios o características que se describen en este documento. El representante local de IBM le puede informar acerca de los productos y servicios que actualmente están disponibles en su localidad. Las referencias hechas a productos, programas o servicios de IBM no pretenden afirmar ni dar a entender que únicamente puedan utilizarse dichos productos, programas o servicios de IBM. Puede utilizarse en su lugar cualquier otro producto, programa o servicio funcionalmente equivalente que no vulnere ninguno de los derechos de propiedad intelectual de IBM. No obstante, es responsabilidad del usuario evaluar y verificar el funcionamiento de cualquier producto, programa o servicio que no sea de IBM.

IBM puede tener patentes o solicitudes de patente pendientes de aprobación que cubran alguno de los temas tratados en este documento. La posesión de este documento no le otorga ninguna licencia sobre dichas patentes. Puede enviar consultas sobre licencias por escrito a:

*IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
Estados Unidos de América*

Para consultas sobre licencias relativas a la información de juego de caracteres de doble byte (DBCS), póngase en contacto con el departamento de propiedad intelectual de IBM en su país o envíe las consultas, por escrito, a:

*Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japón*

El párrafo que sigue no se aplica al Reino Unido ni a ningún otro país en el que tales disposiciones sean incompatibles con la legislación local:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROPORCIONA ESTA PUBLICACIÓN "TAL CUAL", SIN GARANTÍA DE NINGUNA CLASE, YA SEA EXPLÍCITA O IMPLÍCITA, INCLUIDAS, PERO SIN LIMITARSE A ELLAS, LAS GARANTÍAS IMPLÍCITAS DE NO VULNERACIÓN, DE COMERCIALIZACIÓN O IDONEIDAD PARA UN PROPÓSITO DETERMINADO. Algunas legislaciones no contemplan la declaración de limitación de responsabilidad, ni implícitas ni explícitas, en determinadas transacciones, por lo que cabe la posibilidad de que esta declaración no se aplique en su caso.

Esta información puede contener imprecisiones técnicas o errores tipográficos. La información incluida en este documento está sujeta a cambios periódicos; estos cambios se incorporarán en nuevas ediciones de la publicación. IBM puede efectuar mejoras y/o cambios en los productos y/o programas descritos en esta publicación en cualquier momento y sin previo aviso.

Cualquier referencia de esta información a sitios web que no sean de IBM se proporciona únicamente como ayuda y no se consideran en modo alguno como aprobados por IBM. Los materiales de dichos sitios web no forman parte de los materiales para este producto de IBM y el uso de dichos sitios web corre a cuenta y riesgo del Cliente.

IBM puede utilizar o distribuir la información que usted le suministre del modo que IBM considere conveniente sin incurrir por ello en ninguna obligación para con usted.

Los licenciarios de este programa que deseen obtener información acerca de él con el fin de: (i) intercambiar la información entre los programas creados independientemente y otros programas (incluido este) y (ii) utilizar mutuamente la información que se ha intercambiado, deben ponerse en contacto con:

*Intellectual Property Dept. for Rational Software
IBM Corporation
Silicon Valley Lab
555 Bailey Avenue
San Jose, CA 95141-1003
Estados Unidos de América*

Dicha información puede estar disponible, sujeta a los términos y condiciones apropiados, incluyendo en algunos casos el pago de una cantidad.

IBM proporciona el programa bajo licencia descrito en este documento, así como todo el material bajo licencia disponible, según los términos del Acuerdo de Cliente de IBM, del Acuerdo Internacional de Programas bajo Licencia de IBM o de cualquier otro acuerdo equivalente entre ambas partes.

Los datos de rendimiento que se indican en este documento se han obtenido en un entorno controlado. Por consiguiente, es posible que los resultados que se obtengan en otros entornos operativos sean notablemente distintos. Es posible que algunas mediciones se hayan tomado en sistemas de nivel de desarrollo y no existe ningún tipo de garantía de que dichas mediciones sean las mismas en sistemas disponibles para el público en general. Además, es posible que algunas mediciones se hayan estimado mediante extrapolación. Los resultados reales pueden variar. Los usuarios de este documento deberán verificar los datos aplicables para su entorno específico.

La información concerniente a productos no IBM se ha obtenido de los suministradores de dichos productos, de sus anuncios publicados o de otras fuentes de información pública disponibles. IBM no ha comprobado dichos productos y no puede afirmar la exactitud en cuanto a rendimiento, compatibilidad u otras características relativas a productos no IBM. Las consultas acerca de las posibilidades de los productos que no son de IBM deben dirigirse a las personas que los suministran.

Todas las declaraciones relacionadas con la dirección o intención futuras de IBM están sujetas a cambio o retirada sin previo aviso, y únicamente representan objetivos.

Esta información contiene ejemplos de datos e informes utilizados en operaciones comerciales diarias. Para ilustrarlos de la forma más completa posible, los ejemplos incluyen nombres de personas, empresas, marcas y productos. Todos estos nombres son ficticios y cualquier parecido con los nombres y direcciones utilizados por una empresa real es mera coincidencia.

Licencia de copyright

Esta información contiene programas de aplicación de ejemplo en lenguaje fuente, que ilustran las técnicas de programación en diversas plataformas operativas. Puede copiar, modificar y distribuir los programas de ejemplo de cualquier forma, sin tener que pagar a IBM, con intención de desarrollar, utilizar, comercializar o distribuir programas de aplicación que estén en conformidad con la interfaz de programación de aplicaciones (API) de la plataforma operativa para la que están escritos los programas de ejemplo. Los ejemplos no se han probado minuciosamente bajo todas las condiciones. Por lo tanto, IBM no puede garantizar ni dar por sentada la fiabilidad, la facilidad de mantenimiento ni el funcionamiento de los programas. Los programas de ejemplo se proporcionan "TAL CUAL", sin ningún tipo de garantía. IBM no se hace responsable de los daños que se hayan podido causar debido al uso de los programas de ejemplo.

Cada copia o parte de estos programas de ejemplo o cualquier trabajo derivado debe incluir un aviso de copyright como el siguiente:

© (nombre de la empresa) (año). Partes de este código se derivan de IBM Corp. Sample Programs. © Copyright IBM Corp. 1992, 2013.

Si está visualizando esta información en formato de copia software, es posible que las fotografías y las ilustraciones en color no aparezcan.

Consideraciones sobre políticas de privacidad

Los productos de software de IBM, incluyendo el software como soluciones de servicio, ("Ofertas de software") pueden utilizar cookies u otras tecnologías para recopilar información de utilización del producto, para ayudar a mejorar la experiencia del usuario final, para adaptar interacciones con el usuario final o a otros efectos. En muchos casos las Ofertas de software no recopilan información que permita la identificación personal. Algunas de nuestras Ofertas de software pueden ayudarle a recopilar información que permite la identificación personal. Si esta Oferta de software utiliza cookies para recopilar información que permite la identificación personal, a continuación se expondrá información específica sobre el uso de cookies por parte de esta oferta.

Esta Oferta de software no utiliza cookies ni otras tecnologías para recopilar información identificable personalmente.

Marcas registradas

IBM, el logotipo de IBM e ibm.com son marcas comerciales o marcas registradas de International Business Machines Corp., registradas en muchas jurisdicciones de todo el mundo. Otros nombres de productos y servicios pueden ser marcas

registradas de IBM u otras empresas. Hay una lista actualizada de marcas registradas de IBM en la web "Copyright and trademark information" en www.ibm.com/legal/copytrade.shtml.

Documentación de términos y condiciones para el producto

Aplicabilidad

Estos términos y condiciones son adicionales a los términos de uso para el sitio web de IBM.

Utilización personal

Puede reproducir estas publicaciones para su uso personal, no comercial suponiendo que se conserven todos los avisos de propiedad. No puede distribuir ni mostrar estas publicaciones o partes de ellas ni realizar trabajos derivados de ellas sin el consentimiento expreso de IBM.

Utilización comercial

Puede reproducir, distribuir y mostrar estas publicaciones solamente dentro de su empresa suponiendo que se conserven todos los avisos de propiedad. No puede realizar trabajos derivados de estas publicaciones ni reproducir, distribuir o mostrar estas publicaciones o partes de ellas fuera de su empresa sin el consentimiento expreso de IBM.

Derechos

Excepto lo expresamente otorgado en este permiso, no se otorga ningún otro permiso, licencia o derecho, ya sea expresa o implícitamente, sobre las publicaciones o sobre cualesquiera información, datos, software u otro tipo de propiedad intelectual contenida dentro.

IBM se reserva el derecho de retirar los permisos otorgados aquí siempre que, según su criterio, la utilización de las publicaciones vaya en detrimento de sus intereses o, según determine IBM, las instrucciones indicadas más arriba no se sigan adecuadamente.

No puede descargar, exportar ni reexportar esta información si no es en total conformidad con las leyes y regulaciones aplicables, incluyendo todas las leyes y regulaciones de exportación de Estados Unidos de América.

IBM NO GARANTIZA EL CONTENIDO DE ESTAS PUBLICACIONES. LAS PUBLICACIONES SE PROPORCIONAN "TAL-CUAL" Y SIN GARANTÍA DE NINGUNA CLASE, YA SEA EXPRESA O IMPLÍCITA, INCLUYENDO PERO SIN LIMITARSE A ELLAS, LAS GARANTÍAS DE COMERCIALIZACIÓN, NO VULNERACIÓN E IDONEIDAD PARA UN PROPÓSITO DETERMINADO.

Licencia de copyright

Esta información contiene programas de aplicación de ejemplo en lenguaje fuente, que ilustran las técnicas de programación en diversas plataformas operativas. Puede copiar, modificar y distribuir los programas de ejemplo de cualquier forma, sin tener que pagar a IBM, con intención de desarrollar, utilizar, comercializar o distribuir programas de aplicación que estén en conformidad con la interfaz de programación de aplicaciones (API) de la plataforma operativa para la que están

escritos los programas de ejemplo. Los ejemplos no se han probado minuciosamente bajo todas las condiciones. Por lo tanto, IBM no puede garantizar ni dar por sentada la fiabilidad, la facilidad de mantenimiento ni el funcionamiento de los programas. Los programas de ejemplo se proporcionan "TAL CUAL", sin ningún tipo de garantía. IBM no se hace responsable de los daños que se hayan podido causar debido al uso de los programas de ejemplo.

Reconocimientos de marcas registradas

IBM, el logotipo de IBM e ibm.com son marcas comerciales o marcas registradas de International Business Machines Corp., registradas en muchas jurisdicciones de todo el mundo. Otros nombres de productos y servicios pueden ser marcas registradas de IBM u otras empresas. Hay una lista actual de marcas registradas de IBM disponible en la web en www.ibm.com/legal/copytrade.shtml

Adobe y PostScript son marcas registradas de Adobe Systems Incorporated.

Cell Broadband Engine - Sony Computer Entertainment Inc.

Rational es una marca registrada de International Business Machines Corporation y Rational Software Corporation, en los Estados Unidos o en otros países.

Intel, Intel Centrino, Intel SpeedStep, Intel Xeon, Celeron, Itanium y Pentium son marcas registradas de Intel Corporation, en los Estados Unidos y/o en otros países.

IT Infrastructure Library es una marca registrada de Central Computer and Telecommunications Agency

ITIL es una marca registrada de The Minister for the Cabinet Office

Linear Tape-Open, LTO y Ultrium son marcas registradas de HP, IBM Corp. y Quantum

Linux es una marca registrada de Linus Torvalds

Microsoft, Windows y el logotipo de Windows son marcas registradas de Microsoft Corporation en los Estados Unidos y/o en otros países.

Java y todas las marcas registradas y logotipos basados en Java son marcas registradas de Sun Microsystems, Inc. en los Estados Unidos y en otros países.

UNIX es una marca registrada de The Open Group en los Estados Unidos y/o en otros países.

Índice

Caracteres Especiales

.dstoreMemLogging 182
.dstoreTrace 182
_RSE_PORTRANGE 22
/var/rdz/pushtoclient/*install 145, 148

A

acceso a las bibliotecas del sistema,
Mejorar 123
acceso al depurador integrado,
Definir 57
Acceso condicional a los archivos de
spool 29
Acciones condicionales en los
trabajos 26
Acciones en trabajos - limitaciones de
ejecución 28
ACEE, gestionado 44
ACK, retardado 66
ACK retardado 66
activación 135
activación de la política AT-TLS 222
activación de la salida de usuario 165
actualizaciones de seguridad de
AT-TLS 220
actualizar privilegios, no administradores
del sistema 17
acuse de recibo, retardado 66
ADNJSPAU, Programa de utilidad
administrativo 155
almacén de claves con keytool,
Crear 213
almacenamiento de ejemplo, análisis de
utilización 101
Almacenamiento dinámico Java de
tamaño fijo 125
almacenamiento en memoria caché,
ACEE 44
almacenamiento en memoria caché
ACEE 44
alteración temporal del comportamiento
de TCP/IP predeterminado 66
análisis de utilización, almacenamiento
de ejemplo 101
anillo de claves, Crear con RACF 203
Anotar y configurar el análisis mediante
FEKLOGS 181
APF, autorización 195
aplicaciones, Desarrollo de 124
AQEZPCM 21
archivos controlados por programa UNIX
para RSE, Definir 55
archivos controlados por programa z/OS
UNIX para RSE, Definir 55
archivos de configuración, Developer for
System z 44

Archivos de configuración de resolvente
base 227
archivos de configuración diferentes con
niveles de software idénticos 176
archivos de registro
.dstoreMemLogging 182
.dstoreTrace 182
audit.log 182
fa.log 182
fekfivpi.log 182
fekfivps.log 182
ffs.log 182
ffsget.log 182
ffsput.log 182
lock.log 182
rmt_class_loader.cache.jar 182
rsecomm.log 182
rsedaemon.log 182
rseserver.log 182
serverlogs.count 182
stderr.log 182
stdout.log 182
archivos de registro de agrupaciones de
hebras RSE
audit.log 184
rsedaemon.log 184
rseserver.log 184
serverlogs.count 184
stderr*.log 184
stdout*.log 184
archivos de registro de daemon RSE
audit.log 184
rsedaemon.log 184
rseserver.log 184
serverlogs.count 184
stderr*.log 184
stdout*.log 184
archivos de spool, Acceso condicional
a 29
Archivos de vuelco 188
archivos ISPF.conf, utilizar con varias
configuraciones 173
ASCHPMxx
MAX 113
ASSIZEMAX 51
atributo del sistema de archivos,
SETUID 193
Atributo del sistema de archivos
SETUID 193
audit.action, salida de usuario 168
audit.log 183
autenticación, configurar SSL y
X.509 201
autenticación, gestor de depuración 21
Autenticación de cliente mediante
certificados X.509 32
Autenticación del gestor de
depuración 21
Autenticación del supervisor de trabajos
JES 21
autenticación por daemon RSE 35

autenticación por software de
seguridad 34
autenticación x.509, configurar 201
automatizada, sincronización 177
Autorización de APF
FEK.SFEKAUTH 58
Autorización de control de
programa 194

B

base de datos de claves, Crear con
gskkyman 210
bibliotecas controladas por programa
MVS para RSE, Definir 52
Bibliotecas de tiempo de ejecución de
Language Environment 124
bibliotecas del sistema, Mejorar el acceso
a 123
bit de permanencia, disponibilidad de
módulo de carga MVS para z/OS
UNIX 195
bits de permiso, z/OS UNIX 193
Bits de permiso de z/OS UNIX 193
BPX.SUPERUSER, permiso de perfil 41
BPXPRMxx 119
INADDRANYCOUNT 112
MAXASSIZE 51, 111, 198
MAXFILEPROC 111
MAXMMAPAREA 111
MAXPROCSYS 109, 199
MAXPROCUSER 109, 199
MAXSOCKETS 112
MAXTHREADS 109
MAXTHREADTASKS 109
MAXUIDS 111, 200

C

Capa de sockets seguros, Cifrado de
comunicaciones mediante 22
Capa de sockets seguros, Configurar 201
características de salida de usuario 165
cargas de trabajo, Gestión de 125
CARMA, rastreo 191
CARMA, Registro 186
CEE.SCEELPA
SYS1.PARMLIB(LPALSTxx) 124
Certificado X.509 20
certificado X.509 de terceros 20
certificados, autenticación de cliente
mediante X.509 32
certificados X.509, autenticación de cliente
mediante 32
certificados y claves privadas, decida
dónde almacenar 202
CICS, registro de instalación de
recursos 153
CICSplex SM Business Application
Services (BAS) 152

- cifrado, SSL o TLS 202
- Cifrado de comunicaciones mediante SSL 22
- Cifrado de comunicaciones mediante TLS 22
- cifrado mediante TLS, Comunicación 22
- clasificación de carga de trabajo, WLM 73
- CLASSPATH 177
- clonar la configuración RSE
 - existente 205
- cobertura de código, registro 187
- COBOL
 - comprobación remota 192
- coexistencia, actualizar rsed.envvars para habilitar la coexistencia 205
- compartimiento de clases, habilitar en máquinas virtuales Java (JVM) 127
- compartimiento de clases entre máquinas virtuales Java (JVM) 126
- Comportamiento de TCP/IP, alteración temporal del valor predeterminado 66
- comportamiento TCP/IP predeterminado, alteración temporal 66
- comprender Developer for System z 3
- comprobación de POE 23, 36
- comprobación de puerto de entrada 23, 36
- comprobación de seguridad, Mejorar el rendimiento de la 125
- comunicación, cifrada con 155
- comunicación, cifrado SSL/TLS 30
- comunicación cifrada
 - Depurador integrado 31
- comunicación cifrada, SSL 43, 155
- comunicación cifrada, SSL/TLS 30
- Comunicación cifrada con SSL 43, 155
- comunicación cifrada con SSL/TLS 30
- Comunicación externa 64
- comunicación externa con puertos especificados, limitar 22
- Comunicación interna 64
- comunicaciones mediante SSL, Cifrado 22
- concatenación de grupos 136
- condicional a los archivos de spool, Acceso 29
- conexión, Seguridad 21
- conexión de configuración de host de Capa de sockets seguros, Probar la 207
- conexión de configuración de host SSL, Probar la 207
- conexión de host, Emulador 200
- conexión rehusada 199
- Conexión rehusada 199
- configuración, idéntica en todo un sysplex 175
- configuración, Resolución de problemas 181
- configuración AT-TLS, PROFILE.TCPIP 216
- configuración de ejemplo 118
 - definir límites 119
 - determinar los límites mínimos 118
 - recuento de agrupaciones de hebras 118

- configuración de ejemplo, selección de grupo basada en LDAP 143
- configuración de ejemplo, selección de grupo basada en SAF 147
- configuración de grupo LDAP
 - inicial 144
- configuración del agente de política 217
- Configuración del supervisor de trabajos JES
 - GEN_CONSOLE_NAME 29
- Configuración idéntica en todo un sysplex 175
- configurar AT-TLS 215
- configurar syslogd 216
- conjunto de datos, Definir perfiles 57
- consideraciones acerca de los ajustes 81
- Consideraciones de CICSTS 151
- consideraciones de salida de usuario xvi, 165
- Consideraciones relativas a la seguridad 19
- Consideraciones sobre el rendimiento 123
- consideraciones sobre envío a cliente 131
- Consideraciones sobre LDAP 66
- Consideraciones sobre WLM xv, 73
- Consideraciones TLS v1.2 219
- consultar una Lista de certificados revocados (CRL)
 - rsed.envvars 33
 - variables de entorno CRL 33
- contraseña e ID de usuario 20
- contraseña para una sola vez e ID de usuario 20
- control de auditoría
 - _RSE_HOST_CODEPAGE 24
 - daemon.log 24
 - enable.audit.log 24
 - opciones de audit.* 24
- Control de configuración del cliente 134
- Control de versión del cliente 135
- controladas por programa MVS para RSE, Definir bibliotecas 52
- conversión, Tablas de 228

D

- Daemon de bloqueo 13
- daemon de bloqueo (LOCKD) 4
- Daemon RSE 64
- daemon RSE, autenticación por 35
- daemon RSE, Registro 184
- daemon RSE (RSED) 4
- daemon RSE y registro de auditoría 24
- datos de auditoría
 - acciones anotadas 25
- definición de seguridad 148
- definiciones de recursos, varios 112
- definiciones de recursos CICS, administrador 151
- definiciones de recursos CICS, desarrollador 151
- definiciones de recursos clave 108
 - rsed.envvars 108
 - SYS1.PARMLIB(BPXPRMxx) 109
- Definiciones de seguridad 47

- definiciones de seguridad, Lista de comprobación 47
- definiciones de varios recursos 112
 - FEJCNFG 112
 - SYS1.PARMLIB(ASCHPMxx) 113
 - SYS1.PARMLIB(IEASYsxx) 113
 - SYS1.PARMLIB(IVTPRMxx) 113
 - tarjeta EXEC, servidor JCL 112
- definiciones disponibles para el resolvente 231
- Definiciones locales disponibles para el resolvente 231
- Definir acceso al depurador integrado 57
- Definir archivos controlados por programa z/OS UNIX para el servidor 55
- Definir bibliotecas controladas por programa MVS para RSE 52
- Definir comprobación de puerto de entrada para RSE 36
- Definir el soporte de PassTicket para RSE 53
- definir permiso de acceso de archivos z/OS UNIX para RSE 54
- Definir servidor RSE como z/OS UNIX seguro 51
- Dependencia del nombre de host 225
- depuración, transacción CICS 162
- depuración de transacción CICS 162
- depurador, integrado 10
- depurador integrado 10
- Depurador integrado
 - comunicación cifrada 31
- Desarrollo de aplicaciones 124
- Developer for System z, comprender 3
- Developer for System z, visión general de componentes
 - representación gráfica 4
- dirección de host no resuelta, resolvente TCP/IP
 - lock.log 231
- dónde almacenar los certificados y claves privadas 202

E

- editor de definiciones de recurso CICS (CRD), Gestor de despliegue de aplicaciones 151
- Ejecutar varias instancias 175
- Emulador de conexión de host 200
- enlace de espacio de trabajo 137
- entorno TSO, Personalizar el 171
- entorno UNIX, Orden de búsqueda utilizado en 227
- entorno z/OS UNIX, Orden de búsqueda utilizado en 227
- envío a cliente 37
- error de falta de memoria 200
- espacio de direcciones, Tamaño 198
- espacio de disco, máquinas virtuales Java (JVM) 128
- Esquema de LDAP 141
- establecer objetivos, WLM 75
- estimado de tamaño, directrices 100

- estructura de directorios, z/OS UNIX
 - representación gráfica 15
- Estructura de directorios de z/OS UNIX
 - representación gráfica 15
- exec de asignación, utilizar 173
- exec REXX de z/OS UNIX 167
- externa, Comunicación 64
- extremo Envío a cliente, añadir a LDAP 143

F

- fa.log 182
- FEJCNFG 64, 119, 179
 - CONSOLE_NAME 28
 - MAX_THREADS 112
- FEJCNFG, supervisor de trabajos JES 44
- FEKAPPL 21
- fekfivpc, registro de prueba IVP
 - fekfivpc.log 187
- fekfivpc.log 183
- fekfivpi.log 183
- fekfivpi.log, registro de prueba IVP 187
- fekfivps.log 183
- fekfivps.log, registro de prueba IVP 187
- FEKLOGS, anotar y configurar el análisis mediante 181
- FEKRACF, definiciones de seguridad 47
- fekrivp 195
- ffs.log 182
- ffsget.log 182
- ffsput.log 182
- Flujo de conexión 8
 - representación gráfica 8
- Flujo de daemon de bloqueo
 - representación gráfica 13
- frase de contraseña e ID de usuario 20
- Funciones de cliente, alterar 36

G

- GATE, desecho 43
- Gestión de cargas de trabajo 125
- gestor de carga de trabajo 73
- Gestor de despliegue de aplicaciones (ADM) 4
- Gestor de despliegue de aplicaciones, editor de definiciones de recurso CICS 151
- Gestor de despliegue de aplicaciones, personalizar 151
- Gestor de despliegue de aplicaciones, servidor de definiciones de recurso CICS 151
- grupo LDAP, configuración inicial 144
- Grupos LDAP, adición de desarrolladores 144
- gskkyman, Crear una base de datos de claves con 210

H

- habilitar el compartimiento de clases, máquinas virtuales Java (JVM) 127
- hosts locales, Tablas de 228

- ID de usuario, variable, ejecutar con 166
- ID de usuario variable, ejecutar con 166
- ID de usuario y contraseña 20
- ID de usuario y contraseña para una sola vez 20
- ID de usuario y frase de contraseña 20
- idéntico nivel de software, Archivos de configuración diferentes con 176
- IEASYsxx 120
 - MAXUSER 113, 200
- información de configuración, orden de búsqueda de 226
- información de retorno de errores, Rastreo 192
- inicio, Requisitos de JCL 198
- instalación de SMP/E, bit de permanencia 195
- interfaz de servicio Web 152
- interfaz RESTful 152
- interfaz RESTful frente a interfaz de servicio Web 152
- interna, Comunicación 64
- introducción, consideraciones sobre el Envío a cliente 131
- ISP.SISPLOAD
 - pasarela de cliente TSO/ISPF de ISP 52
- ISPF, utilizar varios ejecutables de asignación 173
- ISPF.conf, Personalización básica 172
- IVP fekfivpi, Registro de prueba 187
- IVTPRMxx
 - ECSA MAX 113
 - FIXED MAX 113

J

- Java, vuelcos 188
- JAVA_DUMP_TDUMP_PATTERN 189
- Java Virtual Machines (JVM), compartimiento de clases entre 126
- JCL de inicio, Requisitos 198
- JES, Definir la seguridad de mandatos 55
- JES, Seguridad 26
- JES JMON
 - GEN_CONSOLE_NAME 29
- JMON 56, 179
- JVM, compartimiento de clases entre 126

K

- keytool, Crear un almacén de claves con 213

L

- Language Environment, bibliotecas de tiempo de ejecución 124
- liberar un bloqueo
 - RSE, mandato de modificar cancelación 14
- LIMIT_COMMANDS 27

- LIMIT_VIEW 29
- Limitaciones de ejecución, Acciones en trabajos 28
- limitar la comunicación externa, puertos especificados 22
- límite de tamaño, almacenamiento dinámico Java 98
- límite de tamaño, espacio de direcciones 99
- límite de tamaño de almacenamiento dinámico, Java 98
- Límite de tamaño de almacenamiento dinámico Java 98
- límite de tamaño de espacio de direcciones 99
- límites de tamaño de memoria caché, máquinas virtuales Java (JVM) 127
- Límites del sistema 199
- Lista de certificados revocados (CRL), consulta
 - rsed.envvars 33
 - variables de entorno CRL 33
- locales, Tablas de hosts 228
- lock.log 182
- logfile, seguridad 39
- logon.action, salida de usuario 168
- LPALSTxx 124

M

- mandatos de seguridad de gran utilidad
 - ADDGROUP 17
 - ALTUSER 17
 - CONNECT 17
- mandatos de z/OS UNIX de gran utilidad
 - chgrp 17
 - chmod 17
 - chown 17
 - ls 17
- mandatos JES, definir la seguridad 55
- Mejorar el acceso a las bibliotecas del sistema 123
- Mejorar el rendimiento de la comprobación de seguridad 125
- mensajes, programa de utilidad administrativo 160
- mensajes de consola, salida de usuario 166
- mensajes del programa de utilidad administrativo 160
- metadatos, envío a cliente 133
- metadatos Envío a cliente 133
- método de acceso de Pasarela de cliente TSO/ISPF, Utilizar el 172
- métodos, Autenticación 20
- Métodos de acceso, TSO 171
- Métodos de autenticación 20
- MVS, vuelcos 188
- MVS para RSE, Definir bibliotecas controladas por programa 52

N

netstat 196
nivel de software, idéntico en archivos de configuración diferentes 176
niveles de software idénticos con archivos de configuración diferentes 176
no administradores del sistema, actualizar privilegios 17
nombre de host, Dependencia del 225
nombres de host, aplicar a Developer for System z 229
notas de migración, programa de utilidad administrativo 159

O

objetivos, establecer en WLM 75
OFF.REMOTECOPY.MVS 37
OMVS, Definir segmento 50
opción Java Xquickstart 126
Orden de búsqueda, entorno z/OS UNIX 227
orden de búsqueda de la información de configuración 226
OutOfMemoryError 200

P

Pasarela de cliente TSO/ISPF, Utilizar el método de acceso de 172
pasarela de cliente TSO/ISPF de ISP
ISP.SISPLOAD 52
pasos de configuración 138
PassTicket para RSE, Definir el soporte 53
PassTickets, utilizar 23
perfil de seguridad, Limitaciones almacenadas en 198
perfiles de conjunto de datos, Definir 57
perfiles ISPF existentes, Utilizar 172
periodo de gracia, rechazo de cambios 149
permiso de acceso de archivos z/OS UNIX, definir para RSE 54
permiso de perfil BPX.SUPERUSER 41
permisos de clase, UNIXPRIV 41
personalización - ISPF.conf, 172
Personalizar el entorno TSO 171
personalizar el Gestor de despliegue de aplicaciones 151
Política AT-TLS 218
PORTRANGE 197
Probar la conexión de configuración de host SSL 207
procesamiento de auditoría modify switch 25
PROFILE.TCPIP, configuración AT-TLS 216
Programa de utilidad administrativo, notas de migración 159
programa de utilidad administrativo para administradores CICS funciones proporcionadas 155
programas de utilidad de gestión de memoria caché, máquinas virtuales Java (JVM) 128

propietarios de tareas 7
protección de aplicaciones para RSE, Definir 54
proyectos, basados en host 149
proyectos basados en host 149
prueba IVP de fekfivpi, Registro 187
Publicaciones a las que se hace referencia 233
puertos, CARMA y TCP/IP 65
puertos, limitar la comunicación externa a específicos 22
Puertos CARMA y TCP/IP 65
Puertos TCP/IP 63
puertos TCP/IP, representación gráfica 63
Puertos TCP/IP reservados 196
puntos de salida, disponibles 168
puntos de salida de usuario, disponibles 168
pushtoclient.properties 145, 148

Q

quickstart, opción Java (-Xquickstart) 126

R

RACF
permisos 58
RACF, Crear un anillo de claves con 203
rastreo 190
rastreo, RSE 190
Rastreo de CARMA 191
Rastreo de información de retorno de errores 192
Rastreo del supervisor de trabajos JES 190
Rastreo RSE 190
rechazo de cambios, periodo de gracia 149
Recuento de espacios de direcciones 83
Recuento de hebras 89, 94
Recuento de procesos 86
recursos CICS, registro de instalación 153
red, supervisar 117
referenciadas, publicaciones 233
región propietaria Web 152
regiones de conexión, primarias frente a no primarias 152
regiones de conexión primarias frente a no primarias 152
registro, agrupaciones de hebras 184
registro, Gestor de depuración 184
registro de agrupaciones de hebras 184
registro de auditoría, gestionadas por el daemon RSE 24
Registro de CARMA rsecomm.log 186
registro de cobertura de código 187
Registro de Common Access Repository Manager (CARMA) 186
Registro de instalación de recursos CICS 153
registro de prueba, fekfivpc IVP 187
Registro de prueba IVP fekfivpi.log 187
fekfivps.log 187
Registro de prueba IVP de fekfivpi fekfivpi.log 187
registro de revisión de código 187
Registro de SCLM Developer Toolkit rsecomm.log 186
Registro de usuario de RSE .dstoreMemLogging 185
.dstoreTrace 185
ffs.log 185
ffsget.log 185
ffsput.log 185
lock.log 185
rmt_class_loader.cache.jar 185
rsecomm.log 185
stderr.log 185
stdout.log 185
Registro del daemon RSE 184
registro del gestor de depuración 184
Registro del supervisor de trabajos JES 184
reglas de clasificación, WLM 74
reglas de clasificación WLM 74
rendimiento, Consideraciones relativas 123
rendimiento de la comprobación de seguridad, Mejorar el 125
Repositorio CRD 43
Requisitos de JCL de inicio 198
reserva, puerto TCPIP 65
reserva de puerto, TCP/IP 65
Reserva de puerto TCP/IP 65
reservados, Puertos TCP/IP 196
Resolución de problemas de configuración 181
resolvente, Definiciones locales disponibles para 231
resolvente TCP/IP, dirección de host no resuelta lock.log 231
resolventes, Qué son 226
resolventes base, Archivos de configuración 227
retorno de errores, Rastreo de información 192
revisión de código, registro 187
rmt_class_loader.cache.jar 182
RSE, Definir archivos controlados por programa z/OS UNIX para 55
RSE, Definir bibliotecas controladas por programa MVS para 52
RSE, Definir como servidor z/OS UNIX seguro 51
RSE, Definir comprobación de puerto de entrada para 36
RSE, Definir el soporte de PassTicket para 53
RSE, definir permiso de acceso de archivos z/OS UNIX 54
RSE, definir protección de aplicaciones para 54
RSE, pushtoclient.properties 46
RSE, Registro de usuario 185
RSE, rsed.envvars _RSE_JAVAOPTS 45

- RSE, ssl.properties 46
- RSE, supervisar 114
- RSE como aplicación Java
 - representación gráfica 5
- RSE existente, Clonar la
 - configuración 205
- rsecomm.log 182
 - Registro de SCLM Developer Toolkit 186
- rsecomm.properties 191
- rsed.envvars 107, 145, 148, 177
 - _CMDSERV_CONF_HOME 174
 - _RSE_JAVAOPTS 171, 188
 - _RSE_PORTRANGE 22
 - Dmaximum.clients 108
 - Dmaximum.threadpool.process 108
 - Dmaximum.threads 108
 - Dminimum.threadpool.process 108
 - DSTORE_LOG_DIRECTORY 186, 190
 - STEPLIB 31
 - Xms 108
 - Xmx 108
- rsed.envvars, actualizar para habilitar la coexistencia 205
- rsedaemon.log 182, 183
- rseserver.log 182, 183
- rutina de salida de usuario, escritura 165

S

- salida de usuario, mensajes de
 - consola 166
- salidas del sistema, Limitaciones aplicadas por 198
- SCLM Developer Toolkit 52
- SCLM Developer Toolkit, registro 186
- SCLM Developer Toolkit (SCLMDT) 4
- script de shell de z/OS UNIX 166
- segmento OMVS, Definir 50
- seguridad, Activar valores y clases 49
- seguridad, CICSTS 42
- seguridad, conducto 153
- seguridad, Consideraciones relativas a 19
- Seguridad, definiciones 47
- seguridad, depuración 42
- seguridad, Gestor de despliegue de aplicaciones (ADM) 153
- seguridad, logfile 39
- seguridad, recurso 155
- seguridad, SCLM 43
- seguridad, transacción 153
- Seguridad de CICSTS 42
- seguridad de conducto 153
- Seguridad de conexión 21
- seguridad de depuración 42
- seguridad de hebras en el servidor RSE
 - PassTickets 23
- Seguridad de JES 26
- seguridad de mandatos JES, Definir 55
- seguridad de memoria caché, máquinas virtuales Java (JVM) 127
- seguridad de metadatos 133
- seguridad de recursos 155
- Seguridad de SCLM 43
- seguridad de transacciones 153
- Seguridad del Gestor de despliegue de aplicaciones 153
- seguridad del repositorio, CRD 153
- seguridad del repositorio CRD 153
- selección de grupo, basada en LDAP 140
- selección de grupo, basada en SAF 145
- selección de puerto, restringir 69
- selección de servidor, LDAP 142
- Selección del servidor LDAP 142
- serverlogs.count 182
- Servicio de mandatos TSO 4, 171
- servidor de definiciones de recurso CICS (CRD), Gestor de despliegue de aplicaciones 151
- servidor RSE 64
- servidor UNIX, Definir RSE como 51
- servidor z/OS seguro, Definir RSE como 51
- servidor z/OS UNIX, Definir RSE como 51
- sincronización automatizada 177
- sistema, Límites 199
- sistema, Mejorar el acceso a las bibliotecas del 123
- sistema primario 132
- sistemas de archivos z/OS UNIX, supervisar 117
- sistemas de archivos zFS, utilizar 123
- software de seguridad, autenticación por 34
- soporte de autorización de cliente, añadir X.509 210
- soporte de PassTicket para RSE, Definir 53
- SSL, cifrado 202
- SSL, Cifrado de comunicaciones mediante 22
- SSL, Configurar 201
- ssl.properties, Activar SSL actualizando 206
- ssl.properties, activar SSL creando un daemon RSE nuevo 206
- stderr.*.log 182
- stderr.log 182
- stdout.*.log 182
- stdout.log 182
- STEPLIB, Evitar el uso de 123
- supervisar, red 117
- supervisar RSE 114
- supervisar sistemas de archivos z/OS UNIX 117
- supervisión de z/OS UNIX 115
- Supervisor de trabajos JES, autenticación 21
- supervisor de trabajos JES, FEJCNFG 44
- supervisor de trabajos JES (JMON) 4
- Supervisor de trabajos JES, rastreo 190
- Supervisor de trabajos JES, Registro 184
- SYS1.PARMLIB(BPXPRMxx) 119
 - MAXASSIZE 51, 198
 - MAXPROCSYS 199
 - MAXPROCUSER 199
 - MAXUIDS 200
- SYS1.PARMLIB(BPXPRMxx), Limitaciones establecidas en 198

- SYS1.PARMLIB(BPXPRMxx), máquinas virtuales Java (JVM) 127
- SYS1.PARMLIB(IEASYSxx) 120
 - MAXUSER 200
- sysplex, configuración idéntica en todo 175

T

- Tablas de conversión 228
- Tablas de hosts locales 228
- Tamaño del espacio de direcciones 198
- tamaño fijo, Almacenamiento dinámico Java de 125
- tarea iniciada, agente de política 217
- tarea iniciada del agente de política 217
- tareas iniciadas, Definir para Developer for System z
 - tareas iniciadas JMON 50
 - tareas iniciadas RSED 50
- tareas iniciadas de Developer for System z, Definir 50
- TCP/IP, aplicar a Developer for System z 229
- TCP/IP, Configurar 225
- TCP/IP, Definiciones locales disponibles para el resolvente 231
- TCP/IP, Puertos 63
- TCP/IP, Puertos reservados 196
- tiempo de ejecución de Language Environment, bibliotecas de 124
- tipos de subsistema
 - ASCH 74
 - CICS 74
 - JES 74
 - OMVS 74
 - STC 74
- TLS, cifrado 202
- TLS, Cifrado de comunicaciones mediante 22
- trabajos, Acciones condicionales en 26
- transacciones CICS 43
- TSO, Métodos de acceso 171
- TSO/ISPF, personalización - ISPF.conf, 172
- TSO/ISPF, utilizar con varias configuraciones 173
- TSO/ISPF, Utilizar el método de acceso de Pasarela de cliente 172
- TSO/ISPF, Utilizar perfiles ISPF existentes 172
- TSO/ISPF, Utilizar un exec de asignación 173
- TSO/ISPF, utilizar varios ejecutables de asignación 173

U

- ubicación de metadatos 133
- ubicación de metadatos de grupo 138
- ubicación de servidor, LDAP 142
- Ubicación del servidor LDAP 142
- Ubicaciones de vuelcos de UNIX 190
- Ubicaciones de vuelcos de z/OS
 - UNIX 190
- UID 0 42

- UNIXPRIV, permisos de clase 41
- uso de almacenamiento 98
- uso de espacio, sistema de archivos z/OS
 - UNIX 105
- uso de espacio del sistema de archivos,
 - z/OS UNIX 105
- uso de espacio del sistema de archivos de
 - z/OS UNIX 105
- uso de recursos, temporal 94
- uso de STEPLIB, Evitar 123
- uso del espacio, metadatos 134
- Uso del espacio de metadatos 134
- Uso temporal de recursos 94
- usuario de RSE, Registro 185
- utilización de recursos, ajuste 81
- utilización de recursos, visión
 - general 82
- utilizar PassTickets 23
- utilizar perfiles ISPF existentes 172
- Utilizar un exec asignación 173

V

- validación de la autoridad certificadora
 - conjunto de claves SAF 33
 - gskkyman 33
 - TRUST, HIGHTRUST 33
- valores de seguridad, verificar 60
- valores y clases de seguridad,
 - Activar 49
- variables de patrón de volcado de
 - transacciones 189
- varias configuraciones de Developer for
 - System z, utilizar varios archivos
 - ISPF.conf con 173
- varias instancias, Ejecutar 175
- Varios archivos ISPF.conf 173
- varios ejecutables de asignación,
 - TSO/ISPF 173
- Varios grupos de desarrollador 135
- Verificar valores de seguridad 60
- VIPA, dinámico distribuido 68
- VIPA dinámico distribuido
 - EZBEPOR 68
 - PORT 68
 - PORTRANGE 68
 - SERVERWLM 68
 - SYSPLEXPORTS 68
 - VIPADISTRIBUTE 68
- visión general de componentes,
 - Developer for System z
 - representación gráfica 4
- Vuelcos de Java 188
- Vuelcos de MVS 188

X

- X.509, añadir soporte de autorización de
 - cliente 210
- X.509, certificado 20
- Xquickstart, opción Java 126

Z

- z/OS UNIX, supervisar 115
- z/OS UNIX, ubicaciones de vuelcos 190



SC11-7903-08

