

IBM Rational Developer for System z
Version 9.0.1

*Guide de référence de configuration de
l'hôte*



IBM Rational Developer for System z
Version 9.0.1

*Guide de référence de configuration de
l'hôte*



Important

Avant d'utiliser le présent document et le produit associé, prenez connaissance des informations générales figurant à la section «Mentions légales d'IBM Rational Developer for System z», à la page 225.

Remarque

Certaines illustrations de ce manuel ne sont pas disponibles en français à la date d'édition.

Septième édition - novembre 2013

Réf. US : SC14-7290-05

LE PRESENT DOCUMENT EST LIVRE EN L'ETAT SANS AUCUNE GARANTIE EXPLICITE OU IMPLICITE. IBM DECLINE NOTAMMENT TOUTE RESPONSABILITE RELATIVE A CES INFORMATIONS EN CAS DE CONTREFACON AINSI QU'EN CAS DE DEFAUT D'APTITUDE A L'EXECUTION D'UN TRAVAIL DONNE.

Ce document est mis à jour périodiquement. Chaque nouvelle édition inclut les mises à jour. Les informations qui y sont fournies sont susceptibles d'être modifiées avant que les produits décrits ne deviennent eux-mêmes disponibles. En outre, il peut contenir des informations ou des références concernant certains produits, logiciels ou services non annoncés dans ce pays. Cela ne signifie cependant pas qu'ils y seront annoncés.

Pour plus de détails, pour toute demande d'ordre technique, ou pour obtenir des exemplaires de documents IBM, référez-vous aux documents d'annonce disponibles dans votre pays, ou adressez-vous à votre partenaire commercial.

Vous pouvez également consulter les serveurs Internet suivants :

- <http://www.fr.ibm.com> (serveur IBM en France)
- <http://www.can.ibm.com> (serveur IBM au Canada)
- <http://www.ibm.com> (serveur IBM aux Etats-Unis)

*Compagnie IBM France
Direction Qualité
17, avenue de l'Europe
92275 Bois-Colombes Cedex*

Cette édition concerne IBM Rational Developer for System z version 9.0.1 (numéro de logiciel 5724-T07) et toutes les éditions et modifications ultérieures, sauf mention contraire dans les nouvelles éditions.

© Copyright IBM Corporation 2000, 2013.

Table des matières

Figures	vii
----------------	------------

Tableaux	ix
-----------------	-----------

Avis aux lecteurs canadiens.	xi
-------------------------------------	-----------

A propos de ce manuel	xiii
------------------------------	-------------

A qui s'adresse ce guide	xiii
--------------------------	------

Récapitulatif des changements	xiv
-------------------------------	-----

Description du contenu du document	xv
------------------------------------	----

Comprendre Developer for System z	xv
-----------------------------------	----

Remarques relatives à la sécurité	xvi
-----------------------------------	-----

Remarques relatives à TCP/IP	xvi
------------------------------	-----

Remarques relatives à WLM	xvi
---------------------------	-----

Remarques relatives à l'optimisation	xvi
--------------------------------------	-----

Remarques relatives aux performances	xvi
--------------------------------------	-----

Remarques relatives à la fonction d'envoi au client	xvi
---	-----

Remarques relatives à CICSTS	xvi
------------------------------	-----

Remarques relatives aux exits utilisateur	xvii
---	------

Personnalisation de l'environnement TSO	xvii
---	------

Exécution de plusieurs instances	xvii
----------------------------------	------

Traitement des incidents liés à la configuration	xvii
--	------

Configuration de l'authentification SSL et X.509	xvii
--	------

Configuration de TCP/IP	xvii
-------------------------	------

IBM Rational Developer for System z	
- Guide de référence de	
configuration de l'hôte	1

Chapitre 1. Description de Developer for System z	3
--	----------

Présentation du composant	3
---------------------------	---

RSE comme application Java	5
----------------------------	---

Propriétaires de tâches	7
-------------------------	---

Flux de connexion	8
-------------------	---

Débogueur intégré	10
-------------------	----

CARMA	11
-------	----

Fichiers de configuration CARMA	11
---------------------------------	----

CRASTART	12
----------	----

Soumission par lots	12
---------------------	----

Propriétaire du verrou d'un fichier	13
-------------------------------------	----

Libération d'un verrou	14
------------------------	----

Structure de répertoires z/OS UNIX	15
------------------------------------	----

Droits de mise à jour des administrateurs non système	17
---	----

Commandes de sécurité utiles	17
------------------------------	----

Commandes z/OS UNIX utiles	18
----------------------------	----

Exemple de configuration	18
--------------------------	----

Chapitre 2. Remarques relatives à la sécurité	19
--	-----------

Méthodes d'authentification	20
-----------------------------	----

Un ID utilisateur et un mot de passe	20
--------------------------------------	----

Un ID utilisateur et un mot de passe utilisable une seule fois	20
--	----

Certificat X.509	20
------------------	----

Authentification du moniteur de travaux JES	20
---	----

Authentification du gestionnaire de débogage	21
--	----

Sécurité des connexions	21
-------------------------	----

Limite des communications externes à des ports spécifiques	22
--	----

Chiffrement des communications à l'aide de SSL	22
--	----

ou TLS	22
--------	----

Vérification du port d'entrée	22
-------------------------------	----

Utilisation de PassTickets	23
----------------------------	----

Consignation dans le journal d'audit	24
--------------------------------------	----

Contrôle d'audit	24
------------------	----

Traitement de l'audit	25
-----------------------	----

Données d'audit	25
-----------------	----

Sécurité JES	26
--------------	----

Actions sur les travaux - Limitations sur les cibles	26
--	----

Actions sur les travaux - Limitations liées à l'exécution	28
---	----

Accès aux fichiers spoule	29
---------------------------	----

Communication chiffrée via SSL/TLS	30
------------------------------------	----

Communication chiffrée pour le débogueur intégré	31
--	----

Authentification du client à l'aide de certificats	31
--	----

X.509	31
-------	----

Validation de l'autorité de certification (CA)	32
--	----

(Facultatif) Interrogation d'une liste de révocation de certificat (CRL)	33
--	----

Authentification par votre logiciel de sécurité	33
---	----

Authentification effectuée par le démon RSE	34
---	----

Vérification du port d'entrée (POE)	35
-------------------------------------	----

Modification des fonctions client	35
-----------------------------------	----

OFF.REMOTECOPY.MVS	36
--------------------	----

Groupes de développeurs de la fonction d'envoi au client	37
--	----

Sécurité du débogage	38
----------------------	----

CICSTS, sécurité	38
------------------	----

Référentiel de CRD	39
--------------------	----

Transactions CICS	39
-------------------	----

Communication chiffrée via SSL	39
--------------------------------	----

Informations diverses	39
-----------------------	----

Mise en corbeille GATE	39
------------------------	----

Élément ACEE géré	39
-------------------	----

SCLM, sécurité	40
----------------	----

Fichiers de configuration Developer for System z	40
--	----

Moniteur de travaux JES - FEJJCNFG	40
------------------------------------	----

RSE - rsed.envvars	41
--------------------	----

RSE - ssl.properties	42
----------------------	----

RSE - pushtoclient.properties	42
-------------------------------	----

Définitions de sécurité	43
-------------------------	----

Configuration requise et liste de contrôle	43
--	----

Activation des classes et des paramètres de sécurité	45
--	----

Définition d'un segment OMVS pour les utilisateurs Developer for System z	46
Définition des tâches démarrées de Developer for System z	46
Définition de RSE en tant que serveur z/OS UNIX sécurisé	48
Définition des bibliothèques contrôlées de programme MVS pour RSE	48
Définition de la prise en charge de PassTicket pour RSE	49
Définition de la protection des applications pour RSE	50
Définition de la sécurité des commandes JES	51
Définition des profils de fichier	53
Définition de fichiers contrôlés par programme z/OS UNIX pour RSE	57
Vérification des paramètres de sécurité	58

Chapitre 3. Remarques relatives à

TCP/IP 59

Ports TCP/IP	59
Communications externes	60
Communications internes	60
Réservation de port TCP/IP	61
CARMA et ports TCP/IP	61
Remarques relatives à LDAP	62
Remplacement du comportement TCP/IP par défaut	62
Fonction de retardement d'accusé de réception	62
Piles multiples (CINET)	62
CARMA et affinité entre piles	63
crastart*.conf	63
CRASUB*	63
Distributed Dynamic VIPA	64
Restriction de la sélection de port	65
Exemple de configuration	67
Système SYS1 – Profil TCP/IP	68
Système SYS2 – Profil TCP/IP	68

Chapitre 4. Remarques relatives à WLM 71

Classification des charges de travail	71
Règles de classification	72
Définition des objectifs	73
Remarques relatives à la sélection des objectifs	74
STC	75
OMVS	75
JES	77
ASCH	78
CICS	78

Chapitre 5. Remarques relatives à

l'optimisation 81

Utilisation des ressources	81
Présentation	82
Nombre d'espaces adresse	83
Nombre de processus	86
Nombre d'unités d'exécution	89
Utilisation des ressources temporaires	92
Utilisation de l'espace de stockage	93
Limite de taille de pile Java	93
Limite de la taille d'espace adresse	93

Instructions relatives à l'évaluation de la taille	94
Exemple d'analyse de l'utilisation de l'espace de stockage	95
Utilisation de l'espace du système de fichiers z/OS UNIX	99
Définitions de ressources essentielles	102
/etc/rdz/rsed.envvars	102
SYS1.PARMLIB(BPXPRMxx)	103
Définitions de ressource différentes	106
Carte EXEC dans le JCL de serveur	106
FEK.#CUST.PARMLIB(FEJJCNFG)	106
SYS1.PARMLIB(IEASYSxx)	107
SYS1.PARMLIB(IVTPRMxx)	107
SYS1.PARMLIB(ASCHPMxx)	107
Contrôle	108
Contrôle de RSE	108
Contrôle de z/OS UNIX	109
Contrôle du réseau	111
Contrôle des systèmes de fichiers z/OS UNIX	111
Exemple de configuration	112
Nombre de pools d'unités d'exécution	112
Détermination des limites minimales	112
Définition des limites	113
Utilisation des ressources du moniteur	114

Chapitre 6. Remarques relatives aux performances 117

Utilisation du système de fichiers zFS	117
Eviter l'emploi de STEPLIB	117
Amélioration de l'accès aux bibliothèques du système	118
Bibliothèques d'exécution Language Environment (LE)	118
Développement d'applications	118
Amélioration des performances du contrôle d'autorisations d'accès	119
Gestion de la charge de travail	119
Taille de pile Java fixe	119
Option Java -Xquickstart	120
Partage de classes entre machines virtuelles Java	120
Activer le partage de classes	121
Limites de taille de la mémoire cache	121
Sécurité de la mémoire cache	121
SYS1.PARMLIB(BPXPRMxx)	122
Espace disque	122
Utilitaires de gestion de la mémoire cache	122

Chapitre 7. Remarques relatives à la fonction d'envoi au client 125

Introduction	125
Système principal	126
Métadonnées d'envoi au client	127
Emplacement des métadonnées	127
Métadonnées de sécurité	127
Utilisation de l'espace de métadonnées	128
Contrôle de la configuration client	128
Contrôle de la version client	129
Plusieurs groupes de développeurs	129
Activation	129
Concaténations de groupe	130

Liaison d'espace de travail	131
Emplacement des métadonnées de groupe.	131
Etapes de configuration	132
Sélection de groupe basé sur LDAP	133
Schéma LDAP	134
Sélection de serveur LDAP	134
Emplacement de serveur LDAP	135
Exemple de configuration	135
Ajout à LDAP d'une section dorsale pour la	
fonction d'envoi au client	136
Configuration de groupe LDAP initiale.	136
Ajout de développeurs à des groupes LDAP	
pushtoclient.properties	137
rsed.envvars.	138
/var/rdz/pushtoclient/*install	138
Sélection de groupe basé sur SAF	138
Exemple de configuration	139
Définition de sécurité.	140
pushtoclient.properties	140
rsed.envvars.	140
/var/rdz/pushtoclient/*install	140
Délai de grâce pour le rejet des modifications	
141	
Projets résidant sur l'hôte	141

Chapitre 8. Remarques relatives à CICSTS 143

RESTful par opposition à Web Service	144
Régions de connexion primaires versus régions de	
connexion non primaires	144
Consignation des messages d'installation des	
ressources CICS	145
Gestionnaire de déploiement d'application, sécurité	
CRD, sécurité du référentiel	145
Pipeline, sécurité	145
Sécurité de transaction	145
Communication chiffrée via SSL	147
Protection des ressources	147
Utilitaire d'administration	147
Notes de migration de l'utilitaire	
d'administration	151
Messages de l'utilitaire d'administration	152
Débogage de transaction CICS.	154

Chapitre 9. Remarques relatives aux exits utilisateur 157

Caractéristiques de l'exit utilisateur	157
Activation de l'exit utilisateur	157
Création d'une routine d'exit utilisateur	157
Messages de console	158
Exécution avec un ID utilisateur variable	158
Script de shell z/OS UNIX	158
Commande exec REXX z/OS UNIX	159
Points d'exit disponibles.	160
audit.action	160
logon.action	160

Chapitre 10. Personnalisation de l'environnement TSO 161

Service Commandes TSO	161
Méthodes d'accès	161

Utilisation de la méthode d'accès par passerelle	
client TSO/ISPF	162
ISPF.conf	162
Utilisation des profils ISPF existants.	162
Utilisation d'une commande exec d'allocation	
163	
Utilisation de plusieurs commandes exec	
d'allocation	163
Utilisation de fichiers ISPF.conf multiples avec	
configurations Developer for System z multiples	
164	

Chapitre 11. Exécution de plusieurs instances 165

Configuration identique par sysplex.	165
Niveaux de logiciels identiques, fichiers de	
configuration différents	166
Synchronisation automatisée	167
Dans tous les autres cas	168

Chapitre 12. Traitement des incidents liés à la configuration 171

Journal et analyse de configuration à l'aide de	
FEKLOGS	171
Fichiers journaux	172
Journalisation du moniteur de travaux JES	173
Journalisation du démon RSE et du pool	
d'unités d'exécution	173
Journalisation pour l'utilisateur RSE.	174
SCLM Developer Toolkit, journalisation	175
Journalisation CARMA	176
Consignation des tests du programme de	
vérification de l'installation fekfivpc	176
Consignation des tests du programme de	
vérification de l'installation (IVP) fekfivpi	176
Consignation des tests de la procédure de	
vérification d'installation fekfivps.	177
Journalisation de la révision du code	177
Journalisation de la couverture de code.	177
Fichiers de vidage.	177
Fichiers de vidage MVS	177
Fichiers de vidage Java	178
Emplacements des fichiers de vidage z/OS	
UNIX	179
Traçage	180
Fonction de trace du moniteur de travaux JES	
180	
RSE, traçage.	180
CARMA, traçage	181
Traçage de suivi des erreurs	181
Bits d'autorisation z/OS UNIX	182
SETUID, attribut du système de fichiers	182
Autorisation de contrôle de programmes	183
Autorisation APF	184
Données de rappel	185
Ports TCP/IP réservés	185
Taille d'espace adresse	187
Exigences liées au langage JCL lors du	
démarrage	187
Limitations définies dans	
SYS1.PARMLIB(BPXPRMxx)	187
Limitations stockées dans le profil de sécurité	
187	
Limitations forcées par les sorties du système	
188	

Limitations pour adressage 64 bits	188
Informations diverses.	188
Fin anormale pour manque d'espace B37 lors du retour d'informations.	188
Limites du système	188
Connexion refusée.	189
Erreur liée à une insuffisance de mémoire.	189
Emulateur de connexion à l'hôte	189

Chapitre 13. Configuration de l'authentification SSL et X.509 191

Utilisation de la méthode de chiffrement SSL ou TLS.	192
Choix de l'emplacement de stockage des clés privées et des certificats	192
Création d'un fichier de clés avec RACF	193
(Facultatif) Utilisation d'un certificat signé.	194
Clonage de la configuration RSE existante.	195
Mise à jour du fichier rsed.envvars pour assurer la coexistence	195
Mise à jour du fichier ssl.properties pour activer SSL.	196
Activation de SSL en créant un démon RSE	196
Test de la connexion	197
(Facultatif) Ajout du support d'authentification du client via des certificats X.509	200
(Facultatif) Création d'une base de données de clés avec gskkyman.	200
(Facultatif) Création d'un magasin de clés avec keytool	203

Chapitre 14. Configuration de TCP/IP 205

Dépendance au nom d'hôte.	205
Présentation des programmes de résolution	206
Présentation des ordres de recherche d'informations de configuration	206
Ordres de recherche utilisés dans l'environnement z/OS UNIX	207
Fichiers de configuration du programme de résolution de base	207
Tables de conversion	208
Tables de système hôte local	208
Application de ces informations de configuration à Developer for System z	209
Résolution erronée de l'adresse hôte	211

Bibliographie 213

Publications référencées	213
Publications d'information	216

Glossaire 219

Mentions légales d'IBM Rational Developer for System z 225

Licence de copyright	228
Marques	228

Index 229

Figures

1.	Présentation du composant.	3	20.	Nombre maximal d'unités d'exécution du	
2.	RSE comme application Java	5		moniteur de travaux JES	91
3.	Propriétaires de tâches	7	21.	Utilisation des ressources avec 5 connexions	96
4.	Flux de connexion.	8	22.	Utilisation des ressources avec 5 connexions	
5.	Débogueur intégré	10		(suite)	97
6.	Flux CARMA	11	23.	Utilisation des ressources lors de l'édition d'un	
7.	Flux de détermination de mise en file d'attente			membre PDS	98
	d'un fichier	13	24.	Utilisation de l'espace du système de fichiers	
8.	Structure de répertoires z/OS UNIX	15		z/OS UNIX	100
9.	Règle AT-TLS pour le gestionnaire de		25.	Utilisation des ressources de la configuration	
	débogage	31		modèle.	115
10.	Ports TCP/IP	59	26.	Exemple de définition de schéma LDAP	134
11.	update.sh - prise en charge de la configuration		27.	Utilitaire d'administration ADNJSPAU -	
	DDVIPA avec un pare-feu.	66		CICSTS	149
12.	Exemple d'adresse distribuée DVIPA	68	28.	ADNJSPAU - Utilitaire d'administration	
13.	Classification WLM	71		CICSTS (Partie 2 de 3)	150
14.	Nombre maximal d'espaces adresse	84	29.	ADNJSPAU - Utilitaire d'administration	
15.	Nombre d'espaces adresse par client	85		CICSTS (Partie 3 de 3)	151
16.	Nombre maximal de processus	87	30.	RSEDSSL - Travail de l'utilisateur du serveur	
17.	Nombre de processus pour STCRSE	88		RSE pour SSL	197
18.	Nombre de processus par client.	88	31.	Boîte de dialogue Importation du certificat	
19.	Nombre maximal d'unités d'exécution du pool			hôte.	198
	d'unités d'exécution RSE	91	32.	Boîte de dialogue Préférences - SSL	199

Tableaux

1.	Commandes de la console du moniteur de travaux JES	26	23.	Utilisation des ressources spécifiques de l'utilisateur.	82
2.	Matrice des droits d'accès des commandes LIMIT_COMMANDS	27	24.	Nombre d'espaces adresse.	83
3.	Profils JESSPOOL étendus.	27	25.	Limites d'espace adresse	86
4.	Matrice des droits de la console LIMIT_CONSOLE	28	26.	Nombre de processus	86
5.	Matrice des droits d'accès de consultation LIMIT_VIEW	29	27.	Limites de processus	88
6.	Mécanismes de stockage des certificats SSL	30	28.	Nombre d'unités d'exécution.	89
7.	Informations SAF en vue de la modification des fonctions client	36	29.	Limites d'unités d'exécution	92
8.	Informations SAF pour la fonction d'envoi au client.	37	30.	Paramètres de référence pour l'utilisation de mémoire	95
9.	Informations SAF pour les fonctions de débogage	38	31.	Répertoires de sortie de journal	101
10.	Variables de configuration de la sécurité	43	32.	Directives de sortie temporaire.	102
11.	Commandes d'opérateur du moniteur de travaux JES2	51	33.	Support de groupe de la fonction d'envoi au client pour *.enabled	129
12.	Commandes d'opérateur du moniteur de travaux JES3	52	34.	Support de groupe de la fonction d'envoi au client pour reject.*.updates	130
13.	Sous-systèmes de point d'entrée WLM	72	35.	Concaténations de groupe pour la fonction d'envoi au client	130
14.	Qualificateurs de travaux WLM.	73	36.	Informations LDAP pour la fonction d'envoi au client	133
15.	Charges de travail WLM	74	37.	Informations SAF pour la fonction d'envoi au client	138
16.	Charges de travail et STC WLM.	75	38.	Variables JAVA_DUMP_TDUMP_PATTERN	178
17.	Charges de travail - OMVS WLM	76	39.	Mécanismes de stockage des certificats SSL	192
18.	Charge de travail - JES WLM	77	40.	Définitions locales disponibles pour le programme de résolution	211
19.	Charges de travail - ASCH WLM	78	41.	Publications référencées	213
20.	Charges de travail WLM - CICS.	78	42.	Sites Web référencés	216
21.	Utilisation des ressources communes	82	43.	Publications d'information	216
22.	Utilisation des ressources prérequis spécifiques de l'utilisateur.	82			

Avis aux lecteurs canadiens

Le présent document a été traduit en France. Voici les principales différences et particularités dont vous devez tenir compte.

Illustrations

Les illustrations sont fournies à titre d'exemple. Certaines peuvent contenir des données propres à la France.

Terminologie

La terminologie des titres IBM peut différer d'un pays à l'autre. Reportez-vous au tableau ci-dessous, au besoin.

IBM France	IBM Canada
ingénieur commercial	représentant
agence commerciale	succursale
ingénieur technico-commercial	informaticien
inspecteur	technicien du matériel

Claviers

Les lettres sont disposées différemment : le clavier français est de type AZERTY, et le clavier français-canadien de type QWERTY.








OS/2 et Windows - Paramètres canadiens

Au Canada, on utilise :

- les pages de codes 850 (multilingue) et 863 (français-canadien),
- le code pays 002,
- le code clavier CF.

Nomenclature

Les touches présentées dans le tableau d'équivalence suivant sont libellées différemment selon qu'il s'agit du clavier de la France, du clavier du Canada ou du clavier des États-Unis. Reportez-vous à ce tableau pour faire correspondre les touches françaises figurant dans le présent document aux touches de votre clavier.

France	Canada	Etats-Unis
 (Pos1)		Home
Fin	Fin	End
 (PgAr)		PgUp
 (PgAv)		PgDn
Inser	Inser	Ins
Suppr	Suppr	Del
Echap	Echap	Esc
Attn	Intrp	Break
Impr écran	ImpEc	PrtSc
Verr num	Num	Num Lock
Arrêt défil	Défil	Scroll Lock
 (Verr maj)	FixMaj	Caps Lock
AltGr	AltCar	Alt (à droite)

Brevets

Il est possible qu'IBM détienne des brevets ou qu'elle ait déposé des demandes de brevets portant sur certains sujets abordés dans ce document. Le fait qu'IBM vous fournisse le présent document ne signifie pas qu'elle vous accorde un permis d'utilisation de ces brevets. Vous pouvez envoyer, par écrit, vos demandes de renseignements relatives aux permis d'utilisation au directeur général des relations commerciales d'IBM, 3600 Steeles Avenue East, Markham, Ontario, L3R 9Z7.

Assistance téléphonique

Si vous avez besoin d'assistance ou si vous voulez commander du matériel, des logiciels et des publications IBM, contactez IBM direct au 1 800 465-1234.

A propos de ce manuel

Ce document fournit des informations de base sur les différentes tâches de configuration de IBM® Rational Developer for System z, ainsi que d'autres composants et produits z/OS (tels que WLM et CICS).

A partir de maintenant, les noms suivants sont utilisés dans le présent ouvrage :

- *IBM Rational Developer for System z* a pour nom *Developer for System z*.
- *IBM Rational Developer for System z Integrated Debugger* est appelé débogueur intégré.
- L'abréviation utilisée pour *Common Access Repository Manager* est CARMA.
- *Software Configuration and Library Manager Developer Toolkit* est appelé SCLM Developer Toolkit et parfois abrégé en SCLMDT.
- *z/OS UNIX System Services* est appelé *z/OS UNIX*.
- *Customer Information Control System Transaction Server* est appelé *CICSTS*, abrégé en *CICS*.

Ce document fait partie d'un groupe de documents qui décrivent la configuration de l'hôte Developer for System z. Chacun de ces documents s'adresse à des utilisateurs spécifiques. Il est inutile de lire tous les documents pour configurer Developer for System z.

- *Rational Developer for System z Guide de configuration de l'hôte* (SC23-7658) décrit en détails toutes les tâches de planification, les tâches de configuration et les options (y compris les options facultatives) et offre des scénarios de remplacement.
- *Rational Developer for System z Guide de référence de configuration de l'hôte* (SC14-7290) décrit la conception de Developer for System z et offre des informations de base sur les différentes tâches de configuration des composants Developer for System z, z/OS et d'autres produits (tels que WLM et CICS) liés à Developer for System z.
- *Rational Developer for System z Configuration de l'hôte Guide de démarrage rapide* (GI11-9201) décrit la configuration minimale de Developer for System z.
- *Rational Developer for System z Guide de l'utilitaire de configuration de l'hôte* (SC14-7282) décrit l'utilitaire de configuration de l'hôte, application à panneaux ISPF qui vous aide à exécuter les tâches de personnalisation de base et facultatives courantes pour Developer for System z.

Les informations contenues dans ce document s'appliquent à tous les modules IBM Rational Developer for System z version 9.0.

A qui s'adresse ce guide

Ce document s'adresse aux programmeurs système chargés de configurer et d'optimiser IBM Rational Developer for System z version 9.0.1.

Bien que les étapes de configuration soient décrites dans une autre publication, ce document décrit en détail les sujets associés, tels que l'optimisation, la configuration de la sécurité etc. Avant d'utiliser le présent manuel, vous devez maîtriser les systèmes hôtes z/OS UNIX System Services et MVS.

Récapitulatif des changements

Cette section récapitule les modifications apportées au document *IBM Rational Developer for System z version 9.0 - Guide de référence de configuration de l'hôte*, SC14-7290-05 (mis à jour en décembre 2013).

Les changements et ajouts techniques au texte et illustrations sont indiqués par un trait vertical situé à gauche du changement.

Nouvelles informations :

- Ajout d'informations sur les noms de fichier journal avec horodatage. Voir «Fichiers journaux», à la page 172.
- Ajout d'informations sur de nouveaux événements auditables. Voir Données d'audit.

Ce document reprend certaines informations présentées précédemment dans le document *IBM Rational Developer for System z version 9.0 - Guide de référence de configuration de l'hôte*, SC14-7290-04.

Nouvelles informations :

- Mise à jour de l'utilisation du port TCP/IP. Voir «Ports TCP/IP», à la page 59.
- Ajout d'un exemple pour la synchronisation automatique de 2 démons RSE. Voir «Synchronisation automatisée», à la page 167.
- Ajout d'informations sur de nouveaux fichiers journaux. Voir «Fichiers journaux», à la page 172.

Ce document reprend des informations présentées précédemment dans le document *IBM Rational Developer for System z Version 8.5.1 - Guide de référence de configuration de l'hôte*, SC14-7290-03.

Nouvelles informations :

- Ajout d'informations sur les profils SAF permettant de modifier les fonctions client. Voir «Modification des fonctions client», à la page 35.
- Mise à jour des numéros d'utilisation des ressources. Voir Chapitre 5, «Remarques relatives à l'optimisation», à la page 81
- Mise à jour de la valeur par défaut du nombre maximal d'utilisateurs par pool d'unités d'exécution. Voir Chapitre 5, «Remarques relatives à l'optimisation», à la page 81.

Ce document reprend des informations présentées précédemment dans le document *IBM Rational Developer for System z Version 8.5 - Guide de référence de configuration de l'hôte*, SC11-6869-02.

Nouvelles informations :

- Mise à jour des informations relatives à la sécurité du gestionnaire de travaux JES. Voir Chapitre 2, «Remarques relatives à la sécurité», à la page 19.
- Informations supplémentaires sur les exits utilisateur. Voir Chapitre 9, «Remarques relatives aux exits utilisateur», à la page 157.

Ce document reprend des informations présentées précédemment dans le document *IBM Rational Developer for System z Version 8.0.3 - Guide de référence de configuration de l'hôte*, SC14-7290-01.

Nouvelles informations :

- Mise à jour de la structure de répertoire z/OS UNIX. Voir «Structure de répertoires z/OS UNIX», à la page 15.
- Ajout d'informations concernant le contrôle client basé sur un hôte. Voir Chapitre 7, «Remarques relatives à la fonction d'envoi au client», à la page 125.
- Ajout d'informations d'envoi au client liées à la sécurité. Voir «Groupes de développeurs de la fonction d'envoi au client», à la page 37.
- Utilisation de la documentation sur les éléments ACEE gérés. Voir «Elément ACEE géré», à la page 39.
- Ajout d'informations sur le traitement de journal d'audit automatisé. Voir «Traitement de l'audit», à la page 25.
- Ajout d'informations sur les directives liées aux audits et à la sécurité dans les fichiers de configuration. Voir «Fichiers de configuration Developer for System z», à la page 40.
- Ajout d'informations sur TCP/IP. Voir Chapitre 3, «Remarques relatives à TCP/IP», à la page 59.
- Mise à jour des informations d'autorité de certification pour la communication SSL. Voir Chapitre 13, «Configuration de l'authentification SSL et X.509», à la page 191.
- Mise à jour de l'utilisation des ressources. Voir «Utilisation des ressources», à la page 81.

Ce document reprend des informations présentées précédemment dans le document *IBM Rational Developer for System z Version 8.0.1 - Guide de référence de configuration de l'hôte*, SC14-7290-00.

Nouvelles informations :

- Section CARMA sous Compréhension de Developer for System z. Voir «CARMA», à la page 11.
- Informations d'ordre général sur le protocole TCP/IP. Voir Chapitre 3, «Remarques relatives à TCP/IP», à la page 59.
- Résolution de la fin anormale pour manque d'espace B37. Voir «Fin anormale pour manque d'espace B37 lors du retour d'informations», à la page 188.

Informations supprimées :

- Les informations figurant précédemment dans la documentation *IBM Rational Developer for System z Version 7.6.1 Guide de configuration de l'hôte* (SC11-6285-05) sont désormais réparties entre deux documents : *IBM Rational Developer for System z Guide de configuration de l'hôte* (SC11-6285) et *IBM Rational Developer for System z Guide de référence de configuration de l'hôte* (SC11-6869).
- Les informations relatives à la configuration APPC ont été transférées dans le livre blanc *Using APPC to provide TSO command services* (SC14-7291).
- Configuration de INETD

Description du contenu du document

Cette section récapitule les informations présentées dans ce document.

Comprendre Developer for System z

Le système hôte Developer for System z est composé de plusieurs composants qui interagissent pour permettre au client d'accéder aux services et données de l'hôte. En comprenant bien la conception de ces composants, vous pouvez prendre les bonnes décisions de configuration.

Remarques relatives à la sécurité

Developer for System z offre aux utilisateurs un accès grand système sur un poste de travail qui ne correspond pas à un grand système. La validation des demandes de connexion, l'établissement de communications sécurisées entre l'hôte et le poste de travail, l'autorisation et l'activité d'audit sont donc des aspects fondamentaux de la configuration d'un produit.

Remarques relatives à TCP/IP

Developer for System z repose sur le protocole TCP/IP pour offrir l'accès au mainframe à des utilisateurs travaillant sur un poste de travail autre qu'un mainframe. TCP/IP sert également à assurer la communication entre les différents composants et les autres produits.

Remarques relatives à WLM

Contrairement aux applications z/OS traditionnelles, Developer for System z n'est pas une application monolithique qui peut être identifiée facilement au niveau du Workload Manager (WLM). Les différents composants de Developer for System z interagissent pour offrir au client un accès à des services et des données d'hôte. Certains de ces services sont actifs dans différents espaces adresse, ce qui se traduit par différentes classifications WLM.

Remarques relatives à l'optimisation

RSE (Remote Systems Explorer) est le cœur de Developer for System z. Pour gérer les connexions et charges de travail provenant des clients, RSE est composé d'un espace adresse de démon, qui permet de contrôler les espaces adresse du groupe d'unités d'exécution. Le démon agit comme un point focal pour la connexion et la gestion, alors que les pools d'unités d'exécution traitent les charges de travail du client.

RSE devient une cible privilégiée d'optimisation de la configuration de Developer for System z. Toutefois, la gestion de centaines d'utilisateurs, chacun utilisant au moins 17 unités d'exécution, d'une certaine quantité de mémoire et éventuellement d'un ou de plusieurs espaces adresse implique de configurer correctement Developer for System z et z/OS.

Remarques relatives aux performances

z/OS est un système d'exploitation hautement personnalisable, et des modifications (parfois mineures) du système peuvent présenter un impact très important sur les performances globales. Le présent chapitre met en évidence certaines modifications qui peuvent être apportées afin d'améliorer les performances de Developer for System z.

Remarques relatives à la fonction d'envoi au client

La fonction d'envoi au client, ou contrôle client résidant sur l'hôte, prend en charge la gestion centralisée des éléments suivants :

- Fichiers de configuration client
- Version de produit client
- Définitions de projet

Remarques relatives à CICSTS

Ce chapitre contient des informations utiles pour un administrateur CICS Transaction Server.

Remarques relatives aux exits utilisateur

Ce chapitre vous guide lors du processus d'amélioration de Developer for System z en créant des routines d'exit.

Personnalisation de l'environnement TSO

Ce chapitre vous aide à simuler une procédure d'ouverture de session TSO en ajoutant des instructions de définition de données et des fichiers à l'environnement TSO dans Developer for System z.

Exécution de plusieurs instances

Parfois, vous pouvez avoir besoin de plusieurs instances de Developer for System z actives sur un même système, lors du test d'une mise à niveau, par exemple. Cependant, certaines ressources (les ports TCP/IP, par exemple) ne peuvent pas être partagées. Les paramètres par défaut ne sont donc pas toujours applicables. Consultez les informations de ce chapitre afin de programmer la coexistence des différentes instances de Developer for System z, pour pouvoir ensuite les personnaliser à l'aide de ce guide de configuration.

Traitement des incidents liés à la configuration

Ce chapitre vous aide à résoudre certains problèmes fréquents qui peuvent se produire au cours de la configuration de Developer for System z. Il comporte les sections suivantes :

- Journal et analyse de configuration à l'aide de FEKLOGS
- Fichiers journaux
- Fichiers de vidage
- Traçage
- Bits d'autorisation z/OS UNIX
- Ports TCP/IP réservés
- Taille d'espace adresse
- Transaction APPC et service Commandes TSO
- Informations diverses

Configuration de l'authentification SSL et X.509

Cette annexe vous aide à résoudre certains incidents fréquents qui peuvent survenir lors de la configuration du protocole SSL (Secure Socket Layer), ou au cours de la vérification ou de la modification d'une configuration existante. Elle contient un exemple de configuration pour prendre en charge les utilisateurs qui s'authentifient à l'aide d'un certificat X.509.

Configuration de TCP/IP

Cette annexe vous aide à résoudre certains problèmes fréquents qui peuvent survenir lors de la configuration du protocole TCP/IP, ou lors de la vérification ou de la modification d'une configuration existante.

IBM Rational Developer for System z - Guide de référence de configuration de l'hôte

Chapitre 1. Description de Developer for System z

Les différents composants de l'hôte Developer for System z interagissent pour offrir au client un accès à des services et des données d'hôte. En comprenant bien la conception de ces composants, vous pouvez prendre les bonnes décisions de configuration.

Les rubriques suivantes sont traitées dans le présent chapitre :

- «Présentation du composant»
- «RSE comme application Java», à la page 5
- «Propriétaires de tâches», à la page 7
- «Flux de connexion», à la page 8
- «Débogueur intégré», à la page 10
- «CARMA», à la page 11
- «Propriétaire du verrou d'un fichier», à la page 13
- «Structure de répertoires z/OS UNIX», à la page 15

Présentation du composant

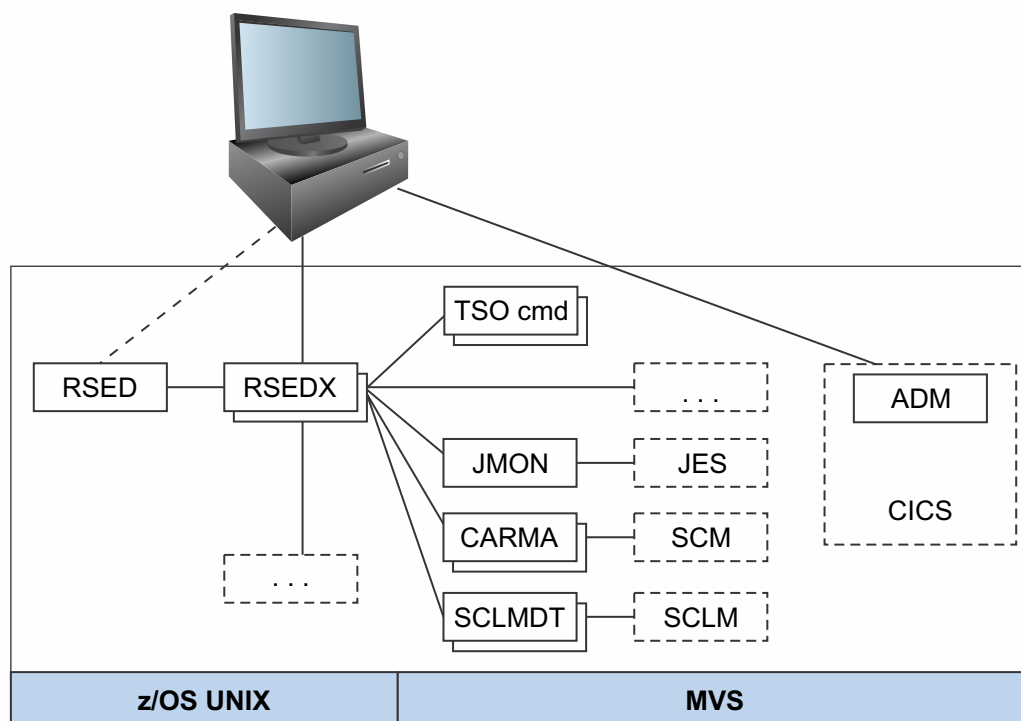


Figure 1. Présentation du composant

La figure 1 illustre une présentation générale de Developer for System z sur votre système hôte.

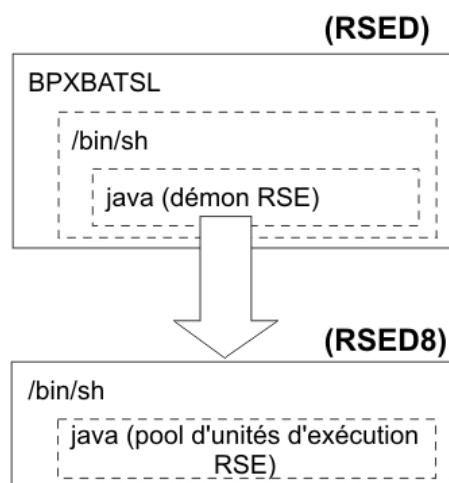
- L'Explorateur de systèmes distants (RSE) fournit des services de base, comme la connexion du client à l'hôte et le démarrage d'autres serveurs pour des services spécifiques. RSE se compose de deux entités logiques :
 - Le démon RSE (RSED), qui gère la configuration de la connexion. Il est également en charge de l'exécution en mode serveur unique. Pour se faire, le démon RSE crée un ou plusieurs processus enfants appelés pools d'unités d'exécution RSE (RSEDx).
 - Le serveur RSE qui gère les demandes client individuelles. Un serveur RSE est actif comme une unité d'exécution à l'intérieur d'un pool d'unités d'exécution RSE.
- Le service de Commandes TSO (TSO cmd) offre une interface de type par lots pour les commandes TSO et ISPF.
- Le moniteur de travaux JES (JMON) fournit tous les services relatifs à JES.
- Common Access Repository Manager (CARMA) offre une interface permettant d'interagir avec les gestionnaires d'accès au référentiel (SCM), comme CA Endevor.
- SCLM Developer Toolkit (SCLMDT) offre une interface permettant d'améliorer et d'interagir avec SCLM.
- Le gestionnaire de déploiement d'application (ADM) propose différents services liés à CICS.
- Plusieurs services sont disponibles. Ils peuvent être fournis par Developer for System z ou par d'autres logiciels corequis.

La description du paragraphe et de la liste précédent illustre le rôle central attribué à RSE. A quelques exceptions près, toute la communication du client passe par RSE. Cela permet de faciliter la configuration du réseau liée à la sécurité, étant donné que seul un ensemble limité de ports est utilisé pour la communication hôte-client.

Pour gérer les connexions et charges de travail provenant des clients, RSE est composé d'un espace adresse de démon, qui permet de contrôler les espaces adresse du groupe d'unités d'exécution. Le démon agit comme un point focal pour la connexion et la gestion, alors que les pools d'unités d'exécution traitent les charges de travail du client. Selon les valeurs définies dans le fichier de configuration `rsed.envvars` et la quantité de connexions client réelles, le démon peut démarrer plusieurs espaces adresse de pool d'unités d'exécution.

RSE comme application Java

Processus z/OS UNIX



Utilisation de mémoire Java

Système - partagé
Système - privé
Code (z/OS UNIX, Java, RSE)
Segment de mémoire Java
Non utilisé

JOBNAME	Etat	PID	PPID	Commande
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...

Figure 2. RSE comme application Java

La figure 2 présente une vue de base de l'utilisation des ressources (processus et stockage) par RSE.

RSE est une application Java[™], ce qui signifie qu'il est actif dans l'environnement z/OS UNIX. Cela permet de faciliter l'accès à différentes plateformes hôte et de simplifier la communication avec le client Developer for System z, qui est également une application Java (reposant sur Eclipse). Par conséquent, il est très utile d'avoir des connaissances de base du fonctionnement de z/OS UNIX et Java pour comprendre Developer for System z.

Dans z/OS UNIX, un programme s'exécute dans un processus, qui est identifié par un PID (ID processus). Chaque programme est actif dans son propre processus. Par conséquent, l'appel d'un autre programme crée un processus. Le processus qui en a démarré un autre est référencé par un PPID (PID parent), le nouveau processus étant appelé processus enfant. Ce dernier peut s'exécuter dans le même espace adresse ou être généré (créé) dans un nouvel espace adresse. L'exécution d'un nouveau processus dans le même espace adresse est comparable à l'exécution d'une commande dans TSO, la génération d'un processus dans un nouvel espace adresse s'apparentant à la soumission d'un nouveau travail.

Notez qu'un processus peut être à unité d'exécution unique ou à unités d'exécution multiples. Dans une application à unités d'exécution multiples (RSE, par exemple), les différentes unités d'exécution rivalisent pour des ressources système comme si elles se trouvaient dans des espaces adresse séparés (avec moins de temps système).

La mise en correspondance de ces informations de processus avec l'exemple RSE de la figure 2, à la page 5 permet d'obtenir le flux suivant :

1. Lorsque la tâche RSED est démarrée, elle exécute BPXBATSL, qui appelle z/OS UNIX et crée un environnement shell – PID 50331904.
2. Dans ce processus, le script de shell `rsed.sh` est exécuté, ce qui permet de lancer un processus distinct (`/bin/sh`) – PID 67109114.
3. Le script de shell définit les variables d'environnement définies dans `rsed.envvars` et exécute Java avec les paramètres requis afin de démarrer le démon RSE – PID 50331949.
4. Le démon RSE génère un shell dans un processus enfant (RSED8) – PID 307.
5. Dans ce shell, une valeur est attribuée aux variables d'environnement définies dans `rsed.envvars` et Java est exécuté avec les paramètres obligatoires afin de démarrer le pool d'unités d'exécution RSE – PID 308.

RSE peut s'exécuter en mode d'adressage 31 ou 64 bits, ce qui provoque différentes limites de stockage. En mode 31 bits, le stockage disponible est limité à 2 Go, alors qu'en mode 64 bits il n'existe aucune limite, sauf spécification particulière dans `SYS1.PARMLIB`.

Les applications Java (RSE, par exemple) n'allouent pas de mémoire directement. Elles utilisent les services de gestion de mémoire Java. Ces services (l'allocation de mémoire, la libération de mémoire et la récupération de place, par exemple) fonctionnent dans les limites du segment de mémoire Java. Les tailles minimale et maximale du segment de mémoire sont définies (de manière implicite ou explicite) au démarrage de Java. En mode 64 bits, Java tente d'allouer un segment de mémoire au dessus de 2 Go, libérant ainsi l'espace en-deçà de ce seuil.

Ainsi, l'occupation de la plus grande partie de l'espace adresse disponible consiste à trouver le juste équilibre entre une taille de segment de mémoire importante et une place suffisante permettant à z/OS de stocker un nombre variable de blocs de contrôle du système (selon le nombre d'unités d'exécution actives).

Propriétaires de tâches

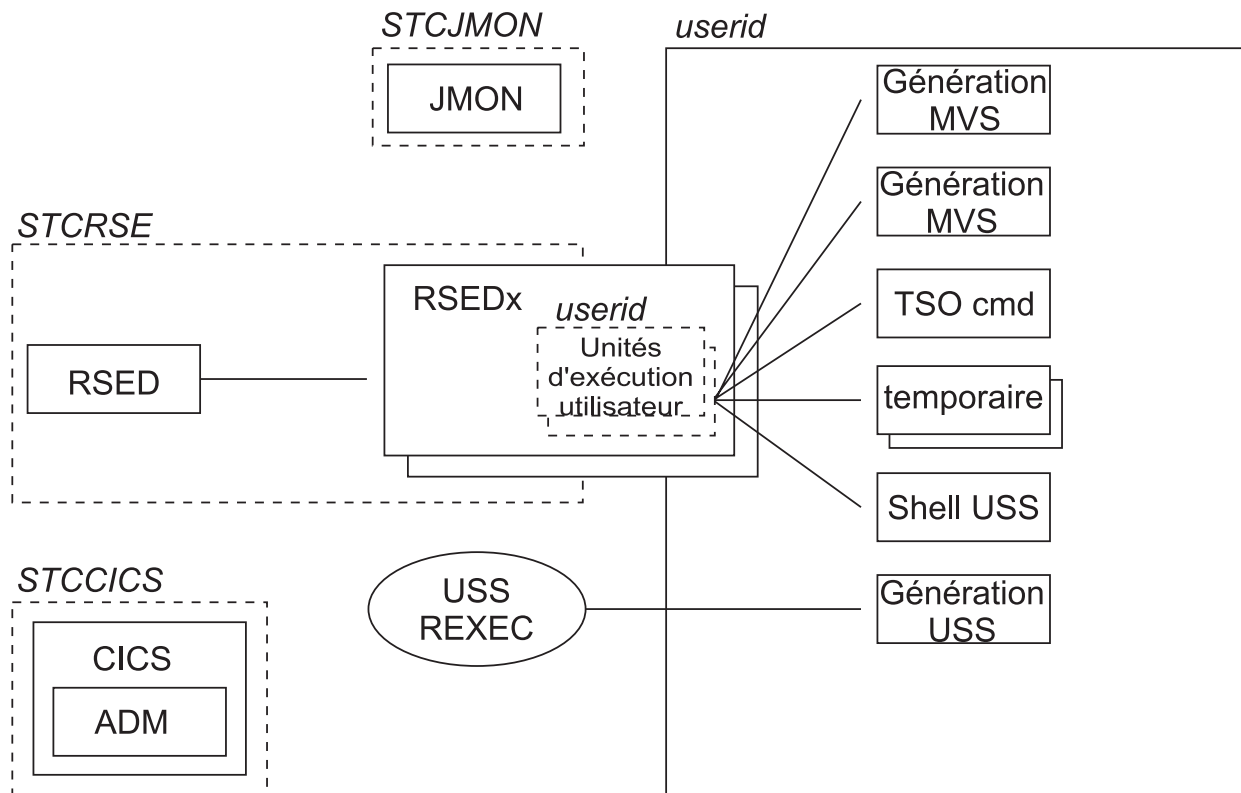


Figure 3. Propriétaires de tâches

La figure 3 offre une présentation de base du propriétaire des données d'identification utilisées par différentes tâches de Developer for System z.

La propriété d'une tâche peut être divisée en deux sections. Les tâches démarrées appartiennent à l'ID utilisateur qui est attribué à la tâche démarrée dans le logiciel de sécurité. Toutes les autres tâches, avec les pools d'unités d'exécution RSE (RSEDx) comme exception, appartiennent à l'ID utilisateur du client.

La figure 3 présente les tâches démarrées de Developer for System z (JMON et RSED) ainsi que des exemples de tâches démarrées et des services système avec lesquels communique Developer for System z. Application Deployment Manager (ADM) est actif au sein d'une région CICS. La balise USS REXEC représente le service z/OS UNIX REXEC (ou SSH).

Le démon RSE (RSED) crée un ou plusieurs espaces adresse de pools d'unités d'exécution (RSEDx) dédiés aux demandes des clients. Chaque pool d'unités d'exécution RSE prend en charge plusieurs clients et appartient au même utilisateur que celui du démon RSE. Chaque client possède sa propre unité d'exécution au sein d'un pool d'unités d'exécution et ces unités d'exécution appartiennent à l'ID utilisateur client.

Selon les actions menées par le client, un ou plusieurs espaces adresse supplémentaires peuvent être démarrés, tous appartenant à l'ID utilisateur du client, pour exécuter l'action demandée. Ces espaces adresse peuvent être un

travail par lots MVS, une transaction APPC ou un processus enfant z/OS UNIX. Notez qu'un processus enfant z/OS UNIX est actif dans un initiateur z/OS UNIX (BPXAS) et le processus enfant apparaît comme une tâche démarrée dans JES.

La création de ces espaces adresse est le plus souvent déclenchée par une unité d'exécution d'utilisateur dans un pool d'unités d'exécution, soit directement soit à l'aide d'un service système comme ISPF. L'espace adresse pourrait très bien être aussi créé par un tiers. Par exemple, File Manager démarrera un nouvel espace adresse pour chaque fichier (ou membre) qu'il a à traiter pour le compte de Developer for System z. REXEX ou SSH z/OS UNIX est impliqué lorsque un démarrage est généré dans z/OS UNIX.

Les espaces adresse spécifiques de l'utilisateur prennent fin à l'achèvement de la tâche ou à l'expiration d'un temps d'inactivité. Les tâches démarrées restent actives. Les espaces adresse répertoriés dans la figure 3, à la page 7 restent dans le système suffisamment longtemps pour être visibles. Toutefois, sachez qu'en raison de la conception de z/OS UNIX, il existe aussi des espaces adresse temporaires de durée de vie courte.

Flux de connexion

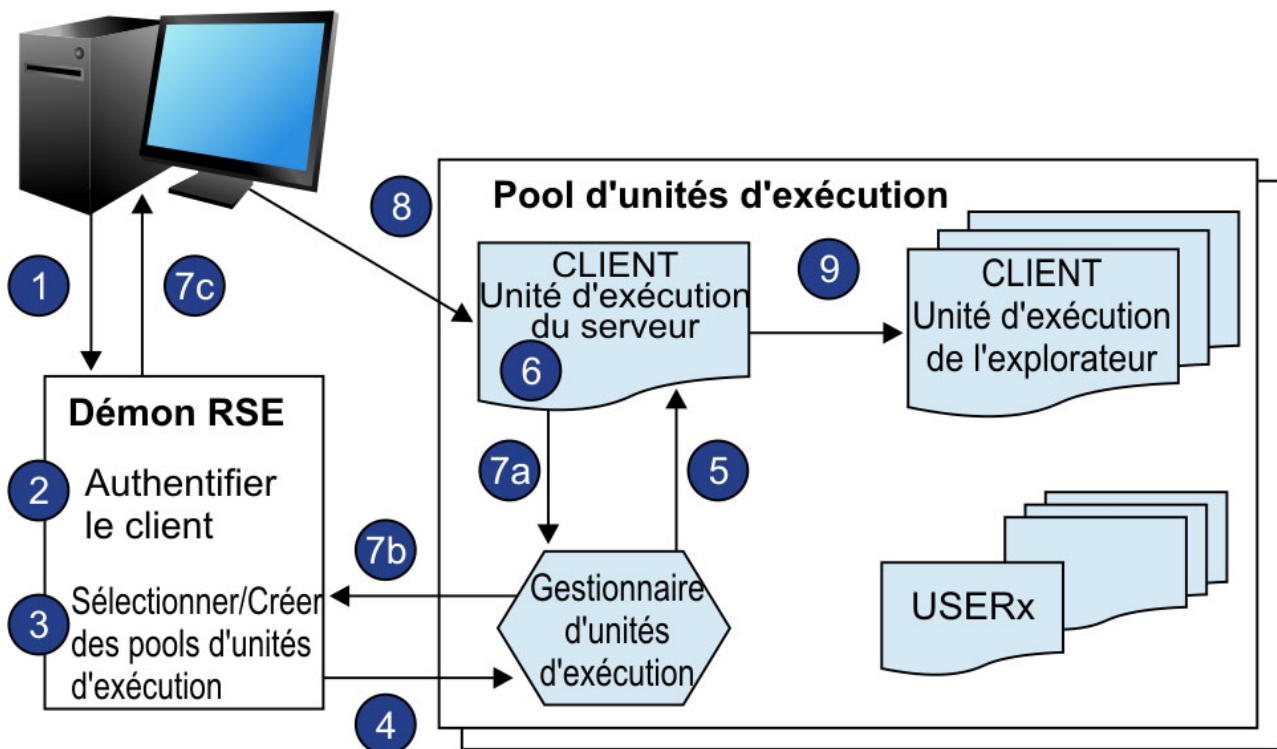


Figure 4. Flux de connexion

La figure 4 explique schématiquement comment un client se connecte à l'hôte via Developer for System z. De même, elle explique brièvement l'utilisation des PassTickets.

1. Le client se connecte au démon (port 4035).
2. Le démon RSE authentifie le client en utilisant les données d'identification présentées par le client.

3. Le démon RSE sélectionne un pool d'unités d'exécution existant ou en démarre un s'ils sont tous saturés.
4. Le démon RSE transmet l'ID utilisateur du client au pool d'unités d'exécution.
5. Le pool d'unités d'exécution crée une unité d'exécution de serveur RSE propre au client, en utilisant l'ID utilisateur du client et un mot de passe PassTicket pour l'authentification.
6. L'unité d'exécution de serveur du client associe un port pour les futures communications du client.
7. L'unité d'exécution de serveur renvoie le numéro de port pour permettre au client de s'y connecter.
8. Le client se déconnecte du démon RSE et se connecte au numéro de port fourni.
9. L'unité d'exécution de serveur du client démarre d'autres unités d'exécution propres à l'utilisateur (mineur), en utilisant toujours l'ID utilisateur du client et un mot de passe PassTicket pour l'authentification. Ces unités d'exécution offrent des services propres à l'utilisateur demandés par le client.

La description précédente illustre la conception orientée unité d'exécution de RSE. Au lieu de démarrer un espace adresse par utilisateur, plusieurs utilisateurs sont gérés par un seul espace adresse du pool d'unités d'exécution. Dans le pool d'unités d'exécution, chaque logiciel de fouille de données (service propre à l'utilisateur) est actif dans sa propre unité d'exécution dans le contexte de sécurité de l'utilisateur qui lui est attribué, ce qui garantit la sécurité de la configuration. Cette conception permet de gérer un grand nombre d'utilisateurs avec une quantité de ressources limitée, ce qui n'implique pas que chaque client va utiliser plusieurs unités d'exécution (au moins 17, selon les tâches réalisées).

Du point de vue du réseau, Developer for system z agit comme FTP en mode passif. Le client se connecte à un point focal (le démon RSE), supprime la connexion, puis se connecte de nouveau à un numéro de port fourni par le point focal. La logique ci-dessous permet de contrôler la sélection du port utilisé pour la deuxième connexion :

1. Si le client a indiqué un numéro de port différent de zéro dans l'onglet des propriétés du sous-système, le serveur RSE utilise ce port pour assurer la liaison. Si ce port n'est pas disponible, la connexion n'aboutit pas.
2. Si `_RSE_PORTRANGE` est spécifié dans `rsed.envvars`, le serveur RSE établit une liaison avec un port à partir de cette plage. Si aucun port n'est disponible, la connexion n'aboutit pas. Le serveur RSE n'a pas exclusivement besoin du port pendant la durée de la connexion client. Aucun autre serveur RSE ne peut établir une liaison avec le port que dans l'intervalle de temps entre la liaison (du serveur) et la connexion (du client). Cela signifie que la plupart des connexions utiliseront le premier port de la plage, les autres valeurs de la plage servant de mémoire tampon dans le cas de plusieurs connexions simultanées.
3. Si aucune limite n'est définie, le serveur RSE établit une liaison avec le port 0. Il en résulte que TCP/IP choisit le numéro de port.

L'utilisation du mot de passe PassTickets pour tous les services z/OS impliquant une authentification permet à Developer for System z d'appeler ces services à volonté, sans stocker le mot de passe ni inviter continuellement l'utilisateur à l'indiquer. L'utilisation des mots de passe PassTickets pour tous les services z/OS permet également d'utiliser d'autres méthodes d'authentification (mots de passe à usage unique et certificats X.509, par exemple).

Débogueur intégré

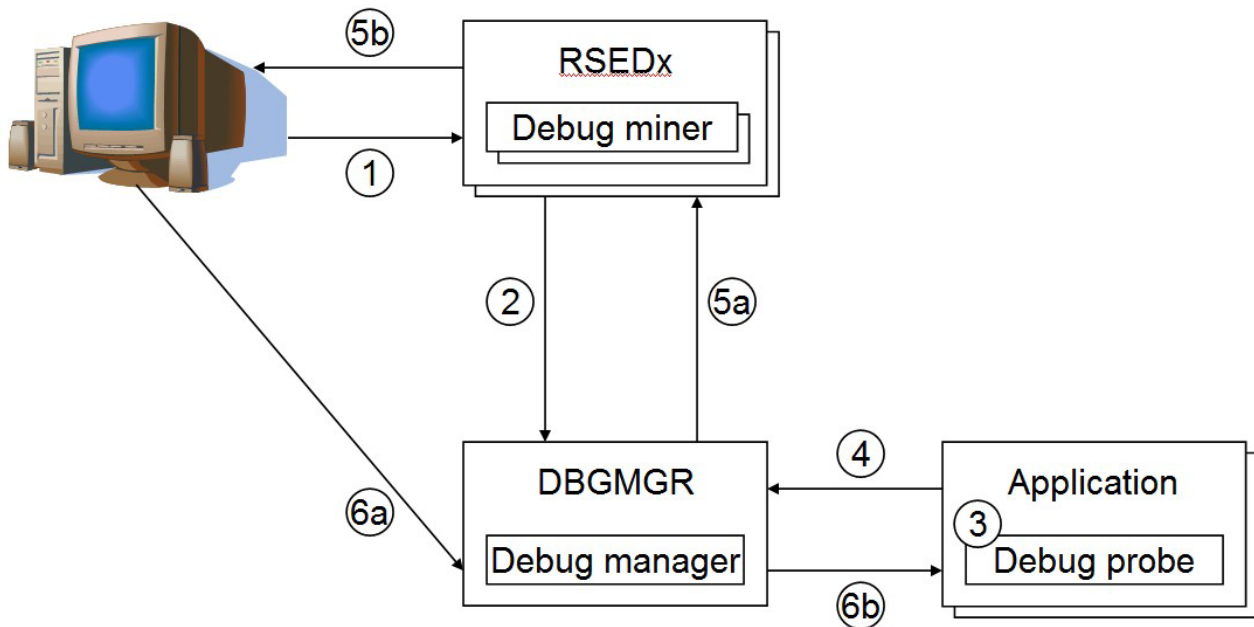


Figure 5. Débogueur intégré

Le débogueur intégré permet de déboguer différentes applications. La figure 5 montre une présentation schématique de la manière dont un client Developer for System z peut déboguer une application.

1. Le client se connecte à l'hôte à l'aide des informations de connexion Developer for System z normales.
2. Dans le cadre de la connexion, un logiciel de fouille de données de débogage enregistre l'utilisateur auprès du gestionnaire de débogage, qui est actif dans la tâche démarrée DBGMGR.
3. Lorsqu'une application est démarrée avec un indicateur signalant que celle-ci doit être déboguée, Language Environment (LE) appelle la sonde de débogage.
4. La sonde de débogage s'enregistre auprès du gestionnaire de débogage.
5. A l'aide du logiciel de fouille de données de débogage, le gestionnaire de débogage avertit le client Developer for System z de l'utilisateur qui reçoit cette session de débogage. Si l'utilisateur n'est pas enregistré à ce stade, la session de débogage se met en veille en attendant que l'utilisateur s'enregistre auprès du gestionnaire de débogage.
6. Le moteur de débogage dans le client contacte le gestionnaire de débogage qui à son tour transmet les données entre le moteur de débogage et la sonde de débogage dans les deux sens.

CARMA

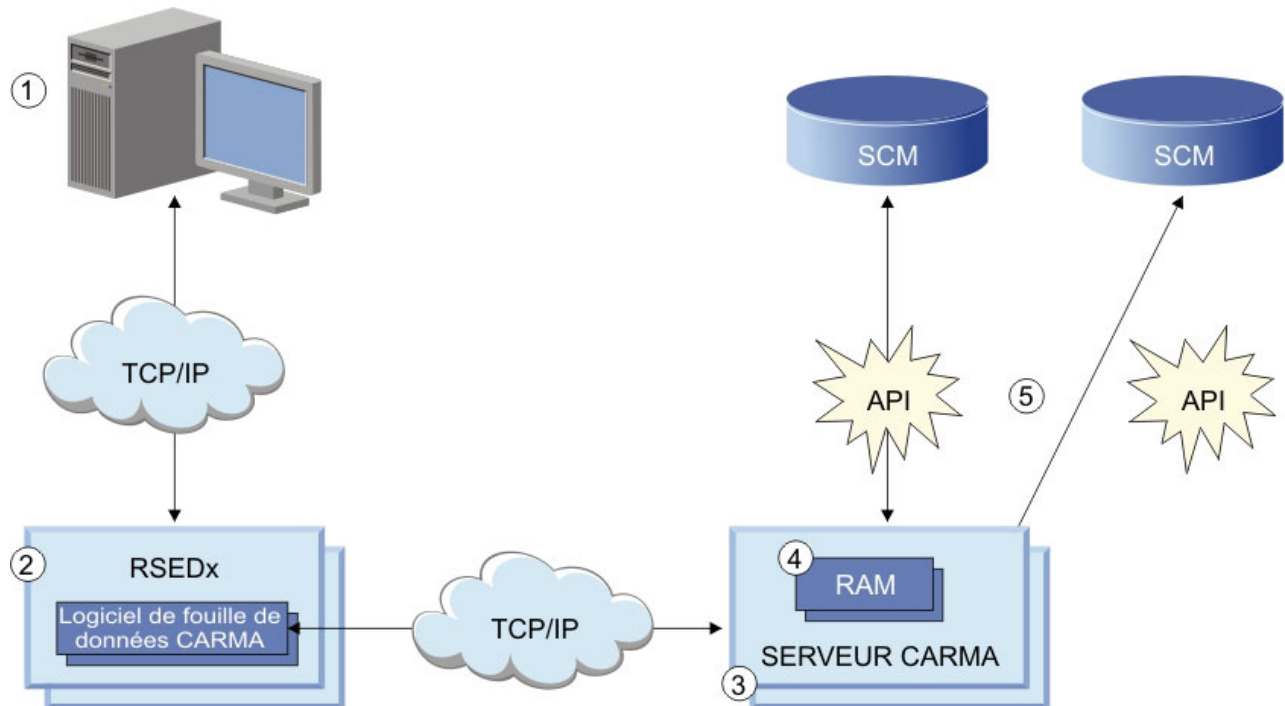


Figure 6. Flux CARMA

CARMA (Common Access Repository Manager) permet d'accéder à un SCM (Software Configuration Manager) basé sur un hôte, par exemple CA Endevor SCM. La figure 6 explique schématiquement comment un client Developer for System z peut accéder à un SCM (Software Configuration Manager) de type hôte pris en charge.

1. Le client dispose d'un plug-in Common Access Repository Manager (CARMA).
2. Ce plug-in communique avec l'exploitant CARMA, actif comme unité d'exécution utilisateur dans le pool d'unités d'exécution RSE (RSEDx). Cette communication est établie par l'intermédiaire de la connexion RSE existante.
3. Lorsque le client demande l'accès à un SCM, l'exploitation CARMA se lie à un port TCP/IP et démarre un serveur CARMA utilisateur avec le numéro de port comme argument de démarrage. Le serveur CARMA se connecte ensuite à ce port et utilise ce chemin pour la communication avec le client. Notez que les SCM basés sur l'hôte s'attendent à ce que les espaces adresse d'utilisateur unique accèdent à leurs services, ce qui nécessite le démarrage par CARMA d'un serveur CARMA par utilisateur. Il n'est pas possible de créer un serveur unique prenant en charge plusieurs utilisateurs.
4. Le serveur CARMA charge le gestionnaire RAM (Repository Access Manager) qui prend en charge le SCM demandé.
5. Le gestionnaire RAM traite les informations techniques de l'interaction avec le SCM et présente une interface commune au client.

Fichiers de configuration CARMA

Developer for System z prend en charge plusieurs méthodes pour démarrer un serveur CARMA. Chaque méthode offre des avantages, mais présente également des inconvénients. Developer for System z fournit également plusieurs RAM

(Repository Access Managers) qui peuvent être divisés en deux groupes : RAM de production et RAM exemples. Diverses combinaisons de RAM et de méthodes de démarrage de serveur sont disponibles dans une installation préconfigurée.

Toutes les méthodes de démarrage de serveur ont un fichier de configuration commun, CRASRV.properties, qui définit, entre autres, la méthode de démarrage utilisée.

CRASTART

La méthode "CRASTART" démarre le serveur CARMA sous la forme d'une sous-tâche dans RSE. Elle offre une configuration très flexible grâce à l'utilisation d'un fichier de configuration distinct qui définit les attributions de fichiers et les appels de programme nécessaires pour démarrer un serveur CARMA. Cette méthode offre les meilleures performances et utilise le moins de ressources mais requiert cependant que le module CRASTART se trouve dans LPA.

RSE appelle le module chargeable CRASTART qui utilise les définitions dans crastart*.conf pour créer un environnement valide pour exécuter des commandes TSO et ISPF par lots. Developer for System z utilise cet environnement pour exécuter le serveur CARMA, CRASERV. Developer for System z fournit plusieurs fichiers crastart*.conf, chaque fichier étant préconfiguré pour un gestionnaire donné.

Soumission par lots

Cette méthode démarre le serveur CARMA en envoyant un travail. Il s'agit de la méthode par défaut utilisée dans les fichiers de configuration fournis. L'avantage de cette méthode est que les journaux CARMA sont facilement accessibles dans la sortie de travaux. Elle permet également d'utiliser un JCL de serveur personnalisé pour chaque développeur qui sera géré par le développeur lui-même. Toutefois, cette méthode utilise un initiateur JES pour chaque développeur qui démarre un serveur CARMA.

RSE appelle CLIST CRASUB* qui envoie un document incorporé JCL pour créer un environnement valide pour exécuter des commandes TSO et ISPF par lots. Developer for System z utilise cet environnement pour exécuter le serveur CARMA, CRASERV. Developer for System z fournit plusieurs membres CRASUB*, chaque membre étant préconfiguré pour un gestionnaire donné.

Propriétaire du verrou d'un fichier

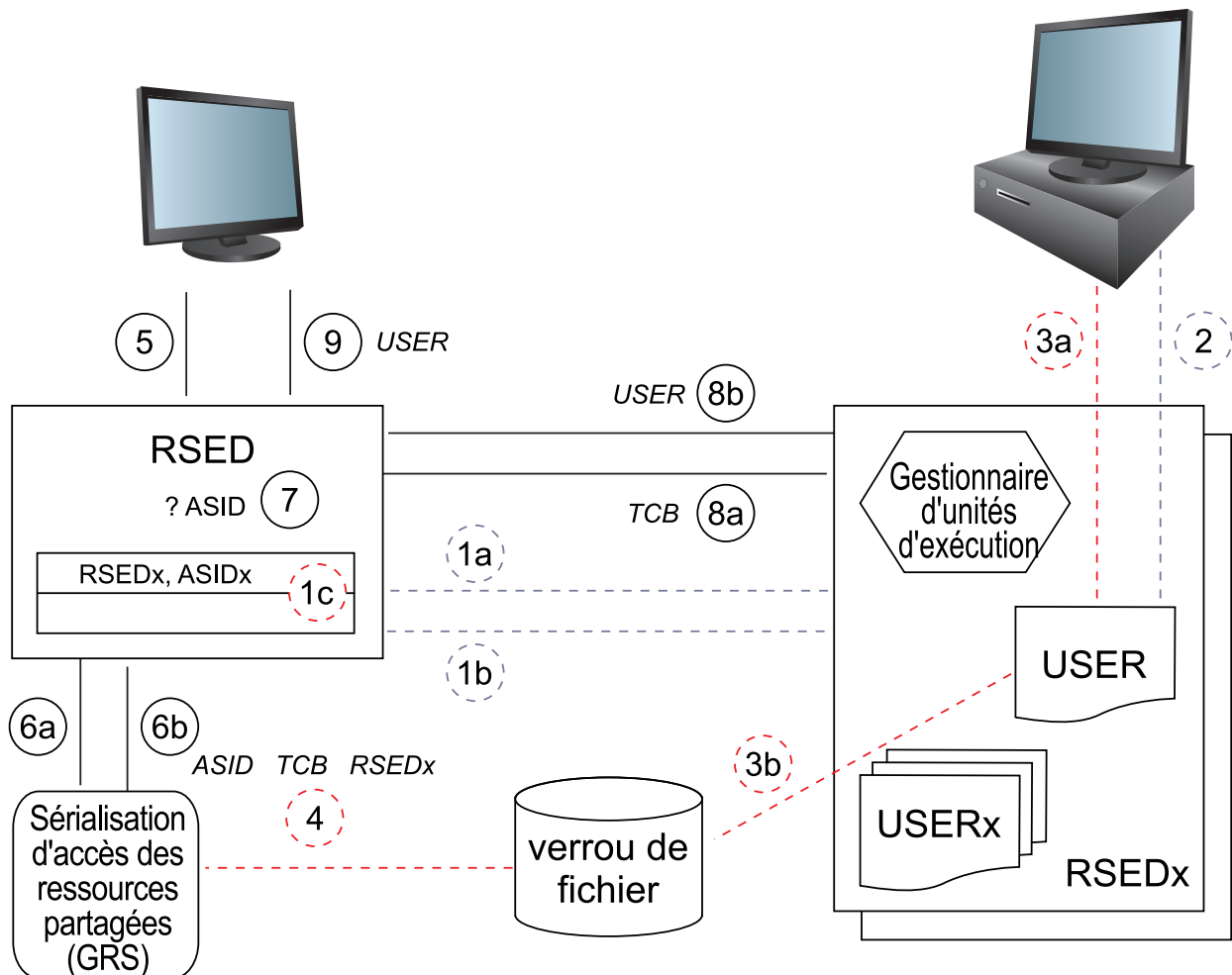


Figure 7. Flux de détermination de mise en file d'attente d'un fichier

La figure 7 explique schématiquement comment le démon RSE détermine quel client Developer for System z possède un verrou de fichier.

1. Le démon RSE (RSED) crée un pool d'unités d'exécution (RSEDx). Pour confirmer un bon démarrage, le pool d'unités d'exécution indique son identificateur d'espace adresse (ASID) au démon RSE qui le stocke dans le bloc de contrôle créé pour le suivi de ce pool d'unités d'exécution.
2. Le client se connecte, ce qui permet de créer une unité d'exécution de serveur RSE propre à l'utilisateur (USER) à l'intérieur d'un pool d'unités d'exécution (RSEDx). Chaque unité d'exécution dispose d'un ID de bloc de contrôle des tâches unique.
3. Le client ouvre un fichier en mode d'édition, qui informe le serveur RSE d'appliquer un verrou exclusif (mise en file d'attente) au fichier.
4. Le système enregistre l'identificateur d'espace adresse, le bloc de contrôle des tâches et le nom de tâche (RSEDx) du demandeur dans le cadre du processus de mise en file d'attente. Ces informations sont stockées dans les files d'attente de sérialisation d'accès des ressources partagées (GRS).

5. Un opérateur interroge le démon RSE pour obtenir le statut de verrouillage du fichier.
6. Le démon RSE analyse les files d'attente GRS pour savoir si le fichier est verrouillé et extrait l'identificateur d'espace adresse, le bloc de contrôle des tâches et le nom de la tâche du propriétaire du verrou.
7. L'identificateur d'espace adresse extrait est comparé à l'identificateur d'espace adresse des différents pools d'unités d'exécution.
8. Le démon RSE demande au pool d'unités d'exécution propriétaire de l'identificateur d'espace adresse de déterminer quel utilisateur possède le bloc de contrôle des tâches.
9. L'ID utilisateur client associé est renvoyé au demandeur en cas de correspondance. Sinon, le nom de tâche extrait de la sérialisation d'accès des ressources partagées est renvoyé.

Avec la configuration à un seul serveur de Developer for System z, où plusieurs utilisateurs sont attribués à un seul espace adresse de pool d'unités d'exécution, z/OS n'a plus la possibilité de savoir qui possède un verrou sur un fichier ou un membre avec la commande d'opérateur **DISPLAY GRS,RES=(*,dataset*)**. Les commandes système s'arrêtent au niveau de l'espace adresse, qui correspond au pool d'unités d'exécution.

Pour régler ce problème, Developer for System z propose la commande d'opérateur **MODIFY rsed APPL=DISPLAY OWNER,DATASET=dataset**, comme indiqué dans "Commandes de l'opérateur" du manuel *Guide de configuration de l'hôte* (SC23-7658). La commande d'opérateur peut résoudre tous les verrous de fichier/membre placés par les utilisateurs RSE, et ceux placés par d'autres produits (ISPF, par exemple).

Libération d'un verrou

Dans des circonstances normales, un fichier ou un membre est verrouillé lorsque le client l'ouvre en mode édition, et libéré lorsque le client ferme la session d'édition.

Certaines conditions d'erreur peuvent gêner le fonctionnement prévu de ce mécanisme. Dans ce cas, l'utilisateur qui détient le verrou peut être annulé à l'aide de la commande d'opérateur **modify cancel** de RSE, tel que décrit dans "Commandes de l'opérateur" de *Guide de configuration de l'hôte* (SC11-6285). Les verrous du fichier actif appartenant à cet utilisateur sont libérés lors du processus.

Structure de répertoires z/OS UNIX

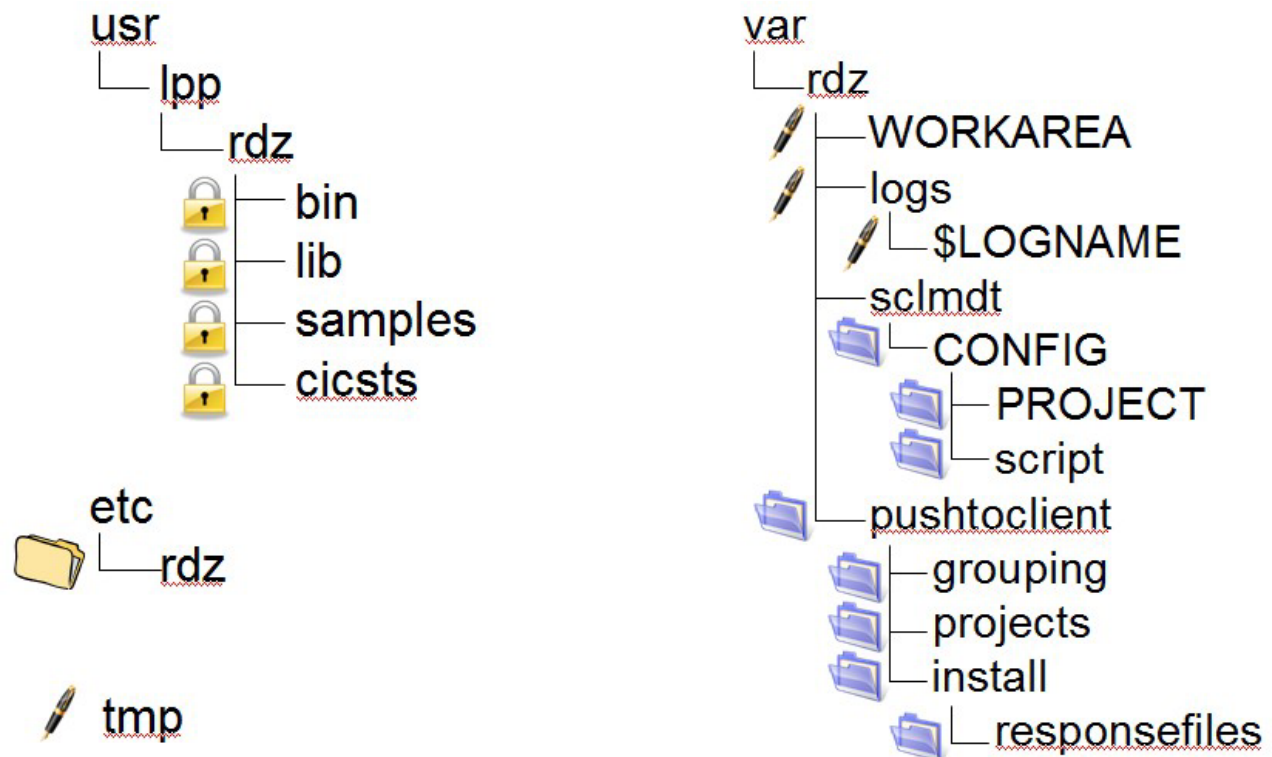


Figure 8. Structure de répertoires z/OS UNIX

La figure 8 présente les répertoires z/OS UNIX utilisés par Developer for System z. La liste suivante décrit chaque répertoire en contact avec Developer for System z, le mode de changement d'emplacement et qui gère les données qu'il contient.

- /usr/lpp/rdz/ est la racine du code produit Developer for System z. L'emplacement réel est spécifié dans la tâche démarrée RSED (variable HOME). Les fichiers sont gérés par SMP/E.
- /etc/rdz/ contient les fichiers de configuration de RSE et du logiciel de fouille de données. L'emplacement réel est spécifié dans la tâche démarrée RSED (variable CNFG). Les fichiers sont gérés par le programmeur système.
- /tmp/ est utilisé par la passerelle de client TSO/ISPF d'ISPF et les différents logiciels de fouille de données pour l'enregistrement des données temporaires. Certains programmes de vérification de l'installation stockent leur sortie dans ce répertoire. Les fichiers qui s'y trouvent sont gérés par ISPF, les logiciels de fouille de données et les programmes de vérification de l'installation. L'emplacement peut être personnalisé à l'aide de la variable TMPDIR dans rsed.envvars. Il s'agit également de l'emplacement par défaut des fichiers de vidage Java, qui peuvent être personnalisés avec la variable _CEE_DUMPTARG de rsed.envvars.

Remarque : /tmp/ requiert le masque de contrôle des données de droits 777 permettant à chaque client de créer des fichiers temporaires.

- /var/rdz/WORKAREA/ est utilisé par la passerelle de client TSO/ISPF et par SCLMDT pour transférer des données entre z/OS UNIX et les espaces adresse MVS. L'emplacement réel est indiqué dans rsed.envvars (variable CGI_ISPWORK). Les fichiers sont gérés par ISPF et SCLMDT.

Remarque : /var/rdz/WORKAREA/ requiert le masque de contrôle des données de droits 777 permettant à chaque client de créer des fichiers temporaires.

- /var/rdz/logs/ comporte les fichiers journaux du démon RSE et des serveurs de pool d'unités d'exécution. L'emplacement réel est indiqué dans rsed.envvars (variable daemon.log). Les fichiers sont gérés par RSE.
- /var/rdz/logs/\$LOGNAME/ comporte les journaux spécifiques à l'utilisateur du serveur RSE et des logiciels de fouille de données. L'emplacement réel est indiqué dans rsed.envvars (variable user.log et DSTORE_LOG_DIRECTORY). Les fichiers sont gérés par RSE et les logiciels de fouille de données.

Remarque : /var/rdz/logs/ requiert le masque de contrôle des données de droits 777 permettant à chaque client de créer son répertoire \$LOGNAME et d'enregistrer les fichiers journaux propres à l'utilisateur.

- /var/rdz/sclmdt/CONFIG/ comporte les fichiers de configuration SCLMDT généraux. L'emplacement réel est spécifié dans rsed.envvars (variable SCLMDT_CONF_HOME). Les fichiers sont gérés par l'administrateur SCLM.
- /var/rdz/sclmdt/CONFIG/PROJECT/ comporte les fichiers de configuration du projet SCLMDT. L'emplacement réel est spécifié dans rsed.envvars (variable SCLMDT_CONF_HOME). Les fichiers sont gérés par l'administrateur SCLM.
- /var/rdz/sclmdt/CONFIG/script/ comporte les scripts associés à SCLMDT qui peuvent être utilisés par d'autres produits. L'emplacement réel n'est indiqué nulle part. Les fichiers sont gérés par l'administrateur SCLM.
- /var/rdz/pushtoclient/ contient les fichiers de configuration du client, les informations de mise à jour du produit client et les informations du projet résidant sur l'hôte envoyées au client lors de la connexion au système hôte. L'emplacement est défini dans pushtoclient.properties (variable pushtoclient.folder). Les fichiers qui s'y trouvent sont gérés par un administrateur de client Developer for System z.
- /var/rdz/pushtoclient/grouping/ contient les fichiers de configuration du client, les informations de mise à jour du produit client et les informations du projet résidant sur l'hôte envoyées au client lors de la connexion au système hôte. L'emplacement réel est spécifié dans pushtoclient.properties (variable pushtoclient.folder plus le suffixe /grouping). Les fichiers qui s'y trouvent sont gérés par un administrateur de client Developer for System z.
- /var/rdz/pushtoclient/projects/ contient les fichiers de définition du projet résidant sur l'hôte. L'emplacement réel est spécifié dans le répertoire /var/rdz/pushtoclient/keymapping.xml, lequel est créé et géré par un administrateur de client Developer for System z. Les fichiers qui s'y trouvent sont gérés par un chef de projet ou un responsable du développement.
- /var/rdz/pushtoclient/install/ contient les fichiers de configuration utilisés pour mettre à jour la version du produit client lors de la connexion à l'hôte. L'emplacement réel est spécifié dans le répertoire /var/rdz/pushtoclient/keymapping.xml, lequel est créé et géré par un administrateur de client Developer for System z. Les fichiers qui s'y trouvent sont gérés par un administrateur de client.
- /var/rdz/pushtoclient/install/responsefiles/ contient les fichiers de configuration utilisés pour mettre à jour la version du produit client lors de la connexion à l'hôte. L'emplacement réel est spécifié dans le répertoire /var/rdz/pushtoclient/keymapping.xml, lequel est créé et géré par un administrateur de client Developer for System z. Les fichiers qui s'y trouvent sont gérés par un administrateur de client.

Droits de mise à jour des administrateurs non système

Les données contenues dans les répertoires `/var/rdz/pushtoclient/` sont gérées par des administrateurs non système, tels que les chefs de projet, qui peuvent disposer de droits de mise à niveau limités sous z/OS UNIX. Par conséquent, il est important de bien comprendre comment z/OS UNIX définit les droits d'accès lors de la création des fichiers pour garantir une configuration à la fois gérable et sécurisée.

Les normes UNIX déterminent la définition des autorisations pour trois types d'utilisateurs : propriétaire, groupe et autres. Des droits de lecture, d'écriture et d'exécution peuvent être définis pour chaque type de façon individuelle.

z/OS UNIX affecte au numéro utilisateur (ID utilisateur) et à l'identificateur de groupe (ID groupe) les valeurs suivantes lors de la création d'un fichier :

- Le numéro utilisateur est défini sur le numéro utilisateur effectif de l'unité d'exécution.
- L'identificateur de groupe est défini sur l'identificateur de groupe du répertoire propriétaire. Si le profil de sécurité `FILE.GROUPOWNER.SETGID` est défini dans la classe `UNIXPRIV`, l'identificateur de groupe effectif de l'unité d'exécution est en revanche utilisé par défaut. Pour plus d'informations, voir *UNIX System Services Planning* (GA22-7800).

Chaque site peut définir son propre masque de droits d'accès par défaut ; toutefois, un masque commun octroie des droits d'accès en lecture et en écriture au propriétaire ainsi que des droits d'accès en lecture au groupe et aux autres.

Les données contenues dans le répertoire `/var/rdz/pushtoclient/` sont créées avec le masque de droits d'accès défini dans la directive `file.permission` de `pushtoclient.properties`. La valeur par défaut prévoit des droits d'accès en lecture et en écriture pour le propriétaire et le groupe ainsi que des droits d'accès en lecture pour les autres. Tous bénéficient de droits d'exécution. Les autorisations d'accès définitives doivent prévoir des droits de lecture et d'exécution pour tous et des droits d'accès en écriture pour les administrateurs de client Developer for System z chargés de la gestion des données.

Les données contenues dans le répertoire `/var/rdz/pushtoclient/projects/` sont créées sans masque de droits d'accès spécifique. Les autorisations d'accès définitives doivent prévoir des droits d'accès en lecture pour tous et des droits d'accès en écriture pour les chefs de projet chargés de la gestion des données.

Commandes de sécurité utiles

Pour faire en sorte qu'un groupe de chefs de projet ou d'administrateurs de client Developer for System z puisse gérer les données dans ces répertoires, il se peut que l'administrateur de la sécurité doive créer un groupe associé à un segment OMVS valide. Il est préférable que ce groupe soit le groupe par défaut des ID utilisateur impliqués. Reportez-vous à la documentation *Security Server RACF Command Language Reference* (SA22-7687) pour obtenir plus d'informations sur les exemples de commandes RACF suivantes :

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

Commandes z/OS UNIX utiles

Reportez-vous à la documentation *UNIX System Services Command Reference* (SA22-7802) pour plus d'informations sur les exemples de commandes z/OS UNIX suivants :

- Utilisez la commande z/OS UNIX **ls** suivante pour afficher tous les fichiers d'un répertoire.
`ls -lR /var/rdz/pushtoclient/`
- Utilisez la commande z/OS UNIX **chown** suivante pour modifier le propriétaire d'un répertoire et de tous les fichiers correspondants.
`chown -R IBMUSER /var/rdz/pushtoclient/`
- Utilisez la commande z/OS UNIX **chgrp** suivante pour affecter un groupe au répertoire et à tous les fichiers correspondants.
`chgrp -R RDZPROJ /var/rdz/pushtoclient/`
- Utilisez la commande z/OS UNIX **chmod** suivante pour octroyer au propriétaire et au groupe les droits d'accès en écriture sur le répertoire et tous les fichiers correspondants. Les autres utilisateurs bénéficient de droits de lecture. Tous bénéficient de droits d'exécution.
`chmod -R 775 /var/rdz/pushtoclient/`

Exemple de configuration

Dans le scénario ci-dessous, tous les chefs de projet de développement, à savoir une équipe composée de trois chefs de projet, sont chargés de jouer le rôle d'administrateur de client Developer for System z.

L'administrateur de la sécurité a déjà affecté à l'équipe un groupe par défaut (RDZPROJ) doté d'un ID groupe unique (1200). L'ID utilisateur n'est pas associé à des privilèges spécifiques (comme numéro utilisateur 0) sous z/OS UNIX. L'administrateur de la sécurité n'a pas défini le profil FILE.GROUPOWNER.SETGID. Par conséquent, z/OS UNIX va utiliser l'ID groupe du répertoire lors de la création de fichiers. L'ID utilisateur IBMUSER (doté du numéro utilisateur 0 et du groupe par défautSYS1) a été utilisé par le programmeur-système pour créer le répertoire /var/rdz/pushtoclient.

1. Le programmeur-système réserve exclusivement les droits d'accès en écriture /var/rdz/pushtoclient au propriétaire et au groupe :

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

Remarque : Le travail FEKSETUP utilisé lors de la configuration de la personnalisation réalise déjà cette étape.

2. Le programmeur-système définit le groupe par défaut RDZPROJ comme étant le groupe propriétaire :

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

Cette étape met fin à la configuration permettant de limiter les droits d'accès en écriture /var/rdz/pushtoclient au programmeur-système (IBMUSER) et aux chefs de projet (RDZPROJ).

Chapitre 2. Remarques relatives à la sécurité

Developer for System z offre aux utilisateurs un accès grand système sur un poste de travail qui ne correspond pas à un grand système. La validation des demandes de connexion, l'établissement de communications sécurisées entre l'hôte et le poste de travail, l'autorisation et l'activité d'audit sont donc des aspects fondamentaux de la configuration d'un produit.

Pour être efficaces, les mécanismes de sécurité utilisés par les serveurs et les services Developer for System z doivent reposer sur la sécurité des fichiers et systèmes de fichiers les contenant. Cela implique que seuls les administrateurs système habilités doivent pouvoir mettre à jour les bibliothèques de programmes et les fichiers de configuration.

Les rubriques suivantes sont traitées dans le présent chapitre :

- «Méthodes d'authentification», à la page 20
- «Sécurité des connexions», à la page 21
- «Utilisation de PassTickets», à la page 23
- «Consignation dans le journal d'audit», à la page 24
- «Sécurité JES», à la page 26
- «Communication chiffrée via SSL/TLS», à la page 30
- «Authentification du client à l'aide de certificats X.509», à la page 31
- «Vérification du port d'entrée (POE)», à la page 35
- «Informations diverses», à la page 39
- «Modification des fonctions client», à la page 35
- «Groupes de développeurs de la fonction d'envoi au client», à la page 37
- «Sécurité du débogage», à la page 38
- «CICSTS, sécurité», à la page 38
- «SCLM, sécurité», à la page 40
- «Fichiers de configuration Developer for System z», à la page 40
- «Définitions de sécurité», à la page 43

Remarque : L'Explorateur de systèmes distants (RSE), qui fournit des services de base comme la connexion du client à l'hôte, se compose de deux entités logiques.

- Le démon RSE qui gère la configuration de la connexion et qui est lancé en tant que tâche démarrée ou en tant que travail à exécution longue.
- Le serveur RSE qui gère des demandes client individuelles et qui est démarré en tant qu'unité d'exécution dans un ou plusieurs processus enfant par le démon RSE.

Voir Chapitre 1, «Description de Developer for System z», à la page 3 pour en savoir plus sur la conception de base de Developer for System z.

Méthodes d'authentification

Developer for System z prend en charge plusieurs méthodes d'authentification d'un ID utilisateur fourni par un client lors de la connexion.

- Un ID utilisateur et un mot de passe
- Un ID utilisateur et un mot de passe utilisable une seule fois
- Certificat X.509

Remarque : Les données d'authentification fournies par le client ne sont utilisées qu'une seule fois, lors de la configuration de la connexion initiale. Une fois qu'un ID utilisateur est authentifié, ce dernier ainsi que les mots de passe PassTicket générés automatiquement sont utilisés pour toutes les actions qui requièrent une authentification.

Un ID utilisateur et un mot de passe

Le client fournit un ID utilisateur et un mot de passe lors de la connexion. L'ID utilisateur et le mot de passe sont utilisés pour authentifier l'utilisateur auprès de votre logiciel de sécurité.

Un ID utilisateur et un mot de passe utilisable une seule fois

Un mot de passe utilisable une seule fois peut être généré par un produit tiers à partir d'un jeton unique. Ce type de mot de passe renforce la configuration de sécurité car le sème unique ne peut pas être copié ni être utilisé sans que l'utilisateur en soit informé et un mot de passe intercepté est inutilisable car il n'est valide qu'une seule fois.

Lors de la connexion, le client indique un ID utilisateur et un mot de passe utilisable une seule fois, qui permet d'authentifier l'ID utilisateur avec l'exit de sécurité fourni par le tiers. Cet exit de sécurité doit ignorer les mots de passe PassTicket utilisés pour traiter les demandes d'authentification lors d'un traitement normal. Les mots de passe PassTicket doivent être traités par votre logiciel de sécurité.

Certificat X.509

Un tiers peut fournir un ou plusieurs certificats X.509 qui permettent l'authentification d'un utilisateur. Lorsqu'ils sont stockés sur des unités sécurisées, les certificats X.509 offrent une configuration sécurisée associée à une grande facilité d'utilisation (pas d'ID utilisateur ni de mot de passe nécessaires).

Lors de la connexion, le client fournit un certificat sélectionné et éventuellement une extension, qui permet d'authentifier l'ID utilisateur auprès de votre logiciel de sécurité.

Remarque : Cette méthode d'authentification est prise en charge uniquement par la méthode de connexion du démon RSE. SSL doit également être activé.

Authentification du moniteur de travaux JES

L'authentification du client est effectuée par le démon RSE (ou REXEC/SSH) dans le cadre d'une demande de connexion client. Une fois que l'utilisateur est authentifié, des mots de passe PassTicket générés automatiquement sont utilisés pour toutes les demandes d'authentification ultérieures, y compris la connexion automatique au moniteur de travaux JES.

Pour que le moniteur de travaux JES puisse valider l'ID utilisateur et le mot de passe PassTicket présenté par RSE, il doit être autorisé à évaluer le mot de passe PassTicket. Cette procédure implique les éléments suivants :

- Le module de chargement FEJMON, situé par défaut dans la bibliothèque de chargement FEK.SFEKAUTH, doit disposer d'une autorisation APF.
- RSE et le moniteur de travaux JES doivent utiliser le même ID d'application (APPLID). Par défaut, les deux serveurs utilisent FEKAPPL comme APPLID mais cette valeur peut être modifiée via la directive APPLID dans rsed.envvars pour RSE et FEJCNFG pour les moniteurs de travaux JES.

Remarque : Les anciens clients (version 7.0 et plus ancienne) communiquent directement avec le moniteur de travaux JES. Pour ces connexions, seule la méthode d'authentification par ID utilisateur et mot de passe est prise en charge.

Authentification du gestionnaire de débogage

L'authentification du client est effectuée par le démon RSE (ou REXEC/SSH) dans le cadre d'une demande de connexion client. Une fois que l'utilisateur est authentifié, des mots de passe PassTicket générés automatiquement sont utilisés pour toutes les demandes d'authentification ultérieures, y compris la connexion automatique au gestionnaire de débogage.

Pour que le gestionnaire de débogage puisse valider l'ID utilisateur et le mot de passe PassTicket présenté par RSE, il doit être autorisé à évaluer le mot de passe PassTicket. Cela implique que le module de chargement AQEZPCM, situé par défaut dans la bibliothèque de chargement FEK.SFEKAUTH, doit disposer d'une autorisation APF.

Sécurité des connexions

Différents niveaux de sécurité des communications sont pris en charge par RSE, qui contrôle les communications entre le client et la plupart des services Developer for System z :

- Les communications (client-hôte) externes peuvent être limitées à des ports spécifiques. Cette fonction est désactivée par défaut.
- Les communications (client-hôte) externes peuvent être chiffrées à l'aide de SSL ou TLS. Cette fonction est désactivée par défaut.
- La vérification du port d'entrée peut être utilisée afin d'autoriser l'accès hôte uniquement aux adresses TCP/IP sécurisées. Cette fonction est désactivée par défaut.

Certains services Developer for System z facultatifs utilisent un chemin de communication (client-hôte) externe distinct :

- Les communications du débogueur intégré peuvent être chiffrées à l'aide de TLS.
- Les communications du gestionnaire de déploiement d'application peuvent être chiffrées à l'aide de SSL lorsque l'interface de service Web est utilisée.

Developer for System z repose sur des produits tiers, tels que le serveur TN3270 pour fournir certains services. Pour plus d'informations sur les options de sécurité de connexion, reportez-vous à la documentation produit associée.

Limite des communications externes à des ports spécifiques

Le programmeur système peut spécifier les ports sur lesquels le serveur RSE peut communiquer avec le client. Par défaut, n'importe quel port disponible peut être utilisé. Cette gamme de ports n'a aucune connexion avec le port du démon RSE.

Afin de mieux comprendre l'utilisation des ports, une brève description du processus de connexion RSE est incluse ci-après :

1. Le client se connecte au port hôte 4035 du démon RSE.
2. Le démon RSE crée une unité d'exécution de serveur RSE.
3. Le serveur RSE ouvre un port hôte pour que le client se connecte. Le choix de ce port peut être configuré par l'utilisateur, soit au niveau du client dans l'onglet des propriétés du sous-système (méthode non recommandée) soit par l'intermédiaire de la définition `_RSE_PORTRANGE` du fichier `rsed.envvars`.
4. Le démon RSE renvoie le numéro de port au client.
5. Le client se connecte au port hôte.

Remarque : Le processus est identique pour la méthode de connexion alternative (facultative) utilisant REXEC/SSH, qui est décrite "(Facultatif) Utilisation de REXEC (ou SSH)" dans *Guide de configuration de l'hôte* (SC11-6285).

Chiffrement des communications à l'aide de SSL ou TLS

Tous les flux de données Developer for System z externes qui transitent par RSE peuvent être chiffrés à l'aide de SSL (Secure Socket Layer) ou Transport Layer Security (TLS). L'utilisation de communications chiffrées est contrôlée par les paramètres du fichier de configuration `ssl.properties`, comme décrit dans la section «Communication chiffrée via SSL/TLS», à la page 30. La variable `DSTORE_SSL_ALGORITHM` de la directive `_RSE_JAVA_OPTS` du fichier `rsed.envvars` vous permet de choisir entre SSL et son successeur TLS pour la méthode de chiffrement, comme indiqué à la section sur la définition de paramètres de démarrage Java supplémentaires avec `_RSE_JAVA_OPTS` dans le document *Guide de configuration de l'hôte* (SC23-7658).

Le moteur de débogueur intégré sur le client se connecte au gestionnaire de débogage sur l'hôte. L'utilisation de SSL ou TLS est contrôlée par une règle AT-TLS (Application Transparent TLS).

L'émulateur de connexion à l'hôte sur le client se connecte à un serveur TN3270 sur l'hôte. L'utilisation de SSL ou TLS est contrôlée par TN3270, comme indiqué dans le document *Communications Server IP Configuration Guide* (SC31-8775).

Les actions à distance (basées sur l'hôte) dans les sous-projets z/OS UNIX utilisent un serveur REXEC ou SSH sur l'hôte. La communication SSH est toujours chiffrée à l'aide de SSL.

Le client du gestionnaire de déploiement d'application utilise le service Web TS CICS ou l'interface RESTful pour appeler les services hôte du gestionnaire de déploiement d'application. L'utilisation de SSL est contrôlée par CICS TS, comme indiqué dans la documentation *RACF Security Guide for CICS TS*.

Vérification du port d'entrée

Developer for System z prend en charge la vérification du port d'entrée, ce qui permet à l'hôte d'accéder uniquement aux adresses TCP/IP sécurisées. L'utilisation du port d'entrée est contrôlée par la définition des profils spécifiques dans votre

logiciel de sécurité et la directive `enable.port.of.entry` dans `rased.envvars` (voir section «Vérification du port d'entrée (POE)», à la page 35).

Notez que l'activation du port d'entrée a une incidence sur d'autres applications TCP/IP prenant en charge la vérification du port d'entrée, telles que INETD.

Utilisation de PassTickets

Après la connexion, des mots de passe PassTicket sont utilisés pour établir la sécurité des unités d'exécution sur le serveur RSE. Cette fonction ne peut pas être désactivée. Les PassTickets sont des mots de passe générés par le système pour une durée d'environ 10 minutes. Les mots de passe PassTicket générés s'appuient sur l'algorithme de chiffrement DES, l'ID utilisateur, l'ID d'application, un horodatage (heure/date) et une clé confidentielle. Cette clé confidentielle est un nombre de 64 bits (16 caractères hexadécimaux) qui doit être définie pour votre logiciel de sécurité. Pour plus de sécurité, le logiciel de sécurité z/OS gère les PassTickets par défaut comme des mots de passe à usage unique.

Afin de mieux comprendre l'utilisation de PassTicket, une brève description du processus de sécurité RSE est incluse ci-après :

1. Le client se connecte au port hôte 4035 du démon RSE.
2. Le démon RSE authentifie le client en utilisant les données d'identification présentées par le client.
3. Le démon RSE crée un ID client unique et une unité d'exécution de serveur RSE.
4. Le serveur RSE génère un PassTicket et crée un environnement de sécurité pour le client, en utilisant le PassTicket comme mot de passe.
5. Le client se connecte au port hôte renvoyé par le démon RSE.
6. Le serveur RSE valide le client à l'aide de l'ID client.
7. Le serveur RSE utilise un PassTicket nouvellement généré comme mot de passe pour toutes les actions futures qui requièrent un mot de passe.

Remarque : Un mécanisme similaire est utilisé pour configurer des connexions sécurisées avec le gestionnaire de débogage.

Le mot de passe réel du client n'est plus nécessaire après l'authentification initiale car les produits de sécurité conformes à SAF peuvent évaluer à la fois les mots de passe Passticket et les mots de passe standard. Le serveur RSE génère et utilise un mot de passe PassTicket chaque fois qu'un mot de passe est requis ; un mot de passe valide (temporaire) est ainsi disponible pour le client.

L'utilisation de mots de passe PassTicket permet à RSE de configurer un environnement de sécurité propre à l'utilisateur sans avoir à stocker tous les ID et les mots de passe dans une table, qui pourrait être illégalement consultée. Ils permettent également de mettre en oeuvre des méthodes d'authentification client qui n'utilisent pas de mots de passe réutilisables, tels que des certificats X.509.

Les profils de sécurité des classes APPL et PTKTDATA sont nécessaires pour permettre l'utilisation de mots de passe PassTicket. Ce profils sont propres à l'application et n'ont pas d'incidence sur la configuration système actuelle.

Comme les mots de passe PassTicket sont propres à l'application, RSE et le moniteur de travaux JES doivent utiliser le même ID d'application (APPLID). Par

défaut, les deux serveurs utilisent FEKAPPL comme APPLID mais cette valeur peut être modifiée via la directive APPLID dans `rsed.envvars` pour RSE et FEJCNFG pour le moniteur de travaux JES.

Vous ne devez pas utiliser OMVSAPPL comme ID d'application, car il ouvrira la clé confidentielle de la plupart des applications z/OS UNIX. De la même manière, vous ne devez pas utiliser l'ID application par défaut MVS, lequel est MVS suivi par l'ID SMF du système, car il ouvrira la clé confidentielle de la plupart des applications MVS (y compris les travaux par lots des utilisateurs).

La plus petite unité d'un horodatage de passticket est 1 seconde. Cela signifie que tous les passtickets générés en moins d'une seconde par la même application pour le même ID utilisateur seront identiques. Ceci, combiné au fait que le logiciel de sécurité z/OS gère les passtickets comme des mots de passe à usage unique, constitue un problème pour Developer for System z lors de la connexion, car plusieurs passtickets seront requis en moins d'une seconde. Par conséquent, Developer for System z requiert l'activation d'un indicateur dans les définitions de passticket qui autorise la réutilisation des passtickets générés.

Avertissement : La demande de connexion du client n'aboutit pas si les mots de passe PassTickets ne sont pas correctement configurés.

Consignation dans le journal d'audit

Developer for System z prend en charge la consignation dans le journal d'audit des actions gérées par le démon RSE. Les journaux d'audit sont conservés sous la forme de fichiers texte dans le répertoire de journalisation du démon, au format CSV.

Contrôle d'audit

Plusieurs options définies dans `rsed.envvars` influencent la fonction d'audit, comme documenté dans "Définition de paramètres de démarrage supplémentaires Java avec `_RSE_JAVAOPTS`" du *Guide de configuration de l'hôte* (SC11-6285).

- La fonction d'audit est activée/désactivée à l'aide de l'option `enable.audit.log`.
- Les valeurs d'audit part défaut sont contrôlées par les options `audit.*`.
- L'emplacement des journaux d'audit est contrôlé par l'option `daemon.log`.
- La page de codes utilisée pour rédiger le journal d'audit est contrôlée par la directive `_RSE_HOST_CODEPAGE`, comme indiqué à la section "rsed.envvars, fichier de configuration RSE" du *Guide de configuration de l'hôte* (SC11-6285).

La commande de l'opérateur **modify switch** permet de passer manuellement à un nouveau fichier journal d'audit, comme indiqué à la section "Commandes de l'opérateur" du *Guide de configuration de l'hôte* (SC11-6285).

Un message d'avertissement est envoyé à la console lorsque l'espace disponible dans le système de fichiers qui contient les fichiers journaux d'audit est insuffisant. Le message de console (FEK103E) s'affiche régulièrement tant que l'incident lié au manque d'espace n'a pas été résolu. Reportez-vous à "Messages de console" dans le *Guide de configuration de l'hôte* (SC11-6285) pour obtenir une liste des messages de console générés par RSE.

Traitement de l'audit

Un nouveau fichier journal d'audit est démarré après un délai défini ou lorsque la commande de l'opérateur **modify switch** est exécutée. L'ancien fichier journal d'audit est enregistré sous `audit.log.yyyyymmdd.hhmmss`, où `yyyymmdd.hhmmss` représente la date/l'horodatage de fermeture de ce journal. La date/l'horodatage système attribué(e) au fichier indique la création du fichier journal. La combinaison des deux dates indique la période couverte par ce fichier journal d'audit.

Les directives `audit.action*` dans `rsed.envvars` vous permettent de spécifier un exit utilisateur (script de shell z/OS UNIX, programme z/OS UNIX REXX ou z/OS UNIX) qui sera appelé par RSE lors de la fermeture d'un journal d'audit. Cet exit utilisateur peut alors traiter les données du journal d'audit.

Les fichiers journaux d'audit disposent du masque de bit de droit 640 (-rw-r-----), si cela n'est pas modifié par la directive `audit.log.mode` dans `rsed.envvars`. Cela signifie que le propriétaire (ID utilisateur z/OS UNIX du démon RSE) dispose des droits d'accès en lecture et en écriture et que le groupe (par défaut) du propriétaire dispose du droit d'accès en écriture. Toutes les autres tentatives d'accès sont refusées, sauf si elles sont effectuées par un superutilisateur (UID 0) ou par un utilisateur disposant des droits d'accès suffisants sur le profil `SUPERUSER.FILESYS` dans la classe de sécurité `UNIXPRIV`.

Données d'audit

Les actions suivantes sont consignées :

- Accès au système (connexion, déconnexion)
- Accès au spoule JES (soumission, affichage, mise en attente, publication, annulation et purge)
- Accès au fichier (lecture, écriture, création, suppression, modification de nom, compression, migration, rappel)
- Accès au fichier (lecture, écriture, création, suppression, modification de nom)
- Exécution de commandes TSO et z/OS UNIX

Chaque action consignée est conservée (avec une date/un horodatage) au format CSV qui peut être lu par un outil d'automatisation ou d'analyse de données. Par exemple :

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

Les statistiques de membre et de fichier sont également consignées à l'ouverture du fichier. Elles sont ajoutées à la ligne présentant l'exécution de l'action `READ` et les zones sont délimitées par `%n`. Par exemple :

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

Les attributs suivants sont consignés dans l'ordre indiqué :

- Date et heure de création (mm/jj/aaaa hh:mm)
- Date et heure de la dernière modification (mm/jj/aaaa hh:mm:ss)
- Date et heure du dernier accès (mm/jj/aaaa hh:mm:ss)
- Format de l'enregistrement (RECFM)
- Indicateur de révision SCLM (N = le numéro de révision est défini, D = le numéro de révision n'est pas défini)
- Numéro de révision SCLM
- Caractères "hexadécimaux incorrects" inclus (Y = oui, N = non)

Remarque : Les caractères “hexadécimaux incorrects” requièrent des services de mappage Developer for System z car ils ne sont pas conservés après un passage sur le client car les pages des codes sont différentes.

- Longueur d'enregistrement logique (LRECL)
- Taille de fichier
- Réservé à une utilisation ultérieure
- Réservé à une utilisation ultérieure
- ID utilisateur
- Propriétaire du verrou (mise en file d'attente) pour ce membre ou fichier
- Points de code d'hôte CR (retour chariot), LF (saut de ligne) et NL (nouvelle ligne) et leurs caractères de substitution (disponibles uniquement lors de l'utilisation d'un client Version 8.0.3 ou version ultérieure)

Sécurité JES

Developer for System z permet aux clients d'accéder au spoule JES via le moniteur de travaux JES. Le serveur fournit un accès de base limité qui peut être étendu à l'aide des fonctions de protection du fichier spoule standard de votre produit de sécurité. Des actions opérateur (Mettre en attente, Publier, Annuler et Purger) sont effectuées sur les fichiers spoule via la console EMCS ; elles nécessitent des autorisations conditionnelles.

Actions sur les travaux - Limitations sur les cibles

Le moniteur de travaux ne fournit pas aux utilisateurs de Developer for System z un accès opérateur intégral au spoule JES. Seules les commandes Mettre en attente, Publier, Annuler et Purger sont disponibles, et par défaut, uniquement pour les fichiers spoule dont l'utilisateur est le propriétaire. Les commandes sont exécutées par la sélection de l'option appropriée dans la structure de menu du client (il n'y a pas d'invite de commande). La portée des commandes peut être réduite à l'aide des profils de sécurité afin de définir les travaux pour lesquels les commandes sont disponibles.

Comparable à l'action SDSF **SJ** SDSF, le moniteur de travaux JES prend en charge la commande Afficher JCL pour extraire le code JCL qui a créé la sortie de travaux sélectionnée et l'afficher dans une fenêtre d'éditeur. Le moniteur de travaux JES extrait le code JCL de JES ce qui est utile dans les situations où le membre JCL d'origine n'est pas facilement localisé.

Tableau 1. Commandes de la console du moniteur de travaux JES

Action	JES2	JES3
Mettre en attente	\$Hx(jobid) avec x = {J, S ou T}	*F,J=jobid,H
Libérer	\$Ax(jobid) avec x = {J, S ou T}	*F,J=jobid,R
Annuler	\$Cx(jobid) avec x = {J, S ou T}	*F,J=jobid,C
Purger	\$Cx(jobid),P avec x = {J, S ou T}	*F,J=jobid,C
Afficher JCL	non applicable	non applicable

Les commandes JES disponibles répertoriées dans le tableau 1, à la page 26 sont, par défaut, limitées aux travaux dont l'utilisateur est le propriétaire. Ce paramétrage peut être modifié à l'aide de la directive `LIMIT_COMMANDS`, comme indiqué à la section "FEJJCNFG, Fichier de configuration Moniteur de travaux JES" du *Guide de configuration de l'hôte* (SC11-6285).

Tableau 2. Matrice des droits d'accès des commandes `LIMIT_COMMANDS`

<code>LIMIT_COMMANDS</code>	Propriétaire du travail	
	Utilisateur	Autre
<code>USERID</code> (valeur par défaut)	Autorisé	Non autorisé
<code>LIMITED</code>	Autorisé	Autorisé uniquement si permis de manière explicite par les profils de sécurité
<code>NOLIMIT</code>	Autorisé	Autorisé si les profils de sécurité l'acceptent ou lorsque la classe JESSPOOL n'est pas active

JES utilise la classe JESSPOOL pour protéger les fichiers SYSIN/SYSOUT. Comme SDSF, le moniteur de travaux JES étend l'utilisation de la classe JESSPOOL pour protéger également les ressources des travaux.

Si `LIMIT_COMMANDS` n'est pas `USERID`, le moniteur de travaux JES demandera le droit d'accès au profil associé dans de la classe JESSPOOL, comme indiqué dans le tableau suivant :

Tableau 3. Profils JESSPOOL étendus

Commande	Profil JESSPOOL	Droit d'accès requis
Mettre en attente	nodeid.userid.jobname.jobid	ALTER
Libérer	nodeid.userid.jobname.jobid	ALTER
Annuler	nodeid.userid.jobname.jobid	ALTER
Purger	nodeid.userid.jobname.jobid	ALTER
Afficher JCL	nodeid.userid.jobname.jobid.JCL	READ

Utilisez les substitutions suivantes dans le tableau précédent :

nodeid	ID du noeud NJE du sous-système JES cible
userid	ID utilisateur local du propriétaire du travail
jobname	Nom du travail
jobid	ID du travail JES

Si la classe JESSPOOL n'est pas active, le comportement est différent pour les valeurs `LIMITED` et `NOLIMIT` de `LIMIT_COMMANDS`, comme décrit à la section "Tableau des autorisations pour la commande `LIMIT_COMMANDS`" dans "Fichier de configuration FEJJCNFG, moniteur de travaux JES" du *Guide de configuration de l'hôte* (SC11-6285). Le comportement est identique lorsque la classe JESSPOOL est active, car, par défaut, elle refuse le droit d'accès si un profil n'est pas défini.

Actions sur les travaux - Limitations liées à l'exécution

Après la définition des cibles autorisées, la seconde phase de la sécurité des commandes du spoule JES comprend la définition des autorisations nécessaires pour exécuter la commande de l'opérateur. Ce droit d'exécution est appliqué par les contrôles de sécurité z/OS et JES.

Notez que la commande Afficher JCL n'est pas une commande de l'opérateur comme une autre (par exemple, Mettre en attente, Libérer, Annuler et Purger). En conséquence, les limitations ci-dessous ne s'appliquent pas car il n'y a pas d'autre contrôle de sécurité.

Le moniteur de travaux JES émet toutes les commandes d'opérateur JES demandées par un utilisateur via une console EMCS dont le nom est contrôlé à l'aide de la directive `CONSOLE_NAME`, comme indiqué à la section "FEJJCNFG, Fichier de configuration Moniteur de travaux JES" du *Guide de configuration de l'hôte* (SC11-6285).

Le moniteur de travaux JES permet de définir les droits accordés à la console EMCS avec la directive `LIMIT_CONSOLE`, comme cela est décrit dans la section "FEJJCNFG, fichier de configuration du moniteur de travaux JES" du document *Guide de configuration de l'hôte* (SC11-6285).

Tableau 4. Matrice des droits de la console `LIMIT_CONSOLE`

<code>LIMIT_CONSOLE</code>	Profil actif dans la classe <code>OPERCMDS</code>	Aucun profil actif dans la classe <code>OPERCMDS</code>
LIMITED (valeur par défaut)	Autorisé, si cela est admis par le profil de sécurité	Non autorisé
NOLIMIT	Autorisé, si cela est admis par le profil de sécurité	Autorisé

Cette configuration permet à l'administrateur de sécurité de définir des droits d'exécution des commandes complexes en utilisant les classes `OPERCMDS` et `CONSOLE`.

- Pour utiliser une console EMCS, un utilisateur doit disposer au moins de droits `READ` dans le profil `MVS.MCSOPER.nom-console` de la classe `OPERCMDS`. Si aucun profil n'est défini, le système accorde la demande de droit.
- Pour exécuter une commande de l'opérateur JES, un utilisateur doit disposer des droits suffisants dans le profil `JES%.**` (ou dans un profil plus restrictif) de la classe `OPERCMDS`. Si aucun profil n'est défini ou que la classe `OPERCMDS` n'est pas active, JES ne parvient pas à exécuter la commande si `LIMIT_CONSOLE=LIMITED` est défini dans `FEJJCNFG`.
- L'administrateur de sécurité peut également demander qu'un utilisateur fasse appel au moniteur de travaux JES lors de l'exécution de la commande de l'opérateur en indiquant `WHEN(CONSOLE(JMON))` dans la définition `PERMIT`. La classe `CONSOLE` doit être active pour permettre le bon fonctionnement de cette configuration. Notez que la classe `CONSOLE` active est suffisante. Aucun profil n'est contrôlé pour les consoles EMCS.

Supposons que l'accès à l'identité du serveur du moniteur de travaux JES lors de la création d'une console JMON à partir d'une session TSO est empêché par votre logiciel de sécurité. Même si la console peut être créée, le point d'entrée est différent (moniteur de travaux JES/TSO). Les commandes JES exécutées par cette console échouent lors du contrôle de sécurité si la sécurité est configurée comme

indiqué dans cette publication et que l'utilisateur ne dispose pas de droits d'accès aux commandes JES via d'autres procédures.

Notez que le moniteur de travaux JES ne peut pas créer la console JMON lorsqu'une commande doit être exécutée si le nom de la console est déjà utilisé. Pour éviter cela, le programmeur système peut définir la directive `GEN_CONSOLE_NAME=ON` dans le fichier de configuration du moniteur de travaux JES ou l'administrateur de sécurité peut définir des profils de sécurité pour empêcher les utilisateurs TSO de créer une console. Les exemples de commandes RACF suivants empêchent tous les utilisateurs (sauf ceux qui sont autorisés) de créer une console TSO ou SDSF :

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

Remarque : Sans autorisation pour ces commandes d'opérateur, les utilisateurs peuvent toujours soumettre des travaux et lire les sorties de travaux via le moniteur de travaux JES s'ils disposent de droits d'accès suffisants à des profils qui protègent ces ressources (comme celles des classes JESINPUT, JESJOBS et JESSPOOL).

Pour plus d'informations sur la protection des commandes d'opérateur, voir *Security Server RACF Security Administrator's Guide (SA22-7683)*.

Accès aux fichiers spoule

Le moniteur de travaux JES permet, par défaut, de parcourir tous les fichiers spoule. Ce paramétrage peut être modifié à l'aide de la directive `LIMIT_VIEW`, comme indiqué à la section "FEJJCNF, Fichier de configuration Moniteur de travaux JES" du *Guide de configuration de l'hôte (SC11-6285)*.

Tableau 5. Matrice des droits d'accès de consultation `LIMIT_VIEW`

LIMIT_VIEW	Propriétaire du travail	
	Utilisateur	Autre
USERID	Autorisé	Non autorisé
NOLIMIT (valeur par défaut)	Autorisé	Autorisé si les profils de sécurité l'acceptent ou lorsque la classe JESSPOOL n'est pas active

Pour limiter l'accès des utilisateurs à leurs propres travaux dans le spoule JES, définissez l'instruction "`LIMIT_VIEW=USERID`" dans le fichier de configuration du moniteur de travaux JES, FEJJCNF. Si les utilisateurs souhaitent accéder à davantage de travaux, mais pas à tous, utilisez les fonctions de protection du fichier spoule standard de votre produit de sécurité (la classe JESSPOOL, par exemple).

Quand vous définissez d'autres protections, notez que le moniteur de travaux fait appel à l'interface SAPI (SYSOUT application program interface) pour accéder au spoule. En conséquence l'utilisateur a besoin, au minimum, de droits d'accès de mise à jour (UPDATE) des fichiers spoule, même pour des fonctionnalités de navigation. Cette exigence ne s'applique pas si vous utilisez z/OS version 1.7 (z/OS 1.8 pour JES3) ou une version ultérieure. Dans ce cas, les droits d'accès en lecture (READ) suffisent pour les fonctionnalités de navigation.

Pour plus d'informations sur la protection du fichier spoule JES, voir *Security Server RACF Security Administrator's Guide* (SA22-7683).

Communication chiffrée via SSL/TLS

Les communications externes (client-hôte) utilisant RSE peuvent être chiffrées à l'aide de SSL (Secure Socket Layer) ou Transport Layer Security (TLS). Cette fonction est désactivée par défaut et est contrôlée par les paramètres du fichier `ssl.properties`. Reportez-vous à la section "(Facultatif) `ssl.properties`, chiffrement RSE SSL" du *Guide de configuration de l'hôte* (SC11-6285).

Le démon RSE et le serveur RSE prennent en charge des mécanismes différents pour stocker des certificats en raison leurs différences architecturales. Cela signifie que des définitions et des certificats SSL sont nécessaires pour le démon et le serveur RSE. Un certificat partagé peut être utilisé si le démon et le serveur RSE utilisent la même méthode de gestion des certificats.

Tableau 6. Mécanismes de stockage des certificats SSL

Stockage des certificats	Créé et géré par	Démon RSE	Serveur RSE
Fichier de clés	Produit de sécurité compatible avec SAF	pris en charge	pris en charge
Base de données de clés	gskkyman de z/OS UNIX	pris en charge	/
Magasin de clés	Outil de clé de Java	/	pris en charge

Remarque : Il est conseillé d'utiliser des fichiers de clés conformes à SAF pour la gestion des certificats.

Les fichiers de clés conformes à SAF peuvent stocker la clé privée du certificat dans la base de données de sécurité ou en utilisant ICSF (Integrated Cryptographic Service Facility), l'interface vers le matériel de chiffrement de System z.

Il est recommandé d'utiliser ICSF pour le stockage des clés privées associées à des certificats numériques. En effet, il s'agit d'une solution plus sûre que la gestion de clé privée non ICSF. ICSF assure le chiffrement des clés privées dans la clé maîtresse ICSF, leur accès étant contrôlé par les ressources générales dans les classes de sécurité CSFKEYS et CSFSERV. De plus, les performances opérationnelles sont améliorées, car ICSF utilise un processeur cryptographique. Pour plus d'informations sur ICSF et pour savoir comment contrôler qui peut utiliser les clés et les services de chiffrement, voir *Cryptographic Services ICSF Administrator's Guide* (SA22-7521).

Le démon RSE utilise les fonctions SSL système pour gérer des communications chiffrées SSL. Cela signifie que SYS1.SIEALNKE doit être contrôlé par programme via le logiciel de sécurité et être à la disposition de RSE via LINKLIST ou la directive STEPLIB dans `rsed.envvars`.

L'ID utilisateur RSE (`stcrse` dans les exemples de commande ci-dessous) doit disposer des droits nécessaires pour accéder à son fichier de clés et aux certificats associés lorsque des fichiers de clés conformes à SAF sont utilisés pour le démon ou le serveur 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 directive `_RSE_JAVA_OPTS` du fichier `rased.envvars` vous permet de choisir entre SSL et son successeur TLS pour la méthode de chiffrement, comme indiqué à la section sur la définition de paramètres de démarrage Java supplémentaires avec `_RSE_JAVA_OPTS` dans le document *Guide de configuration de l'hôte* (SC23-7658).

Pour plus d'informations sur l'activation de SSL pour Developer for System z, voir Chapitre 13, «Configuration de l'authentification SSL et X.509», à la page 191.

Communication chiffrée pour le débogueur intégré

Les communications (client-hôte) externes avec le gestionnaire de débogage facultatif peuvent être chiffrées à l'aide de SSL ou TLS. Pour effectuer un tel chiffrement, créez une règle AT-TLS pour le port utilisé par le gestionnaire de débogage, par défaut 5335. Un exemple de règle est fourni dans figure 9. Voir le document *Communications Server IP Configuration Guide* (SC31-8775) pour plus de détails sur la configuration de AT-TLS (Application Transparent TLS).

```
TTLRule                                RDz_Debug_Manager
{
  LocalPortRange                        5335
  Direction                            Inbound
  TLSGroupActionRef                     grp_Production
  TLSEnvironmentActionRef               RDz_Debug_Manager
}
TLSEnvironmentAction                   RDz_Debug_Manager
{
  HandshakeRole Server
  TLSKeyRingParms
  {
    Keyring dbgmgr.racf                # Keyring must be owned by the Debug Manager
  }
}
TLSGroupAction                         grp_Production
{
  TLSEnabled                           On
  Trace                                2
}
```

Figure 9. Règle AT-TLS pour le gestionnaire de débogage

Authentification du client à l'aide de certificats X.509

Le démon RSE prend en charge les utilisateurs qui s'authentifient eux-mêmes à l'aide d'un certificat X.509. L'utilisation de communications chiffrées SSL est indispensable pour cette fonction car il s'agit d'une extension de l'authentification hôte avec un certificat utilisé dans SSL.

Le démon RSE lance la procédure d'authentification client en validant le certificat client. Les principaux éléments vérifiés sont les dates de validité du certificat et le niveau de confiance de l'autorité de certification utilisée pour signer le certificat. Une liste de révocation de certificat (CRL) d'un tiers peut également être consultée.

Une fois que le démon RSE valide le certificat, celui-ci est traité pour l'authentification. Le certificat est transmis au produit de sécurité à des fins

d'authentification, sauf si la directive `enable.certificate.mapping` de `rsed.envvars` correspond à `false`. Dans ce cas, le démon RSE effectue l'authentification.

Si elle aboutit, la procédure d'authentification détermine l'ID utilisateur à utiliser pour cette session et le soumet au test du démon RSE pour vérifier qu'il est utilisable sur le système hôte où le démon RSE s'exécute.

La dernière vérification (réalisée pour chaque mécanisme d'authentification, et pas simplement pour les certificats X.509) s'assure que l'ID utilisateur est autorisé à utiliser Developer for System z.

Si vous êtes familier des classifications de sécurité SSL utilisées par TCP/IP, la combinaison de ces procédures de validation correspond aux "spécification de niveau 3 du client" (la plus élevée).

Validation de l'autorité de certification (CA)

Une partie de la procédure de validation du certificat consiste à vérifier que le certificat a été signé par une autorité de certification habilitée. Pour effectuer cette opération, le démon RSE doit avoir accès à un certificat qui identifie l'autorité de certification.

Si vous utilisez la base de données de clés **gskkyman** pour la connexion SSL, le certificat de l'autorité de certification doit être ajouté à la base de données de clés.

Si vous utilisez un fichier de clés SAF (méthode recommandée), vous devez ajouter le certificat de l'autorité de certification à votre base de données de sécurité sous la forme d'un certificat CERTAUTH associé à l'attribut TRUST ou HIGHTRUST, comme indiqué dans l'exemple de commande RACF ci-après :

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

La plupart des produits de sécurité possèdent des certificats d'autorité de certification reconnues dans leurs bases de données avec un état NOTRUST. Utilisez les exemples de commande RACF suivantes pour répertorier les certificats des autorités de certification et marquer un certificat comme sécurisé (trusted) en fonction du libellé qui lui est affecté.

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

Remarque : L'état HIGHTRUST est indispensable si vous vous appuyez sur RACF pour authentifier l'utilisateur en fonction de l'extension HostIdMappings dans le certificat. Pour plus d'informations, voir «Authentification par votre logiciel de sécurité», à la page 33.

Une fois que le certificat de l'autorité de certification est ajouté à la base de données de sécurité, il doit être connecté au fichier de clés RSE, comme indiqué dans l'exemple de commande RACF ci-après :

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

Pour obtenir des informations détaillées sur la commande **RACDCERT**, voir le document *Security Server RACF Command Language Reference* (SA22-7687).

Attention : Si vous faites appel au démon RSE au lieu du logiciel de sécurité pour authentifier un utilisateur, veillez à ne pas mélanger les autorités de certification avec un état TRUST et HIGHTRUST dans le fichier de clés SAF ou une base de données **gskkyman**. Le démon RSE n'est pas en mesure d'établir une distinction entre les deux. Les certificats signés par une autorité de certification avec l'état TRUST est valide pour une authentification de l'ID utilisateur.

(Facultatif) Interrogation d'une liste de révocation de certificat (CRL)

Si vous le souhaitez, vous pouvez demander au démon RSE de vérifier une ou plusieurs listes de révocation de certificat (CRL) pour renforcer la sécurité de la procédure de validation. Cette opération est effectuée en ajoutant des variables d'environnement liées aux listes de révocation de certificat au fichier `rsed.envvars`.

- GSK_CRL_SECURITY_LEVEL
- GSK_LDAP_SERVER
- GSK_LDAP_PORT
- GSK_LDAP_USER
- GSK_LDAP_PASSWORD

Pour plus d'informations sur ces variables d'environnement et sur d'autres variables d'environnement utilisées par z/OS System SSL, voir *Cryptographic Services System Secure Sockets Layer Programming* (SC24-5901).

Remarque : Soyez prudent lorsque vous définissez d'autres variables d'environnement z/OS System SSL (`GSK_*`) dans `rsed.envvars`, car elles risquent de modifier la manière dont le démon RSE traite les connexions SSL et l'authentification par certificat.

Authentification par votre logiciel de sécurité

RACF effectue plusieurs vérifications pour authentifier un certificat et renvoyer l'ID utilisateur associé. Notez toutefois que d'autres produits de sécurité peuvent effectuer cette opération différemment. Pour plus d'informations sur la fonction `iniTACEE` utilisée pour effectuer l'authentification (mode requête), reportez-vous à la documentation du produit de sécurité.

1. RACF vérifie si le certificat est défini dans la classe DIGTCERT. Si c'est le cas, RACF renvoie l'ID utilisateur associé à ce certificat lorsque celui-ci a été ajouté à la base de données RACF.

Les certificats sont définis dans RACF à l'aide de la commande `RACDCERT`, comme indiqué dans l'exemple suivant :

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

2. Si le certificat n'est pas défini, RACF vérifie s'il n'y a pas de filtre de nom de certificat concordant défini dans les classes DIGTNMAP ou DIGTCRIT. Si tel est le cas, il renvoie l'ID utilisateur associé au filtre dont la concordance est la plus proche.

Remarque : Il est déconseillé d'utiliser des filtres de nom pour les certificats utilisés par Developer for System z, car ces filtres rattachent tous les certificats à un même ID utilisateur. Cela signifie que tous les utilisateurs Developer for System z se connectent avec le même ID utilisateur.

3. En l'absence de filtre de nom concordant, RACF recherche l'extension de certificat `HostIdMappings` et extrait la paire ID utilisateur et nom d'hôte

imbriquée. Si l'extension est détectée et validée, RACF renvoie l'ID utilisateur défini au sein de l'extension HostIdMappings.

La paire ID utilisateur et nom d'hôte est valide si toutes ces conditions sont remplies :

- Le certificat de l'autorité de certification utilisé pour signer ce certificat est marqué comme HIGHTRUST dans la classe DIGTCERT.
- La longueur de l'ID utilisateur stocké dans l'extension est valide (de 1 à 8 caractères).
- L'ID utilisateur affecté au démon RSE dispose (au minimum) de droits READ dans le profil IRR.HOST.nomhôte de la classe SERVAUTH, où nomhôte est le nom d'hôte stocké dans l'extension. Il s'agit généralement d'un nom de domaine, par exemple CDFMVS08.RALEIGH.IBM.COM.

La définition de l'extension HostIdMappings dans la syntaxe ASN.1 est :

```
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
}
```

Remarque : L'extension HostIdMappings n'est pas prise en compte si l'ID utilisateur cible a été créé après la date de début de validité du certificat contenant l'extension HostIdMappings. Si vous créez des ID utilisateur spécialement pour des certificats avec des extensions HostIdMappings, vérifiez qu'ils sont créés avant la soumission des demandes de certificat.

Pour plus d'informations sur les certificats X.509, sur leur gestion par RACF et sur la définition de filtres de nom de certificat, voir le document *Security Server RACF Security Administrator's Guide* (SA22-7683). Pour obtenir des informations détaillées sur la commande **RACDCERT**, voir le document *Security Server RACF Command Language Reference* (SA22-7687).

Authentification effectuée par le démon RSE

Developer for System z peut effectuer une authentification de base des certificats X.509 sans faire appel à votre produit de sécurité. L'authentification effectuée par le démon RSE requiert la définition d'un ID utilisateur et d'un nom d'hôte dans une extension de certificat et est activée uniquement si la directive `enable.certificate.mapping` définie dans le fichier `rsed.envvars` correspond à `FALSE`.

Cette fonction doit être utilisée si votre produit de sécurité ne prend pas en charge l'authentification d'un utilisateur via un certificat X.509 ou si un certificat échoue aux tests effectués par le produit de sécurité (par exemple, le certificat possède un identificateur erroné pour l'extension HostIdMappings ou il n'y a pas de filtre ou de définition de nom dans DIGTCERT).

Le client demande à l'utilisateur l'identificateur d'extension (OID) à utiliser. Par défaut, il s'agit de l'OID HostIdMappings, {1 3 18 0 2 18 1}.

Le démon RSE doit extraire l'ID utilisateur et le nom d'hôte en utilisant l'extension de format HostIdMappings. Ce format est décrit à la section «Authentification par votre logiciel de sécurité», à la page 33.

La paire ID utilisateur et nom d'hôte est valide si toutes ces conditions sont remplies :

- La longueur de l'ID utilisateur stocké dans l'extension est valide (de 1 à 8 caractères).
- L'ID utilisateur affecté au démon RSE dispose (au minimum) de droits READ dans le profil IRR.HOST.nomhôte de la classe SERVAUTH, où nomhôte est le nom d'hôte stocké dans l'extension. Il s'agit généralement d'un nom de domaine, par exemple CDFMVS08.RALEIGH.IBM.COM.

Avertissement : Il revient à l'administrateur de sécurité de vérifier que toutes les autorités de certification connues du démon RSE sont parfaitement dignes de confiance car le démon RSE ne peut pas vérifier si l'autorité qui a signé le certificat client est parfaitement digne de confiance (highly trusted) ou simplement digne de confiance (trusted). Pour plus d'informations sur les certificats d'autorités de certification accessibles, voir «Validation de l'autorité de certification (CA)», à la page 32.

Vérification du port d'entrée (POE)

Developer for System z prend en charge la vérification du port d'entrée, ce qui permet à l'hôte d'accéder uniquement aux adresses TCP/IP sécurisées. Cette fonction est désactivée par défaut et requiert la définition du profil de sécurité BPX.POE, comme le montrent les exemples de commandes RACF :

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

Remarque :

- RSE doit être configuré pour utiliser POE. Pour cela, il est nécessaire de supprimer la mise en commentaire de l'option "enable.port.of.entry=true" dans `rsed.envvars`, comme indiqué à la section "Définition de paramètres de démarrage supplémentaires Java avec `_RSE_JAVAOPTS`" dans *Guide de configuration de l'hôte* (SC11-6285).
- L'ID utilisateur RSE STCRSE requiert UID(0) lorsque ce profil n'est pas défini et exige que la vérification du port d'entrée soit activée dans `rsed.envvars`.
- La définition de BPX.POE a un impact sur d'autres applications TC/PIP prenant en charge la vérification du port d'entrée (INETD, par exemple).
- Des zones de sécurité (profils EZB.NETACCESS.**, qui sont des zones d'adresses IP) doivent être définies dans la classe SERVAUTH pour bénéficier pleinement de la vérification du port d'entrée.

Pour plus d'informations sur le contrôle d'accès au réseau via la vérification du port d'entrée, voir *Communications Server IP Configuration Guide* (SC31-8775).

Modification des fonctions client

Les clients Developer for System z, version 8.5.1 et supérieure, ont la possibilité de vérifier l'autorisation d'accès aux profils de sécurité SAF, puis en fonction des résultats, peuvent activer ou désactiver la fonction correspondante de l'utilisateur.

Developer for System z vérifie les droits d'accès aux profils répertoriés dans le tableau 7 afin de déterminer quelles options doivent être activées ou désactivées pour l'utilisateur.

Tableau 7. Informations SAF en vue de la modification des fonctions client

Profil FACILITY	Longueur fixe	Droit d'accès requis	Résultat
FEK.USR.OFF.REMOTECOPY.MVS.sysname	27	READ	Le client désactive la fonction de copie et les fonctions connexes dans les fichiers MVS.

Remarque : Developer for System z suppose qu'un utilisateur ne dispose d'aucun droit d'accès lorsque votre logiciel de sécurité indique qu'il ne peut pas déterminer si un utilisateur dispose ou non des droits d'accès à un profil. Cela se produit par exemple lorsque le profil n'est pas défini.

La valeur sysname correspond au nom du système cible.

La colonne "Longueur fixe" indique la longueur de la partie fixe du profil de sécurité associée.

Par défaut, Developer for System z s'attend à ce que les profils FEK.* résident dans la classe de sécurité FACILITY. Notez que les profils figurant dans la classe FACILITY sont limités à 39 caractères. Si la somme de la longueur de la partie fixe du profil (FEK.USR.<key>) et de la longueur de la partie de profil spécifique au site (sysname) est supérieure à ce nombre, vous pouvez placer les profils dans une autre classe et indiquer à Developer for System z qu'il doit utiliser cette dernière à la place. Pour ce faire, mettez en commentaires _RSE_FEK_SAF_CLASS dans rsed.envvars et indiquez le nom de classe de votre choix.

Les exemples de définition de sécurité suivants autorisent l'action REMOTECOPY.MVS sur tous les utilisateurs de CDFMVS08, hormis ceux qui appartiennent au groupe 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

Lorsque les utilisateurs disposent d'un accès en lecture READ au profil FEK.USR.OFF.REMOTECOPY.MVS.sysname, leurs clients Developer for System z version 8.5.1 et supérieure, désactivent les actions glisser, copier, enregistrer sous et travailler hors ligne sur les fichiers MVS. Cela signifie que les utilisateurs peuvent accéder aux fichiers du système, mais qu'ils ne peuvent pas créer de copie de fichier en local sur leur poste de travail. Cela permet d'éviter la fuite de données confidentielles en cas de perte ou de vol du poste de travail en local.

Groupes de développeurs de la fonction d'envoi au client

Les clients Developer for System z version 8.0.1 et suivante peuvent extraire les fichiers de configuration client et les informations de mise à niveau depuis l'hôte lorsqu'ils se connectent, ce qui permet de garantir que tous les clients sont paramétrés de la même façon et qu'ils sont à jour.

Depuis la version 8.0.3, l'administrateur client peut créer plusieurs jeux de configuration client et plusieurs scénarios de mise à jour client afin de répondre aux besoins des différents groupes de développeurs. Cela permet aux utilisateurs de recevoir une configuration personnalisée, basée sur des critères tels que l'appartenance d'un groupe LDAP ou les droits d'accès à un profil de sécurité.

Lorsque vous utilisez des définitions dans votre base de données de sécurité comme mécanisme de sélection (la valeur SAF est spécifiée pour les directives dans `pushtoclient.properties`), Developer for System z vérifie les droits d'accès aux profils répertoriés dans le tableau 8 pour identifier les groupes de développeurs auxquels l'utilisateur appartient et déterminer si un utilisateur est autorisé à rejeter les mises à jour.

Tableau 8. Informations SAF pour la fonction d'envoi au client

Profil FACILITY	Longueur fixe	Droit d'accès requis	Résultat
FEK.PTC.CONFIG.ENABLED. sysname.devgroup	23	READ	Le client accepte les mises à jour de configuration pour le groupe indiqué
FEK.PTC.PRODUCT. ENABLED.sysname.devgroup	24	READ	Le client accepte les mises à jour de produit pour le groupe indiqué
FEK.PTC.REJECT.CONFIG. UPDATES.sysname	30	READ	L'utilisateur peut rejeter les mises à jour de configuration
FEK.PTC.REJECT.PRODUCT. UPDATES.sysname	31	READ	L'utilisateur peut rejeter les mises à jour de produit

Remarque : Developer for System z suppose qu'un utilisateur ne dispose d'aucun droit d'accès lorsque votre logiciel de sécurité indique qu'il ne peut pas déterminer si un utilisateur dispose ou non des droits d'accès à un profil. Cela se produit par exemple lorsque le profil n'est pas défini.

La valeur `devgroup` correspond au nom de groupe affecté à un groupe spécifique de développeurs. Notez que le nom de groupe est visible sur les clients Developer for System z.

La valeur `sysname` correspond au nom du système cible.

La colonne "Longueur fixe" indique la longueur de la partie fixe du profil de sécurité associée.

Par défaut, Developer for System z s'attend à ce que les profils FEK.* résident dans la classe de sécurité FACILITY. Notez que les profils figurant dans la classe

FACILITY sont limités à 39 caractères. Si la somme de la longueur de la partie fixe du profil (FEK.PTC.<key>) et de la longueur de la partie de profil spécifique au site (sysname ou sysname.devgroup) est supérieure à ce nombre, vous pouvez placer les profils dans une autre classe et indiquer à Developer for System z qu'il doit utiliser cette dernière à la place. Pour ce faire, mettez en commentaires _RSE_FEK_SAF_CLASS dans rsed.envvars et indiquez le nom de classe de votre choix.

Notez que l'administrateur client doit figurer sur la liste d'accès des profils FEK.PTC.*.ENABLED.* pour pouvoir définir et gérer les métadonnées d'envoi au client associées. Cela implique que les profils soient définis avec (au moins) l'administrateur client dans la liste d'accès avant l'implémentation de la fonction d'envoi au client avec le support de groupe.

Pour plus d'informations sur l'activation du support de plusieurs groupes, voir la rubrique "(Facultatif) pushtoclient.properties, contrôle client résidant sur l'hôte" dans le document *Guide de configuration de l'hôte* (SC11-6285). Pour plus d'informations sur les concepts et l'implémentation de la fonction d'envoi au client, voir Chapitre 7, «Remarques relatives à la fonction d'envoi au client», à la page 125.

Sécurité du débogage

Le débogueur intégré facultatif peut déboguer des transactions CICS chargées dans la mémoire morte. Lorsqu'il est utilisé pour déboguer une transaction à l'état de problème (non autorisé), le débogueur intégré vérifie que l'utilisateur propriétaire de la session de débogage est autorisé à effectuer une telle opération.

Developer for System z vérifie les droits d'accès aux profils répertoriés dans tableau 9 afin de déterminer les autorisations de débogage accordées.

Tableau 9. Informations SAF pour les fonctions de débogage

Profil FACILITY	Droit d'accès requis	Résultat
AQE.AUTHDEBUG.WRITEBUFFER	UPDATE	L'utilisateur peut déboguer des transactions CICS en lecture seule.

Remarque : Developer for System z suppose qu'un utilisateur ne dispose d'aucun droit d'accès lorsque votre logiciel de sécurité indique qu'il ne peut pas déterminer si un utilisateur dispose ou non des droits d'accès à un profil. Cela se produit par exemple lorsque le profil n'est pas défini.

Les exemples de définition de sécurité suivants autorisent l'action AUTHDEBUG.WRITEBUFFER pour tous les utilisateurs du groupe RDZDEBUG :

```
RDEFINE FACILITY (AQE.AUTHDEBUG.WRITEBUFFER) -
  UACC(NONE) DATA('RATIONAL DEVELOPER FOR SYSTEM Z - DEBUG CICS READ-ONLY')
PERMIT AQE.AUTHDEBUG.WRITEBUFFER CLASS(FACILITY) -
  ID(RDZDEBUG) ACCESS(UPDATE)
SETROPTS RACLIST(FACILITY) REFRESH
```

CICSTS, sécurité

Le débogueur intégré facultatif peut déboguer des transactions CICS. Pour plus d'informations, voir «Débogage de transaction CICS», à la page 154.

Developer for System z permet, via le gestionnaire de déploiement d'application, aux administrateurs CICS de contrôler les définitions de ressource CICS qui peuvent être modifiées par le développeur, leurs valeurs par défaut ainsi que l'affichage d'une définition de ressource CICS à l'aide du serveur de définition de ressource CICS. Pour plus d'informations sur les définitions de sécurité TS CICS, voir Chapitre 8, «Remarques relatives à CICSTS», à la page 143.

Référentiel de CRD

Le fichier VSAM du référentiel du serveur CRD contient toutes les définitions de ressource par défaut ; il doit par conséquent être protégé contre les mises à jour tout en autorisant les développeurs à consulter les valeurs qui y sont conservées.

Transactions CICS

Developer for System z met à disposition plusieurs transactions qui sont utilisées par le serveur CRD lors de la définition et de la consultation des ressources CICS. Quand la transaction est rattachée, la vérification de la sécurité de la ressource CICS, si elle est activée, garantit que l'ID utilisateur est autorisé à exécuter l'ID de transaction.

Communication chiffrée via SSL

Le client du gestionnaire de déploiement d'application client utilise les services Web CICS ou l'interface RESTful pour appeler le serveur CRD. L'utilisation de SSL pour cette communication est contrôlée par la définition TCPIPService CICS TS, comme indiqué dans la documentation *RACF Security Guide for CICS TS*.

Informations diverses

Mise en corbeille GATE

Lorsqu'un espace adresse demande pour la première fois à RACF d'accéder à une classe de ressources ne figurant pas dans une RACLIST (non stockée en mémoire), telle que la classe DATASET, RACF extrait et stocke tous les profils génériques associés dans l'espace adresse de l'utilisateur, dans une liste appelée GATE (Generic Anchor Table Entry). Jusqu'à z/OS 1.12, RACF gère quatre ancres génériques pour chaque espace adresse et quatre pour chaque bloc de contrôle des tâches MVS disposant de son propre élément ACEE. Lorsque les quatre sont utilisés, RACF remplace celui qui est le moins récemment référencé lorsqu'il en arrive un nouveau.

Si vos utilisateurs accèdent souvent à plus de quatre qualificatifs de haut niveau de fichier, les pools d'unités d'exécution RSE (gérant plusieurs utilisateurs à l'aide d'unités d'exécution avec des éléments ACEE propres à l'utilisateur) peuvent faire l'expérience d'une mise en corbeille GATE car RACF doit faire tourner les nouvelles entrées parmi les emplacements d'ancre disponibles.

Dans z/OS 1.12, RACF a ajouté l'option **GENERICANCHOR** de la commande **SET** pour vous permettre d'augmenter la taille de la table. Cette définition peut s'effectuer au niveau du système ou pour chaque nom de travail.

Élément ACEE géré

Developer for System z utilise les services de noyau z/OS UNIX, tels que `pthread_security_np()` et `__passwd()`, lesquels utilisent le service de sécurité `InitACEE`, ce qui a pour conséquence de générer des blocs de contrôle de sécurité ACEE gérés. Un élément ACEE (Accessor Environment Element) géré est mis en

cache par votre produit de sécurité, et ce dernier va ignorer certaines modifications, telles que les modifications de mot de passe en dehors de Developer for System z jusqu'au dépassement délai d'attente du cache. (Le dépassement du délai d'attente peut prendre quelques minutes.)

Actualisez le cache ACEE géré après les modifications de sécurité pour faire en sorte que nouvelles données soient utilisées par Developer for System z.

SCLM, sécurité

SCLM Developer Toolkit offre des fonctionnalités de sécurité facultatives pour les fonctions de génération, de promotion et de déploiement.

Si l'administrateur SCLM a activé la sécurité pour une fonction, des appels SAF sont effectués afin de vérifier l'autorité qui exécute la fonction protégée avec l'ID de l'appelant ou d'un utilisateur de substitution.

Pour de plus amples informations sur les définitions de sécurité SCLM requises, voir le document *SCLM Developer Toolkit - Guide d'administration* (SC11-6464).

Fichiers de configuration Developer for System z

Plusieurs fichiers de configuration Developer for System z contiennent des directives qui ont une incidence sur la configuration des audits et de la sécurité. En fonction des informations de ce chapitre, l'administrateur de sécurité et le programmeur système peuvent déterminer les paramètres à définir pour les directives ci-dessous.

Moniteur de travaux JES - FEJJCNFG

- `LIMIT_COMMANDS={USERID | LIMITED | NOLIMIT }`
Définit les travaux auxquels les actions peuvent être appliquées (à l'exception de la consultation et la soumission). Pour plus d'informations, voir «Actions sur les travaux - Limitations sur les cibles», à la page 26.
- `LIMIT_CONSOLE={LIMITED | NOLIMIT}`
Définissez le niveau d'autorisation de la console EMCS utilisée pour les actions d'exécution. Pour plus d'informations, voir «Actions sur les travaux - Limitations sur les cibles», à la page 26.
- `LIMIT_VIEW={USERID | NOLIMIT}`
Définissez les fichiers de spoule qui peuvent être consultés. Pour plus d'informations, voir «Accès aux fichiers spoule», à la page 29.
- `LOOPBACK_ONLY={ON | OFF}`
Indiquez si le moniteur de travaux JES est accessible en dehors de ce système z/OS. Pour plus d'informations, voir la section *FEJJCNFG, fichier de configuration du moniteur de travaux JES* du chapitre sur la *personnalisation de base* dans le document *Guide de configuration de l'hôte* (SC23-7658).
- `APPLID={FEKAPPL | *}`
ID application utilisé pour la création et la validation de mots de passe PassTicket. Pour plus d'informations, voir «Utilisation de PassTickets», à la page 23.

Remarque : Des informations sur ces directives et d'autres directives FEJCNFG.properties sont disponibles dans "FEJCNFG, Fichier de configuration Moniteur de travaux JES" dans *Guide de configuration de l'hôte* (SC11-6285).

RSE - rsed.envvars

- `_RSE_FEK_SAF_CLASS={FACILITY | *}`
Classe de sécurité contenant les profils FEK.**. Pour plus d'informations, voir «Groupes de développeurs de la fonction d'envoi au client», à la page 37 et «Modification des fonctions client», à la page 35.
- `(_RSE_JAVAOPTS) -DDENY_PASSWORD_SAVE={true | false}`
Empêche les utilisateurs de sauvegarder leur mot de passe hôte sur le client. Pour plus d'informations, reportez-vous au tableau "Définition de paramètres de démarrage supplémentaires Java avec _RSE_JAVAOPTS" du *Guide de configuration de l'hôte* (SC11-6285).
- `(_RSE_JAVAOPTS) -DDSTORE_IDLE_SHUTDOWN_TIMEOUT=value`
Délai de déconnexion des clients inactifs. Pour plus d'informations, reportez-vous au tableau "Définition de paramètres de démarrage supplémentaires Java avec _RSE_JAVAOPTS" du *Guide de configuration de l'hôte* (SC11-6285).
- `(_RSE_JAVAOPTS) -DAPPLID={FEKAPPL | *}`
ID application utilisé pour la création et la validation de mots de passe PassTicket. Pour plus d'informations, voir «Utilisation de PassTickets», à la page 23.
- `(_RSE_JAVAOPTS) -Denable.port.of.entry={true | false}`
Active la vérification du port d'entrée. Pour plus d'informations, voir «Vérification du port d'entrée (POE)», à la page 35.
- `(_RSE_JAVAOPTS) -DDSTORE_SSL_ALGORITHM={TLSv1.2 | SSL}`
Sélectionnez SSL ou TLS comme méthode de chiffrement des communications. Pour plus d'informations, voir «Communication chiffrée via SSL/TLS», à la page 30.
- `(_RSE_JAVAOPTS) -Denable.certificate.mapping={true | false}`
Utilisez le produit de sécurité pour authentifier des utilisateurs avec un certificat X.509. Pour plus d'informations, voir «Authentification du client à l'aide de certificats X.509», à la page 31.
- `GSK_CRL_SECURITY_LEVEL={LOW | MEDIUM | HIGH}`
`GSK_LDAP_SERVER=*`
`GSK_LDAP_PORT={389 | *}`
`GSK_LDAP_USER=*`
`GSK_LDAP_PASSWORD=*`
Contrôles de sécurité supplémentaires pour l'authentification X.509. Pour plus d'informations, voir «(Facultatif) Interrogation d'une liste de révocation de certificat (CRL)», à la page 33.
- `(_RSE_JAVAOPTS) -Ddaemon.log={/var/rdz/logs | *}`
Emplacement des fichiers d'audit. Pour plus d'informations, voir «Consignation dans le journal d'audit», à la page 24.
- `(_RSE_JAVAOPTS) -Daudit.log.mode={RW.R. | * }`
Masque des droits d'accès au fichier des journaux d'audit. Pour plus d'informations, voir «Consignation dans le journal d'audit», à la page 24.
- `(_RSE_JAVAOPTS) -Daudit.action=<shell script>`
`(_RSE_JAVAOPTS) -Daudit.action.id=<userid>`

Exit utilisateur basé sur z/OS UNIX qui traite les journaux d'audit. Pour plus d'informations, voir «Consignation dans le journal d'audit», à la page 24.

Remarque : Des informations sur ces directives et d'autres directives `rsed.properties` sont disponibles dans "`rsed.envvars`, fichier de configuration RSE" dans *Guide de configuration de l'hôte* (SC11-6285).

RSE - `ssl.properties`

- `daemon_keydb_file`={nom du fichier de clés SAF | nom de la base de données de clés gskkyman}
Emplacement du certificat du démon RSE. Pour plus d'informations, voir «Communication chiffrée via SSL/TLS», à la page 30.
- `daemon_key_label`=libellé du certificat
Nom du certificat du démon RSE. Pour plus d'informations, voir «Communication chiffrée via SSL/TLS», à la page 30.
- `server_keystore_file`={nom du fichier de clés SAF | nom du magasin de clés Java}
Emplacement du certificat du serveur RSE. Pour plus d'informations, voir «Communication chiffrée via SSL/TLS», à la page 30.
- `server_keystore_label`=certificate label
Nom du certificat du serveur RSE. Pour plus d'informations, voir «Communication chiffrée via SSL/TLS», à la page 30.
- `server_keystore_type`={JKS | JCRACFKS | JCECCARACFKS}
Type de fichier de clés utilisé (Java ou SAF). Pour plus d'informations, voir «Communication chiffrée via SSL/TLS», à la page 30.

Remarque : Des informations sur ces directives et d'autres directives `ssl.properties` sont disponibles dans "(Facultatif) `ssl.properties`, chiffrement RSE SSL" dans *Guide de configuration de l'hôte* (SC11-6285).

RSE - `pushtoclient.properties`

- `config.enabled`={true | false | SAF | LDAP}
`reject.config.updates`={true | false | SAF | LDAP}
Contrôle basé sur un hôte des fichiers de configuration client de Developer for System z. Pour plus d'informations, voir Chapitre 7, «Remarques relatives à la fonction d'envoi au client», à la page 125.
- `product.enabled`={true | false | SAF | LDAP}
`reject.product.updates`={true | false | SAF | LDAP}
Contrôle basé sur un hôte des mises à jour de produit client de Developer for System z. Pour plus d'informations, voir Chapitre 7, «Remarques relatives à la fonction d'envoi au client», à la page 125.

Remarque : Pour plus d'informations sur ces sujets et sur les autres directives `pushtoclient.properties`, voir la rubrique (Facultatif) `pushtoclient.properties`, contrôle client résidant sur l'hôte" dans le document *Guide de configuration de l'hôte* (SC11-6285).

Définitions de sécurité

Personnalisez et soumettez l'exemple de membre FEKRACF, comportant les commandes RACF et z/OS UNIX permettant de créer les définitions de sécurité de base de Developer for System z.

FEKRACF se trouve dans FEK.#CUST.JCL, sauf si vous avez indiqué un autre emplacement lorsque vous avez personnalisé et soumis le travail FEK.SFEKSAMP(FEKSETUP). Pour plus d'informations, voir la section sur la configuration de la personnalisation dans le document *IBM Rational Developer for System z - Guide de configuration de l'hôte*.

Pour plus d'informations sur les commandes RACF, voir le document *RACF Command Language Reference* (SA22-7687).

Remarque :

- Pour les sites qui utilisent CA ACF2™ for z/OS, voir la page de produit du site de support CA (<https://support.ca.com>) et rechercher le document Developer for System z Knowledge, TEC492389 associé. Ce document Knowledge présente des informations détaillées sur les commandes de sécurité nécessaires pour correctement configurer Developer for System z.
- Pour les sites qui utilisent CA Top Secret for z/OS, voir la page de produit du site de support CA (<https://support.ca.com>) et rechercher le document Developer for System z Knowledge, TEC492091 associé. Ce document Knowledge présente des informations détaillées sur les commandes de sécurité nécessaires pour correctement configurer Developer for System z.

Les sections suivantes décrivent les étapes nécessaires, la configuration facultative et les autres solutions possibles.

Configuration requise et liste de contrôle

Pour effectuer la configuration de la sécurité, l'administrateur de sécurité doit connaître les valeurs indiquées dans le tableau 10. Ces valeurs ont été définies dans les étapes précédentes d'installation et de personnalisation de Developer for System z.

Tableau 10. Variables de configuration de la sécurité

Description	<ul style="list-style-type: none">• Valeur par défaut• Emplacement de la réponse	Valeur
Qualificatif de haut niveau du produit Developer for System z	<ul style="list-style-type: none">• FEK• Installation SMP/E	
Qualificatif de haut niveau de personnalisation de Developer for System z	<ul style="list-style-type: none">• FEK.#CUST• FEK.SFEKSAMP(FEKSETUP) (voir la section relative à la Configuration personnalisée dans le manuel <i>IBM Rational Developer for System z - Guide de configuration de l'hôte</i>).	

Tableau 10. Variables de configuration de la sécurité (suite)

Description	<ul style="list-style-type: none"> • Valeur par défaut • Emplacement de la réponse 	Valeur
Nom de tâche démarrée du débogueur intégré	<ul style="list-style-type: none"> • DBGMGR • FEK.#CUST.PROCLIB(DBGMGR) (voir les modifications de PROCLIB dans le manuel <i>IBM Rational Developer for System z - Guide de configuration de l'hôte</i>) 	
Nom de tâche démarrée du moniteur de travaux JES	<ul style="list-style-type: none"> • JMON • FEK.#CUST.PROCLIB(JMON), comme indiqué dans la section sur les modifications apportées à PROCLIB dans le document <i>IBM Rational Developer for System z - Guide de configuration de l'hôte</i> 	
Nom de tâche démarrée du démon RSE	<ul style="list-style-type: none"> • RSED • FEK.#CUST.PROCLIB(RSED), comme indiqué dans la section sur les modifications apportées à PROCLIB dans le document <i>IBM Rational Developer for System z - Guide de configuration de l'hôte</i> 	
ID application	<ul style="list-style-type: none"> • FEKAPPL • /etc/rdz/rsed.envvars, comme indiqué dans la section sur la définition de paramètres de démarrage Java supplémentaires avec _RSE_JVAOPTS dans le document <i>IBM Rational Developer for System z - Guide de configuration de l'hôte</i> 	

La liste ci-après présente les actions requises pour effectuer la configuration de sécurité de base de Developer for System z. Comme indiqué dans les sections ci-après, différentes méthodes peuvent répondre à vos exigences, en fonction du niveau de sécurité requis. Pour plus d'informations sur la configuration de la sécurité des services facultatifs de Developer for System z, voir les sections précédentes.

- «Activation des classes et des paramètres de sécurité», à la page 45
- «Définition d'un segment OMVS pour les utilisateurs Developer for System z», à la page 46

- «Définition des tâches démarrées de Developer for System z», à la page 46
- «Définition de RSE en tant que serveur z/OS UNIX sécurisé», à la page 48
- «Définition des bibliothèques contrôlées de programme MVS pour RSE», à la page 48
- «Définition de la prise en charge de PassTicket pour RSE», à la page 49
- «Définition de la protection des applications pour RSE», à la page 50
- «Définition de la sécurité des commandes JES», à la page 51
- «Définition des profils de fichier», à la page 53
- «Définition de fichiers contrôlés par programme z/OS UNIX pour RSE», à la page 57
- «Vérification des paramètres de sécurité», à la page 58

Activation des classes et des paramètres de sécurité

Developer for System z utilise différents mécanismes de sécurité pour fournir au client un environnement système hôte sécurisé et contrôlé. Pour ce faire, plusieurs classes et paramètres de sécurité doivent être actifs, comme indiqué par les exemples de commande RACF suivants :

- Affichage des paramètres courants
 - SETROPTS LIST
- Activation de la classe de fonction pour z/OS UNIX et les profils de certificats numériques
 - SETROPTS GENERIC(FACILITY)
 - SETROPTS CLASSACT(FACILITY) RACLIST(FACILITY)
- Activation des définitions de tâche démarrée
 - SETROPTS GENERIC(STARTED)
 - RDEFINE STARTED ** STDATA(USER(=MEMBER) GROUP(STCGROUP) TRACE(YES))
 - SETROPTS CLASSACT(STARTED) RACLIST(STARTED)
- Activation de la sécurité de la console du moniteur de travaux JES
 - SETROPTS GENERIC(CONSOLE)
 - SETROPTS CLASSACT(CONSOLE) RACLIST(CONSOLE)
- Activation de la protection des commandes de l'opérateur du moniteur de travaux JES
 - SETROPTS GENERIC(OPERCMDS)
 - SETROPTS CLASSACT(OPERCMDS) RACLIST(OPERCMDS)
- Activation de la protection des applications pour RSE
 - SETROPTS GENERIC(APPL)
 - SETROPTS CLASSACT(APPL) RACLIST(APPL)
- Activation de l'ouverture de session sécurisée pour RSE à l'aide de mots de passe PassTicket
 - SETROPTS GENERIC(PTKTDATA)
 - SETROPTS CLASSACT(PTKTDATA) RACLIST(PTKTDATA)
- Activation du contrôle de programme pour que seul le code sécurisé puisse être chargé par RSE
 - RDEFINE PROGRAM ** ADDMEM('SYS1.CMDLIB'//NOPADCHK) UACC(READ)
 - SETROPTS WHEN(PROGRAM)

Remarque : Ne créez pas le profil ** si le profil * existe déjà dans la classe PROGRAM. Cela occulterait et compliquerait le chemin de recherche utilisé par le logiciel de sécurité. Dans ce cas de figure, vous devez fusionner la définition * existante et la nouvelle définition **. Utilisez le profil **, comme indiqué dans la documentation *Security Server RACF Security Administrator's Guide* (SA22-7683).

Attention : Certains produits (FTP, par exemple) doivent être contrôlés par programme si "WHEN PROGRAM" est actif. Vous devez essayer ce contrôle de programmes avant de l'activer sur un système de production.

- (Facultatif) Activation du support X.509 HostIdMappings et du port d'entrée étendu
 - SETROPTS GENERIC(SERVAUTH)
 - SETROPTS CLASSACT(SERVAUTH) RACLIST(SERVAUTH)

Définition d'un segment OMVS pour les utilisateurs Developer for System z

Un segment OMVS RACF ou équivalent indiquant un ID utilisateur z/OS UNIX différent de zéro valide, un répertoire principal et une commande shell doivent être définis pour chaque utilisateur de Developer for System z. Leur groupe par défaut requiert également un segment OMVS avec un ID de groupe.

Lors de l'utilisation du débogueur intégré facultatif, l'ID utilisateur sous laquelle l'application déboguée est active et son groupe par défaut nécessitent également un segment OMVS RACF ou équivalent.

Dans les exemples de commandes RACF ci-dessous, remplacez les marques de réservation #userid, #user-identifier, #group-name et #group-identifier par les valeurs réelles :

- ```
ALTUSER #userid
OMVS(UID(#user-identifier) HOME(/u/#userid) PROGRAM(/bin/sh) NOASSIZEMAX)
```
- ```
ALTGROUP #group-name OMVS(GID(#group-identifier))
```

Définition des tâches démarrées de Developer for System z

Les exemples de commande RACF ci-dessous créent les tâches démarrées DBGMR, JMON et RSED avec des ID utilisateur protégés (STCDBGMR, STCJMON et STCRSE), ainsi que le groupe STCGRP qui leur est affecté. Remplacez les marques de réservation #group-id et #user-id-* par des ID OMVS valides.

- ```
ADDGROUP STCGRP OMVS(GID(#group-id))
DATA('GROUP WITH OMVS SEGMENT FOR STARTED TASKS')
```
- ```
ADDUSER STCDBGMR DFLTGRP(STCGRP) NOPASSWORD
NAME('RDZ - DEBUG MANAGER')
OMVS(UID(#user-id-debug) HOME(/tmp) PROGRAM(/bin/sh) )
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
```
- ```
ADDUSER STCJMON DFLTGRP(STCGRP) NOPASSWORD NAME('RDZ - JES JOBMONITOR')
OMVS(UID(#user-id-jmon) HOME(/tmp) PROGRAM(/bin/sh))
DATA('RATIONAL DEVELOPER FOR SYSTEM Z')
```
-

```

| ADDUSER STCRSE DFLTGRP(STCGROUP) NOPASSWORD NAME('RDZ - RSE DAEMON')
| OMVS(UID(#user-id-rse) 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

```

### Remarque :

- Assurez-vous que les ID utilisateur des tâches démarrées sont protégés en indiquant le mot clé NOPASSWORD.
- Vérifiez que le serveur RSE possède un uid OMVS unique en raison des autorisations liées à z/OS UNIX accordées à cet uid.
- Le démon RSE requiert une taille d'espace adresse importante (2 Go) pour fonctionner correctement. Définissez cette valeur dans la variable ASSIZEMAX du segment OMVS de l'ID utilisateur STCRSE. La définition de cette valeur permet de garantir que le démon RSE est doté de la taille de région requise, quelles que soient les modifications apportées à MAXASSIZE dans SYS1.PARMLIB(BPXPRMxx).
- RSE requiert également un grand nombre d'unités d'exécution pour fonctionner correctement. Vous pouvez définir la limite dans la variable THREADSMAX du segment OMVS de l'ID utilisateur STCRSE. La définition de la limite permet de garantir que RSE est doté de la limite d'unité d'exécution requise, quelles que soient les modifications apportées à MAXTHREADS ou MAXTHREADTASKS dans SYS1.PARMLIB(BPXPRMxx). Pour déterminer la valeur correcte de la limite d'unité d'exécution, voir "Remarques relatives à l'optimisation" dans *Référence de configuration de l'hôte* (SC11-6869).
- L'ID utilisateur STCJMON est un autre bon moyen de définir THREADSMAX dans le segment OMVS, le moniteur de travaux JES utilisant une unité d'exécution par connexion client.
- La tâche démarrée Débogueur intégré (DBGMR) est utilisée uniquement par la fonction Débogueur intégré facultative.

Envisagez de restreindre l'ID utilisateur STCRSE. Les utilisateurs possédant l'attribut RESTRICTED ne peuvent pas accéder aux ressources protégées (MVS) auxquelles ils ne sont pas autorisés à accéder de manière spécifique.

```
ALTUSER STCRSE RESTRICTED
```

Pour vous assurer que les utilisateurs restreints n'ont pas accès aux ressources du système de fichiers z/OS UNIX via "d'autres" bits d'autorisation, définissez le profil RESTRICTED.FILESYS.ACCESS dans la classe UNIXPRIV avec UACC(NONE). Pour plus d'informations sur la restriction des ID utilisateurs, voir le manuel *Security Server RACF Security Administrator's Guide* (SA22-7683).

**Avertissement :** Si vous utilisez des ID utilisateur restreints, ajoutez de manière explicite le droit d'accès à une ressource avec les commandes TSO **PERMIT** ou z/OS UNIX **setfacl**. Sont incluses les ressources dans lesquelles la documentation Developer for System z utilise UACC, comme le profil \*\* dans la classe PROGRAM, ou qui se fondent sur les conventions z/OS UNIX, lorsque tous les utilisateurs possèdent les droits d'accès en lecture et en exécution aux bibliothèques Java. Testez l'accès avant de l'activer sur un système de production.

## Définition de RSE en tant que serveur z/OS UNIX sécurisé

RSE requiert un accès UPDATE au profil BPX.SERVER pour créer ou supprimer l'environnement de sécurité de l'unité d'exécution du client. Si ce profil n'est pas défini, UID(0) est nécessaire pour RSE. Cette étape est requise pour la connexion des clients.

Le débogueur intégré requiert un accès UPDATE au profil BPX.SERVER pour créer ou supprimer l'environnement de sécurité de l'unité d'exécution. Si ce profil n'est pas défini, un ID utilisateur UID(0) est requis pour l'ID utilisateur de la tâche démarrée STCDBM. Cette autorisation est requise uniquement lorsque la fonction de débogueur intégrée facultative est utilisée.

- 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

**Avertissement :** La définition du profil BPX.SERVER permet de configurer z/OS UNIX comme un commutateur global qui bascule de la sécurité de niveau UNIX à la sécurité plus étendue de z/OS UNIX. Ce basculement peut avoir une incidence sur d'autres applications et opérations z/OS UNIX. Vous devez tester la sécurité avant de l'activer sur un système de production. Pour plus d'informations sur les différents niveaux de sécurité, voir *UNIX System Services Planning* (GA22-7800).

## Définition des bibliothèques contrôlées de programme MVS pour RSE

Les serveurs disposant des droits BPX.SERVER doivent être exécutés dans un environnement propre, contrôlé par un programme. Cette exigence implique que tous les programmes appelés par RSE doivent également être contrôlés par programme. Pour les bibliothèques de chargement MVS, le contrôle par programme est géré par votre logiciel de sécurité. Cette étape est requise pour la connexion des clients.

RSE utilise la bibliothèque système (SYS1.LINKLIB), l'environnement d'exécution de Language Environment' (CEE.SCEERUN\*) et la bibliothèque de chargement de la passerelle client TSO/ISPF d'ISPF (ISP.SISPLOAD).

- RALTER PROGRAM \*\* UACC(READ) ADDMEM('SYS1.LINKLIB'//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

**Remarque :** N'utilisez pas le profil \*\* si le profil \* existe déjà dans la classe PROGRAM. Le profil occulte et complique le chemin de recherche utilisé par votre logiciel de sécurité. Dans ce cas de figure, vous devez fusionner la définition \* existante et la nouvelle définition \*\*. Utilisez le profil \*\*, comme indiqué dans le manuel dans *Security Server RACF Security Administrator's Guide* (SA22-7683).

Les bibliothèques prérequis suivantes doivent être contrôlées par un programme pour la prise en charge des services facultatifs. Cette liste n'inclut pas les fichiers spécifiques d'un produit avec lequel interagit Developer for System z (IBM File Manager, par exemple).

- Autre bibliothèque d'exécution REXX, pour SCLM Developer Toolkit
  - REXX.\*.SEAGALT
- Bibliothèque de chargement système, pour le chiffrement SSL
  - SYS1.SIEALNKE
- Bibliothèque Developer for System z pour le débogueur intégré
  - FEK.SFEKAUTH

**Remarque :** Les bibliothèques qui sont conçues pour le positionnement LSA requièrent également des autorisations de contrôle de programme si l'utilisateur y accède via LINKLIST ou STEPLIB. La présente publication concerne l'utilisation des bibliothèques LPA suivantes :

- ISPF, pour la passerelle client TSO/ISPF
  - ISP.SISPLPA
- Bibliothèque d'exécution REXX, pour SCLM Developer Toolkit
  - REXX.\*.SEAGLPA
- Developer for System z, pour CARMA
  - FEK.SFEKLPA

## Définition de la prise en charge de PassTicket pour RSE

Le mot de passe du client ou toute autre méthode d'identification, telle qu'un certificat X.509, est utilisé uniquement pour vérifier l'identité lors de la connexion. Par la suite, les mots de passe PassTicket permettent de gérer la sécurité des unités d'exécution. Cette étape est requise pour la connexion des clients.

Les PassTickets sont des mots de passe générés par le système pour un cycle de vie d'environ 10 minutes. Les passtickets générés se fondent sur une clé secrète. Cette clé est un numéro 64 bits (16 caractères hexadécimaux). Dans l'exemple suivant de commandes RACF, remplacez la marque de réservation key16 par une chaîne hexadécimale à 16 caractères fournie par l'utilisateur qui comporte les caractères 0 à 9 et A à 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)
```
- ```
SETOPTS RACLIST(PTKTDATA) REFRESH
```

RSE prend en charge l'utilisation d'un ID application autre que FEKAPPL. Supprimez la mise en commentaire et personnalisez l'option "APPLID=FEKAPPL" dans `rsed.envvars` pour l'activer, comme indiqué à la section "Définition de paramètres de démarrage Java supplémentaires avec `_RSE_JAVAOPTS`" du manuel *IBM Rational Developer for System z - Guide de configuration de l'hôte*. Les définitions de classe PTKTDATA doivent correspondre à l'ID application réel utilisé par RSE.

Vous ne devez pas utiliser OMVSAPPL comme ID d'application, car il ouvrira la clé confidentielle de la plupart des applications z/OS UNIX. De la même manière, vous ne devez pas utiliser l'ID d'application par défaut MVS, qui est MVS suivi par l'ID SMF du système, car il ouvre la clé confidentielle de la plupart des applications MVS, y compris les travaux par lots des utilisateurs.

**Remarque :**

- Si la classe PTKTDATA est déjà définie, vérifiez qu'elle est définie en tant que classe générique avant de créer les profils indiqués ci-dessus. La prise en charge de caractères génériques dans la classe PTKTDATA est une nouveauté disponible à partir de z/OS édition 1.7, avec l'introduction d'une interface Java pour les mots de passe PassTicket.
- Remplacez le caractère générique (\*) dans la définition `IRRPTAUTH.FEKAPPL.*` par un masque d'ID utilisateur valide afin de limiter les ID utilisateur pour lesquels RSE peut générer un mot de passe PassTicket.
- En fonction des paramètres RACF configurés, l'utilisateur qui définit un profil peut également figurer dans la liste d'accès du profil. Supprimez ce droit pour les profils PTKTDATA.
- Le moniteur de travaux JES et RSE doivent posséder le même ID application pour permettre au gestionnaire d'évaluer les mots de passe PassTicket présentés par RSE. Pour le moniteur de travaux JES, l'ID d'application est défini dans le fichier de configuration FEJCNFG avec la directive APPLID.
- Si un produit cryptographique est installé et disponible sur le système, vous pouvez chiffrer la clé de l'application de connexion sécurisée pour renforcer la protection. Pour ce faire, utilisez le mot clé KEYENCRYPTED au lieu du mot clé KEYMASKED. Pour plus d'informations, voir la documentation *Security Server RACF Security Administrator's Guide* (SA22-7683).

**Avertissement :** La demande de connexion client échoue si les passtickets ne sont pas correctement configurés.

## Définition de la protection des applications pour RSE

Lors de la connexion du client, le démon RSE vérifie que l'utilisateur est autorisé à utiliser l'application.

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

**Remarque :**

- Comme indiqué plus en détails dans «Définition de la prise en charge de PassTicket pour RSE», à la page 49, RSE prend en charge l'utilisation d'un ID application autre que FEKAPPL. La définition de classe APPL doit correspondre à l'ID application réel utilisé par RSE.

- La demande de connexion client aboutit si l'ID application n'est pas défini dans la classe APPL.
- La demande de connexion client échoue uniquement si l'ID application est défini et si l'utilisateur ne bénéficie pas d'accès en lecture sur le profil.

## Définition de la sécurité des commandes JES

Le moniteur de travaux JES émet toutes les commandes d'opérateur JES demandées par un utilisateur via une console EMCS dont le nom est contrôlé à l'aide de la directive `CONSOLE_NAME`, comme indiqué dans la section "FEJJCNFG, JES Job Monitor configuration file" du document *IBM Rational Developer for System z Host Configuration Guide*.

Les exemples de commande RACF suivants donnent aux utilisateurs Developer for System z un accès conditionnel à un nombre limité de commandes JES, à savoir Mettre en attente, Libérer, Annuler et Purger. Les utilisateurs possèdent des droits d'exécution uniquement s'ils lancent les commandes via le moniteur de travaux JES. Remplacez la marque de réservation `#console` par le nom réel de la console.

- ```
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
```

### Remarque :

- L'utilisation de la console est autorisé si aucun profil `MVS.MCSOPER.#console` n'a été défini.
- La classe `CONSOLE` doit être active pour permettre le fonctionnement de `WHEN(CONSOLE(JMON))` mais il n'y a pas de vérification réelle du profil dans la classe `CONSOLE` pour les consoles EMCS.
- Ne remplacez pas `JMON` par le nom réel de la console dans la clause `WHEN(CONSOLE(JMON))`. Le mot clé `JMON` représente l'application de point d'entrée, pas le nom de la console.

**Avertissement :** La définition des commandes JES à l'aide de l'accès universel `NONE` dans votre logiciel de sécurité peut avoir une incidence sur les autres applications et opérations. Vous devez tester la sécurité avant de l'activer sur un système de production.

Le tableau 11 et le tableau 12, à la page 52 présentent des commandes d'opérateur soumises pour JES2 et JES3, et les profils de sécurité discrets qui peuvent être utilisés pour les protéger.

Tableau 11. Commandes d'opérateur du moniteur de travaux JES2

| Action            | Commande                            | Profil OPERCMDS                                                            | Droit d'accès requis |
|-------------------|-------------------------------------|----------------------------------------------------------------------------|----------------------|
| Mettre en attente | \$Hx(jobid)<br>avec x = {J, S ou T} | jesname.MODIFYHOLD.BAT<br>jesname.MODIFYHOLD.STC<br>jesname.MODIFYHOLD.TSU | UPDATE               |



Tableau 11. Commandes d'opérateur du moniteur de travaux JES2 (suite)

| Action  | Commande                              | Profil OPERCMDS                                                                     | Droit d'accès requis |
|---------|---------------------------------------|-------------------------------------------------------------------------------------|----------------------|
| Libérer | \$Ax(jobid)<br>avec x = {J, S ou T}   | jesname.MODIFYRELEASE.BAT<br>jesname.MODIFYRELEASE.STC<br>jesname.MODIFYRELEASE.TSU | UPDATE               |
| Annuler | \$Cx(jobid)<br>avec x = {J, S ou T}   | jesname.CANCEL.BAT<br>jesname.CANCEL.STC<br>jesname.CANCEL.TSU                      | UPDATE               |
| Purger  | \$Cx(jobid),P<br>avec x = {J, S ou T} | jesname.CANCEL.BAT<br>jesname.CANCEL.STC<br>jesname.CANCEL.TSU                      | UPDATE               |

Tableau 12. Commandes d'opérateur du moniteur de travaux JES3

| Action            | Commande     | Profil OPERCMDS    | Droit d'accès requis |
|-------------------|--------------|--------------------|----------------------|
| Mettre en attente | *F,J=jobid,H | jesname.MODIFY.JOB | UPDATE               |
| Libérer           | *F,J=jobid,R | jesname.MODIFY.JOB | UPDATE               |
| Annuler           | *F,J=jobid,C | jesname.MODIFY.JOB | UPDATE               |
| Purger            | *F,J=jobid,C | jesname.MODIFY.JOB | UPDATE               |

**Remarque :**

- Les commandes d'opérateur JES Mettre en attente, Libérer, Annuler et Purger, ainsi que la commande Afficher JCL, peuvent être exécutées uniquement sur des fichiers spoule appartenant à l'ID utilisateur du client, sauf si vous avez indiqué LIMIT\_COMMANDS= avec la valeur LIMITED ou NOLIMIT dans le fichier de configuration du moniteur de travaux JES. Pour plus d'informations, reportez-vous au tableau "Actions sur les travaux - Limitations sur les cibles" du *Référence de configuration de l'hôte* (SC11-6869).
- Les utilisateurs peuvent parcourir n'importe quel fichier spoule, sauf si LIMIT\_VIEW=USERID est défini dans le fichier de configuration du moniteur de travaux JES. Pour plus d'informations, voir "Accès aux fichiers spoule" dans *Référence de configuration de l'hôte* (SC11-6869).
- Même si les utilisateurs n'ont pas d'autorisation pour ces commandes d'opérateur, ils peuvent toujours soumettre des travaux et lire les sorties de travaux via le moniteur de travaux JES s'ils disposent de droits d'accès suffisants à des profils qui protègent ces ressources, comme celles des classes JESINPUT, JESJOBS et JESSPOOL.

Supposons que l'accès à l'identité du serveur du moniteur de travaux JES lors de la création d'une console JMON à partir d'une session TSO est empêché par votre logiciel de sécurité. Même si la console peut être créée, le point d'entrée est différent ; par exemple, moniteur de travaux JES/TSO. Les commandes JES exécutées par cette console échouent lors du contrôle de sécurité si la sécurité est configurée comme indiqué dans cette publication et que l'utilisateur ne dispose pas de droits d'accès aux commandes JES via d'autres procédures.



## Définition des profils de fichier

Un accès en lecture pour les utilisateurs et en modification pour les programmeurs système suffit pour la plupart des fichiers Developer for System z. Remplacez la marque de réservation #sysprog par des ID utilisateur ou des noms de groupes RACF. Demandez également au programmeur système qui a installé et configuré le produit de vous fournir les noms de fichier corrects. FEK est le qualificatif de haut niveau par défaut utilisé pendant l'installation, et FEK.#CUST celui relatif aux fichiers créés pendant le processus de personnalisation.

- ```
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(#sysprog)
```
- ```
SETROPTS GENERIC(DATASET) REFRESH
```

### Remarque :

- Protégez FEK.SFEKAUTH des mises à jour car ce fichier dispose de droits APF. Il en est de même pour FEK.SFEKLOAD et FEK.SFEKLPA, mais car ces fichiers sont contrôlés par programme.
- Les exemples de commande utilisés dans la présente publication et dans le travail FEKRACF supposent que l'EGN (Enhanced Generic Naming) est activé. Dans ce cas, le qualificatif \*\* peut être utilisé pour représenter tout nombre de qualificatifs dans la classe DATASET. Remplacez \*\* par \* si l'EGN n'est pas activé dans votre système. Pour plus d'informations sur EGN, voir le manuel *Security Server RACF Security Administrator's Guide* (SA22-7683).

Certains des composants Developer for System z facultatifs requièrent des profils de fichier de sécurité supplémentaires. Remplacez les marques de réservation #sysprog, #ram-developer et #cicsadmin par des ID utilisateur ou par des noms de groupes RACF valides :

- Si la traduction des noms longs/abrégiés de SCLM Developer Toolkit est utilisée, les utilisateurs doivent disposer d'un accès en mise à jour (UPDATE) au mappage VSAM, 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(#sysprog)
—
SETROPTS GENERIC(DATASET) REFRESH
```

- Les développeurs CARMA RAM (Repository Access Manager) requièrent un accès UPDATE aux VSAM 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(#sysprog)
```

```

-
PERMIT 'FEK.#CUST.CRA*.*' CLASS(DATASET) ACCESS(UPDATE) ID(#ram-developer)
-
SETROPTS GENERIC(DATASET) REFRESH

```

- Si le serveur CRD (définition de ressource CICS) du gestionnaire de déploiement d'application est utilisé, l'administrateur CICS doit détenir un accès UPDATE à la méthode VSAM du référentiel CRD.

```

-
ADDSD 'FEK.#CUST.ADNREP*.*' UACC(READ)
DATA('RATIONAL DEVELOPER FOR SYSTEM Z - ADN')
-
PERMIT 'FEK.#CUST.ADNREP*.*' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
-
PERMIT 'FEK.#CUST.ADNREP*.*' CLASS(DATASET) ACCESS(UPDATE) ID(#cicsadmin)
-
SETROPTS GENERIC(DATASET) REFRESH

```

- Si le référentiel de manifestes du gestionnaire de déploiement d'application est défini, tous les utilisateurs de CICS Transaction Server requièrent un accès en mise à jour (UPDATE) au VSAM du référentiel de manifestes.

```

-
ADDSD 'FEK.#CUST.ADNMAN*.*' UACC(UPDATE)
DATA('RATIONAL DEVELOPER FOR SYSTEM Z - ADN')
-
PERMIT 'FEK.#CUST.ADNMAN*.*' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
-
SETROPTS GENERIC(DATASET) REFRESH

```

Utilisez les exemples de commande RACF suivants pour obtenir une configuration encore plus sécurisée dans laquelle l'accès READ est également contrôlé.

- Protection uacc(none) des fichiers

```

-
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')
• Autoriser le programmeur système à gérer toutes les bibliothèques
-
PERMIT 'FEK.*.** CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
-
PERMIT 'FEK.SFEKAUTH CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
-
PERMIT 'FEK.SFEKLOAD CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
-
PERMIT 'FEK.SFEKLMOD CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
-
PERMIT 'FEK.SFEKPROC CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
-
PERMIT 'FEK.#CUST.PARMLIB CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
-
PERMIT 'FEK.#CUST.CNTL CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
-
PERMIT 'FEK.#CUST.SQL CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
-
PERMIT 'FEK.#CUST.LSTRANS.*.**' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
-
PERMIT 'FEK.#CUST.CRA*.**' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
-
PERMIT 'FEK.#CUST.ADNREP*.**' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
-
PERMIT 'FEK.#CUST.ADNMAN*.**' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
• Autoriser les clients à accéder aux bibliothèques de chargement et d'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(*)

```

**Remarque :** Aucune autorisation n'est nécessaire pour FEK.SFEKLPA, car tout le code qui se trouve dans LPA est accessible à tous les utilisateurs.

- Autorisation d'accès du débogueur intégré à la bibliothèque de chargement
  - PERMIT 'FEK.SFEKAUTH' CLASS(DATASET) ACCESS(READ) ID(STCDBM)
- Autorisation d'accès du moniteur de travaux JES à la bibliothèque de chargement et de paramètres
  - 
  - PERMIT 'FEK.SFEKAUTH' CLASS(DATASET) ACCESS(READ) ID(STCJMON)
  - 
  - PERMIT 'FEK.#CUST.PARMLIB' CLASS(DATASET) ACCESS(READ) ID(STCJMON)
- (Facultatif) Autorisation pour les clients de mettre à jour la conversion de nom long/abrégé VSAM pour SCLMDT
  - 
  - PERMIT 'FEK.#CUST.LSTRANS\*.\*\*' CLASS(DATASET) ACCESS(UPDATE) ID(\*)
- (Facultatif) Autorisation pour les développeurs de gestionnaire RAM de mettre à jour la méthode d'accès VSAM CARMA pour CARMA
  - 
  - PERMIT 'FEK.#CUST.CRA\*.\*\*' CLASS(DATASET) ACCESS(UPDATE) ID(#ram-developer)
- (Facultatif) Autorisation pour les utilisateurs CICS de lire le VSAM du référentiel CRD pour le gestionnaire de déploiement d'application
  - 
  - PERMIT 'FEK.#CUST.ADNREP\*.\*\*' CLASS(DATASET) ACCESS(READ) ID(\*)
- (Facultatif) Autorisation pour les administrateurs CICS de mettre à jour le VSAM du référentiel CRD pour le gestionnaire de déploiement d'application
  - 
  - PERMIT 'FEK.#CUST.ADNREP\*.\*\*' CLASS(DATASET) ACCESS(UPDATE) ID(#cicsadmin)
- (Facultatif) Autorisation pour les utilisateurs CICS de mettre à jour le VSAM du référentiel de manifestes pour le gestionnaire de déploiement d'application
  - 
  - PERMIT 'FEK.#CUST.ADNMAN\*.\*\*' CLASS(DATASET) ACCESS(UPDATE) ID(\*)
- (Facultatif) Autorisation d'accès du serveur TS CICS à la bibliothèque de chargement pour les options bidirectionnelles et le Gestionnaire de déploiement d'application
  - 
  - PERMIT 'FEK.SFEKLOAD' CLASS(DATASET) ACCESS(READ) ID(#cicsts)
- (Facultatif) Autorisation d'accès du serveur TS CICS, des régions IMS et des travaux par lots MVS à la bibliothèque de chargement pour les messages 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)

- Activation des profils de sécurité

—

```
SETROPTS GENERIC(DATASET) REFRESH
```

Lorsque vous définissez le contrôle des droits d'accès en lecture (READ) aux fichiers système, vous devez fournir aux serveurs et aux utilisateurs de Developer for System z les droits d'accès en lecture (READ) aux fichiers suivants :

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

**Remarque :** Lorsque vous utilisez le package produit Alternate Library for REXX, le nom de la bibliothèque d'exécution REXX par défaut est REXX.\*.SEAGALT au lieu de REXX.\*.SEAGLPA, comme utilisé dans l'exemple ci-dessus.

## Définition de fichiers contrôlés par programme z/OS UNIX pour RSE

Les serveurs disposant des droits BPX.SERVER doivent être exécutés dans un environnement propre, contrôlé par un programme. Cette exigence signifie que tous les programmes appelés par RSE doivent également être contrôlés par programme. Pour les fichiers z/OS UNIX, le contrôle par programme est géré par la commande **extattr**. Pour exécuter cette commande vous devez disposer du droit d'accès en lecture (READ) sur BPX.FILEATTR.PROGCTL dans la classe FACILITY ou avoir l'ID utilisateur UID(0).

Le serveur RSE utilise la bibliothèque partagée Java de RACF (/usr/lib/libIRRac\*.so).

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

### Remarque :

- Depuis z/OS 1.9, /usr/lib/libIRRac\*.so est installé en mode de contrôle par programme lors de l'installation de SMP/E RACF.
- Depuis z/OS 1.10, /usr/lib/libIRRac\*.so fait partie de SAF, lequel est fourni avec la version z/OS de base ; par conséquent, il est également disponible pour les clients non RACF.
- La configuration peut varier si vous utilisez un produit de sécurité autre que RACF. Pour de plus amples informations, consultez la documentation de votre produit de sécurité.
- L'installation SMP/E de Developer for System z définit le bit de contrôle par programme pour les programmes RSE internes.
- Utilisez la commande **Is -Eog** z/OS UNIX pour afficher l'état en cours du bit de contrôle par programme. Le fichier est contrôlé par un programme si la lettre **p** apparaît dans la deuxième chaîne.

```
$ ls -l /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
```

## Vérification des paramètres de sécurité

Utilisez les exemples de commande ci-dessous pour afficher les résultats de vos personnalisations de la sécurité.

- Paramètres et classes de sécurité
  - SETROPTS LIST
- Segment OMVS pour les utilisateurs
  - LISTUSER #userid NORACF OMVS
  - LISTGRP #group-name NORACF OMVS
- Tâches démarrées
  - LISTGRP STCGROUP OMVS
  - LISTUSER STCDBM OMVS
  - LISTUSER STCJMON OMVS
  - LISTUSER STCRSE OMVS
  - RLIST STARTED DBGMGR.\* ALL STDATA
  - RLIST STARTED JMON.\* ALL STDATA
  - RLIST STARTED RSED.\* ALL STDATA
- RSE comme serveur z/OS UNIX sécurisé
  - RLIST FACILITY BPX.SERVER ALL
- Bibliothèques contrôlées par programme MVS pour RSE
  - RLIST PROGRAM \*\* ALL
- Prise en charge de PassTicket pour RSE
  - RLIST PTKTDATA FEKAPPL ALL SSIGNON
  - RLIST PTKTDATA IRRPTAUTH.FEKAPPL.\* ALL
- Protection des applications pour RSE
  - RLIST APPL FEKAPPL ALL
- Sécurité de commande JES
  - RLIST CONSOLE JMON ALL
  - RLIST OPERCMDS MVS.MCSOPER.JMON ALL
  - RLIST OPERCMDS JES%.\*\* ALL
- Profils de fichier
  - LISTGRP FEK
  - LISTDSD PREFIX(FEK) ALL
- Fichiers contrôlés par programme z/OS UNIX pour RSE
  - ls -E /usr/lib/libIRRRacf\*.so

En option, des profils définissant le comportement de Developer for System z pour un utilisateur spécifique peuvent exister. Ces profils correspondent au filtre FEK.\*\* et se trouvent par défaut dans la classe FACILITY. Voir la directive `_RSE_FEK_SAF_CLASS` dans `rsed.envvars`. La commande **SEARCH** permet de dresser la liste des noms de profils. Vous pouvez ensuite utiliser la commande **RLIST** pour afficher les détails d'un profil.

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

## Chapitre 3. Remarques relatives à TCP/IP

Developer for System z repose sur le protocole TCP/IP pour offrir l'accès au mainframe à des utilisateurs travaillant sur un poste de travail autre qu'un mainframe. TCP/IP sert également à assurer la communication entre les différents composants et les autres produits.

Notez que la plupart des fonctions Developer for System z sont basées sur z/OS UNIX ; par conséquent, TCP/IP utilisera l'ordre de recherche z/OS UNIX pour trouver ses fichiers de configuration. Pour plus d'informations, voir Chapitre 14, «Configuration de TCP/IP», à la page 205.

Les rubriques suivantes sont traitées dans le présent chapitre :

- «Ports TCP/IP»
- «Remplacement du comportement TCP/IP par défaut», à la page 62
- «Piles multiples (CINET)», à la page 62
- «Distributed Dynamic VIPA», à la page 64

### Ports TCP/IP

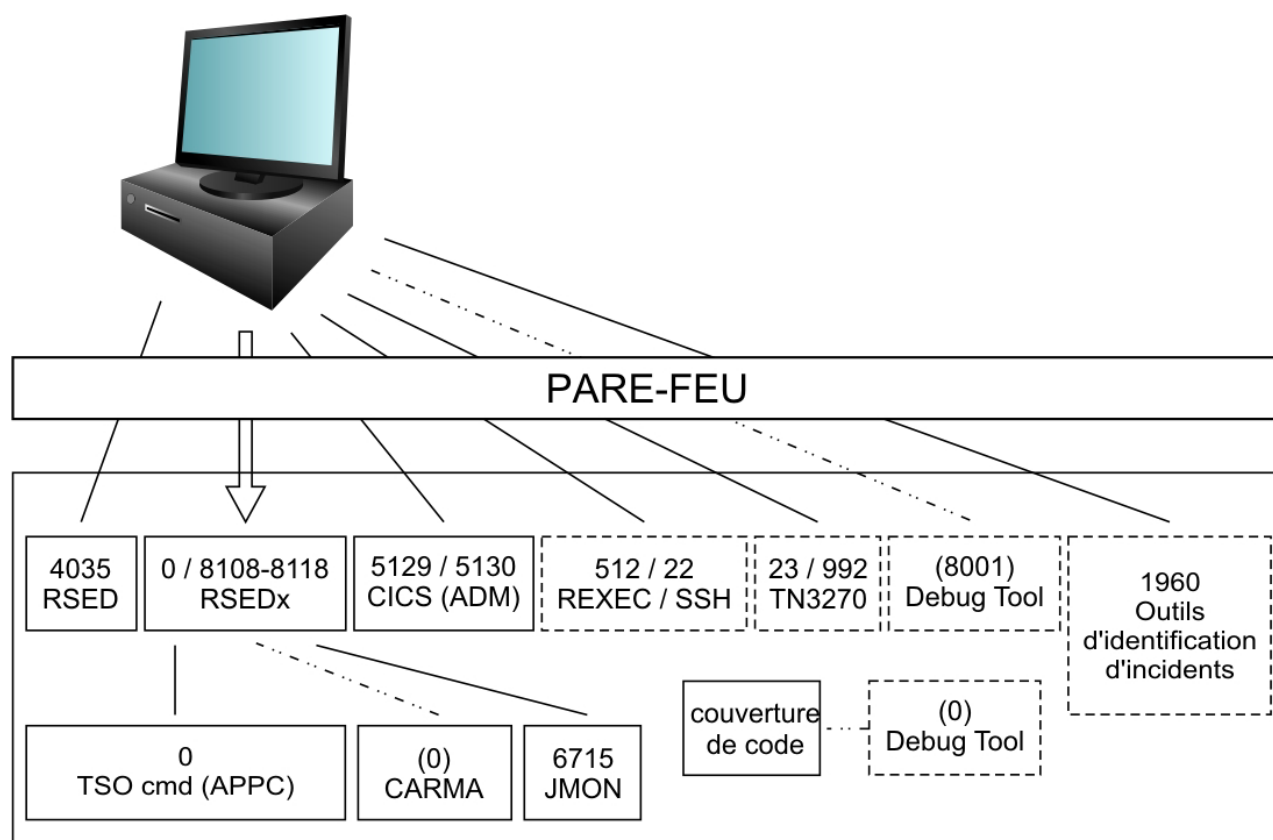


Figure 10. Ports TCP/IP

figure 10, à la page 59 présente les ports TCP/IP qui peuvent être utilisés par Developer for System z. Les flèches indiquent la partie qui assure la liaison (tête de flèche) et celle qui assure la connexion.

## Communications externes

Définissez les ports suivants sur le pare-feu qui protège l'hôte z/OS car ils sont utilisés pour les communications client-hôte (via le protocole tcp) :

- Démon RSE pour la configuration des communications client-hôte, port 4035 par défaut. Le port peut être défini dans le fichier de configuration `rsed.envvars`. La communication sur ce port peut être chiffrée à l'aide de SSL ou TLS.
- Serveur RSE pour la communication client-hôte. Par défaut, tout port disponible est utilisé, mais une plage de ports peut être définie à l'aide de la définition `_RSE_PORTRANGE` dans le fichier `rsed.envvars`. La plage de ports par défaut pour `_RSE_PORTRANGE` est comprise entre 8108 et 8118 (11 ports). La communication sur ce port peut être chiffrée à l'aide de SSL ou TLS.
- (Facultatif) Gestionnaire de débogage pour les services de débogueur intégré, port par défaut 5335. Le port peut être défini dans le langage JCL de la tâche démarrée `DBGMGR`. La communication sur ce port peut être chiffrée à l'aide de SSL ou TLS.
- (Facultatif) Service INETD pour les actions à distance (basées sur l'hôte) dans sous-projets z/OS UNIX :
  - REXEC (version z/OS UNIX), port par défaut 512.
  - SSH (version z/OS UNIX), port par défaut 22. La communication sur ce port est chiffrée à l'aide de la fonction SSL.
- (facultatif) Service Telnet TN3270 pour l'émulateur de connexion à l'hôte, port 23 par défaut. Les communications externes peuvent être chiffrées à l'aide de SSL ou TLS (port par défaut 992). Le port par défaut affecté au service Telnet TN3270 dépend de l'utilisation ou non du chiffrement par l'utilisateur.
- (facultatif) Interfaces d'application CICSTS pour le gestionnaire de déploiement d'application :
  - Interface RESTful, port par défaut 5130. Le port peut être défini dans le CSD CICS.
  - Interface de services Web, port par défaut 5129. Le port peut être défini dans le CSD CICS. La communication sur ce port peut être chiffrée à l'aide de la fonction SSL.

**Remarque :** Normalement, le client indique l'adresse TCP/IP utilisée pour se connecter à l'hôte. Cependant, pour s'assurer que les sessions de débogage communiquent avec l'hôte correct, le gestionnaire de débogage indique l'adresse TCP/IP à utiliser.

## Communications internes

Plusieurs services hôte Developer for System z s'exécutent dans des unités d'exécution ou espaces adresse séparés et utilisent des sockets TCP/IP comme mécanisme de communication. Tous ces services utilisent RSE pour communiquer avec le client et limitent leur flux de données à l'hôte. Pour certains services, n'importe quel port est utilisé. Pour d'autres, le programmeur système peut sélectionner un port ou une plage de ports à utiliser :

- Moniteur de travaux JES pour les services associés à JES, port par défaut 6715. Vous pouvez définir le port dans le membre de configuration `FEJJCNFG` et le répéter dans le fichier de configuration `rsed.envvars`.



- (Facultatif) La communication CARMA utilise par défaut un port temporaire, mais une plage de ports peut être définie dans le fichier de configuration CRASRV.properties.
- (Facultatif) Gestionnaire de débogage pour les services liés aux débogage, port par défaut 5336. Le port peut être défini dans le langage JCL de la tâche démarrée DBGMGR.
- La couverture de code basée sur l'hôte, qui est un travail par lots, alloue un port temporaire pour permettre au débogueur intégré de communiquer avec elle et de fournir les données nécessaires pour le rapport de couverture de code.

## Réservation de port TCP/IP

Si vous utilisez l'instruction PORT ou PORTRANGE dans PROFILE.TCPIP pour réserver les ports utilisés par Developer for System z, notez que de nombreux liens sont établis par les unités d'exécution qui sont actives dans un pool d'unités d'exécution RSE. Le nom de travail du pool d'unités d'exécution RSE est RSEDx, où RSED est le nom de la tâche démarrée RSE et x est un nombre à un chiffre aléatoire, si bien que la définition contient obligatoirement des caractères génériques.

```
PORT 4035 TCP RSED ; Developer for System z - RSE daemon
PORT 6715 TCP JMON ; Developer for System z - JES job monitor
PORT 5335 TCP DBGMGR ; Developer for System z - Integrated
debugger
PORT 5336 TCP DBGMGR ; Developer for System z - Integrated
debugger
PORTRange 8108 11 TCP RSED* ; Developer for System z - _RSE_PORTRANGE
;PORTRange 5227 100 TCP RSED* ; Developer for System z - CARMA
```

## CARMA et ports TCP/IP

CARMA (Common Access Repository Manager) permet d'accéder à un SCM (Software Configuration Manager) basé sur un hôte, par exemple CA Endevor SCM. Dans la plupart des cas, comme pour le démon RSE, un serveur assure la liaison à un port et écoute les demandes de connexion. Toutefois, CARMA utilise une démarche différente, étant donné que le serveur CARMA n'est pas encore actif lorsque le client lance la demande de connexion.

Lorsque le client envoie une demande de connexion, l'exploitant CARMA, qui est actif comme une unité d'exécution utilisateur d'un pool d'unités d'exécution RSE, demande un port temporaire ou trouve un port libre dans la plage indiquée dans le fichier de configuration CRASRV.properties et procède à la liaison. L'exploitant démarre le serveur CARMA et transmet le numéro de port, de sorte que le serveur sache à quel port se connecter. Une fois le serveur connecté, le client peut envoyer les demandes au serveur et recevoir les résultats.

Du point de vue de TCP/IP, RSE (via le logiciel de fouille de données CARMA) constitue le serveur qui établit la liaison au port et le serveur CARMA représente le client qui s'y connecte.

Si vous utilisez l'instruction PORT ou PORTRANGE dans PROFILE.TCPIP pour réserver la plage de ports utilisée par CARMA, notez que le logiciel de fouille de données CARMA est actif dans un pool d'unités d'exécution RSE. Le nom de travail du pool d'unités d'exécution RSE est RSEDx, où RSED correspond au nom de la tâche RSE démarrée et x à un chiffre aléatoire unique, si bien que la définition contient obligatoirement des caractères génériques.

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

## Remarques relatives à LDAP

Le serveur RSE peut être configuré pour lancer une requête sur un ou plusieurs serveurs LDAP portant sur différents services Developer for System z :

- Requête portant sur le support de plusieurs groupes de développeurs pour la fonction d'envoi au client dans les groupes LDAP
- Requête portant sur une ou plusieurs listes de révocation de certificat pour l'authentification X.509

Notez que les mesures de sécurité TCP/IP, telles que des pare-feux, peuvent empêcher le serveur RSE (résidant sur l'hôte) de contacter le serveur LDAP. Utilisez les informations suivantes pour faire en sorte que le serveur LDAP puisse être atteint :

- Les adresses TCP/IP ou les noms DNS du serveur LDAP sont répertoriés dans les variables \*\_LDAP\_SERVER dans `rsed.envvars`.
- Les numéros de port du serveur LDAP sont répertoriés dans les variables \*\_LDAP\_PORT dans `rsed.envvars`.
- LDAP utilise le protocole TCP.
- Le serveur LDAP est contacté par le serveur RSE résidant sur l'hôte.
- Le serveur RSE est actif dans un espace adresse RSEDx, où RSED est le nom de la tâche démarrée RSE et x est un nombre à un chiffre aléatoire, par exemple, RSED8.

---

## Remplacement du comportement TCP/IP par défaut

### Fonction de retardement d'accusé de réception

La fonction de retardement d'accusé de réception retarde la réception d'un accusé de réception d'un paquet TCP de 200 ms au maximum. Ce retard augmente les chances que l'accusé de réception puisse être envoyé avec la réponse au paquet reçu, ce qui réduit le trafic réseau. Toutefois, si l'émetteur attend de recevoir l'accusé de réception pour envoyer un nouveau paquet (par exemple, en raison de l'implémentation d'un algorithme de Nagle) et qu'il n'y a aucune réponse au paquet qui vient d'être envoyé (par exemple, lors d'un transfert de fichiers), la communication est retardée inutilement.

Developer for System z vous permet de désactiver la fonction de retardement d'accusé de réception. Sur l'hôte, cela s'effectue via la directive `DSTORE_TCP_NO_DELAY` dans `rsed.envvars`, comme indiqué dans le document *Guide de configuration de l'hôte* (SC11-6285).

---

## Piles multiples (CINET)

z/OS Communication Server permet l'activation simultanée de plusieurs piles TCP/IP dans un seul système. Il s'agit dans ce cas d'une configuration CINET.

Si Developer for System z n'est pas actif sur la pile par défaut, les fonctions Developer for System z sélectionnées risquent d'échouer. L'utilisation de l'affinité entre piles permet de résoudre ce problème. L'affinité entre piles signale à Developer for System z d'utiliser uniquement une pile TCP/IP spécifique (et non toutes les piles TCP/IP disponibles, ce qui est la valeur par défaut pour les tâches démarrées).

L'affinité entre piles est définie pour les tâches démarrées RSED en supprimant la mise en commentaire et en personnalisant la directive `_BPXK_SETIBMOPT_TRANSPORT` dans le fichier de configuration `rsed.envvars`. Pour plus d'informations sur la personnalisation de ce fichier de configuration, voir la section connexe dans le "Chapitre 2 Personnalisation de base" du document *Guide de configuration de l'hôte* (SC23-7658).

## CARMA et affinité entre piles

CARMA (Common Access Repository Manager) permet d'accéder à un SCM (Software Configuration Manager) basé sur un hôte, par exemple CA Endevor SCM. Pour ce faire, CARMA démarre le serveur spécifique à un utilisateur qui doit être configuré pour une application de l'affinité entre piles.

À l'instar des tâches démarrées *Developer for System z*, l'affinité entre piles d'un serveur CARMA est définie à l'aide de la variable `_BPXK_SETIBMOPT_TRANSPORT` qui doit être transmise à LE (Language Environment). Pour ce faire, réglez la commande de démarrage dans le fichier de configuration `crastart*.conf` ou `CRASUB*` actif.

### Remarque :

- Le nom exact du fichier de configuration contenant la commande de démarrage dépend des différentes options sélectionnées par le programmeur-système qui a configuré CARMA. Pour plus d'informations sur ce sujet, voir le chapitre 3 "(Optional) Common Access Repository Manager (CARMA)" du manuel *Guide de configuration de l'hôte* (SC23-7658).
- `_BPXK_SETIBMOPT_TRANSPORT` spécifie le nom de la pile TCP/IP à utiliser, tel que défini dans l'instruction `TCPIPJOBNAME` dans le `TCPIP.DATA` correspondant.
- Le codage d'une instruction de définition de données `SYSTCPD` ne définit pas l'affinité entre piles demandée.
- Par défaut, CARMA n'utilise pas les piles TCP/IP normales. CARMA utilise l'adresse de bouclage pour la communication entre le logiciel de fouille de données CARMA et le serveur CARMA. Cela améliore la sécurité (seuls les processus locaux ont accès à l'adresse de bouclage) et peut éviter d'avoir à ajouter l'affinité entre piles à la communication CARMA.

### **crastart\*.conf**

Remplacez le segment suivant :

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

par celui-ci (où `TCPIP` représente la pile TCP/IP voulue) :

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

**Remarque :** `CRASTART` ne prend pas en charge les continuations de ligne mais la longueur de ligne admise n'est soumise à aucune limite.

### **CRASUB\***

Remplacez le segment suivant :

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

par celui-ci (où `TCPIP` représente la pile TCP/IP voulue) :

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

**Remarque :** La soumission de travail limite la longueur de ligne à 80 caractères. Pour concaténer deux lignes, vous pouvez couper une ligne trop longue à l'emplacement d'un blanc ( ) et utilisez un signe plus (+) à la fin de la première ligne.

---

## Distributed Dynamic VIPA

L'adressage DVIPA distribué (adressage IP virtuel dynamique) permet d'exécuter simultanément des installations Developer for System z identiques sur différents systèmes du sysplex et de demander à TCP/IP, éventuellement avec l'aide de WLM, de distribuer les connexions client entre ces systèmes.

Vous pouvez configurer un adressage distribué DVIPA de différentes manières, mais Developer for System z impose des restrictions sur ces options.

- Le démon RSE détient le port défini pour l'adressage DVIPA réparti, mais le travail réel est exécuté dans le serveur RSE qui est actif comme unité d'exécution dans un autre espace adresse. Par conséquent, vous ne pouvez pas utiliser la méthode de distribution SERVERWLM pour équilibrer la charge sur les systèmes, car WLM fournit des avis en fonction des statistiques du démon RSE et non par du serveur RSE.
- Le client connaît uniquement l'adresse DVIPA utilisée par le démon Sysplex Distributor for RSE. Le Sysplex Distributor envoie la demande de connexion à l'un des démons RSE disponibles, qui démarre une unité d'exécution de serveur qui se lie à un port du système. Lorsque le client se connecte à ce port, il utilise de nouveau l'adresse DVIPA et non pas l'adresse du système et vous devez donc vérifier que le Sysplex Distributor redirige la nouvelle connexion vers le système approprié.

Par conséquent, Developer for System z nécessite la définition de SYSPLEXPORTS dans l'instruction VIPADISTRIBUTE pour que les ports utilisés par les unités d'exécution du serveur RSE soient uniques dans le sysplex.

### Remarque :

- L'utilisation de SYSPLEXPORTS implique de définir la structure EZBEPOR dans la fonction de couplage.
- L'utilisation de SYSPLEXPORTS implique que TCP/IP sélectionne un port éphémère pour la connexion secondaire. Par conséquent, vous ne pouvez pas réserver des ports pour ces connexions dans le profil TCP/IP avec les directives PORT et PORTRANGE. Vous ne pouvez pas utiliser \_RSE\_PORTRANGE dans rsed.envvars pour limiter les ports utilisés par Developer for System z. Developer for System z ne fournit pas de solution à cette restriction car cela complique la configuration du pare-feu.

Il existe des restrictions dans Developer for System z lorsque vous utilisez l'adressage DVIPA distribué :

- Pour que le client Developer for System z n'interfère pas avec la sélection de port correcte par TCP/IP, activez la directive deny.nonzero.port dans rsed.envvars.
- Tous les serveurs Developer for System z participants doivent avoir la même configuration. Vous devez partager /usr/lpp/rdz et /etc/rdz entre tous les systèmes participants. Il est également conseillé de partager /var/rdz/projects, /var/rdz/pushtoclient et /var/rdz/sclmdt si ces répertoires sont utilisés. Notez que /var/rdz/WORKAREA et /var/rdz/logs doivent être uniques pour chaque système.

- Voir Chapitre 11, «Exécution de plusieurs instances», à la page 165 pour savoir quels composants Developer for System z doivent être partagés et quels composants doivent être uniques dans le système.

Le moniteur de travaux JES, CARMA et les autres serveurs Developer for System z interagissent uniquement avec le RSE local et ils ne nécessitent donc pas de configuration DVIPA.

Le débogueur intégré interagit uniquement avec le RSE local et ne nécessite donc pas de configuration DVIPA. Pour s'assurer que les sessions de débogage communiquent avec l'hôte correct, le gestionnaire de débogage indique l'adresse TCP/IP à utiliser et ne nécessite donc pas de configuration DVIPA.

Les adresses DVIPA distribuées sont définies par les mots clés VIPADefine et VIPABackup du bloc VIPADynamic dans votre profil TCP/IP. Le mot clé VIPADIStribute ajoute les définitions Sysplex Distributor nécessaires. L'adressage DVIPA distribué implique que toutes les piles participantes aient connaissance du sysplex, opération qui est exécuté via les mots clés SYSPLEXRouting et DYNAMICXCF du bloc IPCONFIG dans votre profil TCP/IP. Voir *Communications Server: IP Configuration Reference* (SC31-8776) pour plus d'informations sur ces directives.

Voir *MVS Setting Up a Sysplex* (SA22-7625) et *Communication Server: SNA Network Implementation Guide* (SC31-8777) pour plus d'informations sur la configuration de la structure EZBEPORIS dans votre fonction de couplage.

## Restriction de la sélection de port

L'utilisation de SYSPLEXPORTS implique que TCP/IP sélectionne un port éphémère pour la connexion secondaire. Un port éphémère est un port disponible et non réservé. L'utilisation d'un port éphémère entre en conflit avec les meilleures pratiques de pare-feu qui consistent à limiter le nombre de ports ouverts pour les communications car le port qui sera utilisé n'est pas connu.

Vous pouvez ignorer ce problème en forçant Developer for System z à utiliser des ports connus pour la connexion secondaire en définissant un élément \_RSE\_PORTRANGE unique par système et en vous assurant que les plages de ports utilisées sont réservées à l'utilisation de Developer for System z sur tous les systèmes. Pour cela, vous devez disposer de l'APAR TCP/IP PM63379.

Pour garantir que TCP/IP dirige la connexion secondaire vers le système correct, Developer for System z doit utiliser une plage de ports unique sur chaque système. Cela implique que vous ne pouvez pas utiliser de configuration identique partagée pour les systèmes car \_RSE\_PORTRANGE dans rsed.envvars doit être unique. Pour savoir comment configurer plusieurs serveurs avec différents fichiers de configuration tout en utilisant le même code, voir «Niveaux de logiciels identiques, fichiers de configuration différents», à la page 166 dans Chapitre 11, «Exécution de plusieurs instances», à la page 165. Vous devez utiliser un document maître de rsed.envvars et un script pour régler et copier cet élément dans une configuration spécifique au système afin de garantir que le fichier reste identique dans les différents systèmes.

1. Configurez Developer for System z sur SYS1 comme s'il s'agissait d'une seule configuration système mais vérifiez que les éléments /usr/lpp/rdz et /etc/rdz se trouvent dans un système de fichiers partagés. Tous les éléments de type MVS doivent également être partagés avec SYS2.

2. Utilisez /etc/rdz/rsed.envvars comme document maître et ajoutez une référence à /etc/rdz à la fin du fichier de telle sorte que les copies spécifiques au système puissent utiliser les fichiers de configuration restants.

```
$ oedit /etc/rdz/rsed.envvars
-> add the following at the END:
-- NEEDED TO FIND THE REMAINING CONFIGURATION FILES
CFG_BASE=/etc/rdz
CLASSPATH=.:$CFG_BASE:$CLASSPATH
--
```

3. Créez /etc/rdz/update.sh, script shell qui va copier l'élément rsed.envvars maître, et réglez \_RSE\_PORTRANGE

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

```
#!/bin/sh
Licensed materials - Property of IBM
5724-T07 Copyright IBM Corp. 2012
clone rsed.envvars and set PORTRANGE for use with RDz & DDVIPA

file=rsed.envvars #; echo file $file
sys=${1:-$(sysvar SYSNAME)} #; echo sys $sys
dir=$(dirname $0) #; echo dir $dir
if sysname has a special char, precede it with \ (eg. SYS\1)
case "$sys" in
 "SYS1") range=8108-8118;;
 "SYS2") range=8119-8129;;
esac
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
```

Figure 11. update.sh - prise en charge de la configuration DDVIPA avec un pare-feu

4. Créez les répertoires /etc/rdz/SYS1 et /etc/rdz/SYS2 puis exécutez /etc/rdz/update.sh pour insérer des données dans les répertoires.

```
$ 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. Vérifiez que la tâche démarrée RSED désigne /etc/rdz/&SYSNAME.

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

Vous devez ensuite vérifier que les plages de ports définies sont réservées pour Developer for System z sur tous les systèmes du sysplex afin de vous assurer que le numéro de port est unique dans le sysplex. Utilisez l'instruction PORT ou

PORTRANGE dans PROFILE.TCPIP pour réserver toutes les plages sur chaque système. Le nom de travail du pool d'unités d'exécution RSE est RSEDx, où RSED est le nom de la tâche démarrée RSE et x est un nombre à un chiffre aléatoire, si bien que la définition contient obligatoirement des caractères génériques.

```
PORTRange 8108 22 RSED* ; 8108-8129 - Developer for System z
 ; - secondary connection
```

Comme cela est décrit dans «Flux de connexion», à la page 8, la plage de ports dans \_RSE\_PORTRANGE peut être de petite taille. Le serveur RSE n'a pas exclusivement besoin du port pendant la durée de la connexion client. Aucun autre serveur RSE ne peut établir une liaison avec le port que dans l'intervalle de temps entre la liaison (du serveur) et la connexion (du client). Cela signifie que la plupart des connexions utiliseront le premier port de la plage, les autres valeurs de la plage servant de mémoire tampon dans le cas de plusieurs connexions simultanées.

## Exemple de configuration

Dans l'exemple de configuration suivant, il existe deux systèmes z/OS, SYS1 et SYS2, qui font partie d'un sysplex. Le système SYS1 est défini comme le système qui héberge normalement le Sysplex Distributor de l'adressage distribué Developer for System z.

Après avoir défini l'adressage distribué DVIPA, Developer for System z peut être démarré sur les systèmes pour équilibrer les connexions client entre les systèmes. Le moniteur de travaux JES interagit uniquement avec le RSE local et ne nécessite donc pas de configuration DVIPA. Les clients se connecteront au port 4035 à l'adresse IP 10.10.10.1.







```

; DYNAMICXCF définit l'unité/la liaison avec l'adresse de base 9.9.9.2,
le cas échéant
IGNORERedirect

VIPADYNAMIC
VIPABACKUP 255.255.255.0 10.10.10.1
; VIPABACKUP définit 10.10.10.1 comme adresse de
secours DVIPA sur SYS2 pour RDz
VIPADISTRIBUTE DEFINE
; VIPADISTRIBUTE convertit 10.10.10.1 en adresse distribuée DVIPA :
doit correspondre à SYS1
 SYSLEXPORTS ; RDz prérequis
 DISTMETHOD BASEWLM ; BASEWLM ou ROUNDROBIN
 10.10.10.1 ; Adresse DVIPA utilisée par
 les clients RDz
 PORT 4035 ; Port utilisé par les clients RDz
 DESTIP 9.9.9.1 9.9.9.2 ; RDz actif sur SYS1 et SYS2
ENDVIPADYNAMIC

```



---

## Chapitre 4. Remarques relatives à WLM

Contrairement aux applications z/OS traditionnelles, Developer for System z n'est pas une application monolithique qui peut être identifiée facilement au niveau du Workload Manager (WLM). Les différents composants de Developer for System z interagissent pour offrir au client un accès à des services et des données d'hôte. Comme décrit au Chapitre 1, «Description de Developer for System z», à la page 3, certains de ces services sont actifs dans différents espaces adresse; ce qui se traduit par différentes classifications WLM.

Les rubriques suivantes sont traitées dans le présent chapitre :

- «Classification des charges de travail»
- «Définition des objectifs», à la page 73

---

### Classification des charges de travail

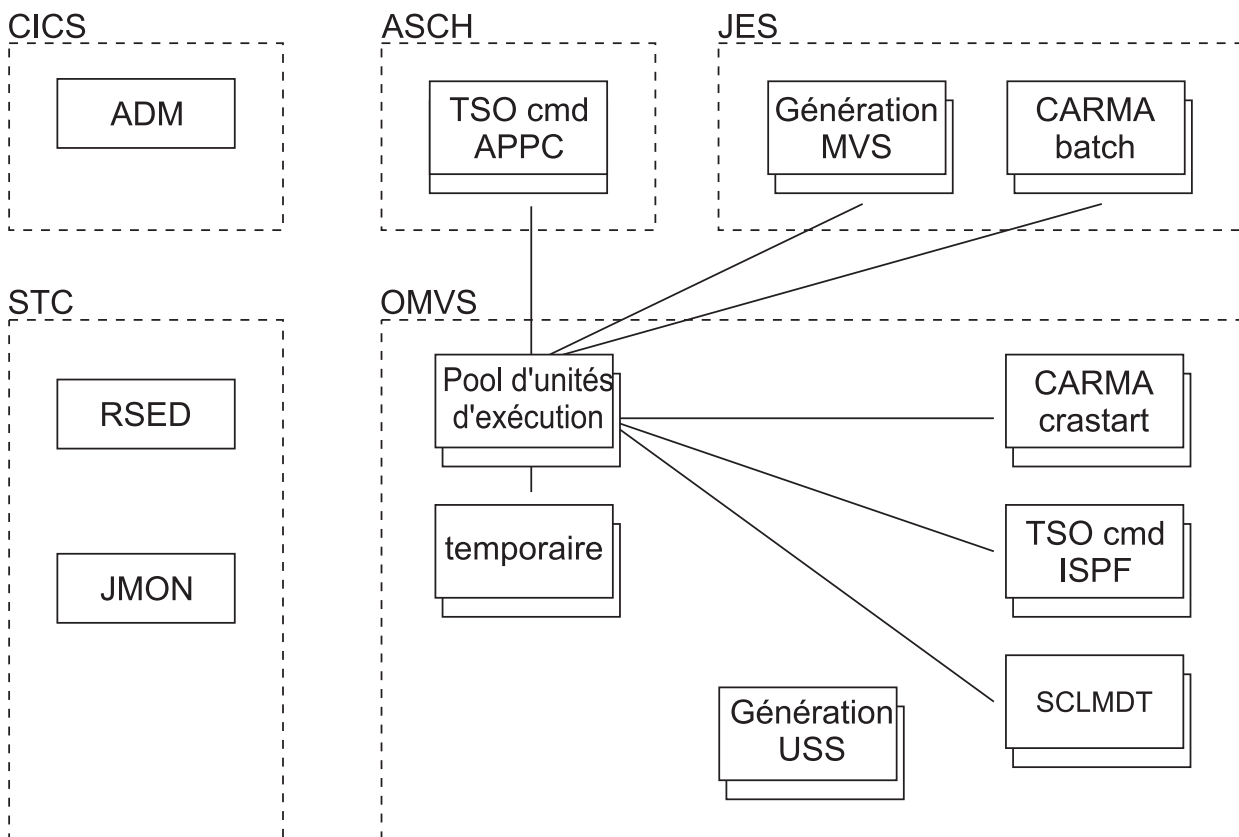


Figure 13. Classification WLM

La figure 13 montre la présentation de base des sous-systèmes par l'intermédiaire desquels les charges de travail de Developer for System z sont présentées au gestionnaire WLM.

ADM (Application Deployment Manager) est actif dans une région CICS et suivra donc les règles de classification CICS dans le gestionnaire WLM.

Le démon RSE (RSED) et le moniteur de travaux JES (JMON) sont des tâches démarrées de Developer for System z (ou des travaux par lots à exécution longue), chacun avec leur espace adresse individuel.

Comme nous l'avons documenté dans «RSE comme application Java», à la page 5, le démon RSE génère un processus enfant pour chaque serveur de pools d'unités d'exécution RSE (qui prend en charge un nombre variable de clients). Chaque pool d'unités d'exécution est actif dans un espace adresse distinct (à l'aide d'un initiateur z/OS UNIX, BPXAS). Puisqu'il s'agit de processus générés, leur classification s'effectue d'après les règles de classification WLM OMVS, mais pas selon les règles de classification des tâches démarrées.

Les clients qui sont actifs dans un pool d'unités d'exécution peuvent créer une multitude d'autres espaces adresse, selon les actions menées par les utilisateurs. Selon la configuration de Developer for System z, certaines charges de travail, comme un service de Commandes TSO (TSO cmd) ou CARMA, peuvent s'exécuter dans des sous-systèmes différents.

Les espaces adresse répertoriés dans la figure 13, à la page 71 restent dans le système suffisamment longtemps pour être visibles, mais sachez qu'en raison de la conception de z/OS UNIX, il existe aussi des espaces adresse temporaires de durée de vie courte. Ces espaces adresse temporaires sont actifs dans le sous-système OMVS.

Notez que tandis que les pools d'unités d'exécution utilisent le même ID utilisateur et un nom de travail similaire au démon RSE, tous les espaces adresse démarrés par un pool d'unités d'exécution appartiennent à l'ID utilisateur du client ayant demandé l'action. L'ID utilisateur du client est aussi utilisé comme (partie du) nom de travail pour tous les espaces adresse basés sur OMVS et déclarés par le pool d'unités d'exécution.

D'autres espaces adresse sont créés par d'autres services qu'utilise Developer for System z, tels que FMNCAS (File Manager) ou z/OS UNIX REXEC (génération USS).

## Règles de classification

WLM utilise des règles de classification pour mapper un travail entrant le système en une classe de service. Cette classification repose sur des qualificatifs de travaux. Le premier qualificatif (obligatoire) est le type de sous-système qui reçoit la demande de travail. Le tableau 13 répertorie les types de sous-systèmes qui peuvent recevoir des charges de travail de Developer for System z.

Tableau 13. Sous-systèmes de point d'entrée WLM

| Type de sous-système | Description du travail                                                                                                                                  |
|----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------|
| ASCH                 | Les demandes de travaux incluent tous les programmes de transactions APPC planifiés par le planificateur de transactions APPC/MVS fourni par IBM, ASCH. |
| CICS                 | Les demandes de travaux incluent toutes les transactions traitées par CICS.                                                                             |
| JES                  | Les demandes de travaux incluent tous les travaux initiés par JES2 ou JES3.                                                                             |

Tableau 13. Sous-systèmes de point d'entrée WLM (suite)

| Type de sous-système | Description du travail                                                                                                                                  |
|----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------|
| OMVS                 | Les demandes de travaux incluent un travail traité dans des espaces adresse enfant en parallèle à des services système z/OS UNIX.                       |
| STC                  | Les demandes de travaux incluent tous les travaux initiés par les commandes START et MOUNT. STC inclut aussi des espaces adresse de composants système. |

Le tableau 14 répertorie des qualificateurs supplémentaires que vous pouvez utiliser pour attribuer une charge de travail à une classe de service spécifique. Pour plus d'informations sur les qualificateurs répertoriés, voir *MVS Planning: Workload Management* (SA22-7602).

Tableau 14. Qualificateurs de travaux WLM

|     |                                         | ASCH | CICS | JES | OMVS | STC |
|-----|-----------------------------------------|------|------|-----|------|-----|
| AI  | Comptabilité des informations           | x    |      | x   | x    | x   |
| LU  | Nom de l'unité logique (*)              |      | x    |     |      |     |
| PF  | Effectuer (*)                           |      |      | x   |      | x   |
| PRI | Priorité                                |      |      | x   |      |     |
| SE  | Nom de l'environnement de planification |      |      | x   |      |     |
| SSC | Nom de collection du sous-système       |      |      | x   |      |     |
| SI  | Instance du sous-système (*)            |      | x    | x   |      |     |
| SPM | Paramètre du sous-système               |      |      |     |      | x   |
| PX  | Nom Sysplex                             | x    | x    | x   | x    | x   |
| SY  | Nom du système (*)                      | x    |      |     | x    | x   |
| TC  | Classe Transaction/travail (*)          | x    |      | x   |      |     |
| TN  | Nom Transaction/travail (*)             | x    | x    | x   | x    | x   |
| UI  | ID utilisateur (*)                      | x    | x    | x   | x    | x   |

**Remarque :** S'agissant des qualificateurs marqués avec (\*), vous pouvez indiquer des groupes de classification en ajoutant un G à l'abréviation du type. Par exemple, un groupe de nom de transaction doit être TNG.

## Définition des objectifs

Comme nous l'avons documenté dans «Classification des charges de travail», à la page 71, *Developer for System z* crée différents types de charges de travail sur votre système. Ces différentes tâches communiquent entre elles, ce qui implique que le temps écoulé réel devienne important pour éviter des problèmes de délai d'attente lors des connexions entre les tâches. En conséquence, une tâche *Developer for System z* doit être placée dans des classes de services de hautes performances avec une priorité élevée.

Une révision, et probablement une mise à jour, de vos objectifs WLM actuels est donc recommandée, notamment s'il agit de charges de travail OMVS critique en temps ou nouvelles des magasins MVS traditionnels.

**Remarque :**

- Les informations d'objectif qui figurent dans cette section sont délibérément maintenues à un niveau descriptif, car les objectifs de performances réels sont très spécifiques du site.
- Pour mieux comprendre l'impact d'une tâche spécifique sur votre système, nous employons des termes comme utilisation de ressources minimale, modérée et substantielle. Tous ces termes sont relatifs à l'utilisation de la totalité des ressources de Developer for System z proprement dit, et non du système dans son intégralité.

Le tableau 15 répertorie les espaces adresse utilisés par Developer for System z. z/OS UNIX remplace "x" dans la colonne "Nom de la tâche" par un nombre aléatoire comportant un seul chiffre.

*Tableau 15. Charges de travail WLM*

| Description                                              | Nom de la tâche                       | Charge de travail |
|----------------------------------------------------------|---------------------------------------|-------------------|
| Moniteur de travaux JES                                  | JMON                                  | STC               |
| Démon RSE                                                | RSED                                  | STC               |
| Pool d'unités d'exécution RSE                            | RSEDx                                 | OMVS              |
| Passerelle client ISPF (service Commandes TSO et SCLMDT) | <id utilisateur>x                     | OMVS              |
| Service Commandes TSO (APPC)                             | FEKFRSRV                              | ASCH              |
| CARMA (lot)                                              | CRA<port>                             | JES               |
| CARMA (crastart)                                         | <id utilisateur>x                     | OMVS              |
| CARMA (passerelle client ISPF)                           | <id utilisateur> et <id utilisateur>x | OMVS              |
| Génération MVS (travail par lots)                        | *                                     | JES               |
| Génération z/OS UNIX (commandes shell)                   | <id utilisateur>x                     | OMVS              |
| Interpréteur de commandes de z/OS UNIX                   | <id utilisateur>                      | OMVS              |
| Tâche File Manager                                       | <id utilisateur>x                     | OMVS              |
| Gestionnaire de déploiement d'application                | CICSTS                                | CICS              |

## Remarques relatives à la sélection des objectifs

Les remarques générales suivantes relatives à WLM peuvent vous aider à bien définir les définitions d'objectifs correctes pour Developer for System z :

- Vous devez baser des objectifs sur ce qui peut être réellement obtenu, et non sur vos souhaits concernant ce qui pourrait arriver. Si vous définissez des supérieurs à ce qui est nécessaire, WLM déplace des ressources d'un travail de moindre importance vers un travail d'importance plus élevée qui pourrait ne pas avoir véritablement besoin des ressources.
- Limite le volume de travail attribué aux classes de service SYSTEM et SYSSTC, car ces classes bénéficient d'une priorité de distribution supérieure à n'importe quelle classe gérée WLM. Utilisez ces classes pour un travail qui est d'importance élevée, bien qu'utilisant peu d'unité centrale.
- Un travail qui n'entre pas dans les règles de classification finit par aboutir dans la classe SYSOTHER, qui a un objectif discrétionnaire. Un objectif discrétionnaire recommande à WLM d'agir au mieux lorsque le système a des ressources disponibles.

Lors de l'utilisation des objectifs de temps de réponse :

- Il doit exister un taux d'arrivée stable de tâches (au moins 10 tâches en 20 minutes) pour permettre à WLM de gérer correctement un objectif de temps de réponse.
- Utilisez des objectifs de temps de réponse moyen uniquement pour bien contrôler des charges de travail, car une transaction longue et unique a un impact énorme sur le temps de réponse moyen et peut contraindre WLM à réagir de façon excessive.

Lors de l'utilisation des objectifs de vitesse :

- D'une manière générale, vous ne pouvez pas obtenir d'objectif de vitesse supérieur à 90 % et ce pour différentes raisons. Par exemple, tous les espaces adresse SYSTEM et SYSSTC bénéficient d'une priorité de distribution supérieure à tout objectif de type vitesse.
- WLM utilise un nombre minimum de modèles (utilisation et délai) sur lesquels se fondent ses décisions en termes d'objectifs de vitesse. Ainsi, moins il y aura de travaux exécutés dans une classe de service, plus cela prendra de temps pour collecter le nombre requis de modèles et ajuster la règle de répartition.
- Réévaluez les objectifs de vitesse lors du changement de votre matériel. Notamment, vers une diminution, des processeurs plus rapides imposent des changements dans les objectifs de vitesse.

## STC

Toutes les tâches démarrées de Developer for System z, démon RSE et moniteur de travaux JES, répondent à des demandes client en temps réel.

Tableau 16. Charges de travail et STC WLM

| Description             | Nom de la tâche | Charge de travail |
|-------------------------|-----------------|-------------------|
| Moniteur de travaux JES | JMON            | STC               |
| Démon RSE               | RSED            | STC               |

- Moniteur de travaux JES

Le moniteur de travaux JES fournit tous les services liés à JES comme la soumission de travaux, la consultation de fichiers spoule et l'exécution de commandes de l'opérateur JES. Vous devez indiquer un objectif de vitesse et de hautes performances sur une période, car la tâche ne signale pas les transactions individuelles à WLM. L'usage des ressources dépend fortement des actions des utilisateurs et donc fluctuera, mais on s'attend à être de type minimal à modéré.

- Démon RSE

Le démon RSE gère les connexions et authentification des clients ainsi que les différents pools d'unités d'exécution RSE. Vous devez indiquer un objectif de vitesse et de hautes performances sur une période, car la tâche ne signale pas les transactions individuelles à WLM. On s'attend à une utilisation des ressources de type modéré, avec un pic au début de la journée de travail.

## OMVS

Les charges de travail OMVS peuvent être réparties en deux groupes, les pool d'unités d'exécution RSE et tout le reste. Ceci parce qu'à l'exception des pools d'unités d'exécution, toutes les charges de travail utilisent l'ID utilisateur du client comme base pour le nom de l'espace adresse. (z/OS UNIX remplace "x" dans la colonne "Nom de la tâche" par un nombre aléatoire comportant un seul chiffre.)

Tableau 17. Charges de travail - OMVS WLM

| Description                                              | Nom de la tâche                       | Charge de travail |
|----------------------------------------------------------|---------------------------------------|-------------------|
| Pool d'unités d'exécution RSE                            | RSEDx                                 | OMVS              |
| Passerelle client ISPF (service Commandes TSO et SCLMDT) | <id utilisateur>x                     | OMVS              |
| CARMA (crastart)                                         | <id utilisateur>x                     | OMVS              |
| CARMA (passerelle client ISPF)                           | <id utilisateur> et <id utilisateur>x | OMVS              |
| Génération z/OS UNIX (commandes shell)                   | <id utilisateur>x                     | OMVS              |
| Interpréteur de commandes de z/OS UNIX                   | <id utilisateur>                      | OMVS              |
| Tâche File Manager                                       | <id utilisateur>x                     | OMVS              |

- Pool d'unités d'exécution RSE

Un pool d'unités d'exécution RSE est comme le coeur et le cerveau de Developer for System z. Presque toutes les données passent par là, tandis que les logiciels de fouille de données (unités d'exécution spécifiques de l'utilisateur) à l'intérieur du pool d'unités d'exécution contrôlent les actions de la plupart des autres tâches liées à Developer for System z. Vous devez indiquer un objectif de vitesse et de hautes performances sur une période, car la tâche ne signale pas les transactions individuelles à WLM. L'usage des ressources dépend fortement des actions des utilisateurs et donc fluctuera, mais on s'attend à être de type substantiel.

Les charges de travail restantes finiront toutes par aboutir dans la même classe de service en raison d'une convention commune d'attribution de nom d'espace adresse. Vous devez indiquer un objectif à périodes multiples pour cette classe de service. Les premières périodes doivent être des objectifs de temps de réponse percentiles à hautes performances, tandis que la dernière période doit avoir un objectif de vitesse à performances modérées. Certaines charges de travail, comme une passerelle client ISPF, signaleront des transactions individuelles et d'autres non.

- Passerelle client ISPF

La passerelle client ISPF est un service ISPF appelé par Developer for System z pour exécuter des commandes TSO et ISPF non-interactives. Ceci inclut des commandes explicites émises par le client ainsi que des commandes implicites émises par Developer for System z, comme l'obtention d'une liste de membres PDS. L'usage des ressources dépend fortement des actions des utilisateurs et donc fluctuera, mais on s'attend à être de type minimal.

- CARMA

CARMA est un serveur Developer for System z facultatif qui permet d'interagir avec des gestionnaires de configuration logicielle (SCM) basés sur l'hôte, comme CA Endevor SCM. Developer for System z autorise différentes méthodes de démarrage pour un serveur CARMA, dont certaines deviennent une charge de travail OMVS. L'usage des ressources dépend fortement des actions des utilisateurs et donc fluctuera, mais on s'attend à être de type minimal.

- Génération z/OS UNIX

Lorsqu'un client initie une génération pour un projet z/OS UNIX, z/OS UNIX REXEC (ou SSH) démarre une tâche qui exécute plusieurs commandes shell



z/OS UNIX pour effectuer la génération. L'utilisation des ressources dépend fortement des actions des utilisateurs et donc fluctuera, mais on s'attend à être de modéré à substantiel, selon la taille du projet.

- Interpréteur de commandes de z/OS UNIX

Cette charge de travail traite les commande shell z/OS UNIX shell émises par le client. L'usage des ressources dépend fortement des actions des utilisateurs et donc fluctuera, mais on s'attend à être de type minimal.

- IBM File Manager

Bien qu'il ne s'agisse pas d'espace adresse dans Developer for System z, les processus enfant File Manager générés sont répertoriés ici car ils peuvent être démarrés sur la demande d'un client Developer for System z client ; ces tâches utilisent la même convention d'attribution de nom que des tâches Developer for System z. Ces tâches File Manager traitent des actions de fichiers MVS complexes, comme une édition formatée d'un fichier VSAM. L'usage des ressources dépend fortement des actions des utilisateurs et donc fluctuera, mais on s'attend à être de type minimal à modéré.

## JES

Developer for System z utilise les processus de traitement par lots gérés par JES de différentes manières. L'usage le plus classique concerne les générations MVS dans lesquelles un travail est soumis et contrôlé pour déterminer quand il prend fin. Toutefois, Developer for System z pourrait aussi démarrer un serveur CARMA dans un traitement par lots et communiquer avec celui-ci via TCP/IP.

Tableau 18. Charge de travail - JES WLM

| Description                       | Nom de la tâche | Charge de travail |
|-----------------------------------|-----------------|-------------------|
| CARMA (lot)                       | CRA<port>       | JES               |
| Génération MVS (travail par lots) | *               | JES               |

- CARMA

CARMA est un serveur Developer for System z facultatif qui permet d'interagir avec des gestionnaires de configuration logicielle (SCM) basés sur l'hôte, comme CA Endevor SCM. Developer for System z autorise différentes méthodes de démarrage pour un serveur CARMA, dont certaines deviennent une charge de travail JES. Vous devez indiquer un objectif de vitesse et de hautes performances sur une période, car la tâche ne signale pas les transactions individuelles à WLM. L'usage des ressources dépend fortement des actions des utilisateurs et donc fluctuera, mais on s'attend à être de type minimal.

- Génération MVS

Lorsqu'un client initie une génération pour un projet MVS, Developer for System z doit démarrer une tâche en traitement par lots pour effectuer la génération. L'utilisation des ressources dépend fortement des actions des utilisateurs et donc fluctuera, mais on s'attend à être de modéré à substantiel, selon la taille du projet. Différentes stratégies d'objectifs à performances modérées peuvent être recommandées, selon des circonstances locales.

- Vous pourriez indiquer un objectif à périodes multiples avec une période à objectif de temps de réponse percentile et une période à objectif de vitesse secondaire. Dans ce cas, vos développeurs doivent utiliser pour la plupart la même procédure de génération et des fichiers d'entrée de tailles similaires pour créer des travaux ayant des temps de réponse uniformes. Il doit aussi

exister un taux d'arrivée stable de travaux (au moins 10 travaux en 20 minutes) pour permettre à WLM de gérer correctement un objectif de temps de réponse.

- Un objectif de vitesse est plus approprié à la plupart des travaux en traitement par lots, car ces objectifs peuvent gérer des temps d'exécution et des taux d'arrivée extrêmement variables.

## ASCH

Dans les versions actuelles de Developer for System z, la passerelle client ISPF permet l'exécution de commandes TSO et ISPF non-interactive. Pour des raisons historiques, Developer for System z prend également en charge l'exécution de ces commandes via une transaction APPC. Notez que la méthode APPC est obsolète.

Tableau 19. Charges de travail - ASCH WLM

| Description                  | Nom de la tâche | Charge de travail |
|------------------------------|-----------------|-------------------|
| Service Commandes TSO (APPC) | FEKFRSRV        | ASCH              |

- Service Commandes TSO

Developer for System z peut démarrer le service Commandes TSO peut être démarré comme une transaction APPC pour exécuter des commandes TSO et ISPF non-interactive. Ceci inclut des commandes explicites émises par le client ainsi que des commandes implicites émises par Developer for System z, comme l'obtention d'une liste de membres PDS. Vous devez indiquer un objectif à périodes multiples pour cette classe de service. Pour les premières périodes, vous devez indiquer des objectifs de temps de réponse percentiles de hautes performances. Pour la dernière période, vous devez indiquer un objectif de vitesse à performances modérées. L'usage des ressources dépend fortement des actions des utilisateurs et donc fluctuera, mais on s'attend à être de type minimal.

## CICS

Le gestionnaire de déploiement d'application est un serveur Developer for System z facultatif qui est actif au sein d'une région CICS Transaction Server.

Tableau 20. Charges de travail WLM - CICS

| Description                               | Nom de la tâche | Charge de travail |
|-------------------------------------------|-----------------|-------------------|
| Gestionnaire de déploiement d'application | CICSTS          | CICS              |

- Gestionnaire de déploiement d'application

Le serveur facultatif du gestionnaire de déploiement d'application qui est actif au sein d'une région CICSTS, vous permet de décharger en toute sécurité des tâches de gestion CICSTS pour les développeurs de logiciel. L'usage des ressources dépend fortement des actions des utilisateurs et donc fluctuera, mais on s'attend à être de type minimal. Le type de classe de service que vous devez utiliser dépend des autres transactions actives dans cette région CICS et il n'est donc pas abordé en détail.

Le gestionnaire WLM prend en charge plusieurs types de gestion que vous pouvez utiliser pour CICS :

- Gestion CICS vers un objectif de région

L'objectif est défini sur une classe de service qui gère des espaces adresse CICS. Vous ne pouvez utiliser qu'un objectif de vitesse d'exécution pour cette classe de service. Le gestionnaire WLM utilise les règles de classification JES ou STC pour les espaces adresse, mais pas les règles de classification de sous-système CICS pour les transactions.

- Gestion CICS vers un objectif de temps de réponse de transaction

Vous pouvez définir un objectif de temps de réponse dans une classe de service attribuée à une transaction unique ou à un groupe de transactions. Le gestionnaire WLM utilise les règles de classification JES ou STC pour les espaces adresse et les règles de classification de sous-système CICS pour les transactions.



---

## Chapitre 5. Remarques relatives à l'optimisation

Comme indiqué dans Chapitre 1, «Description de Developer for System z», à la page 3, RSE (Remote Systems Explorer, Explorateur de systèmes distants) est le coeur de Developer for System z. Pour gérer les connexions et charges de travail provenant des clients, RSE est composé d'un espace adresse de démon, qui permet de contrôler les espaces adresse du groupe d'unités d'exécution. Le démon agit comme un point focal pour la connexion et la gestion, alors que les pools d'unités d'exécution traitent les charges de travail du client.

RSE devient une cible privilégiée d'optimisation de la configuration de Developer for System z. Toutefois, la gestion de centaines d'utilisateurs, chacun utilisant au moins 17 unités d'exécution, d'une certaine quantité de mémoire et éventuellement d'un ou de plusieurs espaces adresse implique de configurer correctement Developer for System z et z/OS.

Les rubriques suivantes sont traitées dans le présent chapitre :

- «Utilisation des ressources»
- «Utilisation de l'espace de stockage», à la page 93
- «Utilisation de l'espace du système de fichiers z/OS UNIX», à la page 99
- «Définitions de ressources essentielles», à la page 102
- «Définitions de ressource différentes», à la page 106
- «Contrôle», à la page 108
- «Exemple de configuration», à la page 112

---

### Utilisation des ressources

Utilisez les informations présentées dans cette section pour estimer l'utilisation normale et optimale des ressources par Developer for System z, de manière à pouvoir planifier la configuration du système en conséquence.

Lors de l'utilisation des nombres et formules présentés dans cette section pour définir les valeurs des limites du système, n'oubliez pas que vous utilisez des estimations assez précises. Lors de la définition des limites du système, prévoyez une marge suffisante afin de permettre aux tâches temporaires, aux autres tâches ou aux utilisateurs se connectant plusieurs fois à l'hôte simultanément d'utiliser les ressources (au moyen, par exemple, de RSE et TN3270).

#### Remarque :

- Les informations sont limitées aux services accessibles par l'intermédiaire de RSE qui sont fournis par Developer for System z lui-même. Par exemple, l'utilisation des ressources de TN3270 (inaccessibles par l'intermédiaire de RSE) et des programmes appelés lors des générations à distance (basées sur l'hôte) des projets MVS ou z/OS UNIX (non fournis par Developer for System z) n'est pas documentée.
- Tout ajout d'extensions tierces à Developer for System z peut augmenter les compteurs d'utilisation des ressources.
- Tous les services comportent des tâches de "nettoyage" de courte durée, qui utilisent les ressources pendant leur exécution, et qui peuvent s'exécuter de

manière séquentielle ou parallèle les unes par rapport aux autres. Les ressources utilisées par ces tâches ne sont pas documentées.

- Lorsque cela s'avère utile, l'utilisation des ressources propres à l'utilisateur des logiciels requis (ISPF Client Gateway, par exemple) est documentée.
- Les nombres indiqués ici peuvent changer sans préavis.

## Présentation

Les tableaux ci-dessous présentent les nombres d'espaces adresse, de processus et d'unités d'exécution utilisés par Developer for System z. Plus d'informations sur les nombres présentés ici sont disponibles dans les sections suivantes :

- «Nombre d'espaces adresse», à la page 83
- «Nombre de processus», à la page 86
- «Nombre d'unités d'exécution», à la page 89

Le tableau 21 donne une présentation générale des ressources essentielles utilisées par les tâches démarrées Developer for System z. Ces ressources ne sont attribuées qu'une seule fois. Elles sont partagées par tous les clients Developer for System z.

*Tableau 21. Utilisation des ressources communes*

| Tâche démarrée | Espaces adresse | Processus | Unités d'exécution |
|----------------|-----------------|-----------|--------------------|
| JMON           | 1               | 1         | 3                  |
| RSED           | 1               | 3         | 11                 |
| RSEDx          | 1 + 1 (a)       | 1 + 2     | 1 + 10             |

**Remarque :** (a) Il existe 1 espace adresse avec des droits APF et au moins 1 espace adresse de pool d'unités d'exécution RSE actifs. Voir «Nombre d'espaces adresse», à la page 83 pour déterminer le nombre réel d'espaces adresse de pool d'unités d'exécution RSE.

Le tableau 22 donne une présentation générale des ressources essentielles utilisées par les logiciels requis. Ces ressources sont attribuées pour chaque client Developer for System z qui appelle la fonction associée.

*Tableau 22. Utilisation des ressources prérequis spécifiques de l'utilisateur*

| Logiciels requis       | Espaces adresse | Processus | Unités d'exécution |
|------------------------|-----------------|-----------|--------------------|
| Passerelle client ISPF | 1               | 2         | 4                  |
| APPC                   | 1               | 1         | 2                  |

Le tableau 23 donne une présentation générale des ressources essentielles utilisées par chaque client Developer for System z lors de l'exécution de la fonction indiquée. Les valeurs non numériques (ISPF, par exemple) font référence à la valeur correspondante du tableau 22.

*Tableau 23. Utilisation des ressources spécifiques de l'utilisateur*

| Action de l'utilisateur | Espaces adresse | Processus      |                | Unités d'exécution |      |
|-------------------------|-----------------|----------------|----------------|--------------------|------|
|                         | ID utilisateur  | ID utilisateur | ID utilisateur | RSEDx              | JMON |
| Connexion               | -               | -              | -              | 17                 | 1    |

Tableau 23. Utilisation des ressources spécifiques de l'utilisateur (suite)

| Action de l'utilisateur                  | Espaces adresse | Processus      | Unités d'exécution |       |      |
|------------------------------------------|-----------------|----------------|--------------------|-------|------|
|                                          | ID utilisateur  | ID utilisateur | ID utilisateur     | RSEDx | JMON |
| Temporisateur pour le délai d'inactivité | -               | -              | -                  | 1     | -    |
| Rechercher                               | -               | -              | -                  | 1     | -    |
| Développement de PDS(E)                  | ISPF            | ISPF           | ISPF               | -     | -    |
| Ouverture du fichier                     | ISPF            | ISPF           | ISPF               | 1     | -    |
| Commande TSO                             | ISPF            | ISPF           | ISPF               | -     | -    |
| Interpréteur de commandes de z/OS UNIX   | 1               | 1              | 1                  | 6     | -    |
| Génération MVS                           | 1               | -              | -                  | -     | -    |
| Génération z/OS UNIX                     | 3               | 3              | 3                  | -     | -    |
| CARMA (lot)                              | 1               | 1              | 2                  | 1     | -    |
| CARMA (crastart)                         | 1               | 1              | 2                  | 4     | -    |
| CARMA (ispf)                             | 4               | 4              | 7                  | 5     | -    |
| SCLMDT                                   | ISPF            | ISPF           | ISPF               | -     | -    |

**Remarque :** ISPF peut être remplacé par APPC, sauf pour SCLM Developer Toolkit.

## Nombre d'espaces adresse

Le tableau 24 répertorie les espaces adresse utilisés par Developer for System z, où la lettre "u" de la colonne "Nombre" indique que la quantité doit être multipliée par le nombre d'utilisateurs actifs simultanés de la fonction. z/OS UNIX remplace "x" de la colonne "Nom de la tâche" par un numéro à un seul chiffre aléatoire.

Tableau 24. Nombre d'espaces adresse

| Nombre | Description                                              | Nom de la tâche   | Partagé | Se termine après                           |
|--------|----------------------------------------------------------|-------------------|---------|--------------------------------------------|
| 1      | Moniteur de travaux JES                                  | JMON              | Oui     | Jamais                                     |
| 1      | Démon RSE                                                | RSED              | Oui     | Jamais                                     |
| 1      | Avec des droits APF RSE                                  | RSEDx             | Oui     | Jamais                                     |
| (a)    | Pool d'unités d'exécution RSE                            | RSEDx             | Oui     | Jamais                                     |
| lu     | Passerelle client ISPF (service Commandes TSO et SCLMDT) | <id utilisateur>x | Non     | 15 minutes ou déconnexion de l'utilisateur |
| lu     | Service Commandes TSO (APPC)                             | FEKFRSRV          | Non     | 60 minutes ou déconnexion de l'utilisateur |
| lu     | CARMA (lot)                                              | CRA<port>         | Non     | 7 minutes ou déconnexion de l'utilisateur  |
| lu     | CARMA (crastart)                                         | <id utilisateur>x | Non     | 7 minutes ou déconnexion de l'utilisateur  |

Tableau 24. Nombre d'espaces adresse (suite)

| Nombre | Description                                                           | Nom de la tâche                             | Partagé | Se termine après                          |
|--------|-----------------------------------------------------------------------|---------------------------------------------|---------|-------------------------------------------|
| 4u     | CARMA (ispf, obsolète)                                                | (1)<id utilisateur> ou (3)<id utilisateur>x | Non     | 7 minutes ou déconnexion de l'utilisateur |
| (b)    | Utilisation de la passerelle client ISPF simultanée par 1 utilisateur | <id utilisateur>x                           | Non     | Fin de la tâche                           |
| 1u     | Génération MVS (travail par lots)                                     | *                                           | Non     | Fin de la tâche                           |
| 3u     | Génération z/OS UNIX (commandes shell)                                | <id utilisateur>x                           | Non     | Fin de la tâche                           |
| 1u     | Interpréteur de commandes de z/OS UNIX                                | <id utilisateur>                            | Non     | Déconnexion de l'utilisateur              |

**Remarque :**

- (a) Il existe au moins un espace adresse de pool d'unités d'exécution RSE actif. Le nombre réel dépend :
  - de la directive `minimum.threadpool.process` de `rsed.envvars`. La valeur par défaut est 1.
  - du nombre d'utilisateurs qu'un pool d'unités d'exécution peut gérer. Les paramètres par défaut autorisent 30 utilisateurs par pool d'unités d'exécution.

**Remarque :** Si la directive `single.logon` est active, au moins deux pools d'unités d'exécution sont démarrés, même si la valeur 1 est attribuée à la directive `minimum.threadpool.process`. Le paramètre par défaut de la directive `single.logon` dans `rsed.envvars` est actif.

- (b) Developer for System z comporte plusieurs unités d'exécution actives par utilisateur. Si l'espace adresse de la passerelle client ISPF n'a pas terminé de gérer la demande d'une unité d'exécution lorsqu'une autre unité d'exécution envoie une requête, ISPF demande à une nouvelle passerelle client de traiter la nouvelle requête. Cet espace adresse se termine à l'issue de la tâche.
- SCLMDT requiert un espace adresse de passerelle client ISPF. SCLMDT partage l'espace adresse avec le service Commandes TSO.
- La plupart des actions liées au fichier MVS utilisent le service Commandes TSO, qui peut être actif dans la passerelle de client ISPF ou une transaction APPC, respectivement.

Utilisez la formule de la figure 14 pour estimer le nombre maximal d'espaces adresse utilisés par Developer for System z.

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

Figure 14. Nombre maximal d'espaces adresse

Où

- "3" correspond au nombre d'espaces adresse de serveur actif permanent.
- "A" représente le nombre d'espaces adresse de pool d'unités d'exécution RSE.
- "N" représente le nombre maximal d'utilisateurs simultanés.
- "x" est l'une des valeurs suivantes, selon les options de configuration sélectionnées.



| X | SCLMDT | TSO au moyen de la passerelle client | TSO au moyen d'APPC |
|---|--------|--------------------------------------|---------------------|
| 1 | Non    | Non                                  | Oui                 |
| 1 | Non    | Oui                                  | Non                 |
| 1 | Oui    | Oui                                  | Non                 |

- "y" est l'une des valeurs suivantes, selon les options de configuration sélectionnées.

| Y |                        |
|---|------------------------|
| 0 | Pas de CARMA           |
| 1 | CARMA (lot)            |
| 1 | CARMA (crastart)       |
| 4 | CARMA (ispf, obsolète) |

- Par défaut, la valeur de "z" est 0, mais elle peut augmenter selon les actions de l'utilisateur :
  - Ajoutez 1 lorsqu'une génération MVS est réalisée. Ces espaces adresse se terminent lorsque la tâche de génération connexe (un travail par lots) se termine.
  - Ajoutez 3 lorsqu'une génération z/OS UNIX est réalisée. Notez que le nombre réel peut être plus élevé, en fonction des besoins des programmes appelés. Ces espaces adresse se terminent à l'issue de la tâche de génération connexe.
- "2 + N\*0.01" permet d'ajouter une mémoire tampon aux espaces adresse temporaires. La taille de mémoire tampon requise peut différer en fonction du site.

Utilisez la formule de la figure 15 pour estimer le nombre maximal d'espaces adresse utilisés par un client Developer for System z (sans compter les espaces adresse temporaires non documentés).

$$x + y + z$$

Figure 15. Nombre d'espaces adresse par client

Où

- "x" dépend des options de configuration sélectionnées et est intégré dans la formule afin de calculer le nombre maximal d'espaces adresse (figure 14, à la page 84).
- "y" dépend des options de configuration sélectionnées et est intégré dans la formule afin de calculer le nombre maximal d'espaces adresse (figure 14, à la page 84).
- Par défaut, la valeur de "z" est 0, mais elle peut augmenter selon les actions de l'utilisateur, comme indiqué dans la formule permettant de calculer le nombre maximal d'espaces adresse (figure 14, à la page 84).

Les définitions du tableau 25, à la page 86 peuvent limiter le nombre réel d'espaces adresse.

Tableau 25. Limites d'espace adresse

| Adresse      | Limite                     | Ressources affectées                                                     |
|--------------|----------------------------|--------------------------------------------------------------------------|
| rsed.envvars | maximum.threadpool.process | Limite le nombre de pools d'unités d'exécution RSE                       |
| IEASYMxx     | MAXUSER                    | Limite le nombre d'espaces adresse                                       |
| ASCHPMxx     | MAX                        | Limite le nombre de demandeurs APPC pour le service Commandes TSO (APPC) |

## Nombre de processus

Le tableau 26 répertorie le nombre de processus par espace adresse utilisés par Developer for System z. La lettre "u" de la colonne "Espaces adresse" indique que la quantité doit être multipliée par le nombre d'utilisateurs actifs simultanés de la fonction.

Tableau 26. Nombre de processus

| Processus | Espaces adresse | Description                                              | ID utilisateur   |
|-----------|-----------------|----------------------------------------------------------|------------------|
| 1         | 1               | Moniteur de travaux JES                                  | STCJMON          |
| 3         | 1               | Démon RSE                                                | STCRSE           |
| 1         | 1               | Avec des droits APF RSE                                  | STCRSE           |
| 2         | (a)             | Pool d'unités d'exécution RSE                            | STCRSE           |
| 2         | (b)             | Passerelle client ISPF (service Commandes TSO et SCLMDT) | <id utilisateur> |
| 1         | 1u              | Service Commandes TSO (APPC)                             | <id utilisateur> |
| 1         | 1u              | CARMA (lot)                                              | <id utilisateur> |
| 1         | 1u              | CARMA (crastart)                                         | <id utilisateur> |
| 1         | 1u              | CARMA (ispf, obsolète)                                   | <id utilisateur> |
| 1         | 3u              | Génération z/OS UNIX (commandes shell)                   | <id utilisateur> |
| 1         | 1u              | Interpréteur de commandes de z/OS UNIX                   | <id utilisateur> |
| (5)       | (u)             | SCLM Developer Toolkit                                   | <id utilisateur> |

### Remarque :

- (a) Il existe au moins 1 espace adresse de pool d'unités d'exécution RSE actif. Voir «Nombre d'espaces adresse», à la page 83 pour déterminer le nombre réel d'espaces adresse de pool d'unités d'exécution RSE.
- Le démon RSE et tous les pools d'unités d'exécution RSE utilisent le même ID utilisateur.
- (b) Dans des situations normales, et lorsque les options de configuration par défaut sont utilisées, il y a une passerelle client ISPF active par utilisateur. Le nombre réel peut varier (voir «Nombre d'espaces adresse», à la page 83).

- SCLMDT requiert un espace adresse de passerelle client ISPF. SCLMDT partage l'espace adresse avec le service Commandes TSO.
- (u) Les processus SCLMDT s'exécutent dans l'espace adresse de la passerelle client ISPF. Par conséquent, elle ne porte aucune valeur de comptage d'espace adresse.
- Les processus SCLMDT sont temporaires et se terminent à l'issue de la tâche, mais plusieurs processus peuvent être actifs simultanément pour un seul utilisateur. Le tableau 26, à la page 86 répertorie le nombre maximal de processus SCLMDT simultanés.
- La plupart des actions liées au fichier MVS utilisent le service Commandes TSO, qui peut être actif dans la passerelle de client ISPF ou une transaction APPC, respectivement.
- Une génération z/OS UNIX utilise trois processus au total, chacun s'exécutant dans leur propre espace adresse.
- Tous les processus répertoriés restent actifs tant que l'espace adresse associé n'est pas terminé, sauf indication contraire.

Utilisez la formule de la figure 16 pour estimer le nombre maximal de processus utilisés par Developer for System z.

$$5 + 2 * A + N * (x + y + z) + (10 + N * 0.05)$$

Figure 16. Nombre maximal de processus

Où

- "5" correspond au nombre de processus utilisés par les espaces adresse de serveur actif permanent.
- "A" représente le nombre d'espaces adresse de pool d'unités d'exécution RSE.
- "N" représente le nombre maximal d'utilisateurs simultanés.
- "x" est l'une des valeurs suivantes, selon les options de configuration sélectionnées.

| X | SCLMDT | TSO au moyen de la passerelle client | TSO au moyen d'APPC |
|---|--------|--------------------------------------|---------------------|
| 1 | Non    | Non                                  | Oui                 |
| 2 | Non    | Oui                                  | Non                 |
| 7 | Oui    | Oui                                  | Non                 |

- "y" est l'une des valeurs suivantes, selon les options de configuration sélectionnées.

| Y |                        |
|---|------------------------|
| 0 | Pas de CARMA           |
| 1 | CARMA (lot)            |
| 1 | CARMA (crastart)       |
| 4 | CARMA (ispf, obsolète) |

- Par défaut, la valeur de "z" est 0, mais elle peut augmenter selon les actions de l'utilisateur :

- Ajoutez 1 lorsqu'un interpréteur de commandes z/OS UNIX est ouvert. Ce processus reste actif tant que l'utilisateur ne se déconnecte pas.
- Ajoutez 3 lorsqu'une génération z/OS UNIX est réalisée. Notez que le nombre réel peut être plus élevé, en fonction des besoins des programmes appelés. Ces processus se terminent à l'issue de la tâche de génération connexe.
- "10 + N\*0.05" permet d'ajouter une mémoire tampon pour les processus temporaires. La taille de mémoire tampon requise peut différer en fonction du site.

Utilisez la formule de la figure 17 pour estimer le nombre maximal de processus utilisés par STCRSE, l'ID utilisateur de la tâche démarrée RSED (sans compter les espaces adresse temporaires non documentés).

$$4 + 2 * A$$

Figure 17. Nombre de processus pour STCRSE

Où

- "4" correspond au nombre de processus utilisés par le démon RSE et les espaces adresse autorisés RSE APF.
- "A" représente le nombre d'espaces adresse de pool d'unité d'exécution RSE.

Utilisez la formule de la figure 18 pour estimer le nombre maximal de processus utilisés par le client Developer for System z (sans compter les processus temporaires non documentés).

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

Figure 18. Nombre de processus par client

Où

- "x" dépend des options de configuration sélectionnées et est intégré dans la formule afin de calculer le nombre maximal de processus (figure 16, à la page 87).
- "y" dépend des options de configuration sélectionnées et est intégré dans la formule afin de calculer le nombre maximal de processus (figure 16, à la page 87).
- Par défaut, la valeur de "z" est 0, mais elle peut augmenter selon les actions de l'utilisateur, comme indiqué dans la formule permettant de calculer le nombre maximal de processus (figure 16, à la page 87).
- "s" est égal à 1 si SCLM Developer Toolkit est utilisé. Il est égal à 0 dans le cas contraire.

Les définitions du tableau 27 peuvent limiter le nombre réel de processus.

Tableau 27. Limites de processus

| Adresse  | Limite      | Ressources affectées                            |
|----------|-------------|-------------------------------------------------|
| BPXPRMxx | MAXPROCSYS  | Limite le nombre de processus                   |
| BPXPRMxx | MAXPROCUSER | limite le nombre de processus par UID z/OS UNIX |

Remarque :

- Le démon RSE et les pools d'unités d'exécution RSE utilisent le même ID utilisateur. Etant donné que le démon RSE démarre un nouveau pool d'unités d'exécution à chaque fois que cela s'avère nécessaire, le nombre de processus associés à cet ID utilisateur peut augmenter. Par conséquent, la valeur attribuée à MAXPROCUSER doit permettre de s'adapter à cette augmentation, qui peut être formulée sous la forme "3 + 2\*A".
- La limite MAXPROCUSER est par ID utilisateur z/OS UNIX unique. Multipliez le nombre de processus par utilisateur estimé par le nombre de clients actifs simultanément si vos utilisateurs partagent le même UID.

## Nombre d'unités d'exécution

Le tableau 28 répertorie le nombre d'unités d'exécution utilisées par les fonctions sélectionnées de Developer for System z. La lettre "u" des colonnes "Unités d'exécution" indique que la quantité doit être multipliée par le nombre d'utilisateurs actifs simultanés de la fonction. Le nombre d'unités d'exécution est indiqué par processus, étant donné que les limites sont définies à ce niveau.

- RSEDx : ces unités d'exécution sont créées dans le pool d'unités d'exécution RSE, que plusieurs clients se partagent. Toutes les unités d'exécution se retrouvant dans le même pool d'unités d'exécution doivent être ajoutés pour obtenir le nombre total.
- Actif : ces unités d'exécution font partie intégrante du processus qui exécute la fonction demandée. Chaque processus est une unité autonome. Par conséquent, il n'est pas utile de faire la somme des unités d'exécution, même si elles sont associées au même ID utilisateur, sauf indication contraire.
- Amorce : les processus d'amorce sont indispensables au démarrage du processus réel. Chacun d'eux dispose d'une unité d'exécution, et il peut exister plusieurs amorces consécutives. Il n'est pas utile de faire la somme des unités d'exécution.

Tableau 28. Nombre d'unités d'exécution

| Unités d'exécution |        |        | ID utilisateur             | Description                                              |
|--------------------|--------|--------|----------------------------|----------------------------------------------------------|
| RSEDx              | Actif  | Amorce |                            |                                                          |
| -                  | 3 + 1u | -      | STCJMON                    | Moniteur de travaux JES                                  |
| -                  | 15     | 2      | STCRSE                     | Démon RSE                                                |
| -                  | 1      | -      | STCRSE                     | Avec des droits APF RSE                                  |
| 10 (a) + 17u       | -      | 1 (a)  | STCRSE                     | Pool d'unités d'exécution RSE                            |
| -                  | 4u (b) | 1u (b) | <id utilisateur>           | Passerelle client ISPF (service Commandes TSO et SCLMDT) |
| -                  | 2u     | -      | <id utilisateur>           | Service Commandes TSO (APPC)                             |
| 1u                 | 2u     | -      | STCRSE et <id utilisateur> | CARMA (lot)                                              |
| 4u                 | 2u     | -      | STCRSE et <id utilisateur> | CARMA (crastart)                                         |

Tableau 28. Nombre d'unités d'exécution (suite)

| Unités d'exécution |        |    | ID utilisateur                   | Description                                  |
|--------------------|--------|----|----------------------------------|----------------------------------------------|
| 5u                 | 4u     | 3u | STCRSE et<br><id<br>utilisateur> | CARMA (ispf,<br>obsolète)                    |
| -                  | 1u (c) | 2u | <id<br>utilisateur>              | Génération z/OS<br>UNIX (commandes<br>shell) |
| 6u                 | 1u     | -  | STCRSE et<br><id<br>utilisateur> | Interpréteur de<br>commandes de<br>z/OS UNIX |
| 1 (d)              | -      | -  | STCRSE                           | Télécharger                                  |
| 1 (e)              | -      | -  | STCRSE                           | Rechercher                                   |
| -                  | (5)    | -  | <id<br>utilisateur>              | SCLM Developer<br>Toolkit                    |
| 1u                 | -      | -  | STCRSE                           | Temporisateur pour<br>le délai d'inactivité  |

**Remarque :**

- (a) Il existe au moins 1 espace adresse de pool d'unités d'exécution RSE actif. Voir «Nombre d'espaces adresse», à la page 83 pour déterminer le nombre réel d'espaces adresse de pool d'unités d'exécution RSE.
- (b) Dans des situations normales, et lorsque les options de configuration par défaut sont utilisées, il y a une passerelle client ISPF active par utilisateur. Le nombre réel peut varier (voir «Nombre d'espaces adresse», à la page 83).
- SCLMDT requiert un espace adresse de passerelle client ISPF. SCLMDT partage l'espace adresse avec le service Commandes TSO.
- Selon l'action sélectionnée, SCLMDT peut utiliser plusieurs processus à une seule unité d'exécution se terminant à l'issue de la tâche. Le tableau 28, à la page 89 répertorie le nombre maximal d'unités d'exécution SCLMDT simultanées.
- La plupart des actions liées au fichier MVS utilisent le service Commandes TSO, qui peut être actif dans la passerelle de client ISPF ou une transaction APPC, respectivement.
- (c) Une génération z/OS UNIX appelle différents utilitaires de génération, qui peuvent comporter plusieurs unités d'exécutions. Le tableau 28, à la page 89 répertorie le nombre minimal d'unités d'exécution de génération z/OS UNIX simultanées.
- (d) Chaque téléchargement de données hôte utilise une unité d'exécution distincte. Cette unité d'exécution prend fin lorsque les données sont transférées au client.
- (e) Chaque recherche distante utilise une unité d'exécution distincte. Cette unité d'exécution prend fin lorsque les résultats sont transférés au client.
- Toutes les unités d'exécution répertoriées restent actives tant que le processus associé n'est pas terminé, sauf indication contraire.
- Le nombre d'unités d'exécution normal pour le code avec des droits APF RSE est 1. Toutefois, au cours du démarrage, il existe au moins 13 unités d'exécution temporairement actives en même temps.

Utilisez la formule de la figure 19 pour estimer le nombre maximal d'unités d'exécution utilisées par un pool d'unités d'exécution RSE. Utilisez la formule de la figure 20 pour estimer le nombre maximal d'unités d'exécution utilisées par le moniteur de travaux JES.

$$10 + N*(17 + x + y + z) + (20 + N*0.1)$$

Figure 19. Nombre maximal d'unités d'exécution du pool d'unités d'exécution RSE

$$3 + N$$

Figure 20. Nombre maximal d'unités d'exécution du moniteur de travaux JES

Où

- "N" représente le nombre maximum d'utilisateurs concurrents dans ce pool d'unités d'exécution ou ce moniteur de travaux JES. Les paramètres par défaut autorisent 30 utilisateurs par pool d'unités d'exécution.
- "x" est l'une des valeurs suivantes, selon les options de configuration sélectionnées.

| X | SCLMDT | TSO au moyen de la passerelle client | TSO au moyen d'APPC | Délai d'attente |
|---|--------|--------------------------------------|---------------------|-----------------|
| 0 | Non    | Non                                  | Oui                 | Non             |
| 0 | Non    | Oui                                  | Non                 | Non             |
| 0 | Oui    | Oui                                  | Non                 | Non             |
| 1 | Non    | Non                                  | Oui                 | Oui             |
| 1 | Non    | Oui                                  | Non                 | Oui             |
| 1 | Oui    | Oui                                  | Non                 | Oui             |

- "y" est l'une des valeurs suivantes, selon les options de configuration sélectionnées.

| Y |                        |
|---|------------------------|
| 0 | Pas de CARMA           |
| 1 | CARMA (lot)            |
| 4 | CARMA (crastart)       |
| 5 | CARMA (ispf, obsolète) |

- Par défaut, la valeur de "z" est 0, mais elle peut augmenter selon les actions de l'utilisateur :
  - Ajoutez 6 lorsqu'un interpréteur de commandes z/OS UNIX est ouvert. Ces unités d'exécution restent actives tant que l'utilisateur ne se déconnecte pas.
- "20 + N\*0.1" permet d'ajouter une mémoire tampon pour des unités d'exécution temporaires. La taille de mémoire tampon requise peut différer en fonction du site. Les téléchargements et recherches multiples et simultanés sont des exemples dans lesquels une augmentation de la taille de mémoire tampon peut être requise.

Les définitions du tableau 29 peuvent limiter le nombre réel d'unités d'exécution d'un processus, qui est en général important pour les pools d'unités d'exécution RSE.

Tableau 29. Limites d'unités d'exécution

| Adresse      | Limite          | Ressources affectées                                                                                                                                                      |
|--------------|-----------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| BPXPRMxx     | MAXTHREADS      | Limite le nombre d'unités d'exécution d'un processus.                                                                                                                     |
| BPXPRMxx     | MAXTHREADTASKS  | Limite le nombre de tâches MVS d'un processus.                                                                                                                            |
| BPXPRMxx     | MAXASSIZE       | Limite la taille d'espace adresse, et donc la mémoire disponible pour les blocs de contrôle liés à l'unité d'exécution.                                                   |
| rsed.envvars | Xmx             | Définit la taille maximale de segment de mémoire Java. Cette mémoire est réservée. Elle n'est donc plus disponible pour les blocs de contrôle liés à l'unité d'exécution. |
| rsed.envvars | maximum.clients | Limite le nombre de clients (et donc leurs unités d'exécution) dans un pool d'unités d'exécution RSE.                                                                     |
| rsed.envvars | maximum.threads | Limite le nombre d'unités d'exécution client dans un pool d'unités d'exécution RSE.                                                                                       |
| FEJJCNFG     | MAX_THREADS     | Limite le nombre d'unités d'exécution dans le moniteur de travaux JES.                                                                                                    |

**Remarque :**

- La valeur de maximum.threads dans rsed.envvars doit être inférieure à celle de MAXTHREADS et MAXTHREADTASKS dans BPXPRMxx.
- La commande de l'opérateur **DISPLAY PROCESS,CPU**, qui présente les unités d'exécution actives d'un pool d'unités d'exécution, est limitée à l'affichage des 4000 premières unités d'exécution.

## Utilisation des ressources temporaires

L'utilisation des ressources documentée dans les sections précédentes est permanente pour le cycle de vie de Developer for System z ou semi-permanente pour certaines tâches spécifiques des utilisateurs.

Cependant, Developer for System z utilise temporairement des ressources supplémentaires pour des tâches de nettoyage et pour satisfaire les demandes suivantes :

- Le traitement d'un événement de fichier de contrôle (directive audit.action dans rsed.envvars) utilise une unité d'exécution supplémentaire, un processus supplémentaire et éventuellement (si audit.action.id est défini) un espace adresse supplémentaire.
- Le traitement d'un événement de connexion (directive logon.action dans rsed.envvars) utilise une unité d'exécution supplémentaire, un processus supplémentaire et éventuellement (si logon.action.id est défini) un espace adresse supplémentaire.
- La commande opérateur IVP PASSTICKET utilisera des unités d'exécution supplémentaires.
- La commande opérateur IVP DAEMON utilisera une unité d'exécution supplémentaire, un processus supplémentaire et un espace adresse supplémentaire.



- La commande opérateur IVP ISPF utilisera une unité d'exécution supplémentaire, un processus supplémentaire et un espace adresse supplémentaire, plus les ressources utilisées par la passerelle client ISPF.

---

## Utilisation de l'espace de stockage

RSE est une application Java, ce qui signifie que l'utilisation de l'espace de stockage (mémoire) de Developer for System z doit s'appuyer sur deux limites d'allocation de mémoire : la taille de pile Java et la taille de l'espace adresse.

### Limite de taille de pile Java

Java offre de nombreux services visant à faciliter le codage des applications Java. L'un de ces services est la gestion de l'espace de stockage.

La gestion de l'espace de stockage Java alloue des blocs d'espace de stockage volumineux et les utilise pour satisfaire les demandes d'espace de stockage de l'application. Cet espace de stockage géré par Java est appelé pile Java. La récupération régulière de place (défragmentation) permet de récupérer l'espace inutilisé dans la pile et de réduire sa taille. Notez que pour économiser des cycles d'unité centrale, l'opération de récupération de place attend généralement que la mémoire occupée soit réellement nécessaire pour s'exécuter, laissant ainsi allouée la mémoire qui n'est plus utilisée (et rejetée) pour une durée plus longue que celle requise.

La taille de pile Java maximale est définie dans `rsed.envvars` avec la directive `Xmx`. Si cette directive n'est pas spécifiée, Java utilise une taille par défaut de 512 Mo. Indiquez une valeur de 256 Mo ou supérieure. En mode 64 bits, Java tente d'allouer un segment de mémoire au dessus de 2 Go, libérant ainsi l'espace en-deçà de ce seuil.

Chaque pool d'unités d'exécution RSE (qui gère les actions du client) est une application Java distincte et possède donc une pile Java personnelle. Notez que tous les pools d'unités d'exécution utilisent le même fichier de configuration `rsed.envvars` et donc la même limite de taille de pile Java.

L'utilisation du pool d'unités d'exécution de la pile Java dépend fortement des actions des clients connectés. Il est nécessaire de surveiller régulièrement l'utilisation de la pile pour définir la limite de taille de pile optimale. Utilisez la commande de l'opérateur **modify display process** pour surveiller l'utilisation de la pile Java par les pools d'unités d'exécution RSE.

### Limite de la taille d'espace adresse

Toutes les applications z/OS, y compris les applications Java, sont actives dans un espace adresse et sont donc liées par les limites de taille de l'espace adresse.

La taille d'espace adresse souhaitée est spécifiée au démarrage (avec le paramètre `REGION` de JCL, par exemple). Toutefois, les caractéristiques du système peuvent limiter la taille d'espace adresse réelle. Voir «Taille d'espace adresse», à la page 187 pour en savoir plus sur ces limites.

- `MAXASSIZE` de `SYS1.PARMLIB(BPXPRMxx)`
- `ASSIZEMAX` du segment OMVS de l'ID utilisateur attribué à la tâche démarrée
- sorties du système `IEFUSI` et `IEALIMIT`
- `MEMLIMIT` dans `SYS1.PARMLIB(SMFPRMxx)` pour le mode d'adressage 64 bits

Les pools d'unités d'exécution RSE héritent des limites de taille d'espace adresse provenant du démon RSE. La taille d'espace adresse doit être suffisante pour héberger la pile Java, Java lui-même, les zones de mémoire communes et tous les blocs de contrôle que le système crée pour prendre en charge l'activité du pool d'unités d'exécution (un bloc de contrôle des tâches par unité d'exécution, par exemple). Notez que certaines de ces utilisations de l'espace de stockage usage est inférieure à la ligne 16 Mo. En mode 64 bits, Java tente d'allouer un segment de mémoire au dessus de 2 Go, libérant ainsi l'espace en-deçà de ce seuil.

Il est recommandé de surveiller la taille réelle de l'espace adresse avant de modifier les paramètres qui l'influencent (la modification de la taille de la pile Java ou le nombre d'utilisateurs pris en charge par un seul pool d'unités d'exécution, par exemple). Utilisez les logiciels de surveillance du système pour suivre l'utilisation réelle de l'espace de stockage par Developer for system z. Si vous ne disposez pas de ce type d'outil, vous pouvez utiliser des outils comme la vue SDSF DA ou TASID (un outil d'informations système en l'état disponible sur la page Web ISPF "Support and downloads") afin de rassembler des informations de base.

## Instructions relatives à l'évaluation de la taille

Comme indiqué précédemment, l'utilisation réelle de l'espace de stockage par Developer for system z est fortement influencée par l'activité de l'utilisateur. Certaines actions utilisent une quantité fixe d'espace de stockage (connexion, par exemple), d'autres étant variables (liste des fichiers avec un qualificatif de haut niveau spécifié, par exemple).

- Utilisez un espace adresse de 2 Go pour RSE afin d'attribuer de l'espace pour la pile Java et tous les blocs de contrôle du système.
- En mode 64 bits, assurez-vous que la mémoire au-delà du seuil de 2 Go est réellement disponible sur RSE.
- Pour plus d'informations sur la configuration d'une limite de la taille de l'espace adresse, voir «Taille d'espace adresse», à la page 187.
- L'exemple de configuration `rsed.envvars` autorise 30 utilisateurs par pool d'unités d'exécution.
  - `maximum.clients=30`
  - `maximum.threads=520` ( $10+17*30 = 520$ , la valeur 520 permet donc l'utilisation de 30 clients)
- L'exemple de configuration `rsed.envvars` permet au segment de mémoire Java d'atteindre jusqu'à 512 Mo , à savoir 30 clients utilisant en moyenne 17 Mo par client ( $30*17 = 510$ ).

Notez que le message de console FEK004I de RSE affiche la limite en cours de la taille de la pile Java et de l'espace adresse lors du démarrage.

Utilisez l'un des scénarios suivants si la surveillance montre que la taille de pile Java est insuffisante comparée à la charge de travail réelle :

- Augmentez la taille maximale de la pile Java avec la directive `Xmx` dans `rsed.envvars`. Mais auparavant, vérifiez que l'espace adresse est suffisant pour l'augmentation de la taille.
- Diminuez le nombre de clients par pool d'unités d'exécution avec la directive `maximum.clients` dans `rsed.envvars`. RSE prend toujours en charge le même nombre de clients, lesquels sont répartis entre plusieurs pools d'unités d'exécution.

Le tableau 30 présente les valeurs utilisées par les clients réels de Developer for System z pour les paramètres clés `rsed.envvars` ayant une incidence sur l'utilisation de la mémoire.

*Tableau 30. Paramètres de référence pour l'utilisation de mémoire*

| <b>xmx (segment de mémoire java max)</b> | <b>Nombre maximum de clients</b> | <b>Type principal de développement</b> |
|------------------------------------------|----------------------------------|----------------------------------------|
| 512 M                                    | 30                               | PL/I                                   |
| 512 M                                    | 10                               | COBOL                                  |
| 384 M                                    | 12                               | COBOL                                  |
| 800 M (64 bits)                          | 20                               | Non spécifié                           |

## Exemple d'analyse de l'utilisation de l'espace de stockage

Les écrans présentés dans les figures ci-dessous montrent des exemples de valeurs d'utilisation de ressources pour une configuration par défaut de Developer for System z reflétant ces modifications.

- `single.logon` est désactivé pour empêcher RSE de créer au moins deux espaces adresse de pool d'unités d'exécution
- La taille maximale de la pile Java est de 10 Mo, une petite valeur maximale donnant lieu à un centile plus important d'utilisation et à des limites de taille de la pile atteintes plus tôt.

Max Heap Size=10MB and private AS Size=1,959MB

startup

BPXM023I (STCRSE)  
ProcessId(268 ) Memory Usage(7%) Clients(0)

| Jobname | Cpu time | Storage | EXCP  |
|---------|----------|---------|-------|
| JMON    | 0.01     | 2740    | 72    |
| RSED    | 4.47     | 32.8M   | 15910 |
| RSED8   | 1.15     | 27.4M   | 12612 |

logon 1

BPXM023I (STCRSE)  
ProcessId(268 ) Memory Usage(13%) Clients(1)

| Jobname | Cpu time | Storage | EXCP  |
|---------|----------|---------|-------|
| JMON    | 0.01     | 2864    | 81    |
| RSED    | 4.55     | 32.8M   | 15980 |
| RSED8   | 3.72     | 55.9M   | 24128 |

logon 2

BPXM023I (STCRSE)  
ProcessId(268 ) Memory Usage(23%) Clients(2)

| Jobname | Cpu time | Storage | EXCP  |
|---------|----------|---------|-------|
| JMON    | 0.02     | 2944    | 86    |
| RSED    | 4.58     | 32.9M   | 16027 |
| RSED8   | 4.20     | 57.8M   | 25205 |

logon 3

BPXM023I (STCRSE)  
ProcessId(268 ) Memory Usage(37%) Clients(3)

| Jobname | Cpu time | Storage | EXCP  |
|---------|----------|---------|-------|
| JMON    | 0.02     | 3020    | 91    |
| RSED    | 4.60     | 32.9M   | 16076 |
| RSED8   | 4.51     | 59.6M   | 26327 |

logon 4

BPXM023I (STCRSE)  
ProcessId(268 ) Memory Usage(41%) Clients(4)

| Jobname | Cpu time | Storage | EXCP  |
|---------|----------|---------|-------|
| JMON    | 0.02     | 3108    | 96    |
| RSED    | 4.61     | 32.9M   | 16125 |
| RSED8   | 4.77     | 62.3M   | 27404 |

Figure 21. Utilisation des ressources avec 5 connexions

logon 5

```
BPXM023I (STCRSE)
ProcessId(268) Memory Usage(41%) Clients(4)
ProcessId(33554706) Memory Usage(13%) Clients(1)
```

| Jobname | Cpu time | Storage | EXCP  |
|---------|----------|---------|-------|
| JMON    | 0.03     | 3184    | 101   |
| RSED    | 4.64     | 32.9M   | 16229 |
| RSED8   | 4.78     | 62.4M   | 27413 |
| RSED9   | 4.60     | 56.6M   | 24065 |

Figure 22. Utilisation des ressources avec 5 connexions (suite)

La figure 21, à la page 96 et la figure 22 illustrent un scénario dans lequel 5 clients se connectent à un démon RSE avec un pile Java de 10 Mo.

- Un pool d'unités d'exécution (RSED8) est un état dormant au démarrage, utilisant environ 27 Mo, dont 0,7 Mo se trouvent dans la pile Java (7 % de 10 Mo).
- Le pool d'unités d'exécution devient actif lorsque le premier client se connecte, utilisant 27 Mo plus 2 Mo pour chaque client qui se connecte.
- Une partie de ces 2 Mo par connexion se trouve dans la pile Java, comme l'illustre l'augmentation de l'utilisation de la pile.
- Toutefois, il n'existe pas de modèle d'utilisation de la pile, car cela dépend des mécanismes Java qui évaluent l'espace requis et en allouent plus que nécessaire. La récupération intermittente de place permet de libérer de l'espace de stockage, ce qui rend les tendances plus difficiles à détecter.
- Les mécanismes internes qui limitent le nombre de connexions par pool d'unités d'exécution pour garantir une taille de pile suffisante pour les unités d'exécutions actives obligent la cinquième connexion à être créée dans un nouveau pool d'unités d'exécution (RSED9). En principe, ces réseaux de sécurité interne ne sont pas appelés lorsqu'une installation est correctement configurée, car d'autres limites seraient atteintes en premier (la plus probable étant `maximum.clients` de `rsed.envvars`).

Max Heap Size=10MB and private AS Size=1,959MB

startup

BPXM023I (STCRSE)  
ProcessId(212 ) Memory Usage(7%) Clients(0)

| Jobname | Cpu time | Storage | EXCP  |
|---------|----------|---------|-------|
| JMON    | 0.01     | 2736    | 71    |
| RSED    | 4.35     | 32.9M   | 15117 |
| RSED8   | 1.43     | 27.4M   | 12609 |

logon

BPXM023I (STCRSE)  
ProcessId(212 ) Memory Usage(13%) Clients(1)

| Jobname | Cpu time | Storage | EXCP  |
|---------|----------|---------|-------|
| JMON    | 0.01     | 2864    | 80    |
| RSED    | 4.48     | 33.0M   | 15187 |
| RSED8   | 3.53     | 53.9M   | 24125 |

expand large MVS tree (195 data sets)

BPXM023I (STCRSE)  
ProcessId(212 ) Memory Usage(13%) Clients(1)

| Jobname | Cpu time | Storage | EXCP  |
|---------|----------|---------|-------|
| JMON    | 0.01     | 2864    | 80    |
| RSED    | 4.58     | 33.1M   | 16094 |
| RSED8   | 4.28     | 56.1M   | 24740 |

expand small PDS (21 members)

BPXM023I (STCRSE)  
ProcessId(212 ) Memory Usage(13%) Clients(1)

| Jobname  | Cpu time | Storage | EXCP  |
|----------|----------|---------|-------|
| IBMUSER2 | 0.22     | 2644    | 870   |
| JMON     | 0.01     | 2864    | 80    |
| RSED     | 4.61     | 33.1M   | 16108 |
| RSED8    | 4.40     | 56.2M   | 24937 |

open medium sized member (86 lines)

BPXM023I (STCRSE)  
ProcessId(212 ) Memory Usage(13%) Clients(1)

| Jobname  | Cpu time | Storage | EXCP  |
|----------|----------|---------|-------|
| IBMUSER2 | 0.22     | 2644    | 870   |
| JMON     | 0.01     | 2864    | 80    |
| RSED     | 4.61     | 33.1M   | 16108 |
| RSED8    | 8.12     | 62.7M   | 27044 |

Figure 23. Utilisation des ressources lors de l'édition d'un membre PDS

La figure 23 illustre un scénario dans lequel un client se connecte au démon RSE avec un pile Java de 10 Mo, puis édite un membre PDS.

- La recherche de catalogue qui a généré 195 noms de fichier a utilisé environ 2 Mo d'espace de stockage, en raison de l'activité du système, car l'utilisation de la pile Java n'augmente pas.

- L'ouverture d'un fichier PDS de 21 membres utilise à peine la mémoire du pool d'unités d'exécution, mais l'affichage montre que le service Commandes TSO a été appelé. Un nouvel espace adresse est actif (IBMUSER2) et utilise la taille de région attribuée à cet ID utilisateur dans TSO. Cet espace adresse reste actif pendant une durée spécifiée de manière à pouvoir être de nouveau utilisé pour de futures requêtes par le service Commandes TSO.
- L'ouverture d'un membre montre des numéros analogues au cours du développement d'un qualificatif de haut niveau. L'utilisation de la pile Java ne change pas, mais l'espace de stockage augmente de 6,5 Mo en raison de l'activité du système.

---

## Utilisation de l'espace du système de fichiers z/OS UNIX

La plupart des données liées à Developer for System z qui ne sont pas écrites dans une instruction de définition de données sont placées dans un fichier z/OS UNIX. Le programmeur système peut décider des données écrites et de leur destination. Toutefois, il ne contrôle pas la quantité de données écrites.

Les données peuvent être regroupées dans les catégories suivantes :

- Analyse du problème (fichiers journaux et fichiers de vidage système), présentée en détails dans le Chapitre 12, «Traitement des incidents liés à la configuration», à la page 171
- Contrôle (voir «Consignation dans le journal d'audit», à la page 24)
- Métadonnées d'envoi au client, comme indiqué dans «Métadonnées d'envoi au client», à la page 127.
- Données temporaires

Developer for System z écrit les journaux de l'hôte associé à RSE dans les répertoires z/OS UNIX suivants (voir Chapitre 12, «Traitement des incidents liés à la configuration», à la page 171) :

- /var/rdz/logs pour les fichiers journaux de la tâche démarrée RSE
- /var/rdz/logs/\$LOGNAME pour les journaux utilisateur

Par défaut, seuls les erreurs et messages d'avertissement sont consignés dans les fichiers journaux. Ainsi, si tout se passe comme prévu, ces répertoires ne contiennent que des fichiers vides ou presque vides (sans compter les journaux d'audit).

Vous pouvez activer la consignation des messages d'information (de préférence avec l'aide du point de service IBM), ce qui augmente sensiblement la taille des fichiers journaux.

```

startup

$ ls -l /var/rdz/logs
total 144
-rw-rw-rw- 1 STCRSE STCGRP 33642 Jul 10 12:10 rsedaemon.log
-rw-rw-rw- 1 STCRSE STCGRP 1442 Jul 10 12:10 rseserver.log

logon

$ ls -l /var/rdz/logs
total 144
drwxrwxrwx 3 IBMUSER SYS1 8192 Jul 10 12:11 IBMUSER
-rw-rw-rw- 1 STCRSE STCGRP 36655 Jul 10 12:11 rsedaemon.log
-rw-rw-rw- 1 STCRSE STCGRP 1893 Jul 10 12:11 rseserver.log
$ ls -l /var/rdz/logs/IBMUSER
total 160
-rw-rw-rw- 1 IBMUSER SYS1 3459 Jul 10 12:11 ffs.log
-rw-rw-rw- 1 IBMUSER SYS1 0 Jul 10 12:11 ffsget.log
-rw-rw-rw- 1 IBMUSER SYS1 0 Jul 10 12:11 ffsput.log
-rw-rw-rw- 1 IBMUSER SYS1 303 Jul 10 12:11 lock.log
-rw-rw-rw- 1 IBMUSER SYS1 126 Jul 10 12:11 rmt_classloader_cache.jar
-rw-rw-rw- 1 IBMUSER SYS1 7266 Jul 10 12:11 rsecomm.log
-rw-rw-rw- 1 IBMUSER SYS1 0 Jul 10 12:11 stderr.log
-rw-rw-rw- 1 IBMUSER SYS1 0 Jul 10 12:11 stdout.log

logoff

$ ls -l /var/rdz/logs
total 80
drwxrwxrwx 3 IBMUSER SYS1 8192 Jul 10 12:11 IBMUSER
-rw-rw-rw- 1 STCRSE STCGRP 36655 Jul 10 12:11 rsedaemon.log
-rw-rw-rw- 1 STCRSE STCGRP 2208 Jul 10 12:11 rseserver.log
$ ls -l /var/rdz/logs/IBMUSER
total 296
-rw-rw-rw- 1 IBMUSER SYS1 6393 Jul 10 12:11 ffs.log
-rw-rw-rw- 1 IBMUSER SYS1 0 Jul 10 12:11 ffsget.log
-rw-rw-rw- 1 IBMUSER SYS1 0 Jul 10 12:11 ffsput.log
-rw-rw-rw- 1 IBMUSER SYS1 609 Jul 10 12:11 lock.log
-rw-rw-rw- 1 IBMUSER SYS1 126 Jul 10 12:11 rmt_classloader_cache.jar
-rw-rw-rw- 1 IBMUSER SYS1 45157 Jul 10 12:11 rsecomm.log
-rw-rw-rw- 1 IBMUSER SYS1 0 Jul 10 12:11 stderr.log
-rw-rw-rw- 1 IBMUSER SYS1 176 Jul 10 12:11 stdout.log

stop

$ ls -l /var/rdz/logs
total 80
drwxrwxrwx 3 IBMUSER SYS1 8192 Jul 10 12:11 IBMUSER
-rw-rw-rw- 1 STCRSE STCGRP 36655 Jul 10 12:11 rsedaemon.log
-rw-rw-rw- 1 STCRSE STCGRP 2490 Jul 10 12:12 rseserver.log

```

*Figure 24. Utilisation de l'espace du système de fichiers z/OS UNIX*

La figure 24 illustre l'utilisation minimale de l'espace du système de fichiers z/OS UNIX lors de l'utilisation du niveau de débogage 2 (messages d'information).

- Les journaux de la tâche démarrée utilisent 34 Ko après le démarrage et augmentent doucement lorsque les utilisateurs se connectent, se déconnectent ou que des commandes d'opérateur sont émises.
- Un répertoire de journalisation client utilise 11 Ko après la connexion et augmente régulièrement lorsque l'utilisateur commence à travailler (cette situation n'est pas illustrée dans l'exemple).
- La déconnexion ajoute 40 Ko supplémentaires dans les journaux utilisateur, soit un total de 51 Ko.



A l'exception de journaux d'audit, les fichiers journaux sont écrasés à chaque redémarrage (pour la tâche démarrée RSE) ou connexion (pour un client), ce qui permet de contrôler la taille totale. La directive `keep.last.log` de `rsed.envvars` peut changer cela, étant donné qu'elle peut demander à RSE de conserver un exemplaire des fichiers journaux précédents. Les exemplaires plus anciens sont toujours supprimés.

Un message d'avertissement est envoyé à la console lorsque l'espace disponible dans le système de fichiers qui contient les fichiers journaux d'audit commence à manquer et que le contrôle est actif. Le message de console (FEK103E) s'affiche régulièrement tant que l'incident lié au manque d'espace n'a pas été résolu. Reportez-vous à "Messages de console" dans le *Guide de configuration de l'hôte* (SC11-6285) pour obtenir une liste des messages de console générés par RSE.

Les définitions du tableau 31 contrôlent les données écrites dans les répertoires de journalisation ainsi que l'emplacement de ces répertoires.

*Tableau 31. Répertoires de sortie de journal*

| Adresse              | Directive                        | Fonction                                                                                        |
|----------------------|----------------------------------|-------------------------------------------------------------------------------------------------|
| resecomm.properties  | debug_level                      | Définition du niveau de détails du journal par défaut.                                          |
| rsed.envvars         | keep.last.log                    | Conservation d'un exemplaire des fichiers journaux précédents avant démarrage/connexion.        |
| rsed.envvars         | enable.audit.log                 | Conservation d'un trace d'audit des actions du client.                                          |
| rsed.envvars         | enable.standard.log              | Ecriture des flux stdout et stderr du/des pool(s) d'unités d'exécution dans un fichier journal. |
| rsed.envvars         | DSTORE_TRACING_ON                | Activation du journal des actions du magasin de données.                                        |
| rsed.envvars         | DSTORE_MEMLOGGING_ON             | Activation du journal relatif à l'utilisation de la mémoire par le magasin de données.          |
| Commande d'opérateur | modify rsecommlog <niveau>       | Modification dynamique du niveau de détail du journal de rsecomm.log                            |
| Commande d'opérateur | modify rsedaemonlog <niveau>     | Modification dynamique du niveau de détail du journal de rsedaemon.log                          |
| Commande d'opérateur | modify rseserverlog <niveau>     | Modification dynamique du niveau de détail du journal de rseserver.log                          |
| Commande d'opérateur | modify rsestandardlog {on   off} | Modification dynamique de la mise à jour de std*.log                                            |
| rsed.envvars         | daemon.log                       | Chemin d'accès au répertoire de base de la tâche démarrée RSE et des journaux d'audit.          |
| rsed.envvars         | user.log                         | Chemin d'accès au répertoire de base des journaux utilisateur.                                  |

Tableau 31. Répertoires de sortie de journal (suite)

| Adresse      | Directive    | Fonction                                                                |
|--------------|--------------|-------------------------------------------------------------------------|
| rsed.envvars | CGI_ISPWORK  | Chemin d'accès au répertoire de base des journaux d'ISPF Client Gateway |
| rsed.envvars | TMPDIR       | Répertoire pour les journaux IVP                                        |
| rsed.envvars | _CEE_DMPTARG | Répertoire pour les vidages Java                                        |

Developer for System z, et les logiciels requis (ISPF Client Gateway, par exemple) écrivent également les données temporaires dans /tmp et /var/rdz/WORKAREA. La quantité de données écrites suite aux actions de l'utilisateur n'est pas prévisible. Il est donc recommandé de prévoir un espace disponible suffisant dans les systèmes de fichiers contenant ces répertoires.

Developer for System z tente toujours de nettoyer ces fichiers temporaires, mais le nettoyage manuel, indiqué dans "(Facultatif) Nettoyage de WORKAREA et /tmp" dans *Guide de configuration de l'hôte* (SC11-6285), peut être réalisé pratiquement à tout moment.

Les définitions figurant dans le tableau 32 régissent l'emplacement des répertoires de données temporaires.

Tableau 32. Directives de sortie temporaire

| Adresse      | Directive   | Fonction                                                      |
|--------------|-------------|---------------------------------------------------------------|
| rsed.envvars | CGI_ISPWORK | Chemin d'accès au répertoire de base des données temporaires. |
| rsed.envvars | TMPDIR      | Répertoire des données temporaires.                           |

## Définitions de ressources essentielles

### /etc/rdz/rsed.envvars

Les variables d'environnement définies dans rsed.envvars sont utilisées par RSE, Java et z/OS UNIX. Le fichier exemple qui accompagne Developer for System z vise les petites et moyennes installations qui n'ont pas besoin des composants facultatifs de Developer for System z. "rsed.envvars, fichier de configuration RSE" du *Guide de configuration de l'hôte* (SC11-6285) décrit toutes les variables définies dans l'exemple de fichier, dont certaines qui méritent une attention particulière :

**\_RSE\_JAVA\_OPTS="\$\_RSE\_JAVA\_OPTS -Xms128m -Xmx512m"**

Définit la taille de pile initiale (Xms) et maximale (Xmx). Les valeurs par défaut sont respectivement 128M et 512M. Modifiez la valeur pour appliquer la taille de pile de votre choix. Si cette directive est mise en commentaire, les valeurs par défaut Java sont alors utilisées, 4M et 512M respectivement.

**#\_RSE\_JAVA\_OPTS="\$\_RSE\_JAVA\_OPTS -Dmaximum.clients=30"**

Nombre maximal de clients pris en charge par un même pool d'unités d'exécution. Le nombre par défaut est 30. Supprimez la mise en commentaire et personnalisez l'option pour limiter le nombre de clients par pool d'unités d'exécution. Notez que d'autres limites risquent d'empêcher RSE d'atteindre cette limite.

**#\_RSE\_JAVA\_OPTS="\$\_RSE\_JAVA\_OPTS -Dmaximum.threads=520"**

Nombre maximum d'unités d'exécution actives d'un pool d'unités d'exécution

pour autoriser de nouveaux clients. La valeur par défaut est 520. Supprimez la mise en commentaire et personnalisez pour limiter le nombre de clients par pool d'unités d'exécution en fonction du nombre d'unités d'exécution utilisées. Notez que chaque connexion client utilise plusieurs unités d'exécution (au moins 17) et que d'autres limites risquent d'empêcher RSE d'atteindre cette valeur maximale.

**Remarque :** Cette valeur doit être inférieure à celle de MAXTHREADS et MAXTHREADTASKS dans SYS1.PARMLIB(BPXPRMxx).

**#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS -Dminimum.threadpool.process=1"**

Nombre minimal de pools d'unités d'exécution actifs. La valeur par défaut est 1. Supprimez la mise en commentaire de cette ligne et personnalisez-la pour lancer au moins le nombre de processus de pool d'unités d'exécution répertoriés. Les processus de pool d'unité d'exécution sont utilisés pour l'équilibrage de charge des unités d'exécution du serveur RSE. Des processus supplémentaires sont démarrés, si nécessaire. Le démarrage immédiat de nouveaux processus permet d'éviter les délais de connexion mais utilise davantage de ressources pendant les phases d'inactivité.

**Remarque :** Si la directive `single.logon` est active, au moins deux pools d'unités d'exécution sont démarrés, même si la valeur 1 est attribuée à la directive `minimum.threadpool.process`. Le paramètre par défaut de la directive `single.logon` dans `rsed.envvars` est actif.

**#\_RSE\_JAVAOPTS="\$\_RSE\_JAVAOPTS -Dmaximum.threadpool.process=100"**

Nombre maximal de pools d'unités d'exécution actifs. La valeur par défaut est 100. Supprimez la mise en commentaire et personnalisez pour limiter le nombre de processus de pool d'unité d'exécution. Les processus de pool d'unités d'exécution sont utilisés pour l'équilibrage de charge des unités d'exécution du serveur RSE ; si vous les limitez, ils limiteront donc la quantité de connexions client actives.

## **SYS1.PARMLIB(BPXPRMxx)**

RSE est une application Java, ce qui signifie qu'il est actif dans l'environnement z/OS UNIX. BPXPRMxx peut donc aisément devenir un membre parmlib essentiel, étant donné qu'il contient les paramètres permettant de contrôler l'environnement et les systèmes de fichiers z/OS UNIX. BPXPRMxx est décrit dans le document *MVS Initialization and Tuning Reference* (SA22-7592). Les directives suivantes sont réputées avoir un impact sur Developer for System z :

### **MAXPROCSYS(nnnnn)**

Indique le nombre maximal de processus que le système autorise.

Gamme de valeurs : nnnnn est une valeur décimale comprise entre 5 et 32767.

Valeur par défaut : 900

### **MAXPROCUSER(nnnnn)**

Indique le nombre maximal de processus qu'un seul ID utilisateur z/OS UNIX peut activer simultanément, quelle que soit la manière dont les processus ont été créés.

Gamme de valeurs : nnnnn est une valeur décimale comprise entre 3 et 32767.

Valeur par défaut : 25

**Remarque :**

- Tous les processus RSE utilisent le même ID utilisateur z/OS UNIX (celui de l'utilisateur attribué au démon RSE), car tous les client fonctionnent comme des unités d'exécution dans les processus RSE.
- Cette valeur peut également être établie avec la variable PROCUSERMAX du segment de profil de sécurité OMVS de l'utilisateur attribué à la tâche démarrée RSED.

**MAXTHREADS (nnnnnn)**

Indique le nombre maximal d'unités d'exécution pthread\_created, y compris celles qui sont en cours d'exécution, mises en file d'attente et arrêtées sans être libérées, qu'un seul processus peut activer simultanément. Si vous indiquez la valeur 0, les applications n'utilisent pas pthread\_create.

Gamme de valeurs : nnnnnn est une valeur décimale comprise entre 0 et 100000.

Valeur par défaut : 200

**Remarque :**

- Chaque client utilise au moins 17 unités d'exécution dans le processus du pool d'unités d'exécution RSE, plusieurs clients étant actifs à l'intérieur du processus.
- Cette valeur peut également être établie avec la variable THREADSMAX du segment de profil de sécurité OMVS de l'utilisateur attribué à la tâche démarrée RSED. Lorsqu'elle est définie, la valeur THREADSMAX est utilisée pour MAXTHREADS et MAXTHREADTASKS.

**MAXTHREADTASKS (nnnnn)**

Indique le nombre maximal de tâches MVS qu'un seul processus peut activer simultanément pour les unités d'exécution pthread\_created.

Gamme de valeurs : nnnnn est une valeur décimale comprise entre 0 et 32768.

Valeur par défaut : 1000

**Remarque :**

- Chaque unité d'exécution active comporte une tâche MVS (bloc de contrôle des tâches).
- Chaque tâche MVS simultanée requiert un espace de stockage supplémentaire, dont certains doivent être inférieures à la ligne de 16 Mo.
- Chaque client utilise au moins 17 unités d'exécution dans le processus du pool d'unités d'exécution RSE, plusieurs clients étant actifs à l'intérieur du processus.
- Cette valeur peut également être établie avec la variable THREADSMAX du segment de profil de sécurité OMVS de l'utilisateur attribué à la tâche démarrée RSED. Lorsqu'elle est définie, la valeur THREADSMAX est utilisée pour MAXTHREADS et MAXTHREADTASKS.

**MAXUIDS (nnnnn)**

Indique le nombre maximal d'ID utilisateur z/OS UNIX (UID) qui peuvent opérer simultanément.

Gamme de valeurs : nnnnn est une valeur décimale comprise entre 1 et 32767.

Valeur par défaut : 200

**MAXASSIZE(nnnnn)**

Indique les valeurs de ressource RLIMIT\_AS qui vont faire office de valeurs initiales pour les nouveaux processus. RLIMIT\_AS indique la taille de la région de l'espace adresse.

Gamme de valeurs : nnnnn est une valeur décimale comprise entre 10485760 (10 mégaoctets)

et 2147483647 (2 gigaoctets).

Valeur par défaut : 209715200 (200 mégaoctets)

**Remarque :**

- Cette valeur doit être de 2G.
- Cette valeur peut également être établie avec la variable ASSIZEMAX du segment de profil de sécurité OMVS de l'utilisateur attribué à la tâche démarrée RSED.

**MAXFILEPROC(nnnnnn)**

Indique le nombre maximal de descripteurs pour les fichiers, sockets, répertoires et autres objets de système de fichiers qu'un seul processus peut activer ou allouer simultanément.

Gamme de valeurs : nnnnnn est une valeur décimale comprise entre 3 et 524287.

Valeur par défaut : 64000

**Remarque :**

- Toutes les unités d'exécution client d'un pool d'unités d'exécution se trouvent dans un seul processus.
- Cette valeur peut également être établie avec la variable FILEPROCMAX du segment de profil de sécurité OMVS de l'utilisateur attribué à la tâche démarrée RSED.

**MAXMAPAREA(nnnnn)**

Indique la quantité d'espace de stockage de l'espace de données (en pages) qui peut être allouée pour les mappages mémoire des fichiers z/OS UNIX. L'espace de stockage n'est pas alloué tant que le mappage mémoire n'est pas actif.

Gamme de valeurs : nnnnn est une valeur décimale comprise entre 1 et 16777216.

Valeur par défaut : 40960

**Remarque :** Cette valeur peut également être établie avec la variable MMAPAREAMAX du segment de profil de sécurité OMVS de l'utilisateur attribué à la tâche démarrée RSED.

Utilisez la commande de l'opérateur **SETOMVS** ou **SET OMVS** pour augmenter ou diminuer de manière dynamique (jusqu'à l'IPL suivant) la valeur de l'une des variables BPXPRMxx précédentes. Pour apporter une modification permanente, éditez le membre BPXPRMxx qui va être utilisé pour les IPL. Voir le document *MVS System Commands* (SA22-7627) pour plus d'informations relatives à ces commandes de l'opérateur.

Les définitions suivantes sont des sous-paramètres de l'instruction NETWORK.

**MAXSOCKETS (nnnnnnnn)**

Indique le nombre maximal de sockets pris en charge par ce système de fichiers pour cette famille d'adresses. Il s'agit d'un paramètre facultatif.

Gamme de valeurs : nnnnnnnn est une valeur décimale comprise entre 0 et 16777215.

Valeur par défaut : 100

**INADDRANYCOUNT (nnnn)**

Indique le nombre de ports que le système réserve pour une utilisation avec PORT 0 et les liaisons INADDR\_ANY, en commençant par le numéro de port spécifié dans le paramètre INADDRANYPORT. Cette valeur est uniquement nécessaire pour CINET (plusieurs piles TCP/IP).

Gamme de valeurs : nnnn est une valeur décimale comprise entre 1 et 4000.

Valeur par défaut : si INADDRANYPORT et INADDRANYCOUNT ne sont pas spécifiés, la valeur par défaut d'INADDRANYCOUNT est 1000.  
Sinon, aucun port n'est réservé (0).

---

## Définitions de ressource différentes

### Carte EXEC dans le JCL de serveur

Il est recommandé d'ajouter les définitions suivantes à la carte EXEC dans le JCL des serveurs Developer for System z.

**REGION=0M**

REGION=0M est recommandé pour les tâches démarrées du démon RSE et du moniteur de travaux JES (RSED et JMON, respectivement). Se faisant, la taille de l'espace adresse est limitée uniquement par l'espace de stockage privé disponible ou par la sorti du système IEFUSI ou IEALIMIT. Notez qu'IBM recommande vivement de ne pas utiliser ces sorties pour les espaces adresse z/OS UNIX (le démon RSE, par exemple).

**TIME=NOLIMIT**

Il est recommandé d'utiliser TIME=NOLIMIT pour tous les serveurs Developer for System z. En effet, les temps UC de tous les clients Developer for System z s'accumulent dans les espaces adresse du serveur.

### FEK.#CUST.PARMLIB(FEJJCNFG)

Les variables d'environnement définies dans FEJJCNFG sont utilisées par le moniteur de travaux JES. Le fichier exemple qui accompagne Developer for System z vise les petites et moyennes installations. "FEJJCNFG, Fichier de configuration Moniteur de travaux JES" du *Guide de configuration de l'hôte* (SC11-6285) décrit toutes les variables définies dans l'exemple de fichier, dont certaines qui méritent une attention particulière :

**MAX\_THREADS**

Nombre maximal d'utilisateurs qui peuvent utiliser simultanément un moniteur de travaux JES. La valeur par défaut est 200. La valeur maximale est 2147483647. Si vous augmentez cette valeur, vous devez augmenter la taille de l'espace adresse du moniteur de travaux JES.

## SYS1.PARMLIB(IEASYSxx)

IEASYSxx contient les paramètres système et est décrit dans le document *MVS Initialization and Tuning Reference* (SA22-7592). Les directives suivantes sont réputées avoir un impact sur Developer for System z :

### MAXUSER=nnnnn

Ce paramètre indique une valeur que le système utilise, sous certaines conditions, pour limiter le nombre de travaux et de tâches démarrées qui peuvent être exécutés simultanément lors d'une IPL donnée.

Gamme de valeurs : nnnnn est une valeur décimale comprise entre 0 et 32767. Notez que la somme des valeurs spécifiées pour les paramètres système MAXUSER, RSVSTRT, et RSVNONR ne peut pas dépasser 32767.  
Valeur par défaut : 255

## SYS1.PARMLIB(IVTPRMxx)

IVTPRMxx permet d'attribuer une valeur aux paramètres du gestionnaire de stockage des communications (CSM) et est décrit dans le document *MVS Initialization and Tuning Reference* (SA22-7592). Les directives suivantes sont réputées avoir un impact sur Developer for System z :

### FIXED MAX(maxfix)

Définit la quantité maximale d'espace de stockage dédié aux mémoires tampons CSM fixes.

Gamme de valeurs : maxfix est une valeur comprise entre 1024 Ko et 2048 Mo.  
Valeur par défaut : 100 Mo

### ECSA MAX(maxecsa)

Définit la quantité maximale d'espace de stockage dédié aux mémoires tampons CSM ECSA.

Gamme de valeurs : maxecsa est une valeur comprise entre 1024 Ko et 2048 Mo.  
Valeur par défaut : 100 Mo

## SYS1.PARMLIB(ASCHPMxx)

Le membre parmlib ASCHPMxx contient des informations de planification pour le programme de transactions ASCH et est décrit dans le document *MVS Initialization and Tuning Reference* (SA22-7592). Les directives suivantes sont réputées avoir un impact sur Developer for System z :

### MAX(nnnnn)

Paramètre facultatif de la définition CLASSADD indiquant le nombre maximal de demandeurs de transaction APPC admis pour une classe particulière de demandeurs de transaction. Lorsque cette limite est atteinte, plus aucun espace adresse n'est créé et les demandes entrantes sont placées dans la file d'attente tant que les espaces adresse existants du demandeur ne sont pas disponibles. La valeur ne doit pas dépasser le nombre maximal d'espaces adresse admis par votre installation. N'oubliez pas de comparer les produits sur le système qui va également avoir besoin d'espaces adresse.



Gamme de valeurs : nnnnn est une valeur décimale comprise entre 1 et 64000.  
Valeur par défaut : 1

**Remarque :** Si vous utilisez APPC pour démarrer le service Commandes TSO, la classe de transaction utilisée doit comporter suffisamment de demandeurs de transaction pour en autoriser un par utilisateur simultané de Developer for System z.

---

## Contrôle

Etant donné que les charges de travail de l'utilisateur peuvent modifier les besoins en ressources système, il est recommandé de contrôler le système régulièrement pour mesurer l'utilisation des ressources, de manière à pouvoir ajuster Rational Developer for System z et les configurations système en fonction des exigences de l'utilisateur. Les commandes suivantes peuvent être utilisées pour faciliter ce processus de contrôle.

### Contrôle de RSE

Les pools d'unités d'exécution RSE sont le point focal de l'activité d'utilisateur dans Developer for System z et doivent donc être contrôlés pour assurer une utilisation optimale. Le démon RSE peut s'avérer nécessaire pour les informations qui ne peuvent pas être rassemblés avec les outils de contrôle du système habituels.

- Utilisez vos outils de contrôle habituels (RMF, par exemple) pour rassembler des données spécifiques de l'espace adresse (l'espace de stockage réel utilisé et le temps UC, par exemple. Si vous ne disposez pas de ce type d'outil, vous pouvez utiliser des outils comme la vue SDSF DA ou TASID (un outil d'informations système en l'état disponible sur la page Web ISPF "Support and downloads") afin de rassembler des informations de base.
- Lors du démarrage, le démon RSE rapporte la taille de l'espace adresse disponible et la taille de pile Java avec le message de console FEK004I.  
FEK004I RseDaemon: Max Heap Size=65MB and private AS Size=1,959MB
- La commande de l'opérateur **MODIFY RSED,APPL=DISPLAY PROCESS** affiche les processus du pool d'unités d'exécution RSE. La zone "Memory Usage" affiche la quantité de pile Java définie réellement utilisée. Reportez-vous à la section "Commandes de l'opérateur" du *Guide de configuration de l'hôte* (SC11-6285) pour obtenir plus d'informations sur cette commande.

```
f rsed,appl=d p
BPXM023I (STCRSE)
ProcessId(16777456) Memory Usage(33%) Clients(4) Order(1)
```

Des informations supplémentaires sont fournies lorsque vous utilisez l'option **DETAIL** de la commande de modification **DISPLAY PROCESS** :

```
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
```

L'option CPU de la commande de modification **DISPLAY PROCESS** affiche l'utilisation totale de l'unité centrale (en millisecondes) de chaque unité d'exécution d'un pool :



```

f rsed,appl=d p,cpu
BPXM023I (STCRSE)
ProcessId(33555087) ASId(002E) JobName(RSED8) Order(1)
USERID THREAD-ID TCB@ ACC_TIME TAG
STCRSE 0EDE54000000000 005E6B60 822 1/ThreadPoolProcess
STCRSE 0EDE8700000000001 005E69C8 001
STCRSE 0EDE9800000000002 005E6518 1814
STCRSE 0EDEBA00000000003 005E66B0 2305
STCRSE 0EDECB00000000004 005E62F8 001
STCRSE 0EDED000000000005 005E60D8 001
STCRSE 0EDF8600000000006 005C2BF8 628 6/ThreadPoolMonitor$Memory
UsageMonitor
STCRSE 0EDF9700000000007 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

```

- Lorsqu'un processus de pool d'unités d'exécution RSE se termine, il affiche des statistiques détaillées sur l'utilisation des ressources, comme si la commande de modification **DISPLAY PROCESS,DETAIL** avait été exécutée uniquement pour ce processus de pool d'unités d'exécution RSE. La cote maximale atteinte affiche l'utilisation des ressources simultanées maximum pour la durée de vie du processus de pool d'unités d'exécution RSE, ce qui permet à un outil d'optimisation du système de déterminer si les ressources affectées à RSE sont surallouées ou sous-allouées.

## Contrôle de z/OS UNIX

La plupart des limites z/OS UNIX qui présentent un intérêt pour Developer for System z peuvent être affichées à l'aide des commandes de l'opérateur. Certaines commandes affichent même l'utilisation réelle et la cote d'alerte haute associées à une limite particulière. Voir le document *MVS System Commands* (SA22-7627) pour plus d'informations relatives à ces commandes.

- La directive LIMMSG(ALL) de SYS1.PARMLIB(BPXPRMxx) demande à z/OS UNIX d'afficher les messages de console (BPXI040I) lorsque l'une des limites parmlib est sur le point d'être atteinte. La valeur par défaut de LIMMSG est NONE, ce qui désactive la fonction. Utilisez la commande de l'opérateur **SETOMVS LIMMSG=ALL** afin d'activer dynamiquement cette fonction (jusqu'à l'IPL suivante). Voir le document *MVS Initialization and Tuning Reference* (SA22-7592) pour plus d'informations relatives à cette directive.
- La commande de l'opérateur **DISPLAY OMVS,OPTIONS** affichent les valeurs en cours des directives z/OS UNIX qui peuvent être définies de manière dynamique.

```

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
MAXPTYS = 256
MAXMAPAREA = 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.DYNSERV.LOADLIB BPXLK1
SERV_LPALIB = POSIX.DYNSERV.LOADLIB BPXLK1
PRIORITYPG VALUES: NONE
PRIORITYGOAL VALUES: NONE
MAXQUEUEDSIGs = 1000 SHRLIBRGNSIZE = 67108864
SHRLIBMAXPAGES = 4096 VERSION = /
SYSCALL COUNTS = NO TTYGROUP = TTY
SYSPLEX = NO BRM SERVER = N/A
LIMMSG = NONE AUTOCVT = OFF
RESOLVER PROC = DEFAULT
AUTHPGMLIST = NONE
SWA = BELOW

```

- La commande de l'opérateur **DISPLAY OMVS,LIMITS** affiche les informations relatives aux limites parmlib z/OS UNIX System Services en cours, leurs cotes d'alertes haute et l'utilisation en cours du système.

```

d omvs,l
BPX0051I 14.05.52 DISPLAY OMVS 904
OMVS 0042 ACTIVE OMVS=(69)
SYSTEM WIDE LIMITS: LIMMSG=SYSTEM

```

|                    | CURRENT<br>USAGE | HIGHWATER<br>USAGE | SYSTEM<br>LIMIT |
|--------------------|------------------|--------------------|-----------------|
| <b>MAXPROCSYS</b>  | <b>1</b>         | <b>4</b>           | <b>256</b>      |
| <b>MAXUIDS</b>     | <b>0</b>         | <b>0</b>           | <b>200</b>      |
| MAXPTYS            | 0                | 0                  | 256             |
| <b>MAXMMAPAREA</b> | <b>0</b>         | <b>0</b>           | <b>256</b>      |
| 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            |

La commande affiche les cotes d'alerte hautes et l'utilisation en cours d'un processus individuel lorsque le mot clé PID=processid est également spécifié.

```

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<br>USAGE | HIGHWATER<br>USAGE | PROCESS<br>LIMIT |
|-----------------------|------------------|--------------------|------------------|
| <b>MAXFILEPROC</b>    | <b>83</b>        | <b>103</b>         | <b>256</b>       |
| MAXFILESIZE           | ---              | ---                | NOLIMIT          |
| <b>MAXPROCUSER</b>    | <b>97</b>        | <b>99</b>          | <b>200</b>       |
| MAXQUEUEDSIGs         | 0                | 1                  | 1000             |
| <b>MAXTHREADS</b>     | <b>9</b>         | <b>14</b>          | <b>200</b>       |
| <b>MAXTHREADTASKS</b> | <b>9</b>         | <b>14</b>          | <b>1000</b>      |
| IPCSHMNSEGS           | 0                | 0                  | 500              |
| MAXCORESIZE           | ---              | ---                | 4194304          |
| MAXMEMLIMIT           | 0                | 0                  | 16383P           |

- La commande de l'opérateur **DISPLAY OMVS,PFS** affiche des informations relatives à chaque système de fichiers physique faisant partie intégrante de la configuration z/OS UNIX, qui inclut les piles 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 EZBPFFINI 50000 244 8146
UDS SOCKETS AF_UNIX BPXTUINIT 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 EZBPFFINI ACT CD
TCP42 SOCKETS EZBPFFINI ACT
TCP43 SOCKETS EZBPFFINI INACT SD
TCP44 SOCKETS EZBPFFINI INACT
PFS PARM INFORMATION
HFS SYNCDEFAULT(60) FIXED(50) VIRTUAL(100)
 CURRENT VALUES: FIXED(55) VIRTUAL(100)
NFS biod(6)

```

- La commande de l'opérateur **DISPLAY OMVS,PID=processid** affiche les informations relatives à l'unité d'exécution pour un processus particulier.

```

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 TCB@ 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

```

## Contrôle du réseau

Lors de la prise en charge d'un nombre important de clients se connectant à l'hôte, Developer for System z et votre infrastructure réseau doivent être en mesure de gérer la charge de travail. La gestion du réseau est un sujet largement abordé qui n'entre pas dans le domaine d'application de la documentation Developer for System z. Par conséquent, seuls les pointeurs suivants sont fournis.

- La commande de l'opérateur **DISPLAY NET,CSM** permet au gestionnaire de stockage des communications de contrôler l'utilisation de l'espace de stockage. Vous pouvez utiliser cette commande afin de déterminer la quantité d'espace de stockage CSM en cours d'utilisation pour ECSA et les pools de stockage d'espace de données (voir *Communications Server SNA Operations* (SC31-8779)).

## Contrôle des systèmes de fichiers z/OS UNIX

Developer for System z utilise les systèmes de fichiers z/OS UNIX pour stocker les différents types de données (les journaux et fichiers temporaires, par exemple). Utilisez la commande z/OS UNIX **df** pour connaître le nombre de descripteurs de fichier encore disponibles et la quantité d'espace libre avant la création de la prochaine étendue du fichier HFS ou zFS sous-jacent.

```
$ df
Mounted on Filesystem Avail/Total Files Status
/tmp (OMVS.TMP) 1393432/1396800 4294967248 Available
/u/ibmuser (OMVS.U.IBMUSER) 1248/1728 4294967281 Available
/usr/lpp/rdz (OMVS.LPP.FEK) 3062/43200 4294967147 Available
/var (OMVS.VAR) 27264/31680 4294967054 Available
```

## Exemple de configuration

L'exemple de configuration ci-dessous illustre la configuration requise permettant de prendre en charge ces exigences :

- 500 connexions client simultanées
- 300 générations MVS simultanées (travail par lots)
- 200 connexions CARMA simultanées (utilisation de la méthode de configuration CRASTART)
- 3 heures de délai d'attente d'inactivité
- Interdisez l'utilisation de z/OS UNIX
- SCLM Developer Toolkit et File Manager Integration ne sont pas utilisés
- Prévoyez une utilisation moyenne de segment de mémoire Java de 20 Mo.
- Les utilisateurs disposent d'UID z/OS UNIX uniques

## Nombre de pools d'unités d'exécution

Par défaut, Developer for System z tente d'ajouter 30 utilisateurs à un seul pool d'unités d'exécution. Toutefois, nos exigences stipulent que le délai d'attente d'inactivité soit actif. Le tableau 28, à la page 89 indique qu'une unité d'exécution va être ajoutée par client connecté. Il s'agit d'une unité d'exécution de temporisation qui doit donc être active en permanence. Cela empêche RSE de placer 30 utilisateurs dans un seul pool d'unités d'exécution, étant donné que  $10 + 30 \times (17 + 1) = 550$ , et que le paramètre `maximum.threads` est défini par défaut sur 520.

La valeur de `maximum.threads` pourrait être augmentée, mais compte tenu de l'obligation de prévoir un segment de mémoire Java moyen de 20 Mo par utilisateur, la valeur de `maximum.clients` a été abaissée à 25 ( $10 + 25 \times 18 = 460$ ). La taille maximale du segment de mémoire Java de 512 Mo ( $20 \times 25 = 500$ ) est respectée.

Avec 25 clients par pool d'unités d'exécution et la nécessité de prendre en charge 500 connexions, nous savons désormais que nous allons avoir besoin de 20 espaces adresse de pool d'unités d'exécution.

## Détermination des limites minimales

A l'aide des formules présentées ci-avant dans ce chapitre et les critères établis au début de la présente section, il est possible de déterminer l'utilisation des ressources.

- Nombre d'espaces adresse - maximal  
 $3 + A + N \times (x + y + z) + (2 + N \times 0.01)$   
 $3 + 20 + 500 \times 1 + 200 \times 1 + 300 \times 1 + (2 + 500 \times 0.01) = 1030$
- Nombre d'espaces adresse - par utilisateur  
 $x + y + z$   
 $1 + 1 + 1 = 3$
- Nombre de processus - maximal

$$5 + 2*A + N*(x + y + z) + (10 + N*0.05)$$

$$5 + 2*20 + 500*2 + 200*1 + 300*0 + (10 + 500*0.05) = 1570$$

- Nombre de processus - STCRSE

$$4 + 2*A$$

$$4 + 2*20 = 44$$

- Nombre de processus - par utilisateur

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

$$(2 + 1 + 0) + 5*0 = 3$$

- Nombre d'unités d'exécution - pool d'unités d'exécution RSE

$$10 + N*(17 + x + y + z) + (20 + N*0.1)$$

$$10 + 25*(17 + 1 + 4 + 0) + (20 + 25*0.1) = 583$$

- Nombre d'unités d'exécution - moniteur de travaux JES

$$3 + N$$

$$3 + 500 = 503$$

- ID utilisateur

$$500 + 2 = 502$$

Les 2 ID utilisateur supplémentaires sont pour STCJMON et STCRSE, les ID utilisateur de la tâche démarrée Developer for System z.

## Définition des limites

Maintenant que les numéros d'utilisation des ressources sont connus, il est possible de personnaliser les directives de personnalisation avec les valeurs appropriées.

- /etc/rdz/rsed.envvars

– Xmx512m

non modifié

– Dmaximum.clients=25

– Dmaximum.threads=520

non modifié

– Dminimum.threadpool.process=10

Ce changement est facultatif ; RSE démarrera des nouveaux pools d'unités d'exécution, si nécessaire

– DHIDE\_ZOS\_UNIX=true

– DDSTORE\_IDLE\_SHUTDOWN\_TIMEOUT=10800000

- FEK.#CUST.PARMLIB(FEJJCENFG)

– MAX\_THREADS=503

- SYS1.PARMLIB(BPXPRMxx)

– MAXPROCSYS(2500)

1572 au moins, mémoire tampon supplémentaire ajoutée  
pour les tâches autres que  
Developer  
for System z

– MAXPROCUSER(80)

44 au moins, mémoire tampon supplémentaire ajoutée lorsque les pools d'unités d'exécution RSE prennent en charge moins de 25 clients projetés.

- MAXTHREADS(1500)

doit être de 503 au moins  
(pour le moniteur de travaux JES) si THREADSMAX du segment OMVS de l'ID utilisateur STCRSE est utilisé pour définir la limite de RSE  
(582 au moins)

- MAXTHREADTASKS(1500)

doit être de 503 au moins (pour le moniteur de travaux JES)  
si THREADSMAX du segment OMVS de l'ID utilisateur STCRSE est utilisé pour définir la limite de RSE  
(582 au moins)

- MAXUIDS(700)

503 au moins, mémoire tampon supplémentaire ajoutée pour les tâches autres que Developer  
for System z

- MAXASSIZE(209715200)

non modifié (valeur par défaut du système de 200 Mo), nous utilisons ASSIZEMAX dans le segment OMVS de l'ID utilisateur STCRSE

- SYS1.PARMLIB(IEASYSxx)

- MAXUSER=2000

1030 au moins, mémoire tampon supplémentaire ajoutée pour les tâches autres que Developer  
for System z

- Segment OMVS de l'ID utilisateur STCRSE

- ASSIZEMAX(2147483647)

2 Go

## Utilisation des ressources du moniteur

Après l'activation des limites du système comme documenté dans «Définition des limites», à la page 113, nous pouvons démarrer le contrôle de l'utilisation des ressources par Developer for System z pour voir si un ajustement de certaines variables est nécessaire. figure 25, à la page 115 affiche l'utilisation des ressources après la connexion de 499 utilisateurs. (L'exemple présenté sur la figure montre seulement la connexion. Aucune action utilisateur n'est indiquée dans l'exemple.)

```

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 44 44 80
 MAXTHREADS 310 311 1500
 MAXTHREADTASKS 311 311 1500

```

| TASID | 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 |

Figure 25. Utilisation des ressources de la configuration modèle





---

## Chapitre 6. Remarques relatives aux performances

z/OS est un système d'exploitation hautement personnalisable, et des modifications (parfois mineures) du système peuvent présenter un impact très important sur les performances globales. Le présent chapitre met en évidence certaines modifications qui peuvent être apportées afin d'améliorer les performances de Developer for System z.

Pour plus d'informations sur l'optimisation du système, voir le document *MVS Initialization and Tuning Guide* (SA22-7591) and *UNIX System Services Planning* (GA22-7800).

---

### Utilisation du système de fichiers zFS

zFS (zSeries File System) et HFS (Hierarchical File System) sont tous deux des systèmes de fichiers UNIX qui peuvent être utilisés dans un environnement z/OS UNIX. Cependant, zFS fournit les fonctions et avantages suivants :

- Des gains de performances sont constatés dans nombre d'environnements de clients lors de l'accès à des fichiers approchant une taille de 8 Ko, fréquemment ouverts et mis à jour. Les performances d'accès aux fichiers de plus petite taille sont équivalentes à celles du système de fichiers HFS.
- Clonage en lecture seule d'un système de fichiers du même fichier. Le système de fichiers cloné peut être accessible aux utilisateurs afin de fournir une copie instantanée en lecture seule d'un système de fichiers. Cette fonction disponible uniquement dans un environnement non sysplex est facultative.
- zFS est le système de fichiers stratégique z/OS UNIX. Les fonctionnalités HFS ont été stabilisées et des optimisations apportées au système de fichiers seront uniquement appliquées à zFS.

Pour plus d'informations sur zFS, voir le document *UNIX System Services Planning* (GA22-7800).

---

### Eviter l'emploi de STEPLIB

Chaque processus z/OS UNIX qui présente un STEPLIB propagé de parent à enfant ou à travers une commande exec utilise environ 200 octets de zone de mémoire commune étendue ECSA. Si aucune variable d'environnement STEPLIB n'est définie ou qu'elle est définie comme STEPLIB=CURRENT, z/OS UNIX propage toutes les allocations TASKLIB, STEPLIB et JOBLIB actuellement actives lors d'une fonction fork(), spawn(), ou exec().

Developer for System z présente une valeur par défaut STEPLIB=NONE codée dans rsed.envvars, comme décrit dans le fichier de configuration rsed.envvars. Pour les raisons mentionnées précédemment, il est conseillé de ne pas modifier cette directive et de placer les fichiers ciblés dans LINKLIST ou dans la zone permanente de programme LPA à la place.

---

## Amélioration de l'accès aux bibliothèques du système

Certaines bibliothèques du système ainsi que certains modules de chargement sont fortement utilisés par z/OS UNIX et par les activités de développement d'applications. En améliorant leur accès (en les ajoutant à la zone permanente de programme (LPA), par exemple), il est possible d'améliorer les performances du système. Pour plus d'informations sur la modification des membres SYS1.PARMLIB décrits ci-après, voir le document *MVS Initialization and Tuning Reference* (SA22-7592).

### Bibliothèques d'exécution Language Environment (LE)

Lorsque des programmes C (y compris le shell z/OS UNIX) sont exécutés, ils utilisent fréquemment des routines issues de la bibliothèque d'exécution Language Environment (LE). En moyenne, environ 4 mégaoctets de bibliothèque d'exécution sont chargés dans la mémoire pour chaque espace adresse qui exécute un programme LE activé, et sont copiés à chaque fourche.

CEE.SCEELPA

Le fichier CEE.SCEELPA contient un sous-ensemble de routines d'exécution LE, qui sont fortement utilisées par z/OS UNIX. Il est conseillé, pour gagner le maximum de performances, d'ajouter ce fichier à SYS1.PARMLIB(LPALSTxx). De cette manière, les modules sont lus une seule fois sur le disque, et sont stockés dans un emplacement partagé.

**Remarque :** Ajoutez les instructions suivantes à SYS1.PARMLIB(PROGxx) si vous préférez ajouter les modules de chargement à la zone permanente de programme (LPA) dynamique.

```
LPA ADD MASK(*) DSNAME(CEE.SCEELPA)
```

Il est également conseillé de placer les bibliothèques d'exécution LE CEE.SCEERUN et CEE.SCEERUN2 dans LINKLIST, en ajoutant les fichiers à SYS1.PARMLIB(LNKLSTxx) ou à SYS1.PARMLIB(PROGxx). Cela élimine le temps système de z/OS UNIX STEPLIB, et il y a moins d'entrées/sorties en raison de la gestion par LLA et VLF ou par des produits similaires.

**Remarque :** Ajoutez également la bibliothèque de classe de la bibliothèque de chargement dynamique C/C++ CBC.SCLBDLL à LINKLIST, pour les mêmes raisons.

Si vous décidez de ne pas mettre ces bibliothèques dans LINKLIST, vous devez alors configurer l'instruction STEPLIB appropriée dans rsed.envvars, comme décrit dans le fichier de configuration rsed.envvars. Même si cette méthode utilise toujours de la mémoire virtuelle supplémentaire, vous pouvez améliorer les performances en définissant les bibliothèques d'exécution LE pour LLA ou un produit similaire. Cela réduit les entrées/sorties nécessaires au chargement des modules.

### Développement d'applications

Les performances des systèmes sur lesquels le développement d'applications est l'activité principale peuvent également être améliorées si vous placez l'éditeur de liens dans la LPA dynamique, en ajoutant les lignes suivantes à SYS1.PARMLIB(PROGxx) :

```
LPA ADD MODNAME(CEEINIT,CEEBLIB,CEEV003,EDCV) DSNAME(CEE.SCEERUN)
LPA ADD MODNAME(IEFIB600,IEFXB603) DSNAME(SYS1.LINKLIB)
```

Pour le développement C/C++, vous pouvez également ajouter le fichier du compilateur CBC.SCCNCMP à SYS1.PARMLIB(LPALSTxx).

Les instructions précédentes sont des exemples de candidats LPA possibles, mais les besoins de votre site peuvent être différents. Voir *Language Environment Customization* (SA22-7564) pour plus d'informations sur le placement d'autres modules de chargement LE dans la LPA dynamique. Pour plus d'informations sur l'insertion de modules de chargement de compilateur C/C++ dans la LPA dynamique, voir le document *UNIX System Services Planning* (GA22-7800).

---

## Amélioration des performances du contrôle d'autorisations d'accès

Afin d'améliorer les performances des contrôles d'autorisations d'accès effectués pour z/OS UNIX, définissez le profil BPX.SAFFASTPATH dans la classe FACILITY de votre logiciel de sécurité. Cela réduit le temps système des contrôles des droits d'accès de z/OS UNIX pour une large plage d'opérations. Cette gamme comprend le contrôle de l'accès aux fichiers, le contrôle de l'accès aux communications interprocessus, et le contrôle de propriété de processus. Pour plus d'informations sur ce profil, voir *UNIX System Services Planning* (GA22-7800).

**Remarque :** Les utilisateurs n'ont pas besoin d'avoir de permission pour le profil BPX.SAFFASTPATH.

---

## Gestion de la charge de travail

Chaque site a des besoins spécifiques, et il est possible de personnaliser le système d'exploitation z/OS pour tirer le meilleur parti des ressources disponibles pour répondre à ces besoins. Avec la gestion de charge de travail, vous pouvez définir des objectifs de performances et leur attribuer une importance. Vous définissez les buts en terme d'activités, et le système décide quelles ressources, comme le stockage ou l'unité centrale, doivent être allouées à la tâche pour qu'elle atteigne son but.

Les performances de Developer for System z peuvent être équilibrées en paramétrant les but adéquats pour ses processus. Vous trouverez des instructions générales ci-dessous :

- En cas d'utilisation, attribuez une transaction APPC à un groupe de performances TSO.
- Attribuez un groupe de performances de tâche démarrée (SYSSTC) aux espaces adresse du serveur Developer for System z : moniteur de travaux JES (JMON), démon RSE (RSED) et pools d'unités d'exécution RSE (RSEDx).

Pour plus d'informations sur ce sujet, voir *MVS Planning Workload Management* (SA22-7602).

---

## Taille de pile Java fixe

Avec une pile à taille fixe, aucune expansion ou contraction de pile ne peut se produire, ce qui peut permettre des gains de performances significatifs dans certaines situations. Toutefois, l'emploi d'une pile de taille fixe n'est généralement pas une bonne idée, dans la mesure où elle retarde le démarrage de la récupération de place jusqu'au moment où la pile est pleine, ce qui est une tâche très importante. Elle augmente également le risque de fragmentation, qui nécessite une compression de la pile. Aussi, n'utilisez les piles à taille fixe qu'après avoir effectué des essais

appropriés, ou sur instructions du point service IBM. Pour plus d'informations sur les tailles de pile et la récupération de place, voir le document *Java Diagnostics Guide* (SC34-6650).

La taille de pile initiale et maximale d'une machine virtuelle Java z/OS peut être définie avec les options de ligne de commande Java `-Xms` (initiale) et `-Xmx` (maximale).

Dans Developer for System z, les options de ligne de commande Java sont définies dans la directive `_RSE_JAVA_OPTS` de `rsed.envvars`, comme indiqué dans "Définition de paramètres de démarrage supplémentaires Java avec `_RSE_JAVA_OPTS`" dans *Guide de configuration de l'hôte* (SC11-6285).

```
#_RSE_JAVA_OPTS="$_RSE_JAVA_OPTS -Xms128m -Xmx128m"
```

---

## Option Java -Xquickstart

**Remarque :** L'option Java `-Xquickstart` est uniquement utile si vous utilisez la méthode de démarrage REXEC/SSH alternative pour le serveur RSE. Cette méthode est documentée à la section "(Facultatif) Utilisation de REXEC (ou SSH)" du *Guide de configuration de l'hôte* (SC11-6285).

L'option `-Xquickstart` peut être utilisée pour améliorer le délai de démarrage de certaines applications Java. L'option `-Xquickstart` entraîne l'exécution du compilateur JIT (Just In Time) avec un sous-ensemble d'optimisations (il s'agit d'une compilation rapide). Cette compilation rapide permet d'améliorer le délai de démarrage.

L'option `-Xquickstart` convient aux applications avec un temps d'exécution plus court, notamment les applications pour lesquelles le délai d'exécution n'est pas concentré dans un petit nombre de méthodes. L'option `-Xquickstart` peut affecter les performances si elle est utilisée avec des applications connaissant un délai d'exécution plus long qui contiennent des méthodes à chaud.

Pour activer l'option `-Xquickstart` pour le serveur RSE, ajoutez la directive suivante à la fin de `rsed.envvars` :

```
_RSE_JAVA_OPTS="$_RSE_JAVA_OPTS -Xquickstart"
```

---

## Partage de classes entre machines virtuelles Java

IBM Java Virtual Machine (JVM) versions 5 et suivantes vous permet de partager entre JVM les amorces et les classes d'application par leur stockage dans un cache de la mémoire partagée. Le partage de classes réduit la consommation globale de mémoire virtuelle quand plusieurs JVM partagent un cache. Le partage de classes réduit également le temps de démarrage d'une JVM une fois que le cache a été créé.

Le cache de classes partagées est indépendant de toute JVM active et se conserve au-delà de la durée de vie de la JVM qui l'a créé. Dans la mesure où le cache de classes partagées se conserve au-delà de la durée de vie de toute JVM, le cache est mis à jour de façon dynamique afin de refléter toute modification qui peut avoir été apportée aux JAR ou aux classes sur le système de fichiers.

Le temps système pour créer et remplir un nouveau cache est minimal. Le coût de démarrage en temps d'une JVM unique est généralement entre 0 et 5 % plus lent que celui d'un système qui n'utilise pas le partage de classes, en fonction du

nombre de classes qui est chargé. L'amélioration du temps de démarrage de JVM avec un cache rempli est généralement entre 10 % et 40 % plus rapide par rapport à celui d'un système n'utilisant pas le partage de classe, en fonction du système d'exploitation et du nombre de classes chargées. Plusieurs JVM exécutées en même temps présenteront des bénéfices plus importants en terme de temps de démarrage.

Pour plus d'informations sur le partage des classes, voir le document *Java SDK and Runtime Environment User Guide*.

## Activer le partage de classes

Pour activer le partage de classes pour le serveur RSE, ajoutez la directive suivante à la fin de `rsed.envvars`. La première instruction définit un cache nommé RSE avec accès de groupe et permet au serveur RSE de démarrer même si le partage de classes échoue. La seconde instruction est facultative. Elle définit la taille du cache à 6 mégaoctets (la valeur par défaut du système étant 16 mégaoctets). La troisième instruction ajoute les paramètres de partage de classe aux options de démarrage 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"
```

**Remarque :** Comme il est mentionné dans «Sécurité de la mémoire cache», tous les utilisateurs des classes partagées doivent avoir le même identificateur de groupe primaire (ID groupe). Cela signifie que les utilisateurs doivent avoir le même groupe par défaut défini dans le logiciel de sécurité, ou que des groupes par défaut différents présentent le même ID groupe dans leur segment OMVS.

## Limites de taille de la mémoire cache

Le maximum théorique de la taille de la mémoire cache est de 2 gigaoctets. La taille de la mémoire cache que vous pouvez indiquer est limitée par la quantité de mémoire physique et par le fichier d'échange disponible pour le système. Dans la mesure où l'espace d'adresse virtuelle d'un processus est partagée entre la mémoire cache de classe partagée et la pile Java, l'augmentation de la taille maximale de la pile Java réduit la taille de la mémoire cache de classes partagées que vous pouvez créer.

## Sécurité de la mémoire cache

L'accès au cache de classes partagées est limité par les autorisation du système d'exploitation et par les autorisations de sécurité de Java.

Par défaut, les caches de classes sont créés avec une sécurité au niveau utilisateur, de sorte que seul l'utilisateur qui a créé le cache peut y avoir accès. Dans z/OS UNIX, l'option `groupAccess` donne accès à tous les utilisateurs du groupe primaire de l'utilisateur qui a créé le cache. Toutefois, quel que soit le niveau d'accès utilisé, un cache peut uniquement être détruit par l'utilisateur qui l'a créé, ou par le superutilisateur (UID 0).

Pour plus d'informations sur les options de sécurité supplémentaires à l'aide d'un gestionnaire de sécurité Java, voir le document *Java SDK and Runtime Environment User Guide*.

## SYS1.PARMLIB(BPXPRMxx)

Certains paramètres de SYS1.PARMLIB(BPXPRMxx) affectent les performances des classes partagées. L'emploi des mauvais paramètres peut arrêter le fonctionnement des classes partagées. Ces paramètres peuvent également avoir un impact sur les performances. Pour de plus amples informations sur les implications relatives aux performances et à l'utilisation de ces paramètres, voir les documents *MVS Initialization and Tuning Reference* (SA22-7592) et *UNIX System Services Planning* (GA22-7800). Les paramètres BPXPRMxx les plus significatifs qui affectent le fonctionnement des classes partagées sont :

- MAXSHAREPAGES, IPCSHMPAGES, IPCSHMPAGES et IPCSHMNSEGS

Ces paramètres affectent la quantité de pages de mémoire partagée disponible pour la machine virtuelle Java. La taille de page partagée pour un service de système z/OS UNIX 31-bit est fixée à 4 kilooctets. Les classes partagées essaient de créer par défaut une mémoire cache de 16 mégaoctets. Définissez donc IPCSHMPAGES à une valeur supérieure à 4096.

Si vous définissez une taille de cache en utilisant -Xscmx, la machine virtuelle Java arrondira la valeur au mégaoctet le plus proche. Vous devez en tenir compte lors de la définition de IPCSHMPAGES sur le système.

- IPCSEMNIIDS et IPCSEMNSEMS

Ces paramètres affectent la quantité de sémaphores disponibles pour les processus UNIX. Les classes partagées utilisent les sémaphores de communication interprocessus pour dialoguer entre les machines virtuelles Java.

## Espace disque

Le cache de classes partagées nécessite de l'espace disque pour stocker les informations d'identification concernant les caches qui existent sur le système. Ces informations sont stockées dans /tmp/javasharedresources. Si le répertoire des informations d'identification est effacé, la machine virtuelle Java ne peut plus identifier les classes partagées sur le système, et doit recréer le cache.

## Utilitaires de gestion de la mémoire cache

La ligne de commande Java -Xshareclasses peut prendre un certain nombre d'options, dont certaines sont des utilitaires de gestion de la mémoire cache. Trois d'entre elles sont présentées dans l'exemple ci-après (\$ est l'invite z/OS UNIX). Pour une présentation complète des options de ligne de commande prises en charge, voir le document *Java SDK and Runtime Environment User Guide*.

```
$ java -Xshareclasses:listAllCaches
Shared Cache OS shmid in use Last detach time
RSE 401412 0 Mon Jun 18 17:23:16 2007
```

Could not create the Java virtual machine.

```
$ java -Xshareclasses:name=RSE,printStats
```

Current statistics for cache "RSE":

```
base address = 0x0F300058
end address = 0x0F8FFFF8
allocation pointer = 0x0F4D2E28
```

```
cache size = 6291368
free bytes = 4355696
ROMClass bytes = 1912272
Metadata bytes = 23400
Metadata % used = 1%
```

```
ROMClasses = 475
Classpaths = 4
URLs = 0
Tokens = 0
Stale classes = 0
% Stale classes = 0%
```

Cache is 30% full

Could not create the Java virtual machine.

```
$ java -Xshareclasses:name=RSE,destroy
JVMSHRC010I Shared Cache "RSE" is destroyed
Could not create the Java virtual machine.
```

**Remarque :**

- Les utilitaires de cache effectuent les opérations nécessaires sur le cache indiqué sans démarrer la machine virtuelle Java, de sorte que le message "Could not create the Java virtual machine" soit normal.
- Un cache ne peut être détruit uniquement lorsque toutes les machines virtuelles JAVA qui l'utilisent sont arrêtées et que l'utilisateur qui émet la commande dispose d'autorisations suffisantes.





---

## Chapitre 7. Remarques relatives à la fonction d'envoi au client

La fonction d'envoi au client, ou contrôle client résidant sur l'hôte, prend en charge la gestion centralisée des éléments suivants :

- Fichiers de configuration client
- Version de produit client
- Définitions de projet

Les rubriques suivantes sont traitées dans le présent chapitre :

- «Introduction»
- «Système principal», à la page 126
- «Métadonnées d'envoi au client», à la page 127
- «Contrôle de la configuration client», à la page 128
- «Contrôle de la version client», à la page 129
- «Plusieurs groupes de développeurs», à la page 129
- «Sélection de groupe basé sur LDAP», à la page 133
- «Sélection de groupe basé sur SAF», à la page 138
- «Projets résidant sur l'hôte», à la page 141

---

### Introduction

Les clients Developer for System z version 8.0.1 et suivante peuvent extraire les fichiers de configuration client et les informations de mise à jour de produit depuis l'hôte lorsqu'ils se connectent, ce qui permet de garantir que tous les clients sont paramétrés de la même façon et qu'ils sont à jour.

Depuis la version 8.0.3, l'administrateur client peut créer plusieurs jeux de configuration client et plusieurs scénarios de mise à jour client afin de répondre aux besoins des différents groupes de développeurs. Cela permet aux utilisateurs de recevoir une configuration personnalisée, basée sur des critères tels que l'appartenance d'un groupe LDAP ou les droits d'accès à un profil de sécurité.

Les projets z/OS peuvent être définis de façon individuelle via la perspective Projets z/OS sur le client, ou de façon centralisée sur l'hôte, puis propagés individuellement sur le client pour chaque utilisateur. Ces projets résidant sur l'hôte ressemblent et fonctionnent exactement comme des projets définis sur le client, sauf que leur structure, leurs membres et leurs propriétés ne peuvent pas être modifiés par le client et qu'ils sont accessibles uniquement lorsque vous êtes connecté à l'hôte.

`pushtoclient.properties` indique au client si ces fonctions sont activées et précise l'emplacement des données associées. Pour plus d'informations, voir la rubrique "(Facultatif) `pushtoclient.properties`, contrôle du client basé sur un hôte" dans le document *Guide de configuration de l'hôte* (SC11-6285).

Généralement, les systèmes z/OS, les postes de travail de développeur et les projets de développement sont gérés par différents groupes de personnes. La conception de la fonction d'envoi au client suit ce principe et affecte des devoirs spécifiques à chaque groupe :

- Le programmeur système z/OS contrôle l'emplacement des métadonnées d'envoi au client, les aspects de sécurité de base et l'activation de la fonction d'envoi au client.
- L'administrateur client gère le contenu des métadonnées d'envoi au client en utilisant le client Developer for System z pour créer une ou plusieurs configurations client et en utilisant IBM Installation Manager pour créer les fichiers de réponses permettant de mettre à jour le client Developer for System z.
- Un responsable de projet de développement définit un projet et affecte chaque développeur à ce projet.

Pour plus d'informations sur la façon dont l'administrateur client et le responsable de projet de développement effectuent les tâches qui leur ont été affectées, voir le centre de documentation Developer for System z (<http://pic.dhe.ibm.com/infocenter/ratdevz/v9r0/index.jsp>).

Lorsque vous activez le support de contrôle de version ou de configuration pour plusieurs groupes de développeurs, une équipe supplémentaire est chargée de la gestion de la fonction d'envoi au client. L'équipe dédiée à cette tâche varie en fonction de l'option qui a été choisie pour identifier les groupes auxquels un développeur appartient. :

- Un administrateur LDAP gère les définitions de groupe qui placent chaque développeur dans aucun groupe LDAP, dans un groupe LDAP ou dans plusieurs groupes LDAP FEK.PTC.\*.
- Un administrateur de sécurité gère les listes d'accès aux profils de sécurité FEK.PTC.\*. Un développeur peut disposer de droits d'accès sur aucun profil, sur un profil ou sur tous les profils.

---

## Système principal

La fonction d'envoi au client est conçue pour stocker des données spécifiques à un système sur chaque système tout en conservant les données communes (globales) sur un seul système (système principal) afin de réduire l'effort de gestion. Le système principal est identifié par la directive `primary.system` dans `pushtoclient.properties`. La valeur par défaut est faux.

Vérifiez qu'un seul et unique système est défini comme système principal. Les administrateurs client Developer for System z ne peuvent exporter les données de configuration globales que si le système cible est un système principal. Les clients Developer for System z peuvent avoir un comportement incohérent lors de la connexion à plusieurs systèmes principaux avec des configurations désynchronisées.

La règle d'un seul système ne s'applique pas lorsque plusieurs systèmes partagent la configuration Developer for System z (`/etc/rdz`) et les métadonnées d'envoi au client (`/var/rdz/pushtoclient`). Etant donné que la configuration est partagée, tous les systèmes concernés sont identifiés comme étant le système principal. Toutefois, tant que tous les systèmes partagent également les métadonnées, cette duplication ne constitue pas un problème.

---

## Métadonnées d'envoi au client

### Emplacement des métadonnées

La directive `pushtoclient.folder` dans `pushtoclient.properties` identifie le répertoire de base dans lequel les métadonnées d'envoi au client sont stockées. La valeur par défaut est `/var/rdz/pushtoclient`.

Le répertoire de base contient le fichier de configuration de la fonction d'envoi au client `racine, keymapping.xml`. Toutes les autres métadonnées résident dans des sous-répertoires.

La plupart des sous-répertoires sont créés dynamiquement lorsque l'administrateur client exporte la configuration d'espace de travail de la fonction d'envoi au client. Ces sous-répertoires regroupent les métadonnées par objet, comme les mappages et les préférences. Comme davantage de composants client Developer for System z peuvent être gérés par la fonction d'envoi au client, un plus grand nombre de sous-répertoires est créé dynamiquement. Voir l'assistant d'exportation dans le client Developer for System z (**Fichier > Exporter... > Rational Developer for System z > Fichiers de configuration**) pour savoir ce que contiennent ces sous-répertoires.

Certains sous-répertoires sont créés pendant la personnalisation initiale de l'hôte. Ces sous-répertoires contiennent des données qui sont gérées manuellement par l'administrateur du client ou le gestionnaire du projet de développement.

- `/var/rdz/pushtoclient/projects/` contient les fichiers de définition du projet résidant sur l'hôte. L'emplacement réel est spécifié dans le répertoire `/var/rdz/pushtoclient/keymapping.xml`, lequel est créé et géré par un administrateur de client Developer for System z. Les fichiers qui s'y trouvent sont gérés par un chef de projet ou un responsable du développement.
- `/var/rdz/pushtoclient/install/` contient les fichiers de configuration utilisés pour mettre à jour la version du produit client lors de la connexion à l'hôte. L'emplacement réel est spécifié dans le répertoire `/var/rdz/pushtoclient/keymapping.xml`, lequel est créé et géré par un administrateur de client Developer for System z. Les fichiers qui s'y trouvent sont gérés par un administrateur de client.
- `/var/rdz/pushtoclient/install/responsefiles/` contient les fichiers de configuration utilisés pour mettre à jour la version du produit client lors de la connexion à l'hôte. L'emplacement réel est spécifié dans le répertoire `/var/rdz/pushtoclient/keymapping.xml`, lequel est créé et géré par un administrateur de client Developer for System z. Les fichiers qui s'y trouvent sont gérés par un administrateur de client.

Pour plus d'informations sur la création de ces sous-répertoires, voir la section sur la configuration de la personnalisation dans le chapitre sur la personnalisation de base dans le document *Guide de configuration de l'hôte* (SC11-6285).

### Métadonnées de sécurité

Par défaut (voir la directive `file.permission` dans `pushtoclient.properties`), tous les fichiers et répertoires créés dans le répertoire de base reçoivent le masque de contrôle des données de droits 775 (`rw-rw-r-x`), lequel octroie au propriétaire et au groupe par défaut du propriétaire un accès en lecture et en écriture à la structure de répertoire et aux fichiers qu'elle contient. Les autres utilisateurs ont uniquement un accès en lecture à la structure de répertoire et aux fichiers qu'elle contient.

Il est essentiel que l'ID utilisateur propriétaire et l'ID groupe appropriés soient définis pour ces répertoires avant de commencer à configurer la fonction d'envoi au client.

Les exemples de commande RACF ci-après permettent de créer un nouveau groupe (RDZADMIN), de lui affecter un ID groupe unique (2) et de le définir comme groupe par défaut pour l'ID utilisateur RDZADM1, lequel reçoit également un ID utilisateur unique (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))
```

L'exemple de commande **chown** z/OS UNIX ci-après permet de remplacer le propriétaire et le groupe de /var/rdz/pushtoclient et de toutes les données de ce répertoire par RDZADM1 et RDZADMIN, respectivement. La commande doit être exécutée par un superutilisateur (UID 0) afin d'éviter tout problème de droit.

```
chown -R rdzadm1:rdzadmin /var/rdz/pushtoclient
```

L'exemple de commande **chmod** z/OS UNIX ci-après permet de remplacer le masque de contrôle des données de droits de /var/rdz/pushtoclient et de toutes les données de ce répertoire par 775. Exécutez cette commande pour faire en sorte que tout ajout manuel effectué sur le répertoire respecte la logique utilisée par Developer for System z. La commande doit être exécutée par un superutilisateur (UID 0) afin d'éviter tout problème de droit.

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

Pour plus d'informations sur les exemples de commande RACF, voir *Security Server RACF Command Language Reference* (SA22-7687). Pour plus d'informations sur les exemples de commande z/OS UNIX, voir *UNIX System Services Command Reference* (SA22-7802). Pour toute information complémentaire, voir «Structure de répertoires z/OS UNIX», à la page 15.

## Utilisation de l'espace de métadonnées

Les métadonnées d'envoi au client utilisent un volume d'espace disque assez réduit dans z/OS UNIX, car elles correspondent à des fichiers XML codés en UTF-8.

Notez que le code produit utilisé pour les scénarios de mise à jour de client peut être stocké n'importe où sur le réseau ; il n'a pas besoin d'être stocké dans z/OS UNIX car les métadonnées d'envoi au client associées (appelées fichiers de réponses) pointent vers l'emplacement approprié pour le client.

---

## Contrôle de la configuration client

Lorsqu'un client Developer for System z (version 8.0.1 et suivante) se connecte à l'hôte, il lit les définitions dans pushtoclient.properties. Si la directive config.enabled est activée, le client compare sa configuration en cours aux définitions dans les métadonnées d'envoi au client. Si des différences sont détectées, le client démarre un assistant qui extrait les données requises et active la configuration comme indiqué par la fonction d'envoi au client.

La directive reject.config.updates dans pushtoclient.properties contrôle si un utilisateur est autorisé à rejeter les mises à jour de configuration que la fonction d'envoi au client est sur le point de distribuer.

Un client Developer for System z (version 8.0.1 et suivante) dispose d'un assistant (à l'intention de l'administrateur client) qui peut exporter la configuration en cours, laquelle est ensuite importée par tous les clients Developer for System z via la fonction d'envoi au client. Notez que cette fonction est disponible dans tous les clients. Par conséquent, vous devez vous assurer que seuls les administrateurs client disposent d'un droit d'accès en écriture sur les répertoires z/OS UNIX qui contiennent les métadonnées d'envoi au client (/var/rdz/pushtoclient).

La version 8.0.3 ou suivante est requise à la fois pour le client et pour l'hôte, de manière à permettre l'activation du support de groupe, comme indiqué dans le «Plusieurs groupes de développeurs».

---

## Contrôle de la version client

Lorsqu'un client Developer for System z (version 8.0.1 et suivante) se connecte à l'hôte, il lit les définitions dans `pushtoclient.properties`. Si la directive `product.enabled` est activée, le client compare sa version de produit en cours aux définitions dans les métadonnées d'envoi au client. Si des différences sont détectées, le client démarre un assistant qui extrait les données requises et active la configuration comme indiqué par la fonction d'envoi au client.

La directive `reject.product.updates` dans `pushtoclient.properties` contrôle si un utilisateur est autorisé à rejeter les mises à jour de produit que la fonction d'envoi au client est sur le point de distribuer.

La version 8.0.3 ou suivante est requise à la fois pour le client et pour l'hôte, de manière à permettre l'activation du support de groupe, comme indiqué dans «Plusieurs groupes de développeurs».

---

## Plusieurs groupes de développeurs

Depuis la version 8.0.3, l'administrateur client peut créer plusieurs jeux de configuration client et plusieurs scénarios de mise à jour client afin de répondre aux besoins des différents groupes de développeurs. Cela permet aux utilisateurs de recevoir une configuration personnalisée, basée sur des critères tels que l'appartenance d'un groupe LDAP ou les droits d'accès à un profil de sécurité.

### Activation

La prise en charge de plusieurs groupes de développeurs, chacun ayant sa propre configuration client et ses propres exigences de mise à jour de produit client, est activée en affectant la valeur souhaitée aux directives connexes (`config.enabled` et `product.enabled`) dans `pushtoclient.properties`, comme indiqué dans tableau 33.

Tableau 33. Support de groupe de la fonction d'envoi au client pour \*.enabled

| Valeur *.enabled | Fonction activée | Plusieurs groupes pris en charge                                                                  |
|------------------|------------------|---------------------------------------------------------------------------------------------------|
| Faux             | Non              | Non                                                                                               |
| Vrai             | Oui              | Non                                                                                               |
| LDAP             | Oui              | Yes, en fonction de l'appartenance des groupes LDAP<br>FEK.PTC.*.ENABLED.sysname.devgroup         |
| SAF              | Oui              | Oui, en fonction des droits d'accès aux profils de sécurité<br>FEK.PTC.*.ENABLED.sysname.devgroup |

Notez que lorsque la fonction est activée (valeur VRAI), les développeurs font toujours partie d'un groupe par défaut. Un développeur peut appartenir à aucun groupe, à un groupe ou à plusieurs groupes supplémentaires.

Le rejet des mises à jour peut également être conditionnel, comme indiqué dans le tableau 34.

*Tableau 34. Support de groupe de la fonction d'envoi au client pour reject.\*.updates*

| Valeur reject.*.updates | Fonction activée                                                                    |
|-------------------------|-------------------------------------------------------------------------------------|
| Faux                    | Non                                                                                 |
| Vrai                    | Oui                                                                                 |
| LDAP                    | Dépend de l'appartenance de groupe LDAP<br>FEK.PTC.REJECT.*.UPDATES.sysname         |
| SAF                     | Dépend des droits d'accès au profil de sécurité<br>FEK.PTC.REJECT.*.UPDATES.sysname |

Notez que les directives dans `pushtoclient.properties` fonctionnent indépendamment les unes des autres. Vous pouvez attribuer n'importe quelle valeur prise en charge aux directives. Il n'est pas nécessaire que les paramètres soient identiques.

Pour plus d'informations sur la configuration requise pour une fonction donnée, voir respectivement «Sélection de groupe basé sur LDAP», à la page 133 et «Sélection de groupe basé sur SAF», à la page 138. Pour plus d'informations sur l'activation du support de plusieurs groupes, voir la rubrique "(Facultatif) `pushtoclient.properties`, contrôle client résidant sur l'hôte" dans le document *Guide de configuration de l'hôte* (SC11-6285).

## Concaténations de groupe

Lorsque la fonction `*.enabled` est activée (valeur VRAI) dans `pushtoclient.properties`, les développeurs font toujours partie d'un groupe par défaut pour la fonction associée. Un développeur peut appartenir à aucun groupe, à un groupe ou à plusieurs groupes supplémentaires.

Pour rendre moins complexe l'application des modifications définies dans plusieurs groupes, Developer for System z limite les définitions qui seront utilisées, sur la base d'une sélection effectuée par l'utilisateur.

*Tableau 35. Concaténations de groupe pour la fonction d'envoi au client*

| Groupes supplémentaires | Définitions utilisées                               |
|-------------------------|-----------------------------------------------------|
| Aucun                   | Valeur par défaut                                   |
| Un                      | Valeur par défaut ou (valeur par défaut + groupe)   |
| Plusieurs               | Valeur par défaut ou (valeur par défaut + 1 groupe) |

Developer for System z utilise la logique suivante lors de la création et de l'application de l'ensemble d'artefacts modifiés :

1. Il applique les mises à jour éventuellement spécifiées dans les définitions par défaut.



2. Il applique les mises à jour éventuellement spécifiées dans la définition de groupe sélectionnée, en remplaçant les éventuelles mises à jour par défaut existantes.
3. Il applique les mises à jour sur le client.

**Remarque :** Les mises à jour peuvent correspondre à des actions de suppression, d'ajout et de chevauchement.

## Liaison d'espace de travail

Même si un développeur peut faire partie de plusieurs groupes simultanément, son espace de travail actif ne peut pas. L'espace de travail actif doit être lié à un groupe spécifique (lequel peut être le groupe par défaut) pour recevoir les mises à jour de configuration ou de produit. Une fois la liaison effectuée, elle ne peut plus être annulée. Un nouvel espace de travail doit être créé si une nouvelle liaison de groupe est requise.

Lorsqu'un espace de travail sans liaison de groupe se connecte à l'hôte et que `config.enabled` (ou `product.enabled`) indique que la fonction d'envoi au client est active, Developer for System z interroge tous les groupes pour identifier ceux auxquels l'utilisateur appartient et il invite ce dernier à sélectionner un groupe pour la fonction associée. Lors des connexions successives, seul le groupe sélectionné est interrogé pour vérifier si l'appartenance au groupe est toujours valide.

Les directives `reject.*.updates` ne fonctionnent pas avec plusieurs groupes, ainsi, leur configuration est plus simple et ne nécessite aucune liaison d'espace de travail. Lorsqu'une mise à jour existe, Developer for System z détermine si l'utilisateur est autorisé ou non à la rejeter et agit en conséquence.

## Emplacement des métadonnées de groupe

Comme indiqué dans «Emplacement des métadonnées», à la page 127, toutes les métadonnées d'envoi au client sont stockées dans une structure de répertoire en haut de `/var/rdz/pushtoclient/` lorsque vous utilisez une configuration sans support de groupe. La présentation des données est conservée lorsque le support de groupe est activé, mais l'interprétation du répertoire de base, `/var/rdz/pushtoclient/`, est légèrement différente :

- Les données qui se trouvent dans `/var/rdz/pushtoclient/` sont interprétées comme les données du groupe par défaut. L'exportation vers le groupe par défaut crée ou met à jour les métadonnées dans `/var/rdz/pushtoclient/`. Cette interprétation garantit la compatibilité avec les clients version 8.0.1 et version 8.0.2, pour lesquels la fonction d'envoi au client est activée, mais qui ne prennent pas en charge plusieurs groupes.
- L'exportation vers un groupe crée ou met à jour les métadonnées dans `/var/rdz/pushtoclient/grouping/<devgroup>/`, comme s'il s'agissait du répertoire de base au lieu de `/var/rdz/pushtoclient/`. La valeur `<devgroup>` correspond au nom de groupe affecté à un groupe spécifique de développeurs.

La personnalisation de produit initiale crée le répertoire `grouping/` dans `var/rdz/pushtoclient/`. L'administrateur client est chargé d'ajouter les répertoires `<devgroup>/` dans `/var/rdz/pushtoclient/grouping/`.

Notez que lors de la personnalisation de produit initiale, les répertoires `projects/`, `install/`, et `install/responsefiles/` sont créés dans `/var/rdz/pushtoclient/`. L'administrateur client doit répéter ces actions de création de répertoire dans

/var/rdz/pushtoclient/grouping/<devgroup>/ si des scénarios de mise à niveau de produit spécifiques à un groupe ou des projets basés sur l'hôte spécifiques à un groupe sont requis.

L'exemple de séquence de commande z/OS UNIX suivant crée les sous-répertoires avec le masque de contrôle des données de droits approprié. Les commandes doivent être exécutées par l'administrateur client pour éviter tout problème de propriété.

```
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
```

Pour plus d'informations sur les exemples de commande z/OS UNIX, voir *UNIX System Services Command Reference* (SA22-7802).

## Etapas de configuration

La configuration du support de plusieurs groupes de développeurs nécessite une certaine coordination entre le programmeur système z/OS, l'administrateur client et l'administrateur qui gère les critères de sélection (administrateur de la sécurité ou du serveur LDAP). Le flux de travaux décrit ci-après illustre la gestion des critères de sélection par l'administrateur de la sécurité.

1. L'administrateur client demande à l'administrateur de la sécurité des informations sur la configuration de regroupement de développeurs existante. Le fait de réutiliser la configuration existante accélère et simplifie la configuration de la fonction d'envoi au client.
2. L'administrateur client détermine la façon dont il veut structurer le support de plusieurs groupes et identifie les utilisateurs qui doivent faire partie de ces groupes d'envoi au client.

### Remarque :

- Il existe toujours un ensemble de configuration par défaut et un scénario de mise à jour de produit par défaut.
  - Les ensembles d'artefacts modifiés de la fonction d'envoi au client peuvent correspondre à des actions de suppression, d'ajout et de remplacement.
  - Les ensembles d'artefacts modifiés de la fonction d'envoi au client peuvent être vides.
  - Un développeur peut appartenir à aucun groupe, à un groupe ou à plusieurs groupes d'envoi au client.
  - L'administrateur client doit être un membre de chaque groupe d'envoi au client.
3. L'administrateur client et l'administrateur de la sécurité s'entendent sur les noms de groupe d'envoi au client à utiliser.
  4. L'administrateur client crée le répertoire  
/var/rdz/pushtoclient/grouping/<devgroup>

pour chaque groupe de fonction d'envoi au client.

**Remarque :** Les bits des droits pour ce répertoire doivent être 775 (drwxrwxr-x).



5. L'administrateur de la sécurité effectue la configuration initiale requise pour définir les profils de critères de sélection pour l'envoi au client et ajoute les groupes d'envoi au client à la liste d'accès.

**Remarque :**

- Les structures de critères de sélection doivent être définies de telle façon qu'au moins l'administrateur client doit être inclus dans la liste d'accès pour pouvoir créer les métadonnées d'envoi au client associées.
  - Pour la configuration initiale, seul l'administrateur client doit figurer dans la liste d'accès pour un groupe d'envoi au client. Ceci a pour objectif d'éviter que les clients Developer for system z reçoivent des configurations qui sont en cours de construction.
6. Le programmeur système z/OS active le support de plusieurs groupes en ajustant `pushtoclient.properties`.

**Remarque :** Les directives `*.enabled` doivent être activées pour que l'administrateur client puisse créer les métadonnées d'envoi au client associées.

7. L'administrateur client crée les espaces de travail pour chaque groupe et les exporte vers le système hôte à l'aide des noms de groupe respectifs. L'administrateur client crée également les fichiers de réponses nécessaires à la création des scénarios de mise à jour de produit spécifiques aux groupes.
8. L'administrateur de la sécurité ajoute les développeurs aux groupes d'envoi au client, ce qui active la fonction d'envoi au client pour les développeurs.

---

## Sélection de groupe basé sur LDAP

Même si LDAP (Lightweight Directory Access Protocol) est le nom d'un protocole basé sur TCP/IP, il est généralement utilisé pour décrire un ensemble de services d'annuaire distribué. Comme une base de données, un annuaire est un ensemble structuré d'enregistrements. Developer for System z peut utiliser un serveur LDAP comme une base de données hiérarchique simple, dans laquelle des groupes contiennent un ou plusieurs membres.

Lorsque vous utilisez des définitions dans votre serveur LDAP comme mécanisme de sélection (la valeur LDAP est spécifiée pour les directives dans `pushtoclient.properties`), Developer for System z vérifie l'appartenance aux groupes répertoriés dans le tableau 36 pour identifier les groupes de développeurs auxquels l'utilisateur appartient et déterminer si un utilisateur est autorisé à rejeter les mises à jour.

*Tableau 36. Informations LDAP pour la fonction d'envoi au client*

| Nom de groupe (cn=)                      | Résultat                                                                   |
|------------------------------------------|----------------------------------------------------------------------------|
| FEK.PTC.CONFIG.ENABLED.sysname.devgroup  | Le client accepte les mises à jour de configuration pour le groupe indiqué |
| FEK.PTC.PRODUCT.ENABLED.sysname.devgroup | Le client accepte les mises à jour de produit pour le groupe indiqué       |
| FEK.PTC.REJECT.CONFIG.UPDATES.sysname    | L'utilisateur peut rejeter les mises à jour de configuration               |
| FEK.PTC.REJECT.PRODUCT.UPDATES.sysname   | L'utilisateur peut rejeter les mises à jour de produit                     |

La valeur devgroup correspond au nom de groupe affecté à un groupe spécifique de développeurs. Notez que le nom de groupe est visible sur les clients Developer for System z.

La valeur sysname correspond au nom du système cible.

## Schéma LDAP

Le schéma LDAP doit respecter les règles suivantes :

1. Chaque groupe de la fonction d'envoi au client doit être défini en tant que groupe dans le schéma.
2. Chaque utilisateur doit être défini en tant qu'utilisateur dans le schéma.
3. Une entrée de groupe comporte les références des entrées utilisateur appartenant à son propre groupe.

La figure 26 est un exemple de définition LDAP pour un groupe et un utilisateur, exprimé au format LDIF.

**Remarque :** Le format LDIF (LDAP Data Interchange Format) est un format de texte standard utilisé pour représenter les objets et les mises à jour LDAP. Les fichiers contenant des enregistrements LDIF sont utilisés pour transférer des données entre des serveurs d'annuaire ou en tant qu'entrées par les utilitaires LDAP.

```
Group Definition
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

User Definition
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
```

Figure 26. Exemple de définition de schéma LDAP

## Sélection de serveur LDAP

Il existe un large choix de serveurs LDAP commerciaux et gratuits. Par exemple, IBM Tivoli Directory Server (<http://www-01.ibm.com/software/tivoli/products/directory-server/>). Il existe également un large choix d'outils de type interface graphique et ligne de commande qui permettent de gérer un serveur LDAP.

Comme indiqué dans «Schéma LDAP», chaque utilisateur doit être défini sur le serveur LDAP. Pour réduire l'effort de gestion, il est préférable de placer le schéma d'envoi au client sur un serveur LDAP qui a déjà accès à toutes les définitions d'utilisateur. Par exemple, vous pouvez utiliser un serveur IBM Tivoli Directory

Server actif sous z/OS à l'aide d'une base de données SDBM (laquelle sert d'encapsuleur pour votre base de données de sécurité).

Selon les règles appliquées sur le site, le schéma d'envoi au client sur le serveur LDAP peut être géré par l'administrateur client. Cet arrangement permet de réduire les besoins en termes de collaboration, ainsi que les retards et erreurs de communication éventuels.

L'un des arguments en faveur de la gestion LDAP par l'administrateur client est que le schéma d'envoi au client ne contient aucune donnée confidentielle ni aucune information liée à la sécurité. Lorsque des définitions utilisateur sont disponibles sur le serveur LDAP via d'autres schémas, les objets LDAP Developer for System z identifient simplement les choix dont dispose un développeur pour la sélection d'une présentation d'espace de travail et des mises à niveau de produit client Developer for System z automatiques.

## Emplacement de serveur LDAP

Tout serveur de base de données qui prend en charge le protocole LDAP peut être utilisé pour héberger le schéma d'envoi au client Developer for System z. Par conséquent, Developer for System z vous permet de spécifier les informations nécessaires à la connexion au serveur LDAP. Vous pouvez également spécifier le suffixe qui rend la base de données unique dans le serveur LDAP.

| Directive rsed.envvars     | Valeur par défaut       |
|----------------------------|-------------------------|
| _RSE_LDAP_SERVER           | Système hôte local      |
| _RSE_LDAP_PORT             | 389                     |
| _RSE_LDAP_PTC_GROUP_SUFFIX | "O=PTC,C=DeveloperForZ" |

Notez que les mesures de sécurité TCP/IP, telles que des pare-feux, peuvent empêcher le serveur RSE (résidant sur l'hôte) de contacter le serveur LDAP. Contactez l'administrateur TCP/IP et communiquez-lui les informations suivantes pour faire en sorte que le serveur LDAP puisse être atteint :

- Adresse TCP/IP ou nom DNS du serveur LDAP
- Numéro de port du serveur LDAP
- LDAP utilise le protocole TCP
- Le serveur LDAP est contacté par le serveur RSE résidant sur l'hôte
- Le serveur RSE est actif dans un espace adresse RSEDx, où RSED est le nom de la tâche démarrée RSE et x est un nombre à un chiffre aléatoire

## Exemple de configuration

Supposons que Developer for System z soit actif sur le système CDFMVS08. IBM Tivoli Directory Server, également actif sur CDFMVS08, est utilisé en tant que serveur LDAP. Le serveur LDAP est configuré comme indiqué dans «Ajout à LDAP d'une section dorsale pour la fonction d'envoi au client», à la page 136.

Les utilisateurs suivants utilisent Developer for System z :

- Les développeurs qui travaillent sur des applications bancaires, ID utilisateur BNK010 -> BNK014
- Les développeurs qui travaillent sur des applications d'assurance, ID utilisateur INS010 -> INS014
- Un administrateur client Developer for System z, ID utilisateur RDZADM1

Chaque groupe de développeurs nécessite des fichiers de configuration client spécifiques, et tous les développeurs sont soumis au même contrôle de version client. Contrairement aux administrateurs client, les développeurs ne sont pas autorisés à rejeter les modifications présentées par la fonction d'envoi au client.

L'administrateur client et l'administrateur LDAP s'entendent pour utiliser les noms de groupe BANKING et INSURANCE pour les mises à jour de configuration.

### Ajout à LDAP d'une section dorsale pour la fonction d'envoi au client

Dans cet exemple, des mises à jour sont apportées à IBM Tivoli Directory Server sous z/OS, qui utilise actuellement uniquement une base de données SDBM (encapsuleur de base de données de sécurité), en ajoutant une base de données LDBM (fichiers z/OS UNIX) pour héberger le schéma de la fonction d'envoi au client.

1. Ajoutez la section dorsale LDBM au fichier de configuration LDAP.

```
filename ds.conf
restart GLDSRV started task to pick up changes

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. Arrêtez et démarrez la tâche démarrée LDAP, GRDSRV, pour appliquer les modifications de configuration.

3. Créez le répertoire /var/ldap/ldbm/rdz.

```
mkdir -p /var/ldap/ldbm/rdz
```

4. Mettez le schéma LDAP à jour pour ajouter la section dorsale 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. Ajoutez l'entrée racine à la section dorsale LDBM.

```
ldapadd -D "cn=LDAP admin" -w password -f
/u/ibmuser/ptc_root.ldif
```

où /u/ibmuser/ptc\_root.ldif contient les éléments suivants :

```
dn: o=PTC,c=DeveloperForZ
objectclass: top
objectclass: organization
o: PTC
```

### Configuration de groupe LDAP initiale

Ajoutez les différents objets groupe LDAP au schéma, puis associez l'administrateur client à chacun d'eux. La définition d'utilisateur de l'ID utilisateur RDZADM1 est extraite du schéma RACF.

```
ldapadd -D "cn=LDAP admin" -w password -f /u/ibmuser/ptc_setup.ldif
```

où /u/ibmuser/ptc\_setup.ldif contient les éléments suivants :

```
banking workspace configuration
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
give client administrator access
uniqueMember: racfID=RDZADM1,profileType=user,cn=RACF,o=IBM,c=US

insurance workspace configuration
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
give client administrator access
uniqueMember: racfID=RDZADM1,profileType=user,cn=RACF,o=IBM,c=US

reject configuration updates
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
give client administrator access
uniqueMember: racfID=RDZADM1,profileType=user,cn=RACF,o=IBM,c=US

reject product updates
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
give client administrator access
uniqueMember: racfID=RDZADM1,profileType=user,cn=RACF,o=IBM,c=US
```

## Ajout de développeurs à des groupes LDAP

Ajoutez les développeurs aux objets groupe LDAP. Les définitions d'utilisateur des ID utilisateur sont extraites du schéma RACF.

```
ldapmodify -D "cn=LDAP admin" -w password -f /u/ibmuser/ptc_add.ldif
```

où /u/ibmuser/ptc\_add.ldif contient les éléments suivants :

```
banking workspace configuration
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

insurance workspace configuration
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 and INSURANCE have different configuration needs
config.enabled=LDAP
everyone receives product updates
product.enabled=TRUE
```

```
only RDZADMIN can reject configuration updates
reject.config.updates=LDAP
only RDZADMIN can reject product updates
reject.product.updates=LDAP
```

### **rsed.envvars**

Aucune mise à jour n'est requise car les valeurs par défaut sont utilisées :

- `_RSE_LDAP_SERVER=CDFMVS08.RALEIGH.IBM.COM`
- `_RSE_LDAP_PORT=389`
- `_RSE_LDAP_PTC_GROUP_SUFFIX="o=PTC,c=DeveloperForZ"`

### **/var/rdz/pushtoclient/\*install**

Lors de l'exportation de la configuration d'espace de travail pour les groupes BANKING et INSURANCE, l'assistant d'exportation crée les répertoires `/var/rdz/pushtoclient/grouping/<devgroup>/` et la structure de répertoire sous-jacente.

- `/var/rdz/pushtoclient/grouping/BANKING/*`
- `/var/rdz/pushtoclient/grouping/INSURANCE/*`

Etant donné qu'il n'existe pas de scénarios de mise à niveau de produit individualisés, l'administrateur client n'a pas besoin de créer ou mettre à jour les sous-répertoires `install/` et `install/responsefiles/` de `/var/rdz/pushtoclient/grouping/<devgroup>`.

L'administrateur client doit créer les fichiers de réponses nécessaires aux mises à jour de produit dans le répertoire de groupe par défaut, `/var/rdz/pushtoclient/install/responsefiles/`.

---

## **Sélection de groupe basé sur SAF**

SAF (Security Access Facility) est une interface qui permet d'accéder à n'importe quel produit de sécurité z/OS. Developer for System z peut utiliser cette interface pour interroger votre produit de sécurité et extraire les informations liées à la fonction d'envoi au client.

Lorsque vous utilisez des définitions dans votre base de données de sécurité comme mécanisme de sélection (la valeur SAF est spécifiée pour les directives dans `pushtoclient.properties`), Developer for System z vérifie les droits d'accès aux profils répertoriés dans le tableau 37 pour identifier les groupes de développeurs auxquels l'utilisateur appartient et déterminer si un utilisateur est autorisé à rejeter les mises à jour.

*Tableau 37. Informations SAF pour la fonction d'envoi au client*

| Profil FACILITY                              | Longueur fixe | Droit d'accès requis | Résultat                                                                   |
|----------------------------------------------|---------------|----------------------|----------------------------------------------------------------------------|
| FEK.PTC.CONFIG.ENABLED.<br>sysname.devgroup  | 23            | READ                 | Le client accepte les mises à jour de configuration pour le groupe indiqué |
| FEK.PTC.PRODUCT.ENABLED.<br>sysname.devgroup | 24            | READ                 | Le client accepte les mises à jour de produit pour le groupe indiqué       |

Tableau 37. Informations SAF pour la fonction d'envoi au client (suite)

| Profil FACILITY                            | Longueur fixe | Droit d'accès requis | Résultat                                                     |
|--------------------------------------------|---------------|----------------------|--------------------------------------------------------------|
| FEK.PTC.REJECT.CONFIG.<br>UPDATES.sysname  | 30            | READ                 | L'utilisateur peut rejeter les mises à jour de configuration |
| FEK.PTC.REJECT.PRODUCT.<br>UPDATES.sysname | 31            | READ                 | L'utilisateur peut rejeter les mises à jour de produit       |

**Remarque :** Developer for System z suppose qu'un utilisateur ne dispose d'aucun droit d'accès lorsque votre logiciel de sécurité indique qu'il ne peut pas déterminer si un utilisateur dispose ou non des droits d'accès à un profil. Cela se produit par exemple lorsque le profil n'est pas défini.

La valeur devgroup correspond au nom de groupe affecté à un groupe spécifique de développeurs. Notez que le nom de groupe est visible sur les clients Developer for System z.

La valeur sysname correspond au nom du système cible.

La colonne "Longueur fixe" indique la longueur de la partie fixe du profil de sécurité associée.

Par défaut, Developer for System z s'attend à ce que les profils FEK.\* résident dans la classe de sécurité FACILITY. Notez que les profils figurant dans la classe FACILITY sont limités à 39 caractères. Si la somme de la longueur de la partie fixe de profil (FEK.PTC.<key>.) et de la longueur de la partie de profil spécifique au site (sysname ou sysname.devgroup) est supérieure à ce nombre, vous pouvez placer les profils dans une autre classe et indiquer à Developer for System z que cette dernière doit être utilisée. Pour ce faire, mettez en commentaires `_RSE_FEK_SAF_CLASS` dans `rsed.envvars` et indiquez le nom de classe de votre choix.

## Exemple de configuration

Supposons que Developer for System z soit actif sur le système CDFMVS08. La base de données de sécurité RACF est partagée entre plusieurs systèmes et les groupes suivants sont définis dans la base de données de sécurité :

- DEVBANK : développeurs qui travaillent sur des applications bancaires
- DEVINSUR : développeurs qui travaillent sur des applications d'assurance
- RDZADMIN : administrateurs client de Developer for System z

Chaque groupe de développeurs nécessite des fichiers de configuration client spécifiques, et tous les développeurs sont soumis au même contrôle de version client. Contrairement aux administrateurs client, les développeurs ne sont pas autorisés à rejeter les modifications présentées par la fonction d'envoi au client. La règle de rejet est valide pour tous les systèmes en vue d'une expansion ultérieure.

L'administrateur de la sécurité et l'administrateur client s'entendent pour utiliser les noms de groupe d'envoi au client BANKING et INSURANCE pour les mises à jour de configuration.



## Définition de sécurité

```
allow RDZADMIN and DEVBANK to select push-to-client group BANKING
RDEFINE FACILITY (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(FACILITY) -
 ID(RDZADMIN DEVBANK) ACCESS(READ)

allow RDZADMIN and DEVINSUR to select push-to-client group INSURANCE
RDEFINE FACILITY (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(FACILITY) -
 ID(RDZADMIN DEVINSUR) ACCESS(READ)

RDZADMIN can reject configuration updates on any system
RDEFINE FACILITY (FEK.PTC.REJECT.CONFIG.UPDATES.*) -
 UACC(NONE) DATA('RATIONAL DEVELOPER FOR SYSTEM Z - PUSH-TO-CLIENT')
PERMIT FEK.PTC.REJECT.CONFIG.UPDATES.* CLASS(FACILITY) -
 ID(RDZADMIN) ACCESS(READ)

RDZADMIN can reject product updates on any system
RDEFINE FACILITY (FEK.PTC.REJECT.PRODUCT.UPDATES.*) -
 UACC(NONE) DATA('RATIONAL DEVELOPER FOR SYSTEM Z - PUSH-TO-CLIENT')
PERMIT FEK.PTC.REJECT.CONFIG.UPDATES.* CLASS(FACILITY) -
 ID(RDZADMIN) ACCESS(READ)

activate changes
SETROPTS RACLIST(FACILITY) REFRESH
```

## pushtoclient.properties

```
BANKING and INSURANCE have different configuration needs
config.enabled=SAF
everyone receives product updates
product.enabled=TRUE
only RDZADMIN can reject configuration updates
reject.config.updates=SAF
only RDZADMIN can reject product updates
reject.product.updates=SAF
```

## rsed.envvars

Aucune mise à jour n'est requise car les valeurs par défaut sont utilisées :

```
_RSE_FEK_SAF_CLASS=FACILITY
```

## /var/rdz/pushtoclient/\*install

Lors de l'exportation de la configuration d'espace de travail pour les groupes BANKING et INSURANCE, l'assistant d'exportation crée les répertoires /var/rdz/pushtoclient/grouping/<devgroup>/ et la structure de répertoire sous-jacente.

- var/rdz/pushtoclient/grouping/BANKING/\*
- /var/rdz/pushtoclient/grouping/INSURANCE/\*

Etant donné qu'il n'existe pas de scénarios de mise à niveau de produit individualisés, l'administrateur client n'a pas besoin de créer ou mettre à jour les sous-répertoires install/ et install/responsefiles/ de /var/rdz/pushtoclient/grouping/<devgroup>/.

L'administrateur client doit créer les fichiers de réponses nécessaires aux mises à jour de produit dans le répertoire de groupe par défaut, /var/rdz/pushtoclient/install/responsefiles/.



## Délai de grâce pour le rejet des modifications

Supposons que pendant que l'exemple de configuration est en vigueur, un groupe de correctifs Developer for System z comportant des correctifs majeurs devienne disponible. Toutefois, le calendrier d'un projet bancaire est tel qu'il est possible que plusieurs développeurs soient lassés de devoir apporter des modifications sur leur poste de travail pour le moment.

Pour résoudre ce problème, l'administrateur de la sécurité peut accorder à tous les développeurs DEVBANK un délai de grâce pendant lequel ils peuvent différer (rejeter) la mise à jour.

La configuration du délai de grâce est très simple à réaliser :

```
start of grace period
PERMIT FEK.PTC.REJECT.PRODUCT.UPDATES.* CLASS(FACILITY) –
 ID(DEVBANK) ACCESS(READ)
```

```
activate changes
SETROPTS RACLIST(FACILITY) REFRESH
```

A la fin du délai de grâce, les droits d'accès supplémentaires peuvent être supprimés à nouveau :

```
end of grace period
PERMIT FEK.PTC.REJECT.PRODUCT.UPDATES.* CLASS(FACILITY) –
 ID(DEVBANK) DELETE
```

```
activate changes
SETROPTS RACLIST(FACILITY) REFRESH
```

---

## Projets résidant sur l'hôte

Les projets z/OS peuvent être définis de façon individuelle via la perspective Projets z/OS sur le client, ou de façon centralisée sur l'hôte, puis propagés individuellement sur le client pour chaque utilisateur. Ces projets résidant sur l'hôte ressemblent et fonctionnent exactement comme des projets définis sur le client, sauf que leur structure, leurs membres et leurs propriétés ne peuvent pas être modifiés par le client et qu'ils sont accessibles uniquement lorsque vous êtes connecté à l'hôte.

Le répertoire de base pour les projets résidant sur l'hôte est défini (par l'administrateur client) dans `/var/rdz/pushtoclient/keymapping.xml`. Il s'agit du répertoire par défaut `/var/rdz/pushtoclient/projects`.

Pour configurer des projets résidant sur l'hôte, le responsable de projet ou le développeur doit définir les types de fichier de configuration suivants. Tous les fichiers sont des fichiers XML codés en UTF-8.

- Les fichiers d'instance de projet sont spécifiques à un ID utilisateur unique et pointent vers des fichiers de définition de projet réutilisables. Chaque utilisateur qui fonctionne avec des projets résidant sur l'hôte a besoin d'un sous-répertoire, `/var/rdz/pushtoclient/projects/<userid>/`, qui contient un fichier d'instance de projet (`*.hbpin`) pour chaque projet à télécharger.
- Les fichiers de définition de projet définissent la structure et le contenu du projet et sont réutilisables par plusieurs utilisateurs. Les fichiers de définition de projet (`*.hbppd`) répertorient les sous-projets contenus par le projet et sont situés dans le répertoire de définition de projet racine ou dans l'un de ses sous-répertoires.
- Les fichiers de définition de sous-projet définissent la structure et le contenu du sous-projet et sont réutilisables par plusieurs utilisateurs. Les fichiers de

définition de sous-projet (\*.hbpsd) définissent l'ensemble de ressources nécessaires pour générer un module de chargement et sont situés dans le répertoire de définition de projet racine ou dans l'un de ses sous-répertoires.

- Les fichiers de propriétés de sous-projet sont des fichiers de propriétés avec prise en charge de substitution de variable réutilisables par plusieurs sous-projets. Les fichiers de propriétés de sous-projet (\*.hbppr) prennent en charge la substitution de variable afin de permettre le partage de fichiers de propriétés entre plusieurs utilisateurs et sont situés dans le répertoire de définition de projet racine ou dans l'un de ses sous-répertoires.

Les projets résidant sur l'hôte peuvent également être sélectionnés pour faire partie de la configuration de plusieurs groupes (présentée dans «Plusieurs groupes de développeurs», à la page 129). Cela signifie que les projets résidant sur l'hôte peuvent également être définis dans /var/rdz/pushtoclient/grouping/<devgroup>/projects/.

Lorsqu'un espace de travail est lié à un groupe spécifique et qu'il existe des définitions de projet pour un utilisateur dans ce groupe et dans le groupe par défaut, l'utilisateur reçoit les définitions de projet depuis ces deux groupes.

---

## Chapitre 8. Remarques relatives à CICSTS

Habituellement, la définition des ressources dans CICS est gérée par un administrateur système CICS. Autoriser les développeurs d'applications à définir les ressources CICS n'est pas si simple, pour les raisons suivantes :

- La plupart des définitions de ressources CICS se composent de nombreux paramètres qui, par leur complexité, leurs liens avec d'autres définitions de ressource et les normes d'usine, nécessitent des connaissances en administration CICS pour effectuer une définition correcte. Une mauvaise définition risque d'entraîner des résultats imprévus qui peuvent avoir une incidence sur la totalité de la région CICS.
- La plupart des magasins fournissent des environnements de test et de développement CICS qui doivent être disponibles pour une utilisation partagée par plusieurs développeurs et groupes d'applications. De nombreux magasins ont mis en place des contrats de service pour ces environnements. Un contrôle strict des environnements est nécessaire pour satisfaire ces contrats.

Developer for System z traite ces problèmes en permettant aux administrateurs CICS de contrôler les paramètres par défaut de définition de ressource CICS ainsi que les propriétés d'affichage d'un paramètre de définition de ressource CICS à l'aide du serveur de définition de ressource CICS, qui fait partie du gestionnaire de déploiement d'application.

Par exemple, l'administrateur CICS peut fournir certains paramètres de définition de ressource CICS qui risquent de ne pas avoir été mis à jour par le développeur d'applications. D'autres paramètres de définition de ressource CICS sont réactualisables, avec ou sans les valeurs fournies par défaut, ou le paramètre de définition de ressource CICS peut être masqué pour éviter toute complexité inutile.

Une fois que le développeur d'applications est satisfait des définitions de ressource CICS, il peut les installer immédiatement dans l'environnement de test CICS en cours d'exécution ou les exporter dans un manifeste pour qu'un administrateur CICS puisse les éditer et les valider. L'administrateur CICS peut utiliser l'utilitaire d'administration (utilitaire de traitement par lots) ou l'outil de traitement de manifestes pour mettre en oeuvre les modifications de définition de ressource.

**Remarque :** L'outil de traitement de manifestes est un module d'extension pour IBM CICS Explorer.

Voir "(Facultatif) Gestionnaire de déploiement d'application" dans *Guide de configuration de l'hôte* (SC11-6285) pour plus d'informations sur les tâches nécessaires pour configurer le gestionnaire de déploiement d'application sur votre système hôte.

La personnalisation du gestionnaire de déploiement d'application ajoute les services suivants à Developer for System z :

- (Sur le client) IBM CICS Explorer fournit une infrastructure Eclipse pour afficher et gérer les ressources CICS et permet d'effectuer une intégration plus étroite entre les outils CICS
- (Sur le client) Editeur CRD (CICS Resource Definition)
- (Sur l'hôte) Serveur CRD (CICS Resource Definition) qui s'exécute comme application CICS

Le serveur CRD (CICS Resource Definition) du gestionnaire de déploiement d'application est constitué du serveur CRD, d'un référentiel CRD, des définitions de ressource CICS associées et, lorsque vous utilisez l'interface Web Service, des fichiers de liaison Web Service et un exemple de gestionnaire de messages de pipeline. Le serveur CRD doit être exécuté dans une région gérant le Web (WOR, Web Owning Region) référencée dans la documentation Developer for System z comme région de connexion principale CICS.

Pour en savoir plus sur les services du gestionnaire de déploiement d'application disponibles dans l'édition en cours de Developer for System z, voir le centre de documentation Developer for System z (<http://pic.dhe.ibm.com/infocenter/ratdevz/v9r0/index.jsp>).

---

## RESTful par opposition à Web Service

CICS Transaction Server fournit dans la version 4.1 et les versions suivantes le support d'une interface HTTP conçue en utilisant les principes RESTful (Representational State Transfer). Cette interface RESTful est désormais l'interface CICSTS stratégique utilisée par les applications client. L'ancienne interface Web Service a été stabilisée, les améliorations concernant uniquement l'interface RESTful.

Le gestionnaire de déploiement d'application suit cette logique et demande au serveur CRD RESTful tous les nouveaux services de Developer for System version 7.6 ou ultérieures.

Les interfaces RESTful et Web Service peuvent être activées simultanément dans une seule région CICS, si nécessaire. Dans ce cas, la région contient deux serveurs CRD actifs. Les deux serveurs partagent le même référentiel de CRD. Notez que CICS émet des avertissement sur les définitions dupliquées lorsque la seconde interface est définie dans la région.

---

## Régions de connexion primaires versus régions de connexion non primaires

Un environnement de test CICS peut se composer de plusieurs régions connectées par MRO (exploitation multirégionale). Au cours du temps, des désignations non officielles ont été utilisées pour catégoriser ces régions. Les désignations standard sont : région TOR (région gérant les terminaux), région WOR (région gérant le Web), région AOR (région gérant les applications) et région DOR (région gérant les données).

Une région gérant le Web est utilisée pour mettre en oeuvre la prise en charge des services Web CICS et le serveur de définition de ressource CICS du gestionnaire de déploiement d'application doit être exécuté dans cette région. Cette région est connue dans le Gestionnaire de déploiement d'application en tant que région de connexion CICS primaire. Le client CRD ADM implémente une connexion de service Web à la région primaire de connexion CICS.

Les régions de connexion non primaires CICS constituent toutes les autres régions que le serveur CRD peut gérer. Ce service englobe l'affichage des ressources à l'aide d'IBM CICS Explorer et des ressources de définition à l'aide de l'éditeur de définition de ressource CICS.

Si les services d'application métier (BAS) SM CICSplex sont utilisés pour gérer les définitions de ressource CICS de la région de connexion primaire CICS, toutes les autres régions CICS gérées par BAS peuvent être gérées par le serveur CRD.

Les régions CICS non gérées par BAS requièrent des modifications supplémentaires afin de pouvoir être gérées par le serveur CRD.

---

## Consignation des messages d'installation des ressources CICS

Les actions effectuées par le serveur CRD sur les ressources CICS sont consignées dans la file d'attente TD CSDL CICS qui indique généralement la définition de données MSGUSR de votre région CICS.

Si CICSplex SM Business Application Services (BAS) est utilisé pour gérer les définitions de ressource CICS, la directive CICSplex SM EYUPARM BASLOGMSG doit avoir pour valeur (YES) pour que la consignation soit activée.

---

## Gestionnaire de déploiement d'application, sécurité

### CRD, sécurité du référentiel

Le fichier VSAM du référentiel du serveur CRD contient toutes les définitions de ressource par défaut ; il doit par conséquent être protégé contre les mises à jour tout en autorisant les développeurs à consulter les valeurs qui y sont conservées. Pour protéger le référentiel CRD, reportez-vous à «Définition des profils de fichier», à la page 53 afin d'obtenir des exemples de commande RACF.

### Pipeline, sécurité

Quand le message SOAP est reçu par CICS par l'intermédiaire de l'interface Web Service, il est traité par un pipeline. Un pipeline désigne un ensemble de gestionnaires des messages qui sont exécutés dans l'ordre. CICS lit le fichier de configuration du pipeline pour déterminer les gestionnaires de messages à appeler dans le pipeline. Un gestionnaire des messages est un programme qui permet d'exécuter un traitement spécial des demandes et réponses de service Web.

Application Deployment Manager procure un exemple de fichier de configuration de pipeline qui spécifie l'appel d'un gestionnaire des messages et d'un programme de traitement de l'en-tête SOAP.

Le gestionnaire de message de pipeline (ADNTMSGH) est utilisé par sécurité dans le traitement de l'ID utilisateur et des mots de passe dans l'en-tête du protocole SOAP. ADNTMSGH est référencé par le fichier de configuration de pipeline et doit donc être placé dans la concaténation RPL CICS.

### Sécurité de transaction

CPIH correspond à l'ID de transaction par défaut sous lequel une application appelée par un pipeline sera exécutée. En règle générale, CPIH est défini pour un niveau minimal d'autorisation.

Developer for System z met à disposition plusieurs transactions qui sont utilisées par le serveur CRD lors de la définition et de la consultation des ressources CICS. Ces ID de transaction sont définis par le serveur CRD en fonction de l'opération demandée. Reportez-vous à la section "(Facultatif) Gestionnaire de déploiement d'application" du *Guide de configuration de l'hôte* (SC11-6285) pour obtenir plus d'informations sur la personnalisation des ID de transaction.

| Transaction | Description                                                                                                                                                                                                                             |
|-------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ADMS        | Pour les demandes, par l'outil de traitement des manifestes, de modification des ressources CICS. Généralement destiné aux administrateurs CICS. Cette transaction requiert des droits d'accès de haut niveau.                          |
| ADMI        | Pour les demandes qui définissent, installent ou désinstallent les ressources CICS. Cette transaction peut requérir des droits d'accès de niveau intermédiaire en fonction des règles d'administration de votre site.                   |
| ADMR        | Pour toutes les autres demandes qui récupèrent des informations sur les ressources ou l'environnement CICS. Cette transaction peut requérir des droits d'accès de niveau minimum en fonction des règles d'administration de votre site. |

Certaines ou toutes les demandes de définition de ressource effectuées par les transactions du serveur CRD doivent être sécurisées. Au minimum, les commandes de mise à jour (mise à jour des paramètres de service Web par défaut, des paramètres de descripteur par défaut et de liaison de noms de fichiers) doivent être sécurisées pour éviter que les administrateurs CICS émettent ces commandes permettant de définir les paramètres par défaut de la ressource globale.

Quand la transaction est rattachée, la vérification de la sécurité de la ressource CICS, si elle est activée, garantit que l'ID utilisateur est autorisé à exécuter l'ID de transaction.

La vérification de la ressource est contrôlée par l'option RESSEC dans la transaction qui est en cours de fonctionnement, c'est-à-dire le paramètre d'initialisation système RESSEC, et pour le serveur CRD, le paramètre d'initialisation système XPCT.

La vérification de la ressource a lieu uniquement si la valeur du paramètre d'initialisation système XPCT est différente de NO et que l'option RESSEC de la définition TRANSACTION est définie sur YES ou que le paramètre d'initialisation système RESSEC est défini sur ALWAYS.

Les commandes RACF suivantes fournissent un exemple de procédé de protection des transactions du serveur CRD. Pour plus d'informations relatives à la définition de la sécurité CICS, voir *RACF Security Guide for CICSTS*.

- 
- RALTER GCICSTRN SYSADM UACC(NONE) ADDMEM(ADMS)
- 
- PERMIT SYSADM CLASS(GCICSTRN) ID(#cicsadmin)
- 
- RALTER GCICSTRN DEVELOPER UACC(NONE) ADDMEM(ADMI)
- 
- PERMIT DEVELOPER CLASS(GCICSTRN) ID(#cicsdeveloper)
- 
- RALTER GCICSTRN ALLUSER UACC(READ) ADDMEM(ADMR)
-

## Communication chiffrée via SSL

Le chiffrement SSL du flux de données est pris en charge lorsque le client Application Deployment Manager utilise l'interface Web Services pour appeler le serveur CRD. L'utilisation de SSL pour ces communications est contrôlée par le mot clé SSL(YES) dans la définition CICSTS TCPIPSERVICE (voir le document *RACF Security Guide for CICSTS*).

## Protection des ressources

CICSTS offre la possibilité de protéger les ressources et les commandes permettant de les manipuler. Certaines actions du gestionnaire de déploiement d'application risquent de ne pas aboutir si la sécurité est active mais pas intégralement configurée (autorisation de manipuler les nouveaux types de ressources, par exemple)

En cas de problème de fonctionnement dans le gestionnaire de déploiement d'application, examinez le journal CICS des messages, tel que celui indiqué ci-dessous, et effectuez l'action corrective indiquée dans le document *RACF Security Guide for CICSTS*.

```
DFHXS1111 %date %time %applid %tranid Security violation by user
%userid at netname %portname for resource %resource in class
%classname. SAF codes are (X'safresp',X'safreas'). ESM codes are
(X'esmpresp',X'esmpreas').
```

---

## Utilitaire d'administration

Developer for System z offre l'utilitaire d'administration qui permet aux administrateurs CICS de fournir des valeurs par défaut pour les définitions de ressource CICS. Ces valeurs par défaut peuvent être accessibles en lecture seulement ou peuvent être éditées par le développeur d'application.

L'utilitaire d'administration offre les fonctions suivantes :

- Il fournit le nom CICSplex pour les environnements de test gérés par CICSplex
- Il fournit le nom du groupe de transfert de CICSplex SM
- Il fournit le paramètre de la règle d'exportation des manifestes
- Il fournit les droits d'affichage et les valeurs par défaut des attributs de ressource CICS
- Il fournit la liaison logique-physique CICS utilisée pour les définitions de fichier VSAM

L'utilitaire d'administration est appelé par un modèle de travail ADNJSPAU dans le fichier FEK.#CUST.JCL. L'utilisation de cet utilitaire requiert les droits d'accès UPDATE au référentiel CRD.

ADNJSPAU se trouve dans FEK.#CUST.JCL, sauf si le programmeur système z/OS a indiqué un autre emplacement lorsqu'il a personnalisé et soumis le travail FEK.SFEKSAMP(FEKSETUP). Reportez-vous à la section "Configuration personnalisée" du *Guide de configuration de l'hôte* (SC11-6285) pour obtenir plus de détails.

**Remarque :** Le référentiel CRD doit être fermé dans CICS avant l'exécution du travail ADNJSPAU. Vous pourrez rouvrir le référentiel une fois le travail exécuté. Par exemple, après vous être connecté à CICS, entrez les commandes suivantes pour fermer et ouvrir le fichier :



- CEMT S FILE(ADNREPF0) CLOSED
- CEMT S FILE(ADNREPF0) OPEN

Les instructions de contrôle d'entrée permettent de mettre à jour le référentiel CRD pour un environnement de test CICS dans lequel les règles de syntaxe suivantes sont d'application :

- Un astérisque en position 1 indique une ligne de commentaire.
- Une commande DEFINE doit débiter en position 1, suivie par un espace unique, puis d'un mot clé valide comme TRANSACTION.
- Une valeur de mot clé doit suivre immédiatement un mot clé. Aucun espace n'est autorisé. La seule exception concerne les mots clés d'autorisation d'affichage UPDATE, PROTECT et HIDDEN qui ne possèdent pas de valeurs.
- Les valeurs des mots clés sont placées entre parenthèses.
- Un mot clé et sa valeur doivent se trouver sur la même ligne.

Les exemples de définition ci-dessous suivent la structure des commandes DFHCSDUP (voir le document *CICS Resource Definition Guide for CICSTS*). La seule différence concerne l'insertion des mots clés d'autorisation d'affichage suivants pour regrouper les valeurs d'attribut en trois ensembles de droits :

|         |                                                                                                                                                                                                         |
|---------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| UPDATE  | Les attributs qui suivent ce mot clé peuvent être mis à jour par un développeur d'applications à l'aide de Developer for System z. Il s'agit également de la valeur par défaut pour les attributs omis. |
| PROTECT | Les attributs qui suivent ce mot clé sont affichés, mais ils sont protégés contre toute mise à jour par un développeur d'applications qui utilise Developer for System z.                               |
| HIDDEN  | Les attributs qui suivent ce mot clé ne sont pas affichés et sont protégés contre toute mise à jour par un développeur d'applications qui utilise Developer for System z.                               |

Voir l'exemple de code ADNJSPAU.



```

//ADNJSPAU JOB <JOB PARAMETERS>
//*
//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 *
*
* CICSplex SM parameters
*
DEFINE CPSMNAME()
*DEFINE STAGINGGROUPNAME(ADMSTAGE)
*
* Manifest export rule
*
DEFINE MANIFESTEXPORTRULE(installOnly)
*
* CICS resource definition defaults
* Omitted attributes default to UPDATE.
*
* DB2TRAN default attributes
*
DEFINE DB2TRAN()
 UPDATE DESCRIPTION()
 ENTRY()
 TRANSID()
*
* DOCTEMPLATE default attributes
*
DEFINE DOCTEMPLATE()
 UPDATE DESCRIPTION()
 TEMPLATENAME()
 FILE() TSQUEUE() TDQUEUE() PROGRAM() EXITPGM()
 DDNAME(DFHHTML) MEMBERNAME()
 HFSFILE()
 APPENDCRLF(YES) TYPE(EBCDIC)
*
* File default attributes
*
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) JNLSYNCWRITE(YES)
 RECOVERY(NONE) FWDRECOVLOG(NO)
 BACKUPTYPE(STATIC)
 PASSWORD() NSRGROUP()
 CFDTPOOL() TABLENAME()

```

Figure 27. Utilitaire d'administration ADNJSAPU - CICSSTS

```

*
* Mapset default attributes
*
DEFINE MAPSET()
 UPDATE DESCRIPTION()
 PROTECT RESIDENT(NO) STATUS(ENABLED)
 USAGE(NORMAL) USELPACOPY(NO)
** Processtype default attributes
*
DEFINE PROCESSTYPE()
 UPDATE DESCRIPTION()
 FILE(BTS)
 PROTECT STATUS(ENABLED)
 AUDITLOG() AUDITLEVEL(OFF)
*
* Program default attributes
*
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)
*
* TDQueue default attributes
*
DEFINE TDQUEUE()
 UPDATE DESCRIPTION()
 TYPE(INTRA)
* Extra partition parameters
 DDNAME() DSNAME()
 REMOTENAME() REMOTESYSTEM() REMOTELength(1)
 RECORDSIZE() BLOCKSIZE(0) RECORDFORMAT(UNDEFINED)
 BLOCKFORMAT() PRINTCONTROL() DISPOSITION(SHR)
* Intra partition parameters
 FACILITYID() TRANSID() TRIGERRLEVEL(1)
 USERID()
* Indirect parameters
 INDIRECTNAME()
 PROTECT WAIT(YES) WAITACTION(REJECT)
* Extra partition parameters
 DATABUFFERS(1)
 SYSOUTCLASS() ERROROPTION(IGNORE)
 OPENTIME(INITIAL) REWIND(LEAVE) TYPEFILE(INPUT)
* Intra partition parameters
 ATIFACILITY(TERMINAL) RECOVSTATUS(NO)

```

*Figure 28. ADNJSAPU - Utilitaire d'administration CICSTS (Partie 2 de 3)*

```

*
* Transaction default attributes
*
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) STORAGECLEAR(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()

*
* URDIMAP attributes
*
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()

*
* Optional file name to VSAM data set name binding
*
*DEFINE DSBINDING() DSNAME()
/*

```

Figure 29. ADNJSAPU - Utilitaire d'administration CICSTS (Partie 3 de 3)

## Notes de migration de l'utilitaire d'administration

Developer for System z version 7.6.1 bénéficie désormais de la prise en charge d'URIMAP pour l'utilitaire d'administration. Pour pouvoir utiliser la prise en

charge d'URIMAP, vous devez allouer au fichier VSAM du référentiel CRD la taille d'enregistrement maximale de 3000. Jusqu'à Developer for System z version 7.6.1, le modèle du travail d'allocation du référentiel CRD utilise une taille d'enregistrement maximale de 2000.

Les étapes suivantes décrivent l'activation de la prise en charge d'URIMAP, si vous utilisez un ancien référentiel CRD :

1. Créez une sauvegarde de votre référentiel CRD existant, FEK.#CUST.ADNREPF0.
2. Supprimez le référentiel CRD existant.
3. Personnalisez et soumettez le travail FEK.SFEKSAMP(ADNVCRD) pour allouer et initialiser un nouveau référentiel CRD. Pour obtenir les instructions de personnalisation, consultez la documentation qui se trouve dans le membre.
4. Personnalisez et soumettez le travail FEK.SFEKSAMP(ADNJSPAU) pour utiliser l'utilitaire d'administration pour remplir le référentiel CRD.

**Remarque :**

- La migration du référentiel existant n'est pas nécessaire car l'utilitaire d'administration remplace l'intégralité du contenu du référentiel CRD chaque fois qu'il est exécuté.
- Il n'existe aucun problème de compatibilité avec le référentiel CRD. Tous les codes client et hôte Developer for System z pris en charge fonctionneront avec l'une ou l'autre des tailles d'enregistrement maximales. Notez toutefois que la prise en charge d'URIMAP sera désactivée si la taille d'enregistrement maximale n'est pas 3000.

## Messages de l'utilitaire d'administration

Les messages ci-après sont générés par l'utilitaire d'administration dans SYSPRINT DD. Les messages CRAZ1803E, CRAZ1891E, CRAZ1892E, et CRAZ1893E contiennent les codes de statut de fichier ainsi que les codes de retour, de fonction et de commentaire VSAM. Les codes de retour, de fonction et de commentaire de VSAM sont expliqués dans le document *DFSMS Macro Instructions for Data Sets* (SC26-7408). Les codes de statut de fichier sont expliqués dans le document *Enterprise COBOL for z/OS Language Reference* (SC27-1408).

**CRAZ1800I**

**Exécution achevée avec succès à la ligne <dernier numéro de ligne d'instruction de contrôle>**

**Explication :** L'utilitaire d'administration du programmeur système a été correctement exécuté.

**Réponse de l'utilisateur :** Aucune.

**CRAZ1801W**

**Exécution achevée avec des avertissements à la ligne <dernier numéro de ligne d'instruction de contrôle>**

**Explication :** L'utilitaire d'administration du programmeur système a fini de traiter un ou plusieurs avertissements détectés lors du traitement des instructions de contrôle.

**Réponse de l'utilisateur :** Vérifiez les autres messages d'avertissement.

**CRAZ1802E**

**Apparition d'une erreur à la ligne <numéro de ligne>**

**Explication :** L'utilitaire d'administration du programmeur système a rencontré une erreur grave.

**Réponse de l'utilisateur :** Vérifiez les autres messages d'avertissement.

**CRAZ1803E**

**Erreur d'ouverture de référentiel, status=<code d'état de fichier>  
RC=<Code retour VSAM> FC=<Code de fonction VSAM> FB=<Code de  
commentaire VSAM>**

**Explication :** L'utilitaire d'administration du programmeur système a rencontré une erreur grave lors de l'ouverture du référentiel CRD.

**Réponse de l'utilitaire :** Vérifiez les codes de statut, de retour, de fonction et de commentaire VSAM.

**CRAZ1804E**

**Enregistrement d'entrée non reconnu à la ligne <numéro de ligne>**

**Explication :** L'utilitaire d'administration du programmeur système a rencontré une instruction de contrôle d'entrée non reconnue.

**Réponse de l'utilisateur :** Vérifiez qu'une commande **DEFINE** est bien suivie d'un seul espace, puis du mot clé CPSMNAME, STAGINGGROUPNAME, MANIFESTEXPORTRULE, DSBINDING, DB2TRAN, DOCTEMPLATE, FILE, MAPSET, PROCESSTYPE, PROGRAM, TDQUEUE ou TRANSACTION.

**CRAZ1805E**

**Traitement du mot clé mot clé <mot clé> sur la ligne <numéro de ligne>**

**Explication :** L'utilitaire d'administration du programmeur système traite l'instruction de contrôle d'entrée du mot clé DEFINE.

**Réponse de l'utilisateur :** Aucune.

**CRAZ1806E**

**Règle d'exportation de manifeste non valide à la ligne <numéro de ligne>**

**Explication :** L'utilitaire d'administration du programmeur système a rencontré une règle d'exportation de manifeste non valide.

**Réponse de l'utilisateur :** Vérifiez que la valeur du mot clé MANIFESTEXPORTRULE est "installOnly", "exportOnly" ou "both".

**CRAZ1807E**

**Mot clé DSNNAME manquant à la ligne <numéro de ligne>**

**Explication :** L'utilitaire d'administration du programmeur système traitait une instruction de contrôle DEFINE DSBINDING dans laquelle il manque le mot clé DSNNAME.

**Réponse de l'utilisateur :** Vérifiez que l'instruction de contrôle DEFINE DSBINDING contient le mot clé DSNNAME.

**CRAZ1808E**

**Valeur de mot clé non valide pour le mot clé <mot clé> à la ligne  
<numéro de ligne>**

**Explication :** L'utilitaire d'administration du programmeur système traitait une instruction de contrôle DEFINE lorsqu'il a rencontré une valeur non valide pour le mot clé spécifié.

**Réponse de l'utilisateur :** Vérifiez que la longueur et la valeur du mot clé spécifié sont correctes.

**CRAZ1890W**

**Erreur de syntaxe de mot clé à la ligne <numéro de ligne>**

**Explication** : L'utilitaire d'administration du programmeur système traitait une instruction de contrôle DEFINE lorsqu'il a rencontré une erreur de syntaxe pour un mot clé ou sa valeur.

**Réponse de l'utilisateur** : Vérifiez que la valeur du mot clé est placée entre parenthèses et qu'elle est immédiatement suivie du mot clé. Le mot clé et sa valeur doivent se trouver sur la même ligne.

#### **CRAZ1891W**

**Erreur d'écriture de clé en double de référentiel, status=<Code d'état de fichier> RC=<Code retour VSAM> FC=<Code de fonction VSAM> FB=<Code de commentaire VSAM>**

**Explication** : L'utilitaire d'administration du programmeur système a rencontré une erreur de clé en double lors de l'enregistrement dans le référentiel CRD.

**Réponse de l'utilitaire** : Vérifiez les codes de statut, de retour, de fonction et de commentaire VSAM.

#### **CRAZ1892W**

**Erreur d'écriture dans le référentiel, status=<Code d'état de fichier> RC=<Code retour VSAM> FC=<Code de fonction VSAM> FB=<Code de commentaire VSAM>**

**Explication** : L'utilitaire d'administration du programmeur système a rencontré une grave erreur lors de l'écriture dans le référentiel CRD.

**Réponse de l'utilitaire** : Vérifiez les codes de statut, de retour, de fonction et de commentaire VSAM.

#### **CRAZ1893W**

**Erreur de lecture du référentiel, status=<Code d'état de fichier> RC=<Code retour VSAM> FC=<Code de fonction VSAM> FB=<Code de commentaire VSAM>**

**Explication** : L'utilitaire d'administration du programmeur système a rencontré une erreur grave lors de lecture dans le référentiel CRD.

**Réponse de l'utilitaire** : Vérifiez les codes de statut, de retour, de fonction et de commentaire VSAM.

---

## **Débogage de transaction CICS**

Pour déboguer des transactions CICS, le débogueur intégré a besoin des mises à jour de CICS suivantes :

- Mise à jour du langage JCL CICS :
  - Définissez la bibliothèque de chargement FEK,SFEKAUTH dans l'instruction de définition de données de la région DFHRPL si la bibliothèque n'est pas LINKLIST.
  - Définissez la bibliothèque de chargement SYS1.SIEAMIGE dans l'instruction de définition de données de la région STEPLIB si la bibliothèque n'est pas LINKLIST.
- Mises à jour de CDS CICS :

Définissez le débogueur sur une région CICS, comme indiqué dans l'exemple de travail de mise à jour de CDS AQECSD. AQECSD se trouve dans FEK.#CUST.JCL, sauf si le programmeur système z/OS a indiqué un autre emplacement lorsqu'il a personnalisé et soumis le travail FEK.SFEKSAMP(FEKSETUP). Pour plus de détails, reportez-vous à "Configuration personnalisée" dans le *guide de configuration de l'hôte* (SC23-7658).

| Pour déboguer des transactions CICS chargées dans la mémoire morte, le  
| débogueur intégré a besoin des mises à jour système suivantes :

- | • Appel de superviseur (SVC) du débogueur intégré défini sur votre système.  
| Pour plus de détails, reportez-vous à "Modifications de PARMLIB" dans le *guide*  
| *de configuration de l'hôte* (SC23-7658).
- | • L'appel du superviseur nécessite que des utilisateurs aient des droits d'accès sur  
| profil de sécurité s'il est utilisé dans un environnement d'état de problème (non  
| autorisé). Pour plus d'informations, voir «Sécurité du débogage», à la page 38.

| Notez qu'un seul débogueur basé sur Language Environment (LE) peut être actif  
| dans une région CICS donnée. Vous reconnaîtrez facilement un débogueur basé sur  
| LE car celui-ci fournit un module de chargement ou un alias CEEVDBG qui doit être  
| disponible pour l'application.





---

## Chapitre 9. Remarques relatives aux exits utilisateur

Ce chapitre vous guide lors du processus d'amélioration de Developer for System z via la création de routines d'exit.

Developer for System z fournit des points d'exit pour la sélection d'événements Developer for System z. Un point d'exit est un point spécifique dans le traitement d'une fonction où la fonction appelle une routine d'appel s'il en existe une. Vous pouvez créer une routine d'exit pour effectuer un traitement supplémentaire.

Contrairement aux points d'exit standard, les points d'exit Developer for System z ne vous permettent pas de changer le comportement de la fonction. La routine d'exit, s'il en existe une, est appelée de manière asynchrone une fois la fonction terminée. Le traitement Developer for System z n'attend pas la fin de la routine. Il ne vérifie pas non plus l'état d'achèvement.

---

### Caractéristiques de l'exit utilisateur

#### Activation de l'exit utilisateur

Les exits utilisateur sont activés avec les variables `_RSE_JAVAOPTS` `<point_exit>.action` dans `rsed.envvars`, où `<point_exit>` représente un mot clé identifiant un point d'exit spécifique, comme cela est décrit dans «Points d'exit disponibles», à la page 160.

```
#_RSE_JAVAOPTS="$_RSE_JAVAOPTS -D<point_exit>.action=<exit_utilisateur>"
```

Par défaut, tous les points d'exit sont désactivés. Annulez la mise en commentaire et indiquez le chemin d'accès complet de la routine d'exit utilisateur afin d'activer le point d'exit.

```
#_RSE_JAVAOPTS="$_RSE_JAVAOPTS -D<point_exit>.action.id=<idutilisateur>"
```

Par défaut, l'ID utilisateur attribué au démon RSE permet d'exécuter la routine utilisateur fournie. Supprimez la mise en commentaire et spécifiez un ID utilisateur afin d'utiliser l'ID spécifié pour l'exécution de l'ID utilisateur. Il n'est pas nécessaire d'indiquer de mot de passe car RSE génère un PassTicket à utiliser comme mot de passe lorsqu'il passe à l'ID utilisateur indiqué.

#### Création d'une routine d'exit utilisateur

Les routines d'exit utilisateur sont appelées en tant que commande shell z/OS UNIX à laquelle il est possible d'ajouter un ou plusieurs arguments. La routine d'exit que vous développez doit donc être exécutable à partir de la ligne de commande z/OS UNIX. Les techniques communes de codage incluent le script shell z/OS UNIX et la commande `exec REXX` z/OS UNIX mais il est également possible d'utiliser un code compilé, tel C/C++.

Voir *UNIX System Services User's Guide* (SA22-7801) pour plus d'informations sur les scripts de shell z/OS UNIX. Voir le document *Using REXX and z/OS UNIX System Services* (SA22-7806) pour en savoir plus sur les extensions du langage REXX spécifiques à z/OS.

La routine d'exit peut être exécutée par un ID utilisateur disposant de droits spéciaux (ID utilisateur de la tâche démarré RSE, qui dispose de droits lui permettant de générer des PassTicket). Il est donc important de limiter les droits de mise à jour à la routine d'exit afin d'éviter des comportements non souhaités. Les commandes z/OS UNIX exemple suivantes limitent les droits en écriture au propriétaire alors que tous les utilisateurs peuvent lire et exécuter le 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
```

Les définitions dans `rsed.envvars` sont disponibles dans la routine d'exit utilisateur en tant que variables d'environnement.

RSE appelle la routine d'exit utilisateur avec une seule chaîne d'arguments. La chaîne d'arguments peut être une seule valeur ou une seule chaîne qui contient plusieurs mots clés et valeurs délimités par des espaces. Pour plus d'informations, voir «Points d'exit disponibles», à la page 160.

## Messages de console

Developer for System z utilise l'ID de message de console FEK910I pour afficher les données relatives aux exits utilisateur.

L'appel de la routine d'exit est indiqué par le message de console suivant :

```
FEK910I <POINT_EXIT> EXIT: invoking <point_exit> processing exit
 in thread <id_unité_exécution>
```

Toutes les données placées dans `stdout` (commande **echo** dans un script shell, commande **say** dans une exécution REXX) sont envoyées à la console :

```
FEK910I <POINT_EXIT> EXIT: <message>
```

La fin de la routine d'exit est indiquée par le message de console suivant :

```
FEK910I <POINT_EXIT> EXIT: completed <point_exit> processing exit
 in thread <id_unité_exécution>
```

## Exécution avec un ID utilisateur variable

Developer for System z permet d'exécuter une routine d'utilisateur avec l'ID utilisateur de la tâche démarrée ou avec un ID utilisateur indiqué. Toutefois, vous pouvez souhaiter exécuter certaines actions dans la routine utilisateur en utilisant un autre ID utilisateur, tel l'ID utilisateur client se trouvant dans la routine d'exit de connexion. Pour cela, utilisez les services z/OS UNIX standard, comme cela est présenté dans les exemples suivants.

### Script de shell z/OS UNIX

Comme cela est décrit dans le document *UNIX System Services Command Reference* (SA22-7802), z/OS UNIX inclut la commande **su** qui permet d'utiliser les droits d'un superutilisateur ou d'un autre utilisateur. Plusieurs éléments sont à prendre en compte lors de l'utilisation de la commande **su**.

- L'ID utilisateur exécutant la commande **su** doit disposer de droits en lecture dans le profil `BPX.SRV.<idutilisateur>` se trouvant dans la classe SURROGAT de votre produit de sécurité afin de pouvoir employer l'ID utilisateur identifié par `<idutilisateur>` sans indiquer de mot de passe.
- La commande **su** lance un nouveau shell, afin que les commandes restantes dans votre script de shell ne soient pas exécutées avant la fin du shell lancé par la commande **su**. Afin que les commandes soient exécutées dans le nouveau shell démarré par la commande **su**, vous pouvez utiliser la commande **echo** pour

créer la commande souhaitée et insérer le caractère de barre verticale dans le nouveau shell, comme cela est présenté dans l'exemple suivant. Notez que les règles de scriptage de shell standard s'appliquent aux caractères d'échappement spéciaux.

```
#!/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
```

Cet exit de connexion exemple, exécuté par l'ID utilisateur de la tâche démarrée, génère les messages de console suivants :

```
+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
```

## Commande exec REXX z/OS UNIX

Comme cela est décrit dans *Using REXX and z/OS UNIX System Services* (SA22-7806), z/OS UNIX inclut la commande **seteu**id SYSCALL qui permet de définir l'UID du processus en cours. Plusieurs éléments sont à prendre en compte lors de l'utilisation de la commande **seteu**id.

- La commande **seteu**id utilise l'ID utilisateur z/OS UNIX et non l'ID utilisateur MVS. Vous devez tout d'abord déterminer l'UID de l'ID utilisateur cible, opération qui peut être effectuée avec la commande **getpwnam** SYSCALL.
- L'ID utilisateur exécutant la commande **seteu**id doit avoir des droits en écriture pour le profil BPX.SRV.<idutilisateur> dans la classe SURROGAT de votre produit de sécurité afin qu'il soit possible d'employer l'ID utilisateur identifié par <idutilisateur> sans spécifier de mot de passe. Lorsque plusieurs ID utilisateur ont le même UID, il n'est pas possible déterminer quel ID utilisateur sera employé.

```
/* rexx */
myID='ibmuser'
say userid()
address SYSCALL 'getpwnam' myID 'pw.'
say pw.1 pw.2 pw.3 pw.4 pw.5
address SYSCALL 'seteu' pw.2 /* PW_UID = 2 */
say retval errno
say userid()
```

Cet exit de connexion exemple, exécuté par l'ID utilisateur de la tâche démarré, génère les messages de console suivants :

```
+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
```

---

## Points d'exit disponibles

Les points d'exit suivants sont fournis par Developer for System z :

- «audit.action»
- «logon.action»

### audit.action

- **Moment de l'appel :**

L'exit utilisateur d'audit est appelé à la fermeture du fichier journal d'audit actif. (L'audit se poursuit car RSE est passé à un nouveau fichier d'audit.)

- **Arguments d'appel (1) :**

– <journal\_audit> : chemin complet du fichier journal d'audit fermé

- **Exemple :**

/usr/lpp/rdz/samples/process\_audit.rex

Cette commande exec REXX z/OS UNIX génère un travail par lots qui traitera le journal d'audit fermé.

### logon.action

- **Moment de l'appel :**

L'exit utilisateur de connexion est appelé lorsqu'un utilisateur a terminé le processus de connexion.

- **Arguments d'appel (6) :**

– -i <idutilisateur> : ID utilisateur client, la casse est telle qu'elle est fournie par le client

– -u <chemin\_journal\_utilisateur> : répertoire dans lequel les journaux utilisateur de ce client sont conservés

– -s <chemin\_journal\_serveur> : répertoire dans lequel les journaux du serveur sont conservés

– -c <chemin\_config> : répertoire dans lequel les fichiers de configuration sont conservés

– -b <chemin\_éléments\_binaires> : répertoire où Developer for System z est installé

– -p <port> : port de démon RSE

- **Exemple :**

/usr/lpp/rdz/samples/process\_logon.sh

Ce script de shell z/OS UNIX exemple crée un message de connexion dans la console.

---

## Chapitre 10. Personnalisation de l'environnement TSO

Ce chapitre vous aide à simuler une procédure d'ouverture de session TSO en ajoutant des instructions de définition de données et des fichiers à l'environnement TSO dans Developer for System z.

---

### Service Commandes TSO

Le service Commandes TSO est le composant Developer for System z qui exécute les commandes TSO et ISPF (par lots) et renvoie le résultat au client demandeur. Ces commandes peuvent être demandées implicitement par le produit, ou explicitement par l'utilisateur.

Les exemples de membres fournis avec Developer for System z créent un environnement TSO/ISPF minimal. Si les développeurs doivent accéder à des bibliothèques personnalisées ou tierces, le programmeur système z/OS doit ajouter les instructions de définition de données et les bibliothèques nécessaires à l'environnement du service Commandes TSO. Bien que l'implémentation soit différente dans Developer for System z, la logique sous-jacente est identique à la procédure d'ouverture de session TSO.

**Remarque :** Le service Commandes TSO est un outil de ligne de commande non interactif, par conséquent, les commandes ou les procédures d'invite de saisie de données ou d'affichage de panneaux ISPF ne fonctionnent pas. Un émulateur 3270, comme un émulateur de connexion à l'hôte, qui fait partie du client Developer for System z, est nécessaire à leur exécution.

### Méthodes d'accès

Depuis la version 7.1, Developer for System z permet de choisir le mode d'accès au service Commandes TSO.

- Service de passerelle client TSO/ISPF d'ISPF, qui nécessite un niveau de service ISPF minimal. Il s'agit de la méthode par défaut utilisée dans les exemples fournis.
- Transaction APPC (comme dans les éditions précédant la version 7.1). La méthode est obsolète.

**Remarque :**

- Le service de passerelle client TSO/ISPF d'ISPF remplace la fonction SCLM Developer Toolkit utilisée dans la version 7.1.
- L'utilisation d'APPC par Developer for System z est indiquée comme étant obsolète. Les informations associées à APPC ont été supprimées de cette publication. Pour plus d'informations, reportez-vous au livre blanc *Using APPC to provide TSO command services* (SC14-7291), disponible dans la bibliothèque Developer for System z à l'adresse <http://www-01.ibm.com/support/docview.wss?uid=swg27038517>.

Consultez le fichier `rsed.envvars` pour déterminer la méthode d'accès utilisée pour les hôtes de la version 7.1 ou d'une version supérieure. Si les valeurs par défaut ont été utilisées pendant la procédure de configuration, `rsed.envvars` se trouve dans `/etc/rdz/`.

- Si l'instruction `_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DTSO_SERVER=APPC"` est absente ou si elle est mise en commentaire, le service de passerelle client TSO/ISPF d'ISPF est utilisé.
- Si l'instruction `_RSE_JAVAOPTS="$_RSE_JAVAOPTS -DTSO_SERVER=APPC"` est présente (et non mise en commentaire), APPC est utilisé.

---

## Utilisation de la méthode d'accès par passerelle client TSO/ISPF

### ISPF.conf

Le fichier de configuration ISPF.conf (situé par défaut dans le répertoire /etc/rdz/) définit l'environnement TSO/ISPF utilisé par Developer for System z. Il existe un seul fichier de configuration ISPF.conf actif, lequel est utilisé par tous les utilisateurs Developer for System z.

La section principale du fichier de configuration définit les noms de définition de données et les concaténations de fichiers associées, comme ceux de l'exemple suivant :

```
sysproc=ISP.SISPLIB,FEK.SFEKPROC
ispmlib=ISP.SISPMENU
isptlib=ISP.SISPTEU
ispplib=ISP.SISPPENU
ispplib=ISP.SISPLIB
ispllib=ISP.SISLOAD
myDD=HLQ1.LLQ1,HLQ2.LLQ2
```

- Chaque définition de données utilise exactement une ligne (il n'y a pas de prise en charge de plusieurs lignes) et il n'y a pas de limites de longueur.
- Il n'y a pas de distinction maj/min dans les définitions et tous les blancs sont ignorés.
- Les lignes mises en commentaire commencent par un astérisque (\*).
- Les noms de définition de données sont suivies d'un signe égale (=), lui-même suivi de la concaténation de fichiers. Les noms de fichier sont séparés par une virgule (,).
- La recherche des concaténations de fichiers s'effectue dans l'ordre où elles sont affichées.
- Les fichiers doivent avoir un nom qualifié complet, sans guillemets (') et sans variables.
- Tous les fichiers sont alloués avec DISP=SHR.
- De nouveaux noms de définition de données peuvent être ajoutés à volonté, mais ils doivent obéir aux règles applicables aux noms de définitions (JCL) et ne pas entrer en conflit avec les autres paramètres de configuration du fichier ISPF.conf. De plus, ISPPROF est alloué dynamiquement DISP=NEW,DELETE) par le service de passerelle client TSO/ISPF.

### Utilisation des profils ISPF existants

Par défaut, la passerelle client TSO/ISPF crée un profil ISPF temporaire pour le service Commandes TSO. Cependant, vous pouvez demander à la passerelle client TSO/ISPF z d'utiliser une copie d'un profil ISPF existant. La clé ici est l'instruction `_RSE_ISPF_OPTS` du fichier `rzed.envvars`.

```
#_RSE_ISPF_OPTS="$_RSE_ISPF_OPTS&ISPPROF=&SYSUID..ISPPROF"
```

Supprimez la mise en commentaire de l'instruction (en retirant le signe dièse (#) initial) et fournissez le nom qualifié complet de fichier de données du profil ISPF existant pour utiliser cette fonction.

Les variables suivantes peuvent être utilisées dans le nom du fichier :

- &SYSUID. en remplacement de l'ID utilisateur du développeur
- &SYSPREF. en remplacement du préfixe TSO du développeur
- &SYSNAME. en remplacement du nom de système, tel qu'il est spécifié dans le membre parmlib IEASYMxx

**Remarque :**

- Si le nom de fichier transmis dans "ISPPROF" n'est pas valide, un profil ISPF vide temporaire est utilisé à la place.
- Le profil ISPF (temporaire et copié) est supprimé à la fin de la session. Les modifications apportées au profil ne sont pas fusionnées dans le profil ISPF existant.

## Utilisation d'une commande exec d'allocation

L'instruction `allocjob` du fichier `ISPF.conf` (mise en commentaire par défaut) pointe sur une commande `exec` qui peut être utilisée pour fournir des allocations de fichiers supplémentaires par ID utilisateur.

```
*allocjob = ISP.SISPSAMP(ISPZISP2)
```

Supprimez la mise en commentaire de l'instruction (en retirant l'astérisque (\*) initial) et fournissez la référence complète à la commande `exec` d'allocation pour utiliser cette fonction.

- La commande `exec` est exécutée après l'allocation d'ISPPROF et des définitions de données définies dans `ISPF.conf`, mais avant l'initialisation d'ISPF. Vérifiez que la commande `exec` d'allocation n'annule pas ces définitions.
- Un paramètre, l'ID utilisateur du demandeur, est transmis à la commande `exec`.
- Un exemple d'instruction `exec CRAISPRX` est fourni dans l'exemple de bibliothèque `FEK.#CUST.CNTL`, sauf si vous avez indiqué un autre emplacement lorsque vous avez personnalisé et soumis le travail `FEK.SFEKSAMP(FEKSETUP)`. Reportez-vous à la section "Configuration personnalisée" du *Guide de configuration de l'hôte* (SC11-6285) pour obtenir plus de détails.

**Remarque :** Du fait que la commande `exec` est appelée avant l'initialisation d'ISPF, vous ne pouvez pas utiliser **VPUT** et **VGET**. Vous pouvez cependant créer votre propre implémentation de ces fonctions à l'aide d'un fichier PDS(E) ou VSAM.

## Utilisation de plusieurs commandes exec d'allocation

Bien que `ISPF.conf` prenne uniquement en charge l'appel d'une seule commande `exec`, cette dernière peut en revanche appeler une autre commande `exec` sans limite. L'ID utilisateur du client transmis comme paramètre donne la possibilité d'appeler des commandes `exec` d'allocation personnalisées. Vous pouvez, par exemple, vérifier si le membre `USERID'.EXEC(ALLOC)'` existe et l'exécuter.

Un certain nombre de variations de ce scénario permettent d'utiliser les procédures d'ouverture de session TSO :

- Lisez le fichier de configuration propre à l'utilisateur (`USERID'.FEKPROF'`, par exemple).
- Identification de la procédure d'ouverture de session mentionnée dans le fichier.
- Lecture et analyse de la procédure mentionnée à partir de `SYS1.PROCLIB` en vue d'en extraire les instructions de définition de données et les allocations de fichiers.



- Allocation du fichier selon un mode similaire à la procédure d'ouverture de session réelle.

## Utilisation de fichiers ISPF.conf multiples avec configurations Developer for System z multiples

Si les scénarios exec d'allocation décrits dans les sections précédentes ne répondent pas à vos besoins spécifiques, vous pouvez créer des instances de serveur de communication RSE Developer for System z différentes, chacune d'elle utilisant son propre fichier ISPF.conf. L'inconvénient majeur de la méthode décrite ci-dessous est que les utilisateurs de Developer for System z doivent se connecter à différents serveurs sur le même système hôte pour obtenir l'environnement TSO voulu.

**Remarque :** La création d'une deuxième instance sur le serveur RSE nécessite la duplication et la mise à jour des fichiers de configuration ainsi que des définitions de JCL de démarrage et de tâche démarrées. Aucune nouvelle installation du produit n'est nécessaire, ni aucune duplication de code.

```
$ 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
-> change: _RSE_RSED_PORT=4037
-> change: CGI_ISPCONF=/etc/rdz/tso2
-> change: -Ddaemon.log=/var/rdz/logs/tso2
-> change: -Duser.log=/var/rdz/logs/tso2
-> add at the END:
-- NEEDED TO FIND THE REMAINING CONFIGURATION FILES
CFG_BASE=/etc/rdz
CLASSPATH=.:$CFG_BASE:$CLASSPATH
--
$ oedit /etc/rdz/tso2/ISPF.conf
-> change: change as needed
```

Les commandes de l'exemple ci-dessus copient les fichiers de configuration Developer for System z à modifier dans le répertoire qui vient d'être créé, tso2. La variable CGI\_ISPCONF dans rsed.envvars doit être mise à jour pour définir le nouveau répertoire de base ISPF.conf. De plus, daemon.log et user.log doivent être mis à jour pour définir un nouvel emplacement de journal (lequel est créé automatiquement s'il n'existe pas). La mise à jour de \_RSE\_RSED\_PORT permet de garantir que le démon RSE existant et le nouveau démon RSE utiliseront des numéros de port uniques. La mise à jour de la variable CLASSPATH permet de s'assurer que RSE peut localiser les fichiers de configuration qui n'ont pas été copiés dans tso2. Le fichier ISPF.conf peut lui-même être mis à jour pour que vous puissiez l'adapter à vos besoins. Notez que la zone de travail ISPF (variable CGI\_ISPWORK dans rsed.envvars) peut être partagée entre les deux instances.

Les éléments restants créent une nouvelle tâche démarrée pour RSE qui utilise un nouveau numéro de port et les nouveaux fichiers de configuration /etc/rdz/tso2. Notez que si la variable \_RSE\_RSED\_PORT n'est pas modifiée dans rsed.envvars, la nouvelle tâche démarrée doit spécifier un nouveau port comme argument de démarrage.

Pour plus d'informations sur les actions présentées précédemment dans cette section, voir *IBM Rational Developer for System z - Guide de configuration de l'hôte* (SC23-7658).



---

## Chapitre 11. Exécution de plusieurs instances

Parfois, vous pouvez avoir besoin de plusieurs instances de Developer for System z actives sur un même système, lors du test d'une mise à niveau, par exemple. Cependant, certaines ressources (les ports TCP/IP, par exemple) ne peuvent pas être partagées. Les paramètres par défaut ne sont donc pas toujours applicables. Consultez les informations de cette annexe afin de programmer la coexistence des différentes instances de Developer for System z, pour pouvoir ensuite les personnaliser à l'aide de ce guide de configuration.

Bien qu'il soit possible de partager certaines parties de Developer for System z entre deux instances (ou plus), il est recommandé de NE PAS le faire, sauf si leurs niveaux de logiciel sont identiques et que les seules modifications sont effectuées dans les membres de configuration. Developer for System z offre suffisamment de possibilités de personnalisation pour que plusieurs instances ne se chevauchent pas et nous vous recommandons vivement d'utiliser ces fonctions.

### Remarque :

- FEK et /usr/lpp/rdz sont le qualificatif de haut niveau et le chemin utilisé lors de l'installation du produit. FEK.#CUST, /etc/rdz et /var/rdz sont les emplacements par défaut utilisés lors de la personnalisation du produit (voir "Configuration personnalisée" du *Guide de configuration de l'hôte* (SC11-6285) pour plus d'informations)..
- Il est recommandé d'installer Developer for System z dans un système de fichiers privé (HFS ou zFS) pour faciliter le déploiement des composants z/OS UNIX du produit.
- Si vous ne pouvez pas utiliser un système de fichiers privé, il est recommandé d'utiliser un outil d'archivage (la commande z/OS UNIX tar, par exemple) pour transférer les répertoires z/OS UNIX d'un système à un autre. Il s'agit de préserver les attributs (tels que le contrôle par programme) des fichiers et répertoires Developer for System z.

Pour plus d'informations sur les exemples de commande ci-dessous qui permettent d'archiver et de restaurer le répertoire d'installation de Developer for System z, voir le document *UNIX System Services Command Reference* (SA22-7802)

- Archivage : `cd /SYS1/usr/lpp/rdz; tar -cSf /u/userid/rdz.tar`
- Restauration : `cd /SYS2/usr/lpp/rdz; tar -xSf /u/userid/rdz.tar`

---

## Configuration identique par sysplex

Les fichiers de configuration (et le code) Developer for System z peuvent être partagés entre différents systèmes dans un sysplex, chaque système exécutant sa propre copie identique de Developer for System z, si de nouvelles instructions sont respectées. Notez que ces informations sont valables pour les instances Developer for System z autonomes. Des règles supplémentaires pour la configuration de TCP/IP s'appliquent lors de l'utilisation de l'adressage DVIPA distribué afin de regrouper plusieurs serveurs (chacun situé dans un système distinct) en un seul serveur virtuel, comme indiqué dans «Distributed Dynamic VIPA», à la page 64.

- Il est recommandé de placer les fichiers journaux dans des emplacements uniques, afin d'éviter qu'un système n'écrase les données d'un autre. En routant les journaux z/OS UNIX vers des emplacements spécifiques avec les directives `daemon.log` et `user.log` dans `rsed.envvars`, vous pouvez partager les fichiers de

configuration si vous montez un système de fichiers d'un système spécifique z/OS UNIX dans le chemin spécifié. De cette manière, tous les fichiers journaux sont écrits dans le même emplacement logique, mais compte tenu du système de fichiers non partagé du dessous, ils sont placés dans des emplacements physiques différents.

- Les répertoires de type configuration, tels que `/etc/rdz/` et `/var/rdz/pushtoclient/`, peuvent être partagés dans le sysplex, car Developer for System z les utilise en lecture seule.
- Les répertoires de données temporaires (`/tmp/` et `/var/rdz/WORKAREA/`, par exemple) doivent être uniques par système, étant donné que les noms de fichier temporaire ne sont pas compatible avec sysplex.
- Si vous partagez le code, il est également recommandé de partager les fichiers de configuration pour vous assurez que tous les systèmes sont synchronisés après les opérations de maintenance.
- Si vous partagez un fichier de configuration `/etc/rdz/pushtoclient.properties` actif, vous devez également partager le répertoire de métadonnées associé, `/var/rdz/pushtoclient/`.

---

## Niveaux de logiciels identiques, fichiers de configuration différents

Dans un nombre limité de circonstances, vous pouvez partager tout sauf (certains) des composants personnalisables. La mise à disposition d'un accès non-SSL pour un utilisation sur site, et d'une communication encodée SSL pour une utilisation hors site est un exemple.

**Avvertissement :** La configuration partagée NE peut PAS être utilisée de manière sûre pour tester la maintenance, ou effectuer une prévisualisation technique ou une nouvelle édition.

Pour configurer une autre instance d'une installation active de Developer for System z, suivez de nouveau la procédure de personnalisation pour les composants qui sont différents, en utilisant d'autres fichiers, répertoires et ports afin d'éviter un chevauchement avec la configuration en cours.

Dans l'exemple SSL mentionné précédemment, la configuration du démon RSE en cours peut être clonée, après quoi la configuration clonée peut être mise à jour. Le JCL de démarrage du démon RSE peut ensuite être cloné et personnalisé avec un nouveau port TCP/IP et l'emplacement des fichiers de configuration mis à jour. Les personnalisations du système MVS (moniteur de travaux JES etc.) peuvent être partagées entre les instances SSL et les instances non-SSL. Il en résulte les actions suivantes :

```
$ 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
-> change: _RSE_RSED_PORT=4047
-> change: -Ddaemon.log=/var/rdz/logs/ssl
-> change: -Duser.log=/var/rdz/logs/ssl
-> add at the END:
-- NEEDED TO FIND THE REMAINING CONFIGURATION FILES
CFG_BASE=/etc/rdz
```

```

CLASSPATH=.:$CFG_BASE:$CLASSPATH
--
$ oedit /etc/rdz/ssl/ssl.properties
-> change: change as needed

```

Les commandes de l'exemple précédent copient les fichiers de configuration Developer for System z à modifier dans le répertoire qui vient d'être créé, `ssl`. Les variables `daemon.log` et `user.log` dans `rsed.envvars` doivent être mises à jour pour définir un nouvel emplacement de journal (qui est créé automatiquement, s'il n'existe pas). La mise à jour de la variable `CLASSPATH` permet de s'assurer que RSE peut localiser les fichiers de configuration qui n'ont pas été copiés dans `ssl`. Le fichier `ssl.properties` peut lui-même être mis à jour en fonction de vos besoins.

Les éléments restants créent une nouvelle tâche démarrée pour RSE qui utilise un nouveau numéro de port et les nouveaux fichiers de configuration `/etc/rdz/ssl`.

Pour plus d'informations sur les actions présentées précédemment dans cette section, voir les sections connexes du document *IBM Rational Developer for System z - Guide de configuration de l'hôte* (SC23-7658).

## Synchronisation automatisée

Dans l'exemple SSL mentionné précédemment, les modifications entre le démon RSE non-SSL et le démon RSE compatible SSL sont minimales, ce qui permet d'automatiser le processus de maintien de la synchronisation de leurs fichiers `rsed.envvars`. Cela simplifie le déploiement de service car un seul fichier `rsed.envvars` doit être géré.

L'exemple suivant ajoute le numéro de port `RSED` aux noms de répertoire de journaux et met à jour le `CLASSPATH` de sorte que les clones trouvent les fichiers de configuration restants. L'exemple étend ensuite le langage JCL de la tâche démarrée du démon RSE compatible SSL pour cloner le fichier `rsed.envvars` du démon RSE non-SSL au démarrage, en mettant à jour le numéro de port pendant le processus. Le numéro de port étant intégré dans le nom de répertoire de journaux, il est automatiquement différent pour chacun des deux démons.

1. Préparez le fichier `rsed.envvars` maître.

```

$ oedit /etc/rdz/rsed.envvars
-> change: -Ddaemon.log=/var/rdz/logs/$RSE_RSED_PORT
-> change: -Duser.log=/var/rdz/logs/$RSE_RSED_PORT
-> add at the END:
-- NEEDED BY CLONES TO FIND THE REMAINING CONFIGURATION FILES
CFG_BASE=/etc/rdz
CLASSPATH=.:$CFG_BASE:$CLASSPATH
--

```

2. Préparez les autres fichiers de configuration (différents des fichiers `rsed.envvars`) qui diffèrent entre le document maître (non-SSL) et le clone (SSL).

```

$ mkdir /etc/rdz/ssl
$ cp /etc/rdz/ssl.properties /etc/rdz/etc/rdz/ssl
$ oedit /etc/rdz/ssl/ssl.properties
-> change: change as needed

```

3. Créez une tâche démarrée `RSED` qui clonera le fichier `rsed.envvars` de base et modifiez le port de démon RSE (4035 -> 4034).

```

/**
/** RSE DAEMON - SSL
/**
//RSED PROC IVP=, * 'IVP' to do an IVP test
// 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=*
// *
// * start RSED with the newly created rsed.envvars
// *
//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
// *

```

---

## Dans tous les autres cas

Lorsque des modifications de code sont effectuées (maintenance, prévisualisation technique, nouvelle édition) ou que les modifications que vous effectuez sont complexes, il est recommandé de procéder à une autre installation de Developer for System z. La présente section décrit les points possibles de conflit entre différentes installations.

La liste ci-dessous décrit brièvement les éléments qui doivent être différents entre les instances de Developer for System z (fortement recommandé) :

- SMP/E CSI
- Bibliothèques d'installation
- Port TCP/IP du moniteur de travaux JES, ainsi que son fichier de configuration FEJJC�FG
- Langage JCL d'initialisation du moniteur de travaux JES
- Nom de transaction APPC
- Fichiers de configuration RSE, rsed.envvars, \*.properties et \*.conf
- Port TCP/IP RSE
- Langage JCL de démarrage de RSE

Vous trouverez ci-dessous une présentation plus détaillée :

- SMP/E CSI
  1. Installez chaque instance de Developer for System z dans un CSI distinct. SMP/E empêchera une seconde installation du même FMID dans un CSI mais acceptera l'installation d'un autre FMID. Si le second FMID correspond à une version plus récente, elle remplace la version existante du produit. Si le second FMID est une version plus ancienne, l'installation échoue en raison de noms de partie en double.
- Bibliothèques d'installation
  1. Installez chaque instance de Developer for System z dans des fichiers et des répertoires différents. N'oubliez pas que vous pouvez uniquement modifier le chemin z/OS UNIX en indiquant un préfixe pour le chemin /usr/lpp/rdz fourni par défaut par IBM. Voici un exemple valide : /service/usr/lpp/rdz .

2. Le travail de définition de la configuration FEK.SFEKSAMP(FEKSETUP) crée les fichiers et les répertoires utilisés pour enregistrer les fichiers de configuration. Les fichiers de configuration doivent être uniques et c'est la raison pour laquelle vous devez utiliser des noms de fichiers et de répertoires uniques lorsque vous soumettez ce travail pour éviter de remplacer les personnalisations existantes.
- Composants obligatoires
    1. Le fichier de configuration du moniteur de travaux JES  
FEK.#CUST.PARMLIB(FEJJCNFG) contient son numéro de port TCP/IP et ne peut par conséquent pas être partagé. Le membre lui-même peut être renommé (si le langage JCL est également mis à jour), afin de pouvoir placer toutes les versions personnalisées de ce membre dans un même fichier, si vous n'effectuez pas les mises à jour dans le fichier d'installation.
    2. Le langage JCL de démarrage du moniteur de travaux JES  
FEK.#CUST.PROCLIB(JMON) se rapporte à FEJJCNFG et ne peut pas être partagé non plus. Après avoir renommé le membre (et la carte de travail si vous l'avez démarrée en tant que travail), vous pouvez placer tous ceux du langage JCL dans le même fichier.
    3. Le fichier de configuration RSE /etc/rdz/rsed.envvars contient des références au chemin d'installation et éventuellement au chemin du journal du serveur, qui doit être unique. Le nom de fichier est obligatoire, vous ne pouvez donc pas conserver les différentes copies dans le même répertoire.
    4. Le fichier de configuration ISPF.conf contient une référence à FEK.SFEKPROC(FEKFRSRV), le serveur de commandes TSO. Elle est propre au niveau du logiciel, par conséquent, vous devez créer un fichier ISPF.conf.
    5. Tous les autres fichiers de configuration z/OS UNIX (\*.properties, par exemple) doivent résider dans le même répertoire que rsed.envvars et ne peuvent donc pas être partagés, étant donné que rsed.envvars doit se trouver dans un emplacement non partagé.
    6. Le langage JCL de démarrage de FEK.#CUST.PROCLIB(RSED) ne peut pas être partagé car il définit le numéro de port TCP/IP et possède une référence aux répertoires d'installation et de configuration, qui doit être unique. Après avoir renommé le membre (et la carte de travail si vous l'avez démarrée en tant que travail), vous pouvez placer tous ceux du langage JCL dans le même fichier.
  - Composants facultatifs
    1. Les ports TCP/IP REXEC et SSH peuvent être partagés sans aucune restriction.
    2. La transaction APPC contient une référence à FEK.SFEKPROC(FEKFRSRV), le serveur de commandes TSO. Elle est propre au niveau du logiciel, par conséquent, vous devez créer une transaction APPC par instance. Notez que, dans la mesure où le nom de la transaction APPC est modifié, la variable \_FEKFSCMD\_TP\_NAME\_ doit être définie dans rsed.envvars.
    3. Certaines procédures ELAXF\* font référence à hlq.SFEKLOAD, la bibliothèque de chargement de Developer for System z. Lisez la remarque relative à JCLLIB à la section "Procédures de construction à distance ELAXF\*" du *Guide de configuration de l'hôte* (SC11-6285) pour trouver une éventuelle solution à la mise à disposition de plusieurs ensembles auprès des utilisateurs.
    4. Pour activer deux instances de la procédure mémorisée DB2, vous devez procéder comme suit. Notez toutefois qu'il s'agit d'une description présentée en l'état, non prise en charge :
      - a. Copiez hlq.SFEKPROC(ELAXMREX) dans un membre appelé différemment, par exemple, ELAXMRXX.

- b. Copiez le modèle de membre hlq.SFEKSAMP(ELAXMSAM) dans un membre appelé différemment, par exemple, ELAXMWDZ.
- c. Modifiez le modèle de membre hlq.SFEKSAMP(ELAXMJCL) pour refléter ces modifications de dénomination, par exemple :

```
//SYSIN DD *
CREATE PROCEDURE SYSPROC.ELAXMRXX
 (IN FUNCTION_REQUEST VARCHAR(20) CCSID EBCDIC
 ...
 , OUT RETURN_VALUE VARCHAR(255) CCSID EBCDIC)
PARAMETER STYLE GENERAL RESULT SETS 1
LANGUAGE REXX
EXTERNAL NAME ELAXMRXX
COLLID DSNREXCS WLM ENVIRONMENT ELAXMWDZ
PROGRAM TYPE MAIN
MODIFIES SQL DATA
STAY RESIDENT NO
COMMIT ON RETURN NO
ASUTIME NO LIMIT
SECURITY USER;

COMMENT ON PROCEDURE SYSPROC.ELAXMRXX IS
'PLI & COBOL PROCEDURE PROCESSOR (ELAXMRXX), INTERFACE LEVEL 0.01';

GRANT EXECUTE ON PROCEDURE SYSPROC.ELAXMRXX TO PUBLIC;
//
```

- d. Effectuez la personnalisation, comme indiqué à la section "(Facultatif) Procédure mémorisée DB2" du *Guide de configuration de l'hôte* (SC11-6285) en utilisant toutefois de nouveaux membres.
  - e. Le nouveau nom de l'environnement WLM (ELAXMWDZ par exemple) doit être utilisé dans l'assistant de la procédure mémorisée DB2 du client.
5. La prise en charge bidirectionnelle dans des régions CICS dépend d'un membre de bibliothèque de chargement et par conséquent n'est pas partageable entre les éditions. Toutefois, si le nom du module de chargement est identique pour toutes les instances, vous pouvez partager la version la plus récente entre les instances, et même entre les éditions. La compatibilité amont est indisponible si le nom du module de chargement a été modifié.
  6. Les modules de chargement du gestionnaire de déploiement d'application qui sont inclus dans des régions CICS présentent une compatibilité amont, ainsi, la version la plus récente peut être partagée entre les éditions.
  7. La méthode d'accès VSAM CRD du gestionnaire de déploiement d'application présente une compatibilité amont, et la version la plus récente peut ainsi être partagée entre les éditions.
  8. Les définitions de ressource CICS du gestionnaire de déploiement d'application présentent une compatibilité amont, la version la plus récente pouvant ainsi être partagée entre les éditions.
  9. Les méthodes d'accès VSAM CARMA peuvent changer d'un niveau de logiciel à l'autre. Il est donc recommandé de ne pas les partager.



---

## Chapitre 12. Traitement des incidents liés à la configuration

Ce chapitre vous aide à résoudre certains problèmes fréquents qui peuvent se produire au cours de la configuration de Developer for System z. Il comporte les sections suivantes :

- «Journal et analyse de configuration à l'aide de FEKLOGS»
- «Fichiers journaux», à la page 172
- «Fichiers de vidage», à la page 177
- «Traçage», à la page 180
- «Bits d'autorisation z/OS UNIX», à la page 182
- «Ports TCP/IP réservés», à la page 185
- «Taille d'espace adresse», à la page 187
- «Informations diverses», à la page 188

Le guide *Developer for System z Messages and Codes* (SC14-7497) présente les messages et les codes retour générés par les composants Developer for System z. Le document *Developer for System z - Réponses aux questions les plus courantes en matière de maintenance et de configuration d'hôte* (SC14-7373) décrit différents problèmes pouvant survenir ainsi que la résolution de ces derniers.

Pour plus d'informations, reportez-vous à la section Support du site Web de Developer for System z (<http://www-03.ibm.com/software/products/us/en/developerforsystemz/>) pour consulter les notes techniques et disposer des dernières informations produites par notre équipe de support technique.

Dans la section Library du site Web (<http://www-01.ibm.com/support/docview.wss?uid=swg27038517>), vous pouvez également consulter la dernière version de la documentation de Developer for System z, notamment les livres blancs.

Le centre de documentation de Developer for System z (<http://pic.dhe.ibm.com/infocenter/ratdevz/v9r0/index.jsp>) fournit des informations sur le client Developer for System z et son interaction avec l'hôte (du point de vue du client).

La bibliothèque z/OS en ligne contient également des informations importantes, que vous pouvez consulter à l'adresse suivante : <http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/>.

Veuillez nous tenir informé de l'absence d'une certaine fonction de Developer for System z. Vous pouvez ouvrir une demande d'amélioration (RFE) à l'adresse suivante :

<https://www.ibm.com/developerworks/support/rational/rfe/>

---

### Journal et analyse de configuration à l'aide de FEKLOGS

Developer for System z propose un modèle de travail (FEKLOGS), qui rassemble tous les fichiers journaux z/OS UNIX, ainsi que les informations d'installation et de configuration relatives à Developer for System z.

Le modèle de travail FEKLOGS se trouve dans FEK.#CUST.JCL, sauf si vous avez indiqué un autre emplacement lorsque vous avez personnalisé et soumis le travail

FEK.SFEKSAMP(FEKSETUP). Reportez-vous à la section "Configuration personnalisée" du *Guide de configuration de l'hôte* (SC11-6285) pour obtenir plus de détails.

La personnalisation de FEKLOGS est présentée dans JCL. La personnalisation porte sur la mise à disposition de quelques variables clés.

**Remarque :** Les clients SDSF peuvent utiliser la ligne de commande **XDC** dans SDSF pour sauvegarder la sortie de travaux dans un fichier, qui peut être donné au point de service IBM. Il convient de noter que le fichier de sortie doit être alloué en tant que VB 2051 (la valeur par défaut dans SDSF est VB 240) pour éviter la troncature de l'enregistrement.

---

## Fichiers journaux

Developer for System z crée des fichiers journaux utiles pour vous et pour le point de service IBM dans l'identification et la résolution des incidents. La liste ci-après présente les fichiers journaux que vous pouvez créer sur le système hôte z/OS. Situé en regard des journaux spécifiques au produit, vérifiez bien le SYSLOG de tous les messages associés.

Les fichiers journaux basés sur le système MVS peuvent être localisés par l'intermédiaire de l'instruction de définition de données appropriée. Les fichiers journaux basés sur z/OS UNIX sont situés dans les répertoires suivants :

- userlog/\$LOGNAME/

Les fichiers journaux propres à l'utilisateur sont placés dans userlog/\$LOGNAME/, où userlog est la valeur combinée des directives user.log et DSTORE\_LOG\_DIRECTORY dans rsed.envvars, et \$LOGNAME l'ID utilisateur de connexion (en majuscules). Si la directive user.log est mise en commentaire ou omise, le chemin du répertoire de base est utilisé. Ce chemin est défini dans le segment de sécurité OMVS de l'ID utilisateur. Si la directive DSTORE\_LOG\_DIRECTORY est mise en commentaire ou omise, .eclipse/RSE/ est ajouté à la valeur user.log.

- .dstoreMemLogging - Consignation sur l'utilisation de la mémoire DataStore
- .dstoreTrace - Consignation sur l'action DataStore
- .dstoreHashMap.\* - Image instantanée de la mappe de hachage DataStore active
- .dstoreStackTrace.\* - Image instantanée des unités d'exécution DataStore actives et de l'endroit où elles ont été appelées
- ffs.log - Journal du serveur FFS (Foreign File System), qui exécute des fonctions MVS natives
- ffsget.log - Journal du programme de lecture de fichier, qui lit un fichier séquentiel ou un membre d'un fichier partitionné
- ffsput.log - Journal du programme d'écriture de fichier, qui écrit un fichier séquentiel ou un membre d'un fichier partitionné
- lock.log - Journal du gestionnaire de verrouillage, qui verrouille/déverrouille un fichier séquentiel ou un membre d'un fichier partitionné
- rmt\_class\_loader.cache.jar - Mémoire cache des classes chargées par le programme de chargement de classe distante RSE
- rsecomm.log - Journal du serveur RSE qui traite les commandes du client et la consignation des communications de tous les services qui utilisent RSE (peut contenir une trace de pile d'exception Java)

**Remarque :**



- Les noms du répertoire `.eclipse` et des fichiers journaux `.dstore*` commencent par un point (.) et sont par conséquent cachés. Utilisez la commande `z/OS UNIX ls -lA` pour répertorier les fichiers et répertoires cachés. Lorsque vous utilisez le client Developer for System z, sélectionnez la page **Fenêtre > Préférences... > Systèmes distants > Fichiers** et activez "Afficher les fichiers cachés".

- **rép\_base\_utilisateur**

Les fichiers journaux propres au démon RSE et au pool d'unités d'exécution RSE se trouvent dans le répertoire `rép_base_utilisateur`, où `rép_base_utilisateur` est la valeur de la directive `daemon.log` dans `rsed.envvars`. Si la directive `daemon.log` est mise en commentaire ou omise, le répertoire de base de l'ID utilisateur affecté à la tâche démarrée RSED est utilisé. Le répertoire de base est défini dans le segment de sécurité OMVS de l'ID utilisateur.

- `rsedaemon.log` - Journal du démon RSE
- `rseserver.log` - Journal des pools d'unités d'exécution RSE
- `audit.log` - Trace d'audit RSE
- `serverlogs.count` - Compteur des flux du pool d'unités d'exécution RSE
- `stderr.*.log` - Flux d'erreurs standard du pool d'unités d'exécution RSE
- `stdout.*.log` - Fichier stream de sortie standard du pool d'unités d'exécution RSE

- **/tmp**

Les fichiers journaux du programme de vérification d'installation se situent dans le répertoire référencé par `TMPDIR`, si cette variable est définie dans `rsed.envvars`. Si ce n'est pas le cas, les fichiers sont créés dans le répertoire `/tmp`.

- `fekfivpi.log` - Journal du test du programme de vérification d'installation `fekfivpi`
- `fekfivps.log` - Journal du test du programme de vérification d'installation `fekfivps`
- `fekfivpc.log` - Journal de communication du test du programme de vérification d'installation `fekfivpc`

**Remarque :** Il existe des commandes de l'opérateur qui permettent de contrôler la quantité de données consignées dans certains des fichiers journaux mentionnés. Reportez-vous à la section "Commandes de l'opérateur" du *Guide de configuration de l'hôte* (SC11-6285) pour plus d'informations.

## Journalisation du moniteur de travaux JES

- **SYSOUT DD**

Journalisation des opérations classiques. La valeur par défaut du modèle JCL `FEK.#CUST.PROCLIB(JMON)` est `SYSOUT=*`.

- **SYSPRINT DD**

Journalisation de trace. La valeur par défaut du modèle JCL `FEK.#CUST.PROCLIB(JMON)` est `SYSOUT=*`. La fonction de trace est activée à l'aide du paramètre `-TV`, voir «Fonction de trace du moniteur de travaux JES», à la page 180 pour des informations détaillées.

## Journalisation du démon RSE et du pool d'unités d'exécution

- **STDOUT DD**

Données réacheminées de `stdout`, la sortie Java standard du démon RSE. La valeur par défaut du modèle JCL `FEK.#CUST.PROCLIB(RSED)` est `SYSOUT=*`.

- **STDERR DD**

Données réacheminées de stderr, la sortie d'erreur standard Javadu démon RSE. La valeur par défaut du modèle JCL FEK.#CUST.PROCLIB(RSED) est SYSOUT=\*

- **rép\_base\_utilisateur**

Les fichiers journaux propres au démon RSE et au pool d'unités d'exécution RSE se trouvent dans le répertoire rép\_base\_utilisateur, où rép\_base\_utilisateur est la valeur de la directive daemon.log dans rsed.envvars. Si la directive daemon.log est mise en commentaire ou omise, le répertoire de base de l'ID utilisateur affecté à la tâche démarrée RSED est utilisé. Le répertoire de base est défini dans le segment de sécurité OMVS de l'ID utilisateur.

- rsedaemon.log - Journal du démon RSE
- rseserver.log - Journal des pools d'unités d'exécution RSE
- audit.log - Trace d'audit RSE
- serverlogs.count - Compteur des flux du pool d'unités d'exécution RSE
- stderr.\*.log - Flux d'erreurs standard du pool d'unités d'exécution RSE
- stdout.\*.log - Fichier stream de sortie standard du pool d'unités d'exécution RSE

**Remarque :**

- serverlogs.count, stderr.\*.log et stdout.\*.log sont uniquement créés si la directive enable.standard.log de rsed.envvars est active ou si la fonction est activée dynamiquement avec la commande de l'opérateur **modify rsestandardlog on**.
- Par défaut, le signe \* dans stderr.\*.log et stdout.\*.log est 1. Toutefois, il peut exister plusieurs pools d'unités d'exécution RSE, auquel cas le nombre est incrémenté pour chacun d'eux afin de garantir le caractère unique des noms de fichier.
- Il existe des commandes de l'opérateur qui permettent de contrôler la quantité de données consignées dans certains des fichiers journaux mentionnés. Reportez-vous à la section "Commandes de l'opérateur" du *Guide de configuration de l'hôte* (SC11-6285) pour plus d'informations.
- Les fichiers rse\*.log peuvent également exister avec l'extension ".last" au lieu de l'extension ".log" si keep.last.log=true est spécifié dans rsed.envvars. Par défaut, les fichiers journaux ".last" ne sont pas créés.
- Les fichiers rse\*.log auront un nom étendu si keep.all.logs=true est spécifié dans rsed.envvars. Par défaut, le nom étendu est utilisé. Voici un exemple de nom étendu, dans lequel RSED représente le nom d'espace adresse du démon RSE et aaaammjjhhmmss correspond à l'horodatage (année, mois, jour, heure, minute, seconde) : rseserver.RSED#aaaammjjhhmmss.log

## Journalisation pour l'utilisateur RSE

- **userlog/\$LOGNAME/**

Plusieurs fichiers journaux sont créés par les composants associés à RSE. Ils sont tous placés dans userlog/\$LOGNAME/, où userlog est la valeur combinée des directives user.log et DSTORE\_LOG\_DIRECTORY dans rsed.envvars, et \$LOGNAME l'ID utilisateur de connexion (en majuscules). Si la directive user.log est mise en commentaire ou omise, le chemin du répertoire de base est utilisé. Ce chemin est défini dans le segment de sécurité OMVS de l'ID utilisateur. Si la directive DSTORE\_LOG\_DIRECTORY est mise en commentaire ou omise, .eclipse/RSE/ est ajouté à la valeur user.log.

- .dstoreMemLogging - Consignation sur l'utilisation de la mémoire DataStore

- .dstoreTrace - Consignation sur l'action DataStore
- .dstoreHashmap.\* - Image instantanée de la mappe de hachage DataStore active
- .dstoreStackTrace.\* - Image instantanée des unités d'exécution DataStore actives et de l'endroit où elles ont été appelées
- ffs.log - Journal du serveur FFS (Foreign File System), qui exécute des fonctions MVS natives
- ffsget.log - Journal du programme de lecture de fichier, qui lit un fichier séquentiel ou un membre d'un fichier partitionné
- ffsput.log - Journal du programme d'écriture de fichier, qui écrit un fichier séquentiel ou un membre d'un fichier partitionné
- lock.log - Journal du gestionnaire de verrous, qui verrouille et déverrouille un fichier séquentiel ou un membre d'un fichier partitionné
- rmt\_class\_loader.cache.jar - Mémoire cache des classes chargées par le programme de chargement de classe distante RSE
- rsecomm.log - Journal du serveur RSE qui traite les commandes du client et la consignation des communications de tous les services qui utilisent RSE (peut contenir une trace de pile d'exception Java)

#### Remarque :

- Les noms du répertoire .eclipse et des fichiers journaux .dstore\* commencent par un point (.) et sont par conséquent cachés. Utilisez la commande z/OS UNIX **ls -lA** pour répertorier les fichiers et répertoires cachés. Lorsque vous utilisez le client Developer for System z, sélectionnez la page **Fenêtre > Préférences... > Systèmes distants > Fichiers** et activez "Afficher les fichiers cachés".
- La création des fichiers journaux .dstore\* est contrôlée par les options de démarrage -DDSTORE\_\* Java, comme indiqué dans "Définition de paramètres de démarrage supplémentaires Java avec \_RSE\_JAVAOPTS" dans *Guide de configuration de l'hôte* (SC11-6285).
- Les fichiers journaux .dstore\* sont créés en ASCII. Utilisez la commande z/OS UNIX **iconv -f ISO8859-1 -t IBM-1047 .dstore\*** pour les afficher en code EBCDIC (avec la page de codes IBM-1047).
- Contrairement à tous les fichiers \*.log, les fichiers journaux .dstore\* ne sont pas supprimés automatiquement lors de la reconnexion du client. La suppression de ces fichiers est une action manuelle.
- Il existe des commandes de l'opérateur qui permettent de contrôler la quantité de données consignées dans certains des fichiers journaux mentionnés. Reportez-vous à la section "Commandes de l'opérateur" du *Guide de configuration de l'hôte* (SC11-6285) pour plus d'informations.
- Les fichiers ffs\*.log, lock.log et rsecomm.log peuvent également exister avec l'extension ".last" au lieu de l'extension ".log" si keep.last.log=true est spécifié dans rsed.envvars. Par défaut, les fichiers journaux ".last" ne sont pas créés.
- Les fichiers ffs\*.log, lock.log et rsecomm.log auront un nom étendu si keep.all.logs=true est spécifié dans rsed.envvars. Par défaut, le nom étendu est utilisé. Voici un exemple de nom étendu, dans lequel RSEDx représente le nom d'espace adresse du pool d'unités d'exécution dans lequel l'utilisateur est actif et aaaammjjhhmmss correspond à l'horodatage (année, mois, jour, heure, minute, seconde) : ffs.RSEDx#aaaammjjhhmmss.log

## SCLM Developer Toolkit, journalisation

- userlog/\$LOGNAME/rsecomm.log

Journalisation de la communication de SCLM Developer Toolkit, où `userlog` est la valeur combinée des directives `user.log` et `DSTORE_LOG_DIRECTORY` dans `rsed.envvars`, et `$LOGNAME` l'ID utilisateur de connexion (en majuscules). Si la directive `user.log` est mise en commentaire ou omise, le chemin du répertoire de base est utilisé. Ce chemin est défini dans le segment de sécurité OMVS de l'ID utilisateur. Si la directive `DSTORE_LOG_DIRECTORY` est mise en commentaire ou omise, `.eclipse/RSE/` est ajouté à la valeur `user.log`.

## Journalisation CARMA

- **Travail de serveur CARMA**

Quand vous ouvrez une connexion avec CARMA à l'aide de l'interface de traitement par lots, `FEK.#CUST.SYSPROC(CRASUBMT)` démarre un travail de serveur (avec l'ID utilisateur de l'utilisateur en tant que propriétaires) appelé `CRAport`, où port est le port TCP/IP utilisé.

- **Définition de données CARMALOG**

Si l'instruction de définition de données CARMALOG est indiquée dans la méthode de démarrage CARMA sélectionnée, le journal de CARMA est réacheminé vers cette instruction de définition de données dans le travail de serveur. Dans le cas contraire, elle est dirigée vers `SYSPRINT`.

- **SYSPRINT DD**

L'instruction de définition de données `SYSPRINT` du travail de serveur contient le journal de CARMA, si l'instruction de définition de données CARMALOG n'est pas définie.

- **SYSTSPRT DD**

L'instruction de définition de données `SYSTSPRT` du travail de serveur contient les messages système (TSO) pour le démarrage du serveur CARMA.

- **userlog/\$LOGNAME/rsecomm.log**

Journalisation de la communication de CARMA, où `userlog` est la valeur combinée des directives `user.log` et `DSTORE_LOG_DIRECTORY` dans `rsed.envvars`, et `$LOGNAME` l'ID utilisateur de connexion (en majuscules). Si la directive `user.log` est mise en commentaire ou omise, le chemin du répertoire de base est utilisé. Ce chemin est défini dans le segment de sécurité OMVS de l'ID utilisateur. Si la directive `DSTORE_LOG_DIRECTORY` est mise en commentaire ou omise, `.eclipse/RSE/` est ajouté à la valeur `user.log`.

## Consignation des tests du programme de vérification de l'installation fekfivpc

- **/tmp/fekfivpc.log**

La commande `fekfivpc` (test du programme de vérification de l'installation lié à CARMA) crée le fichier `fekfivpc.log` pour documenter la communication entre RSE et CARMA. Le journal est créé dans le répertoire référencé par `TMPDIR`, si cette variable est définie dans `rsed.envvars`. Si ce n'est pas le cas, le fichier est créé dans le répertoire `/tmp`.

## Consignation des tests du programme de vérification de l'installation (IVP) fekfivpi

- **/tmp/fekfivpi.log**

Sortie de la commande `fekfivpi -file` (test du programme de vérification de l'installation lié à la passerelle client TSO/ISPF). Le journal est créé dans le répertoire référencé par `TMPDIR`, si cette variable est définie dans `rsed.envvars`. Si ce n'est pas le cas, le fichier est créé dans le répertoire `/tmp`.

## Consignation des tests de la procédure de vérification d'installation fekfivps

- /tmp/fekfivps.log

Sortie de la commande fekfivps -file (test du programme de vérification de l'installation lié à SCLMDT). Le journal est créé dans le répertoire référencé par TMPDIR, si cette variable est définie dans rsed.envvars. Si ce n'est pas le cas, le fichier est créé dans le répertoire /tmp.

## Journalisation de la révision du code

- SYSTSPRT DD

La définition de données SYSTSPRT de l'étape appelant la procédure de révision du code contient les messages du système frontal qui pilote le processus d'analyse de code.

- WORKSPCE DD

La définition de données WORKSPCE de l'étape appelant la procédure de révision du code contient les messages de journal de l'espace de travail Eclipse du processus d'analyse de code.

- ERRMSGs DD

La définition de données ERRMSGs de l'étape appelant la procédure de révision du code contient la sortie stderr du processus d'analyse de code.

## Journalisation de la couverture de code

- SYSTSPRT DD

La définition de données SYSTSPRT de l'étape appelant la procédure de révision du code contient les messages du système frontal qui pilote le processus d'analyse de code.

- WORKSPCE DD

La définition de données WORKSPCE de l'étape appelant la procédure de révision du code contient les messages de journal de l'espace de travail Eclipse du processus d'analyse de code.

- ERRMSGs DD

La définition de données ERRMSGs de l'étape appelant la procédure de révision du code contient la sortie stderr du processus d'analyse de code.

---

## Fichiers de vidage

Quand un produit subit une fin anormale, un vidage mémoire est créé pour aider à l'identification de l'incident. La disponibilité et l'emplacement de ces fichiers de vidage dépendent pour une grande part des paramètres spécifiques du site. Les vidages ne peuvent pas être créés, ou les vidages peuvent être créés dans des emplacements différents de ceux mentionnés dans les sections suivantes.

## Fichiers de vidage MVS

Quand le programme fonctionne sous MVS, vérifiez les fichiers de vidage système ainsi que votre JCL pour les instructions de définition de données suivantes (selon le produit) :

- SYSABEND
- SYSMDUMP
- SYSUDUMP
- CEEDUMP

- SYSPRINT
- SYSOUT

Pour plus d'informations sur ces instructions de définition de données, voir *MVS JCL Reference* (SA22-7597) et *Language Environment Debugging Guide* (GA22-7560).

## Fichiers de vidage Java

Dans z/OS UNIX, la plupart des fichiers de vidage de Developer for System z sont commandés par la machine virtuelle Java (JVM).

La JVM crée un ensemble d'agents de vidage par défaut lors de son initialisation (SYSTDUMP et JAVADUMP). Vous pouvez changer cet ensemble d'agents de vidage à l'aide de la variable d'environnement `JAVA_DUMP_OPTS` et même changer l'ensemble à l'aide de `-Xdump` sur la ligne de commande. Les options de ligne de commande de la JVM sont définies dans la directive `_RSE_JAVA0PTS` de `rsed.envvars`. Ne modifiez pas les paramètres de vidage sans instruction du point service IBM.

**Remarque :** Vous pouvez utiliser l'option `-Xdump:what` sur la ligne de commande pour déterminer quels agents de vidage existent à l'exécution du démarrage.

Les types de vidage qui peuvent être produits sont :

### SYSTDUMP

Cliché de transaction Java. Vidage mémoire non formaté généré par z/OS.

Le vidage est consigné dans un fichier séquentiel MVS avec un nom par défaut au format `%uid.JVM.TDUMP.%job.D%ym%d.T%H%M%S`, ou selon la configuration de la variable d'environnement `JAVA_DUMP_TDUMP_PATTERN`.

**Remarque :** `JAVA_DUMP_TDUMP_PATTERN` permet l'utilisation de variables qui sont converties en valeurs réelles lorsque le cliché de la transaction est effectué.

Tableau 38. Variables `JAVA_DUMP_TDUMP_PATTERN`

| Variable          | Utilisation          |
|-------------------|----------------------|
| <code>%uid</code> | ID utilisateur       |
| <code>%job</code> | Nom du travail       |
| <code>%y</code>   | Année (2 chiffres)   |
| <code>%m</code>   | Mois (2 chiffres)    |
| <code>%d</code>   | Jour (2 chiffres)    |
| <code>%H</code>   | Heure (2 chiffres)   |
| <code>%M</code>   | Minute (2 chiffres)  |
| <code>%S</code>   | Seconde (2 chiffres) |

### CEEDUMP

Fichier de vidage Language Environment (LE). Récapitulatif formaté de vidage système qui montre les traces de pile pour chaque unité d'exécution du processus JVM, avec les informations de registre et un stockage de vidage court pour chaque registre.

Le vidage est inscrit dans un fichier z/OS UNIX nommé `CEEDUMP.yyyymmdd.hhmmss.pid`, où `yyyymmdd` est la date du jour, `hhmmss` est



l'heure et pid est l'ID processus en cours. Les emplacements possibles de ce fichier sont décrits dans «Emplacements des fichiers de vidage z/OS UNIX».

### HEAPDUMP

Vidage formaté (liste) des objets qui sont sur le tas Java.

Le vidage est inscrit dans un fichier z/OS UNIX nommé `HEAPDUMP.yyyymmdd.hhmmss.pid.TXT`, où `yyyymmdd` est la date du jour, `hhmmss` est l'heure et `pid` est l'ID processus en cours. Les emplacements possibles de ce fichier sont décrits dans «Emplacements des fichiers de vidage z/OS UNIX».

Notez que Developer for System z fournit une commande de l'opérateur pour déclencher ce vidage. Pour plus d'informations, reportez-vous au chapitre "Commandes de l'opérateur" du manuel *Guide de configuration de l'hôte* (SC23-7658).

### JAVADUMP

Analyse formatée de la JVM. Contient des données de diagnostic relatives à la JVM et à l'application Java (l'environnement d'application, les unités d'exécution, les piles natives, les verrous et la mémoire, par exemple).

Le vidage est inscrit dans un fichier z/OS UNIX nommé `JAVADUMP.yyyymmdd.hhmmss.pid.TXT`, où `yyyymmdd` est la date du jour, `hhmmss` est l'heure et `pid` est l'ID processus en cours. Les emplacements possibles de ce fichier sont décrits dans «Emplacements des fichiers de vidage z/OS UNIX».

Notez que Developer for System z fournit une commande de l'opérateur pour déclencher ce vidage. Pour plus d'informations, reportez-vous au chapitre "Commandes de l'opérateur" du manuel *Guide de configuration de l'hôte* (SC23-7658).

Pour plus d'informations sur les fichiers de vidages JVM, voir *Java Diagnostic Guide* (SC34-6358) et pour des informations spécifiques de l'environnement de langage, voir *Language Environment Debugging Guide* (GA22-7560).

## Emplacements des fichiers de vidage z/OS UNIX

La machine virtuelle Java vérifie l'existence et les droits d'accès en écriture pour chacun des emplacements suivants, et stocke les fichiers CEEDUMP, HEAPDUMP et JAVADUMP dans le premier emplacement disponible. Notez que vous devez disposer d'un espace disque suffisant pour que le fichier de vidage soit écrit correctement.

1. Le répertoire dans la variable d'environnement `_CEE_DMPTARG`, s'il est trouvé. Cette variable est définie dans `rsed.envvars` en tant que `/tmp`. Elle peut être remplacée par `/dev/null` afin d'éviter de créer les fichiers de vidage.
2. Le répertoire de travail en cours, s'il ne s'agit pas du répertoire de base (`/`), et qu'il est inscriptible.
3. Le répertoire dans la variable d'environnement `TMPDIR` (une variable d'environnement qui indique l'emplacement d'un répertoire temporaire autre que `/tmp`), s'il est trouvé.
4. Le répertoire `/tmp`.
5. Si le vidage ne peut être stocké dans aucun des emplacements mentionnés précédemment, il est mis dans `stderr`.

### Fonction de trace du moniteur de travaux JES

La fonction de trace du moniteur de travaux JES est contrôlée par l'opérateur système, comme l'indique la section "Commandes de l'opérateur" du *Guide de configuration de l'hôte* (SC11-6285).

- Le lancement de la tâche activée JMON avec le paramètre PRM=-TV active le mode prolix (fonction de trace).
- Les commandes de l'opérateur **modify trace** et **modify message** vous permettent de sélectionner le niveau de détails souhaité pour les messages de journal.

### RSE, traçage

Plusieurs fichiers journaux sont créés par les composants associés à RSE. La plupart se trouve dans `userlog/$LOGNAME/`, où `userlog` est la valeur combinée des directives `user.log` et `DSTORE_LOG_DIRECTORY` dans `rsed.envvars`, et `$LOGNAME` l'ID utilisateur de connexion (en majuscules). Si la directive `user.log` est mise en commentaire ou omise, le chemin du répertoire de base est utilisé. Ce chemin est défini dans le segment de sécurité OMVS de l'ID utilisateur. Si la directive `DSTORE_LOG_DIRECTORY` est mise en commentaire ou omise, `.eclipse/RSE/` est ajouté à la valeur `user.log`.

La quantité de données consignées dans `ffs*.log`, `lock.log` et `rsecomm.log` est déterminée par la commande d'opérateur **modify rsecommlog** ou par le paramètre `debug_level` dans `rsecomm.properties`. Reportez-vous à la section "Commandes de l'opérateur" du *Guide de configuration de l'hôte* (SC11-6285) et à la section "(Facultatif) Fonction de trace RSE" du *Guide de configuration de l'hôte* (SC11-6285) pour obtenir plus de détails.

La création des fichiers journaux `.dstore*` est contrôlée par les options de démarrage `--DDSTORE_*` Java, comme indiqué dans "Définition de paramètres de démarrage supplémentaires Java avec `_RSE_JAVAOPTS`" dans *Guide de configuration de l'hôte* (SC11-6285).

#### Remarque :

- Les noms du répertoire `.eclipse` et des fichiers journaux `.dstore*` commencent par un point (.) et sont par conséquent cachés. Utilisez la commande `z/OS UNIX ls -lA` pour répertorier les fichiers et répertoires cachés. Lorsque vous utilisez le client Developer for System z, sélectionnez la page **Fenêtre > Préférences... > Systèmes distants > Fichiers** et activez "Afficher les fichiers cachés".
- Les fichiers journaux `.dstore*` sont créés en ASCII. Utilisez la commande `z/OS UNIX iconv -f ISO8859-1 -t IBM-1047 .dstore*` pour les afficher en code EBCDIC (avec la page de codes IBM-1047).
- Contrairement à tous les fichiers `*.log`, les fichiers journaux `.dstore*` ne sont pas supprimés automatiquement lors de la reconnexion du client. La suppression de ces fichiers est une action manuelle.

Les fichiers journaux propres au démon RSE et au pool d'unités d'exécution RSE se trouvent dans le répertoire `rép_base_utilisateur`, où `rép_base_utilisateur` est la valeur de la directive `daemon.log` dans `rsed.envvars`. Si la directive `daemon.log` est mise en commentaire ou omise, le répertoire de base de l'ID utilisateur affecté à la tâche démarrée RSED est utilisé. Le répertoire de base est défini dans le segment de sécurité OMVS de l'ID utilisateur.



La quantité de données consignées dans `rsedaemon.log` et `rserver.log` est commandée par les commandes de l'opérateur **modify rsedaemonlog** et **modify rserverlog** ou par le paramètre `debug_level` de `rsecomm.properties`. Reportez-vous à la section "Commandes de l'opérateur" du *Guide de configuration de l'hôte* (SC11-6285) et à la section "(Facultatif) Fonction de trace RSE" du *Guide de configuration de l'hôte* (SC11-6285) pour obtenir plus de détails.

`serverlogs.count`, `stderr.*.log` et `stdout.*.log` sont uniquement créés si la directive `enable.standard.log` de `rsed.envvars` est active ou si la fonction est activée dynamiquement avec la commande de l'opérateur **modify rstandardlog on**.

## CARMA, traçage

L'utilisateur peut contrôler la quantité d'informations de trace générées par CARMA en définissant le niveau de trace dans l'onglet des propriétés de la connexion CARMA du client. Les différents Niveaux de trace sont les suivants :

- Désactiver la journalisation
- Journalisation des erreurs
- Journalisation des avertissements
- Journalisation informative
- Déboguer la journalisation

La valeur par défaut est la suivante :

Journalisation des erreurs

Pour plus d'informations sur l'emplacement des fichiers journaux, voir «Fichiers journaux», à la page 172.

## Traçage de suivi des erreurs

La procédure suivante permet de rassembler les informations nécessaires au diagnostic des incidents de suivi des erreurs avec les procédures d'assemblage à distance. La fonction de trace réduit les performances et ne doit être effectuée que sur indication du centre de support IBM. Toutes les références à `hlq` que vous trouverez dans cette section se rapportent au qualificatif de haut niveau utilisé au cours de l'installation de Developer for System z. L'installation par défaut est FEK, mais peut ne pas s'appliquer à votre site.

1. Faites une copie de sauvegarde de votre procédure de compilation active ELAXFCOC. Cette procédure est livrée par défaut dans le fichier `hlq.SFEKSAMP` mais peut être copiée à un autre emplacement, par exemple `SYS1.PROCLIB`, comme le décrit la section "Procédures de construction à distance ELAXF\*" du *Guide de configuration de l'hôte* (SC11-6285).
2. Modifiez la procédure active ELAXFCOC pour inclure la chaîne 'MAXTRACE' dans l'option de compilation `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)
```

**Remarque :** Vous devez doubler les apostrophes pour MAXTRACE. L'option est maintenant : `EXIT(ADEXIT('MAXTRACE',ELAXMGUX))`.

3. Effectuez une vérification de la syntaxe à distance sur le programme en langage COBOL pour lequel vous souhaitez obtenir des données de trace détaillées.
4. La partie SYSOUT de la sortie JES démarre en générant la liste des noms de fichier pour SIDEFILE1, SIDEFILE2, SIDEFILE3 et 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'
```

**Remarque :** Selon vos paramètres, SIDEFILE1 et SIDEFILE2 peuvent pointer vers une instruction de définition de données (SUCCESSFUL OPEN SIDEFILE1 - NAME = DD:WSEDSF1). Reportez-vous au composant JESJCL de la sortie (qui est situé avant le composant SYSOUT) pour connaître le nom réel du fichier.

```
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. Copiez ces quatre fichiers sur votre PC, par exemple en créant un projet local en langage COBOL dans Developer for System z et en ajoutant les fichiers SIDEFILE1->4.
6. Copiez le journal de travail JES sur votre PC, par exemple, en ouvrant la sortie de travaux dans Developer for System z et en la sauvegardant dans le projet local à l'aide de **Fichier > Enregistrer sous...**
7. Restaurez la procédure ELAXFCOC à son état original, soit en annulant les modifications (retirez la chaîne "MAXTRACE" des options de compilation) soit en restaurant la sauvegarde.
8. Envoyez les fichiers collectés (SIDEFILE1->4 et le journal de travail) au service d'assistance technique IBM.

---

## Bits d'autorisation z/OS UNIX

Developer for System z requiert que le système de fichiers z/OS UNIX et certains fichiers z/OS UNIX comportent des données de droits spécifiques définies.

### SETUID, attribut du système de fichiers

L'Explorateur de systèmes distants (RSE) est le composant Developer for System z qui fournit des services de base, comme la connexion du client à l'hôte. Il doit pouvoir effectuer des tâches telles que la création de l'environnement de sécurité de l'utilisateur.

Le système de fichiers (HFS ou zFS) dans lequel Developer for System z est installé doit être monté avec le contrôle des données de droits SETUID activé (il s'agit de la valeur par défaut du système). Le montage du système de fichier avec le paramètre NOSETUID empêchera Developer for System z de créer l'environnement de sécurité utilisateur et l'exécution de la requête de connexion. Vous trouverez ci-dessous d'autres indicateurs de ce problème de configuration :

- Message de console "FEK999E The module, fekfomvs must be marked as APF-authorized"
- Echec du programme de vérification d'installation (IVP) de passTicket avec le message "ICH409I 282-010 ABEND DURING RACHECK PROCESSING"

Plusieurs erreurs (messages BPXP014I et BPXP015I, par exemple) peuvent survenir si les systèmes de fichiers hébergeant les binaires Java ou z/OS UNIX sont montés à l'aide du paramètre NOSETUID.

Utilisez la commande TSO **ISHELL** afin de répertorier l'état actuel des données SETUID. Dans le panneau ISHELL, sélectionnez **Systèmes\_de\_fichiers > 1. Table de montage...** pour répertorier les systèmes de fichiers montés. La commande-ligne **a** affichera les attributs du système de fichiers sélectionné où le champ "Ignorer SETUID" sera 0.

## Autorisation de contrôle de programmes

L'Explorateur de systèmes distants (RSE) est le composant Developer for System z qui fournit des services de base, comme la connexion du client à l'hôte. L'exécution doit être contrôlée par le programme afin d'effectuer des tâches telles que la commutation sur l'ID utilisateur du client.

Les données de contrôle de programmes z/OS UNIX sont définies au cours de l'installation SMP/E si nécessaire, sauf pour l'interface Java à votre produit de sécurité (voir le Chapitre 2, «Remarques relatives à la sécurité», à la page 19). Cette donnée de droits pourrait être perdue si vous ne la conservez pas lors de la copie manuelle des répertoires Developer for System z.

Les fichiers Developer for System z suivants doivent être contrôlés par programme :

- /usr/lpp/rdz/bin/
  - fekfdivp
  - fekfomvs
  - fekfriwp
- /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

Utilisez la commande z/OS UNIX **ls -E** pour répertorier les attributs étendus, dans lesquels le bit de contrôle de programme est marqué avec la lettre **p**, comme indiqué dans l'exemple suivant (\$ est l'invite 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
```

Utilisez la commande z/OS UNIX **extattr +p** pour définir manuellement le bit de contrôle de programme, comme indiqué dans l'exemple suivant (\$ et # sont les invites 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
```

**Remarque :** Pour utiliser la commande **extattr +p**, vous devez disposer au moins d'un accès READ au profil BPX.FILEATTR.PROGCTL dans la classe FACILITY de votre logiciel de sécurité ou être un superutilisateur (UID 0) si ce profil n'est pas défini. Pour plus d'informations, voir *UNIX System Services Planning* (GA22-7800).

## Autorisation APF

L'Explorateur de systèmes distants (RSE) est le composant Developer for System z qui fournit des services de base, comme la connexion du client à l'hôte. Il doit être exécuté avec des droits APF pour effectuer des tâches comme l'affichage de l'usage détaillé des ressources du processus.

Le bit APF z/OS UNIX est défini au cours de l'installation SMP/E lorsque cela est nécessaire. Cette donnée de droits pourrait être perdue si vous ne la conservez pas lors de la copie manuelle des répertoires Developer for System z.

Les fichiers Developer for System z suivants doivent avoir des droits APF :

- /usr/lpp/rdz/bin/
  - BWBTSOW
  - CRASTART
  - fekfomvs
  - fekfriwp

Utilisez la commande z/OS UNIX, **ls -E**, pour lister les attributs d'extension, dans lesquels le bit APF est marqué avec la lettre a, comme indiqué dans l'exemple suivant (\$ est l'invite 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
```

Utilisez la commande z/OS UNIX, **extattr +a**, pour définir le bit APF manuellement, comme indiqué dans l'exemple suivant (\$ et # sont les invites 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
```

**Remarque :** Pour utiliser la commande **extattr +a**, vous devez disposer au moins d'un accès READ au profil BPX.FILEATTR.APF dans la classe FACILITY de votre

logiciel de sécurité ou être un superutilisateur (UID 0) si ce profil n'est pas défini. Pour plus d'informations, voir *UNIX System Services Planning* (GA22-7800).

## Données de rappel

Certains services Developer for System z facultatifs requièrent que les modules de chargement MVS soient disponibles pour z/OS UNIX. Pour ce faire, créez un module de remplacement (fichier factice) dans z/OS UNIX avec les données de rappel activées. Lorsque le module de remplacement est exécuté, z/OS UNIX recherche un module chargeable avec le même nom et exécute ce module chargeable MVS à la place.

Les données de rappel de z/OS UNIX sont définies pendant l'installation SMP/E, si nécessaire. Ces bits d'autorisation peuvent être perdus si vous ne les conservez pas lors de la copie manuelle des répertoires Developer for System z.

Les données de rappel des fichiers Developer for System z suivants doivent se trouver sous :

- /usr/lpp/rdz/bin/
  - AZUTSTRN
  - BWBTSOW
  - BWBTRANT
  - CRASTART

Utilisez la commande **ls -l** de z/OS UNIX pour afficher les droits. Les données de droit sont marquées par la lettre avec **t**, comme indiqué dans l'exemple ci-après (\$ représente l'invite 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
```

Utilisez la commande **chmod +t** de z/OS UNIX pour définir les données de rappel manuellement, comme indiqué dans l'exemple ci-après (\$ et # représentent l'invite 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
```

**Remarque :** Afin de pouvoir utiliser la commande **chmod**, vous devez disposer au minimum des droits d'accès READ au profil SUPERUSER.FILES.CHANGEPERMS dans la classe UNIXPRIV de votre logiciel de sécurité ou être un superutilisateur (UID 0) si ce profil n'est pas défini. Pour plus d'informations, voir *UNIX System Services Planning* (GA22-7800).

---

## Ports TCP/IP réservés

A l'aide de la commande **netstat** (TSO ou z/OS UNIX), vous pouvez connaître les ports actuellement utilisés. Le résultat de cette commande s'apparente à l'exemple ci-après. Les ports utilisés sont le dernier chiffre (après les ..) dans la colonne "Local Socket". Ces ports étant déjà utilisés, vous ne pouvez pas vous en servir pour la configuration de Developer for System z.

IPv4

|                            |          |                |  |                   |          |
|----------------------------|----------|----------------|--|-------------------|----------|
| MVS TCP/IP NETSTAT CS VxRy |          |                |  | TCPIP Name: TCPIP | 16:36:42 |
| User Id                    | Conn     | Local Socket   |  | Foreign Socket    | State    |
| -----                      | ----     | -----          |  | -----             | -----    |
| BPX0INIT                   | 00000014 | 0.0.0.0..10007 |  | 0.0.0.0..0        | Listen   |
| INETD4                     | 0000004D | 0.0.0.0..512   |  | 0.0.0.0..0        | Listen   |
| RSED                       | 0000004B | 0.0.0.0..4035  |  | 0.0.0.0..0        | Listen   |
| JMON                       | 00000038 | 0.0.0.0..6715  |  | 0.0.0.0..0        | Listen   |

#### IPv6

|                            |                 |                |  |                   |          |
|----------------------------|-----------------|----------------|--|-------------------|----------|
| MVS TCP/IP NETSTAT CS VxRy |                 |                |  | TCPIP Name: TCPIP | 12:46:25 |
| User Id                    | Conn            | State          |  |                   |          |
| -----                      | ----            | -----          |  |                   |          |
| BPX0INIT                   | 00000018        | Listen         |  |                   |          |
|                            | Local Socket:   | 0.0.0.0..10007 |  |                   |          |
|                            | Foreign Socket: | 0.0.0.0..0     |  |                   |          |
| INETD4                     | 00000046        | Listen         |  |                   |          |
|                            | Local Socket:   | 0.0.0.0..512   |  |                   |          |
|                            | Foreign Socket: | 0.0.0.0..0     |  |                   |          |
| RSED                       | 0000004B        | Listen         |  |                   |          |
|                            | Local Socket:   | 0.0.0.0..4035  |  |                   |          |
|                            | Foreign Socket: | 0.0.0.0..0     |  |                   |          |
| JMON                       | 00000037        | Listen         |  |                   |          |
|                            | Local Socket:   | 0.0.0.0..6715  |  |                   |          |
|                            | Foreign Socket: | 0.0.0.0..0     |  |                   |          |

Les ports TCP/IP réservés peuvent présenter une autre limitation. Il existe deux espaces communs pour réserver des ports TCP/IP :

- **PROFILE.TCPIP**

Il s'agit du fichier auquel se rapporte l'instruction de définition de données PROFILE de la tâche lancée par TCP/IP, souvent appelée SYS1.TCPPARMS(TCPPROF).

- PORT : réserve un port pour des noms de travaux spécifiés.
- PORTRANGE : réserve une plage de ports pour des noms de travaux spécifiés.

Pour plus d'informations sur ces instructions, voir *Communications Server: IP Configuration Guide* (SC31-8775).

- **SYS1.PARMLIB(BPXPRMxx)**

- INADDRANYPORT : indique le numéro de port de démarrage pour la série de numéros de port que le système réserve pour être utilisés avec les liaisons PORT 0, INADDR\_ANY. Cette valeur est uniquement nécessaire pour CINET (plusieurs piles TCP/IP actives sur un seul hôte).
- INADDRANYCOUNT : indique le nombre de ports que le système réserve, en commençant par le numéro de port spécifié dans le paramètre INADDRANYPORT. Cette valeur est uniquement nécessaire pour CINET (plusieurs piles TCP/IP actives sur un seul hôte).

Pour plus d'informations sur ces instructions, consultez les documents *UNIX System Services Planning* (GA22-7800) et *MVS Initialization and Tuning Reference* (SA22-7592).

Les ports réservés peuvent être répertoriés à l'aide de la commande **netstat portl** (TSO ou z/OS UNIX), qui crée une sortie comparable à celle de l'exemple ci-dessous :

|                            |      |          |       |                   |            |
|----------------------------|------|----------|-------|-------------------|------------|
| MVS TCP/IP NETSTAT CS VxRy |      |          |       | TCPIP Name: TCPIP | 17:08:32   |
| Port#                      | Prot | User     | Flags | Range             | IP Address |
| -----                      | ---- | -----    | ----- | -----             | -----      |
| 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 |

Pour plus d'informations sur la commande **NETSTAT**, voir le document *Communications Server: IP System Administrator's Commands* (SC31-8781).

**Remarque :** La commande **NETSTAT** présente uniquement les informations définies dans PROFILE.TCPIP, qui doivent coïncider avec les définitions BPXPRMxx. En cas de doute ou d'incident, vérifiez dans le membre parmlib BPXPRMxx les ports réservés dans ce cas.

---

## Taille d'espace adresse

Le démon RSE, qui est un processus z/OS UNIX Java, requiert une taille de région élevée pour exécuter ses fonctions. Il est donc important de définir des limites de mémoire importantes pour les espaces adresse OMVS.

## Exigences liées au langage JCL lors du démarrage

Le démon RSE est démarré par le JCL à l'aide de BPXBATSL, dont la taille de la région doit être 0.

## Limitations définies dans SYS1.PARMLIB(BPXPRMxx)

Attribuez la valeur 2G à MAXASSIZE dans SYS1.PARMLIB(BPXPRMxx) pour définir la taille de la région de l'espace adresse (processus) OMVS par défaut. Il s'agit de la valeur maximale autorisée. Il s'agit d'une limite à l'échelle du système. Elle est donc active pour tous les espaces adresse z/OS UNIX. Si elle ne répond pas à vos attentes, vous pouvez la définir uniquement pour Developer for System z dans votre logiciel de sécurité.

Cette valeur peut être vérifiée et définie de manière dynamique (jusqu'au prochain démarrage du système) à l'aide des commandes de console suivantes, décrites dans le document *MVS System Commands* (GC28-1781):

1. DISPLAY OMVS,0
2. SETOMVS MAXASSIZE=2G

## Limitations stockées dans le profil de sécurité

Vérifiez ASSIZEMAX dans le segment OMVS de l'ID utilisateur du démon et définissez-le à 2147483647 ou, de préférence, à NONE pour utiliser la valeur SYS1.PARMLIB(BPXPRMxx).

Avec RACF, cette valeur peut être vérifiée et définie à l'aide des commandes TSO suivantes, décrites dans le document *Security Server RACF Command Language Reference* (SA22-7687) :

1. LISTUSER userid NORACF OMVS
2. ALTUSER userid OMVS(NOASSIZEMAX)



## Limitations forcées par les sorties du système

Assurez-vous que vous n'autorisez pas aux sorties du système IEFUSI ou IEALIMIT de contrôler les tailles de région d'adresse OMVS. Il est possible de réaliser cela en codant SUBSYS(OMVS,NOEXITS) dans SYS1.PARMLIB(SMFPRMxx).

Les valeurs de SYS1.PARMLIB(SMFPRMxx) peuvent être vérifiées et activées à l'aide des commandes de console suivantes, comme indiqué dans la documentation *MVS System Commands* (GC28-1781) :

1. DISPLAY SMF,0
2. SET SMF=xx

## Limitations pour adressage 64 bits

Le mot clé MEMLIMIT dans SYS1.PARMLIB(SMFPRMxx) limite la quantité de stockage virtuel qu'une tâche 64 bits peut allouer au-delà de la barre des 2 Go. Contrairement au paramètre REGION du JCL, MEMLIMIT=0M signifie qu'il n'est pas possible d'utiliser un stockage virtuel au-delà de la barre.

Si MEMLIMIT n'est pas spécifié dans SMFPRMxx, la valeur par défaut est 0M, et donc des tâches sont liées aux 2 Go (31 bits) situés en dessous de la barre. La valeur par défaut est remplacée dans z/OS 1.10 par 2G, ce qui permet aux tâches 64 bits d'utiliser jusqu'à 4 GB (les 2 Go en dessous de la barre et les 2 Go au-dessus de la barre, accordés par MEMLIMIT).

Les valeurs de SYS1.PARMLIB(SMFPRMxx) peuvent être vérifiées et activées à l'aide des commandes de console suivantes, comme indiqué dans la documentation *MVS System Commands* (GC28-1781) :

1. DISPLAY SMF,0
2. SET SMF=xx

Vous pouvez aussi spécifier MEMLIMIT comme paramètre sur une carte EXEC dans un JCL. Si aucun paramètre MEMLIMIT n'est spécifié, la valeur par défaut est la valeur définie pour SMF, en revanche, si REGION=0M est spécifié, la valeur par défaut est NOLIMIT.

---

## Informations diverses

### Fin anormale pour manque d'espace B37 lors du retour d'informations

Lorsqu'un utilisateur sélectionne un retour d'informations sur les erreurs pendant une action de compilation, plusieurs fichiers temporaires sont créés par Developer for System z. Si l'espace est insuffisant pour l'un de ces fichiers, la tâche de compilation s'interrompt avec une erreur de type fin anormale pour manque d'espace B37-04.

Attribuez suffisamment d'espace dans FEK.SFEKPROC(FEKFERRF) lorsque les utilisateurs rencontrent ce problème. La valeur par défaut est SPACE(200,40) TRACKS.

### Limites du système

SYS1.PARMLIB(BPXPRMxx) définit plusieurs limitations liées à z/OS UNIX, qui peuvent être atteintes lorsque plusieurs clients de Developer for System z sont



actifs. La plupart des valeurs BPXPRMxx peuvent être modifiées de façon dynamique avec les commandes de la console **SETOMVS** et **SET OMVS**.

Utilisez la commande de console **SETOMVS LIMMSG=ALL** pour que z/OS UNIX affiche les messages de console (BPXI040I) lorsque l'une des limites BPXPRMxx est sur le point d'être atteinte.

## Connexion refusée

Chaque connexion RSE lance plusieurs processus qui sont actifs de façon permanente. De nouvelles connexions peuvent être refusées en raison de la limite définie dans SYS1.PARMLIB(BPXPRMxx) concernant la quantité de processus, particulièrement lorsque les utilisateurs partagent le même ID utilisateur (lorsque le segment OMVS par défaut est utilisé, par exemple).

- La limite par ID utilisateur est définie par le mot clé MAXPROCUSER, et a une valeur par défaut de 25.
- La limite au niveau du système est définie par le mot clé MAXPROCSYS, et a une valeur par défaut de 200.

La limite d'espaces adresse z/OS et d'utilisateurs z/OS UNIX actifs représente une autre source de connexions refusées.

- La quantité maximale d'ID espace adresse (ASID) est définie dans SYS1.PARMLIB(IEASYSxx) avec le mot-clé MAXUSER et présente une valeur par défaut de 255.
- La quantité maximale d'ID utilisateur (UID)z/OS UNIX est définie dans SYS1.PARMLIB(BPXPRMxx) avec le mot clé MAXUIDS, et présente une valeur par défaut de 200.

## Erreur liée à une insuffisance de mémoire

Un pool d'unités d'exécution RSE peut échouer et un message relatif à une erreur liée à une insuffisance de mémoire peut être consigné. Cette erreur est liée à la taille de segment de mémoire Java et peut se produire si les utilisateurs actifs dans ce pool d'unités d'exécution utilisent plus de ressources que prévu. Les causes courantes de cette erreur sont les suivantes :

- Développement de filtres de fichiers volumineux dans l'explorateur de système à distance
- Ouverture de PDS(E) avec un nombre important de membres
- Ouverture de membres ou de fichiers séquentiels volumineux

Pour résoudre ce problème, procédez comme suit :

- Augmentez la directive -Xmx dans rsed.envvars, car c'est elle qui contrôle la taille maximale de segment de mémoire Java. Notez que le segment de mémoire Java doit tenir dans les limites de l'espace adresse.
- Réduisez la directive -Dmaximum.clients dans rsed.envvars, car c'est elle qui contrôle le nombre d'utilisateurs pouvant être placés dans un seul pool d'unités d'exécution (et, par conséquent, qui partagent un seul segment de mémoire Java).

---

## Emulateur de connexion à l'hôte

- L'émulateur de connexion à l'hôte utilise telnet TN3270 et non le serveur RSE pour se connecter à l'hôte.
- Lorsque vous utilisez le Telnet sécurisé (SSL) et que vous travaillez avec des certificats qui ne sont pas signés par une autorité de certification bien connue,

chaque client doit ajouter le certificat de CA à la liste d'autorités de certification dignes de confiance de son émulateur de connexion à l'hôte.

- L'option NOSNAEXT de TELNETPARMS de TCP/IP peut être nécessaire pour désactiver les extensions fonctionnelles de l'architecture SNA. Dans le cas où NOSNAEXT est indiqué, le serveur telnet TN3270 ne négocie pas pour la résolution des conflits et les fonctions de détection d'architecture SNA.

---

## Chapitre 13. Configuration de l'authentification SSL et X.509

Cette annexe vous aide à résoudre certains incidents fréquents qui peuvent survenir lors de la configuration du protocole SSL (Secure Socket Layer), ou au cours de la vérification ou de la modification d'une configuration existante. Elle contient un exemple de configuration pour prendre en charge les utilisateurs qui s'authentifient à l'aide d'un certificat X.509.

Une communication sécurisée vous assure que votre partenaire de communication est bien celui qu'il prétend être, et que la transmission des informations se fait d'une manière qui rend difficile toute interception et lecture des données par des tiers. Le protocole SSL fournit cette capacité dans un réseau TCP/IP. Il fonctionne par l'emploi de certificats numériques pour vous identifier et d'un protocole à clé publique pour chiffrer la communication. Voir le document *Security Server RACF Security Administrator's Guide* (SA22-7683) pour de plus amples informations sur les certificats numériques et le protocole de clé publique utilisés par le protocole SSL.

Les actions nécessaires pour la configuration des communications SSL pour Developer for System z varient largement d'un site à l'autre, selon les véritables besoins, la méthode de communication RSE employée, et ce qui est déjà disponible au niveau du site.

Dans la présente annexe, nous allons cloner les définitions RSE en cours de manière à avoir une seconde connexion au démon RSE qui utilisera le protocole SSL. Nous créerons également nos propres certificats de sécurité destinés à être utilisés par les différents composants de la connexion RSE.

- «Utilisation de la méthode de chiffrement SSL ou TLS», à la page 192
- «Choix de l'emplacement de stockage des clés privées et des certificats», à la page 192
- «Création d'un fichier de clés avec RACF», à la page 193
- «Clonage de la configuration RSE existante», à la page 195
- «Mise à jour du fichier rsed.envvars pour assurer la coexistence», à la page 195
- «Mise à jour du fichier ssl.properties pour activer SSL», à la page 196
- «Activation de SSL en créant un démon RSE», à la page 196
- «Test de la connexion», à la page 197
- «(Facultatif) Ajout du support d'authentification du client via des certificats X.509», à la page 200
- «(Facultatif) Création d'une base de données de clés avec gskkyman», à la page 200
- «(Facultatif) Création d'un magasin de clés avec keytool», à la page 203

Une convention d'attribution de nom uniforme est utilisée dans cette annexe :

- Certificat : rdzrse
- Stockage de clés et de certificats : rdzssl.\*
- Mot de passe : rsessl
- ID utilisateur du démon : stcrse

Certaines des tâches décrites dans les sections suivantes nécessitent des actions de votre part dans z/OS UNIX. Vous pouvez les effectuer en lançant la commande TSO OMVS. Utilisez la commande **exit** pour retourner à TSO.

---

## Utilisation de la méthode de chiffrement SSL ou TLS

La variable `DSTORE_SSL_ALGORITHM` de la directive `_RSE_JAVAOPTS` du fichier `rsed.envvars` vous permet de choisir entre SSL et son successeur TLS (Transport Layer Security) pour la méthode de chiffrement, comme indiqué à la section sur la définition de paramètres de démarrage Java supplémentaires avec `_RSE_JAVAOPTS` dans le document *Guide de configuration de l'hôte* (SC23-7658).

---

## Choix de l'emplacement de stockage des clés privées et des certificats

Les certificats d'identité et les clés de chiffrement/déchiffrement utilisés par le protocole SSL sont stockés dans un fichier de clés. Différentes implémentations de ce fichier de clés existent, selon le type d'application.

Toutefois, toutes les implémentations suivent le même principe. Une commande génère une paire de clés (une clé publique et la clé privée associée). La commande intègre ensuite la clé publique à un certificat X.509 autosigné, qui est stocké comme une chaîne de certificats à un seul élément. Cette chaîne de certificats et la clé privée sont stockées en tant qu'entrée (identifiée par un alias) dans un fichier de clés.

Le démon RSE est une application du système SSL, qui utilise un fichier de base de données de clés. Cette base de données de clés peut être un fichier physique créé par gskkyman ou un fichier de clés géré par votre logiciel de sécurité conforme à SAF (RACF, par exemple). Le serveur RSE (démarré par le démon) est une application SSL Java qui utilise un magasin de clés créé par keytool ou un fichier de clés géré par votre logiciel de sécurité.

Tableau 39. Mécanismes de stockage des certificats SSL

| Stockage des certificats | Créé et géré par                        | Démon RSE      | Serveur RSE    |
|--------------------------|-----------------------------------------|----------------|----------------|
| Fichier de clés          | Produit de sécurité compatible avec SAF | pris en charge | pris en charge |
| Base de données de clés  | gskkyman de z/OS UNIX                   | pris en charge | /              |
| Magasin de clés          | Outil de clé de Java                    | /              | pris en charge |

Pour établir la connexion via SSL, le magasin de clés et le fichier de la base de données de clés sont nécessaires sous la forme d'un fichier z/OS UNIX ou d'un jeu de clés conforme à SAF :

- Magasin de clés (RACF ou outil de clé)
- base de données de clés (RACF ou gskkyman)

### Remarque :

- Il est conseillé d'utiliser des fichiers de clés conformes à SAF pour la gestion des certificats.
- Un certificat partagé peut être utilisé si le démon et le serveur RSE utilisent la même méthode de gestion des certificats.

- L'exécution du démon RSE doit être contrôlé par programme. L'utilisation de System SSL implique que SYS1.SIEALNKE soit contrôlé par programme via le logiciel de sécurité.
- Pour exécuter une application du système SSL (connexion par démon), SYS1.SIEALNKE doit être dans LINKLIST ou dans STEPLIB. Si vous préférez la méthode STEPLIB, ajoutez l'instruction suivante à la fin de rsed.envvars.  
STEPLIB=\$STEPLIB:SYS1.SIEALNKE  
Gardez toutefois les remarques suivantes à l'esprit :
  - L'utilisation de STEPLIB dans z/OS UNIX a un impact négatif sur les performances.
  - Si une bibliothèque STEPLIB est autorisée par APF, il doit en être de même pour toutes les bibliothèques. Les bibliothèques perdent leur autorisation APF lorsqu'elles sont mélangées avec des bibliothèques non autorisées dans STEPLIB.
- Le système SSL utilise ICFS (Integrated Cryptographic Service Facility) si celui-ci est disponible. ICFS met à disposition un support de chiffrement matériel qui est utilisé à la place des algorithmes logiciels du système SSL. Pour plus d'informations, voir *System SSL Programming* (SC24-5901).

Pour obtenir des informations sur RACF et les certificats numériques, voir le document *Security Server RACF Security Administrator's Guide* (SA22-7683). La documentation relative à gskkyman est disponible dans le document *System SSL Programming* (SC24-5901) et la documentation de keytool se trouve à l'adresse <http://java.sun.com/j2se/1.5.0/docs/tooldocs/solaris/keytool.html>.

---

## Création d'un fichier de clés avec RACF

N'exécutez pas cette étape si vous utilisez gskkyman pour créer la base de données de clés du démon RSE et keytool pour créer le magasin de clés du serveur RSE.

La commande **RACDCERT** installe et maintient les clés privées et les certificats dans RACF. RACF prend en charge la gestion en groupe de multiples clés privées et certificats. Ces groupes sont des fichiers de clés.

Pour obtenir des informations détaillées sur la commande **RACDCERT**, voir le document *Security Server RACF Command Language Reference* (SA22-7687).

```
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
```

```
RACDCERT ID(stcrse) GENCERT SUBJECTSDN(CN('rdz rse ssl') +
OU('rdz') O('IBM') L('Raleigh') SP('NC') C('US')) +
NOTAFTER(DATE(2017-05-21)) WITHLABEL('rdzrse') KEYUSAGE(HANDSHAKE)
```

```
RACDCERT ID(stcrse) ADDRING(rdzssl.racf)
RACDCERT ID(stcrse) CONNECT(LABEL('rdzrse') RING(rdzssl.racf) +
DEFAULT USAGE(PERSONAL))
```

L'exemple précédent commence par la création des profils nécessaires et par l'autorisation d'accès de l'ID utilisateur STCRSE aux jeux de clés et aux certificats détenus par cet ID utilisateur. L'ID utilisateur utilisé doit correspondre à celui employé pour exécuter le démon RSE SSL. L'étape suivante crée un certificat auto-signé avec l'intitulé rdzrse. Aucun mot de passe n'est nécessaire. Ce certificat

est alors ajouté au fichier de clés nouvellement créé (rdzssl.racf). Exactement comme avec le certificat, aucun mot de passe n'est nécessaire pour le fichier de clés.

Le résultat peut être vérifié par l'intermédiaire de l'option list :

```
RACDCERT ID(stcrse) LIST
Digital certificate information for user STCRSE:

Label: rdzrse
Certificate ID: 2QjW10Xi0sXZ1aaEqZmihUBA
Status: TRUST
Start Date: 2007/05/24 00:00:00
End Date: 2017/05/21 23:59:59
Serial Number:
 >00<
Issuer's Name:
 >CN=rdz rse ssl.OU=rdz.O=IBM.L=Raleigh.SP=NC.C=US<
Subject's Name:
 >CN=rdz rse ssl.OU=rdz.O=IBM.L=Raleigh.SP=NC.C=US<
Private Key Type: Non-ICSF
Private Key Size: 1024
Ring Associations:
 Ring Owner: STCRSE
 Ring:
 >rdzssl.racf<
```

## (Facultatif) Utilisation d'un certificat signé

Les certificats peuvent être autosignés ou signés par une autorité de certification (CA). Un certificat signé par une autorité de certification signifie que l'autorité de certification garantit que le propriétaire du certificat est bien la personne qu'il prétend être. La procédure de signature ajoute les données d'identification (certificat) de l'autorité de certification à votre certificat pour former une chaîne de certificats à plusieurs éléments.

Lorsque vous utilisez un certificat signé par une autorité de certification, vous pouvez éviter les questions de validation de la relation de confiance du client Developer for System z, si le client fait déjà confiance à l'autorité de certification.

Suivez les étapes suivantes pour créer et utiliser un certificat signé par une autorité de certification :

1. Créez un certificat autosigné.  
RACDCERT ID(stcrse) GENCERT WITHLABEL('rdzrse') . . .
2. Créez une demande de signature pour ce certificat.  
RACDCERT ID(stcrse) GENREQ (LABEL('rdzrse')) DSN(dsn)
3. Envoyez la demande de signature à l'autorité de certification de votre choix.
4. Vérifiez si les données d'identification (certificat) de l'autorité de certification sont déjà connues.  
RACDCERT CERTAUTH LIST
5. Marquez l'autorité de certification comme digne de confiance.  
RACDCERT CERTAUTH ALTER(LABEL('CA cert')) TRUST  
Vous pouvez ajouter le certificat de l'autorité de certification de la base de données.  
RACDCERT CERTAUTH ADD(dsn) WITHLABEL('CA cert') TRUST
6. Ajoutez le certificat signé à la base de données. Il doit remplacer le certificat autosigné.  
RACDCERT ID(stcrse) ADD(dsn) WITHLABEL('rdzrse') TRUST

**Remarque :** Ne supprimez PAS le certificat autosigné avant de le remplacer. Si tel est le cas, ne perdez pas la clé privée associée au certificat. Le certificat serait inutilisable.

7. Créez un jeu de clés.

```
RACDCERT ID(stcrse) ADDRING(rdzssl.racf)
```

8. Ajoutez le certificat signé au jeu de clés.

```
RACDCERT ID(stcrse) CONNECT(ID(stcrse) LABEL('rdzrse'))
RING(rdzssl.racf))
```

9. Ajouter le certificat de l'autorité de certification au jeu de clés.

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

Notez que le certificat de l'autorité de certification utilisé pour signer votre certificat peut, à son tour, également être signé par un autre certificat de l'autorité de certification de niveau supérieur. Si cela se produit, le certificat de l'autorité de certification de niveau supérieur doit également être ajouté au fichier de clés. Ce processus se répète jusqu'à ce que le certificat de l'autorité de certification de niveau supérieur soit un certificat de l'autorité de certification racine, lequel est toujours un certificat autosigné.

---

## Clonage de la configuration RSE existante

Dans cette étape, une nouvelle instance des fichiers de configuration RSE est créée, pour que la configuration SSL puisse être exécutée en parallèle avec celle(s) qui existe(nt) déjà. Les exemples de commandes suivants prévoient le stockage des fichiers de configuration dans le répertoire `/etc/rdz/`, ce qui est l'emplacement par défaut utilisé dans la section "Configuration personnalisée" du *Guide de configuration de l'hôte* (SC11-6285).

```
$ cd /etc/rdz
$ mkdir ssl
$ cp rsed.envvars ssl
$ cp ssl.properties ssl
$ ls ssl
rsed.envvars ssl.properties
```

Les commandes z/OS UNIX répertoriées ci-dessus permettent de créer un sous-répertoire appelé `ssl` et d'y placer les fichiers de configuration qui nécessitent des modifications. Les autres fichiers de configuration, le répertoire d'installation et les composants MVS sont partagés car ils ne sont pas propres à SSL.

La réutilisation de la plupart des fichiers de configuration permet de se consacrer aux modifications nécessaires à la configuration SSL et d'éviter de réaliser de nouveau la configuration RSE (par exemple, vous pouvez éviter de définir un nouvel emplacement pour `ISPF.conf`.)

---

## Mise à jour du fichier `rsed.envvars` pour assurer la coexistence

Jusqu'à présent, les définitions sont une copie exacte de la configuration en cours, ce qui signifie que les journaux du nouveau démon RSE remplacent les fichiers journaux du serveur en cours. RSE doit également savoir où se trouvent les fichiers de configuration qui n'ont pas été copiés dans le répertoire `ssl`. Ces deux problèmes peuvent être corrigés en apportant des modifications mineures au fichier `rsed.envvars`.

```
$ oedit /etc/rdz/ssl/rsed.envvars
-> change: _RSE_RSED_PORT=4047
-> change: -Ddaemon.Tog=/var/rdz/logs/ssl
```



```

-> change: -Duser.log=/var/rdz/logs/ssl
-> add at the END:
-- NEEDED TO FIND THE REMAINING CONFIGURATION FILES
CFG_BASE=/etc/rdz
CLASSPATH=.:$CFG_BASE:$CLASSPATH
--

```

Les modifications illustrées dans l'exemple précédent définissent un nouvel emplacement de journal (qui est créé par le démon RSE si l'emplacement de journal n'existe pas). Elles mettent également à jour la variable CLASSPATH afin que les processus RSE SSL recherchent les fichiers de configuration dans le répertoire en cours (/etc/rdz/ssl), puis dans le répertoire d'origine (/etc/rdz).

---

## Mise à jour du fichier `ssl.properties` pour activer SSL

Le fichier `ssl.properties` est mis à jour afin de demander à RSE de démarrer en utilisant une communication chiffrée SSL.

```

$ oedit /etc/rdz/ssl/ssl.properties
-> change: enable_ssl=true
-> uncomment and change: daemon_keydb_file=rdzssl.racf
-> uncomment and change: daemon_key_label=rdzrse
-> uncomment and change: server_keystore_file=rdzssl.racf
-> uncomment and change: server_keystore_label=rdzrse
-> uncomment and change: server_keystore_type=JCERACFKS

```

Les modifications indiquées précédemment activent SSL et indiquent au démon RSE et au serveur RSE que leur certificat (partagé) est stocké sous le nom `rdzrse` dans le fichier de clés `rdzssl.racf`. Le mot clé `JCERACFKS` indique au serveur RSE qu'un fichier de clés conforme à SAF est utilisé en tant que magasin de clés.

Notez que System SSL (utilisé par le démon) utilise toujours ICSF, l'interface avec le matériel de chiffrement de System z, si elle est disponible. Pour pouvoir partager les définitions de démon avec le serveur lors de l'utilisation d'ICSF, vous devez spécifier `server_keystore_type JCECCARACFKS`. Ici, un fichier de clés compatible avec SAF est également utilisé en tant que magasin de clés pour les clés publiques, mais la clé privée est stockée dans ICSF. Comme indiqué dans le document *Cryptographic Services ICSF Administrator's Guide* (SA22-7521), ICSF utilise des profils dans les classes de sécurité `CSFKEYS` et `CSFSERV` pour contrôler qui peut utiliser les clés et les services de chiffrement.

---

## Activation de SSL en créant un démon RSE

Comme indiqué précédemment, nous allons créer une deuxième connexion qui utilisera la couche SSL, ce qui implique la création d'un démon RSE. Le démon RSE peut être une tâche démarrée ou un travail utilisateur. Nous utiliserons la méthode du travail utilisateur pour la configuration initiale (test). Les instructions suivantes prévoient le stockage de l'exemple de langage JCL dans le répertoire `FEK.#CUST.PROCLIB(RSED)`, ce qui est l'emplacement par défaut utilisé dans la section "Configuration personnalisée" du document *Guide de configuration de l'hôte* (SC11-6285) :

1. Créez un membre `FEK.#CUST.PROCLIB(RSEDSSL)` et copiez dans ce dernier l'exemple de langage JCL `FEK.#CUST.PROCLIB(RSED)`.
2. Personnalisez `RSEDSSL` en ajoutant une carte de travail en haut et une instruction `exec` en bas. Indiquez également l'emplacement des fichiers de configuration SSL (/etc/rdz/ssl), comme illustré dans l'exemple de code suivant. Notez que nous appliquons l'ID utilisateur `STCRSE` car il a reçu des droits d'accès aux certificats et aux fichiers de clés à l'étape précédente.



```

//RSEDSSL JOB CLASS=A,MSGCLASS=A,MSGLEVEL=(1,1),USER=STCRSE
//*
/* RSE DAEMON - SSL
/*
//RSED PROC TMPDIR=,
// PORT=,
// IVP=, * 'IVP' to do an IVP test
// 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
/*

```

Figure 30. RSEDSSL - Travail de l'utilisateur du serveur RSE pour SSL

**Remarque :** L'ID utilisateur affecté au travail RSEDSSL dispose des mêmes droits que le démon RSE d'origine. Le profil de FACILITY BPX.SERVER et le profil de PTKTDATA IRRPTAUTH.FEKAPPL.\* sont des éléments clés ici.

---

## Test de la connexion

La configuration de l'hôte SSL est maintenant terminée et le démon RSE pour la couche SSL peut être démarré par la soumission du travail FEK.#CUST.PROCLIB(RSEDSSL) précédemment créé.

La nouvelle configuration peut maintenant être testée via la connexion au client System z. Dans la mesure où nous avons créé une configuration pour l'utilisation avec SSL (par clonage d'une configuration existante), une nouvelle connexion doit être définie sur le client avec l'utilisation du port 4047 pour le démon RSE.

A la connexion, l'hôte et le client démarrent avec un protocole d'établissement de liaison pour configurer un chemin d'accès sécurisé. L'échange de certificats fait partie de ce protocole d'établissement de liaison. Si le client Developer for System z ne reconnaît pas le certificat de l'hôte ou l'autorité de certification qui l'a signé, il demande à l'utilisateur si ce certificat est digne de confiance.

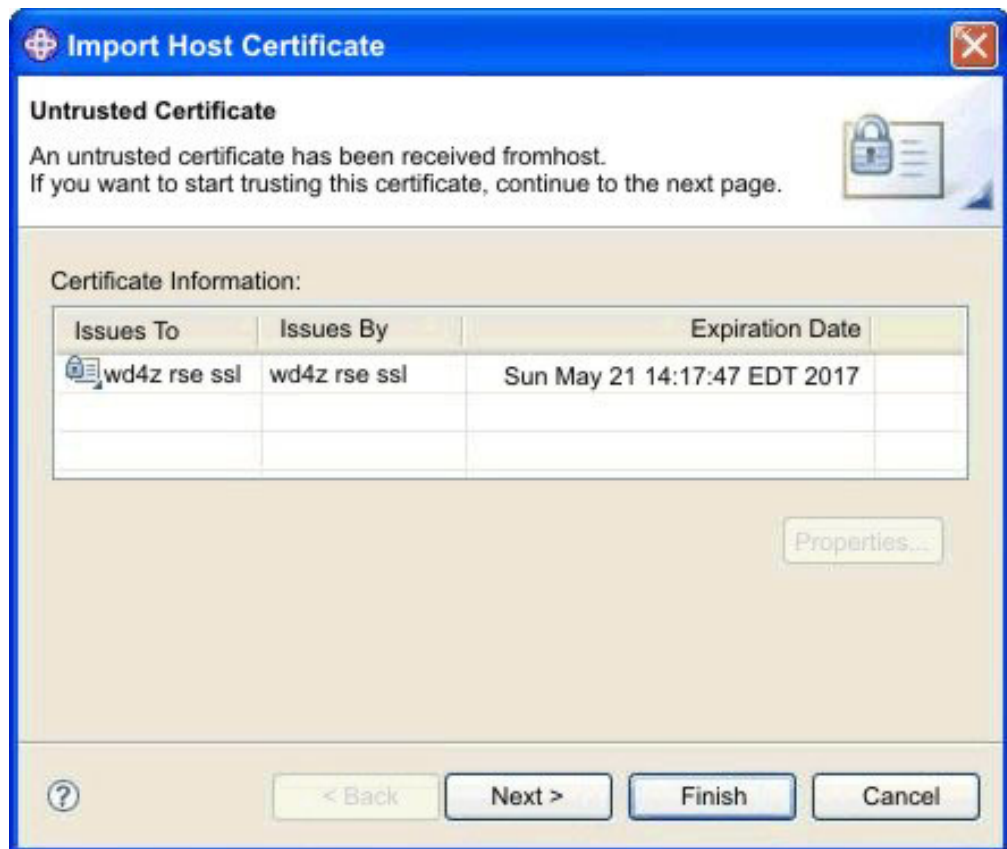


Figure 31. Boîte de dialogue Importation du certificat hôte

En cliquant sur le bouton Finish, l'utilisateur peut accepter le certificat comme étant sécurisé, après quoi l'initialisation de la connexion se poursuit.

**Remarque :** Le démon et le serveur RSE peuvent utiliser deux emplacements de certificat différents, ce qui donne deux certificats différents et par conséquent, deux confirmations.

Une fois que le client connaît le certificat, cette boîte de dialogue n'est plus affichée. La liste des certificats de confiance peut être gérée en sélectionnant **Fenêtre > Préférences... > Systèmes distants > SSL**, ce qui provoque l'affichage de la boîte de dialogue suivante :



du serveur et de l'utilisateur, comme indiqué à la section «Journalisation du démon RSE et du pool d'unités d'exécution», à la page 173 et «Journalisation pour l'utilisateur RSE», à la page 174.

---

## (Facultatif) Ajout du support d'authentification du client via des certificats X.509

Le démon RSE prend en charge les utilisateurs qui s'authentifient eux-mêmes à l'aide d'un certificat X.509. L'utilisation de communications chiffrées SSL est indispensable pour cette fonction car il s'agit d'une extension de l'authentification hôte avec un certificat utilisé dans SSL.

Il y a plusieurs méthodes pour effectuer l'authentification d'un utilisateur via un certificat, comme indiqué à la section «Authentification du client à l'aide de certificats X.509», à la page 31. Les étapes suivantes décrivent la configuration nécessaire pour que votre logiciel de sécurité authentifie le certificat à l'aide de l'extension de certificat HostIdMappings.

1. Remplacez le certificat qui identifie l'autorité de certification utilisée pour signer le certificat client par un certificat d'autorité de certification hautement sécurisée. Bien que l'état TRUST soit suffisant pour une validation de certificat, l'état HIGHTRUST est appliqué car il est utilisé pour l'authentification du certificat dans le cadre de la procédure de connexion.

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

2. Ajoutez le certificat de l'autorité de certification au fichier de clés, `rdzssl.racf` afin qu'il soit disponible pour valider les certificats client.

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

La configuration du certificat de l'autorité de certification est terminée.

3. Définissez une ressource (format `IRR.HOST.hostname`) dans la classe `SERVAUTH` du nom d'hôte, `CDFMVS08.RALEIGH.IBM.COM`, défini dans l'extension `HostIdMappings` du certificat client.

```
RDEFINE SERVAUTH IRR.HOST.CDFMVS08.RALEIGH.IBM.COM UACC(NONE)
```

4. Accordez à l'ID utilisateur de la tâche démarrée, `STCRSE`, l'accès à cette ressource avec les droits `READ`.

```
PERMIT IRR.HOST.CDFMVS08.RALEIGH.IBM.COM CLASS(SERVAUTH) +
ACCESS(READ) ID(stcrse)
```

5. Activez les modifications apportées à la classe `SERVAUTH`. Utilisez la première commande si la classe `SERVAUTH` n'est pas encore active. Utilisez la seconde pour régénérer une configuration active.

```
SETOPTS CLASSACT(SERVAUTH) RACLIST(SERVAUTH)
ou
SETOPTS RACLIST(SERVAUTH) REFRESH
```

La configuration du logiciel de sécurité pour l'extension `HostMappingscat` de l'autorité de certification est terminée.

6. Redémarrez la tâche démarrée RSE pour commencer à accepter des connexions client à l'aide de certificats X.509.

---

## (Facultatif) Création d'une base de données de clés avec gskkyman

N'exécutez pas cette étape si vous utilisez un fichier de clés conforme à SAF pour la base de données de clés du démon RSE.

gskkyman est un programme z/OS UNIX basé sur le shell, piloté par menus, qui crée, remplit et gère un fichier z/OS UNIX qui contient les clés privées, les demandes de certificats et les certificats. Ce fichier z/OS UNIX est une base de données de clés.

**Remarque :** Les instructions suivantes peuvent être nécessaires pour configurer l'environnement pour gskkyman. Pour plus d'informations à ce sujet, voir *System SSL Programming* (SC24-5901).

```
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 Menu de la base de données
```

```
1 - Create new database
```

```
Enter option number: 1
```

```
Enter key database name (press ENTER to return to menu): rdzssl.kdb
```

```
Enter database password (press ENTER to return to menu): rsessl
```

```
Re-enter database password: rsessl
```

```
Enter password expiration in days (press ENTER for no expiration):
```

```
Enter database record length (press ENTER to use 2500):
```

```
Key database /etc/rdz/ssl/rdzssl.kdb created.
```

```
Press ENTER to continue.
```

```
Key Management Menu
```

```
6 - Create a self-signed certificate
```

```
Enter option number (press ENTER to return to previous menu): 6
```

```
Certificate Type
```

```
5 - User or server certificate with 1024-bit RSA key
```

```
Select certificate type (press ENTER to return to menu): 5
```

```
Enter label (press ENTER to return to menu): rdzrse
```

```
Enter subject name for certificate
```

```
Common name (required): rdz rse ssl
```

```
Organizational unit (optional): rdz
```

```
Organization (required): IBM
```

```
City/Locality (optional): Raleigh
```

```
State/Province (optional): NC
```

```
Country/Region (2 characters - required): US
```

```
Enter number of days certificate will be valid (default 365): 3650
```

```
Enter 1 to specify subject alternate names or 0 to continue: 0
```

```
Please wait
```

```
Certificate created.
```

```
Press ENTER to continue.
```

```
Key Management Menu
```

```
0 - Exit program
```

```
Enter option number (press ENTER to return to previous menu): 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
```

L'exemple précédent commence par la création d'une base de données de clés appelée `rdzssl.kdb` avec le mot de passe `rsessl`. Lorsque la base de données existe, elle est enrichie en créant un certificat autosigné valide pendant 10 ans environ (sans compter les jours des années bissextiles). Le certificat est conservé sous le nom `rdzrse` et avec même le mot de passe (`rsessl`) que celui utilisé pour la base de données de clés (il s'agit d'un élément prérequis par RSE).

`gskkyman` attribue un masque de bits d'autorisation (très sûr, accès du seul propriétaire) de 600 à la base de données de clés. Mis à part le cas où le démon utilise le même ID utilisateur que le créateur de la base de données de clés, les autorisations doivent être définies d'une manière moins restrictive. Le masque 644 (le propriétaire a des droits d'accès en lecture/écriture, tout le monde a des droits d'accès en lecture) est utilisable pour la commande **`chmod`**.

Ce résultat peut être vérifié en sélectionnant l'option **Show certificate information** dans le sous-menu **Manage keys and certificates** :

```
$ gskkyman
```

```
Database Menu
```

```
2 - Open database
```

```
Enter option number: 2
```

```
Enter key database name (press ENTER to return to menu): rdzssl.kdb
```

```
Enter database password (press ENTER to return to menu): rsessl
```

```
Key Management Menu
```

```
1 - Manage keys and certificates
```

```
Enter option number (press ENTER to return to previous menu): 1
```

```
Key and Certificate List
```

```
1 - rdzrse
```

```
Enter label number (ENTER to return to selection menu, p for previous list): 1
```

```
Key and Certificate Menu
```

```
1 - Show certificate information
```

```
Enter option number (press ENTER to return to previous menu): 1
```

```
Certificate Information
```

```
Label: rdzrse
Record ID: 14
Issuer Record ID: 14
Trusted: Yes
Version: 3
Serial number: 45356379000ac997
Issuer name: rdz rse ssl
rdz
IBM
Raleigh
NC
US
```

```

 Subject name: rdz rse ssl
 rdz
 IBM
 Raleigh
 NC
 US
 Effective date: 2007/05/24
 Expiration date: 2017/05/21
 Public key algorithm: rsaEncryption
 Public key size: 1024
 Signature algorithm: sha1WithRsaEncryption
 Issuer unique ID: None
 Subject unique ID: None
 Number of extensions: 3

```

Enter 1 to display extensions, 0 to return to menu: 0

Key and Certificate Menu

0 - Exit program

Enter option number (press ENTER to return to previous menu): 0

L'exemple de fichier `ssl.properties` ci-après indique que les directives `daemon_*` diffèrent de celles de l'exemple de fichier de clés SAF présenté plus haut.

```

$ oedit /etc/rdz/ssl/ssl.properties
-> change: enable_ssl=true
-> uncomment and change: daemon_keydb_file=rdzssl.kdb
-> uncomment and change: daemon_keydb_password=rsessl
-> uncomment and change: daemon_key_label=rdzrse
-> uncomment and change: server_keystore_file=rdzssl.racf
-> uncomment and change: server_keystore_label=rdzrse
-> uncomment and change: server_keystore_type=JCERACFKS

```

Les modifications précédentes activent SSL et indiquent au démon RSE que le certificat est stocké sous le nom `rdzrse` dans la base de données de clés `rdzssl.kdb` avec le mot de passersessl. Le serveur RSE continue à utiliser un fichier de clés conforme à SAF.

---

## (Facultatif) Création d'un magasin de clés avec keytool

N'exécutez pas cette étape si vous utilisez un fichier de clés conforme à SAF pour le magasin de clés du serveur RSE.

"`keytool -genkey`" génère une paire de clés privées et un certificat autosigné associé, qui est stocké sous la forme d'une entrée (identifiée par un alias) dans un (nouveau) magasin de clés.

**Remarque :** Java doit être inclus dans vos répertoires de recherches de commandes. L'instruction suivante peut être nécessaire pour exécuter `keytool`, où `/usr/lpp/java/J5.0` est le répertoire d'installation de Java : `PATH=$PATH:/usr/lpp/java/J5.0/bin`

Toutes les informations peuvent être transmises comme paramètres, mais en raison des limitations de longueur de la ligne de commande, une certaine interactivité est nécessaire :

```

$ cd /etc/rdz/ssl
$ keytool -genkey -alias rdzrse -validity 3650 -keystore rdzssl.jks -storepass
rsessl -keypass rsessl
What is your first and last name?
[Unknown]: rdz rse ssl

```

```

What is the name of your organizational unit?
[Unknown]: rdz
What is the name of your organization?
[Unknown]: IBM
What is the name of your City or Locality?
[Unknown]: Raleigh
What is the name of your State or Province?
[Unknown]: NC
What is the two-letter country code for this unit?
[Unknown]: US
Is CN=rdz rse ssl, OU=rdz, O=IBM, L=Raleigh, ST=NC, C=US correct? (type "yes"
or "no")
[no]: yes
$ ls -l rdzssl.*
-rw-r--r-- 1 IBMUSER SYS1 1224 May 24 14:17 rdzssl.jks

```

Le certificat autosigné créé dans l'exemple précédent est valide pendant environ 10 ans (sans compter les jours des années bissextiles). Il est stocké dans /etc/rdz/ssl/rdzssl.jks avec l'alias rdzrse. Son mot de passe (rsessl) est identique au mot de passe du fichier de clés, ce qui est un élément prérequis pour RSE.

Le résultat peut être vérifié par l'intermédiaire de l'option -list :

```

$ keytool -list -alias rdzrse -keystore rdzssl.jks -storepass rsessl -v
Alias name: rdzrse
Creation date: May 24, 2007
Entry type: keyEntry
Certificate chain length: 1
Certificate 1:
Owner: CN=rdz rse ssl, OU=rdz, O=IBM, L=Raleigh, ST=NC, C=US
Issuer: CN=rdz rse ssl, OU=rdz, O=IBM, L=Raleigh, ST=NC, C=US
Serial number: 46562b2b
Valid from: 5/24/07 2:17 PM until: 5/21/17 2:17 PM
Certificate fingerprints:
 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

```

Dans l'exemple de fichier ssl.properties ci-après, les directives server\_\* diffèrent de celles de l'exemple de fichier de clés SAF présenté plus haut.

```

$ oedit /etc/rdz/ssl/ssl.properties
-> change: enable_ssl=true
-> uncomment and change: daemon_keydb_file=rdzssl.racf
-> uncomment and change: daemon_key_label=rdzrse
-> uncomment and change: server_keystore_file=rdzssl.jks
-> uncomment and change: server_keystore_password=rsessl
-> uncomment and change: server_keystore_label=rdzrse
-> optionally uncomment and change: server_keystore_type=JKS

```

Les modifications précédentes activent SSL et indiquent au serveur RSE que le certificat est stocké sous le nom rdzrse dans le magasin de clés rdzssl.jks avec le mot de passe rsessl. Le démon RSE utilise toujours un fichier de clés conforme à SAF.



---

## Chapitre 14. Configuration de TCP/IP

Cette annexe vous aide à résoudre certains problèmes fréquents qui peuvent survenir lors de la configuration du protocole TCP/IP, ou lors de la vérification ou de la modification d'une configuration existante.

Pour plus d'informations sur la configuration TCP/IP, voir *Communications Server: IP Configuration Guide* (SC31-8775) et *Communications Server: IP Configuration Reference* (SC31-8776).

---

### Dépendance au nom d'hôte

Lorsque vous utilisez APPC pour le service Commandes TSO, Developer for System z dépend de la validité du nom d'hôte du protocole TCP/IP quand il est initialisé. Cela implique que les différents fichiers de configuration TCP/IP et du programme de résolution soient configurés correctement.

Vous pouvez tester votre configuration TCP/IP à l'aide du programme de vérification d'installation fekfivpt. La commande doit renvoyer un résultat comparable à celui de cet exemple (\$ correspond à l'invite z/OS UNIX) :

```
$ fekfivpt
```

```
Wed Jul 2 13:11:54 EDT 2008
uid=1(USERID) gid=0(GROUP)
using /etc/rdz/rsed.envvars
```

```

TCP/IP resolver configuration (z/OS UNIX search order):

```

```
Resolver Trace Initialization Complete -> 2008/07/02 13:11:54.745964
```

```
res_init Resolver values:
```

```
Global Tcp/Ip Dataset = None
Default Tcp/Ip Dataset = None
Local Tcp/Ip Dataset = /etc/resolv.conf
Translation Table = Default
UserId/JobName = USERID
Caller API = LE C Sockets
Caller Mode = 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 Succeeded
```

```
res_init Started: 2008/07/02 13:11:54.755363
```

```
res_init Ended: 2008/07/02 13:11:54.755371
```

```

```

```
MVS TCP/IP NETSTAT CS V1R9 TCPIP Name: TCPIP 13:11:54
```

```
Tcpip started at 01:28:36 on 06/23/2008 with IPv6 enabled
```

```

```

host IP address:

```

hostName=CDFMVS08
hostAddr=9.42.112.75
bindAddr=9.42.112.75
localAddr=9.42.112.75
```

Success, addresses match

---

## Présentation des programmes de résolution

Le programme de résolution joue le rôle des programmes comme un client qui accède aux serveurs de noms pour une résolution nom/adresse ou adresse/nom. Pour résoudre la requête du programme demandeur, le programme de résolution peut accéder aux serveurs de noms disponibles, utiliser des définitions locales (/etc/resolv.conf, /etc/hosts, /etc/ipnodes, HOSTS.SITEINFO, HOSTS.ADDRINFO, ou ETC.IPNODES, par exemple) ou utiliser une combinaison des deux.

Quand l'espace adresse du programme de résolution démarre, il lit un fichier de configuration du programme de résolution facultatif indiqué par la carte SETUP DD dans la procédure JCL du programme de résolution. Si les informations de configuration ne sont pas fournies, le programme de résolution utilise l'ordre de recherche MVS ou z/OS UNIX natif applicable, sans aucune information GLOBALTCPIPDATA, DEFAULTTCPIPDATA, GLOBALIPNODES, DEFAULTIPNODES ou COMMONSEARCH.

---

## Présentation des ordres de recherche d'informations de configuration

Il est important de connaître les fonctions de l'ordre de recherche des fichiers de configuration utilisés par TCP/IP, et de savoir quand vous pouvez remplacer l'ordre de recherche par défaut par des variables d'environnement, JCL ou autre variable fournie. Vous pouvez ainsi adapter votre fichier de données locales et les normes de dénomination d'un fichier d'un système hiérarchique ; il est également utile de savoir s'il s'agit du fichier de données de configuration ou du fichier du système hiérarchique qui est utilisé au moment d'identifier les incidents.

Autre point important, quand un ordre de recherche est appliqué à n'importe quel fichier de configuration, la recherche s'arrête au premier fichier trouvé. Par conséquent, vous risquez de trouver des résultats inattendus si vous enregistrez des informations de configuration dans un fichier qui n'est jamais trouvé, soit parce que d'autres fichiers le précèdent dans l'ordre de recherche ou parce que le fichier n'est pas inclus dans l'ordre de recherche choisi par l'application.

Quand vous recherchez des fichiers de configuration, vous pouvez indiquer clairement au protocole TCP/IP l'emplacement de la plupart des fichiers de configuration en utilisant des instructions de définition de données dans les procédures JCL ou en définissant des variables d'environnement. Sinon, vous pouvez laisser le protocole TCP/IP déterminer de manière dynamique l'emplacement des fichiers de configuration, en fonction des ordres de recherche documentés dans le guide *Communications Server: IP Configuration Guide* (SC31-8775).

Le composant de configuration de la pile TCP/IP utilise TCPIP.DATA au cours de l'initialisation de la pile TCP/IP afin d'en déterminer le paramètre HOSTNAME. Pour obtenir sa valeur, l'ordre de recherche de l'environnement z/OS UNIX est utilisé.

**Remarque :** L'utilitaire du programme de résolution de trace permet de déterminer les valeurs TCPIP.DATA qui sont utilisées par le programme de résolution et leur emplacement au moment de la lecture. Pour plus d'informations sur le démarrage dynamique de la trace, voir le document *Communications Server: IP Diagnosis Guide* (GC31-8782). Une fois que la fonction de trace est active, exécutez une commande TSO **NETSTAT HOME** et une commande shell z/OS UNIX **netstat -h** pour afficher les valeurs. Une commande PING d'un nom d'hôte lancée via une commande TSO et l'interpréteur de commandes z/OS UNIX affiche également l'activité de tous les serveurs DNS qui peuvent être configurés.

---

## Ordres de recherche utilisés dans l'environnement z/OS UNIX

Le tableau ou fichier particulier recherché correspond à un fichier MVS ou à un fichier de système hiérarchique, en fonction des paramètres de configuration du programme de résolution et de la présence de ces fichiers sur le système.

### Fichiers de configuration du programme de résolution de base

Le fichier de configuration du programme de résolution de base contient des instructions TCPIP.DATA. Outre les directives du programme de résolution, il est référencé pour déterminer, entre autres, le préfixe du fichier (valeur de l'instruction DATASETPREFIX) à utiliser lors de la tentative d'accès à certains des fichiers de configuration spécifiés dans cette section.

L'ordre de recherche permettant d'accéder au fichier de configuration du programme de résolution de base est le suivant :

1. **GLOBALTCPIPDATA**

Si ce fichier est défini, la valeur de l'instruction de configuration GLOBALTCPIPDATA du programme de résolution est utilisée (voir aussi «Présentation des programmes de résolution», à la page 206). La recherche se poursuit pour trouver un autre fichier de configuration. La recherche s'arrête avec le fichier trouvé suivant.

2. La valeur de la variable d'environnement **RESOLVER\_CONFIG**

La valeur utilisée est celle de la variable d'environnement. La recherche échoue si le fichier n'existe pas ou s'il se trouve dans un autre emplacement.

3. **/etc/resolv.conf**

4. **//SYSTCPD DD card**

Le fichier utilisé est celui qui est attribué au nom DD SYSTCPD. Dans l'environnement z/OS UNIX, un processus enfant n'a pas accès à SYSTCPD DD. En effet, l'attribution SYSTCPD n'est pas héritée du processus père sur le processus parallèle de traitement() ou les appels de fonction de commande exec.

5. **userid.TCPIP.DATA**

userid désigne l'ID utilisateur qui est associé à l'environnement de sécurité en cours (espace adresse, tâche ou unité d'exécution).

6. **jobname.TCPIP.DATA**

jobname correspond au nom indiqué dans l'instruction JCL JOB pour les travaux par lots ou le nom de la procédure pour une procédure démarrée.

7. **SYS1.TCPPARMS(TCPDATA)**

8. **DEFAULTTCPIPDATA**

Si ce fichier est défini, la valeur de l'instruction de configuration DEFAULTTCPIPDATA du programme de résolution est utilisée (voir aussi «Présentation des programmes de résolution», à la page 206).

#### 9. TCPIP.TCPIP.DATA

### Tables de conversion

Les tables de conversion (EBCDIC en ASCII et ASCII en EBCDIC) permettent de déterminer les fichiers de conversion à utiliser. L'ordre de recherche permettant d'accéder à ce fichier de configuration est le suivant. L'ordre de recherche s'arrête au premier fichier trouvé :

1. La valeur de la variable d'environnement **X\_XLATE**. La valeur de la variable d'environnement correspond au nom de la table de conversion créée par la commande TSO CONVXLAT.
2. **userid.STANDARD.TCPXLBIN**  
userid désigne l'ID utilisateur qui est associé à l'environnement de sécurité en cours (espace adresse ou tâche/unité d'exécution).
3. **jobname.STANDARD.TCPXLBIN**  
jobname correspond au nom indiqué dans l'instruction JCL JOB pour les travaux par lots ou le nom de la procédure pour une procédure démarrée.
4. **hlq.STANDARD.TCPXLBIN**  
hlq représente la valeur de l'instruction DATASETPREFIX spécifiée dans le fichier de configuration du programme de résolution de base (le cas échéant) ; sinon, hlq correspond au protocole TCPIP par défaut.
5. Si aucune table n'est trouvée, le programme de résolution utilise une table codée en dur par défaut identique à la table figurant dans le membre de fichier SEZATCPX (STANDARD).

### Tables de système hôte local

Par défaut, le programme de résolution tente tout d'abord d'utiliser n'importe quel serveur de noms de domaine configuré pour les demandes de résolution. Si la demande de résolution n'est pas satisfaite, les tables de système hôte local sont utilisées. Le comportement du programme de résolution est contrôlé par les instructions TCPIP.DATA.

Les instructions du programme de résolution TCPIP.DATA définissent si et comment des serveurs de noms de domaine doivent être utilisés. L'instruction LOOKUP TCPIP.DATA permet également de contrôler l'utilisation des serveurs de noms de domaine et des tables de système hôte local. Pour de plus amples informations sur les instructions TCPIP.DATA, voir le document *Communications Server: IP Configuration Reference* (SC31-8776).

Le programme de résolution se sert de l'ordre de recherche unique Ipv4 pour trouver des informations de nom de site sans restrictions pour les appels API getnetbyname. L'ordre de recherche unique Ipv4 pour des informations de nom de site est le suivant. La recherche s'arrête au premier fichier trouvé :

1. La valeur de la variable d'environnement **X\_SITE**  
La valeur de la variable d'environnement correspond au nom du fichier d'informations HOSTS.SITEINFO créé par la commande TSO **MAKESITE**.
2. La valeur de la variable d'environnement **X\_ADDR**  
La valeur de la variable d'environnement correspond au nom du fichier d'informations HOSTS.ADDRINFO créé par la commande TSO **MAKESITE**.

3. **/etc/hosts**
4. **userid.HOSTS.SITEINFO**  
userid désigne l'ID utilisateur qui est associé à l'environnement de sécurité en cours (espace adresse ou tâche/unité d'exécution).
5. **jobname.HOSTS.SITEINFO**  
jobname correspond au nom indiqué dans l'instruction JCL JOB pour les travaux par lots ou le nom de la procédure pour une procédure démarrée.
6. **hlq.HOSTS.SITEINFO**  
hlq représente la valeur de l'instruction DATASETPREFIX spécifiée dans le fichier de configuration du programme de résolution de base (le cas échéant) ; sinon, hlq correspond au protocole TCPIP par défaut.

---

## Application de ces informations de configuration à Developer for System z

Comme indiqué précédemment, Developer for System z dépend de la validité du nom d'hôte TCP/IP lors de son initialisation, lors de l'utilisation d'APPC. Cela implique que les différents fichiers de configuration TCP/IP et du programme de résolution soient configurés correctement.

L'exemple ci-dessous présente certaines tâches de la configuration TCP/IP et du programme de résolution. Il ne couvre pas l'ensemble de la configuration TCP/IP ou du programme de résolution, il souligne uniquement certains aspects clés qui peuvent s'appliquer à votre site :

1. Dans le JCL ci-dessous, comme vous pouvez le constater, le protocole TCP/IP va se servir de SYS1.TCPPARMS(TCPDATA) pour déterminer le nom d'hôte de la pile.

```
//TCPIP PROC PARM='CTRACE(CTIEZB00)',PROF=TCPPROF,DATA=TCPDATA
//*
//* TCP/IP NETWORK
//*
//TCPIP EXEC PGM=EZBTCP,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) indique que le nom du système doit être le nom d'hôte et qu'aucun serveur de noms de domaine (DNS) n'est utilisé. Tous les noms sont résolus en consultant le tableau du site.

```
; HOSTNAME indique le nom d'hôte TCP de ce système. S'il n'est pas
; spécifié, le paramètre HOSTNAME par défaut sera le nom de noeud spécifié
; dans le membre IEFSSNx PARMLIB.
;
; HOSTNAME
;
; DOMAINORIGIN indique l'origine du domaine qui sera ajoutée
; aux noms d'hôte transmis au programme de résolution. Si un nom d'hôte contient
; des points, le paramètre DOMAINORIGIN ne sera pas ajouté au
; nom d'hôte.
;
DOMAINORIGIN RALEIGH.IBM.COM
;
; NSINTERADDR indique l'adresse IP du serveur de noms.
```

```

; LOOPBACK (14.0.0.0) indique votre serveur de noms local. Si vous n'utilisez
; pas de serveur de noms, ne codez pas d'instruction NSINTERADDR.
; (Mettez en commentaire la ligne NSINTERADDR ci-dessous). Cette action va
; résoudre tous les noms via la consultation du tableau de site.
;
; NSINTERADDR 14.0.0.0
;
; TRACE RESOLVER va créer une trace complète des requêtes à destination
; et des réponses provenant du serveur de noms ou des tableaux de site
; à écrire dans la console de l'utilisateur. Cette commande s'applique à
des fins de débogage uniquement.
;
; TRACE RESOLVER

```

3. Dans le langage JCL du programme de résolution, vous pouvez constater que l'instruction SETUP DD n'est pas utilisée. Comme mentionné dans «Présentation des programmes de résolution», à la page 206, cela signifie que GLOBALTCPIPDATA et les autres variables ne seront pas utilisées.

```

//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. Supposez que la variable d'environnement RESOLVER\_CONFIG n'est pas définie, le tableau 40, à la page 211 vous montre que le programme de résolution va tenter d'utiliser /etc/resolv.conf comme fichier de configuration de base.

```

TCPIPJOBNAME TCPIP
DomainOrigin RALEIGH.IBM.COM
HostName CDFMVS08

```

Comme mentionné dans «Ordres de recherche utilisés dans l'environnement z/OS UNIX», à la page 207, le fichier de configuration de base contient des instructions TCPIP.DATA. Si le nom du système est CDFMVS08 (TCPDATA indique que le nom du système est utilisé comme nom d'hôte), vous pouvez constater que /etc/resolv.conf est en synchronisation avec SYS1.TCPPARMS(TCPDATA). Aucune définition de système de nom de domaine n'existe, par conséquent la consultation du tableau de site sera utilisée.

5. Le tableau 40, à la page 211 vous indique également que vous n'avez rien à faire pour utiliser la table de conversion ASCII-EBCDIC par défaut.
6. En supposant que la commande TSO **MAKESITE** n'est pas utilisée (possibilité de créer les variables X\_SITE et X\_ADDR), /etc/hosts sera le tableau de site désigné pour la consultation du nom.

```

Resolver /etc/hosts file cdfmvs08
9.42.112.75 cdfmvs08 # CDFMVS08 Host
9.42.112.75 cdfmvs08.raleigh.ibm.com # CDFMVS08 Host
127.0.0.1 localhost

```

Le contenu minimal de ce fichier comprend des informations sur le système actuel. Dans l'exemple ci-dessus, cdfmvs08 et cdfmvs08.raleigh.ibm.com sont définis en tant que nom valide pour l'adresse IP du système z/OS.

Si vous utilisiez un serveur de noms de domaine (DNS), le DNS contiendrait des informations /etc/hosts, et /etc/resolv.conf et SYS1.TCPPARMS(TCPDATA) auraient des instructions identifiant le DNS sur votre système.

Pour éviter toute confusion, il est recommandé de conserver les fichiers de configuration TCP/IP et du programme de résolution en synchronisation les uns avec les autres.

Tableau 40. Définitions locales disponibles pour le programme de résolution

| description de type de fichier                               | API concernée(s)                                                                                                                                                                                                                                          | Fichiers candidats                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|--------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Fichiers de configuration du programme de résolution de base | Toutes les API                                                                                                                                                                                                                                            | <ol style="list-style-type: none"> <li>1. GLOBALTCPIPDATA</li> <li>2. Variable d'environnement RESOLVER_CONFIG</li> <li>3. /etc/resolv.conf</li> <li>4. SYSTCPD DD-name</li> <li>5. userid.TCPIP.DATA</li> <li>6. jobname.TCPIP.DATA</li> <li>7. SYS1.TCPPARMS(TCPDATA)</li> <li>8. DEFAULTTCPIPDATA</li> <li>9. TCPIP.TCPIP.DATA</li> </ol>                                                                                                                                                                                                  |
| Tables de conversion                                         | Toutes les API                                                                                                                                                                                                                                            | <ol style="list-style-type: none"> <li>1. Variable d'environnement X_XLATE</li> <li>2. userid.STANDARD.TCPXLBIN</li> <li>3. jobname.STANDARD.TCPXLBIN</li> <li>4. hlq.STANDARD.TCPXLBIN</li> <li>5. Table de conversion fournie par le programme de résolution, membre STANDARD dans SEZATCPX</li> </ol>                                                                                                                                                                                                                                      |
| Tables de système hôte local                                 | endhostent<br>endnetent<br>getaddrinfo<br>gethostbyaddr<br>gethostbyname<br>gethostent<br>GetHostNumber<br>GetHostResol<br>GetHostString<br>getnameinfo<br>getnetbyaddr<br>getnetbyname<br>getnetent<br>IsLocalHost<br>Resolve<br>sethostent<br>setnetent | IPv4 <ol style="list-style-type: none"> <li>1. Variable d'environnement X_SITE</li> <li>2. Variable d'environnement X_ADDR</li> <li>3. /etc/hosts</li> <li>4. userid.HOSTS.xxxxINFO</li> <li>5. jobname.HOSTS.xxxxINFO</li> <li>6. hlq.HOSTS.xxxxINFO</li> </ol> IPv6 <ol style="list-style-type: none"> <li>1. GLOBALIPNODES</li> <li>2. Variable d'environnement RESOLVER_IPNODES</li> <li>3. userid.ETC.IPNODES</li> <li>4. jobname.ETC.IPNODES</li> <li>5. hlq.ETC.IPNODES</li> <li>6. DEFAULTIPNODES</li> <li>7. /etc/ipnodes</li> </ol> |

**Remarque :** Le tableau 40 est une copie partielle d'un tableau situé dans le document *Communications Server: IP Configuration Guide* (SC31-8775). Consultez ce manuel pour voir le tableau complet.

## Résolution erronée de l'adresse hôte

Des incidents dans lesquels le programme de résolution TCP/IP ne parvient pas à résoudre correctement l'adresse hôte, peuvent être engendrés par un fichier de configuration manquant ou incomplet du programme de résolution. Le message `lock.log` suivant est une indication évidente de l'apparition de cet incident :

```
clientip(0.0.0.0) <> callerip(<adresse IP de l'hôte>)
```



Pour le vérifier, exécutez la procédure de vérification d'installation TCP/IP fekfivpt, comme l'indique la section "Vérification de l'installation" du *Guide de configuration de l'hôte* (SC11-6285). La section relative à la configuration du programme de résolution de la sortie ressemblera à l'exemple suivant :

```
Resolver Trace Initialization Complete -> 2008/07/02 13:11:54.745964
```

```
res_init Resolver values:
Global Tcp/Ip Dataset = None
Default Tcp/Ip Dataset = None
Local Tcp/Ip Dataset = /etc/resolv.conf
Translation Table = Default
UserId/JobName = USERID
Caller API = LE C Sockets
Caller Mode = EBCDIC
```

Vérifiez que les définitions du fichier référencées par "Fichier Tcp/Ip local" sont correctes.

Cette zone est vide si vous n'utilisez pas un nom par défaut pour le fichier du programme de résolution IP (à l'aide de l'ordre de recherche z/OS UNIX). Si tel est le cas, ajoutez l'instruction suivante dans `rsed.envvars`, où <fichier du programme de résolution> ou <données du programme de résolution> représente le nom de votre fichier de programme de résolution IP :

```
RESOLVER_CONFIG=<fichier du programme de résolution>
```

ou

```
RESOLVER_CONFIG='<données du programme de résolution>'
```



---

# Bibliographie

---

## Publications référencées

Les publications suivantes sont référencées dans ce document :

Tableau 41. Publications référencées

| Titre de la publication                                                                                                          | Référence de la commande | Référence              | Site Web de référence                                                                                                                                                               |
|----------------------------------------------------------------------------------------------------------------------------------|--------------------------|------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Répertoire du programme d'IBM Rational Developer for System z                                                                    | GI11-7314                | Developer for System z | <a href="http://www-01.ibm.com/support/docview.wss?uid=swg27038517">http://www-01.ibm.com/support/docview.wss?uid=swg27038517</a>                                                   |
| Répertoire du programme pour l'utilitaire hôte IBM Rational Developer for System z                                               | GI13-2864                | Developer for System z | <a href="http://www-01.ibm.com/support/docview.wss?uid=swg27038517">http://www-01.ibm.com/support/docview.wss?uid=swg27038517</a>                                                   |
| Conditions requises pour Rational Developer for System z                                                                         | SC23-7659                | Developer for System z | <a href="http://www-01.ibm.com/support/docview.wss?uid=swg27038517">http://www-01.ibm.com/support/docview.wss?uid=swg27038517</a>                                                   |
| Rational Developer for System z - Guide de démarrage rapide de la configuration hôte                                             | GI11-7313                | Developer for System z | <a href="http://www-01.ibm.com/support/docview.wss?uid=swg27038517">http://www-01.ibm.com/support/docview.wss?uid=swg27038517</a>                                                   |
| Rational Developer for System z - Guide de configuration de l'hôte                                                               | SC11-6285                | Developer for System z | <a href="http://www-01.ibm.com/support/docview.wss?uid=swg27038517">http://www-01.ibm.com/support/docview.wss?uid=swg27038517</a>                                                   |
| Rational Developer for System z - Guide de référence de configuration de l'hôte                                                  | SC11-6869                | Developer for System z | <a href="http://www-01.ibm.com/support/docview.wss?uid=swg27038517">http://www-01.ibm.com/support/docview.wss?uid=swg27038517</a>                                                   |
| Rational Developer for System z - Guide de l'utilitaire de configuration de l'hôte                                               | SC11-6859                | Developer for System z | <a href="http://www-01.ibm.com/support/docview.wss?uid=swg27038517">http://www-01.ibm.com/support/docview.wss?uid=swg27038517</a>                                                   |
| Rational Developer for System z - Messages et codes                                                                              | SC14-7497                | Developer for System z | <a href="http://www-01.ibm.com/support/docview.wss?uid=swg27038517">http://www-01.ibm.com/support/docview.wss?uid=swg27038517</a>                                                   |
| Rational Developer for System z - Réponses aux questions les plus courantes en matière de maintenance et de configuration d'hôte | SC14-7373                | Developer for System z | <a href="http://www-01.ibm.com/support/docview.wss?uid=swg27038517">http://www-01.ibm.com/support/docview.wss?uid=swg27038517</a>                                                   |
| Rational Developer for System z Common Access Repository Manager Developer's Guide                                               | SC23-7660                | Developer for System z | <a href="http://www-01.ibm.com/support/docview.wss?uid=swg27038517">http://www-01.ibm.com/support/docview.wss?uid=swg27038517</a>                                                   |
| Conditions requises pour Rational Developer for System z                                                                         | SC23-7659                | Developer for System z | <a href="http://www.ibm.com/software/rational/products/developer/systemz/library/index.html">http://www.ibm.com/software/rational/products/developer/systemz/library/index.html</a> |

Tableau 41. Publications référencées (suite)

| Titre de la publication                                                                | Référence de la commande | Référence              | Site Web de référence                                                                                                                                                               |
|----------------------------------------------------------------------------------------|--------------------------|------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Rational Developer for System z - Guide de démarrage rapide de configuration de l'hôte | GI11-7313                | Developer for System z | <a href="http://www.ibm.com/software/rational/products/developer/systemz/library/index.html">http://www.ibm.com/software/rational/products/developer/systemz/library/index.html</a> |
| SCLM Developer Toolkit - Guide d'administration                                        | SC11-6464                | Developer for System z | <a href="http://www-01.ibm.com/support/docview.wss?uid=swg27038517">http://www-01.ibm.com/support/docview.wss?uid=swg27038517</a>                                                   |
| Using APPC to provide TSO command services                                             | SC14-7291                | Livre blanc            | <a href="http://www-01.ibm.com/support/docview.wss?uid=swg27038517">http://www-01.ibm.com/support/docview.wss?uid=swg27038517</a>                                                   |
| Using ISPF Client Gateway to provide CARMA services                                    | SC14-7292                | Livre blanc            | <a href="http://www-01.ibm.com/support/docview.wss?uid=swg27038517">http://www-01.ibm.com/support/docview.wss?uid=swg27038517</a>                                                   |
| Communications Server IP Configuration Guide                                           | SC31-8775                | z/OS 1.13              | <a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>                                                   |
| Communications Server IP Configuration Reference                                       | SC31-8776                | z/OS 1.13              | <a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>                                                   |
| Communications Server IP Diagnosis Guide                                               | GC31-8782                | z/OS 1.13              | <a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>                                                   |
| Communications Server IP System Administrator's Commands                               | SC31-8781                | z/OS 1.13              | <a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>                                                   |
| Communications Server SNA Network Implementation Guide                                 | SC31-8777                | z/OS 1.13              | <a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>                                                   |
| Communications Server SNA Operations                                                   | SC31-8779                | z/OS 1.13              | <a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>                                                   |
| Cryptographic Services System SSL Programming                                          | SC24-5901                | z/OS 1.13              | <a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>                                                   |
| DFSMS Macro Instructions for Data Sets                                                 | SC26-7408                | z/OS 1.13              | <a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>                                                   |
| DFSMS Using data sets                                                                  | SC26-7410                | z/OS 1.13              | <a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>                                                   |
| Language Environment Customization                                                     | SA22-7564                | z/OS 1.13              | <a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>                                                   |
| Language Environment Debugging Guide                                                   | GA22-7560                | z/OS 1.13              | <a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>                                                   |
| MVS Diagnosis: Tools and Service Aids                                                  | GA22-7589                | z/OS 1.13              | <a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>                                                   |
| MVS Initialization and Tuning Guide                                                    | SA22-7591                | z/OS 1.13              | <a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>                                                   |
| MVS Initialization and Tuning Reference                                                | SA22-7592                | z/OS 1.13              | <a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>                                                   |
| MVS JCL Reference                                                                      | SA22-7597                | z/OS 1.13              | <a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>                                                   |
| MVS Planning APPC/MVS Management                                                       | SA22-7599                | z/OS 1.13              | <a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>                                                   |
| MVS Planning Workload Management                                                       | SA22-7602                | z/OS 1.13              | <a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>                                                   |

Tableau 41. Publications référencées (suite)

| Titre de la publication                             | Référence de la commande | Référence                 | Site Web de référence                                                                                                                                                                                                                                             |
|-----------------------------------------------------|--------------------------|---------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| MVS System Commands                                 | SA22-7627                | z/OS 1.13                 | <a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>                                                                                                                                 |
| Security Server RACF Command Language Reference     | SA22-7687                | z/OS 1.13                 | <a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>                                                                                                                                 |
| Security Server RACF Security Administrator's Guide | SA22-7683                | z/OS 1.13                 | <a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>                                                                                                                                 |
| TSO/E Customization                                 | SA22-7783                | z/OS 1.13                 | <a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>                                                                                                                                 |
| TSO/E REXX Reference                                | SA22-7790                | z/OS 1.13                 | <a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>                                                                                                                                 |
| UNIX System Services Command Reference              | SA22-7802                | z/OS 1.13                 | <a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>                                                                                                                                 |
| UNIX System Services Planning                       | GA22-7800                | z/OS 1.13                 | <a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>                                                                                                                                 |
| UNIX System Services User's Guide                   | SA22-7801                | z/OS 1.13                 | <a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>                                                                                                                                 |
| Using REXX and z/OS UNIX System Services            | SA22-7806                | z/OS 1.13                 | <a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>                                                                                                                                 |
| Java Diagnostic Guide                               | SC34-6650                | Java 6.0                  | <a href="http://www.ibm.com/developerworks/java/jdk/diagnosis/">http://www.ibm.com/developerworks/java/jdk/diagnosis/</a>                                                                                                                                         |
| Java SDK and Runtime Environment User Guide         | /                        | Java 6.0                  | <a href="http://www-03.ibm.com/servers/eserver/zseries/software/java/">http://www-03.ibm.com/servers/eserver/zseries/software/java/</a>                                                                                                                           |
| Resource Definition Guide                           | SC34-6430                | CICSTS 3.1                | <a href="http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html">http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html</a>                                                                                                                       |
| Resource Definition Guide                           | SC34-6815                | CICSTS 3.2                | <a href="http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html">http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html</a>                                                                                                                       |
| Resource Definition Guide                           | SC34-7000                | CICSTS 4.1                | <a href="https://publib.boulder.ibm.com/infocenter/cicsts/v4r1/index.jsp?topic=/com.ibm.cics.ts.home.doc/library/library_html.html">https://publib.boulder.ibm.com/infocenter/cicsts/v4r1/index.jsp?topic=/com.ibm.cics.ts.home.doc/library/library_html.html</a> |
| Resource Definition Guide                           | SC34-7181                | CICSTS 4.2                | <a href="https://publib.boulder.ibm.com/infocenter/cicsts/v4r2/index.jsp?topic=/com.ibm.cics.ts.home.doc/library/library_html.html">https://publib.boulder.ibm.com/infocenter/cicsts/v4r2/index.jsp?topic=/com.ibm.cics.ts.home.doc/library/library_html.html</a> |
| RACF Security Guide                                 | SC34-6454                | CICSTS 3.1                | <a href="http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html">http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html</a>                                                                                                                       |
| RACF Security Guide                                 | SC34-6835                | CICSTS 3.2                | <a href="http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html">http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html</a>                                                                                                                       |
| RACF Security Guide                                 | SC34-7003                | CICSTS 4.1                | <a href="https://publib.boulder.ibm.com/infocenter/cicsts/v4r1/index.jsp?topic=/com.ibm.cics.ts.home.doc/library/library_html.html">https://publib.boulder.ibm.com/infocenter/cicsts/v4r1/index.jsp?topic=/com.ibm.cics.ts.home.doc/library/library_html.html</a> |
| RACF Security Guide                                 | SC34-7179                | CICSTS 4.2                | <a href="https://publib.boulder.ibm.com/infocenter/cicsts/v4r2/index.jsp?topic=/com.ibm.cics.ts.home.doc/library/library_html.html">https://publib.boulder.ibm.com/infocenter/cicsts/v4r2/index.jsp?topic=/com.ibm.cics.ts.home.doc/library/library_html.html</a> |
| Language Reference                                  | SC27-1408                | Enterprise COBOL for z/OS | <a href="http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html">http://www-03.ibm.com/systems/z/os/zos/bkserv/zapplsbooks.html</a>                                                                                                                       |

Les sites Web suivants sont référencés dans le présent document :

Tableau 42. Sites Web référencés

| Description                                              | Site Web de référence                                                                                                                                                                           |
|----------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Developer for System z - Centre de documentation         | <a href="http://pic.dhe.ibm.com/infocenter/ratdevz/v9r0/index.jsp">http://pic.dhe.ibm.com/infocenter/ratdevz/v9r0/index.jsp</a>                                                                 |
| Bibliothèque Developer for System z                      | <a href="http://www-01.ibm.com/support/docview.wss?uid=swg27038517">http://www-01.ibm.com/support/docview.wss?uid=swg27038517</a>                                                               |
| Page d'accueil Developer for System z                    | <a href="http://www-03.ibm.com/software/products/us/en/developerforsystemz/">http://www-03.ibm.com/software/products/us/en/developerforsystemz/</a>                                             |
| Developer for System z Recommended service               | <a href="http://www-01.ibm.com/support/docview.wss?rs=2294&amp;context=SS2QJ2&amp;uid=swg27006335">http://www-01.ibm.com/support/docview.wss?rs=2294&amp;context=SS2QJ2&amp;uid=swg27006335</a> |
| Developer for System z - Demande d'amélioration          | <a href="https://www.ibm.com/developerworks/support/rational/rfe/">https://www.ibm.com/developerworks/support/rational/rfe/</a>                                                                 |
| Bibliothèque Internet z/OS                               | <a href="http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www-03.ibm.com/servers/eserver/zseries/zos/bkserv/</a>                                                               |
| Centre de documentation CICSTS                           | <a href="https://publib.boulder.ibm.com/infocenter/cicsts/v4r1/index.jsp">https://publib.boulder.ibm.com/infocenter/cicsts/v4r1/index.jsp</a>                                                   |
| IBM Tivoli Directory Server                              | <a href="http://www-01.ibm.com/software/tivoli/products/directory-server/">http://www-01.ibm.com/software/tivoli/products/directory-server/</a>                                                 |
| Plug-ins d'outils d'identification des incidents         | <a href="http://www-01.ibm.com/software/awdtools/deployment/pdplugins/">http://www-01.ibm.com/software/awdtools/deployment/pdplugins/</a>                                                       |
| Télécharger Apache Ant                                   | <a href="http://ant.apache.org/">http://ant.apache.org/</a>                                                                                                                                     |
| Documentation du l'outil de clé Java                     | <a href="http://java.sun.com/j2se/1.5.0/docs/tooldocs/solaris/keytool.html">http://java.sun.com/j2se/1.5.0/docs/tooldocs/solaris/keytool.html</a>                                               |
| Page d'accueil du support de l'autorité de certification | <a href="https://support.ca.com/">https://support.ca.com/</a>                                                                                                                                   |

## Publications d'information

Les publications suivantes peuvent s'avérer utiles pour vous aider à comprendre les incidents de configuration pour les composants de système hôte requis :

Tableau 43. Publications d'information

| Titre de la publication                                                        | Référence de la commande | Référence | Site Web de référence                                                   |
|--------------------------------------------------------------------------------|--------------------------|-----------|-------------------------------------------------------------------------|
| ABCs of z/OS System Programming Volume 9 (z/OS UNIX)                           | SG24-6989                | Redbook   | <a href="http://www.redbooks.ibm.com/">http://www.redbooks.ibm.com/</a> |
| Guide du programmeur système pour : Workload Manager                           | SG24-6472                | Redbook   | <a href="http://www.redbooks.ibm.com/">http://www.redbooks.ibm.com/</a> |
| TCPIP Implementation Volume 1: Base Functions, Connectivity, and Routing       | SG24-7532                | Redbook   | <a href="http://www.redbooks.ibm.com/">http://www.redbooks.ibm.com/</a> |
| TCPIP Implementation Volume 3: High Availability, Scalability, and Performance | SG24-7534                | Redbook   | <a href="http://www.redbooks.ibm.com/">http://www.redbooks.ibm.com/</a> |
| TCP/IP Implementation Volume 4: Security and Policy-Based Networking           | SG24-7535                | Redbook   | <a href="http://www.redbooks.ibm.com/">http://www.redbooks.ibm.com/</a> |

Tableau 43. Publications d'information (suite)

| Titre de la publication          | Référence de la commande | Référence | Site Web de référence                                                   |
|----------------------------------|--------------------------|-----------|-------------------------------------------------------------------------|
| Tivoli Directory Server for z/OS | SG24-7849                | Redbook   | <a href="http://www.redbooks.ibm.com/">http://www.redbooks.ibm.com/</a> |



---

## Glossaire

### action de verrouillage

Verrouille un membre.

### attribut bidirectionnel

Type de texte, orientation du texte, Permutation numérique et Permutation symétrique.

### base de données

Collection d'éléments de données liés entre eux ou indépendants, stockés ensemble, destinée à être utilisée dans une ou plusieurs applications.

### bibliothèque de chargement

Bibliothèque contenant des modules de chargement.

### bidirectionnel (bidi)

Caractérise des scripts, tels que l'arabe et l'hébreu, qui s'exécutent généralement de droite à gauche, à l'exception des nombres, qui s'exécutent de gauche à droite. La définition de ce terme est extraite du glossaire Localization Industry Standards Association (LISA).

### compiler

1. Dans les langages Integrated Language Environment (ILE), traduire des instructions source en modules qui peuvent ensuite être associés en programmes ou programmes de service.
2. Traduire tout ou partie d'un programme exprimé en langage haut niveau en un programme exprimé en langage intermédiaire, un langage d'assemblage ou un langage machine.

### conteneur

1. Dans CoOperative Development Environment/400, objet système qui contient et organise des fichiers source. Par exemple, un conteneur peut être une bibliothèque i5/OS ou un fichier partitionné pour MVS.
2. Dans l'architecture Java EE, entité qui fournit la gestion du cycle de vie, la

sécurité, le déploiement et des services d'exécution pour les composants.

(Sun) Chaque type de conteneur (EJB, Web, JSP, servlet, applet et client d'application) fournit également des services spécifiques des composants.

3. Dans Backup Recovery and Media Services, objet physique utilisé pour stocker ou déplacer des supports, tels qu'une case, un chemin ou une armoire.
4. Dans un serveur Virtual Tape Server (VTS), réceptacle dans lequel un ou plusieurs volumes logiques exportés (LVOL) peuvent être stockés. Un volume empilé qui contient un ou plusieurs LVOL et qui réside en dehors d'une bibliothèque VTS est considéré comme le conteneur de ces volumes.
5. Lieu de stockage physique des données. Par exemple, un fichier, un répertoire ou un périphérique.
6. Colonne ou rang utilisé pour la disposition d'un portlet ou d'un autre conteneur sur une page.
7. Élément de l'interface utilisateur qui contient des objets. Dans le gestionnaire de dossier, objet qui contient les autres dossiers ou documents.

### débogage

Détecter, diagnostiquer et éliminer les erreurs des programmes.

### demande de génération

Demande du client pour effectuer une transaction de génération.

### désinstallation en mode silencieux

Processus de désinstallation qui n'envoie pas les messages vers la console mais stocke les messages et erreurs dans des fichiers journaux après que la commande d'installation a été appelée.

**fichier** Unité principale de stockage et d'extraction des données, qui est constituée d'une collection de données disposée selon une des structures imposées et décrites par les données de contrôle auxquelles le système a accès.

**fichier de réponses**

1. Fichier qui contient un ensemble de réponses prédéfinies aux questions envoyées par un programme afin d'éviter d'entrer ces valeurs une par une.
2. Fichier ASCII qui peut être personnalisé au moyen des données de configuration pour automatiser une installation. Les données de configuration sont généralement entrées lors d'une installation interactive alors qu'un fichier de réponses permet d'effectuer l'installation sans aucune intervention.

**ID action**

Identificateur numérique d'une action entre 0 et 999

**installation en mode silencieux**

Installation qui n'envoie pas les messages vers la console mais stocke les messages et les erreurs dans des fichiers journaux. Une installation en mode silencieux peut également utiliser des fichiers de réponses pour entrer les données.

**instance de référentiels**

Projet ou composant existant dans un SCM.

**interpréteur**

Programme qui traduit et exécute successivement toutes les instructions en langage de programmation haut niveau.

**interpréteur de commandes**

Interface logicielle entre les utilisateurs et le système d'exploitation qui interprète les commandes et les interactions utilisateur et les communique au système d'exploitation. Un ordinateur possède des interpréteurs de commandes sur plusieurs niveaux, qui correspondent aux différents niveaux d'interaction avec l'utilisateur.

**isomorphe**

A chaque élément composé (c'est-à-dire, contenant d'autres éléments) du document

d'instance XML lancé à partir de la racine correspond un seul et unique élément de groupe COBOL dont la profondeur d'imbrication est identique à la profondeur d'imbrication de son équivalent XML. Chaque élément non composé (à savoir, ne contenant pas d'autres éléments) dans le document d'instance XML, en partant du haut, comporte une seule donnée élémentaire COBOL correspondante dont la profondeur d'imbrication est identique à la profondeur d'imbrication de son équivalent XML et dont l'adresse mémoire lors de l'exécution peut être identifiée de manière unique.

**liste des tâches**

Liste des procédures qui peuvent être exécutées par un seul flux de contrôle.

**mémoire tampon des erreurs**

Partie de mémoire servant à contenir provisoirement les données de sortie des erreurs.

**nom d'interpréteur de commandes**

Nom de l'interface de l'interpréteur de commandes.

**non isomorphe**

Mappage simple d'éléments COBOL et d'éléments XML faisant partie de documents XML et de groupes COBOL de forme non identique (non isomorphe). Un mappage non isomorphe peut également être créé entre des éléments non isomorphes de structures isomorphes.

**passerelle**

1. Composant intermédiaire qui relie Internet aux environnements intranet lors des appels de service Web.
2. Logiciel qui fournit des services entre les points d'arrêt final et le reste de l'environnement Tivoli.



3. Composant de Voice over Internet Protocol constituant un pont entre VoIP et les environnements commutés par circuit.
4. Périphérique ou programme utilisé pour la connexion de réseaux ou systèmes à d'autres architectures réseau. Ces systèmes peuvent présenter des caractéristiques différentes, telles que des protocoles de communication différents, une architecture réseau ou des stratégies de sécurité différentes, la passerelle réalisant alors leur traduction et leur connexion.

### **perspective**

Groupe de vues présentant les divers aspects des ressources d'un plan de travail. L'utilisateur du plan de travail peut basculer entre les perspectives en fonction de la tâche en cours et personnaliser l'affichage des vues et des éditeurs depuis la perspective.

### **perspective Systèmes distants**

Offre une interface permettant de gérer des systèmes distants par l'intermédiaire de conventions similaires à ISPF.

**RAM** Repository Access Manager

### **référentiel**

1. Zone de stockage des données. Chaque référentiel comporte un nom et un type d'élément métier associé. Par défaut, son nom sera le même que celui de l'élément métier. Par exemple, un référentiel de factures sera appelé Factures. Il existe deux types de référentiels d'information : local (spécifiques du processus) et global (réutilisable).
2. Fichier VSAM dans lequel sont stockés les états des processus du BTS. Lorsqu'un processus ne s'exécute pas sous de contrôle du BTS, son état (et les états des tâches qui le composent) sont protégés en écriture dans un fichier de référentiel. Les états de tous les processus d'un type de processus donné (et les instances de leurs tâches) sont stockés dans le même fichier de référentiel. Les enregistrements des types multiprocessus peuvent être écrits dans ce même référentiel.

3. Zone de stockage permanente du code source et des autres ressources d'application. Dans un environnement de programmation en équipe, un référentiel partagé permet à plusieurs utilisateurs d'accéder en même temps aux ressources de l'application.
4. Collection d'informations sur les gestionnaires de file d'attente qui sont membres d'un cluster. Ces informations comprennent les noms des gestionnaires de files d'attente, leurs emplacements, leurs canaux, les files d'attente qu'ils hébergent, etc.

### **script de shell**

Fichier contenant des commandes qui peuvent être interprétées par l'interpréteur de commandes. Pour que le shell exécute les commandes du script, l'utilisateur doit saisir le nom du fichier de script à l'invite de commande du shell.

### **section de liaison**

Section de la division des données d'une unité activée (programme ou méthode appelé(e)) qui décrit les éléments de données disponibles à partir d'une unité d'activation (programme ou méthode). L'unité activée et l'unité d'activation peuvent toutes deux se référer à ces éléments de données.

### **serveur d'applications**

1. Programme qui traite toutes les opérations d'une application qui s'exécutent entre des ordinateurs dotés d'un navigateur et les applications dorsales ou bases de données de l'entreprise. Une classe spécifique de serveurs d'applications Java prend en charge la norme Java EE. Le code Java EE peut être facilement porté entre ces différents serveurs. Ils peuvent supporter des JSP et des servlets destinés au contenu Web dynamique et des EJB pour les transactions et l'accès aux bases de données.
2. Cible d'une demande émise à partir d'une application distante. Dans l'environnement DB2, la fonction du serveur d'applications est fournie par la fonction de données réparties et permet d'accéder aux données DB2 à partir d'applications distantes.

3. Programme serveur dans un réseau réparti qui fournit l'environnement d'exécution d'un programme d'application.
4. Cible d'une demande émise par un demandeur d'application. Le système de gestion de base de données du site du serveur de l'application fournit les données demandées.
5. Logiciel qui traite les communications avec le client lorsque celui-ci demande un actif et les requêtes du Content Manager.

#### **session de débogage**

Tâches de débogage exécutées entre l'heure à laquelle le développeur lance le débogage, et l'heure à laquelle il sort de l'application.

#### **sidedeck**

Bibliothèque qui publie les fonctions d'un programme DLL. Les noms des entrées et des modules sont stockés dans la bibliothèque après la compilation du code source.

#### **système de fichiers distant**

Système de fichiers résidant sur un serveur ou un système d'exploitation séparé.

#### **système distant**

Tout autre système du réseau avec lequel votre système peut communiquer.

#### **transaction de génération**

Travail démarré sur MVS pour effectuer des générations après qu'une demande de génération a été reçue du client.

**URL** Uniform Resource Locator

#### **utilitaire ISPF (Interactive System Productivity Facility)**

Logiciel sous licence IBM servant d'éditeur plein écran et de gestionnaire de boîte de dialogue. Utilisé dans l'écriture de programmes d'application, il permet de générer des panneaux d'écran standard et des boîtes de dialogue interactives entre le programmeur et l'utilisateur final. ISPF est constitué de quatre composants principaux : DM, PDF, SCLM et C/S. Le composant DM (Dialog Manager) est le gestionnaire qui fournit des services pour

les boîtes de dialogue et les utilisateurs finaux. Le composant PDF (Program Development Facility) offre des services d'aide au développeur de boîtes de dialogue ou d'applications. Le composant SCLM (Software Configuration Library Manager) offre aux développeurs d'applications des services destinés à leurs bibliothèques de développement d'applications. Le composant C/S (Client/Server), qui permet d'exécuter ISPF sur un poste de travail programmable, d'afficher les panneaux au moyen de la fonction d'affichage sur le système d'exploitation de votre poste de travail et d'intégrer des outils et données de poste de travail au moyen des outils et des données de l'hôte.

#### **vue Console de sortie**

Affiche les données de sortie d'un processus et vous permet d'entrer à partir du clavier les données d'un processus.

#### **vue Définition de données**

Affiche une image locale des bases de données, ainsi que des objets qu'elles contiennent. Elle fournit également les fonctions nécessaires pour manipuler ces objets et les exporter vers une base de données distante.

#### **vue de sortie**

Affiche les messages, les paramètres et résultats associés aux objets sur lesquels vous travaillez.

#### **vue du navigateur**

Fournit une vue hiérarchique des ressources du plan de travail.

#### **vue Référentiels**

Affiche les emplacements des référentiels CVS qui ont été ajoutés à votre plan de travail.

#### **vue Serveurs**

Présente une liste de tous les serveurs et des configurations qui leur sont associées.





---

## Mentions légales d'IBM Rational Developer for System z

© Copyright IBM Corporation 2009, 2013.

U.S. Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Le présent document peut contenir des informations ou des références concernant certains produits, logiciels ou services IBM non annoncés dans certains pays. Pour plus de détails, référez-vous aux documents d'annonce disponibles dans votre pays ou adressez-vous à votre partenaire commercial IBM. Toute référence à un produit, logiciel ou service IBM n'implique pas que seul ce produit, logiciel ou service IBM puisse être utilisé. Tout autre élément fonctionnellement équivalent peut être utilisé, s'il n'enfreint aucun droit d'IBM. Toutefois, il appartient à l'utilisateur d'évaluer et de vérifier le fonctionnement de produits, logiciels ou services non expressément référencés par IBM.

IBM peut détenir des brevets ou des demandes de brevet couvrant les produits mentionnés dans la présente documentation. La remise de ce document ne vous donne aucun droit de licence sur ces brevets ou demandes de brevet. Si vous désirez recevoir des informations concernant l'acquisition de licences, veuillez en faire la demande par écrit à l'adresse suivante :

*IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
U.S.A.*

Pour le Canada, veuillez adresser votre courrier à :

*IBM Director of Commercial Relations  
IBM Canada Ltd.  
3600 Steeles Avenue East  
Markham, Ontario  
L3R 9Z7  
Canada*

Les informations sur les licences concernant les produits utilisant un jeu de caractères double octet peuvent être obtenues auprès du IBM Intellectual Property Department de votre pays ou par écrit à l'adresse suivante :

*Intellectual Property Licensing  
Legal and Intellectual Property Law  
IBM Japan, Ltd.  
1623-14, Shimotsuruma, Yamato-shi  
Kanagawa 242-8502 Japan*

**Le paragraphe suivant ne s'applique ni au Royaume-Uni, ni dans aucun pays dans lequel il serait contraire aux lois locales. LE PRESENT DOCUMENT EST LIVRE "EN L'ETAT" SANS AUCUNE GARANTIE EXPLICITE OU IMPLICITE. IBM DECLINE NOTAMMENT TOUTE RESPONSABILITE RELATIVE A CES INFORMATIONS EN CAS DE CONTREFACON AINSI QU'EN CAS DE DEFAULT**

D'APTITUDE A L'EXECUTION D'UN TRAVAIL DONNE. Certaines juridictions n'autorisent pas l'exclusion des garanties implicites, auquel cas l'exclusion ci-dessus ne vous sera pas applicable.

Le présent document peut contenir des inexactitudes ou des coquilles. Ce document est mis à jour périodiquement. Chaque nouvelle édition inclut les mises à jour. IBM peut, à tout moment et sans préavis, modifier les produits et logiciels décrits dans ce document.

Les références à des sites web non IBM sont fournies à titre d'information uniquement et n'impliquent en aucun cas une adhésion aux données qu'ils contiennent. Les éléments figurant sur ces sites web ne font pas partie des éléments du présent produit IBM et l'utilisation de ces sites relève de votre seule responsabilité.

IBM pourra utiliser ou diffuser, de toute manière qu'elle jugera appropriée et sans aucune obligation de sa part, tout ou partie des informations qui lui seront fournies.

Les licenciés souhaitant obtenir des informations permettant : (i) l'échange des données entre des logiciels créés de façon indépendante et d'autres logiciels (dont celui-ci), et (ii) l'utilisation mutuelle des données ainsi échangées, doivent adresser leur demande à :

*Intellectual Property Dept. for Rational Software  
IBM Corporation  
5 Technology Park Drive  
Westford, MA 01886  
U.S.A.*

Ces informations peuvent être soumises à des conditions particulières, prévoyant notamment le paiement d'une redevance.

Le logiciel sous licence décrit dans cette documentation et tous les éléments sous licence disponibles s'y rapportant sont fournis par IBM conformément aux dispositions de l'IBM Customer Agreement, des Conditions internationales d'utilisation des logiciels IBM ou de tout autre accord équivalent.

Les données de performance indiquées dans ce document ont été déterminées dans un environnement contrôlé. Par conséquent, les résultats peuvent varier de manière significative selon l'environnement d'exploitation utilisé. Certaines mesures évaluées sur des systèmes en cours de développement ne sont pas garanties sur tous les systèmes disponibles. En outre, elles peuvent résulter d'extrapolations. Les résultats peuvent donc varier. Il incombe aux utilisateurs de ce document de vérifier si ces données sont applicables à leur environnement d'exploitation.

Les informations concernant des produits non IBM ont été obtenues auprès des fournisseurs de ces produits, par l'intermédiaire d'annonces publiques ou via d'autres sources disponibles. IBM n'a pas testé ces produits et ne peut confirmer l'exactitude de leurs performances ni leur compatibilité. Elle n'accepte aucune réclamation concernant des produits non IBM. Toute question concernant les performances de produits non IBM doit être adressée aux fournisseurs de ces produits.

Toute instruction relative aux intentions d'IBM pour ses opérations à venir est susceptible d'être modifiée ou annulée sans préavis, et doit être considérée uniquement comme un objectif.

Ces informations sont fournies uniquement à titre de planification. Elles peuvent faire l'objet de modifications tant que les produits décrits ne seront pas disponibles.

La présente documentation peut contenir des exemples de données et de rapports utilisés couramment dans l'environnement professionnel. Ces exemples mentionnent des noms fictifs de personnes, de sociétés, de marques ou de produits à des fins illustratives ou explicatives uniquement. Toute ressemblance avec des noms de personnes, de sociétés ou des données réelles serait purement fortuite.

## **Licence de copyright**

Le présent logiciel contient des exemples de programmes d'application en langage source destinés à illustrer les techniques de programmation sur différentes plateformes d'exploitation. Vous avez le droit de copier, de modifier et de distribuer ces exemples de programmes sous quelque forme que ce soit et sans paiement d'aucune redevance à IBM, à des fins de développement, d'utilisation, de vente ou de distribution de programmes d'application conformes aux interfaces de programmation des plateformes pour lesquels ils ont été écrits ou aux interfaces de programmation IBM. Ces exemples de programmes n'ont pas été rigoureusement testés dans toutes les conditions. Par conséquent, IBM ne peut garantir expressément ou implicitement la fiabilité, la maintenabilité ou le fonctionnement de ces programmes. Les programmes exemple sont fournis "EN L'ETAT" sans aucune garantie d'aucune sorte. IBM ne sera en aucun cas responsable des dommages résultant de l'utilisation de ces exemples de programme.

Toute copie totale ou partielle de ces programmes exemples et des oeuvres qui en sont dérivées doit comprendre une notice de copyright libellée comme suit:

© (nom de votre société) (année). Les segments de code sont dérivés des Programmes exemples d'IBM Corp. © Copyright IBM Corp. 2009, 2013.

Si vous visualisez la copie logicielle de ces informations, les photographies et les illustrations en couleurs peuvent ne pas s'afficher.

## **Marques**

IBM, le logo IBM et ibm.com sont des marques d'International Business Machines Corp. dans de nombreux pays. Les autres noms de produits et services sont des marques d'IBM ou d'autres sociétés. La liste actualisée de toutes les marques d'IBM est disponible sur la page web "Copyright and trademark information" à <http://www.ibm.com/legal/copytrade.shtml>.

Adobe, le logo Adobe, PostScript et logo PostScript sont des marques d'Adobe Systems Incorporated aux Etats-Unis et/ou dans certains autres pays.

Linux est une marque de Linus Torvalds aux Etats-Unis et/ou dans certains autres pays.

Windows est une marque de Microsoft Corporation aux Etats-Unis et/ou dans certains autres pays.

UNIX est une marque enregistrée de The Open Group aux Etats-Unis et/ou dans certains autres pays.

Java ainsi que tous les logos et toutes les marques incluant Java sont des marques d'Oracle et/ou de ses sociétés affiliées.

Les autres noms de produits et de services peuvent être des marques d'IBM ou d'autres sociétés.

---

## Licence de copyright

Le présent logiciel contient des exemples de programmes d'application en langage source destinés à illustrer les techniques de programmation sur différentes plateformes d'exploitation. Vous avez le droit de copier, de modifier et de distribuer ces exemples de programmes sous quelque forme que ce soit et sans paiement d'aucune redevance à IBM, à des fins de développement, d'utilisation, de vente ou de distribution de programmes d'application conformes aux interfaces de programmation des plateformes pour lesquels ils ont été écrits ou aux interfaces de programmation IBM. Ces exemples de programmes n'ont pas été rigoureusement testés dans toutes les conditions. Par conséquent, IBM ne peut garantir expressément ou implicitement la fiabilité, la maintenabilité ou le fonctionnement de ces programmes. Les exemples de programme sont fournis "EN L'ETAT", sans garantie d'aucune sorte. IBM ne peut en aucun cas être tenu pour responsable des dommages liés à l'utilisation de ces exemples de programme.

---

## Marques

IBM, le logo IBM et [ibm.com](http://www.ibm.com) sont des marques d'International Business Machines Corp. dans de nombreux pays. Les autres noms de produit et service sont des marques d'IBM ou d'autres sociétés. La liste actualisée de toutes les marques d'IBM est disponible sur la page Web "Copyright and trademark information" à <http://www.ibm.com/legal/copytrade.shtml>.

CA Endevor est une marque de CA Technologies.

Rational est une marque d'International Business Machines Corporation et de Rational Software Corporation aux Etats-Unis et/ou dans certains autres pays.

Intel et Pentium sont des marques d'Intel Corporation aux Etats-Unis et/ou dans certains autres pays.

Microsoft, Windows et le logo Windows sont des marques ou des marques déposées de Microsoft Corporation au Etats-Unis et/ou dans certains autres pays.

Java et toutes les marques et logos incluant Java sont des marques de Sun Microsystems, Inc. aux Etats-Unis et/ou dans certains autres pays.

UNIX est une marque enregistrée de The Open Group aux Etats-Unis et/ou dans certains autres pays.



---

# Index

## Caractères spéciaux

.dstoreMemLogging 172  
.dstoreTrace 172  
\_RSE\_PORTRANGE 22  
/var/rdz/pushtoclient/\*install 138, 140

## A

accès aux bibliothèques du système,  
Amélioration 118  
accès aux fichiers spoule,  
Conditionnel 29  
accès conditionnel aux fichiers spoule 29  
accusé de réception, retardement 62  
ACEE, élément géré 39  
actions conditionnelles sur les  
travaux 26  
actions sur les travaux - Limitations liées  
à l'exécution 28  
activation 129  
activation de l'exit utilisateur 157  
activer le partage de classes, machines  
virtuelles Java (JVM) 121  
administrateur non-système, droits de  
mise à jour 17  
ADNJSPAU, utilitaire  
d'administration 147  
adresse hôte non résolue, programme de  
résolution TCP/IP  
lock.log 211  
amélioration de l'accès aux bibliothèques  
du système 118  
amélioration des performances du  
contrôle d'autorisations d'accès 119  
analyse d'utilisation, espace de  
stockage 95  
APF, autorisation 184  
AQEZPCM 21  
ASCHPMxx  
MAX 107  
ASSIZEMAX 47  
attribut du système de fichiers,  
SETUID 182  
audit.action, exit utilisateur 160  
audit.log 173  
authentification, configuration SSL et  
X.509 191  
authentification, moniteur de travaux  
JES 20  
authentification du client à l'aide de  
certificats X.509 31  
authentification du gestionnaire de  
débogage 21  
authentification du moniteur de travaux  
JES 20  
authentification par le démon RSE 34  
authentification par logiciel de  
sécurité 33

autorisation APF  
FEK.SFEKAUTH 53  
autorisation de contrôle de  
programmes 183

## B

base de données de clés, Création avec  
gskkyman 200  
bibliothèques, amélioration de l'accès au  
système 118  
bibliothèques, exécution Language  
Environment 118  
bibliothèques contrôlées par programme  
MVS pour RSE, définition 48  
bibliothèques contrôlées pour RSE,  
définition par programme MVS 48  
bibliothèques d'exécution, Language  
Environment 118  
bibliothèques d'exécution Language  
Environment 118  
bibliothèques du système, amélioration  
de l'accès à 118  
bibliothèques pour RSE, définition par  
programme MVS 48  
BPXPRMxx 113  
INADDRANYCOUNT 106  
MAXASSIZE 47, 104, 187  
MAXFILEPROC 104  
MAXMMAPAREA 104  
MAXPROCSYS 103, 189  
MAXPROCUSER 103, 189  
MAXSOCKETS 106  
MAXTHREADS 103  
MAXTHREADTASKS 103  
MAXUIDS 104, 189

## C

caractéristiques de l'exit utilisateur 157  
CARMA, traçage 181  
CARMA et ports TCP/IP 61  
CEE.SCEELPA  
SYS1.PARMLIB(LPALSTxx) 118  
certificat, X.509 20  
certificat signé, auto-signé ou signé par  
l'autorité de certification 194  
certificat X.509 20  
certificats, authentification du client via  
X.509 31  
certificats X.509, authentification du  
client 31  
chiffrement, SSL ou TLS 192  
chiffrement à l'aide de la couche SSL,  
communications 22  
chiffrement à l'aide de la couche TLS,  
communications 22  
chiffrement des communications à l'aide  
de la couche SSL 22

chiffrement des communications à l'aide  
de la couche TLS 22  
choix de l'emplacement de stockage des  
clés privées et des certificats 192  
CICSplex SM Business Application  
Services (BAS) 144  
CICSTS, sécurité 38  
classification des charges de travail,  
WLM 71  
CLASSPATH 167  
clés privées et certificats, choix de  
l'emplacement de stockage 192  
clonage de la configuration RSE  
existante 195  
COBOL  
vérification à distance 181  
coexistence, mise à jour de rsed.envvars  
pour activer la coexistence 195  
commande exec d'allocation,  
utilisation 163  
commande exec REXX z/OS UNIX 159  
commandes de sécurité, utiles  
ADDGROUP 17  
ALTUSER 17  
CONNECT 17  
commandes z/OS UNIX, utiles  
chgrp 18  
chmod 18  
chown 18  
ls 18  
communication, Externe 60  
communication, interne 60  
communication, via SSL 147  
communication chiffrée  
débugueur intégré 31  
communication chiffrée, SSL 39, 147  
communication chiffrée, SSL/TLS 30  
communication chiffrée via SSL 39, 147  
communication chiffrée via SSL/TLS 30  
communications externes 60  
communications externes à des ports  
spécifiques, limitations 22  
communications internes 60  
comportement TCP/IP par défaut,  
remplacement 62  
concaténations de groupe 130  
configuration, identique par sysplex 165  
configuration de groupe LDAP  
initiale 136  
configuration du moniteur de travaux JES  
GEN\_CONSOLE\_NAME 29  
configuration identique par sysplex 165  
configuration RSE, Clonage de  
l'existant 195  
configurations Developer for System z  
multiples, utilisation de fichiers  
ISPF.conf multiples avec 164  
connexion de la configuration de l'hôte  
SSL, test 197  
connexion refusée 189

- consignation, test du programme de vérification de l'installation fekfivpi 176
- consignation dans le journal d'audit, géré par le démon RSE 24
- consignation des messages d'installation des ressources CICS 145
- consignation des messages de l'installation, ressources CICS 145
- consignation des messages de l'installation des ressources, CICS 145
- consignation des tests, programme de vérification de l'installation fekfivpc 176
- consignation des tests, programme de vérification de l'installation fekfivpi 176
- consignation des tests de la procédure de vérification d'installation fekfivps.log 177
- consignation des tests du programme de vérification de l'installation (IVP) fekfivpi.log 176
- consignation des tests du programme de vérification de l'installation (IVP) fekfivpi fekfivpi.log 176
- consignation des tests du programme de vérification de l'installation fekfivpc fekfivpc.log 176
- contrôle, réseau 111
- contrôle d'audit
  - \_RSE\_HOST\_CODEPAGE 24
  - audit.\* options 24
  - daemon.log 24
  - enable.audit.log 24
- contrôle d'autorisations d'accès, amélioration des performances de 119
- contrôle de la configuration client 128
- contrôle de la version client 129
- contrôle de RSE 108
- contrôle de z/OS UNIX 109
- contrôle du système de fichiers z/OS UNIX 111
- CRD, sécurité du référentiel 145

## D

- débogage, sécurité 38
- débogage, transaction CICS 154
- débogage de transaction CICS 154
- débogueur intégré 10
  - communication chiffrée 31
- définition de fichiers contrôlés par programme z/OS UNIX pour RSE 57
- définition de la prise en charge de PassTicket pour RSE 49
- définition de la vérification du port d'entrée pour RSE 35
- définition de ressource, différentes 106
- définition de sécurité 140
- définition des bibliothèques contrôlées par un programme MVS pour RSE 48
- définition des objectifs, WLM 73
- définition du serveur RSE en tant que serveur z/OS UNIX sécurisé 48
- définitions, sécurité 43

- définitions de ressource CICS, administrateur 143
- définitions de ressource CICS, développeur de logiciel 143
- définitions de ressource différentes 106
  - carte EXEC, JCL serveur 106
  - FEJJCENFG 106
  - SYS1.PARMLIB(ASCHPMxx) 107
  - SYS1.PARMLIB(IEASYSxx) 107
  - SYS1.PARMLIB(IVTPRMxx) 107
- définitions de ressources essentielles 102
  - rsed.envvars 102
  - SYS1.PARMLIB(BPXPRMxx) 103
- définitions de sécurité 43
- définitions de sécurité, Liste de contrôle 43
- définitions disponibles pour le programme de résolution 211
- définitions locales disponibles pour le programme de résolution 211
- délai de grâce, rejet des modifications 141
- démarrage rapide, option Java (-Xquickstart) 120
- démon lock 13
- démon Lock (LOCKD) 4
- démon RSE 60
- démon RSE, authentification par 34
- démon RSE (RSED) 4
- démon RSE et consignation dans le journal d'audit 24
- dépendance, nom d'hôte 205
- dépendance au nom d'hôte 205
- description de Developer for System z 3
- Developer for System z, description 3
- Developer for System z, présentation du composant
  - représentation graphique 3
- développement, applications 118
- développement d'applications 118
- différents fichiers de configuration avec des niveaux de logiciels identiques 166
- Distributed Dynamic VIPA
  - EZBEPOR 64
  - PORT 64
  - PORTRANGE 64
  - SERVERWLM 64
  - SYSPLXPORTS 64
  - VIPADISTRIBUTE 64
- données d'audit
  - actions consignées 25
- données de droits, UNIX z/OS 182
- données de rappel, disponibilité du module de chargement sur z/OS UNIX 185
- droits de mise à jour, administrateur non-système 17

## E

- éditeur de définition de ressources CICS (CRD), gestionnaire de déploiement d'application 143
- emplacement de serveur LDAP 135
- emplacement des métadonnées 127
- emplacement des métadonnées de groupe 131

- emplacements des fichiers de vidage, z/OS UNIX 179
- emplacements des fichiers de vidage UNIX 179
- emplacements des fichiers de vidage z/OS UNIX 179
- emploi de STEPLIB, Eviter 117
- émulateur, connexion à l'hôte 189
- émulateur de connexion à l'hôte 189
- environnement TSO,
  - personnalisation 161
- environnement UNIX, ordres de recherche dans 207
- environnement UNIX z/OS, ordres de recherche dans 207
- envoi au client 37
- envoi au client, métadonnées 127
- envoi au client, remarques sur la fonction 125
- erreur liée à une insuffisance de mémoire 189
- espace adresse, limite de taille 93
- espace de métadonnées, utilisation 128
- espace de stockage, analyse d'utilisation 95
- espace disque, machines virtuelles Java (JVM) 122
- étapes de configuration 132
- exécution de plusieurs instances 165
- exemple de configuration 112
  - compte de pools d'unités d'exécution 112
  - définition des limites 113
  - détermination des limites minimales 112
- exemple de configuration, sélection de groupe basé sur SAF 139
- exemple de configuration, sélection de groupe LDAP 135
- exigences, langage JCL de démarrage 187
- exigences liées au langage JCL, démarrage 187
- exigences liées au langage JCL lors du démarrage 187
- exit utilisateur, messages de console 158

## F

- fa.log 172
- FEJJCENFG 60, 113, 169
  - CONSOLE\_NAME 28
  - MAX\_THREADS 106
- FEJJCENFG, moniteur de travaux JES 40
- FEKAPPL 20
- fekfivpc.log 173
- fekfivpi.log 173
- fekfivpi.log, consignation des tests du programme de vérification d'installation 176
- fekfivps.log 173
- fekfivps.log, consignation des tests de la procédure de vérification d'installation 177
- FEKLOGS, journal et analyse de configuration 171
- FEKRACE, définitions de sécurité 43

- fekrivp 184
- ffs.log 172
- ffsget.log 172
- ffsput.log 172
- fichier de clés, Création avec RACF 193
- Création 203
- fichiers contrôlés par programme UNIX pour RSE, Définition 57
- fichiers contrôlés par programme z/OS UNIX pour RSE, Définition 57
- fichiers de configuration, Developer for System z 40
- fichiers de configuration, Niveau de logiciels identique, différent 166
- fichiers de configuration, programme de résolution de base 207
- fichiers de configuration du programme de résolution de base 207
- fichiers de vidage 177
- fichiers de vidage, Java 178
- fichiers de vidage, MVS 177
- fichiers de vidage Java 178
- fichiers de vidage MVS 177
- fichiers ISPF.conf, utilisation multiple avec configurations 164
- fichiers ISPF.conf multiples 164
- fichiers journaux
  - .dstoreMemLogging 172
  - .dstoreTrace 172
  - audit.log 172
  - fa.log 172
  - fekfivpi.log 172
  - fekfivps.log 172
  - ffs.log 172
  - ffsget.log 172
  - ffsput.log 172
  - lock.log 172
  - rmt\_class\_loader.cache.jar 172
  - rsecomm.log 172
  - rsedaemon.log 172
  - rseserver.log 172
  - serverlogs.count 172
  - stderr.log 172
  - stdout.log 172
- fichiers journaux du démon RSE
  - audit.log 174
  - rsedaemon.log 174
  - rseserver.log 174
  - serverlogs.count 174
  - stderr\*.log 174
  - stdout\*.log 174
- fichiers journaux du pool d'unités d'exécution RSE
  - audit.log 174
  - rsedaemon.log 174
  - rseserver.log 174
  - serverlogs.count 174
  - stderr\*.log 174
  - stdout\*.log 174
- fichiers spoule, accès conditionnel aux 29
- flux de connexion 8
- représentation graphique 8
- flux du démon lock
- représentation graphique 13

- fonction d'envoi au client, ajout d'une section dorsale à LDAP 136
- fonction de retardement d'accusé de réception 62
- fonction de trace, moniteur de travaux JES 180
- fonction de trace du moniteur de travaux JES 180
- fonctions client, modification 35

## G

- GATE, mise en corbeille 39
- gestion, charge de travail 119
- gestion de la charge de travail 119
- gestionnaire de débogage, authentification 21
- gestionnaire de déploiement d'application (ADM) 4
- gestionnaire de déploiement d'application, éditeur de définition de ressource CICS 143
- gestionnaire de déploiement d'application, personnalisation 143
- gestionnaire de déploiement d'application, sécurité 145
- gestionnaire de déploiement d'application, serveur de définition de ressource CICS 143
- groupes LDAP, ajout de développeurs 137
- gskkyman, Création d'une base de données de clés avec 200

## I

- ID utilisateur, variable, exécution avec 158
- ID utilisateur variable, exécution avec 158
- IEASYsxx 114
- MAXUSER 107, 189
- informations de configuration, ordres de recherche de 206
- insuffisance de mémoire, erreur 189
- intégré, débogueur 10
- interface de service Web 144
- interface RESTful 144
- interface RESTful par opposition à l'interface de service Web 144
- interrogation d'une liste de révocation de certificat (CRL)
  - rsed.envvars 33
  - variables d'environnement CRL 33
- introduction, remarques relatives à la fonction d'envoi au client 125
- ISPSISPLOAD
  - ISPF TSO/ISPF, passerelle client 48
- ISPF, Utilisation de plusieurs commandes exec d'allocation 163
- ISPF.conf, Personnalisation de base 162
- ISPF TSO/ISPF, passerelle client
  - ISPSISPLOAD 48
- IVTPRMxx
  - ECSA MAX 107
  - FIXED MAX 107

## J

- Java, limite de taille de pile 93
- JAVA\_DUMP\_TDUMP\_PATTERN 178
- JES JMON
  - GEN\_CONSOLE\_NAME 29
- JMON 51, 169
- journal et analyse de configuration à l'aide de FEKLOGS 171
- journalisation, CARMA 176
- journalisation, couverture de code 177
- journalisation, démon RSE 173
- journalisation, moniteur de travaux JES 173
- journalisation, pool d'unités d'exécution 173
- journalisation, révision du code 177
- journalisation, SCLM Developer Toolkit 175
- journalisation, utilisateur RSE 174
- journalisation CARMA
  - rsecomm.log 176
- journalisation de la couverture de code 177
- journalisation de la révision du code 177
- journalisation du moniteur de travaux JES 173
- journalisation du pool d'unités d'exécution 173
- journalisation pour l'utilisateur, RSE 174
- journalisation pour l'utilisateur RSE
  - .dstoreMemLogging 174
  - .dstoreTrace 174
  - ffs.log 174
  - ffsget.log 174
  - ffsput.log 174
  - lock.log 174
  - rmt\_class\_loader.cache.jar 174
  - rsecomm.log 174
  - stderr.log 174
  - stdout.log 174
- journalisation pour le Common Access Repository Manager 176
- journalisation pour le démon RSE 173
- JVM, partage de classes entre 120

## K

- keytool, Création d'un fichier de clés avec 203

## L

- LDAP, configuration de groupe
  - initiale 136
- liaison d'espace de travail 131
- libération d'un verrou
  - RSE, commande d'annulation de modification 14
- LIMIT\_COMMANDS 27
- LIMIT\_VIEW 29
- limitation de communication externe, ports spécifiés 22
- limitations d'exécution, actions sur les travaux 28
- limite de taille, espace adresse 93

- limite de taille de pile, Java 93
- limites, système 188
- limites de taille de la mémoire cache, machines virtuelles Java (JVM) 121
- limites du système 188
- liste de révocation de certificat (CRL), interrogation
  - rsed.envvars 33
  - variables d'environnement CRL 33
- lock.log 172
- logiciel de sécurité, authentification par 33
- logon.action, exit utilisateur 160
- LPALSTxx 118

## M

- machines virtuelles Java (JVM), partage de classes entre 120
- messages, utilitaire d'administration 152
- messages de console, exit utilisateur 158
- messages de l'utilitaire d'administration 152
- métadonnées d'envoi au client 127
- métadonnées de sécurité 127
- méthode d'accès, Utilisation de la passerelle client TSO/ISPF 162
- méthode d'accès par passerelle client, Utilisation de TSO/ISPF 162
- méthode d'accès par passerelle client TSO/ISPF, Utilisation 162
- méthodes, authentification 20
- méthodes d'accès, TSO 161
- méthodes d'accès TSO 161
- méthodes d'authentification 20
- mode de passe utilisable une seule fois et ID utilisateur 20
- moniteur de travaux JES, FEJJCNFG 40
- moniteur de travaux JES (JMON) 4
- mots de passe et ID utilisateur 20

## N

- netstat 185
- niveau de logiciels, identique dans des fichiers de configuration différents 166
- niveaux de logiciels identiques avec des fichiers de configuration différents 166
- nombre d'espaces adresse 83
- nombre d'unités d'exécution 89
- nombre de processus 86
- noms d'hôte, application à Developer for System z 209
- notes de migration, utilitaire d'administration 151

## O

- objectifs, définition dans WLM 73
- OFF.REMOTECOPY.MVS 36
- option Java Xquickstart 120
- ordres de recherche, environnement UNIX z/OS 207
- ordres de recherche d'informations de configuration 206

## P

- paramètres de sécurité, vérification 58
- paramètres et classes, activation de la sécurité 45
- paramètres et classes de sécurité, activation 45
- partage de classes, activation dans des machines virtuelles Java (JVM) 121
- partage de classes entre machines virtuelles Java (JVM) 120
- PassTickets, utilisation 23
- performances du contrôle d'autorisations d'accès, amélioration 119
- personnalisation - ISPF.conf, 162
- personnalisation de l'environnement TSO 161
- personnalisation du gestionnaire de déploiement d'application 143
- pile Java, limite de taille 93
- pipeline,sécurité 145
- plusieurs commandes exec d'allocation, TSO/ISPF 163
- plusieurs groupes de développeurs 129
- plusieurs instances, exécution 165
- points d'exit, disponibles 160
- points d'exit utilisateur, disponibles 160
- port d'entrée, vérification 22, 35
- port TCP/IP, réservation 61
- PORTRANGE 186
- ports, CARMA et TCP/IP 61
- ports, TCP/IP 59
- ports, TCP/IP réservés 185
- ports spécifiques, limitation de communication externe à des 22
- ports TCP/IP 59
- ports TCP/IP, représentation graphique 59
- ports TCP/IP, réservés 185
- ports TCP/IP réservés 185
- présentation du composant, Developer for System z
  - représentation graphique 3
- principal par opposition à non-principal, régions de connexion 144
- profil de sécurité, Limitations stockées 187
- profils, définition de fichier 53
- profils de fichier, définition 53
- profils ISPF, Utilisation de l'existant 162
- programme de résolution, définitions locales disponibles pour 211
- programme de résolution TCP/IP, adresse hôte non résolue
  - lock.log 211
- programmes de résolution, Présentation 206
- projets, résidant sur l'hôte 141
- projets résidant sur l'hôte 141
- propriétaires de tâches 7
- protection, ressources 147
- protection d'application pour RSE, Définition 50
- publications, Références 213
- Publications référencées 213
- pushtoclient.properties 137, 140

## R

- RACF
  - permis 54
- RACF, Création d'un fichier de clés avec 193
- référentiel de CRD 39
- région gérant le Web 144
- régions de connexion, principal par opposition à non-principal 144
- règles de classification, WLM 72
- rejet des modifications, délai de grâce 141
- remarques, performances 117
- remarques, sécurité 19
- remarques à propos de WLM xvi, 71
- remarques relatives à CICSTS 143
- remarques relatives à l'optimisation 81
- remarques relatives à la sécurité 19
- remarques relatives à LDAP 62
- remarques relatives aux exits
  - utilisateur xvii, 157
- remarques relatives aux performances 117
- remplacement du comportement TCP/IP par défaut 62
- réseau, contrôle 111
- réservation, port TCP/IP 61
- réservation de port TCP/IP 61
- ressources, protection 147
- ressources temporaires, utilisation 92
- retardement d'accusé de réception 62
- rmt\_class\_loader\_cache.jar 172
- routine d'exit utilisateur, création 157
- RSE, contrôle 108
- RSE, définition d'un serveur z/OS UNIX sécurisé 48
- RSE, définition de fichiers contrôlés par programme z/OS UNIX pour 57
- RSE, définition de la protection d'application pour 50
- RSE, définition de la vérification du port d'entrée pour 35
- RSE, définition des bibliothèques contrôlées par programme MVS 48
- RSE, définition du support PassTicket pour 49
- RSE, pushtoclient.properties 42
- RSE, rsed.envvars
  - \_RSE\_JAVAOPTS 41
- RSE, ssl.properties 42
- RSE, traçage 180
- RSE comme application Java
  - représentation graphique 5
- rsecomm.log 172
  - SCLM Developer Toolkit, journalisation 175
- rsecomm.properties 181
- rsed.envvars 101, 138, 140, 167
  - \_CMDSEV\_CONF\_HOME 164
  - \_RSE\_JAVAOPTS 161, 178
  - \_RSE\_PORTRANGE 22
  - Dmaximum.clients 102
  - Dmaximum.threadpool.process 102
  - Dmaximum.threads 102
  - Dminimum.threadpool.process 102
  - DSTORE\_LOG\_DIRECTORY 176, 180
  - STEPLIB 30



rsed.envvars (*suite*)  
 Xms 102  
 Xmx 102  
 rsed.envvars, mise à jour pour assurer la  
 coexistence 195  
 rsedaemon.log 172, 173  
 rserver.log 172, 173

## S

schéma LDAP 134  
 SCLM, sécurité 40  
 SCLM Developer Toolkit 49  
 SCLM Developer Toolkit, journalisation  
 rsecomm.log 175  
 SCLM Developer Toolkit (SCLMDT) 4  
 script de shell z/OS UNIX 158  
 Secure Socket Layer, chiffrement des  
 communications à l'aide de 22  
 Secure Socket Layer, Configuration 191  
 sécurité, CICS 38  
 sécurité, connexion 21  
 sécurité, Définition des commandes  
 JES 51  
 sécurité, gestionnaire de déploiement  
 d'application (ADM) 145  
 sécurité, JES 26  
 sécurité, pipeline 145  
 sécurité, SCLM 40  
 sécurité, transaction 145  
 sécurité de l'unité d'exécution dans un  
 serveur RSE  
 PassTickets 23  
 sécurité de la mémoire cache, machines  
 virtuelles Java (JVM) 121  
 sécurité des commandes, Définition  
 JES 51  
 sécurité des commandes JES,  
 Définition 51  
 sécurité des connexions 21  
 sécurité du débogage 38  
 sécurité du référentiel, CRD 145  
 sécurité JES 26  
 segment, définition OMVS 46  
 segment OMVS, définition 46  
 sélection de groupe, basé sur LDAP 133  
 sélection de groupe, basé sur SAF 138  
 sélection de port, restriction 65  
 sélection de serveur LDAP 134  
 serverlogs.count 172  
 serveur, sélection LDAP 134  
 serveur de définition de ressources CICS  
 (CRD), gestionnaire de déploiement  
 d'application 143  
 serveur LDAP, emplacement 135  
 serveur RSE 60  
 serveur UNIX, définition de RSE en tant  
 que 48  
 serveur z/OS UNIX, définition de RSE en  
 tant que 48  
 serveur z/OS UNIX sécurisé, définition  
 de RSE en tant que 48  
 service Commandes TSO 4, 161  
 SETUID, attribut du système de  
 fichiers 182  
 SMP/E - installation, données de  
 rappel 185

sorties du système, Limitations  
 forcées 188  
 SSL, chiffrement 192  
 SSL, chiffrement des communications à  
 l'aide de 22  
 SSL, configuration 191  
 ssl.properties, activation de SSL en créant  
 un démon RSE 196  
 ssl.properties, Activation du protocole  
 SSL via la mise à jour 196  
 SSL/TLS, communication chiffrée 30  
 stderr.\*.log 172  
 stderr.log 172  
 stdout.\*.log 172  
 stdout.log 172  
 STEPLIB, Eviter l'emploi de 117  
 structure de répertoire, z/OS UNIX  
 représentation graphique 15  
 structure de répertoire z/OS UNIX  
 représentation graphique 15  
 support d'authentification du client, ajout  
 de X.509 200  
 support PassTicket pour RSE,  
 Définition 49  
 support pour RSE, définition du  
 PassTicket 49  
 synchronisation, automatisée 167  
 synchronisation automatisée 167  
 SYS1.PARMLIB(BPXPRMxx) 113  
 MAXASSIZE 47, 187  
 MAXPROCSYS 189  
 MAXPROCUSER 189  
 MAXUIDS 189  
 SYS1.PARMLIB(BPXPRMxx), Limitations  
 définies 187  
 SYS1.PARMLIB(BPXPRMxx), machines  
 virtuelles Java (JVM) 122  
 SYS1.PARMLIB(IEASYSxx) 114  
 MAXUSER 189  
 sysplex, configuration identique par 165  
 système de fichiers zFS, utilisation 117  
 système principal 126  
 systèmes de fichiers, zFS 117  
 systèmes de fichiers z/OS UNIX,  
 contrôle 111

## T

tables, conversion 208  
 tables, hôte local 208  
 tables de conversion 208  
 tables de système hôte local 208  
 tables hôte, local 208  
 tâches démarrées, définir pour Developer  
 for System z  
 tâches démarrées JMON 46  
 tâches démarrées RSED 46  
 tâches démarrées Developer for System z,  
 définir 46  
 taille, espace adresse 187  
 taille d'espace adresse 187  
 taille de pile Java, fixe 119  
 taille de pile Java fixe 119  
 taille estimée, instructions 94  
 TCP/IP, application à Developer for  
 System z 209  
 TCP/IP, Configuration 205

TCP/IP, définitions locales disponibles  
 pour le programme de résolution 211  
 TCP/IP, remplacement du comportement  
 par défaut 62  
 test de la connexion de la configuration  
 de l'hôte SSL 197  
 tiers et certificat X.509 20  
 tincidents liés à la configuration,  
 traitement 171  
 TLS, chiffrement 192  
 TLS, chiffrement des communications à  
 l'aide de 22  
 traçage 180  
 traçage, CARMA 181  
 traçage, RSE 180  
 traçage, suivi des erreurs 181  
 traçage de suivi, erreur 181  
 traçage de suivi des erreurs 181  
 traitement de l'audit  
 modify switch 25  
 traitement des incidents liés à la  
 configuration 171  
 transaction, sécurité 145  
 transactions CICS 39  
 travaux, Actions conditionnelles sur 26  
 TSO/ISPF, personnalisation -  
 ISPF.conf, 162  
 TSO/ISPF, utilisation avec configurations  
 multiples 164  
 TSO/ISPF, utilisation d'une commande  
 exec d'allocation 163  
 TSO/ISPF, Utilisation de plusieurs  
 commandes exec d'allocation 163  
 TSO/ISPF, Utilisation des profils ISPF  
 existants 162  
 types de sous-système  
 ASCH 72  
 CICS 72  
 JES 72  
 OMVS 72  
 STC 72

## U

un ID utilisateur et un mot de passe 20  
 un ID utilisateur et un mot de passe  
 utilisable une seule fois 20  
 UNIX z/OS, données de droits 182  
 utilisation d'une commande exec  
 d'allocation 163  
 utilisation de l'espace, système de fichiers  
 z/OS UNIX 99  
 utilisation de l'espace de  
 métadonnées 128  
 utilisation de l'espace de stockage 93  
 utilisation de l'espace du système de  
 fichiers, z/OS UNIX 99  
 utilisation de l'espace du système de  
 fichiers z/OS UNIX 99  
 utilisation de PassTickets 23  
 utilisation des profils ISPF existants 162  
 utilisation des ressources,  
 optimisation 81  
 utilisation des ressources,  
 présentation 82  
 utilisation des ressources temporaires 92

- utilitaire d'administration, notes de migration 151
- utilitaire d'administration pour des administrateur CICS
  - fonctions fournies 147
- utilitaires de gestion de la mémoire
  - cache, machines virtuelles Java (JVM) 122

## V

- validation de l'autorité de certification
  - fichier de clés SAF 32
  - gskkyman 32
  - TRUST, HIGHTRUST 32
- variables de modèle de cliché de transaction 178
- vérification des paramètres de sécurité 58
- vérification du port d'entrée 22
- vérification du port d'entrée (POE) 35
- VIPA, dynamique distribué 64

## W

- WLM, règles de classification 72
- workload manager 71

## X

- X.509, ajout du support d'authentification du client 200
- x.509, configuration de l'authentification 191
- Xquickstart, option Java 120

## Z

- z/OS UNIX, contrôle 109





SC11-6869-06

