

IBM Rational Developer for z Systems  
Version 9.5.1

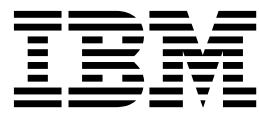
## *Hostkonfigurationsreferenz*





IBM Rational Developer for z Systems  
Version 9.5.1

## *Hostkonfigurationsreferenz*



**Hinweis**

Vor Verwendung dieser Informationen sollten die allgemeinen Informationen unter „Bemerkungen“ auf Seite 63 gelesen werden.

**Zehnte Ausgabe (September 2015)**

- | Diese Ausgabe bezieht sich auf IBM Rational Developer for z Systems Version 9.5 (Programmnummer 5724-T07
- | oder einen Teil von Programmnummer 5697-CDT) und - sofern in neuen Ausgaben nicht anders angegeben - auf
- | alle nachfolgenden Releases und Modifikationen.

Diese Veröffentlichung ist eine Übersetzung des Handbuchs  
*IBM Rational Developer for z Systems Version 9.5 Host Configuration Reference Guide*,  
IBM Form SC27-8578-00,  
herausgegeben von International Business Machines Corporation, USA

(C) Copyright International Business Machines Corporation 2000, 2015

Informationen, die nur für bestimmte Länder Gültigkeit haben und für Deutschland, Österreich und die Schweiz nicht zutreffen, wurden in dieser Veröffentlichung im Originaltext übernommen.

Möglicherweise sind nicht alle in dieser Übersetzung aufgeführten Produkte in Deutschland angekündigt und verfügbar; vor Entscheidungen empfiehlt sich der Kontakt mit der zuständigen IBM Geschäftsstelle.

Änderung des Textes bleibt vorbehalten.

Herausgegeben von:  
TSC Germany  
Kst. 2877  
Dezember 2015

© Copyright IBM Corporation 2015, 2015.

# Inhalt

<b>Abbildungen . . . . .</b>	<b>v</b>
------------------------------	----------

<b>Tabellen . . . . .</b>	<b>vii</b>
---------------------------	------------

<b>Zu diesem Handbuch . . . . .</b>	<b>ix</b>
-------------------------------------	-----------

Zielgruppe . . . . .	x
Zusammenfassung der Änderungen . . . . .	x
Beschreibung der Dokumentinhalte . . . . .	xii
Wissenswertes zu Developer for z Systems . . . . .	xii
Sicherheitsaspekte . . . . .	xii
Hinweise zu TCP/IP . . . . .	xii
Hinweise zu WLM . . . . .	xii
Push-to-Client-Aspekte . . . . .	xiii
CICSTS-Aspekte . . . . .	xiii
AT-TLS konfigurieren . . . . .	xiii

<b>Developer for System z Hostkonfigurationsreferenz . . . . .</b>	<b>1</b>
--	----------

<b>Kapitel 1. Wissenswertes zu Developer for z Systems . . . . .</b>	<b>3</b>
--	----------

Komponentenübersicht . . . . .	3
Taskeigner . . . . .	4
Integrated Debugger . . . . .	6
CARMA . . . . .	7
CARMA-Konfigurationsdateien . . . . .	8
CRASTART . . . . .	8
Batchübergabe . . . . .	8
z/OS UNIX-Verzeichnisstruktur . . . . .	9

<b>Kapitel 2. Sicherheitsaspekte . . . . .</b>	<b>11</b>
--	-----------

Authentifizierungsmethoden . . . . .	11
Debug Manager-Authentifizierung . . . . .	11
Verbindungssicherheit . . . . .	12
Mit Integrated Debugger verschlüsselte Kommunikation . . . . .	12
Debug-Sicherheit . . . . .	13
CICSTS-Sicherheit . . . . .	13
SCLM-Sicherheit . . . . .	14
Sicherheitsdefinitionen . . . . .	14
Voraussetzungen und Prüfliste . . . . .	14
Sicherheitseinstellungen und -klassen aktivieren OMVS-Segment für Benutzer von Developer for z Systems definieren . . . . .	15
Gestartete Tasks für Developer for z Systems definieren . . . . .	16
Debug Manager als sicheren z/OS UNIX-Server definieren . . . . .	17
Programmgesteuerte MVS-Bibliotheken für Debug Manager definieren . . . . .	17
PassTicket-Unterstützung für RSE definieren . . . . .	18
z/OS UNIX-Dateizugriffsberechtigung für RSE definieren . . . . .	19

Anwendungsschutz für RSE definieren . . . . .	19
Programmgesteuerte z/OS UNIX-Dateien für RSE definieren . . . . .	20
JES-Befehlssicherheit definieren . . . . .	20
Zugriff auf Integrated Debugger definieren . . . . .	22
Dateiprofile definieren . . . . .	22
Sicherheitseinstellungen prüfen . . . . .	23

<b>Kapitel 3. Hinweise zu TCP/IP . . . . .</b>	<b>25</b>
--	-----------

TCP/IP-Ports . . . . .	25
Externe Kommunikation . . . . .	25
Interne Kommunikation . . . . .	26
TCP/IP-Portreservierung . . . . .	27
<b>CARMA und TCP/IP . . . . .</b>	<b>27</b>
CARMA und TCP/IP-Ports . . . . .	27
CARMA und Stackaffinität . . . . .	28
crastart*.conf . . . . .	28
CRASUB* . . . . .	28

<b>Kapitel 4. Hinweise zu WLM . . . . .</b>	<b>31</b>
---	-----------

Klassifikation für Verarbeitungsprozesse . . . . .	31
Klassifikationsregeln . . . . .	32
Ziele festlegen . . . . .	33
Hinweise zur Zielauswahl . . . . .	34
STC . . . . .	35
OMVS . . . . .	35
JES . . . . .	36

<b>Kapitel 5. Push-to-Client-Aspekte . . . . .</b>	<b>39</b>
--	-----------

Einführung . . . . .	39
Hostbasierte Projekte . . . . .	40

<b>Kapitel 6. CICSTS-Aspekte . . . . .</b>	<b>41</b>
--	-----------

Unterstützung bidirektionaler Sprachen . . . . .	41
IRZ-Diagnosenachrichten für Enterprise Service Tools . . . . .	41
CICS-Transaktionsdebugging . . . . .	41

<b>Kapitel 7. AT-TLS konfigurieren . . . . .</b>	<b>43</b>
--	-----------

syslogd konfigurieren . . . . .	44
AT-TLS-Konfiguration in PROFILE.TCPIP . . . . .	44
Gestartete Task von Policy Agent . . . . .	45
Konfiguration von Policy Agent . . . . .	45
AT-TLS-Richtlinie . . . . .	46
Hinweise zu TLS V1.2 . . . . .	48
AT-TLS-Sicherheitsupdates . . . . .	48
Aktivierung der AT-TLS-Richtlinie . . . . .	51

<b>Literaturübersicht . . . . .</b>	<b>53</b>
-------------------------------------	-----------

Referenzierte Veröffentlichungen . . . . .	53
Veröffentlichungen mit weiteren Informationen . . . . .	54

<b>Glossar . . . . .</b>	<b>57</b>
--------------------------	-----------

<b>Bemerkungen. . . . .</b>	<b>63</b>
-----------------------------	-----------

Informationen zu Programmierschnittstellen . .	65
Marken. . . . .	65
Nutzungsbedingungen für die Produktdokumentation . . . . .	65

Copyrightlizenz . . . . .	66
Marken. . . . .	66

<b>Index . . . . .</b>	<b>67</b>
------------------------	-----------

---

## Abbildungen

1.	Komponentenübersicht . . . . .	3	5.	z/OS UNIX-Verzeichnisstruktur . . . . .	9
2.	Taskeigner . . . . .	5	6.	AT-TLS-Richtlinie für Debug Manager . . .	12
3.	Integrated Debugger . . . . .	6	7.	TCP/IP-Ports . . . . .	25
4.	CARMA-Flow . . . . .	7	8.	WLM-Klassifikation. . . . .	31





---

## Tabellen

1. SAF-Informationen für Debugfunktionen	13		7. WLM-Verarbeitungsprozesse . . . . .	34
2. Variablen der Sicherheitskonfiguration . ..	14		8. WLM-Verarbeitungsprozesse - STC. . . ..	35
3. Bedienerbefehle von JES2 Job Monitor . ..	21		9. WLM-Verarbeitungsprozesse - OMVS . . ..	35
4. Bedienerbefehle von JES3 Job Monitor . ..	21		10. WLM-Verarbeitungsprozesse - JES . . . ..	36
5. WLM-Einstiegspunkt-Subsysteme . . . . .	32		11. Referenzierte Veröffentlichungen . . . . .	53
6. WLM-Qualifikationsmerkmale für			12. Referenzierte Websites . . . . .	54
Arbeitsvorgänge . . . . .	33		13. Veröffentlichungen mit weiteren Informationen	54



---

## Zu diesem Handbuch

Dieses Dokument enthält Hintergrundinformationen zu verschiedenen Konfigurationstasks von IBM® Rational Developer for z Systems selbst sowie zu anderen z/OS-Komponenten und -Produkten (wie WLM und TCP/IP).

Im weiteren Verlauf dieses Handbuchs werden die folgenden Namen verwendet:

- *IBM Explorer for z/OS* wird als *z/OS Explorer* bezeichnet.
- *IBM Rational Developer for z Systems* wird als *Developer for z Systems* bezeichnet.
- *IBM Rational Developer for z Systems Integrated Debugger* wird als *Integrated Debugger* bezeichnet.
- *Common Access Repository Manager* wird mit *CARMA* abgekürzt.
- *Software Configuration and Library Manager Developer Toolkit* wird als *SCLM Developer Toolkit* bezeichnet und mit *SCLMDT* abgekürzt.
- *z/OS UNIX System Services* wird als *z/OS UNIX* bezeichnet.
- *Customer Information Control System Transaction Server* wird als *CICSTS* bezeichnet und mit *CICS* abgekürzt.

Dieses Dokument ist Teil einer Reihe von Dokumenten, in denen die Hostkonfiguration von Developer for z Systems beschrieben wird. Jedes dieser Dokumente hat eine spezielle Zielgruppe. Sie müssen nicht alle Dokumente lesen, um die Konfiguration von Developer for z Systems abzuschließen.

- Im Handbuch *IBM Rational Developer for z Systems Hostkonfiguration* (IBM Form SC43-2896) werden alle Planungstasks, Konfigurationstasks und Optionen (einschließlich der optionalen) ausführlich beschrieben und alternative Szenarien bereitgestellt.
- In *IBM Rational Developer for z Systems Hostkonfigurationsreferenz* (SC43-2898) wird das Design von Developer for z Systems beschrieben. Das Handbuch enthält außerdem Hintergrundinformationen für verschiedene Konfigurationstasks von Developer for z Systems, z/OS-Komponenten und weiteren Produkten (wie WLM und TCP/IP) in Verbindung mit Developer for z Systems.
- Im Handbuch *IBM Rational Developer for z Systems Leitfaden für den Schnelleinstieg in die Hostkonfiguration* (IBM Form GI11-3191) wird eine Minimalkonfiguration von Developer for z Systems beschrieben.

Die Informationen in diesem Dokument gelten für alle Pakete von IBM Rational Developer for z Systems Version 9.5.1.

Die aktuellsten Versionen dieses Dokuments finden Sie im Handbuch *IBM Rational Developer for z Systems Hostkonfigurationsreferenz* (SC43-2898) unter '<http://www-05.ibm.com/e-business/linkweb/publications/servlet/pbi.wss?CTY=US&FNC=SRX&PBL=SC43-2898>'.

Die aktuellsten Versionen der kompletten Dokumentation, einschließlich Installationsanweisungen, White Papers, Podcasts und Lernprogrammen, finden Sie auf der Bibliotheksseite der IBM Rational Developer for z Systems-Website ([http://www-01.ibm.com/software/sw-library/en\\_US/products/Z964267S85716U24/](http://www-01.ibm.com/software/sw-library/en_US/products/Z964267S85716U24/)).

---

## Zielgruppe

Dieses Handbuch wendet sich an Systemprogrammierer, die IBM Rational Developer for z Systems Version 9.5.1 konfigurieren und optimieren.

Während die eigentlichen Konfigurationsschritte in einer anderen Veröffentlichung beschrieben werden, werden in dieser Veröffentlichung verschiedene zugehörige Themen (wie Optimierung, Sicherheitskonfiguration usw.) ausführlich aufgelistet. Voraussetzung für die Verwendung dieses Handbuchs ist, dass Sie mit z/OS UNIX System Services und MVS-Hostsystemen vertraut sind.

---

## Zusammenfassung der Änderungen

In diesem Abschnitt werden die Änderungen für *IBM Rational Developer for z Systems Version 9.5.1 Hostkonfigurationsreferenz, SC43-2898-00* (Aktualisierung vom Dezember 2015) zusammengefasst.

Technische Änderungen oder Zusätze zum Text und den Abbildungen sind durch eine vertikale Linie auf der linken Seite der Änderung angegeben.

Neue Informationen:

- Verwenden Sie die neuen MVS-Datensatznamen und z/OS UNIX-Pfade

Entfernte Informationen:

In Version 9.5.1 wurden die zu RSE und JES Job Monitor gehörigen Funktionen von IBM Rational Developer for z Systems in ein anderes Produkt (IBM Explorer for z/OS) verschoben. Dies gilt auch für die zugehörige Dokumentation.

- Daten, die sich auf RSE beziehen, werden aus allen Kapiteln entfernt.
- Daten, die sich auf JES Job Monitor beziehen, werden aus allen Kapiteln entfernt.
- Daten, die sich auf den TSO-Befehlsservice beziehen, werden aus allen Kapiteln entfernt.
- Push-to-client-Daten für das Konfigurations- und Versionsmanagement wurden aus allen Kapiteln entfernt
- Dokumentation zur Einrichtung von TCP/IP wurde entfernt

Dieses Dokument enthält Informationen, die bisher im Handbuch *IBM Rational Developer for z Systems Version 9.5 Hostkonfigurationsreferenz* (IBM Form SC12-4489-09) enthalten waren.

Neue Informationen:

- Dokumentierte Sicherheitsprüfungen für die neue Funktion zum Senden von Nachrichten. Siehe Sicherheit beim Senden von Nachrichten
- Zusätzliche Details zu verwendeten JES-Bedienerbefehlen. Siehe Aktionen für Beschränkungen der Jobausführung
- Zusätzliche Informationen zu Begrenzungen für Push-to-Client-Gruppennamen. Siehe Einschränkungen für Gruppennamen
- Zusätzliche Informationen zu SYSPLEX-Einschränkungen. Siehe SYSPLEX
- Zusätzliche Informationen für das Verwalten von Verschlüsselungsprotokollen und Verschlüsselungen. Siehe Verschlüsselungsprotokolle und Verschlüsselungswerte verwalten

- Zusätzliche Anweisungen für eine einfache Konfiguration mit mehreren Servern. Siehe Identischer Software-Level, verschiedene Konfigurationsdateien

Entfernte Informationen:

- Application Deployment Manager wird nicht mehr bereitgestellt. Daher wurden sämtliche zugehörigen Informationen entfernt.

Dieses Dokument enthält Informationen, die bisher im Handbuch *IBM Rational Developer for System z Version 9.1.1 Hostkonfigurationsreferenz* (IBM Form SC12-4489-08) enthalten waren.

Neue Informationen:

- Aktualisierte Integrated Debugger-Sicherheitsprofile. Lesen Sie hierzu den Abschnitt „Debug-Sicherheit“ auf Seite 13.
- Hinzugefügte Informationen zur Unterstützung von Kennphrasen. Lesen Sie hierzu den Abschnitt „Authentifizierungsmethoden“ auf Seite 11.

Dieses Dokument enthält Informationen, die bisher im Handbuch *IBM Rational Developer for System z Version 9.1.1 Hostkonfigurationsreferenz* (IBM Form SC12-4489) enthalten waren.

Neue Informationen:

- Hinzugefügte Informationen zur Protokolldateisicherheit. Siehe Protokolldateisicherheit.
- Hinzugefügte Informationen zur Gruppenunterstützung für abgelehnte Push-to-Client-Aktualisierungen. Siehe Mehrere Entwicklergruppen.
- Aktualisierte Informationen zur Ressourcennutzung. Siehe Aspekte bei der Optimierung.
- Aktualisierte Protokolldatei- und Traceinformationen. Siehe Fehlerbehebung bei Konfigurationsproblemen.

Dieses Dokument enthält Informationen, die bisher im Handbuch *IBM Rational Developer for System z Version 9.0.1 Hostkonfigurationsreferenz* (IBM Form SC12-4489-06) enthalten waren.

Neue Informationen:

- Hinzugefügte Informationen zum Einrichten von AT-TLS. Lesen Sie hierzu den Abschnitt Kapitel 7, „AT-TLS konfigurieren“, auf Seite 43.

Dieses Dokument enthält Informationen, die bisher im Handbuch *IBM Rational Developer for System z Version 9.0.1 Hostkonfigurationsreferenz* (IBM Form SC12-4489-05) enthalten waren.

Neue Informationen:

- Hinzugefügte Informationen zu den Protokolldateinamen mit Zeitmarke. Siehe Protokolldateien.
- Hinzugefügte Informationen zu den neuen prüfbaren Ereignissen. Siehe Prüfdaten.

Dieses Dokument enthält Informationen, die bisher im Handbuch *IBM Rational Developer for System z Version 9.0 Hostkonfigurationsreferenz* (IBM Form SC12-4489) enthalten waren.

Neue Informationen:

- Aktualisierte Nutzung des TCP/IP-Ports. Lesen Sie hierzu den Abschnitt „TCP/IP-Ports“ auf Seite 25.
- Hinzugefügtes Beispiel für die automatische Synchronisierung von zwei RSE-Dämonen. Siehe Automatisierte Synchronisierung.
- Hinzugefügte Informationen zu neuen Protokolldateien. Siehe Protokolldateien.

Dieses Dokument enthält Informationen, die bisher im Handbuch *IBM Rational Developer for System z Version 8.5.1 Hostkonfigurationsreferenz* (IBM Form SC12-4489-03) enthalten waren.

Neue Informationen:

- Hinzugefügte Informationen über SAF-Profilen zum Ändern von Clientfunktionen. Siehe Clientfunktionen ändern.
- Aktualisierte Zahlen für Ressourcennutzung. Siehe Aspekte bei der Optimierung
- Aktualisierter Standardwert für maximale Anzahl von Benutzern pro Thread-Pool. Siehe Aspekte bei der Optimierung.

Dieses Dokument enthält Informationen, die bisher im Handbuch *IBM Rational Developer for System z Version 8.5 Hostkonfigurationsreferenz* (IBM Form SC12-4489-02) enthalten waren.

Neue Informationen:

- Aktualisierte JES Job Monitor-Sicherheitsinformationen. Lesen Sie hierzu den Abschnitt Kapitel 2, „Sicherheitsaspekte“, auf Seite 11.
- Hinzugefügte Informationen über Benutzerexits. Siehe Aspekte bei Benutzerexits.

---

## Beschreibung der Dokumentinhalte

In diesem Abschnitt werden die in diesem Dokument enthaltenen Informationen zusammengefasst.

### Wissenswertes zu Developer for z Systems

Der Host von Developer for z Systems umfasst einige interagierende Komponenten, damit der Client auf die Host-Services und -daten zugreifen kann. Wenn Sie das Design dieser Komponenten verstehen, können Sie die richtigen Konfigurationsentscheidungen treffen.

### Sicherheitsaspekte

Developer for z Systems interagiert mit anderen Hostkomponenten, was sich auf die Sicherheit auswirkt.

### Hinweise zu TCP/IP

Developer for z Systems verwendet TCP/IP, um Benutzern einer Workstation den Zugriff auf Mainframe-Computer bereitzustellen, wenn diese selbst kein Mainframe-Computer ist. TCP/IP wird außerdem für die Datenübertragung zwischen verschiedenen Komponenten und anderen Produkten verwendet.

### Hinweise zu WLM

Im Gegensatz zu herkömmlichen z/OS-Anwendungen ist Developer for z Systems keine einzelne Anwendung, die von Workload Manager (WLM) auf einfache Weise erkannt wird. Developer for z Systems umfasst mehrere interagierende Komponenten, damit der Client auf die Host-Services und -daten zugreifen kann.

Einige dieser Services sind in verschiedenen Adressräumen aktiv und werden somit verschiedenen WLM-Klassifikationen zugeordnet.

## **Push-to-Client-Aspekte**

Developer for z Systems erweitert den Push-to-Client von z/OS Explorer (der hostbasierten Clientsteuerung) mit der Unterstützung für Projektdefinitionen.

## **CICSTS-Aspekte**

Dieses Kapitel enthält nützliche Informationen für CICS Transaction Server-Administratoren.

## **AT-TLS konfigurieren**

Dieser Abschnitt soll Sie bei einigen allgemeinen Problemen unterstützen, die beim Konfigurieren von AT-TLS (Application Transparent Transport Layer Security) oder beim Überprüfen oder Modifizieren einer vorhandenen Konfiguration auftreten können.





---

## **Developer for System z Hostkonfigurationsreferenz**



# Kapitel 1. Wissenswertes zu Developer for z Systems

Der Host von Developer for z Systems umfasst einige interagierende Komponenten, damit der Client auf die Host-Services und -daten zugreifen kann. Wenn Sie das Design dieser Komponenten verstehen, können Sie die richtigen Konfigurationsentscheidungen treffen.

Dieses Kapitel enthält die folgenden Abschnitte:

- „Komponentenübersicht“
- „Taskeigner“ auf Seite 4
- „Integrated Debugger“ auf Seite 6
- „CARMA“ auf Seite 7
- „z/OS UNIX-Verzeichnisstruktur“ auf Seite 9

Developer for z Systems baut auf IBM Explorer for z/OS auf. z/OS Explorer-spezifische Informationen finden Sie in „Aspekte bei der Sicherheit“ (“Security consideration”) im Handbuch *IBM Explorer for z/OS Host Configuration Reference* (IBM Form SC27-8438).

## Komponentenübersicht

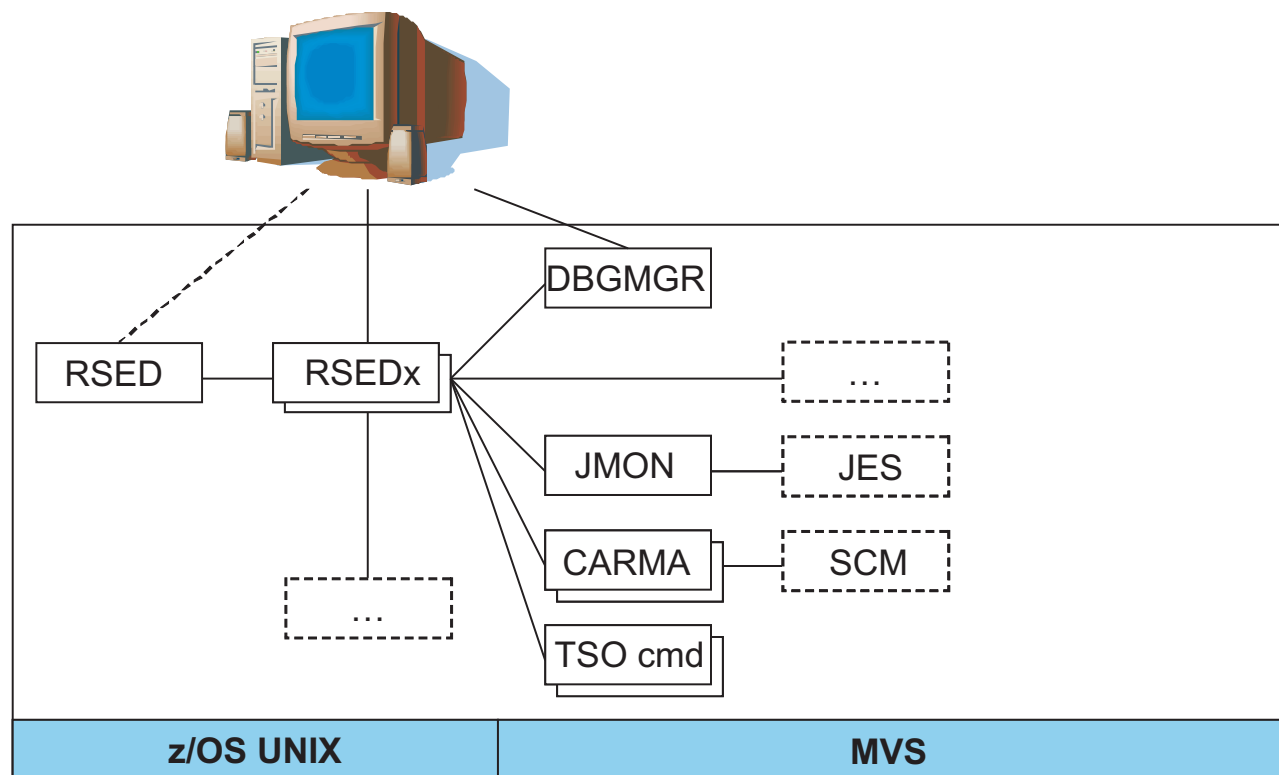


Abbildung 1. Komponentenübersicht

Abb. 1 zeigt eine allgemeine Übersicht des kombinierten Layouts von z/OS Explorer und Developer for z Systems auf Ihrem Hostsystem.

- Remote Systems Explorer (RSE) stellt Kernservices wie den Verbindungsaufbau vom Client zum Host und das Starten anderer Server für bestimmte Services bereit. RSE umfasst zwei logische Einheiten:
  - RSE-Dämon (RSED), der den Verbindungsaufbau verwaltet. Der RSE-Dämon ist auch für die Ausführung im Einzervermodus verantwortlich. Um dies zu erreichen, erstellt der RSE-Dämon mindestens einen untergeordneten Prozess, auch als RSE-Thread-Pool(s) (RSEDx) bekannt.
  - RSE-Server für die einzelnen Clientanforderungen. Ein RSE-Server ist innerhalb eines RSE-Thread-Pools als Thread aktiv.
- Debug Manager (DBGMGR) koordiniert Integrated Debugger-Aktivitäten.
- (z/OS Explorer) TSO Commands Service (TSO cmd) stellt eine batchähnliche Schnittstelle für TSO- und ISPF-Befehle bereit.
- (z/OS Explorer) JMON (JES Job Monitor) stellt alle Services mit Bezug zum JES bereit.
- Common Access Repository Manager (CARMA) bietet eine Schnittstelle für die Interaktion mit Software Configuration Managers (SCMs), beispielsweise CA Endevor.
- Es sind weitere Services verfügbar. Diese können von Developer for z Systems selbst oder von zusätzlich erforderlicher Software bereitgestellt werden.

Die Beschreibung im vorherigen Abschnitt und in der Liste verdeutlichen die zentrale Rolle von RSE. Mit ein paar wenigen Ausnahmen läuft jede Clientkommunikation über RSE ab. Dies ermöglicht eine sicherheitsbezogene Netzkonfiguration, da nur eine eingeschränkte Menge an Ports für die Kommunikation zwischen Client und Host verwendet wird.

RSE besteht aus einem Dämonadressbereich, der Thread-Pooling und Adressräume steuert, um die Verbindungen und die Arbeitslast der Clients zu verwalten. Der Dämon wird als Sammelpunkt für Verbindungs- und Verwaltungszwecke eingesetzt, während die Thread-Pools die Clientarbeitslast verarbeiten. Auf Basis der in der Konfigurationsdatei `rse.env` definierten Werte und der Summe aller Clientverbindungen können mehrere Adressräume von Thread-Pools durch den Dämon gestartet werden.

---

## Taskeigner

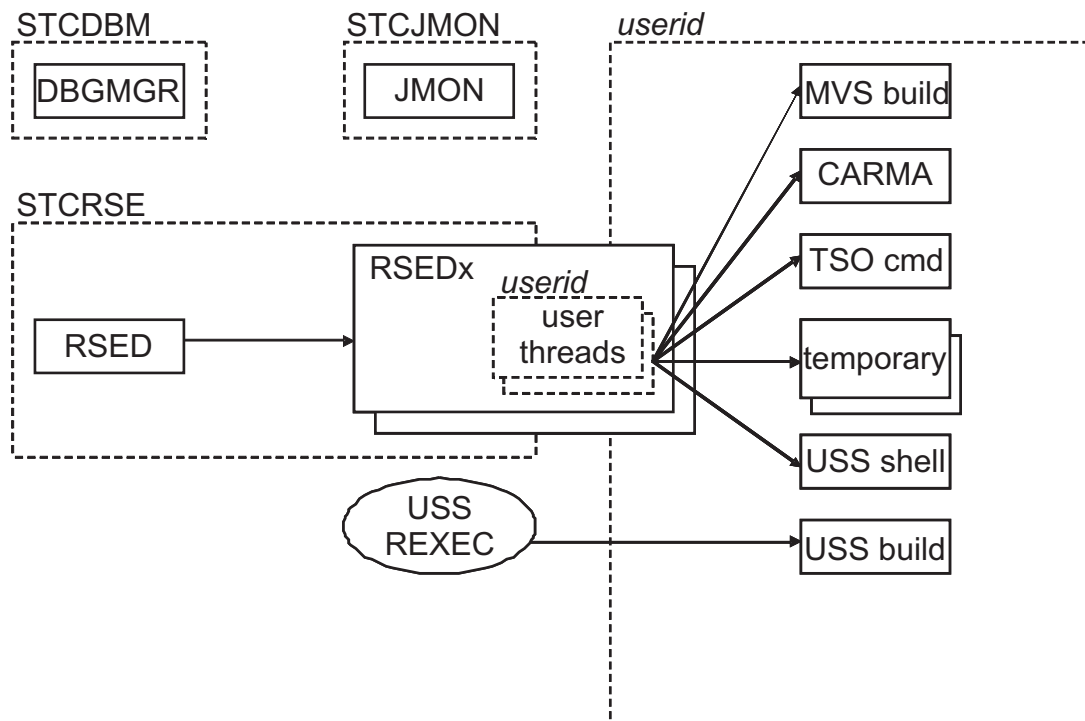


Abbildung 2. Taskeigner

Abb. 2 zeigt eine Basisübersicht über die Eigner der Sicherheitsberechtigungs-nachweise, die für verschiedene Tasks in z/OS Explorer und Developer for z Systems verwendet werden.

Das Eigentumsrecht an einer Task kann in zwei Abschnitte unterteilt werden. Gestartete Tasks gehören der Benutzer-ID, die der gestarteten Task in Ihrer Sicherheitssoftware zugewiesen wird. Alle anderen Tasks, mit Ausnahme der RSE-Thread-Pools (RSEDx), gehören der Client-Benutzer-ID.

Abb. 2 zeigt die gestarteten Tasks in z/OS Explorer und Developer for z Systems (DBGMGR, JMON und RSED) sowie gestartete Beispieltasks und Beispielsystemservices, mit denen Developer for z Systems kommuniziert. Der USS REXEC-Tag stellt den z/OS UNIX-REXEC-Service (oder SSH-Service) dar.

Der RSE-Dämon erstellt für die Verarbeitung von Prozessclientanforderungen mindestens einen Adressraum der RSE-Thread-Pools (RSEDx). Jeder RSE-Thread-Pool unterstützt mehrere Clients und gehört demselben Benutzer wie der RSE-Dämon. Jeder Client verfügt über eigene Threads innerhalb eines Thread-Pools. Diese Threads gehören der Client-Benutzer-ID.

Abhängig von den vom Client ausgeführten Aktionen können für die Ausführung der angeforderten Aktion zusätzliche Adressräume gestartet werden. Diese gehören alle der Client-Benutzer-ID. Diese Adressräume können ein MVS-Batch-Job, eine APPC-Transaktion oder ein untergeordneter z/OS UNIX-Prozess sein. Beachten Sie, dass ein untergeordneter z/OS UNIX-Prozess in einem z/OS UNIX-Initiator (BPXAS) aktiv ist und als gestartete Task in JES angezeigt wird.

Die Erstellung dieser Adressräume wird in den meisten Fällen von einem Benutzerthread in einem Thread-Pool entweder direkt oder mithilfe von Systemservices wie ISPF ausgelöst. Der Adressraum kann aber auch von einem

Fremdanbieter erstellt werden. Der z/OS UNIX-RExec-Service oder der SSH-Service sind beim Starten von Builds in z/OS UNIX beteiligt.

Die benutzerspezifischen Adressräume werden bei Abschluss der Tasks oder bei Ablauf eines Inaktivitätszeitgebers beendet. Die gestarteten Tasks bleiben aktiv. Die in Abb. 2 auf Seite 5 aufgeführten Adressräume bleiben für einen längeren Zeitraum im System sichtbar. Sie sollten allerdings beachten, dass z/OS UNIX so entwickelt wurde, dass es auch einige kurz andauernde, temporäre Adressräume gibt.

## Integrated Debugger

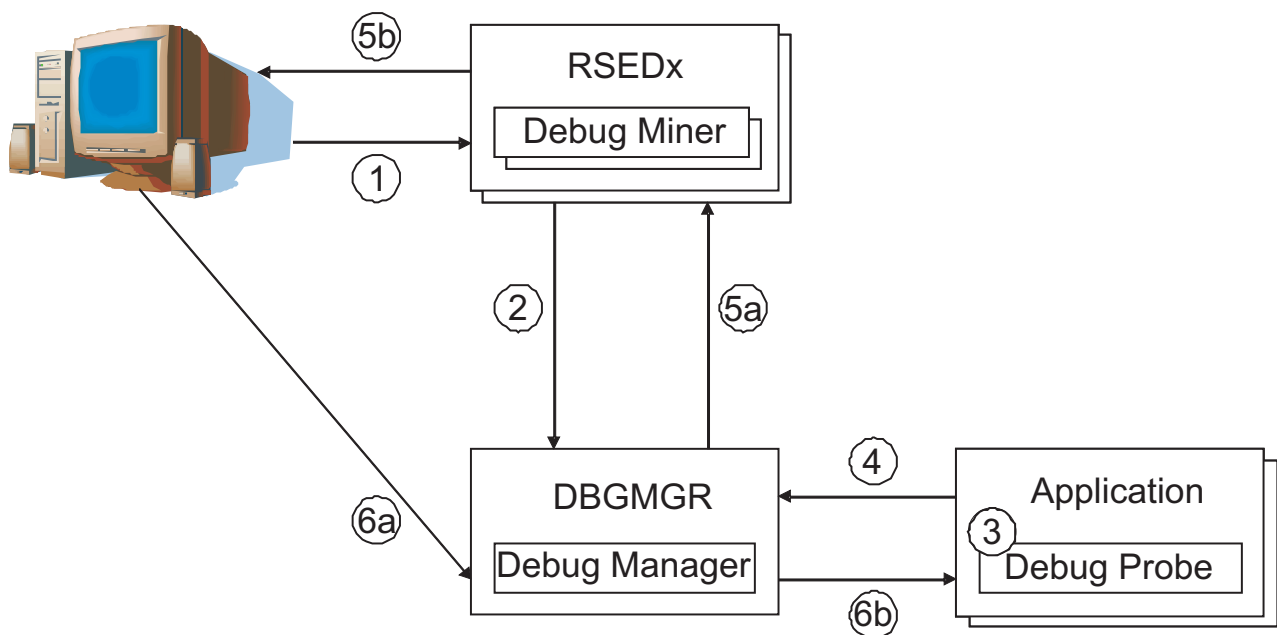


Abbildung 3. Integrated Debugger

Integrated Debugger wird zum Debuggen verschiedener Anwendungen verwendet. In Abbildung 5 wird eine schematische Übersicht gezeigt, wie ein Developer for z Systems-Client ein Debugging für eine Anwendung durchführen kann.

1. Der Client wird mit der normalen Developer for z Systems-Host-Anmeldung mit dem Host verbunden.
2. Als Teil der Anmeldung registriert Debug Miner den Benutzer bei Debug Manager, der in der gestarteten DBGMGR-Task aktiv ist.
3. Wenn eine Anwendung mit einem Anzeiger gestartet wird, ruft Language Environment (LE) den Debug-Testmonitor auf.
4. Der Debug-Testmonitor wird bei Debug Manager registriert.
5. Mithilfe von Debug Miner benachrichtigt Debug Manager den Developer for z Systems-Client des Benutzers, der diese Debugsitzung empfängt. Wenn der

- Benutzer zu diesem Zeitpunkt nicht registriert wird, ruht die Debugsitzung und wartet darauf, dass der Benutzer bei Debug Manager registriert wird.
6. Die Debug-Engine im Client kontaktiert Debug Manager, der wiederum die Daten zwischen der Debug-Engine und dem Debug-Testmonitor übergibt.

## CARMA

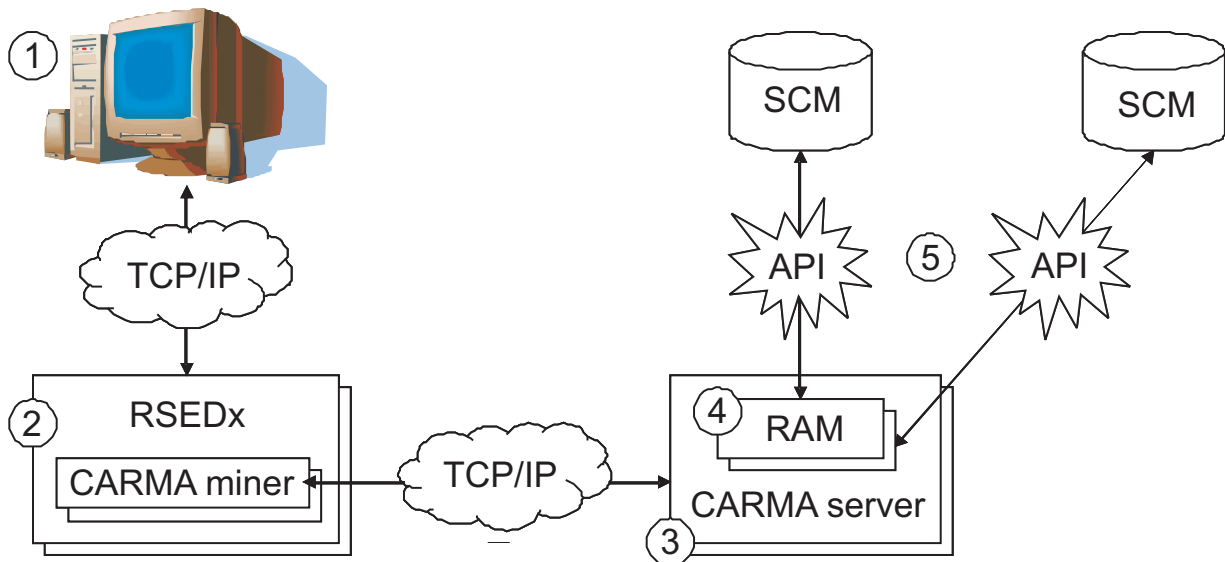


Abbildung 4. CARMA-Flow

CARMA (Common Access Repository Manager) wird für den Zugriff auf einen hostbasierten Software Configuration Manager (SCM) verwendet, beispielsweise CA Endevor® SCM. In Abb. 4 ist in einer schematischen Übersicht dargestellt, wie ein Developer for z Systems-Client auf jeden beliebigen unterstützten, hostbasierten Software Configuration Manager (SCM) zugreifen kann.

1. Der Client verwendet ein CARMA-Plug-in (Common Access Repository Manager).
2. Das CARMA-Plug-in kommuniziert mit dem CARMA-Miner, der als benutzerspezifischer Thread im RSE-Thread-Pool (RSEDx) aktiv ist. Diese Kommunikation erfolgt über eine vorhandene RSE-Verbindung.
3. Wenn der Client Zugriff auf einen SCM anfordert, bindet der CARMA-Miner an einen TCP/IP-Port und startet einen benutzerspezifischen CARMA-Server mit der Portnummer als Startargument. Der CARMA-Server stellt dann eine Verbindung zu diesem Port her und verwendet den Pfad für die Kommunikation mit dem Client. Beachten Sie, dass hostbasierte SCMs Einzelbenutzeradressräume erwarten, um auf ihre Services zuzugreifen; hierfür muss CARMA pro Benutzer einen CARMA-Server starten. Es ist nicht möglich, einen einzigen Server zu erstellen, der mehrere Benutzer unterstützt.
4. Der CARMA-Server lädt den Repository Access Manager (RAM), der den angeforderten SCM unterstützt.
5. Der RAM bearbeitet die technischen Details der Interaktion mit dem spezifischen SCM und stellt eine gemeinsame Schnittstelle für den Client dar.

## CARMA-Konfigurationsdateien

Developer for z Systems unterstützt mehrere Methoden für den Start eines CARMA-Servers. Alle Methoden haben Vor- und Nachteile. Außerdem stellt Developer for z Systems mehrere Repository Access Manager (RAM) bereit, die in zwei Gruppen eingeteilt werden können: Produktions-RAM und Muster-RAM. Es sind verschiedene Kombinationen von RAM und Serverstartmethoden als vorkonfigurierte Installation verfügbar.

Alle Serverstartmethoden nutzen eine gemeinsame Konfigurationsdatei (CRASRV.properties), die unter anderem angibt, welche Startmethode verwendet wird.

### CRASTART

Die Methode "CRASTART" startet den CARMA-Server als Subtask innerhalb von RSE. Bei dieser sehr flexiblen Konfiguration wird eine gesonderte Konfigurationsdatei verwendet, die für den Start eines CARMA-Servers erforderliche Dateizuordnungen und Programmaufrufe definiert. Mit dieser Methode wird die beste Leistung erreicht. Sie nutzt am wenigsten Ressourcen, erfordert jedoch, dass sich das Modul CRASTART im LPA befindet.

RSE ruft das Lademodul CRASTART auf, das ausgehend von den Definitionen in crastart\*.conf eine gültige Umgebung für die Ausführung von TSO- und ISPF-Batchbefehlen erstellt. Developer for z Systems kann in dieser Umgebung den CARMA-Server CRASERV ausführen. Developer for z Systems stellt mehrere Dateien crastart\*.conf bereit, die jeweils für einen bestimmten RAM vorkonfiguriert sind.

### Batchübergabe

Die Methode der Batchübergabe startet den CARMA-Server durch Übergabe eines Jobs. Dies ist die in den bereitgestellten Beispielkonfigurationsdateien verwendete Standardmethode. Sie hat den Vorteil, dass in der Jobausgabe ohne großen Aufwand auf die CARMA-Protokolle zugegriffen werden kann. Bei dieser Methode kann jeder Entwickler auch eigene Server-JCL verwenden, die er selbst verwaltet. Allerdings wird bei dieser Methode pro Entwickler, der einen CARMA-Server startet, ein JES-Initiator verwendet.

RSE ruft die CLIST CRASUB\* auf, die wiederum eine eingebettete JCL übergibt, um eine gültige Umgebung für die Ausführung von TSO- und ISPF-Batchbefehlen zu erstellen. Developer for z Systems führt in dieser Umgebung den CARMA-Server (CRASERV) aus. Developer for z Systems stellt mehrere CRASUB\*-Member bereit, die jeweils für einen bestimmten RAM vorkonfiguriert sind.



## z/OS UNIX-Verzeichnisstruktur

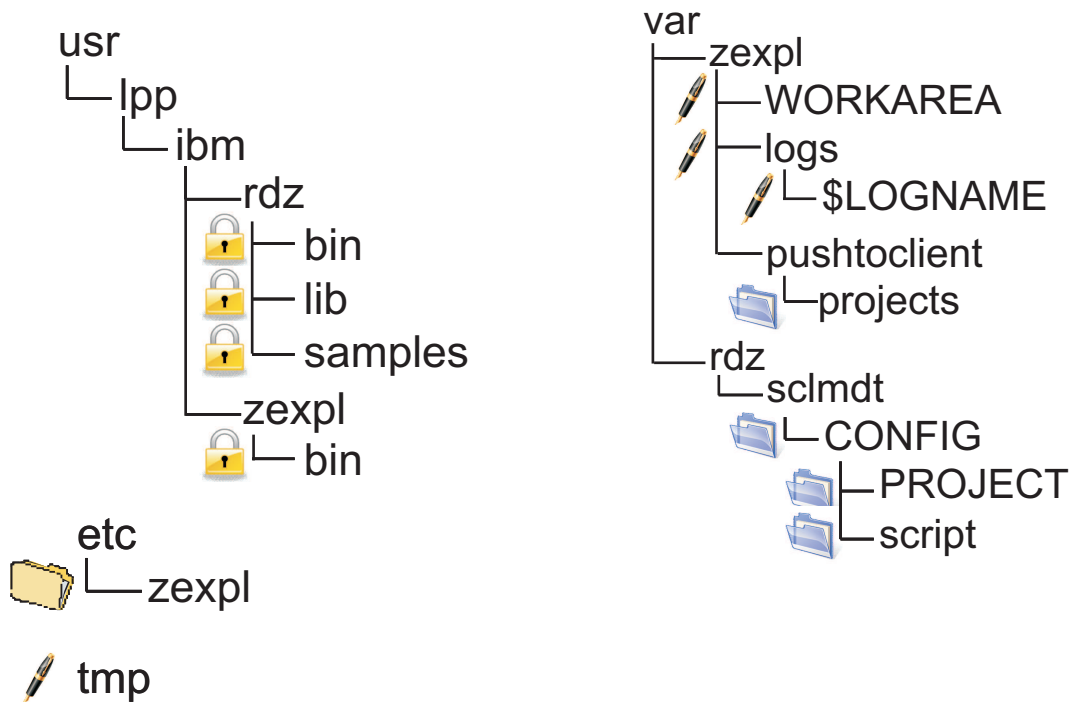


Abbildung 5. z/OS UNIX-Verzeichnisstruktur

Abb. 5 liefert einen Überblick über die von Developer for z Systems verwendeten z/OS UNIX-Verzeichnisse. Die folgende Liste enthält nicht nur Informationen zu jedem von Developer for z Systems verwendeten Verzeichnis, sondern gibt auch an, wie die Position geändert werden kann und wer die darin enthaltenen Daten verwaltet.

- /usr/lpp/ibm/rdz/ ist der Stammverzeichnispfad für den Produktcode von Developer for z Systems. Die eigentliche Position ist in der Konfigurationsdatei rdz.env angegeben (Variable RDZ\_HOME). Die enthaltenen Dateien werden von SMP/E verwaltet.
- Developer for z Systems stellt Dateien in /usr/lpp/ibm/zexpl/bin - dem Verzeichnis für Binärdateien von z/OS Explorer. Die eigentliche Position ist in der z/OS Explorer-Konfiguration angegeben. Die enthaltenen Dateien werden von SMP/E verwaltet.
- /etc/zexpl/ beinhaltet die Konfigurationsdateien von z/OS Explorer und Developer for z Systems. Die eigentliche Position ist in der gestarteten Task RSED angegeben (Variable CNFG). Die enthaltenen Dateien werden vom Systemprogrammierer verwaltet.
- /tmp/ wird vom Legacy ISPF Gateway zum Speichern von temporären Daten verwendet. Einige IVPs speichern ihre Ausgabe hier. Die enthaltenen Dateien werden von ISPF und IVPs verwaltet. Die eigentliche Position kann mit der Variablen TMPDIR in der Datei rse.env angepasst werden. Das Verzeichnis ist außerdem die Standardposition für Java™-Speicherauszugsdateien, die mit der Variable \_CEE\_DUMPTARG in rse.env angepasst werden kann.

**Anmerkung:** /tmp/ erfordert die Berechtigungsbitmaske 777, um jedem Client das Erstellen von temporären Daten zu ermöglichen.

- `/var/zexpl/WORKAREA/` wird vom Legacy ISPF Gateway und SCLMDT verwendet, um Daten zwischen z/OS UNIX und MVS-basierten Adressräumen zu übertragen. Die eigentliche Position ist in `rse.env` angegeben (Variable `CGI_ISPWORK`). Die enthaltenen Dateien werden von ISPF und SCLMDT verwaltet.

**Anmerkung:** `/var/zexpl/WORKAREA/` erfordert die Berechtigungsbitmaske 777, um jedem Client das Erstellen von temporären Daten zu ermöglichen. Developer for z Systems schreibt Protokollnachrichten in die Protokolldateien von z/OS Explorer, die sich in `/var/zexpl/zexpl/logs/$LOGNAME` befinden. Die eigentliche Position ist in der z/OS Explorer-Konfiguration angegeben. Die darin enthaltenen Dateien werden vom Produktcode von z/OS Explorer und Developer for z Systems verwaltet.

- `/var/rdz/sclmdt/CONFIG/` sperrt allgemeine SCLMDT-Konfigurationsdateien. Die eigentliche Position ist in `rdz.env` angegeben (Variable `SCLMDT_CONF_HOME`). Die enthaltenen Dateien werden vom SCLM-Administrator verwaltet.
- `/var/rdz/sclmdt/CONFIG/PROJECT/` sperrt SCLMDT-Projektkonfigurationsdateien. Die eigentliche Position ist in `rdz.env` angegeben (Variable `SCLMDT_CONF_HOME`). Die enthaltenen Dateien werden vom SCLM-Administrator verwaltet.
- `/var/rdz/sclmdt/CONFIG/script/` sperrt SCLMDT-bezogene Scripts, die von anderen Produkten verwendet werden können. Die eigentliche Position ist nirgendwo angegeben. Die enthaltenen Dateien werden vom SCLM-Administrator verwaltet.
- `/var/rdz/pushtoclient/` enthält Clientkonfigurationsdateien, Update-Informationen zum Clientprodukt und hostbasierte Projektinformationen, die bei der Verbindung mit dem Host mittels Push auf den Client übertragen werden. Die eigentliche Position ist in `pushtoclient.properties` angegeben (Variable `pushtoclient.folder`). Die darin enthaltenen Dateien werden von einem Clientadministrator von Developer for z Systems verwaltet.
- `/var/rdz/pushtoclient/projects/` enthält die hostbasierten Projektdefinitionsdateien. Die tatsächliche Position wird in der Datei `/var/rdz/pushtoclient/keymapping.xml` angegeben, die von einem Administrator der Developer for z Systems-Clients erstellt und verwaltet wird. Die enthaltenen Dateien werden von einem Projektleiter oder einem leitenden Entwickler verwaltet.

---

## Kapitel 2. Sicherheitsaspekte

Developer for z Systems erweitert z/OS Explorer, indem zusätzliche Funktionen bereitgestellt werden. Einige davon interagieren mit anderen Systemkomponenten und Produkten, wie Software Configuration Manager (SCM). Developer for z Systems specific security definitions are used to secure the provided functions.

Die von den Servern und Services von Developer for z Systems verwendeten Sicherheitsmechanismen sind nur wirksam, wenn die zugrunde liegenden Dateisysteme geschützt sind. Dies impliziert, dass die Programmbibliotheken und Konfigurationsdateien nur von vertrauenswürdigen Systemadministratoren aktualisiert werden können.

Developer for z Systems baut auf IBM Explorer for z/OS auf. z/OS Explorer-spezifische Informationen finden Sie in "Aspekte bei der Sicherheit" ("Security consideration") im Handbuch *IBM Explorer for z/OS Host Configuration Reference* (IBM Form SC27-8438).

Dieses Kapitel enthält die folgenden Abschnitte:

- „Authentifizierungsmethoden“
- „Verbindungssicherheit“ auf Seite 12
- „Debug-Sicherheit“ auf Seite 13
- „CICSTS-Sicherheit“ auf Seite 13
- „SCLM-Sicherheit“ auf Seite 14
- „Sicherheitsdefinitionen“ auf Seite 14

---

### Authentifizierungsmethoden

#### CARMA-Authentifizierung

Die Clientauthentifizierung wird vom RSE-Dämon als Teil der Verbindungsanforderung des Clients vorgenommen. CARMA wird von einem benutzerspezifischen aus gestartet und übernimmt die Sicherheitsumgebung des Benutzers, wodurch die Notwendigkeit einer zusätzlichen Authentifizierung umgangen wird.

#### SCLM Developer Toolkit-Authentifizierung

Die Clientauthentifizierung wird vom RSE-Dämon als Teil der Verbindungsanforderung des Clients vorgenommen. SCLMDT wird von einem benutzerspezifischen aus gestartet und übernimmt die Sicherheitsumgebung des Benutzers, wodurch die Notwendigkeit einer zusätzlichen Authentifizierung umgangen wird.

### Debug Manager-Authentifizierung

Die Clientauthentifizierung wird vom RSE-Dämon als Teil der Verbindungsanforderung des Clients vorgenommen. Sobald der Benutzer authentifiziert ist, werden selbsterstellte PassTickets für alle zukünftigen Authentifizierungsanforderungen verwendet, einschließlich des automatischen Anmeldens beim Debug Manager.

Debug Manager muss für die Überprüfung von PassTickets berechtigt sein, damit eine Überprüfung durch Debug Manager für die vom RSE übermittelten Benutzer-IDs und PassTickets möglich ist. Das Lademodul AQEZPCM, das sich standardmäßig in der Ladebibliothek FEL.SFEKAUTH befindet, muss deshalb für APF-autorisiert sein.

Wenn eine clientbasierte Debug-Engine eine Verbindung zu Debug Manager herstellt, muss sie ein gültiges Sicherheitstoken für die Authentifizierung vorlegen.

## Verbindungssicherheit

Die Hostkommunikation zwischen dem Developer for z Systems-Client und dem Host geht über RSE: dadurch wird die Verbindungssicherheit verwendet, die in z/OS Explorer bereitgestellt wird.

Einige Developer for z Systems-Services verwenden einen separaten, externen Kommunikationspfad (Client-Host):

- Die Integrated Debugger-Engine des Clients wird mit dem Debug Manager des Hosts verbunden. Die Verschlüsselungsdetails werden von einer AT-TLS-Richtlinie (Application Transparent Transport Layer Security) gesteuert.
- Ferne (hostbasierte) Aktionen in z/OS UNIX-Unterprojekten verwenden einen REXEC- oder SSH-Server auf dem Host. Die SSH-Kommunikation ist stets verschlüsselt.

## Mit Integrated Debugger verschlüsselte Kommunikation

Die externe Kommunikation (Client-Host) mit dem optionalen Debug Manager kann ebenfalls verschlüsselt werden. Erstellen Sie zum Aktivieren der Verschlüsselung eine AT-TLS-Richtlinie (Application Transparent TLS) für den Port, der von Debug Manager für die externe Kommunikation verwendet wird (Standardport 5335). Sie finden eine Beispielrichtlinie unter Abb. 6. Details zum Einrichten von AT-TLS finden Sie im Abschnitt Kapitel 7, „AT-TLS konfigurieren“, auf Seite 43.

```

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
{
  TTLEnabled                           On
  Trace                                2
}

```

Abbildung 6. AT-TLS-Richtlinie für Debug Manager

**Anmerkung:** Die Kommunikationsmethode, die von der Debug-Engine auf dem Developer for z Systems-Client für die Kommunikation mit dem Debug Manager auf dem Host verwendet wird, ist standardmäßig an die Kommunikationsmethode

gebunden, die von dem Developer for z Systems-Client für die Kommunikation mit dem RSE-Dämon verwendet wird. Dies impliziert, dass davon ausgegangen wird, dass die Verschlüsselung auch für den Debug Manager aktiviert ist, wenn sie es für RSE ist. Für andere Konfigurationen sind jedoch auch alternative Szenarios verfügbar.

---

## Debug-Sicherheit

Für die optionale Komponente Integrated Debugger ist es erforderlich, dass Benutzer über ausreichende Zugriffsberechtigungen für angegebene Sicherheitsprofile verfügen. Wenn der Benutzer nicht über die erforderliche Berechtigung verfügt, wird die Debugsitzung nicht gestartet.

Developer for z Systems überprüft den Zugriff auf die Profile, die in Tabelle 1 aufgeführt sind, um festzustellen, welche Debugberechtigungen erteilt wurden.

*Tabelle 1. SAF-Informationen für Debugfunktionen*

FACILITY-Profil	Erforderlicher Zugriff	Ergebnis
AQE.AUTHDEBUG.STDPGM	READ	Benutzer kann ein Debugging für Anwendungen mit Fehlerstatus durchführen
AQE.AUTHDEBUG.AUTHPGM	READ	Benutzer kann ein Debugging für Anwendungen mit Fehlerstatus sowie für berechnete Anwendungen durchführen

### Anmerkung:

- Developer for z Systems geht davon aus, dass ein Benutzer über keine Zugriffsberechtigung verfügt, wenn die Sicherheitssoftware nicht feststellen kann, ob der Benutzer über die Berechtigung für den Zugriff auf ein Profil verfügt. Dies ist zum Beispiel der Fall, wenn das Profil gar nicht definiert ist.
- In Versionen von Developer for z Systems vor Version 9.1.1 wurde geprüft, ob die Berechtigung UPDATE für das Profil AQE.AUTHDEBUG.WRITEBUFFER vorhanden war, um das Debugging von schreibgeschützten CICS-Transaktionen zu ermöglichen. Dieses Profil wird nicht mehr verwendet und kann gelöscht werden, wenn Ihr Hostsystem nur noch über Developer for z Systems ab Version 9.1.1 verfügt.

Die folgenden Beispielsicherheitsdefinitionen ermöglichen allen Benutzern in der Gruppe RDZDEBUG das Debugging von Anwendungen mit Fehlerstatus:

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

---

## CICSTS-Sicherheit

Mit dem optionalen Integrated Debugger kann ein Debugging von CICS-Transaktionen durchgeführt werden. Ausführliche Informationen hierzu enthält der Abschnitt „CICS-Transaktionsdebugging“ auf Seite 41.

---

## SCLM-Sicherheit

SCLM Developer Toolkit stellt optionale Sicherheitsfunktionen für die Builderstellung, die Umstufung und das Deployment bereit.

Wenn ein SCLM-Administrator die Sicherheit für eine Funktion aktiviert hat, wird SAF aufgerufen, um zu überprüfen, ob die geschützte Funktion mit der ID des Aufrufenden oder einer Ersatzbenutzer-ID ausgeführt werden darf.

Weitere Informationen zu den erforderlichen SCLM-Sicherheitsdefinitionen enthält der *SCLM Developer Toolkit Administrator's Guide* (IBM Form SC23-9801).

---

## Sicherheitsdefinitionen

Passen Sie den Beispieljob FELRACF an, der RACF-Beispielbefehle enthält, und übergeben Sie ihn, um die Basissicherheitsdefinitionen für Developer for z Systems zu erstellen. Passen Sie den Beispieljob AQERACF an, der RACF-Beispielbefehle enthält, und übergeben Sie ihn, um die Sicherheitsdefinitionen für Integrated Debugger zu erstellen.

FELRACF und AQERACF befinden sich in FEL.#CUST.JCL, sofern sie bei der Anpassung und Übergabe des Jobs FEL.SFELSAMP(FELSETUP) keine andere Position angegeben haben. Weitere Informationen finden Sie im Abschnitt "Anpassungskonfiguration" im Handbuch *Rational Developer for z Systems Hostkonfiguration*.

Weitere Informationen zu RACF-Befehlen finden Sie in der Veröffentlichung *RACF Command Language Reference* (IBM Form SA22-7687).

## Voraussetzungen und Prüfliste

Der Sicherheitsadministrator muss die in Tabelle 2 aufgelisteten Werte kennen, um die Sicherheitskonfiguration durchzuführen. Diese Werte wurden in früheren Schritten der Installation und Anpassung von Rational Developer for z Systems definiert.

*Tabelle 2. Variablen der Sicherheitskonfiguration*

Beschreibung	<ul style="list-style-type: none"><li>Standardwert</li><li>Entsprechende Quelle</li></ul>	Wert
Übergeordnetes Qualifikationsmerkmal für das Developer for z Systems-Produkt	<ul style="list-style-type: none"><li>FEL</li><li>SMP/E-Installation</li></ul>	
Übergeordnetes Qualifikationsmerkmal für die Developer for z Systems-Anpassung	<ul style="list-style-type: none"><li>FEL.#CUST</li><li>FEL.SFELSAMP(FELSETUP), wie im Abschnitt "Anpassungskonfiguration" im Handbuch <i>Rational Developer for z Systems Hostkonfiguration</i> beschrieben.</li></ul>	

Tabelle 2. Variablen der Sicherheitskonfiguration (Forts.)

Beschreibung	<ul style="list-style-type: none"> <li>• Standardwert</li> <li>• Entsprechende Quelle</li> </ul>	Wert
Name der gestarteten Task für Integrated Debugger	<ul style="list-style-type: none"> <li>• DBGMR</li> <li>• FEL.#CUST.PROCLIB(DBGMR), wie im Abschnitt "PROCLIB-Änderungen" im Handbuch <i>Rational Developer for z Systems Hostkonfiguration</i> beschrieben.</li> </ul>	

Die folgende Liste enthält eine Übersicht über die Aktionen, die zur Durchführung der Basissicherheitskonfiguration von Developer for z Systems erforderlich sind. Um diese Anforderungen zu erfüllen, können je nach erforderlicher Sicherheitsstufe verschiedene Methoden wie in den folgenden Abschnitten dokumentiert verwendet werden.

- „Sicherheitseinstellungen und -klassen aktivieren“
- „Gestartete Tasks für Developer for z Systems definieren“ auf Seite 16
- „Debug Manager als sicheren z/OS UNIX-Server definieren“ auf Seite 17
- „Programmgesteuerte MVS-Bibliotheken für Debug Manager definieren“ auf Seite 17
- „Zugriff auf Integrated Debugger definieren“ auf Seite 22
- „Dateiprofile definieren“ auf Seite 22
- „Sicherheitseinstellungen prüfen“ auf Seite 23

## Sicherheitseinstellungen und -klassen aktivieren

Developer for z Systems verwendet eine Reihe von Sicherheitsmechanismen, um für den Client eine geschützte und kontrollierte Hostsystemumgebung bereitzustellen. Zu diesem Zweck müssen mehrere Klassen und Sicherheitseinstellungen aktiv sein, wie in den folgenden RACF-Beispielbefehlen gezeigt:

- Anzeige der aktuellen Einstellungen
  - SETROPTS LIST
- Aktivieren der Funktionsklasse für Integrated Debugger
  - SETROPTS GENERIC(FACILITY)
  - SETROPTS CLASSACT(FACILITY) RACLIST(FACILITY)
- Aktivieren von Definitionen gestarteter Tasks für Integrated Debugger
  - SETROPTS GENERIC(STARTED)
  - RDEFINE STARTED \*\* STDATA(USER(=MEMBER) GROUP(STCGROUP) TRACE(YES))
  - SETROPTS CLASSACT(STARTED) RACLIST(STARTED)
- Aktivieren der Programmsteuerung für Integrated Debugger
  - RDEFINE PROGRAM \*\* ADDMEM('SYS1.COMDLIB'//NOPADCHK) UACC(READ)
  - SETROPTS WHEN(PROGRAM)

**Anmerkung:** Wenn die Klasse PROGRAM bereits ein Profil \* enthält, sollten Sie das Profil \*\* nicht erstellen. Dadurch wird der von der Sicherheitssoftware verwendete Suchpfad unbestimmt und kompliziert. Führen Sie in einem

solchen Fall die vorhandenen Definitionen aus dem Profil \* mit den neuen Definitionen des Profils \*\* zusammen. Verwenden Sie das Profil \*\*, wie in *Security Server RACF Security Administrator's Guide* (IBM Form SA22-7683) dokumentiert.

**Achtung:** Wenn "WHEN PROGRAM" aktiv ist, müssen einige Produkte (beispielsweise FTP) programmgesteuert sein. Testen Sie diese Programmsteuerung, bevor Sie sie auf einem Produktionssystem aktivieren.

## OMVS-Segment für Benutzer von Developer for z Systems definieren

Für jeden Benutzer von Developer for z Systems muss ein RACF-OMVS-Segment oder ein funktional entsprechendes Element definiert werden, das eine gültige z/OS UNIX-Benutzer-ID (UID, ungleich null) angibt. Darüber hinaus müssen für jeden Benutzer ein Ausgangsverzeichnis und ein Shellbefehl definiert werden. Für die Standardgruppe jedes Benutzers ist ebenfalls ein OMVS-Segment mit einer Gruppen-ID erforderlich.

Bei Verwendung der optionalen Komponente 'Integrated Debugger' ist für die Benutzer-ID zu der Anwendung, bei der Fehler behoben werden soll, sowie die zugehörige Standardgruppe ebenfalls ein gültiges RACF OMVS-Segment oder ein Äquivalent erforderlich.

Ersetzen Sie in den folgenden RACF-Beispielbefehlen die Platzhalter #userid, #user-identifizier, #group-name und #group-identifizier durch tatsächliche Werte:

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

## Gestartete Tasks für Developer for z Systems definieren

Die folgenden RACF-Beispielbefehle erstellen die gestartete DBGMGR-Task mit der zugeordneten geschützten Benutzer-ID (STCDBM) und der Gruppe STCGROUP.

- ADDGROUP STCGROUP OMVS(AUTOGID)  
DATA('GROUP WITH OMVS SEGMENT FOR STARTED TASKS')
- ADDUSER STCDBM DFLTGRP(STCGROUP) NOPASSWORD NAME('DEBUG MANAGER')  
OMVS(AUTOUID HOME(/tmp) PROGRAM(/bin/sh) )  
DATA('Rational Developer for z Systems')
- RDEFINE STARTED DBGMGR.\* DATA('DEBUG MANAGER')  
STDATA(USER(STCDBM) GROUP(STCGROUP) TRUSTED(NO))
- SETROPTS RACLIST(STARTED) REFRESH

### Anmerkung:

- Stellen Sie sicher, dass die Benutzer-IDs der gestarteten Tasks durch Angabe des Schlüsselworts NOPASSWORD geschützt sind.
- Die gestartete Task von Debug Manager (DBGMGR) wird nur von der Funktion 'Integrated Debugger' verwendet.



## Debug Manager als sicheren z/OS UNIX-Server definieren

Integrated Debugger benötigt die Zugriffsberechtigung UPDATE für das Profil BPX.SERVER, um die Sicherheitsumgebung für den Debug-Thread erstellen oder löschen zu können. Beachten Sie, dass die Verwendung von UID(0) zum Umgehen dieser Anforderung nicht unterstützt wird. Diese Berechtigung ist nur bei Verwendung der optionalen Funktion 'Integrated Debugger' erforderlich.

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

**Achtung:** Mit dem Definieren des Profils BPX.SERVER wechselt z/OS UNIX vollständig von der Sicherheit auf UNIX-Ebene zur Sicherheit auf z/OS UNIX-Ebene, die bedeutend sicherer ist. Möglicherweise hat dieser Wechsel Auswirkungen auf andere z/OS UNIX-Anwendungen und -Operationen. Testen Sie die Sicherheit, bevor Sie sie auf einem Produktionssystem aktivieren. Weitere Informationen zu den verschiedenen Sicherheitsstufen finden Sie in der Veröffentlichung *UNIX System Services Planning* (IBM Form GA22-7800).

## Programmgesteuerte MVS-Bibliotheken für Debug Manager definieren

Server mit der Berechtigung für BPX.SERVER müssen in einer sauberen, programmgesteuerten Umgebung ausgeführt werden. Diese Voraussetzung impliziert, dass alle von Debug Manager aufgerufenen Programme ebenfalls programmgesteuert sein müssen. Die Programmsteuerung von MVS-Ladebibliotheken wird von Ihrer Sicherheitssoftware verwaltet.

Debug Manager verwendet Systembibliotheken, die Language Environment-Laufzeit und die Ladebibliothek (ISP.SISPLoad) von Developer for z Systems.

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

**Anmerkung:** Wenn die Klasse PROGRAM bereits ein Profil \* enthält, sollten Sie das Profil \*\* nicht verwenden. Durch das Profil wird der von Ihrer Sicherheitssoftware verwendete Suchpfad unbestimmt und kompliziert. Führen Sie in einem solchen Fall die vorhandenen Definitionen aus dem Profil \* mit den neuen Definitionen des Profils \*\* zusammen. Verwenden Sie das Profil \*\*, wie in *Security Server RACF Security Administrator's Guide* (IBM Form SA22-7683) dokumentiert.

Zur Unterstützung optionaler Services müssen die folgenden zusätzlich vorausgesetzten Bibliotheken programmgesteuert sein. Diese Liste enthält keine Dateien, die für ein Produkt spezifisch sind, mit dem Developer for z Systems interagiert, beispielsweise IBM Explorer for z/OS.

- Alternative REXX-Laufzeitbibliothek für SCLM Developer Toolkit
  - REXX.\*.SEAGALT

**Anmerkung:** Bibliotheken, die in den Link-Pack-Bereich (LPA) gestellt werden müssen, erfordern Programmsteuerberechtigungen, wenn für den Zugriff auf diese Bibliotheken LINKLIST oder STEPLIB verwendet wird. In dieser Veröffentlichung ist die Verwendung der folgenden LPA-Bibliotheken dokumentiert:

- REXX-Laufzeitbibliothek für SCLM Developer Toolkit
  - REXX.\*.SEAGLPA
- Developer for z Systems für CARMA
  - FEL.SFELLPA

## PassTicket-Unterstützung für RSE definieren

Das Kennwort des Clients bzw. ein anderes Identifikationsmechanismus, wie ein X.509-Zertifikat, wird nur verwendet, um die Identität beim Herstellen der Verbindung zu überprüfen. Anschließend wird die Threadsicherheit mit PassTickets verwaltet. Dieser Schritt ist erforderlich, damit Clients die Verbindung herstellen können.

PassTickets sind vom System generierte Kennwörter mit einer Lebensdauer von ca. 10 Minuten. Die generierten PassTickets basieren auf einem geheimen Schlüssel. Dieser Schlüssel ist eine 64-Bit-Zahl (16 Hexadezimalzeichen). Ersetzen Sie in den folgenden RACF-Beispielbefehlen den Platzhalter key16 durch eine benutzerdefinierte 16 Zeichen lange Hexadezimalzeichenfolge aus den Zeichen 0-9 und A-F.

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

RSE unterstützt die Verwendung von anderen Anwendungs-IDs als FEKAPPL. Entfernen Sie in `rdz.env` die Kommentarzeichen vor der Option 'APPLID=FEKAPPL' und passen Sie die Option an, um sie zu aktivieren. Lesen Sie hierzu die Informationen im Abschnitt 'Zusätzliche Java-Startparameter mit `_RSE_JVAOPTS` definieren' im Handbuch *IBM Rational Developer for z Systems Hostkonfiguration*. Die Definitionen der Klasse PTKTDATA müssen mit der eigentlichen, von RSE verwendeten Anwendungs-ID übereinstimmen.

Sie sollten OMVSAPPL nicht als Anwendungs-ID verwenden, da diese ID den geheimen Schlüssel für die meisten z/OS UNIX-Anwendungen offenlegt. Ebenso wenig sollten Sie die MVS-Standardanwendungs-ID (MVS gefolgt von der SMF-ID des Systems) verwenden, da diese ID den geheimen Schlüssel für die meisten MVS-Anwendungen, einschließlich Benutzer-Batch-Jobs, offenlegt.

### Anmerkung:

- Wenn die Klasse PTKTDATA bereits definiert ist, überprüfen Sie, ob diese als eine generische Klasse definiert ist, bevor Sie die oben aufgeführten Profile erstellen. Ab z/OS Release 1.7 werden mit der Einführung der Java-Schnittstelle zu PassTickets generische Zeichen in der Klasse PTKTDATA unterstützt.
- Ersetzen Sie den Platzhalter (\*) in der Definition IRRPTAUTH.FEKAPPL.\* durch eine gültige Maske der Benutzer-ID, um die Benutzer-IDs einzuschränken, für die RSE ein PassTicket generieren kann.

- Abhängig von Ihren RACF-Einstellungen steht der Benutzer, der ein Profil definiert, möglicherweise auch auf der Zugriffsliste des Profils. Entfernen Sie diese Berechtigung für die PTKTDATA-Profile.
- Damit JES Job Monitor die vom RSE angegebenen PassTickets überprüfen kann, müssen JES Job Monitor und RSE dieselbe Anwendungs-ID besitzen. Für JES Job Monitor wird die Anwendungs-ID in der Konfigurationsdatei FEJJCNFG mit der Anweisung APPLID festgelegt.
- Wenn Sie auf Ihrem System ein Verschlüsselungsprodukt installiert haben und dieses verfügbar ist, kann der Anwendungsschlüssel zur sicheren Anmeldung für einen zusätzlichen Schutz verschlüsselt werden. Verwenden Sie dazu das Schlüsselwort KEYENCRYPTED anstelle von KEYMASKED. Weitere Informationen finden Sie in *Security Server RACF Security Administrator's Guide* (IBM Form SA22-7683).

**Achtung:** Die Clientverbindungsanforderung schlägt fehl, wenn PassTickets nicht richtig konfiguriert sind.

## z/OS UNIX-Dateizugriffsberechtigung für RSE definieren

Der Operatorbefehl **MODIFY LOGS** verwendet die Benutzer-ID der gestarteten RSED-Task, um Hostprotokolle und Konfigurationsdaten zu erfassen. Und standardmäßig werden Benutzerprotokolldateien mit Berechtigungen für einen sicheren Dateizugriff (nur der Eigner hat Zugriff) erstellt. Damit sichere Benutzerprotokolldateien erfasst werden können, muss die Benutzer-ID der gestarteten RSED-Task über eine Leseberechtigung verfügen.

Das Argument OWNER des Operatorbefehls **MODIFY LOGS** führt dazu, dass die angegebene Benutzer-ID zum Eigner der erfassten Daten wird. Um das Eigentumsrecht zu ändern, muss die Benutzer-ID der gestarteten RSED-Task über die Berechtigung verfügen, den z/OS UNIX-Dienst CHOWN zu verwenden.

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

Beachten Sie, dass wenn das Profil SUPERUSER.FILESYS.ACLOVERRIDE definiert ist, die in der Zugriffskontrollliste (ACL - Access Control List) definierten Zugriffsberechtigungen Vorrang vor den durch SUPERUSER.FILESYS genehmigten Berechtigungen haben. Die Benutzer-ID der gestarteten RSED-Task benötigt eine READ-Zugriffsberechtigung auf das Profil SUPERUSER.FILESYS.ACLOVERRIDE, um ACL-Definitionen zu umgehen.

## Anwendungsschutz für RSE definieren

Während der Clientanmeldung prüft der RSE-Dämon, ob ein Benutzer die Anwendung verwenden darf.

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

**Anmerkung:**

- RSE unterstützt die Verwendung von anderen Anwendungs-IDs als FEKAPPL. Ausführlichere Informationen dazu finden Sie in „PassTicket-Unterstützung für RSE definieren“ auf Seite 18. Die Klassendefinition APPL muss mit der eigentlichen, von RSE verwendeten Anwendungs-ID übereinstimmen.
- Die Clientverbindungsanforderung ist erfolgreich, wenn die Anwendungs-ID nicht in der Klasse APPL definiert ist.
- Die Clientverbindungsanforderung schlägt nur fehl, wenn die Anwendungs-ID definiert ist und der Benutzer keinen Lesezugriff auf das Profil hat.

## Programmgesteuerte z/OS UNIX-Dateien für RSE definieren

Server mit der Berechtigung für BPX.SERVER müssen in einer sauberen, programmgesteuerten Umgebung ausgeführt werden. Diese Voraussetzung impliziert, dass alle von RSE aufgerufenen Programme ebenfalls programmgesteuert sein müssen. Die Programmsteuerung für z/OS UNIX-Dateien wird mit dem Befehl **extattr** verwaltet. Für die Ausführung dieses Befehls benötigen Sie die Zugriffsberechtigung READ für BPX.FILEATTR.PROGCTL in der Klasse FACILITY oder die UID(0).

Der RSE-Server verwendet die gemeinsam genutzte Java-Bibliothek von RACF (/usr/lib/libIRRRacf\*.so).

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

### Anmerkung:

- Ab z/OS 1.9 wird /usr/lib/libIRRRacf\*.so während der SMP/E-Installation von RACF im programmgesteuerten Modus installiert.
- Ab z/OS 1.10 ist /usr/lib/libIRRRacf\*.so Teil der System Authorization Facility (SAF), die zum z/OS-Basisprodukt gehört. Damit ist die JAR-Datei auch für Kunden verfügbar, die kein RACF verwenden.
- Wenn Sie ein anderes Sicherheitsprodukt als RACF verwenden, kann eine andere Konfiguration erforderlich sein. Weitere Informationen finden Sie in der Dokumentation zu Ihrem Sicherheitsprodukt.
- Bei der SMP/E-Installation von Developer for z Systems wird das Programmsteuerungsbit für interne RSE-Programme gesetzt.
- Verwenden Sie zum Anzeigen des aktuellen Status des Programmsteuerbits den z/OS UNIX-Befehl **ls -Eog**. Die Datei ist programmgesteuert, wenn der Buchstabe **p** in der zweiten Zeichenfolge angezeigt wird.

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

## JES-Befehlssicherheit definieren

JES Job Monitor setzt alle von einem Benutzer angeforderten JES-Bedienerbefehle über eine erweiterte MCS-Konsole (EMCS) ab, deren Name durch die Anweisung `CONSOLE_NAME` gesteuert wird, wie im Abschnitt "FEJJCNFG - Konfigurationsdatei für JES Job Monitor" im Handbuch *Rational Developer for z Systems Hostkonfiguration* dokumentiert.

Die folgenden RACF-Beispielbefehle geben Developer for z Systems-Benutzern bedingten Zugriff auf eine eingeschränkte Gruppe von JES-Befehlen: Hold, Release, Cancel und Purge. Benutzer haben die Ausführungsberechtigung nur, wenn sie die Befehle über JES Job Monitor absetzen. Ersetzen Sie den Platzhalter `#console` durch den aktuellen Konsolennamen.

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

**Anmerkung:**

- Wenn kein Profil MVS.MCSOPER.#console definiert ist, wird die Verwendung der Konsole zugelassen.
- Damit WHEN(CONSOLE(JMON)) funktioniert, muss die Klasse CONSOLE aktiviert sein. In der Klasse CONSOLE für EMCS-Konsolen ist jedoch keine aktuelle Profilüberprüfung vorhanden.
- Ersetzen Sie JMON nicht durch den aktuellen Konsolennamen in der Klausel WHEN(CONSOLE(JMON)). Das Schlüsselwort JMON stellt die Eingangsportanwendung und nicht den Konsolennamen dar.

**Achtung:** Wenn Sie in Ihrer Sicherheitssoftware die JES-Befehle mit dem universellen Zugriffsrecht NONE definieren, kann sich das negativ auf andere Anwendungen und Operationen auswirken. Testen Sie die Sicherheit, bevor Sie sie auf einem Produktionssystem aktivieren.

In Tabelle 3 und Tabelle 4 sehen Sie die Bedienerbefehle, die für JES2 und JES3 abgesetzt werden, sowie die eigenständigen Sicherheitsprofile zu deren Schutz.

*Tabelle 3. Bedienerbefehle von JES2 Job Monitor*

Aktion	Befehl	OPERCMDS-Profil	Erforderlicher Zugriff
Hold	\$Hx(jobid) x = {J, S oder T}	jesname.MODIFYHOLD.BAT jesname.MODIFYHOLD.STC jesname.MODIFYHOLD.TSU	UPDATE
Release	\$Ax(jobid) x = {J, S oder T}	jesname.MODIFYRELEASE.BAT jesname.MODIFYRELEASE.STC jesname.MODIFYRELEASE.TSU	UPDATE
Cancel	\$Cx(jobid) x = {J, S oder T}	jesname.CANCEL.BAT jesname.CANCEL.STC jesname.CANCEL.TSU	UPDATE
Purge	\$Cx(jobid),P x = {J, S oder T}	jesname.CANCEL.BAT jesname.CANCEL.STC jesname.CANCEL.TSU	UPDATE

*Tabelle 4. Bedienerbefehle von JES3 Job Monitor*

Aktion	Befehl	OPERCMDS-Profil	Erforderlicher Zugriff
Hold	*F,J=jobid,H	jesname.MODIFY.JOB	UPDATE
Release	*F,J=jobid,R	jesname.MODIFY.JOB	UPDATE
Cancel	*F,J=jobid,C	jesname.MODIFY.JOB	UPDATE
Purge	*F,J=jobid,C	jesname.MODIFY.JOB	UPDATE

**Anmerkung:**

- Die JES-Bedienerbefehle 'Hold', 'Release', 'Cancel' und 'Purge' sowie der Befehl 'Show JCL' können nur für Spooldateien abgesetzt werden, deren Eigner die Client-Benutzer-ID ist, sofern nicht in der Konfigurationsdatei für JES Job Monitor für LIMIT\_COMMANDS= der Wert LIMITED oder NOLIMIT angegeben ist. Weitere Informationen finden Sie im Abschnitt "Aktionen für Beschränkungen der Jobziele" in der *Hostkonfigurationsreferenz* (IBM Form SC12-4489-02).
- Benutzer können jede Spooldatei anzeigen, sofern in der Konfigurationsdatei für JES Job Monitor nicht LIMIT\_VIEW=USERID definiert ist. Weitere Informationen finden Sie in "Zugriff auf Spooldateien" in der *Hostkonfigurationsreferenz* (IBM Form SC12-4489-02).
- Auch Benutzer, die nicht berechtigt sind, diese Bedienerbefehle auszuführen, können mit JES Job Monitor Jobs übergeben und Jobausgaben lesen, sofern sie über eine ausreichende Berechtigung für mögliche Profile verfügen, die diese Ressourcen schützen, z. B. diejenigen in den Klassen JESINPUT, JESJOBS und JESSPOOL).

Ihre Sicherheitssoftware verhindert, dass ein Benutzer in einer TSO-Sitzung eine Konsole JMON erstellt, weil er sich so als JES Job Monitor Server ausgeben könnte. Auch wenn die Konsole erstellt werden kann, unterscheidet sich der Eingangspunkt (z. B. JES Job Monitor oder TSO). Von dieser Konsole abgesetzte JES-Befehle werden jedoch nicht die Sicherheitsprüfung bestehen, wenn Ihre Sicherheitssoftware wie in dieser Veröffentlichung beschrieben konfiguriert ist und der Benutzer nicht autorisiert ist, JES-Befehle über andere Mechanismen zu verwenden.

## Zugriff auf Integrated Debugger definieren

Benutzer müssen über einen Lesezugriff auf eines der aufgelisteten AQE.AUTHDEBUG.\*-Profile verfügen, um Integrated Debugger für die Fehlerbehebung bei Programmen mit Problemstatus einsetzen zu können. Benutzer mit einer Berechtigung für das Profil AQE.AUTHDEBUG.AUTHPGM sind auch zur Fehlerbehebung bei Programmen mit APF-Berechtigung autorisiert. Ersetzen Sie den Platzhalter #apf durch gültige Benutzer-IDs oder RACF-Gruppennamen für die Benutzer, die die Fehlerbehebung bei berechtigten Programmen durchführen dürfen.

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

**Anmerkung:** Bei Versionen von IBM Rational Developer for System z vor Version 9.1.1 wurde ein anderes FACILITY-Klassenprofil (AQE.AUTHDEBUG.WRITEBUFFER) eingesetzt, das nicht mehr verwendet wird. Es kann gelöscht werden, wenn auf Ihrem Hostsystem nur noch IBM Rational Developer for System z ab Version 9.1.1 eingesetzt wird.

## Dateiprofile definieren

Für die meisten Dateien (Datasets) von Developer for z Systems reicht das Zugriffsrecht READ für Benutzer und ALTER für Systemprogrammierer aus. Ersetzen Sie den Platzhalter #sysprog durch gültige Benutzer-IDs oder RACF-Gruppennamen. Fragen Sie außerdem den Systemprogrammierer, der das Produkt installiert und konfiguriert hat, nach den korrekten Dateinamen. Das bei der Installation verwendete übergeordnete Standardqualifikationsmerkmal ist FEK. Das übergeordnete Standardqualifikationsmerkmal für Dateien, die während des Anpassungsprozesses erstellt werden, ist FEL.#CUST.

```

•
|      ADDGROUP (FEL) OWNER(IBMUSER) SUPGROUP(SYS1)
|      DATA('IBM Rational Developer for z Systems - HLQ STUB')
|
|      •
|
|      ADDSD 'FEL.*.**' UACC(READ)
|      DATA('IBM Rational Developer for z Systems')
|
|      •
|
|      PERMIT 'FEL.*.**' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
|
|      •
|
|      SETROPTS GENERIC(DATASET) REFRESH

```

#### Anmerkung:

- Schützen Sie FEL.SFELAUTH gegen Aktualisierungen, weil diese Datei eine APF-Berechtigung hat.
- Bei den Beispielbefehlen in dieser Veröffentlichung und im Job FELRACF wird vorausgesetzt, dass EGN (Enhanced Generic Naming) aktiv ist. Wenn EGN aktiv ist, kann das Qualifikationsmerkmal \*\* verwendet werden, um eine beliebige Anzahl von Qualifikationsmerkmalen in der Klasse DATASET darzustellen. Ersetzen Sie \*\* durch \*, wenn EGN auf Ihrem System nicht aktiv ist. Weitere Informationen zu EGN finden Sie in *Security Server RACF Security Administrator's Guide* (IBM Form SA22-7683).

Einige der Komponenten von Developer for z Systems erfordern zusätzliche Sicherheitsdateipprofile. Ersetzen Sie die Platzhalter #sysprog und #ram-developer durch gültige Benutzer-IDs oder RACF-Gruppennamen.

- Wenn die Umsetzung langer/kurzer Namen des SCLM Developer Toolkit verwendet wird, benötigen Benutzer das Zugriffsrecht UPDATE für die Zuordnungs-VSAM FEL.#CUST.LSTRANS.FILE.

```

-
|      ADDSD 'FEL.#CUST.LSTRANS.*.**' UACC(UPDATE)
|      DATA('IBM Rational Developer for z Systems - SCLMDT')
|
|      -
|
|      PERMIT 'FEL.#CUST.LSTRANS.*.**' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
|
|      -
|
|      SETROPTS GENERIC(DATASET) REFRESH

```

- CARMA-RAM-Entwickler (Repository Access Manager) benötigen das Zugriffsrecht UPDATE für die CARMA-VSAMs (FEL.#CUST.CRA\*).

```

-
|      ADDSD 'FEL.#CUST.CRA*.*' UACC(READ)
|      DATA('IBM Rational Developer for z Systems - CARMA')
|
|      -
|
|      PERMIT 'FEL.#CUST.CRA*.*' CLASS(DATASET) ACCESS(ALTER) ID(#sysprog)
|
|      -
|
|      PERMIT 'FEL.#CUST.CRA*.*' CLASS(DATASET) ACCESS(UPDATE) ID(#ram-developer)
|
|      -
|
|      SETROPTS GENERIC(DATASET) REFRESH

```

## Sicherheitseinstellungen prüfen

Verwenden Sie die folgenden Beispielbefehle, um die Ergebnisse Ihrer Anpassungen in Bezug auf die Sicherheit anzuzeigen.

- Sicherheitseinstellungen und -klassen

	– SETROPTS LIST
	• Gestartete Tasks
	– LISTGRP STCGROUP OMVS
	– LISTUSER STCDBM OMVS
	– RLIST STARTED DBGMR.* ALL STDATA
	• Debug Manager als sicherer z/OS UNIX-Server
	– RLIST FACILITY BPX.SERVER ALL
	• Programmgesteuerte MVS-Bibliotheken für Debug Manager
	– RLIST PROGRAM ** ALL
	• Integrierter Debuggerzugriff
	– RLIST FACILITY AQE.** ALL
	• Dateiprofile
	– LISTGRP FEL
	– LISTDSD PREFIX(FEL) ALL



## Kapitel 3. Hinweise zu TCP/IP

Developer for z Systems verwendet TCP/IP, um Benutzern einer Workstation den Zugriff auf Mainframe-Computer bereitzustellen, wenn diese selbst kein Mainframe-Computer ist. TCP/IP wird außerdem für die Datenübertragung zwischen verschiedenen Komponenten und anderen Produkten verwendet.

Dieses Kapitel enthält die folgenden Abschnitte:

- „TCP/IP-Ports“
- „CARMA und TCP/IP-Ports“ auf Seite 27

Developer for z Systems baut auf IBM Explorer for z/OS auf. z/OS Explorer-spezifische Informationen finden Sie in „TCP/IP considerations“ im Handbuch *IBM Explorer for z/OS Host Configuration Reference* (IBM Form SC27-8438).

### TCP/IP-Ports

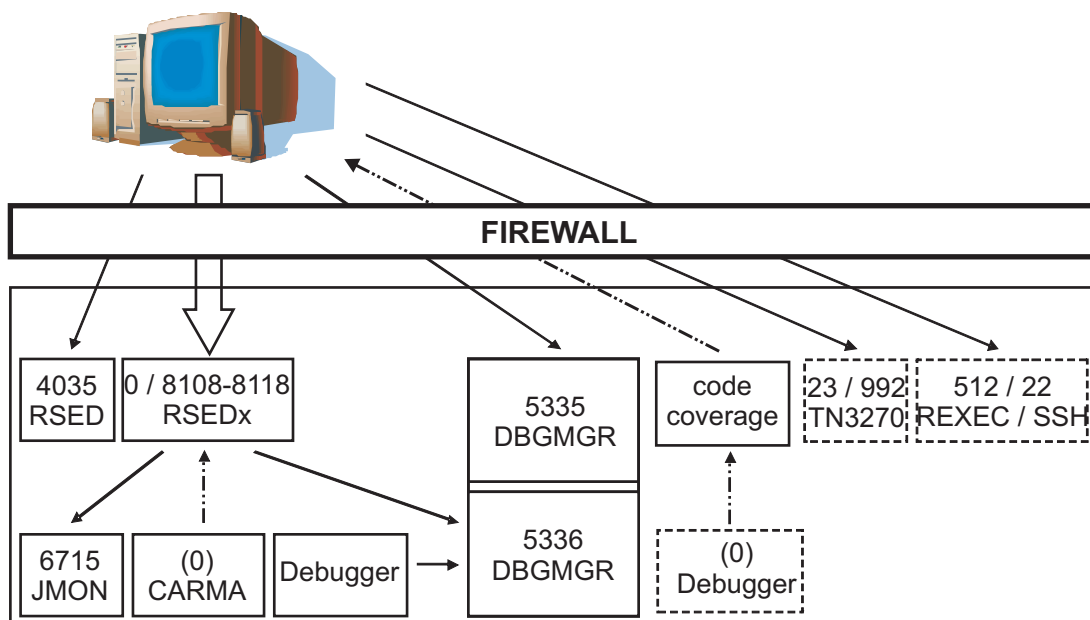


Abbildung 7. TCP/IP-Ports

Abb. 7 stellt die TCP/IP-Ports dar, die mit z/OS Explorer und Developer for z Systems verwendet werden können. Die Pfeilspitzen deuten an, welcher Teilnehmer für die Bindung (Pfeilspitzenseite) verantwortlich ist und welcher die Verbindung herstellt.

### Externe Kommunikation

Definieren Sie für die Firewall, die Ihren z/OS-Host schützt, die folgenden Ports für die Client-Host-Kommunikation (unter Verwendung des TCP-Protokolls):

- (z/OS Explorer) RSE-Dämon für die Einrichtung der Client-Host-Kommunikation (Standardport 4035). Der Port kann in der Konfigurationsdatei `rse.env` festgelegt werden. Die Kommunikation an diesem Port kann verschlüsselt sein.
- (z/OS Explorer) RSE-Server für die Einrichtung der Client-Host-Kommunikation. Standardmäßig kann jeder verfügbare Port verwendet werden. Mit der Definition `_RSE_PORTRANGE` in `rse.env` ist jedoch eine Einschränkung auf einen bestimmten Portbereich möglich. Der Standardportbereich für `_RSE_PORTRANGE` ist 8108-8118 (11 Ports). Die Kommunikation an diesem Port kann verschlüsselt sein.
- Debug Manager für Integrated Debugger-Services, Standardport 5335. Der Port kann in der gestarteten DBGMGR-Task-JCL festgelegt werden. Die Kommunikation an diesem Port kann verschlüsselt sein.
- Alle INETD-Services für ferne (hostbasierte) Aktionen in z/OS UNIX-Unterprojekten:
  - REXEC (z/OS UNIX-Version), Standardport 512
  - SSH (z/OS UNIX-Version), Standardport 22. Die Kommunikation über diesen Port ist verschlüsselt.
- (z/OS Explorer) TN3270-Telnet-Service für den Host-Connect-Emulator (Standardport 23). Die Kommunikation kann verschlüsselt sein (Standardport 992). Welcher Standardport dem Telnet-Service TN3270 zugeordnet wird, hängt davon ab, ob der Benutzer sich für oder gegen die Verwendung der Verschlüsselung entscheidet.
- Die hostbasierte Codeabdeckung kann angewiesen werden, eine Verbindung zur Integrated Debugger-Engine eines Clients von Developer for z Systems herzustellen. Die Kommunikation an diesem Port kann verschlüsselt sein. Beachten Sie in diesem Szenario, dass der z/OS-basierte Kollektor für Codeabdeckung ein Client für TCP/IP ist und die Integrated Debugger-Engine auf dem Personal Computer des Benutzers ein Server für TCP/IP ist. Das Standardszenario ist das lokale Arbeiten mit IBM Debug Tool auf demselben Host.

**Anmerkung:** In der Regel gibt der Client an, welche TCP/IP-Adresse für die Verbindung zum Host verwendet werden soll. Um jedoch sicherzustellen, dass Debugsitzungen mit dem korrekten Host kommunizieren, wird mit Debug Manager der Client festgelegt, dessen TCP/IP-Adresse verwendet werden muss.

## Interne Kommunikation

Mehrere Hostservices von Developer for z Systems werden in gesonderten Threads oder Adressräumen ausgeführt und verwenden TCP/IP-Sockets als Kommunikationsmechanismus, unter Verwendung der Loopback-Adresse Ihres Systems. Alle diese Services nutzen RSE für die Kommunikation mit dem Client und beschränken ihren Datenstrom nur auf den Host. Für einige Services kann jeder verfügbare Port verwendet werden. Für andere kann der Systemprogrammierer wie folgt auswählen, welcher Port oder Portbereich verwendet werden soll:

- JES Job Monitor für JES-bezogene Services, Standardport 6715. Der Port kann im Konfigurationsmember `FEJJCNFG` gesetzt werden und wird in der Konfigurationsdatei `rse.env` wiederholt.
- (Optional) Die CARMA-Kommunikation verwendet standardmäßig einen ephemeren Port. In der Konfigurationsdatei `CRASRV.properties` kann jedoch ein Portbereich festgelegt werden.

- (Optional) Debug Manager für Debug-bezogene Services, Standardport 5336. Der Port kann in der gestarteten DBGMR-JCL festgelegt werden.
- Bei der hostbasierten Codeabdeckung, einem Batch-Job, wird ein ephemerer Port so zugeordnet, dass das IBM Debug Tool for z/OS mit ihm kommuniziert und die für den Codeabdeckungsbericht benötigten Daten liefert.

## TCP/IP-Portreservierung

Wenn Sie die Anweisung PORT oder PORTRANGE in PROFILE.TCPIP verwenden, um die Ports zu reservieren, die von z/OS Explorer und Developer for z Systems verwendet werden; beachten Sie, dass viele Binds von Threads gemacht werden, die in einem RSE-Thread-Pool aktiv ist. Der Jobname des RSE-Thread-Pools lautet RSEDx, wobei RSED der Name der gestarteten RSE-Task und x eine zufällige Ziffer ist. Daher sind in der Definition Platzhalter erforderlich.

```
PORT      4035      TCP RSED   ; z/OS Explorer - RSE daemon
PORT      6715      TCP JMON   ; z/OS Explorer - JES job monitor
PORT      5335      TCP DBGMR  ; Developer for z Systems - Integrated debugger
PORT      5336      TCP DBGMR  ; Developer for x Systems - Integrated debugger
PORTRange 8108 11   TCP RSED*  ; z/OS Explorer - RSE_PORTRANGE
;PORTRange 5227 100 TCP RSED*  ; Developer for z Systems - CARMA
```

## CARMA und TCP/IP

### CARMA und TCP/IP-Ports

CARMA (Common Access Repository Manager) wird für den Zugriff auf einen hostbasierten Software Configuration Manager (SCM) verwendet, beispielsweise CA Endevor® SCM. In den meisten Fällen, beispielsweise bei einem RSE-Dämon, bindet ein Server an einen Port und wartet auf Verbindungsanforderungen. CARMA verwendet eine andere Methode, da der CARMA-Server während des Initialisierens der Verbindungsanforderung durch den Client noch nicht aktiv ist.

Wenn der Client eine Verbindungsanforderung sendet, fordert der CARMA-Miner, der als Benutzerthread in einem RSE-Thread-Pool aktiv ist, einen ephemeren Port an oder sucht in dem Bereich, der in der Konfigurationsdatei CRASRV.properties angegeben ist, nach einem freien Port und bindet an diesen Port. Der Miner startet anschließend den CARMA-Server und übergibt die Portnummer, sodass der Server eine Verbindung zu dem entsprechenden Port herstellen kann. Wenn der Server mit dem Port verbunden ist, kann der Client Anforderungen an den Server senden und Ergebnisse empfangen.

Aus Sicht des TCP/IP ist also RSE (über den CARMA-Miner) der Server, der an einen Port bindet, und der CARMA-Server der Client, der eine Verbindung mit dem Server herstellt.

Wenn Sie die Anweisung PORT oder PORTRANGE in PROFILE.TCPIP verwenden, um den Portbereich zu reservieren, der von CARMA verwendet wird, beachten Sie, dass der CARMA-Miner in einem RSE-Thread-Pool aktiv ist. Der Jobname des RSE-Thread-Pools lautet RSEDx, wobei RSED der Name der gestarteten RSE-Task und x eine zufällige Ziffer ist. Daher sind in der Definition Platzhalter erforderlich.

```
PORTRange 5227 100 RSED*      ; DEVELOPER FOR Z SYSTEMS - CARMA
```

**Anmerkung:** Der zu CARMA gehörende IVP-Test fekfivpc schlägt fehl, wenn Sie die CARMA-Ports für die Verwendung durch die RSE-Adressräume reservieren. Damit müssen Sie rechnen, weil das Installationsprüfprogramm (IVP - Installation

Verification Program) im Adressraum der Person ausgeführt wird, die das IVP ausführt (und nicht im RSE-Adressraum) und die Bindungsanforderung durch TCP/IP fehlschlägt.

## CARMA und Stackaffinität

CARMA (Common Access Repository Manager) wird für den Zugriff auf einen hostbasierten Software Configuration Manager (SCM) verwendet, beispielsweise CA Endeavor® SCM. Dazu startet CARMA einen benutzerspezifischen Server, der eine zusätzliche Konfiguration erfordert, um Stackaffinität umzusetzen.

Ähnlich den gestarteten z/OS Explorer- und Developer for z Systems-Tasks wird Stackaffinität für einen CARMA-Server mit der Variable `_BPXK_SETIBMOPT_TRANSPORT` festgelegt, die an LE (Language Environment) weitergegeben werden muss. Dies erfolgt durch Anpassung des Startbefehls in der aktiven Konfigurationsdatei `crastart*.conf` oder `CRASUB*`.

### Anmerkung:

- Der genaue Name der Konfigurationsdatei, in der der Startbefehl enthalten ist, hängt von verschiedenen, vom Systemprogrammierer, der CARMA konfiguriert hat, getroffenen Auswahlmöglichkeiten ab. Weitere Informationen hierzu finden Sie in "Kapitel 3. Common Access Repository Manager (optional)" im Handbuch *Hostkonfiguration* (IBM Form SC43-2896).
- `_BPXK_SETIBMOPT_TRANSPORT` gibt den Namen des zu verwendenden TCP/IP-Stacks an und wird in der Anweisung `TCPIPJOBNAME` in der zugehörigen Datei `"TCPIP.DATA"` definiert.
- Die Codierung einer SYSTCPD DD-Anweisung legt nicht die erforderliche Stackaffinität fest.
- Standardmäßig verwendet CARMA nicht die herkömmlichen TCP/IP-Stacks. CARMA verwendet vielmehr die Loopback-Adresse für die Kommunikation zwischen dem CARMA-Miner und dem CARMA-Server. Hierdurch wird nicht nur die Sicherheit gesteigert (nur lokale Prozesse haben Zugriff auf die Loopback-Adresse), sondern wahrscheinlich auch vermieden, dass Stackaffinität zur CARMA-Kommunikation hinzugefügt werden muss.

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

Ersetzen Sie den folgenden Abschnitt:

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

durch Folgendes (dabei stellt TCPIP den gewünschten TCP/IP-Stack dar):

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

**Anmerkung:** CRASTART unterstützt keine Zeilenfortsetzung. Die zulässige Zeilenlänge ist jedoch nicht begrenzt.

### **CRASUB\***

Ersetzen Sie den folgenden Abschnitt:

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

durch Folgendes (dabei stellt TCPIP den gewünschten TCP/IP-Stack dar):

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

**Anmerkung:** Eine Jobübergabe begrenzt die Zeilenlänge auf 80 Zeichen. Sie können eine längere Zeile bei einem Leerzeichen ( ) umbrechen und ein Pluszeichen (+) am Ende der ersten Zeile verwenden, um zwei Zeilen zu verknüpfen.



---

## Kapitel 4. Hinweise zu WLM

Im Gegensatz zu herkömmlichen z/OS-Anwendungen ist Rational Developer for z Systems keine einzelne Anwendung, die von Workload Manager (WLM) auf einfache Weise erkannt wird. Developer for z Systems umfasst mehrere interagierende Komponenten, damit der Client auf die Host-Services und -daten zugreifen kann. Wie in Kapitel 1, „Wissenswertes zu Developer for z Systems“, auf Seite 3 beschrieben, sind einige dieser Services in verschiedenen Adressräumen aktiv und werden somit verschiedenen WLM-Klassifikationen zugeordnet.

Dieses Kapitel enthält die folgenden Abschnitte:

- „Klassifikation für Verarbeitungsprozesse“
- „Ziele festlegen“ auf Seite 33

Developer for z Systems baut auf IBM Explorer for z/OS auf. z/OS Explorer-spezifische Informationen finden Sie in „WLM considerations“ im Handbuch *IBM Explorer for z/OS Host Configuration Reference* (IBM Form SC27-8438).

---

### Klassifikation für Verarbeitungsprozesse

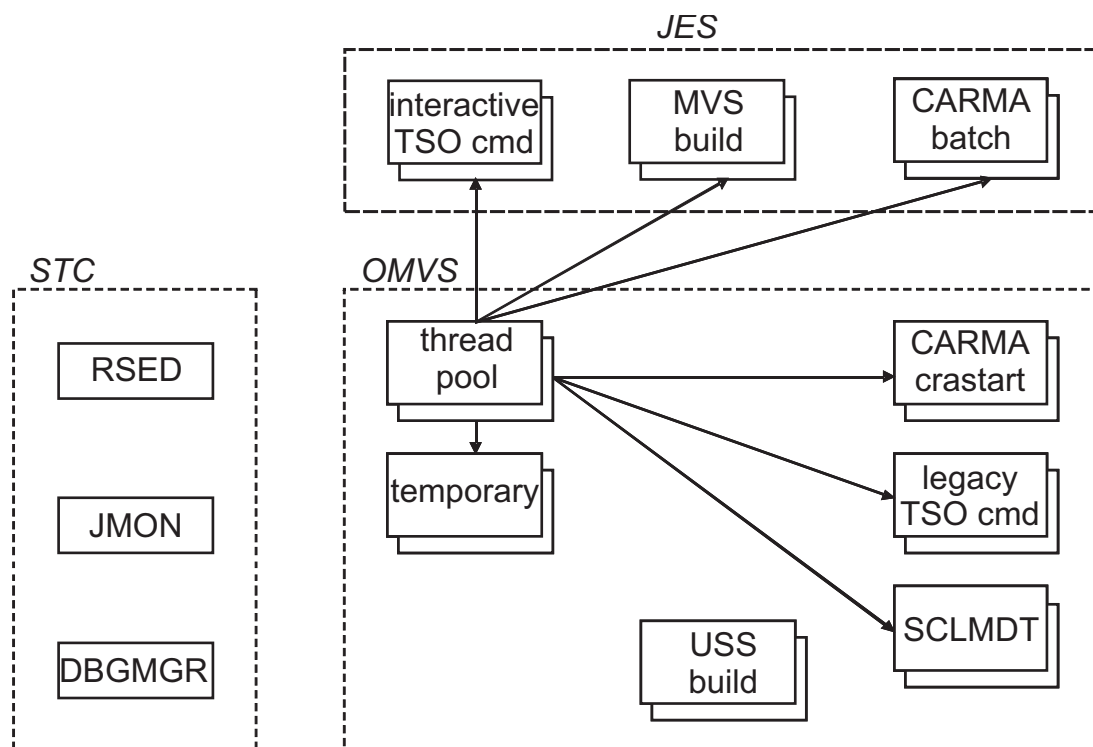


Abbildung 8. WLM-Klassifikation

Abb. 8 zeigt eine Basisübersicht über die Subsysteme, über die die Informationen zu den Verarbeitungsprozessen von z/OS Explorer und Developer for z Systems an WLM weitergegeben werden.

Der RSE-Dämon (RSED), Debug Manager (DBGMGR) und JES Job Monitor (JMON) sind gestartete Tasks in z/OS Explorer and Developer for z Systems (oder lang laufende Batch-Jobs) mit individuellen Adressräumen.

Der RSE-Dämon startet für jeden RSE-Thread-Pool-Server (der eine variable Anzahl von Clients unterstützt) einen untergeordneten Prozess. Jeder Thread-Pool ist (mithilfe eines z/OS UNIX-Initiators, BPXAS) in einem separaten Adressraum aktiv. Da es sich hierbei um gestartete Prozesse handelt, werden diese nach den WLM-OMVS-Klassifikationsregeln und nicht nach den Klassifikationsregeln für gestartete Tasks klassifiziert.

Abhängig von den Aktionen der Benutzer können die Clients, die in einem Thread-Pool aktiv sind, eine Vielzahl anderer Adressräume erstellen. Abhängig von der Konfiguration von Developer for z Systems, können einige Verarbeitungsprozesse, wie TSO Commands Service (TSO cmd) oder CARMA, in anderen Subsystemen ausgeführt werden.

Die in Abb. 8 auf Seite 31 aufgeführten Adressräume bleiben für einen längeren Zeitraum im System sichtbar. Sie sollten allerdings beachten, dass z/OS UNIX so entwickelt wurde, dass es auch einige kurz andauernde, temporäre Adressräume gibt. Diese temporären Adressräume sind im OMVS-Subsystem aktiv.

Während die RSE-Thread-Pools dieselbe Benutzer-ID und einen ähnlichen Jobnamen wie der RSE-Dämon verwenden, gehören alle von einem Thread-Pool gestarteten Adressräume der Client-Benutzer-ID, die die Aktion anfordert. Die Client-Benutzer-ID wird außerdem als Teil des Jobnamens für alle vom Thread-Pool gestarteten OMVS-basierten Adressräume verwendet.

Weitere Adressräume werden von anderen Services erstellt, die Developer for z Systems verwendet, zum Beispiel z/OS UNIX-REXEC (USS-Build).

## Klassifikationsregeln

WLM verwendet Klassifikationsregeln, um im System eingehende Arbeit einer Serviceklasse zuzuordnen. Diese Klassifikation basiert auf Qualifikationsmerkmalen für Arbeit. Das erste (verbindliche) Merkmal ist der Subsystemtyp, der die Verarbeitungsanforderung empfängt. In Tabelle 5 werden die Subsystemtypen aufgeführt, die Verarbeitungsanforderungen von Developer for z Systems empfangen können.

*Tabelle 5. WLM-Einstiegspunkt-Subsysteme*

Subsystemtyp	Beschreibung der Arbeit
ASCH	Die Verarbeitungsanforderungen umfassen alle APPC-Transaktionsprogramme, die von dem von IBM gelieferten APPC/MVS-Transaktionsscheduler (ASCH) geplant werden.
JES	Die Verarbeitungsanforderungen umfassen alle Jobs, die von JES2 oder JES3 initialisiert werden.
OMVS	Die Verarbeitungsanforderungen umfassen Arbeit, die in verzweigten untergeordneten Adressräumen von z/OS UNIX System Services verarbeitet wird.
STC	Die Verarbeitungsanforderungen umfassen Arbeit, die von den Befehlen 'START' und 'MOUNT' initialisiert wird. STC umfasst außerdem Adressräume der Systemkomponente.



Tabelle 6 listet zusätzliche Merkmale auf, die für die Zuordnung von Verarbeitungsprozessen zu einer bestimmten Serviceklasse verwendet werden können. Weitere Details zu den aufgelisteten Merkmalen enthält MVS Planning: Workload Management (IBM Form SA22-7602).

*Tabelle 6. WLM-Qualifikationsmerkmale für Arbeitsvorgänge*

		ASCH	JES	OMVS	STC
AI	Accountinformationen				
LU	LU-Name (*)				
PF	Ausführung (*)				
PRI	Priorität				
SE	Name der Terminierungsumgebung				
SSC	Objektgruppenname des Subsystems				
SI	Subsysteminstanz (*)				
SPM	Subsystemparameter				
PX	Sysplex-Name				
SY	Systemname (*)				
TC	Transaktions-/Jobklasse (*)				
TN	Transaktions-/Jobname (*)				
UI	Benutzer-ID (*)				

**Anmerkung:** Für die mit Stern (\*) markierten Merkmale können Klassifikationsgruppen angegeben werden, indem der Abkürzung des Typs ein 'G' hinzugefügt wird. Eine Gruppe für den Transaktionsnamen würde beispielsweise 'TNG' lauten.

## Ziele festlegen

Wie unter „Klassifikation für Verarbeitungsprozesse“ auf Seite 31 dokumentiert, erstellt Developer for z Systems unterschiedliche Typen von Verarbeitungsprozessen auf Ihrem System. Diese verschiedenen Tasks kommunizieren miteinander. Dafür ist die eigentliche Antwortzeit wichtig, um Zeitüberschreitungsprobleme bezüglich der Verbindungen zwischen den Tasks zu vermeiden. Deshalb sollten Tasks in Developer for z Systems in leistungsfähige Serviceklassen oder in Serviceklassen mit mittlerer Leistung mit hoher Priorität eingeordnet werden.

Es wird daher eine Überarbeitung und gegebenenfalls eine Aktualisierung Ihrer aktuellen WLM-Ziele empfohlen. Dies gilt insbesondere für herkömmliche MVS-Unternehmen, für die zeitkritische OMVS-Verarbeitungsprozesse neu sind.

### **Anmerkung:**

- Die Zielinformationen in diesem Abschnitt sind bewusst beschreibend gehalten, da die eigentlichen Leistungsziele sehr vom jeweiligen Standort abhängig sind.
- Um die Auswirkungen einer bestimmten Task auf Ihrem System besser zu verstehen, werden Bezeichnungen wie 'minimale Ressourcennutzung', 'mäßige Ressourcennutzung' und 'erhebliche Ressourcennutzung' verwendet. Diese

Angaben sind relativ zur Gesamtressourcennutzung von Developer for z Systems, nicht vom gesamten System, zu verstehen.

In Tabelle 7 werden die Adressräume aufgelistet, die von z/OS Explorer und Developer for z Systems. verwendet werden. z/OS UNIX ersetzt den Wert "x" in der Spalte "Taskname" durch eine zufällige einstellige Zahl.

*Tabelle 7. WLM-Verarbeitungsprozesse*

Beschreibung	Taskname	Verarbeitungsprozess
Debug Manager	DBGMGR	STC
(z/OS Explorer) JES Job Monitor	JMON	STC
(z/OS Explorer) RSE-Dämon	RSED	STC
(z/OS Explorer) RSE-Thread-Pool	RSEDx	OMVS
(ISPF) Interactive ISPF Gateway (TSO-Befehlsservice)	<Benutzer-ID>	JES
(ISPF) Legacy ISPF Gateway (TSO-Befehlsservice und SCLMDT)	<Benutzer-ID>x	OMVS
(z/OS Explorer) TSO Commands Service (APPC)	FEKFRSRV	ASCH
CARMA (batch)	CRA<Port>	JES
CARMA (crastart)	<Benutzer-ID>x	OMVS
CARMA (ISPF-Client-Gateway)	<Benutzer-ID> und <Benutzer-ID>x	OMVS
MVS-Build (Batch-Job)	*	JES
z/OS UNIX-Build (Shellbefehle)	<Benutzer-ID>x	OMVS
z/OS UNIX-Shell	<Benutzer-ID>	OMVS

## Hinweise zur Zielauswahl

Die folgenden allgemeinen Hinweise zu WLM unterstützen Sie beim Definieren der Zieldefinitionen für Developer for z Systems:

- Ihre Zieldefinitionen sollten darauf aufbauen, was tatsächlich erreicht werden kann, und nicht darauf, was Sie gern erreichen möchten. Wenn Sie Ziele höher als notwendig setzen, verschiebt WLM Ressourcen von Arbeitsvorgängen geringerer Wichtigkeit zu Arbeitsvorgängen größerer Wichtigkeit, die die Ressourcen möglicherweise gar nicht benötigen.
- Begrenzen Sie den Arbeitsbetrag, der den Serviceklassen "SYSTEM" und "SYSSTC" zugewiesen wird. Diese Klassen haben eine höhere Zuteilungspriorität als alle anderen von WLM verwalteten Klassen. Verwenden Sie diese Klassen für Arbeitsvorgänge, die sehr wichtig sind, aber eine geringe CPU-Auslastung verursachen.
- Arbeitsvorgänge, die den Klassifikationsregeln nicht entsprechen, werden der Klasse "SYSOTHER" zugeordnet. Diese Klasse verfolgt ein ressourcenabhängiges Ziel. Ein ressourcenabhängiges Ziel bewirkt, dass WLM im Fall freier Ressourcen die Arbeitsvorgänge dieser Klasse berücksichtigt.

Bei der Verwendung von Antwortzeitzielen:

- Damit WLM ein Antwortzeitziel erfolgreich steuern kann, muss eine stetige Taskrate eingehen (mindestens 10 Tasks in 20 Minuten).

- Verwenden Sie durchschnittliche Antwortzeitziele nur bei gut gesteuerten Verarbeitungsprozessen. Eine einzelne lange Transaktion hat eine erhebliche Auswirkung auf die durchschnittliche Antwortzeit und kann eine Überreaktion von WLM hervorrufen.

Bei der Verwendung von Geschwindigkeitszielen:

- Sie erreichen Geschwindigkeitsziele normalerweise nur zu 90 Prozent. Das hat verschiedene Ursachen. Die Adressräume "SYSTEM" und "SYSSTC" haben beispielsweise eine höhere Zuteilungspriorität als Geschwindigkeitsziele.
- WLM basiert seine Geschwindigkeitszielentscheidungen auf einer minimalen Anzahl von Stichproben. Je weniger Arbeit in einer Serviceklasse ausgeführt wird, umso länger dauert es, die erforderliche Anzahl von Stichproben zu sammeln und die Zuteilungsrichtlinie anzupassen.
- Überprüfen Sie Geschwindigkeitsziele erneut, wenn Sie Ihre Hardware ändern. Insbesondere der Einsatz von weniger und schnelleren Prozessoren erfordert Änderungen an den Geschwindigkeitszielen.

## STC

Alle von Developer for z Systems gestarteten Tasks bedienen Echtzeitclientanforderungen.

*Tabelle 8. WLM-Verarbeitungsprozesse - STC*

Beschreibung	Taskname	Verarbeitungsprozess
Debug Manager	DBGMGR	STC

- Debug Manager

Debug Manager bietet Services zur Herstellung einer Verbindung von Programmen, deren Debug ausgeführt wird, mit den Clients, die das Debugging ausführen. Sie sollten ein leistungsfähiges Geschwindigkeitsziel für einen Zeitraum angeben, da die Task WLM keine einzelnen Transaktionen zurückmeldet. Die Ressourcennutzung hängt im großen Maße von den Benutzeraktionen ab und kann deshalb schwanken. Es ist jedoch zu erwarten, dass sie minimal ist.

## OMVS

Alle Verarbeitungsprozess verwenden die Client-Benutzer-ID als Grundlage für den Adressraumnamen. (z/OS UNIX ersetzt den Wert "x" in der Spalte "Taskname" durch eine zufällige einstellige Zahl.)

Die Verarbeitungsprozesse werden alle aufgrund einer allgemeinen Namenskonvention für Adressräume derselben Serviceklasse zugeordnet. Für diese Serviceklasse sollten Sie ein Ziel für mehrere Zeiträume angeben. Für die ersten Zeiträume sollten Sie leistungsfähige Perzentilantwortzeitziele und für den letzten Zeitraum ein Geschwindigkeitsziel mit mittlerer Leistung angeben. Einige Verarbeitungsprozesse, wie das ISPF-Client-Gateway, melden WLM einzelne Transaktionen zurück.

*Tabelle 9. WLM-Verarbeitungsprozesse - OMVS*

Beschreibung	Taskname	Verarbeitungsprozess
Legacy ISPF Gateway (TSO-Befehlsservice und SCLMDT)	<Benutzer-ID>x	OMVS
CARMA (crastart)	<Benutzer-ID>x	OMVS

Tabelle 9. WLM-Verarbeitungsprozesse - OMVS (Forts.)

Beschreibung	Taskname	Verarbeitungsprozess
CARMA (ISPF-Client-Gateway)	<Benutzer-ID> und <Benutzer-ID>x	OMVS
z/OS UNIX-Build (Shellbefehle)	<Benutzer-ID>x	OMVS
z/OS UNIX-Shell	<Benutzer-ID>	OMVS

- Legacy ISPF Gateway  
Das Legacy ISPF Gateway ist ein ISPF-Service, der von Developer for z Systems aufgerufen wird, um nicht interaktive TSO- und ISPF-Befehle auszuführen. Dazu gehören sowohl vom Client ausgegebene, explizite Befehle als auch von der Komponente SCLMDT von Developer for z Systems ausgegebene, implizite Befehle. Die Ressourcennutzung hängt im großen Maße von den Benutzeraktionen ab und kann deshalb schwanken. Es ist jedoch zu erwarten, dass sie minimal ist.
- CARMA  
CARMA ist ein optionaler Developer for z Systems-Server, der zur Interaktion mit hostbasierten SCMs (Software Configuration Managers), wie CA Endeavor<sup>®</sup> SCM, verwendet wird. Developer for z Systems lässt verschiedene Startmethoden für einen CARMA-Server zu. Einige davon werden als OMVS-Verarbeitungsprozess gehandhabt. Die Ressourcennutzung hängt im großen Maße von den Benutzeraktionen ab und kann deshalb schwanken. Es ist jedoch zu erwarten, dass sie minimal ist.
- z/OS UNIX-Build  
Wenn ein Client einen Build für ein z/OS UNIX-Projekt initialisiert, startet die z/OS UNIX-REXEC (oder SSH) eine Task, die zur Ausführung des Builds eine Reihe von z/OS UNIX-Shellbefehlen ausführt. Die Ressourcennutzung hängt im großen Maße von den Benutzeraktionen ab und kann deshalb schwanken. Es ist jedoch zu erwarten, dass sie mäßig bis erheblich ist (abhängig von der Größe des Projekts).
- z/OS UNIX-Shell  
Bei dieser Workload werden vom Client ausgegebene z/OS UNIX-Shellbefehle verarbeitet. Die Ressourcennutzung hängt im großen Maße von den Benutzeraktionen ab und kann deshalb schwanken. Es ist jedoch zu erwarten, dass sie minimal ist.

## JES

JES-verwaltete Batchprozesse werden von Developer for z Systems auf verschiedene Weisen verwendet. Die bekannteste Nutzung ist für MVS-Builds, für die ein Job übergeben und überwacht wird, um sein Ende zu bestimmen. Developer for z Systems kann jedoch auch einen CARMA-Server mit Batchübergabe starten und mit ihm über TCP/IP kommunizieren.

Tabelle 10. WLM-Verarbeitungsprozesse - JES

Beschreibung	Taskname	Verarbeitungsprozess
CARMA (batch)	CRA<Port>	JES
MVS-Build (Batch-Job)	*	JES

- CARMA

CARMA ist ein Developer for z Systems-Server, der zur Interaktion mit hostbasierten SCMs (Software Configuration Managers), wie CA Endevor® SCM, verwendet wird. Developer for z Systems lässt verschiedene Startmethoden für einen CARMA-Server zu. Einige davon werden als JES-Verarbeitungsprozess gehandhabt. Sie sollten ein leistungsfähiges Geschwindigkeitsziel für einen Zeitraum angeben, da die Task WLM keine einzelnen Transaktionen zurückmeldet. Die Ressourcennutzung hängt im großen Maße von den Benutzeraktionen ab und kann deshalb schwanken. Es ist jedoch zu erwarten, dass sie minimal ist.

- MVS-Build

Wenn ein Client einen Build für ein MVS-Projekt initiiert, startet Developer for z Systems einen Batch-Job zur Ausführung des Builds. Die Ressourcennutzung hängt im großen Maße von den Benutzeraktionen ab und kann deshalb schwanken. Es ist jedoch zu erwarten, dass sie mäßig bis erheblich ist (abhängig von der Größe des Projekts). Abhängig von Ihren lokalen Bedingungen können verschiedene Zielstrategien mit mittlerer Leistung sinnvoll sein.

- Sie können ein Ziel für mehrere Zeiträume angeben: Für den ersten Zeitraum legen Sie ein Perzentilantwortzeitziel und für den zweiten Zeitraum ein Geschwindigkeitsziel fest. In diesem Fall sollten Ihre Entwickler hauptsächlich dieselbe Buildprozedur und Eingabedateien ähnlicher Größe verwenden, um Jobs mit einheitlichen Antwortzeiten zu erstellen. Damit WLM ein Antwortzeitziel erfolgreich steuern kann, muss auch eine stetige Jobrate eingehen (mindestens 10 Jobs in 20 Minuten).
- Ein Geschwindigkeitsziel ist für die meisten Batch-Jobs am besten geeignet, da dieses Ziel stark schwankende Ausführungszeiten und Eingangsraten handhaben kann.



---

## Kapitel 5. Push-to-Client-Aspekte

Push-to-Client bzw. die hostbasierte Clientsteuerung unterstützt die zentrale Verwaltung der folgenden Komponenten:

- Clientkonfigurationsdateien
- Clientproduktversion
- Projektdefinitionen

Dieses Kapitel enthält die folgenden Abschnitte:

- „Einführung“
- „Hostbasierte Projekte“ auf Seite 40

Developer for z Systems baut auf IBM Explorer for z/OS auf. z/OS Explorer-spezifische Informationen finden Sie in “Push-to-client considerations” im Handbuch *IBM Explorer for z/OS Host Configuration Reference (IBM Form SC27-8438)*.

---

### Einführung

Clients von Developer for z Systems können beim Verbindungsaufbau Clientkonfigurationsdateien und Informationen zu Produktaktualisierungen vom Host abrufen. Dadurch wird sichergestellt, dass alle Clients über die gleichen Einstellungen verfügen und auf dem neuesten Stand sind.

Der Clientadministrator kann mehrere Clientkonfigurationssätze und Clientaktualisierungsszenarios für die Anforderungen verschiedener Entwicklergruppen erstellen. Dadurch erhalten Benutzer eine angepasste Konfiguration, die auf Kriterien wie LDAP-Gruppenzugehörigkeit oder Zulassung zu einem Sicherheitsprofil basiert.

z/OS-Projekte können einzeln auf dem Client in der Perspektive für z/OS-Projekte erstellt werden. Sie können aber auch zentral auf dem Host erstellt werden, in welchem Fall sie auf Benutzerbasis auf den Client repliziert werden. Solche hostbasierten Projekte sind hinsichtlich Aussehen und Funktionsweise mit auf dem Client definierten Projekten identisch. Die Struktur, die Member und die Eigenschaften dieser Projekte können jedoch nicht vom Client geändert werden und sind nur bei bestehender Verbindung mit dem Host verfügbar.

Manager für Entwicklungsprojekte definieren ein Projekt und weisen ihm Entwickler zu.

Details zur Durchführung der zugewiesenen Aufgaben durch den Manager für Entwicklungsprojekte finden Sie im Developer for z Systems IBM Knowledge Center ([http://www-01.ibm.com/support/knowledgecenter/SSQ2R2/rdz\\_welcome.html](http://www-01.ibm.com/support/knowledgecenter/SSQ2R2/rdz_welcome.html)).

Wenn die Konfigurations- oder Versionssteuerung Unterstützung für mehrere Entwicklergruppen umfasst, ist ein weiteres Team für die Verwaltung von Push-to-Client erforderlich. Wie sich dieses Team zusammensetzt, richtet sich danach, welche Option zur Identifizierung der Gruppen ausgewählt wurde, denen die Entwickler zugeteilt werden können:

- Ein LDAP-Administrator verwaltet Gruppendefinitionen, durch die die Entwickler keiner, einer oder mehreren FEL.PTC.\*-LDAP-Gruppen zugewiesen werden können.
- Ein Sicherheitsadministrator verwaltet Zugriffslisten für FEL.PTC.\*-Sicherheitsprofile. Ein Entwickler kann Zugriff auf keines, eines oder mehrere dieser Profile haben.

---

## Hostbasierte Projekte

z/OS-Projekte können einzeln auf dem Client in der Perspektive für z/OS-Projekte erstellt werden. Sie können aber auch zentral auf dem Host erstellt werden, in welchem Fall sie auf Benutzerbasis auf den Client repliziert werden. Solche hostbasierten Projekte sind hinsichtlich Aussehen und Funktionsweise mit auf dem Client definierten Projekten identisch. Die Struktur, die Member und die Eigenschaften dieser Projekte können jedoch nicht vom Client geändert werden und sind nur bei bestehender Verbindung mit dem Host verfügbar.

Das Basisverzeichnis für hostbasierte Projekte wird vom Clientadministrator in `/var/rdz/pushtoclient/keymapping.xml` definiert. Standardmäßig lautet es `/var/rdz/pushtoclient/projects`.

Zur Konfiguration hostbasierter Projekte definiert der Projektmanager bzw. der leitende Entwickler die folgenden Typen von Konfigurationsdateien. Bei all diesen Dateien handelt es sich um UTF-8-codierte XML-Dateien.

- Projektinstanzdateien sind spezifisch für bestimmte Benutzer-IDs und verweisen auf wiederverwendbare Projektdefinitionsdateien. Jeder Benutzer, der mit hostbasierten Projekten arbeitet, benötigt sein eigenes Unterverzeichnis `/var/zexpl/pushtoclient/projects/<userid>/` mit einer Projektinstanzdatei (`*.hbpin`) für jedes herunterzuladende Projekt.
- Projektdefinitionsdateien legen die Struktur und den Inhalt eines Projekts fest und sind für mehrere Benutzer wiederverwendbar. Diese Dateien (`*.hbppd`) listen die im Projekt enthaltenen Unterprojekte auf. Sie befinden sich im Verzeichnis mit der Stammprojektdefinition oder in einem seiner Unterverzeichnisse.
- Unterprojektdefinitionsdateien legen die Struktur und den Inhalt eines Unterprojekts fest und sind für mehrere Benutzer wiederverwendbar. Diese Dateien (`*.hbpsd`) legen den für die Erstellung eines einzelnen Lademoduls erforderlichen Ressourcensatz fest. Sie befinden sich im Verzeichnis mit der Stammprojektdefinition oder in einem seiner Unterverzeichnisse.
- Unterprojekteigenschaftendateien sind Eigenschaftendateien mit Unterstützung für Variablensubstitution. Sie können daher von mehreren Unterprojekten verwendet werden. Diese Dateien (`*.hbppr`) unterstützen die Variablensubstitution und ermöglichen somit die gemeinsame Verwendung der Eigenschaftendateien durch mehrere Benutzer. Sie befinden sich im Verzeichnis mit der Stammprojektdefinition oder in einem seiner Unterverzeichnisse.

Hostbasierte Projekte können auch in die Konfiguration für mehrere Gruppen integriert werden. Dies bedeutet, dass hostbasierte Projekte auch in `/var/rdz/pushtoclient/grouping/<devgroup>/projects/` definiert werden können.

Wenn ein Arbeitsbereich an eine bestimmte Gruppe gebunden ist und für einen Benutzer dieser Gruppe und in der Standardgruppe Projektdefinitionen vorliegen, erhält der Benutzer die Projektdefinitionen sowohl aus der Standardgruppe als auch aus der spezifischen Gruppe.



---

## Kapitel 6. CICS-Aspekte

In diesem Kapitel werden Referenzen auf Komponenten von Developer for z Systems, die in CICS-Regionen ausgeführt werden können, in einer Gruppe zusammengefasst.

---

### Unterstützung bidirektionaler Sprachen

Weitere Informationen zur Unterstützung bidirektionaler Sprachen finden Sie im Abschnitt "Unterstützung bidirektionaler Sprachen für CICS" im Kapitel "Weitere Anpassungstasks" des Handbuchs *Rational Developer for z Systems Hostkonfiguration* (SC43-2896).

---

### IRZ-Diagnosenachrichten für Enterprise Service Tools

Weitere Informationen zu IRZ-Diagnosenachrichten für Enterprise Service Tools finden Sie im Abschnitt "IRZ-Diagnosenachrichten für Enterprise Service Tools" im Kapitel "Weitere Anpassungstasks" im Handbuch *Rational Developer for z Systems Hostkonfiguration* (SC43-2896).

---

### CICS-Transaktionsdebugging

Weitere Informationen zum CICS-Transaktionsdebugging finden Sie im Abschnitt "CICS-Aktualisierungen für Integrated Debugger" im Kapitel "Integrated Debugger (optional)" des Handbuchs *IBM Rational Developer for z Systems Hostkonfiguration* (SC12-4062).



---

## Kapitel 7. AT-TLS konfigurieren

Dieser Abschnitt ist zur Unterstützung bei einigen allgemeinen Problemen vorgesehen, die beim Konfigurieren von Application Transparent Transport Layer Security (AT-TLS) oder beim Überprüfen oder Ändern einer bestehenden Konfiguration auftreten können.

Das TLS-Protokoll (Transport Layer Security), das in RFC 2246 definiert wird, bietet Datenschutz für die Kommunikation im Internet. Ähnlich wie sein Vorgänger Secure Socket Layer (SSL) ermöglicht es dieses Protokoll Client- und Serveranwendungen, auf eine Art und Weise zu kommunizieren, die das Ausspionieren, die Manipulation und das Fälschen von Nachrichten verhindert. Application Transparent Transport Layer Security (AT-TLS) konsolidiert die TLS-Implementierung für z/OS-basierte Anwendungen an einer einzigen Position, sodass alle Anwendungen die TLS-basierte Verschlüsselung unterstützen können, ohne das TLS-Protokoll zu kennen. Weitere Informationen zu AT-TLS enthält das Dokument *Communications Server IP Configuration Guide* (IBM Form SC31-8775).

Integrated Debugger in Developer for z Systems benötigt AT-TLS für die verschlüsselte Kommunikation mit dem Client, da die Daten für die Debugsitzung nicht durch dieselbe Pipe geleitet werden wie die übrige Client-Host-Kommunikation in Developer for z Systems.

Welche Aktionen für die Konfiguration von AT-TLS erforderlich sind, hängt von den genauen Anforderungen am jeweiligen Standort und von den am Standort verfügbaren Ressourcen ab.

Die Informationen in diesem Abschnitt zeigen, wie der TCP/IP Policy Agent konfiguriert wird, der AT-TLS verwaltet, und wie eine Richtlinie für die Verwendung durch Developer for z Systems Integrated Debugger auf einem z/OS 1.13-System mit Unterstützung für TLS V1.2 definiert wird.

1. „syslogd konfigurieren“ auf Seite 44
2. „AT-TLS-Konfiguration in PROFILE.TCPIP“ auf Seite 44
3. „Gestartete Task von Policy Agent“ auf Seite 45
4. „Konfiguration von Policy Agent“ auf Seite 45
5. „AT-TLS-Richtlinie“ auf Seite 46
6. „AT-TLS-Sicherheitsupdates“ auf Seite 48
7. „Aktivierung der AT-TLS-Richtlinie“ auf Seite 51

In diesem Abschnitt wird die folgende einheitliche Namenskonvention verwendet:

- Debug Manager-Port für die externe Kommunikation: 5335
- Debug Manager-Benutzer-ID: stcdbm
- Policy Agent-Benutzer-ID: pagent
- Zertifikat: dbgmgr
- Schlüssel- und Zertifikatsspeicher: dbgmgr.racf

Für einige der in den folgenden Abschnitten beschriebenen Tasks wird vorausgesetzt, dass Sie aktivierter z/OS UNIX-Benutzer sind. Zum Aktivieren

können Sie den TSO-Befehl **OMVS** absetzen. Verwenden Sie den Befehl **oedit**, um Dateien unter z/OS UNIX zu bearbeiten. Mit dem Befehl **exit** können Sie zu TSO zurückkehren.

---

## syslogd konfigurieren

In der TCP/IP-Dokumentation wird empfohlen, Policy Agent-Nachrichten in syslog unter z/OS UNIX zu schreiben, nicht in die Standardprotokolldatei. AT-TLS schreibt Nachrichten stets in syslog unter z/OS UNIX.

Für diesen Zweck muss der Dämon von syslog unter z/OS UNIX, **syslogd**, konfiguriert und aktiv sein. Außerdem benötigen Sie einen Mechanismus zum Steuern der Größe der Protokolldateien, die durch **syslogd** erstellt werden.

Mithilfe der folgenden Updates an der Beispielkonfigurationsdatei kann **syslogd** mit einem einfachen Mechanismus zur Protokolldateiverwaltung konfiguriert und gestartet werden (vorhandene Protokolle entfernen, wenn z/OS UNIX gestartet wird, und neue beim Start von **syslogd** erstellen).

- /etc/services

```
syslog      514/udp
```
- /etc/syslog.conf

```
# /etc/syslog.conf - control output of syslogd
# 1. all files with will be printed to /tmp/syslog.auth.log
auth.*      /tmp/syslog.auth.log
# 2. all error messages printed to /tmp/syslog.error.log
*.err       /tmp/syslog.error.log
# 3. all debug and above messages printed to /tmp/syslog.debug.log
*.debug     /tmp/syslog.debug.log
# The files named must exist before the syslog daemon is started,
# unless -c startup option is used
```
- /etc/rc

```
# Start the SYSLOGD daemon for logging
# (clean up old logs)
sed -n '/^#/!s/.* \\.*/\1/p' /etc/syslog.conf | xargs -i rm {}
# (create new logs and add userid of message sender)
_BPX_JOBNAME='SYSLOGD' /usr/sbin/syslogd -cuf /etc/syslog.conf &
sleep 5
```

---

## AT-TLS-Konfiguration in PROFILE.TCPIP

Die AT-TLS-Unterstützung wird über den Parameter TTLS in der Anweisung TCPCONFIG in der Datei PROFILE.TCPIP aktiviert. AT-TLS wird von Policy Agent verwaltet. Policy Agent muss aktiv sein, damit die AT-TLS-Richtlinie erzwungen werden kann. Da Policy Agent warten muss, bis TCP/IP aktiv ist, ist die Anweisung AUTOSTART in PROFILE.TCPIP eine gute Position zum Auslösen des Starts dieses Servers.

Diese Anforderungen führen zu folgenden Änderungen an der Datei PROFILE.TCPIP, die häufig TCPIP.TCPPARMS(TCPPROF) genannt wird.

```
TCPCONFIG TTLS          ; Required for AT-TLS
AUTOLOG
  PAGENT                ; POLICY AGENT, required for AT-TLS
ENDAUTOLOG
```

---

## Gestartete Task von Policy Agent

Wie bereits erwähnt, wird AT-TLS durch Policy Agent verwaltet. Diese Komponente wiederum kann als gestartete Task gestartet werden. Erstellen Sie mithilfe der folgenden JCL SYS1.PROCLIB(PAGENT) und verwenden Sie dazu die Standardkonfigurationsdatei sowie die empfohlene Protokollposition (SYSLOGD). Die erforderlichen Definitionen in Ihrer Sicherheitssoftware werden später erläutert.

```
//PAGENT  PROC PRM='-L SYSLOGD'                                * '' or '-L SYSLOGD'
//*
//* TCP/IP POLICY AGENT
//*                                     (PARM) (envar)
//* default cfg file: /etc/pagent.conf      (-C) (PAGENT_CONFIG_FILE)
//* default log file: /tmp/pagent.log       (-L) (PAGENT_LOG_FILE)
//* default log size: 300,3 (3x 300KB files) (PAGENT_LOG_FILE_CONTROL)
//*
//PAGENT  EXEC PGM=PAGENT,REGION=0M,TIME=NOLIMIT,
//          PARM='ENVAR("TZ=EST5DST")/&PRM'
//SYSPRINT DD SYSOUT=*
//SYSOUT DD SYSOUT=*
//*
```

---

## Konfiguration von Policy Agent

Policy Agent setzt TCP/IP-Richtlinien um, die vom TCP/IP-Administrator erstellt werden. Dabei werden Richtlinien für AT-TLS (als TTLS bezeichnet), aber auch für andere Services wie IPSec verwaltet. Policy Agent verwendet eine Konfigurationsdatei, um festzustellen, welche Richtlinien erzwungen werden müssen und wo diese zu finden sind. Die Standardkonfigurationsdatei ist /etc/pagent.conf, in der JCL der gestarteten Task von Policy Agent kann jedoch eine andere Position angegeben werden.

```
#
# TCP/IP Policy Agent configuration information.
#
TTLSConfig /etc/pagent.ttls.conf
# Specifies the path of a TTLS policy file holding stack specific
# statements.
#
#TcpImage TCPIP /etc/pagent.conf
# If no TcpImage statement is specified, all policies will be installed
# to the default TCP/IP stack.
#
#LogLevel 31
# The sum of the following values that represent log levels:
#  LOGL_SYSERR      1
#  LOGL_OBJERR      2
#  LOGL_PROTERR     4
#  LOGL_WARNING     8
#  LOGL_EVENT      16
#  LOGL_ACTION      32
#  LOGL_INFO        64
#  LOGL_ACNTING     128
#  LOGL_TRACE       256
# Log Level 31 is the default log loglevel.
#
#Codepage IBM-1047
# Specify the EBCDIC code page to be used for reading all configuration
# files and policy definition files. IBM-1047 is the default code page.
```

Diese Beispielkonfigurationsdatei gibt an, wo Policy Agent die TTLS-Richtlinie finden kann. Für andere Anweisungen werden Standardwerte von Policy Agent verwendet.

## AT-TLS-Richtlinie

Mithilfe einer TTLS-Richtlinie werden die gewünschten AT-TLS-Regeln beschrieben. Entsprechend der Definition in der Policy Agent-Konfigurationsdatei befindet sich die TTLS-Richtlinie in der Datei `/etc/pagent.ttls.conf`. Die erforderlichen Definitionen in Ihrer Sicherheitssoftware werden später erläutert.

In diesem Beispiel wird eine recht einfache, aus zwei Regeln bestehende Richtlinie beschrieben, mit der Unterstützung für SSL v3 inaktiviert und die Unterstützung für TLS v1, TLS v1.1 und TLS v1.2 für beide von Developer for z Systems Integrated Debugger, Debug Manager und Probe-Client unterstützten Kommunikationspfade aktiviert wird. Entsprechend der Definition in der Policy Agent-Konfigurationsdatei befindet sich die TTLS-Richtlinie in der Datei `/etc/pagent.ttls.conf`.

```
##
## TCP/IP Policy Agent AT-TLS configuration information.
##
##-----
TTLRule                                RDz_Debug_Manager
{
    LocalPortRange                      5335
    Direction                          Inbound
    TTLSGroupActionRef                  grp_Production
    TTLSEnvironmentActionRef            act_RDz_Debug_Manager
}
##-----
TTLSEnvironmentAction                  act_RDz_Debug_Manager
{
    HandshakeRole Server
    TTLSKeyRingParms
    {
        Keyring dbgmgr.racf             # Keyring must be owned by the Debug Manager
    }
    TTLSEnvironmentAdvancedParms
    {
        ## TLSV1.2 only for z/OS 2.1 and higher
        # TLSV1.2 On                     # TLSv1 & TLSv1.1 are on by default
        SSLV3 Off                       # disable SSLv3 }
    }
}
##-----
TTLRule                                RDz_Debug_Probe-Client
{
    RemotePortRange                    8001
    Direction                          Outbound
    TTLSGroupActionRef                  grp_Production
    TTLSEnvironmentActionRef            act_RDz_Debug_Probe-Client
}
##-----
TTLSEnvironmentAction                  act_RDz_Debug_Probe-Client
{
    HandshakeRole                      Client
    TTLSKeyRingParms
    {
        Keyring *AUTH/*                 # virtual key ring holding CA certificates
    }
    TTLSEnvironmentAdvancedParms
    {
        ## TLSV1.2 only for z/OS 2.1 and higher
        # TLSV1.2 On                     # TLSv1 & TLSv1.1 are on by default
```

```

}
}
##-----
TTLSGroupAction          grp_Production
{
    TTLSEnabled            On
## TLSv1.2zOS1.13 only for z/OS 1.13
    TTLSGroupAdvancedParmsRef TLSv1.2zOS1.13
    Trace                  3      # Log Errors to syslogd & IP joblog
#Trace                    254    # Log everything to syslogd
}
##-----
TTLSGroupAdvancedParms    TLSv1.2zOS1.13
{
    Envfile /etc/pagent.ttls.TLS1.2zOS1.13.env
}

```

Eine TTLS-Richtlinie ermöglicht eine große Bandbreite an Filtern, um anzugeben, in welchen Fällen eine Regel zutrifft.

Debug Manager ist ein Server, der am Port 5335 für eingehende Verbindungen von der Debug-Engine empfangsbereit ist. Diese Informationen werden in der Regel RDz\_Debug\_Manager erfasst.

Da für die verschlüsselte Kommunikation die Nutzung eines Serverzertifikats erforderlich ist, müssen Sie angeben, dass Policy Manager die Zertifikate des Schlüsselrings dbgmgr.racf verwenden muss, dessen Eigner die Benutzer-ID für die gestartete Task von Debug Manager ist. Standardmäßig ist die Unterstützung für TLS V1.2 inaktiviert, also wird sie durch diese Richtlinie explizit aktiviert. SSLv3.0 ist aufgrund bekannter Sicherheitslücken in diesem Protokoll explizit inaktiviert.

Wenn der Debug-Testmonitor mit der Option TEST(,,TCP/IP&&ipaddress%8001:\*) für die Language Environment (Language Environment - LE) gestartet wird, wird er angewiesen, den Debug Manager nicht zu verwenden, sondern den Developer for z Systems-Client am Port 8001 direkt zu kontaktieren. Aus einer TCP/IP-Perspektive bedeutet dies, dass der hostbasierte Debug-Testmonitor ein Client ist, der einen Server (die Debugbenutzerschnittstelle) im Developer for z Systems-Client kontaktiert. Diese Informationen werden in der Regel RDz\_Debug\_Probe-Client erfasst.

Da der Host ein TCP/IP-Client ist, benötigt der Richtlinienmanager eine Methode zum Validieren der von der Debugbenutzerschnittstelle bereitgestellten Serverzertifikate. In diesem Fall wird kein einheitlich benannter Schlüsselring für alle Benutzer verwendet, für die möglicherweise eine verschlüsselte Debugsitzung erforderlich wäre; stattdessen wird der virtuelle Schlüsselring (\*AUTH\*/\*) der RACF-CERTAUTH verwendet. Dieser virtuelle Schlüsselring enthält die öffentlichen Zertifikate von Zertifizierungsstellen (Certificate Authorities - CAs) und kann verwendet werden, wenn die Debugbenutzerschnittstelle ein von einer der akzeptierten CAs unterzeichnetes Serverzertifikat bereitstellt.

Beachten Sie, dass Sie für komplexere Richtlinien den IBM Konfigurationsassistenten für z/OS Communications Server verwenden sollten. Dabei handelt es sich um ein grafisch orientiertes Tool mit einer geführten Schnittstelle für die Konfiguration von auf Richtlinien basierenden TCP/IP-Netzfunktionen, das als Task in IBM z/OS Management Facility (z/OSMF) sowie als eigenständige Workstationanwendung verfügbar ist.

## Hinweise zu TLS V1.2

Die Unterstützung für TLS V1.2 ist ab z/OS 2.1 verfügbar und ist standardmäßig inaktiviert. Diese Richtlinie enthält den Befehl (TLSV1.2 On) zur expliziten Aktivierung der Unterstützung. Dieser Befehl ist jedoch auf Kommentar gesetzt, da das Zielsystem z/OS 1.13 verwendet.

Durch Anwenden der folgenden beiden APARs wird die Unterstützung für TLS V1.2 zu z/OS 1.13 hinzugefügt:

- APAR OA39422 für System SSL
- APAR PM62905 für Communications Server (AT-TLS)

z/OS 1.13 System SSL wird von AT-TLS für die Implementierung von mit TLS verschlüsselter Kommunikation verwendet und benötigt einige zusätzliche Parameter für die Unterstützung für TLS V1.2. Diese Parameter werden über die AT-TLS-Richtlinie bereitgestellt, indem die Datei /etc/pagent.ttls.TLS1.2zOS1.13.env mit System SSL-Umgebungsvariablen eingesetzt wird.

```
#
# Add TLSv1.2 support to AT-TLS
# requires z/OS 1.13 with OA39422 and PM62905
#
GSK_RENEGOTIATION=ALL
GSK_PROTOCOL_TLSV1_2=ON
```

---

## AT-TLS-Sicherheitsupdates

Für Ihre Sicherheitskonfiguration sind mehrere Updates erforderlich, damit AT-TLS ordnungsgemäß funktioniert. Dieser Abschnitt enthält RACF-Beispielfehle für die Ausführung der erforderlichen Konfiguration.

Wie in Abschnitt „Gestartete Task von Policy Agent“ auf Seite 45 erwähnt, wird zur Ausführung von Policy Agent eine gestartete Task verwendet. Dazu ist die Definition einer Benutzer-ID und eines Profils der gespeicherten Task in der Klasse STARTED erforderlich.

```
# define started task user ID
# BPX.DAEMON permit is required for non-zero UID
ADDUSER PAGENT DFLTGRP(SYS1) OMVS(UID(0) SHARED HOME('/')) +
  NAME('TCP/IP POLICY AGENT') NOPASSWORD

# define started task
RDEFINE STARTED PAGENT.* STDATA(USER(PAGENT) GROUP(SYS1)) +
  DATA('TCP/IP POLICY AGENT')

# refresh to make the changes visible
SETROPTS RACLIST(STARTED) REFRESH
```

Definieren Sie ein Profil mit dem Namen MVS.SERVMgr.PAGENT in der Klasse OPERCMDS und weisen Sie der Benutzer-ID PAGENT den Zugriff CONTROL darauf zu. Das Profil beschränkt, welche Benutzer Policy Agent starten können. Wenn das Profil nicht definiert und der Zugriff darauf über ein generisches Profil verhindert wird, kann PAGENT Policy Agent nicht starten, wodurch die Initialisierung des TCP/IP-Stacks verhindert wird.

```
# restrict startup of policy agent
RDEFINE OPERCMDS MVS.SERVMgr.PAGENT UACC(NONE) +
  DATA('restrict startup of policy agent')
```



```

PERMIT MVS.SERVGR.PAGENT CLASS(OPERCMDS) ACCESS(CONTROL) ID(PAGENT)

# refresh to make the changes visible
SETROPTS RACLIST(OPERCMDS) REFRESH

```

Wie in Abschnitt „AT-TLS-Konfiguration in PROFILE.TCPIP“ auf Seite 44 erwähnt, wird Policy Agent nach der Initialisierung von TCP/IP gestartet. Dies bedeutet, es gibt ein (kleines) Zeitfenster, in dem Anwendungen den TCP/IP-Stack ohne Erzwingen der TTLS-Richtlinie verwenden können. Definieren Sie das Profil EZB.INITSTACK.\*\* in der Klasse SERVAUTH, um den Zugriff auf den Stack während dieses Zeitfensters zu verhindern. Ausgenommen hiervon sind Anwendungen mit Lesezugriff (READ) auf das Profil. Sie müssen eine begrenzte Menge an Verwaltungsanwendungen für das Profil zulassen, um die vollständige Initialisierung des Stacks sicherzustellen. Dies wird im Abschnitt “TCP/IP stack initialization access control” des Handbuchs *Communications Server IP Configuration Guide* (SC31-8775) beschrieben.

**Anmerkung:** Der Policy Agent gibt die Nachricht EZD1586I zurück, wenn alle Richtlinien aktiv sind.

```

# block stack access between stack and AT-TLS availability
# SETROPTS GENERIC(SERVAUTH)
# SETROPTS CLASSACT(SERVAUTH) RACLIST(FACILITY)
# RDEFINE SERVAUTH EZB.INITSTACK.** UACC(NONE)
# Policy Agent
  PERMIT EZB.INITSTACK.** CLASS(SERVAUTH) ACCESS(READ) ID(PAGENT)
# OMROUTE daemon
  PERMIT EZB.INITSTACK.** CLASS(SERVAUTH) ACCESS(READ) ID(OMROUTE)
# SNMP agent and subagents
  PERMIT EZB.INITSTACK.** CLASS(SERVAUTH) ACCESS(READ) ID(OSNMPD)
  PERMIT EZB.INITSTACK.** CLASS(SERVAUTH) ACCESS(READ) ID(IOBSNMP)
# NAME daemon
  PERMIT EZB.INITSTACK.** CLASS(SERVAUTH) ACCESS(READ) ID(NAMED)

# refresh to make the changes visible
SETROPTS RACLIST(SERVAUTH) REFRESH

```

(Optional) Der z/OS UNIX-Befehl **pasearch** zeigt aktive Richtliniendefinitionen an. Definieren Sie das Profil EZB.PAGENT.\*\* in der Klasse SERVAUTH, um den Zugriff auf den Befehl **pasearch** einzuschränken.

```

# restrict access to pasearch command
# RDEFINE SERVAUTH EZB.PAGENT.** UACC(NONE) +
# DATA('restrict access to pasearch command')
# PERMIT EZB.PAGENT.** CLASS(SERVAUTH) ACCESS(READ) ID(tcadmin)

# refresh to make the changes visible
# SETROPTS RACLIST(SERVAUTH) REFRESH

```

Wie in Abschnitt „AT-TLS-Richtlinie“ auf Seite 46 erwähnt, benötigt Debug Manager ein Zertifikat, damit AT-TLS verschlüsselte Kommunikation für Debug Manager konfigurieren kann. Die folgenden Beispielbefehle erstellen ein neues Zertifikat mit dem Namen dbgmgr, das in einem RACF-Schlüsselring mit dem Namen dbgmgr.racf gespeichert wird. Sowohl das Zertifikat als auch der Schlüsselring haben den Eigner STCDBM, die Benutzer-ID der gespeicherten Task des Debug Managers.

```

# permit Debug Manager to access certificates
#RDEFINE FACILITY IRR.DIGTCERT.LIST UACC(NONE)
#RDEFINE FACILITY IRR.DIGTCERT.LISTRING UACC(NONE)
  PERMIT IRR.DIGTCERT.LIST CLASS(FACILITY) ACCESS(READ) ID(stcdbm)
  PERMIT IRR.DIGTCERT.LISTRING CLASS(FACILITY) ACCESS(READ) ID(stcdbm)

# refresh to make the changes visible

```

```

SETROPTS RACLIST(FACILITY) REFRESH

# create self-signed certificate
RACDCERT ID(stcddb) GENCERT SUBJECTSDN(CN('RDz Debug Manager') +
OU('RTP labs') O('IBM') L('Raleigh') SP('NC') C('US')) +
NOTAFTER(DATE(2015-12-31)) KEYUSAGE(HANDSHAKE) WITHLABEL('dbgmgr')

# (optional) additional steps required to use a signed certificate
# 1. create a signing request for the self-signed certificate
RACDCERT ID(stcddb) GENREQ (LABEL('dbgmgr')) DSN(dsn)
# 2. send the signing request to your CA of choice
# 3. check if the CA credentials (also a certificate) are already known
RACDCERT CERTAUTH LIST
# 4. mark the CA certificate as trusted
RACDCERT CERTAUTH ALTER(LABEL('CA cert')) TRUST
# or add the CA certificate to the database
RACDCERT CERTAUTH ADD(dsn) WITHLABEL('CA cert') TRUST
# 5. add the signed certificate to the database;
# this will replace the self-signed one
RACDCERT ID(stcddb) ADD(dsn) WITHLABEL('dbgmgr') TRUST
# Do NOT delete the self-signed certificate before replacing it.
# If you do, you lose the private key that goes with the certificate,
# which makes the certificate useless.

# create key ring
RACDCERT ID(stcddb) ADDRING(dbgmgr.racf)

# add certificate to key ring
RACDCERT ID(stcddb) CONNECT(LABEL('dbgmgr') +
RING(dbgmgr.racf) USAGE(PERSONAL) DEFAULT)

# additional step required to use a signed certificate
# 6. add CA certificate to key ring
RACDCERT ID(stcddb) CONNECT(CERTAUTH LABEL('CA cert') +
RING(dbgmgr.racf))

# refresh to make the changes visible
SETROPTS RACLIST(DIGTCERT) REFRESH

```

Die AT-TLS-Richtlinie dokumentiert die Verwendung des virtuellen Schlüsselsrings CERTAUTH zur Validierung des von der Debugbenutzerschnittstelle im Szenario 'Probe-Client' bereitgestellten Serverzertifikats. Dabei wird davon ausgegangen, dass der z/OS-Host dem von der Debugbenutzerschnittstelle verwendeten CA-Zertifikat vertraut.

```

# check if the CA credentials (also a certificate) are already known
RACDCERT CERTAUTH LIST
# mark the CA certificate as trusted
RACDCERT CERTAUTH ALTER(LABEL('CA cert')) TRUST
# or add the CA certificate to the database
RACDCERT CERTAUTH ADD(dsn) WITHLABEL('CA cert') TRUST

# refresh to make the changes visible
SETROPTS RACLIST(DIGTCERT) REFRESH

```

Prüfen Sie Ihre Konfiguration mithilfe der folgenden Befehle:

```

# verify started task setup
LISTGRP SYS1 OMVS
LISTUSER PAGENT OMVS
RLIST STARTED PAGENT.* ALL STDATA

# verify Policy Agent startup permission
RLIST OPERCMDS MVS.SERVCMGR.PAGENT ALL

# verify initstack protection
RLIST SERVAUTH EZB.INITSTACK.** ALL

```

```
# verify pasearch protection
RLIST SERVAUTH EZB.PAGENT.** ALL

# verify certificate setup
RACDCERT CERTAUTH LIST(LABEL('CA cert'))
RACDCERT ID(stcdbm) LIST(LABEL('dbgmgr'))
RACDCERT ID(stcdbm) LISTRING(dbgmgr.racf)
```

---

## Aktivierung der AT-TLS-Richtlinie

Die AT-TLS-Konfiguration ist nun abgeschlossen und die Richtlinie wird beim nächsten einleitenden Programmladen des Systems aktiviert. Führen Sie folgende Schritte aus, um mit der Verwendung der Richtlinie ohne einleitendes Programmladen zu beginnen:

1. Aktivieren Sie die AT-TLS-Unterstützung im TCP/IP-Stack.  
Erstellen Sie eine TCP/IP-Obeydatei, z. B. TCPIP.TCPPARMS(OBEY), mit folgendem Inhalt:  
TCPCONFIG TTLS  
Aktivieren Sie diese Datei mit folgendem Bedienerbefehl:  
V TCPIP,,OBEY,TCPIP.TCPPARMS(OBEY)  
Prüfen Sie das Ergebnis, indem Sie folgende Konsolennachricht suchen:  
EZZ4249I stackname INSTALLED TTLS POLICY HAS NO RULES
2. Starten Sie Policy Agent.  
Geben Sie folgenden Bedienerbefehl aus:  
S PAGENT  
Prüfen Sie das Ergebnis, indem Sie folgende Konsolennachricht suchen:  
EZD1586I PAGENT HAS INSTALLED ALL LOCAL POLICIES FOR stackname
3. Starten Sie Debug Manager erneut, um alle aktiven, nicht verschlüsselten Sitzungen zu unterbrechen.  
Geben Sie folgende Bedienerbefehle aus:  
P DBGMGR  
S DBBMGR



# Literaturübersicht

## Referenzierte Veröffentlichungen

In diesem Dokument werden die folgenden Veröffentlichungen referenziert:

Tabelle 11. Referenzierte Veröffentlichungen

Titel der Veröffentlichung	Formnummer	Bezug	Referenzwebsite
Program Directory for IBM Rational Developer for z Systems	GI11-8298	Developer for z Systems	<a href="http://www-01.ibm.com/support/docview.wss?uid=swg27038517">http://www-01.ibm.com/support/docview.wss?uid=swg27038517</a>
Program Directory for IBM Rational Developer for z Systems Host Utilities	GI13-2864	Developer for z Systems	<a href="http://www-01.ibm.com/support/docview.wss?uid=swg27038517">http://www-01.ibm.com/support/docview.wss?uid=swg27038517</a>
IBM Rational Developer for z Systems Hostkonfiguration	SC27-8577	Developer for z Systems	<a href="http://www-01.ibm.com/support/docview.wss?uid=swg27038517">http://www-01.ibm.com/support/docview.wss?uid=swg27038517</a>
IBM Rational Developer for z Systems Hostkonfigurationsreferenz	SC27-8578	Developer for z Systems	<a href="http://www-01.ibm.com/support/docview.wss?uid=swg27038517">http://www-01.ibm.com/support/docview.wss?uid=swg27038517</a>
IBM Rational Developer for z Systems Common Access Repository Manager Developer's Guide	SC23-7660	Developer for z Systems	<a href="http://www-01.ibm.com/support/docview.wss?uid=swg27038517">http://www-01.ibm.com/support/docview.wss?uid=swg27038517</a>
SCLM Developer Toolkit Administrator's Guide	SC23-9801	Developer for z Systems	<a href="http://www-01.ibm.com/support/docview.wss?uid=swg27038517">http://www-01.ibm.com/support/docview.wss?uid=swg27038517</a>
IBM Explorer for z/OS Host Configuration Guide	SC27-8437	z/OS Explorer	
IBM Explorer for z/OS Host Configuration Reference	SC27-8438	z/OS Explorer	
Communications Server IP CICS Sockets Guide	SC31-8807	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 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>
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 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>
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>

Tabelle 11. Referenzierte Veröffentlichungen (Forts.)

Titel der Veröffentlichung	Formnummer	Bezug	Referenzwebsite
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>
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>

In diesem Dokument werden die folgenden Websites referenziert:

Tabelle 12. Referenzierte Websites

Beschreibung	Referenzwebsite
Developer for z Systems IBM Knowledge Center	<a href="http://www-01.ibm.com/support/knowledgecenter/SSQ2R2/rdz_welcome.html">http://www-01.ibm.com/support/knowledgecenter/SSQ2R2/rdz_welcome.html</a>
Developer for z Systems-Bibliothek	<a href="http://www-01.ibm.com/support/docview.wss?uid=swg27038517">http://www-01.ibm.com/support/docview.wss?uid=swg27038517</a>
Developer for z Systems-Homepage	<a href="http://www-03.ibm.com/software/products/en/developerforsystemz/">http://www-03.ibm.com/software/products/en/developerforsystemz/</a>
Empfohlener Service für Developer for z Systems	<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>
Verbesserungsvorschlag für Developer for z Systems	<a href="https://www.ibm.com/developerworks/support/rational/rfe/">https://www.ibm.com/developerworks/support/rational/rfe/</a>
Download von Apache Ant	<a href="http://ant.apache.org/">http://ant.apache.org/</a>

## Veröffentlichungen mit weiteren Informationen

Die folgenden Veröffentlichungen können Antworten auf Fragen enthalten, die vielleicht bei der Konfiguration der erforderlichen Hostsystemkomponenten auftreten.

Tabelle 13. Veröffentlichungen mit weiteren Informationen

Titel der Veröffentlichung	Formnummer	Bezug	Referenzwebsite
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>
System Programmer's Guide to: 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>

*Tabelle 13. Veröffentlichungen mit weiteren Informationen (Forts.)*

<b>Titel der Veröffentlichung</b>	<b>Formnummer</b>	<b>Bezug</b>	<b>Referenzwebsite</b>
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>
Tivoli Directory Server for z/OS	SG24-7849	Redbook	<a href="http://www.redbooks.ibm.com/">http://www.redbooks.ibm.com/</a>





---

# Glossar

## Aktions-ID

Eine numerische Kennung zwischen 0 und 999 für eine Aktion.

## Anwendungsserver

1. Ein Programm, das alle Anwendungsoperationen zwischen browserbasierten Computern und den Back-End-Geschäftsanwendungen oder -Datenbanken einer Organisation bearbeitet. Es gibt eine spezielle Klasse von Java-basierten Anwendungsservern, die dem Standard Java EE entsprechen. Java EE-Code kann ohne großen Aufwand zwischen diesen Anwendungsservern portiert werden. Diese Anwendungsserver können JSPs und Servlets für dynamischen Webinhalt und EJBs für Transaktionen und Datenbankzugriffe unterstützen.
2. Das Ziel einer Anforderung, die von einer fernen Anwendung stammt. In der DB2-Umgebung wird die Anwendungsserverfunktion von der Distributed Data Facility bereitgestellt und für den Zugriff auf DB2-Daten in fernen Anwendungen verwendet.
3. Ein Serverprogramm in einem verteilten Netz, das die Ausführungsumgebung für ein Anwendungsprogramm bereitstellt.
4. Das Ziel einer Anforderung, die von einem Anwendungsrequester stammt. Das Datenbankverwaltungssystem (DBMS) auf der Anwendungsserversite stellt die angeforderten Daten bereit.
5. Software, die die Kommunikation mit dem Client, der ein Asset anfordert, und Abfragen von Content Manager bearbeitet.

## Bidirektional (BIDI)

Bezeichnung für Scripts in Sprachen wie Arabisch und Hebräisch, die im Allgemeinen von rechts nach links geschrieben werden. Ausnahmen sind

Zahlen, die von links nach rechts geschrieben werden. Diese Definition stammt aus dem LISA-Glossar (Localization Industry Standards Association).

## Bidirektionales Attribut

Texttyp, Textausrichtung, numerischer Richtungswechsel und symmetrischer Richtungswechsel.

## Buildanforderung

Eine Anforderung eines Clients zum Ausführen einer Buildtransaktion.

## Buildtransaktion

Ein unter MVS gestarteter Job, der Builds erstellt, wenn vom Client eine Buildanforderung empfangen wird.

## Kompilieren

1. In ILE-Sprachen (Integrated Language Environment) das Umsetzen von Quellenanweisungen in Module, die anschließend in Programme oder Serviceprogramme eingebunden werden können.
2. Das Umsetzen eines vollständigen Programms oder von Teilen eines Programms, das in einer höheren Programmiersprache geschrieben ist, in ein Computerprogramm in IL, Assemblersprache oder Maschinensprache.

## Container

1. In CoOperative Development Environment/400 ein Systemobjekt, das Quellendateien enthält und organisiert. Beispiele für einen Container sind eine i5/OS-Bibliothek und eine partitionierte MVS-Datei.
2. In Java EE eine Entität, die Komponenten Sicherheits-, Deployment- und Laufzeitservices sowie Services für die Verwaltung des Lebenszyklus bereitstellt. (Sun) Jeder Containertyp (EJB, Web, JSP, Servlet, Applet und Anwendungsclient) stellt außerdem komponentenspezifische Services bereit.

3. In Backup Recovery and Media Services das physische Objekt, das zum Lagern und Umlagern von Datenträgern verwendet wird, wie z. B. Boxen, Schachteln oder Regale.
4. In einem Virtual Tape Server (VTS) ein Behälter, in dem exportierte logische Datenträger gespeichert werden können. Ein Stapel datenträger mit einem oder mehreren logischen Datenträger(n), der sich außerhalb einer VTS-Bibliothek befindet, wird als Container für diese Datenträger betrachtet.
5. Eine physische Speicherposition der Daten, z. B. eine Datei, ein Verzeichnis oder eine Einheit.
6. Eine Spalte oder Zeile, die verwendet wird, um das Layout eines Portlets oder anderer Container auf einer Seite zu gestalten.
7. Ein Element der Benutzerschnittstelle, das Objekte enthält. Im Ordnermanager ein Objekt, das andere Ordner oder Dokumente enthalten kann

#### **Datenbank**

Eine Sammlung von in Wechselbeziehung zueinander stehenden oder unabhängigen Datenelementen, die zur Bereitstellung für eine oder mehrere Anwendung(en) zusammen gespeichert werden.

#### **Datendefinitionssicht**

Enthält eine lokale Darstellung von Datenbanken und ihren Objekten und stellt Features für die Bearbeitung dieser Objekte und deren Export in eine ferne Datenbank bereit.

**Datei** Die Haupteinheit für das Speichern und Abrufen von Daten, die sich aus einer Sammlung von Daten in einer von mehreren vorgegebenen Zusammenstellungen zusammensetzt und durch Steuerinformationen beschrieben wird, auf die das System Zugriff hat.

#### **Debug**

Fehler in Programmen finden, diagnostizieren und beheben.

#### **Debugsitzung**

Die Debugaktivitäten, die in dem Zeitraum zwischen dem Starten eines

Debuggers durch den Entwickler und dem Beenden des Debuggers stattfinden.

#### **Fehlerpuffer**

Ein Teil des Speichers, in dem Fehlernachrichten vorübergehend gespeichert werden.

#### **Gateway**

1. Eine Middlewarekomponente, die eine Brücke zwischen Internet und Intranetumgebungen während Web-Service-Aufrufen bildet.
2. Software, die Services zwischen Endpunkten und dem Rest der Tivoli-Umgebung bereitstellt.
3. Eine Komponente eines Voice over Internet Protocol, die eine Brücke zwischen VoIP und Umgebungen mit Wählverbindungen darstellt.
4. Eine Einheit oder ein Programm, mit der bzw. dem Netze oder Systeme mit unterschiedlichen Netzarchitekturen miteinander verbunden werden können. Die Systeme können unterschiedliche Eigenschaften haben, z. B. unterschiedliche Kommunikationsprotokolle, unterschiedliche Netzarchitekturen oder unterschiedliche Sicherheitsrichtlinien. In diesem Fall übernimmt das Gateway sowohl eine Umsetzungs- als auch eine Verbindungsrolle.

#### **Interactive System Productivity Facility (ISPF)**

Ein IBM Lizenzprogramm, das als Gesamtanzeigeditor und Dialogmanager eingesetzt wird. Das Programm wird für das Schreiben von Anwendungsprogrammen verwendet und ermöglicht dem Benutzer, Standardanzeigen und Dialoge zwischen dem Anwendungsprogrammierer und dem Endbenutzer zu generieren. ISPF setzt sich aus vier Hauptkomponenten

zusammen: DM, PDF, SCLM und C/S.  
Die Komponente DM ist Dialog Manager, das die Services für Dialoge und Endbenutzer bereitstellt. Die Komponente PDF ist Program Development Facility, das Services für die Unterstützung von Dialog- und Anwendungsentwicklern bereitstellt. Die Komponente SCLM ist Software Configuration Library Manager, das Anwendungsentwicklern Services für die Verwaltung Ihrer Anwendungsentwicklungsbibliotheken bereitstellt. Die Komponente C/S ist die Client/Serverkomponente, die es Ihnen ermöglicht, ISPF auf programmierbaren Workstations auszuführen, um die Anzeigen mit der Anzeigefunktion des Workstationbetriebssystems anzuzeigen und Workstation-Tools und -daten in Host-Tools und -daten zu integrieren.

#### **Interpreter**

Ein Programm, das jede Instruktion einer höheren Programmiersprache übersetzt und ausführt, bevor es die nächste Instruktion übersetzt und ausführt.

#### **Isomorph**

Jedes zusammengesetzte Element (in anderen Worten jedes Element, das weitere Elemente enthält) des XML-Instanzdokuments hat ausgehend vom Stammverzeichnis genau ein entsprechendes COBOL-Gruppenelement, dessen Verschachtelungstiefe mit der Verschachtelungstiefe seines XML-Äquivalents identisch ist. Jedes nicht zusammengesetzte Element (in anderen Worten jedes Element, das keine weiteren Elemente enthält) im XML-Instanzdokument hat ausgehend vom Stamm genau ein entsprechendes Datenelement, dessen Verschachtelungstiefe mit der Verschachtelungstiefe seines XML-Äquivalents identisch ist und dessen Speicheradresse zur Laufzeit eindeutig identifiziert werden kann.

#### **LINKAGE SECTION**

Der Abschnitt im Datenteil einer aktivierten Einheit (einem aufgerufenen Programm oder einer aufgerufenen Methode), der Datenelemente beschreibt, die von der aktivierten Einheit (Programm oder Methode) zur Verfügung gestellt werden. Die aktivierte Einheit und die aktivierende Einheit können auf diese Datenelemente verweisen.

#### **Ladebibliothek**

Eine Bibliothek mit Lademodulen.

#### **Sperraktion**

Sperrt ein Member.

#### **Navigatorsicht**

Eine hierarchische Sicht der Ressourcen in der Workbench.

#### **Nicht isomorph**

Eine einfache Zuordnung von COBOL-Elementen und XML-Elementen, die zu XML-Dokumenten und COBOL-Gruppen gehören, die keine identische Form haben (nicht isomorph sind). Eine nicht isomorphe Zuordnung kann auch zwischen nicht isomorphen Elementen isomorpher Strukturen erstellt werden.

#### **Ausgabesicht der Konsole**

Zeigt die Ausgabe eines Prozesses an und ermöglicht Ihnen, über die Tastatur Eingaben an einen Prozess zu senden.

#### **Ausgabesicht**

Zeigt Nachrichten, Parameter und Ergebnisse an, die sich auf die von Ihnen bearbeiteten Objekte beziehen.

#### **Perspektive**

Eine Gruppe von Sichten, die verschiedene Aspekte der Ressourcen in der Workbench zeigen. Der Workbench-Benutzer kann - je nach auszuführender Task - die Perspektive wechseln und auch das Layout der Sichten und Editoren innerhalb einer Perspektive anpassen.

**RAM** Repository Access Manager.

**Fernes Dateisystem**

Ein Dateisystem, das sich auf einem anderen Server oder Betriebssystem befindet.

**Fernes System**

Jedes andere System im Netz, mit dem Ihr System kommunizieren kann.

**Perspektive für ferne Systeme**

Eine Schnittstelle für die Verwaltung ferner Systeme unter Einhaltung von Konventionen, die ISPF ähnlich sind.

**Repository**

1. Ein Speicherbereich für Daten. Jedes Repository hat einen Namen und einen zugehörigen Geschäftselementtyp. Standardmäßig ist der Repositoryname identisch mit dem Namen des Geschäftselements. Beispielsweise hat ein Repository für Rechnungen den Namen 'Rechnungen'. Es gibt zwei Typen von Informationsrepositories: lokale (prozessspezifische) und globale (wiederverwendbare) Repositories.
2. Eine VSAM-Datei, in der die Status von BTS-Prozessen gespeichert werden. Wenn ein Prozess nicht unter der Steuerung von BTS ausgeführt wird, werden der Prozessstatus (und die Status der zugehörigen Aktivitäten) erhalten, indem sie in eine Repository-Datei geschrieben werden. Die Status aller Prozesse eines bestimmten Prozesstyps (und der zugehörigen Aktivitätsinstanzen) werden in derselben Repository-Datei gespeichert. Es können Datensätze für mehrere Prozesstypen in dasselbe Repository geschrieben werden.
3. Ein permanenter Speicherbereich für Quellcode und andere Anwendungsressourcen. In einer Teamprogrammierungsumgebung ermöglicht ein gemeinsam benutztes Repository den Zugriff mehrerer Benutzer auf Anwendungsressourcen.

4. Eine Sammlung von Informationen über die Warteschlangenmanager, die zu einem Cluster gehören. Zu diesen Informationen gehören die Namen der Warteschlangenmanager, ihre Positionen, ihre Channel, die zugehörigen Warteschlangen usw.

**Repositoryinstanz**

Ein Projekt oder eine Komponente, das bzw. die in einem SCM-System vorhanden ist.

**Repositorysicht**

Zeigt die CVS-Repository-Positionen an, die Ihrer Workbench hinzugefügt wurden.

**Antwortdatei**

1. Eine Datei, die vordefinierte Antworten auf Fragen enthält, die ein Programm stellt. Die Antworten werden verwendet, sodass diese Werte nicht einzeln eingegeben werden müssen.
2. Eine ASCII-Datei, die mit Installations- und Konfigurationsdaten angepasst werden kann, die eine Installation automatisieren. Die Installations- und Konfigurationsdaten müssen während einer interaktiven Installation eingegeben werden, aber mit einer Antwortdatei kann die Installation ohne jeglichen Benutzereingriff durchgeführt werden.

**Serveransicht**

Zeigt eine Liste mit allen Servern und den zugehörigen Konfigurationen an.

**Shell**

Eine Softwareschnittstelle zwischen Benutzern und dem Betriebssystem, die Befehle und Benutzerinteraktionen interpretiert und diese an das Betriebssystem übermittelt. Ein Computer kann mehrere Schellebenen für unterschiedliche Ebenen von Benutzerinteraktionen haben.

**Shellname**

Der Name der Shellschnittstelle.

**Shell-Script**

Eine Datei mit Befehlen, die von der Shell interpretiert werden können. Der Benutzer gibt den Namen der Scriptdatei an der Shelleingabeaufforderung ein und

veranlasst die Shell damit, die Scriptbefehle auszuführen.

**Sidedeck**

Eine Bibliothek, in der die Funktionen eines DLL-Programms veröffentlicht werden. Die Eintrags- und Modulnamen werden nach der Kompilierung des Quellcodes in der Bibliothek gespeichert.

**Unbeaufsichtigte Installation**

Eine Installation, bei der keine Nachrichten an die Konsole gesendet, sondern Nachrichten und Fehler in Protokolldateien gespeichert werden. Bei einer unbeaufsichtigten Installation können Antwortdateien für die Dateneingabe verwendet werden.

**Unbeaufsichtigte Deinstallation**

Ein Deinstallationsprozess, bei dem keine Nachrichten an die Konsole gesendet werden, sondern Nachrichten und Fehler nach dem Aufruf des Deinstallationsbefehls in Protokolldateien gespeichert werden.

**Taskliste**

Eine Liste mit Prozeduren, die in einem Steuerungsablauf ausgeführt werden können.

**URL** Uniform Resource Locator.



---

## Bemerkungen

Die vorliegenden Informationen wurden für Produkte und Services entwickelt, die auf dem deutschen Markt angeboten werden. IBM stellt dieses Material möglicherweise auch in anderen Sprachen zur Verfügung. Für den Zugriff auf das Material in einer anderen Sprache kann eine Kopie des Produkts oder der Produktversion in der jeweiligen Sprache erforderlich sein.

Möglicherweise bietet IBM die in dieser Dokumentation beschriebenen Produkte, Services oder Funktionen in anderen Ländern nicht an. Informationen über die gegenwärtig im jeweiligen Land verfügbaren Produkte und Services sind beim zuständigen IBM Ansprechpartner erhältlich. Hinweise auf IBM Lizenzprogramme oder andere IBM Produkte bedeuten nicht, dass nur Programme, Produkte oder Services von IBM verwendet werden können. Anstelle der IBM Produkte, Programme oder Services können auch andere, ihnen äquivalente Produkte, Programme oder Services verwendet werden, solange diese keine gewerblichen oder anderen Schutzrechte von IBM verletzen. Die Verantwortung für den Betrieb von Produkten, Programmen und Services anderer Anbieter liegt beim Kunden.

Für in diesem Handbuch beschriebene Erzeugnisse und Verfahren kann es IBM Patente oder Patentanmeldungen geben. Mit der Auslieferung dieses Handbuchs ist keine Lizenzierung dieser Patente verbunden. Lizenzanforderungen sind schriftlich an folgende Adresse zu richten (Anfragen an diese Adresse müssen auf Englisch formuliert werden):

*IBM Director of Licensing  
IBM Europe, Middle East & Africa  
North Castle Drive, MD-NC119  
92066 Paris La Defense  
US*

Trotz sorgfältiger Bearbeitung können technische Ungenauigkeiten oder Druckfehler in dieser Veröffentlichung nicht ausgeschlossen werden. Die Angaben in diesem Handbuch werden in regelmäßigen Zeitabständen aktualisiert. Die Änderungen werden in Überarbeitungen oder in Technical News Letters (TNLs) bekannt gegeben. IBM kann ohne weitere Mitteilung jederzeit Verbesserungen und/oder Änderungen an den in dieser Veröffentlichung beschriebenen Produkten und/oder Programmen vornehmen.

Verweise in diesen Informationen auf Websites anderer Anbieter werden lediglich als Service für den Kunden bereitgestellt und stellen keinerlei Billigung des Inhalts dieser Websites dar. Das über diese Websites verfügbare Material ist nicht Bestandteil des Materials für dieses IBM Produkt. Die Verwendung dieser Websites geschieht auf eigene Verantwortung.

IBM kann alle von Ihnen bereitgestellten Informationen beliebig verwenden oder verteilen, ohne dass eine Verpflichtung gegenüber Ihnen entsteht.

Lizenznehmer des Programms, die Informationen zu diesem Produkt wünschen mit der Zielsetzung: (i) den Austausch von Informationen zwischen unabhängigen, erstellten Programmen und anderen Programmen (einschließlich des vorliegenden Programms) sowie (ii) die gemeinsame Nutzung der ausgetauschten Informationen zu ermöglichen, wenden sich an folgende Adresse:

*IBM Director of Licensing  
IBM Europe, Middle East & Africa  
North Castle Drive, MD-NC119  
92066 Paris La Defense  
US*

Die Bereitstellung dieser Informationen kann unter Umständen von bestimmten Bedingungen - in einigen Fällen auch von der Zahlung einer Gebühr - abhängig sein.

Die Lieferung des im Dokument aufgeführten Lizenzprogramms sowie des zugehörigen Lizenzmaterials erfolgt auf der Basis der IBM Rahmenvereinbarung bzw. der Allgemeinen Geschäftsbedingungen von IBM, der IBM Internationalen Nutzungsbedingungen für Programmpakete oder einer äquivalenten Vereinbarung.

Die aufgeführten Leistungsdaten und Clientbeispiele sind nur zur Veranschaulichung gedacht. Tatsächliche Leistungsergebnisse können je nach Konfiguration und Betriebsbedingungen variieren.

Alle Informationen zu Produkten anderer Anbieter stammen von den Anbietern der aufgeführten Produkte, deren veröffentlichten Ankündigungen oder anderen allgemein verfügbaren Quellen. IBM hat diese Produkte nicht getestet und kann daher keine Aussagen zu Leistung, Kompatibilität oder anderen Eigenschaften machen. Fragen zu den Leistungsmerkmalen von Produkten anderer Anbieter sind an den jeweiligen Anbieter zu richten.

Die oben genannten Erklärungen bezüglich der Produktstrategien und Absichtserklärungen von IBM stellen die gegenwärtige Absicht von IBM dar, unterliegen Änderungen oder können zurückgenommen werden und repräsentieren nur die Ziele von IBM.

Diese Veröffentlichung enthält Beispiele für Daten und Berichte des alltäglichen Geschäftsablaufs. Sie sollen nur die Funktionen des Lizenzprogramms illustrieren; sie können Namen von Personen, Firmen, Marken oder Produkten enthalten. Alle diese Namen sind frei erfunden; Ähnlichkeiten mit tatsächlichen Namen und Unternehmen sind rein zufällig.

#### **COPYRIGHTLIZENZ:**

Diese Veröffentlichung enthält Musteranwendungsprogramme, die in Quellsprache geschrieben sind und Programmier Techniken in verschiedenen Betriebsumgebungen veranschaulichen. Sie dürfen diese Musterprogramme in beliebiger Form kopieren, ändern und verteilen, ohne dass dafür Zahlungen an IBM anfallen, wenn dies zu dem Zweck geschieht, Anwendungsprogramme zu entwickeln, zu verwenden, zu vermarkten oder zu verteilen, die mit der Anwendungsprogrammierschnittstelle für die Betriebsumgebung konform sind, für die diese Musterprogramme geschrieben werden. Diese Beispiele wurden nicht unter allen denkbaren Bedingungen getestet. Daher kann IBM die Zuverlässigkeit, Wartungsfreundlichkeit oder Funktion dieser Programme weder zusagen noch gewährleisten. Die Beispielpprogramme werden auf der Grundlage des gegenwärtigen Zustands (auf "as-is"-Basis) und ohne Gewährleistung zur Verfügung gestellt. IBM haftet nicht für Schäden, die durch Verwendung oder im Zusammenhang mit diesen Beispielpprogrammen entstehen.



---

## Informationen zu Programmierschnittstellen

---

### Marken

IBM, das IBM Logo und [ibm.com](http://ibm.com) sind Marken oder eingetragene Marken der International Business Machines Corp. Weitere Produkt- und Servicenamen können Marken von IBM oder anderen Unternehmen sein. Eine aktuelle Liste der IBM Marken finden Sie im Web unter [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml) im Abschnitt "Copyright and trademark information".

---

## Nutzungsbedingungen für die Produktdokumentation

Die Berechtigung zur Nutzung dieser Veröffentlichungen wird Ihnen auf der Basis der folgenden Bedingungen gewährt.

### Anwendbarkeit

Diese Bedingungen sind eine Ergänzung der Nutzungsbedingungen auf der IBM Website.

### Persönliche Nutzung

Sie dürfen diese Veröffentlichungen für Ihre persönliche, nicht kommerzielle Nutzung unter der Voraussetzung vervielfältigen, dass alle Eigentumsvermerke erhalten bleiben. Sie dürfen diese Veröffentlichungen oder Teile der Veröffentlichungen ohne ausdrückliche Genehmigung von IBM weder weitergeben oder anzeigen noch abgeleitete Werke davon erstellen.

### Kommerzielle Nutzung

Sie dürfen diese Veröffentlichungen nur innerhalb Ihres Unternehmens und unter der Voraussetzung, dass alle Eigentumsvermerke erhalten bleiben, vervielfältigen, weitergeben und anzeigen. Sie dürfen diese Veröffentlichungen oder Teile der Veröffentlichungen ohne ausdrückliche Genehmigung von IBM außerhalb Ihres Unternehmens weder vervielfältigen, weitergeben oder anzeigen noch abgeleitete Werke davon erstellen.

### Rechte

Abgesehen von den hier gewährten Berechtigungen erhalten Sie keine weiteren Berechtigungen, Lizenzen oder Rechte (veröffentlicht oder stillschweigend) in Bezug auf die Veröffentlichungen oder darin enthaltene Informationen, Daten, Software oder geistiges Eigentum.

IBM behält sich das Recht vor, die in diesem Dokument gewährten Berechtigungen nach eigenem Ermessen zurückzuziehen, wenn sich die Nutzung der Veröffentlichungen für IBM als nachteilig erweist oder wenn die obigen Nutzungsbedingungen nicht genau befolgt werden.

Sie dürfen diese Informationen nur in Übereinstimmung mit allen anwendbaren Gesetzen und Verordnungen, einschließlich aller US-amerikanischen Exportgesetze und Verordnungen, herunterladen und exportieren.

IBM übernimmt keine Gewährleistung für den Inhalt dieser Veröffentlichungen. Diese Veröffentlichungen werden auf der Grundlage des gegenwärtigen Zustands

(auf "as-is"-Basis) und ohne eine ausdrückliche oder stillschweigende Gewährleistung für die Handelsüblichkeit, die Verwendungsfähigkeit oder die Freiheit der Rechte Dritter zur Verfügung gestellt.

---

## Copyrightlizenz

Diese Veröffentlichung enthält Musteranwendungsprogramme, die in Quellsprache geschrieben sind und Programmier Techniken in verschiedenen Betriebsumgebungen veranschaulichen. Sie dürfen diese Musterprogramme kostenlos kopieren, ändern und verteilen, wenn dies zu dem Zweck geschieht, Anwendungsprogramme zu entwickeln, zu verwenden, zu vermarkten oder zu verteilen, die mit der Anwendungsprogrammierschnittstelle für die Betriebsumgebung konform sind, für die diese Musterprogramme geschrieben werden. Diese Beispiele wurden nicht unter allen denkbaren Bedingungen getestet. Daher kann IBM die Zuverlässigkeit, Wartungsfreundlichkeit oder Funktion dieser Programme weder zusagen noch gewährleisten. Die Beispielprogramme werden auf der Grundlage des gegenwärtigen Zustands (auf "as-is"-Basis) und ohne Gewährleistung zur Verfügung gestellt. IBM haftet nicht für Schäden, die durch Verwendung oder im Zusammenhang mit den Beispielprogrammen entstehen.

---

## Marken

IBM, das IBM Logo und [ibm.com](http://ibm.com) sind Marken oder eingetragene Marken der International Business Machines Corp. Weitere Produkt- und Servicenamen können Marken von IBM oder anderen Unternehmen sein. Eine aktuelle Liste der IBM Marken finden Sie im Web unter [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml).

Adobe und PostScript sind Marken von Adobe Systems Incorporated.

Cell Broadband Engine - Sony Computer Entertainment Inc.

Rational ist eine Marke der International Business Machines Corporation und der Rational Software Corporation in den USA und/oder anderen Ländern.

Intel, Intel Centrino, Intel SpeedStep, Intel Xeon, Celeron, Itanium und Pentium sind Marken der Intel Corporation in den USA und/oder anderen Ländern.

IT Infrastructure Library ist eine Marke der Central Computer and Telecommunications Agency.

ITIL ist eine Marke des Cabinet Office (The Minister for the Cabinet Office).

Linear Tape-Open, LTO und Ultrium sind Marken von HP, IBM Corp. und Quantum.

Linux ist eine Marke von Linus Torvalds.

Microsoft, Windows und das Windows-Logo sind Marken oder eingetragene Marken der Microsoft Corporation in den USA und/oder anderen Ländern.

Java und alle Java-basierten Marken und Logos sind Marken oder eingetragene Marken von Sun Microsystems, Inc. in den USA und/oder anderen Ländern.

UNIX ist eine eingetragene Marke von The Open Group in den USA und anderen Ländern.

---

# Index

## A

Aktivierung der AT-TLS-Richtlinie 51  
Anwendungsschutz für RSE  
    definieren 19  
APF-Autorisierung  
    FELSFELAUTH 23  
Application Deployment Manager  
    (ADM) 4  
AQEZPCM 11  
Aspekte der Sicherheit 11  
AT-TLS-Konfiguration 43  
AT-TLS-Konfiguration,  
    PROFILE.TCPIP 44  
AT-TLS-Richtlinie 46  
AT-TLS-Sicherheitsupdates 48  
Authentifizierung, Debug Manager 11  
Authentifizierungsmethoden 11

## B

Befehlssicherheit definieren, JES 20  
Bidirektionale Sprachen,  
    Unterstützung 41

## C

CARMA und TCP/IP-Ports 27  
CICS-Transaktionsdebugging 41  
CICSTS-Aspekte 41  
CICSTS-Sicherheit 13

## D

Dateiprofile definieren 22  
Debug Manager-Authentifizierung 11  
Debug-Sicherheit 13  
Debugger, Integrated 6  
Debugging, CICS-Transaktion 41  
Definieren, Zugriff auf Integrated  
    Debugger 22  
Definieren der z/OS  
    UNIX-Dateizugriffsberechtigung für  
        RSE 19  
Definitionen, Sicherheit 14  
Developer for z Systems, gestartete Tasks  
    definieren 16  
Developer for z Systems,  
    Komponentenübersicht  
        grafische Darstellung 3  
Developer for z Systems,  
    Wissenswertes 3  
Diagnosenachrichten, IRZ 41

## E

Einführung, Push-to-Client-Aspekte 39  
Einstellungen und Klassen für Sicherheit  
    aktivieren 15  
Enterprise Service Tools 41

Externe Kommunikation 25

## F

FEJJCNFG 26  
FELRACE, Sicherheitsdefinitionen 14

## G

Gestartete Task, Policy Agent 45  
Gestartete Task von Policy Agent 45  
Gestartete Tasks für Developer for z  
    Systems definieren  
        JMON, gestartete Tasks 16  
        RSED, gestartete Tasks 16

## H

Hinweise zu WLM xii, 31  
Hostbasierte Projekte 40

## I

Integrated Debugger 6  
    verschlüsselte Kommunikation 12  
Interne Kommunikation 26  
IRZ-Nachrichten 41

## J

JES-Befehlssicherheit definieren 20  
JES Job Monitor (JMON) 4  
JMON 21

## K

Klassifikation für Verarbeitungsprozesse,  
    WLM 31  
Klassifikationsregeln, WLM 32  
Kommunikation, extern 25  
Kommunikation, intern 26  
Komponentenübersicht, Developer for z  
    Systems  
    grafische Darstellung 3  
Konfiguration von Policy Agent 45

## M

Methoden zur Authentifizierung 11  
MVS-Bibliotheken für RSE definieren 17

## O

OMVS-Segment definieren 16

## P

PassTicket-Unterstützung für RSE  
    definieren 18  
Portreservierung, TCP/IP 27  
Ports, TCP/IP 25  
Ports, TCP/IP und CARMA 27  
Profile für Dateien definieren 22  
PROFILE.TCPIP, AT-TLS-  
    Konfiguration 44  
Programmgesteuerte MVS-Bibliotheken  
    für RSE definieren 17  
Programmgesteuerte UNIX-Dateien für  
    RSE definieren 20  
Programmgesteuerte z/OS UNIX-Dateien  
    für RSE definieren 20  
Projekte, hostbasiert 40  
Prüfen von Sicherheitseinstellungen 23  
Push-to-Client-Aspekte 39

## R

Referenzierte Veröffentlichungen 53  
Reservierung, TCP/IP-Port 27  
RSE, Anwendungsschutz definieren 19  
RSE, PassTicket-Unterstützung  
    definieren 18  
RSE, programmgesteuerte  
    MVS-Bibliotheken definieren 17  
RSE, programmgesteuerte z/OS  
    UNIX-Dateien definieren 20  
RSE, z/OS UNIX-  
    Dateizugriffsberechtigung  
        definieren 19  
RSE als sicheren z/OS UNIX-Server  
    definieren 17  
RSE-Dämon 25  
RSE-Dämon (RSED) 4  
RSE-Server 25  
RSE-Server als sicheren z/OS  
    UNIX-Server definieren 17

## S

SCLM Developer Toolkit 17  
SCLM Developer Toolkit (SCLMDT) 4  
SCLM-Sicherheit 14  
Segment definieren, OMVS 16  
Sicherer z/OS UNIX-Server, RSE  
    definieren 17  
Sicherheit, CICSTS 13  
Sicherheit, Debug 13  
Sicherheit, SCLM 14  
Sicherheit für JES-Befehle definieren 20  
Sicherheit für Verbindungen 12  
Sicherheitsaspekte 11  
Sicherheitsdefinitionen 14  
Sicherheitsdefinitionen, Prüfliste 14  
Sicherheitseinstellungen prüfen 23  
Sicherheitseinstellungen und -klassen  
    aktivieren 15

- Sperddämon (LOCKD) 4
- Sprachunterstützung, bidirektional 41
- Subsystemtypen
  - ASCH 32
  - CICS 32
  - JES 32
  - OMVS 32
  - STC 32
- syslogd, Konfiguration 44

## T

- Taskeigner 4
- TCP/IP-Portreservierung 27
- TCP/IP-Ports 25
- TCP/IP-Ports, grafische Darstellung 25
- TLS V1.2, Hinweise 48
- TSO Commands Service 4

## U

- UNIX-Server, RSE definieren 17
- Unterstützung für RSE, PassTicket definieren 18

## V

- Verbindungssicherheit 12
- Veröffentlichungen, referenzierte 53
- Verschlüsselte Kommunikation
  - Integrated Debugger 12
- Verzeichnisstruktur, z/OS UNIX
  - grafische Darstellung 9

## W

- Wissenswertes zu Developer for z Systems 3
- WLM-Klassifikationsregeln 32
- Workload Manager 31

## Z

- z/OS UNIX-Dateizugriffsberechtigung, für RSE definieren 19
- z/OS UNIX-Server, RSE definieren 17
- z/OS UNIX-Verzeichnisstruktur
  - grafische Darstellung 9
- Ziele festlegen, WLM 33
- Ziele in WLM festlegen 33
- Zugriff auf Integrated Debugger definieren 22

---

# Antwort

IBM Rational Developer for z Systems  
Version 9.5.1  
Hostkonfigurationsreferenz

IBM Form SC43-2898-00

Anregungen zur Verbesserung und Ergänzung dieser Veröffentlichung nehmen wir gerne entgegen. Bitte informieren Sie uns über Fehler, ungenaue Darstellungen oder andere Mängel.

Zur Klärung technischer Fragen sowie zu Liefermöglichkeiten und Preisen wenden Sie sich bitte entweder an Ihre IBM Geschäftsstelle, Ihren IBM Geschäftspartner oder Ihren Händler.

**Unsere Telefonauskunft "HALLO IBM" (Telefonnr.: 0180 3 313233) steht Ihnen ebenfalls zur Klärung allgemeiner Fragen zur Verfügung.**

Kommentare:

Danke für Ihre Bemühungen.

Als Brief an die Postanschrift auf der Rückseite dieses Formulars

\_\_\_\_\_  
Name

\_\_\_\_\_  
Adresse

\_\_\_\_\_  
Firma oder Organisation

\_\_\_\_\_  
Rufnummer

\_\_\_\_\_  
E-Mail-Adresse

IBM Corporation  
Building 501  
P.O Box 12195  
Research Triangle Park, NC  
USA





Gedruckt in Deutschland

SC43-2898-00

