CA Embedded Entitlements Manager

Handbuch "Erste Schritte"

r8.4 SP3

CA-Produktreferenzen

Dieses Dokument bezieht sich auf die folgenden CA-Produkte:

- CA® Embedded Entitlements Manager (CA EEM)
- CA® Directory
- CA® SiteMinder® Web Access Manager (CA SiteMinder)
- CA® Identity Manager
- CA® Security Command Center
- CA® Integrated Threat Management
- CA® Enterprise Log Manager

Technischer Support – Kontaktinformationen

Wenn Sie technische Unterstützung für dieses Produkt benötigen, wenden Sie sich an den Technischen Support unter http://www.ca.com/worldwide. Dort finden Sie eine Liste mit Standorten und Telefonnummern sowie Informationen zu den Bürozeiten.

Inhalt

Kapitel 1: Einführung

Dieses Kapitel enthält folgende Themen:

Übersicht

CA Embedded Entitlements Manager (CA EEM) ermöglicht Anwendungen die gemeinsame Verwendung von allgemeinen Diensten zur Verwaltung von Zugriffsrichtlinien sowie Authentifizierungs- und Autorisierungsdiensten.

Funktionen

CA EEM stellt eine Reihe von Sicherheitsdiensten bereit. Die folgenden Sicherheitsdienste sind verfügbar:

- Konfigurationsdienste:
  - Anwendungsinstanzen registrieren und ihre Registrierung aufheben
  - Administratives Scoping von Anwendungsadministratoren
  - Delegieren von administrativen Rechten
  - Verwalten von Benutzern und Gruppen
- Administrations-Sicherheitsdienste:
  - Verwalten von Richtlinien für Zugriffe, Ereignisse und verbindliche Aufgaben
  - Verwalten von Kalendern
- Laufzeit-Sicherheitsdienste:
  - Authentifizieren von Benutzern
  - Zugriffsautorisierung
  - Protokollieren von Sicherheitsereignissen

Leistungsmerkmale

CA EEM enthält folgende Leistungsmerkmale:

Allgemein

- Richtlinienisolierung: Jede registrierte Anwendungsinstanz speichert ihre anwendungsspezifischen Daten in einem eigenen Verzeichnis
- Laufzeit-SDK für Java, C++ und C# erhältlich
- Administratives SDK für Java, C++ und C# erhältlich
- Unterstützung einer Befehlszeilenschnittstelle für administrative Funktionen (Einfügen/Ändern/Entfernen von Objekten):
  - Export und Import von XML
  - Laufzeitprüfungen
  - Migrations-Tools
- Webschnittstelle zur Unterstützung von Standalone- und Launch-in-Context-Zugriffen
- Sichere HTTP-Kommunikation
- Integriert mit CA Security Command Center und CA Audit zur Verwaltung von Sicherheitsereignissen
- Integriert mit CA SiteMinder zum Abrufen von Benutzer- und Gruppeninformationen aus dem CA SiteMinder-Datenspeicher

Identitätsverwaltung

- Gemeinsame globale Benutzer und Attribute für alle Anwendungen
- Unterstützung für verschiedene Modi für globale Benutzer
  - Interne globale Benutzer, komplett mit Verwaltung von Kennwortrichtlinien
  - Externe globale Benutzer aus LDAP-Verzeichnisservern.
  - Externe globale Benutzer von CA Identity Manager
- Integriert mit CA Identity Manager für rollenbasierte Benutzerbereitstellung und -verwaltung.
- Export/Import-Unterstützung für portable Sitzungen für Single-Sign-On

Zugriffsverwaltung

- Zugriffsverwaltung umfasst die Zugriffskontrolllisten (Access Control Lists, ACLs) und die Geschäftsrichtlinien.
- Richtlinienterminologie ermöglicht die Verwendung von Benutzer-, Sitzungs-, Umgebungs- und Ressourcenattributen beim Erstellen von Richtlinienentscheidungen
- Integriertes administratives Scoping aller Objekte

- Integrierte Unterstützung für delegierte Administration
- Integrierte Unterstützung für benutzerdefinierte Aufgabenprüfungen, die anwendungsspezifische Aktionen erfordern
  - Lokale Evaluierung von Berechtigungsprüfungen während des Vorgangs
  - SDK und Webschnittstelle zum Definieren von Zugriffsrichtlinien, ACLs, administrativen Scoping-Richtlinien und delegierter Autorität.

Clientzugriff

Sie können auf CA EEM Server über Standard-Web- und Web-Services-Schnittstellen zugreifen. Auf diese Weise wird die Integration von Produkten anderer Hersteller ohne Client-Modul ermöglicht. Die Schnittstellen sind:

- HTML und iTechnology für Konfiguration und Administration
- iTechnology zur Übermittlung von CA Audit-Ereignissen

iTechnology ist eine CA-Technologie auf der Grundlage von Webstandards wie HTTP, HTTPS, HTML, XML und SSL. Mit dieser Technologie steht ein Framework zur Verfügung, mit dem Webdienste erstellt und über das Internet bereitgestellt werden können.

Unterstützung für Datenspeicher

CA EEM unterstützt die Erkennung von einzelnen externen Benutzerquellen, wie Microsoft Active Directory. Unabhängig vom Speicherort der Benutzerobjekte speichert CA EEM seine Konfiguration und Richtlinien in CA Directory.

Kapitel 2: Installation unter Windows

Dieses Kapitel enthält folgende Themen:

Installationsübersicht

Bei der Installation von CA EEM in der Windows-Betriebsumgebung werden folgende Anwendungen installiert:

CA EEM Server

Mit Hilfe von CA EEM Server können Sie unter Verwendung einer Webschnittstelle Autorisierungsrichtlinien für Anwendungsressourcen festlegen. Über die webbasierte Administrationsschnittstelle können Sie Identitäten und Zugriffsrichtlinien verwalten. Die vorhandene Sicherheitsinfrastruktur wird verwendet, um Regeln auf Grundlage der Geschäftslogik zu implementieren, indem Ressourcen und Benutzerattribute verwendet werden, die in zentralen Benutzerspeichern und anderen Unternehmenssystemen definiert sind.

CA EEM Software Development Kit (SDK)

Verwenden Sie das CA EEM-SDK, um identitätsbasierte Sicherheitsfunktionen in Anwendungen einzubetten. Das SDK besteht aus Bibliotheken, Java-Klassen, Header-Dateien und einem Tutorial. Mit Hilfe des SDKs können Sie CA EEM in jeder beliebigen Anwendung implementieren. Weitere Informationen zur Implementierung von CA EEM mit Hilfe des SDKs finden Sie im *Programmierhandbuch*.

Jede Anwendung wird getrennt installiert und läuft unabhängig von der anderen.

Server installieren

Sie können CA EEM Server entweder mit Hilfe des Installationsassistenten oder über die Befehlszeile installieren. Die Installation von CA EEM erfolgt über die Befehlszeile im automatischen Modus und mit Hilfe des Installationsassistenten interaktiv.

JRE ist keine Mindestanforderung mehr für die Installation und Verwendung von CA EEM. Sie können CA EEM mit oder ohne JRE installieren und verwenden. Wenn Sie CA EEM ohne JRE als Mindestanforderung installieren möchten, müssen Sie im Installationsassistenten den Bildschirm zur Auswahl des JRE-Pfads überspringen. Wenn Sie CA EEM Server im automatischen Modus ohne JRE installieren möchten, müssen Sie den Parameter "javahome" auf "None" setzen.

In den folgenden Abschnitten wird die Installation von CA EEM Server beschrieben.

Weitere Informationen:

Überspringen des Bildschirms zur Auswahl des JRE-Pfads im Installationsassistenten (siehe Seite 15)
Checkliste für das Assistenten-Setup (siehe Seite 15)
Installieren des Servers mit Hilfe des Installationsassistenten (siehe Seite 16)
Installation von CA EEM Server im automatischen Modus (siehe Seite 23)

Checkliste für das Assistenten-Setup

Während der Installation von CA EEM Server unter Windows benötigen Sie die folgenden Informationen:

Feld	Wert
CA EEM-Installationspfad	Der Speicherort auf Ihrem Computer, an dem Sie CA EEM installieren möchten.
JRE-Installationspfad	Der Speicherort der JRE-Installation auf Ihrem Computer. **Hinweis**: Wenn Sie CA EEM ohne JRE installieren und verwenden möchten, müssen Sie die Variable "Javahome" über die Befehlszeile auf "None" setzen, bevor Sie den CA EEM-Installationsassistenten ausführen.
"EiamAdmin"-Kennwort	Das dem CA EEM-Administrator EiamAdmin zugewiesene Kennwort.
Sicherungsverzeichnispfad	Der Speicherort auf Ihrem Computer, an dem Sie die Dateien aus einer früheren Installation von CA EEM sichern möchten. **Hinweis:** Sie benötigen diese Information nur, wenn Sie eine frühere Version von CA EEM auf die aktuelle Version aktualisieren.

Überspringen des Bildschirms zur Auswahl des JRE-Pfads im Installationsassistenten

JRE ist keine Mindestanforderung mehr für die Installation und Verwendung von CA EEM. Wenn Sie CA EEM ohne JRE installieren möchten, führen Sie die folgenden Schritte aus:

1. Legen Sie für den Parameter "javahome" den Wert "None" fest.

   Hinweis: Wenn Sie den Parameter "javahome" auf "None" setzen, zeigt der Installationsassistent den Bildschirm zur Auswahl des Java-Pfads nicht an.

2. Installieren Sie CA EEM mit Hilfe des Installationsassistenten.

Festlegen des Parameters "javahome"

Sie müssen den Parameter "javahome" auf den Wert "None" setzen, bevor Sie CA EEM mit Hilfe des Installationsassistenten installieren. Legen Sie den Wert für den Parameter "javahome" wie folgt über die Befehlszeile fest:

EEMServer_*[Versionsnummer]*_win32.exe -s -a /z"javahome=None; "

Installieren des Servers mit Hilfe des Installationsassistenten

Der Installationsassistent von CA EEM Server führt Sie durch den Installationsvorgang und stellt Ihnen Optionen zum Definieren der Installationsparameter zur Verfügung.

So installieren Sie CA EEM Server:

1. Führen Sie einen der folgenden Schritte aus:
   - Öffnen Sie den Windows Explorer, und doppelklicken Sie auf dem Zielcomputer auf das Installationspaket EEMServer_[*Versionsnummer*].[*Build-Nummer*]_win32.exe.
   - Geben Sie an der Eingabeaufforderung unter Verwendung der Installationsparameter den folgenden Befehl ein:

     EEMServer_*[Versionsnummer].[Build-Nummer]*_win32.exe -s -a /z "eiampath=<*Benutzerdefinierter Installationspfad für CA EEM*>; etdirpath=<*Benutzerdefinierter Installationspfad für CA Directory*>; igpath=<*Benutzerdefinierter Installationspfad für iGateway*>;"

     Sie können einen benutzerdefinierten Installationspfad mit Hilfe von Installationsparametern angeben. Nähere Informationen über die Installationsparameter finden Sie unter Server-Installationsparameter.

     Der Installationsassistent wird angezeigt.
2. Führen Sie die Installation anhand der Anweisungen des Installationsassistenten durch.

Weitere Informationen:

Überspringen des Bildschirms zur Auswahl des JRE-Pfads im Installationsassistenten (siehe Seite 15)

Server aktualisieren

Sie können die vorhandene Installation von CA EEM Server auf die aktuelle Version aktualisieren.

So aktualisieren Sie eine vorhandene Installation von CA EEM Server:

1. Führen Sie EEMServer_<Versionsnummer>_win32.exe auf dem Zielcomputer aus.
2. Abhängig davon, welche Version von CA EEM Server installiert ist, geschieht Folgendes:
   - Wenn die vorhandene Version von CA EEM Server älter ist als die Version, die installiert wird, sichert der Installationsassistent die vorhandene Version und führt automatisch die Aktualisierung auf die neuere Version aus.
   - Wenn die vorhandene Version von CA EEM Server dieselbe ist wie die Version, die installiert werden soll, fragt der Installationsassistent, ob Sie CA EEM Server deinstallieren möchten. Sie können CA EEM Server deinstallieren und danach neu installieren.
   - Falls die Version, die installiert wird, älter als die vorhandene Version ist, gibt der Installationsassistent eine Fehlermeldung aus und bricht die Installation ab.

Bei einem Upgrade von CA EEM Server wird Folgendes aktualisiert:

- CA EEM Server im Ordner "\\CA\SharedComponents\iTechnology"
- iGateway
- CA Directory

Außerdem werden alle P12-Zertifikate in PEM-Zertifikate migriert.

Weitere Informationen:

Checkliste für das Assistenten-Setup (siehe Seite 15)
Installieren des Servers mit Hilfe des Installationsassistenten (siehe Seite 16)

Server starten

Sie müssen CA EEM Server starten, um die Identitäten und Zugriffsrichtlinien von registrierten Anwendungen zu verwalten.

So beginnen Sie mit der Verwendung von CA EEM Server:

1. Führen Sie einen der folgenden Schritte aus:
   - Geben Sie die URL https://*Rechnername* oder *ip-Adresse*:5250/spin/eiam im Browser ein. Wenn Sie auf dem Computer arbeiten, auf dem CA EEM Server installiert ist, geben Sie http://l*ocalhost*:5250/spin/eiam an.
   - Wählen Sie in Windows-Betriebsumgebungen "Start", "Programme", "CA", "Embedded Entitlements Manager", "EEM UI".

     Eine Anmeldeseite wird angezeigt.
2. Geben Sie folgende Informationen im Anmeldungsdialogfeld ein:
   a. Wählen Sie eine Anwendungsinstanz aus, die Sie mit Hilfe des Dropdown-Menüs "Anwendung" registriert haben. Die Standardvorgabe ist <Global>. Der Standard-Benutzername des Administrators lautet "EiamAdmin".

      Hinweis: Sie können andere globale Benutzer für die Anmeldung hinzufügen und deren Benutzernamen den Voreinstellungen entsprechend festlegen.

   b. Geben Sie Ihr Kennwort ein. Dies ist dasselbe Kennwort, das Sie während der Installation von CA EEM Server für EiamAdmin angegeben haben.

   c. Wählen Sie "Einstellungen speichern", wenn Sie sich das nächste Mal mit denselben Einstellungen bei CA EEM Server anmelden möchten.
3. Klicken Sie auf "Anmelden".

   Die Startseite der Benutzeroberfläche von CA EEM wird angezeigt. Weitere Informationen zur Verwendung von CA EEM Server finden Sie in der *Online-Hilfe*.

Aktivieren der Eingabehilfe im CA EEM-Server

Mit den Eingabehilfen im CA EEM-Server können Benutzer, ungeachtet Ihrer Fähigkeiten, die Produkte und unterstützende Dokumentation erfolgreich verwenden, um wichtige Geschäftsaufgaben zu erfüllen. Wenn Sie die Eingabehilfen aktivieren, können Benutzer wichtige Geschäftsaufgaben mit Hilfe der Tastatur oder der Sprachausgabe erfüllen.

So aktivieren Sie Eingabehilfen

1. Führen Sie einen der folgenden Schritte aus:
   - Geben Sie die URL https://*Rechnername* oder *ip-Adresse*:5250/spin/eiam im Browser ein. Wenn Sie auf dem Computer arbeiten, auf dem CA EEM-Server installiert ist, geben Sie http://l*ocalhost*:5250/spin/eiam an.
   - Wählen Sie in Windows-Betriebsumgebungen "Start", "Programme", "CA", "Embedded Entitlements Manager", "EEM UI".

     Eine Anmeldeseite wird angezeigt.
2. Geben Sie folgende Informationen im Anmeldungsdialogfeld ein:
   a. Wählen Sie eine Anwendungsinstanz aus, die Sie mit Hilfe des Dropdown-Menüs "Anwendung" registriert haben. Die Standardvorgabe ist <Global>. Der Standard-Benutzername des Administrators lautet "EiamAdmin".

      Hinweis: Sie können andere globale Benutzer für die Anmeldung hinzufügen und deren Benutzernamen den Voreinstellungen entsprechend festlegen.

   b. Geben Sie Ihr Kennwort ein. Dies ist dasselbe Kennwort, das Sie während der Installation von CA EEM Server für EiamAdmin angegeben haben.

   c. Wählen Sie "Einstellungen speichern", wenn Sie sich das nächste Mal mit denselben Einstellungen bei CA EEM Server anmelden möchten.
3. Klicken Sie auf "Eingabehilfen aktivieren".

   Eingabehilfen werden in der CA EEM-Benutzeroberfläche aktiviert.
4. Klicken Sie auf "Anmelden".

   Die Startseite der Benutzeroberfläche von CA EEM wird angezeigt.

Server entfernen

Sie können CA EEM Server in der Systemsteuerung unter "Software" deinstallieren.

Hinweis: Sie können CA EEM Server nicht entfernen, wenn Anwendungen in CA EEM registriert sind. Sie müssen die Registrierung der Anwendungen zunächst aufheben, bevor Sie CA EEM Server deinstallieren. Informationen zum Aufheben der Registrierung einer Anwendung finden Sie in der *Online-Hilfe.*

SDK installieren

Der Installationsassistent des CA EEM-SDKs führt Sie durch den Installationsvorgang.

So installieren Sie das CA EEM-SDK:

1. Öffnen Sie den Windows Explorer, und doppelklicken Sie auf das Installationspaket EEMSDK_<Versionsnummer>_win32.exe. Alternativ können Sie auch die Installationsdatei über die Befehlszeile ausführen.

   Der Installationsassistent wird angezeigt.

2. Klicken Sie auf "Ich stimme zu", um die Vertragsbedingungen zu akzeptieren.

   Hinweis: Die Schaltfläche "Ich stimme zu" wird erst aktiviert, wenn Sie den Text der Vertragsbedingungen gelesen bzw. an das Textende geblättert haben.

   Das Dialogfeld Zielspeicherort auswählen wird angezeigt: Standardmäßig installiert der Installationsassistent das CA EEM-SDK am folgenden Speicherort: C:\Programme\CA\Embedded IAM SDK

3. Klicken Sie auf "Weiter".

   Or

   Klicken Sie auf "Durchsuchen", wählen Sie auf dem Computer ein Verzeichnis aus, in das das CA EEM-SDK installiert werden soll, und klicken Sie dann auf "Weiter".

   Daraufhin wird die Installation des CA EEM-SDKs gestartet.

4. Klicken Sie auf "Fertig stellen".

   Das CA EEM-SDK wurde installiert.

   Hinweis: Während der Installation wird die Umgebungsvariable %EIAM_SDK% erstellt, die den Installationspfad enthält. Verwenden Sie diese Variable bei der Pfadangabe in Windows Explorer, um den Installationsordner zu öffnen.

SDK starten

Sie können das CA EEM-SDK starten, indem Sie auf "Start", "Programme", "CA", "Embedded Entitlements Manager", "EEM SDK" klicken.

Das Fenster mit der Dokumentation des CA EEM-SDKs wird angezeigt.

SDK entfernen

Sie können das CA EEM-SDK in der Systemsteuerung unter "Software" deinstallieren.

Server-Installationsparameter

Während der Installation von CA EEM unter Windows müssen Sie Informationen zu den folgenden Befehlszeilenparametern sammeln:

-eiampath

Gibt den Pfad an, unter dem CA EEM Server installiert wird. Der Standardpfad ist C:\Programme\CA \SharedComponents\Embedded IAM.

-etdirpath [Pfad]

Gibt den Pfad an, unter dem CA Directory installiert wird. Der Standardpfad ist "C:\Programme\CA\Directory".

-igpath [Pfad]

Legt den Pfad fest, in dem iGateway installiert wird. Der Standardpfad ist C:\Programme\CA\SharedComponents\iTechnology.

backupdir

Gibt den Speicherort an, an dem die Daten aus der vorhandenen Installation gesichert werden.

-capkiinstalldir

Gibt den Pfad des Installationsverzeichnisses für das CAPKI-Modul an. Der Standardpfad ist C:\Programme\CA\SC\CAPKI.

-javahome [Verzeichnis]

Legt die JAVA_HOME-Variable für das [Verzeichnis] fest, die für den Aufruf des iGateway-Installationsprogramms verwendet wird. Im CA EEM-Installationsprogramm werden Sie aufgefordert, die Variablen festzulegen, auch wenn sie schon angegeben wurden. Dieser Parameter hat keinen Standardwert.

Hinweis: Wenn Sie CA EEM ohne Java installieren möchten, müssen Sie den Parameter "javahome" auf "Keine" setzen.

CA EEM verwendet während der CA Directory-Installation die folgenden Parameter. Sie können die Parameter Ihren Anforderungen entsprechend konfigurieren.

Wichtig! Stellen Sie vor dem Anpassen der standardmäßigen Portnummern sicher, dass keine anderen Dienste für die Verwendung derselben Ports konfiguriert sind.

-dxadminport

Gibt den Port an, auf dem DXadmind Anfragen von DXmanager abhört. Dieser Port wird für die LDAP-Kommunikation zwischen DXadmind und DXmanager verwendet. *DXadmind* ist ein Hintergrundvorgang, der auf jedem Host läuft, der DSA enthält. DXmanager verwendet DXadmind, um mit den DSAs zu kommunizieren.

Standard: 2123

-dsaport

Gibt den Port an, den der DSA verwendet, um an ihn gerichtete Anfragen zu überwachen.

Standard: 509

-ssldport

Gibt den Port an, auf dem CA Directory den SSLD-Server überwacht. Der SSLD-Server ist ein Hintergrundprozess, der die SSL- und TLS-Authentifizierung, -Verschlüsselung und -Entschlüsselung für CA Directory durchführt.

Standard: 21847

-routerport

Gibt den Port an, den der DSA zum Verbinden mit dem Router-DSA verwendet. Ein Router-DSA besitzt keine lokalen Daten und keinen Datenspeicher. Er kann nur Datenverkehr an andere DSAs weiterleiten.

Standard: 1684

-dxdbsize

Gibt die maximale Größe des Datenspeichers für CA EEM an.

Standard: 500 MB

-dxuser

Gibt einen Nicht-DSA-Benutzer an, der CA Directory installieren, verwalten und deinstallieren kann. Der dxuser kann ein Benutzer des lokalen Systems oder ein Netzwerkbenutzer sein.

Hinweis: Wenn Sie CA Directory mit einem Benutzer des lokalen Systems als "dxuser" installiert haben, wird dieser Systembenutzer möglicherweise während der Deinstallation gelöscht. Stellen Sie daher sicher, dass der Benutzer nicht zum Ausführen anderer Programme eingerichtet ist, wenn Sie zur Installation von CA Directory einen Benutzer des lokalen Systems als "dxuser" verwenden.

Hinweis: Auf einem Computer mit Microsoft Windows Server 2003 beträgt die maximale Länge für die Zeichenfolge, die Sie in der Befehlszeile verwenden können, 8.191 Zeichen. Bei Microsoft Windows 2000 beträgt die maximale Länge des zu verwendenden Strings an der Eingabeaufforderung 2.047 Zeichen. Weitere Informationen zur Länge von InstallShield-Befehlen finden Sie in den *Versionshinweisen*.

Installation von CA EEM Server im automatischen Modus

Die Installation von CA EEM Server im automatischen Modus erfordert zwei Schritte:

1. Erstellen der Antwortdatei.
2. Ausführen des Befehls, der die Antwortdatei bestimmt.

   Während der automatischen Installation wird zur Protokollierung von Installationsfehlern die Protokolldatei "eiaminstall.log" erstellt.

Hinweis: Wenn Sie CA EEM Server automatisch installieren, können Sie diese Software auch automatisch deinstallieren.

Erstellen der Antwortdatei

Sie können Ihre Installationseingaben in einer Antwortdatei protokollieren, die für die automatische Installation von CA EEM Server verwendet wird. Für jeden Build, den Sie installieren möchten, müssen Sie eine neue Antwortdatei erstellen.

So erstellen Sie eine Antwortdatei:

1. Führen Sie das CA EEM Server-Installationspaket auf dem Zielcomputer aus.
2. Geben Sie an der Eingabeaufforderung den folgenden Befehl ein, um eine Antwortdatei im angegebenen Verzeichnis zu erstellen:

   EEMServer_[*Versionsnummer*].[*Build-Nummer*]_win32.exe -s -a /r /f1"*Pfadname der Antwortdatei*"

   Beispiel:

   EEMServer_8.4.0.55_win32.exe -s -a /r /f1"c:\resp.iss"
3. Geben Sie Werte für die Installationsparameter ein, die in der Antwortdatei gespeichert sind.

Ausführen des Befehls, der die Antwortdatei bestimmt.

In den folgenden Beispielen werden die Optionen für eine automatische Installation beschrieben:

- Geben Sie in der Befehlszeile den folgenden Befehl ein, um CA EEM Server im automatischen Modus zu installieren:

  EEMServer_[*Versionsnummer*].[*Build-Nummer*]_win32.exe -s -a /s /f1"*Pfadname der Antwortdatei*"

  Beispiel:

  EEMServer_8.4.0.55_win32.exe -s -a /s /f1"c:\resp.iss"
- Um während der automatischen Installation eine Protokolldatei der Installation zu erstellen, geben Sie folgenden Befehl in der Befehlszeile ein:

  EEMServer_[*Versionsnummer*].[*Build-Nummer*]_win32.exe -s -a /s /v"/qn /L*v <Pfad zum Erstellen der Protokolldatei>" /f1"*Pfadname der Antwortdatei*"

  Beispiel:

  EEMServer_8.4.0.55_win32.exe -s -a /s /v"/qn /L*v c:\install.txt" /f1"c:\resp.iss"

Mit diesem Befehl wird CA EEM Server im automatischen Modus unter Verwendung der angegebenen Antwortdatei installiert.

Hinweis: Sie können die Installationsparameter zusammen mit dem Installationsskript angeben. Nähere Informationen über die Parameter finden Sie unter Server-Installationsparameter.

Entfernen von CA EEM Server im automatischen Modus

Sie müssen eine Antwortdatei verwenden, die mit demselben Build von CA EEM Server erstellt wurde, um das Produkt erfolgreich zu entfernen. Geben Sie in der Befehlszeile den folgenden Befehl ein, um CA EEM Server im automatischen Modus zu entfernen:

EEMServer_[*Versionsnummer*].[*Build-Nummer*]_win32.exe -s -a /s /f1"*Pfadname der Antwortdatei*" /z"uninstall"

Beispiel:

EEMServer_8.4.0.55_win32.exe -s -a /s /f1"c:\resp.iss" /z"uninstall"

Dies entfernt CA EEM Server im automatischen Modus.

Hinweis: Sie können CA EEM Server nicht entfernen, wenn Anwendungen in CA EEM registriert sind. Sie müssen die Registrierung der Anwendungen zunächst aufheben, bevor Sie CA EEM Server deinstallieren. Informationen zum Aufheben der Registrierung einer Anwendung finden Sie in der *Online-Hilfe.*

Kapitel 3: Installation unter Linux und UNIX

Dieses Kapitel enthält folgende Themen:

Installationsübersicht

Für die Installation von CA EEM in Linux- und UNIX-Betriebsumgebungen müssen die folgenden Anwendungen installiert werden:

CA EEM Server

Mit Hilfe von CA EEM Server können Sie unter Verwendung einer Webschnittstelle Autorisierungsrichtlinien für Anwendungsressourcen festlegen. Über die webbasierte Administrationsschnittstelle können Sie Identitäten und Zugriffsrichtlinien verwalten. Die vorhandene Sicherheitsinfrastruktur wird verwendet, um Regeln auf Grundlage der Geschäftslogik zu implementieren, indem Ressourcen und Benutzerattribute verwendet werden, die in zentralen Benutzerspeichern und anderen Unternehmenssystemen definiert sind.

CA EEM Software Development Kit (SDK)

Verwenden Sie das CA EEM-SDK, um identitätsbasierte Sicherheitsfunktionen in Anwendungen einzubetten. Das SDK besteht aus Bibliotheken, Java-Klassen, Header-Dateien und einem Tutorial. Mit Hilfe des SDKs können Sie CA EEM in jeder beliebigen Anwendung implementieren. Weitere Informationen zur Implementierung von CA EEM mit Hilfe des SDKs finden Sie im *Programmierhandbuch*.

Jede Anwendung wird separat installiert und läuft unabhängig von der anderen.

Server installieren

CA EEM Server für Linux und UNIX verwendet ein selbstextrahierendes Shell-Skript, das Sie durch den Installationsprozess führt. Während der Installation zeigt das Skript die Lizenzinformationen an und fordert Sie auf, die Installationsparameter einzugeben. Sobald Sie die Installationsparameter eingegeben haben, wird die Installation gestartet.

So installieren Sie CA EEM Server für Linux und UNIX:

1. Führen Sie das Installationsskript EEMServer_[*Versionsnummer*].[*Build-Nummer*]_[*Name des Betriebssystems*].sh auf dem Zielcomputer aus.

   Beispiel:

   EEMServer_8.4.0.55_sunos.sh

   Die Datei wird dekomprimiert, und die Installation beginnt.

2. Geben Sie "J (Y)" ein, um die Geschäftsbedingungen der Lizenzvereinbarung zu akzeptieren (oder "N", um abzulehnen und die Installation abzubrechen).

   Das Skript fordert zur Eingabe der Installationsparameter auf.

3. Geben Sie die Installationsparameter ein.

   Hinweis: Weitere Informationen zu den verfügbaren Installationsparametern finden Sie unter Skriptparameter für die Serverinstallation.

   Beispiel:

   a. Geben Sie den Installationspfad für CA EEM Server ein (oder akzeptieren Sie die Standardvorgabe).

   Ein Bestätigungsbildschirm mit den von Ihnen eingegeben Installationsparametern wird angezeigt.

4. Wenn die Angaben auf dem Bestätigungsbildschirm korrekt sind, geben Sie "J (Y)" ein, um die Installation fortzusetzen (wenn Sie "N" eingeben, wird das Installationsprogramm geschlossen).

5. Geben Sie das EiamAdmin-Kennwort ein.

   Hinweis: Der Standard-Benutzername des Administrators lautet EiamAdmin.

   Der Installationsvorgang hängt von den Befehlszeilenparametern und dem Typ des installierten CA EEM Server-Pakets ab.

   Das Skript des Installationsprogramms schließt die Installation von CA EEM Server auf Ihrem Computer ab.

Server aktualisieren

Sie können die vorhandene Installation von CA EEM Server auf die aktuelle Version aktualisieren.

So aktualisieren Sie eine vorhandene Installation von CA EEM Server:

1. Führen Sie EEMServer_[*Versionsnummer*].[*Build-Nummer*]_[*Name des Betriebssystems*] auf dem Zielcomputer aus.
2. Abhängig davon, welche Version von CA EEM Server installiert ist, geschieht Folgendes:
   - Wenn die vorhandene Version von CA EEM Server älter ist als die Version, die installiert wird, führt der Installationsassistent automatisch eine Aktualisierung auf die neuere Version aus.
   - Wenn die vorhandene Version von CA EEM Server dieselbe ist wie die Version, die installiert werden soll, fragt der Installationsassistent, ob Sie CA EEM Server deinstallieren möchten. Sie können CA EEM Server deinstallieren und danach neu installieren.
   - Falls die Version, die installiert wird, älter als die vorhandene Version ist, gibt der Installationsassistent eine Fehlermeldung aus und bricht die Installation ab.

   Weitere Informationen zur Installation von CA EEM Server finden Sie unter Server installieren (siehe Seite 28).

Bei einem Upgrade von CA EEM Server wird Folgendes aktualisiert:

- CA EEM Server im Ordner "\\CA\SharedComponents\iTechnology"
- iGateway
- CA Directory

Server entfernen

Um CA EEM Server zu entfernen, führen Sie das Skript "eiamuninstall.sh" im Installationsverzeichnis aus.

Hinweis: Sie können CA EEM Server nicht entfernen, wenn Anwendungen in CA EEM registriert sind. Sie müssen die Registrierung der Anwendungen zunächst aufheben, bevor Sie CA EEM Server deinstallieren. Informationen zum Aufheben der Registrierung einer Anwendung finden Sie in der *Online-Hilfe.*

SDK installieren

Das CA EEM-SDK für Linux und UNIX verwendet ein selbstextrahierendes Shell-Skript, das Sie durch den Installationsprozess führt. Während der Installation zeigt das Skript die Lizenzinformationen an und fordert Sie auf, die Installationsparameter einzugeben. Sobald Sie die Installationsparameter eingegeben haben, wird die Installation gestartet.

So installieren Sie das CA EEM-SDK für Linux und UNIX:

1. Führen Sie das Installationsskript EEMSDK_[*Versionsnummer*].[*Build-Nummer*]_[*Name des Betriebssystems*].sh auf dem Zielcomputer aus.

   Beispiel:

   EEM_8.4.0.55_sunos.sh

   Die Datei wird dekomprimiert, und die Installation beginnt.

2. Geben Sie "J (Y)" ein, um die Geschäftsbedingungen der Lizenzvereinbarung zu akzeptieren (oder "N", um abzulehnen und die Installation abzubrechen).
3. Geben Sie den Installationspfad für das CA EEM-SDK ein (oder akzeptieren Sie die Standardvorgabe).
4. Wählen Sie "Produkt installieren" aus.

   Das CA EEM-SDK wird auf Ihrem Computer installiert.

Starten des CA EEM-SDKs

Um das CA EEM-SDK zu starten, geben Sie in Ihrem Webbrowser die Adresse "/opt/CA/eIAMSDK/Doc/index.html" (bzw. den Installationsort des CA EEM-SDKs) ein.

SDK entfernen

Sie können das CA EEM-SDK aus Linux- und UNIX-Betriebssystemen entfernen.

So entfernen Sie das CA EEM-SDK:

1. Führen Sie das Installationsskript EEMSDK_[*Versionsnummer*].[*Build-Nummer*]_[*Name des Betriebssystems*].sh auf dem Zielcomputer aus.

   Beispiel:

   EEM_8.4.0.55_sunos_linux.sh

   Die Datei wird dekomprimiert.

2. Wählen Sie "Produkt deinstallieren/entfernen" aus.

   Das Installationsskript entfernt das CA EEM-SDK von Ihrem Computer.

Skriptparameter für die Serverinstallation

Während der Installation von CA EEM müssen Sie Informationen zu den folgenden Befehlszeilenparametern sammeln, zu deren Eingabe das Skript Sie während der Installation auffordert.

Das Skript akzeptiert die folgenden Befehlszeilenparameter:

backupdir

Gibt den Speicherort an, an dem die Daten aus der vorhandenen Installation gesichert werden.

-capkiinstalldir

Gibt den Pfad des Installationsverzeichnisses für das CAPKI-Modul an.

Standard: /opt/CA/SharedComponents/capki

CA EEM verwendet während der CA Directory-Installation die folgenden Parameter. Sie können die Parameter Ihren Anforderungen entsprechend konfigurieren.

Wichtig! Stellen Sie vor dem Anpassen der standardmäßigen Portnummern sicher, dass keine anderen Dienste für die Verwendung derselben Ports konfiguriert sind.

-dxadminport

Gibt den Port an, auf dem DXadmind Anfragen von DXmanager abhört. Dieser Port wird für die LDAP-Kommunikation zwischen DXadmind und DXmanager verwendet. *DXadmind* ist ein Hintergrundvorgang, der auf jedem Host läuft, der DSA enthält. DXmanager verwendet DXadmind, um mit den DSAs zu kommunizieren.

Standard: 2123

-dsaport

Gibt den Port an, den der DSA verwendet, um an ihn gerichtete Anfragen zu überwachen.

Standard: 509

-ssldport

Gibt den Port an, auf dem CA Directory den SSLD-Server überwacht. Der SSLD-Server ist ein Hintergrundprozess, der die SSL- und TLS-Authentifizierung, -Verschlüsselung und -Entschlüsselung für CA Directory durchführt.

Standard: 21847

-routerport

Gibt den Port an, den der DSA zum Verbinden mit dem Router-DSA verwendet. Ein Router-DSA besitzt keine lokalen Daten und keinen Datenspeicher. Er kann nur Datenverkehr an andere DSAs weiterleiten.

Standard: 1684

-dxdbsize

Gibt die maximale Größe des Datenspeichers für CA EEM an.

Standard: 500 MB

-dxuser

Gibt einen Nicht-DSA-Benutzer an, der CA Directory installieren, verwalten und deinstallieren kann. Der dxuser kann ein Benutzer des lokalen Systems oder ein Netzwerkbenutzer sein.

Hinweis: Wenn Sie CA Directory mit einem Benutzer des lokalen Systems als "dxuser" installiert haben, wird dieser Systembenutzer möglicherweise während der Deinstallation gelöscht. Stellen Sie daher sicher, dass der Benutzer nicht zum Ausführen anderer Programme eingerichtet ist, wenn Sie zur Installation von CA Directory einen Benutzer des lokalen Systems als "dxuser" verwenden.

-eiamadminpw [Kennwort]

Setzt das EiamAdmin-Kennwort auf [Kennwort]

-eiampath

Legt den Pfad fest, unter dem CA EEM Server installiert wird. Standardpfad: /opt/CA/SharedComponents/EmbeddedIAM.

-etdirpath [Pfad]

Legt den Pfad fest, unter dem CA Directory installiert wird.

-igpath [Verzeichnis]

Legt den Pfad für iGateway fest. Der Pfad muss voll qualifiziert sein, wie beispielsweise "-iisystem". Standardpfad: /opt/CA/SharedComponents/iTechnology.

-javahome [Verzeichnis]

Legt JAVA_HOME fest. Die Standardvorgabe dieses Parameters ist der Inhalt der JAVA_HOME-Umgebungsvariable. Dieser Parameter wird nur abgefragt, wenn $JAVA_HOME nicht festgelegt ist.

Hinweis: Wenn Sie CA EEM ohne Java installieren möchten, müssen Sie "javahome" auf "Keine" festlegen. Dies gilt nicht auf HP-UX.

-logfile [Dateiname]

Das Installationsprogramm protokolliert Daten in [Dateiname]. Standardwert: /tmp/eiam-install.log

-silent

Führt die Installation im automatischen Modus aus. Falls ein erforderlicher Parameter nicht in der Befehlszeile eingegeben wurde, wird die Installation abgebrochen, und eine entsprechende Meldung wird ausgegeben. Sofern alle erforderlichen Parameter angegeben wurden, werden keine Änderungen am System vorgenommen.

-tempdir [Verzeichnis]

Gibt das Verzeichnis an, das zum Speichern der Temporärdatei verwendet werden soll. Die Standardvorgabe ist "/tmp/eiam_temp". Dies muss ein voll qualifizierter Pfad mit einem eigenen Unterverzeichnis sein. Zur Beendigung des Skripts entfernt das Skript mit "rm –rf" das Verzeichnis, das Sie hier angeben.

Server im automatischen Modus installieren

Geben Sie in der Befehlszeile den folgenden Befehl ein, um CA EEM Server unter Linux oder UNIX im automatischen Modus zu installieren:

EEMServer_[*Versionsnummer*].[*Build-Nummer*]_[*Name des Betriebssystems*].sh -silent -eiamadminpw *Kennwort -javahome-Verzeichnis*

Beispiel: Der folgende Befehl für Sun-Betriebsumgebungen enthält die mindestens erforderlichen Parameter:

EEMServer_8.4.0.55_sunos.sh -silent -eiamadminpw *Kennwort -javahome-Verzeichnis*

Sie können zusätzliche Installationsparameter angeben. Für die meisten Installationsparameter sind Standardwerte vorhanden. Weitere Informationen zu den Skriptparametern finden Sie unter Skriptparameter für die Serverinstallation.

Die Datei wird dekomprimiert, und die Installation beginnt.

Entfernen von CA EEM Server im automatischen Modus

Um den CA EEM-Server zu entfernen, führen Sie das Skript "eiamuninstall.sh -silent" im Installationsverzeichnis aus.

Hinweis: Sie können CA EEM Server nur dann entfernen, wenn keine Anwendungen registriert sind. Sie müssen die Registrierung aller Anwendungen aufheben, um die Deinstallation erfolgreich abzuschließen. Informationen zum Aufheben der Registrierung einer Anwendung finden Sie in der *Online-Hilfe*.

Kapitel 4: Konfigurieren von CA EEM-SDKs

Neue Binärdateien für die Erstellung von Anwendungen mit Hilfe von CA EEM-SDKs

Neben den CA EEM-SDK-DLLs aus älteren Versionen müssen die folgenden neuen Binärdateien verwendet werden, um das CA EEM-SDK r8.4 SR02 in Ihre Anwendungen einzubetten.

Java

- xml-apis.jar

C++

Kopieren Sie je nach Betriebssystem die folgenden Dateien aus dem Ordner "EIAMSDK/lib/$OS2":

Windows

- log4cxx.dll
- log4cxx.lib
- libexpat-2,0.1.dll
- libexpat-2.0.1.lib

HP-UX

- *log4cxx*, zum Beispiel liblog4cxx.sl, liblog4cxx.sl.10, liblog4cxx.sl.10.0
- libapr*, zum Beispiel libapr-1.sl.3, libaprutil-1.sl.3, libapr-1.sl.3.3, libaprutil-1.sl.3.4
- libexpat-2.0.1.sl

UNIX ausschließlich HP-UX

- *log4cxx*, zum Beispiel liblog4cxx.so, liblog4cxx.so.10, liblog4cxx.so.10.0
- libexpat*

Erstellen von Anwendungen mit Hilfe der neuen Java-Binärdateien

1. Aktualisieren Sie "ClassPath" mit Verweisen auf xml-apis.jar.
2. Aktualisieren Sie Ihr Installationsprogramm, um die neuen Binärdateien und die Logger-Konfigurationsdatei zu packen.
3. Stellen Sie die neuen Binärdateien und die Logger-Konfigurationsdatei zusammen mit den CA EEM-SDK-Binärdateien bereit.

Dateien, die zur Ausführung von Anwendungen mit Hilfe von CA EEM C++ SDK nötig sind

Die folgenden Binärdateien werden benötigt, um mit Hilfe von CA EEM C++ SDK Anwendungen einbetten und ausführen zu können:

Windows

- ipthread.dll
- libcurl_7_18_2.dll
- libexpat-2.0.1.dll
- log4cxx.dll
- msvcm80.dll
- msvcm90.dll
- msvcp71.dll
- msvcp80.dll
- msvcp90.dll
- msvcr70.dll
- msvcr71.dll
- msvcr80.dll
- msvcr90.dll
- pcre.dll
- pthread.dll
- pthreadVCE.dll
- xerces-c_2_8.dll
- zlib.dll
- Microsoft.VC80.CRT.manifest
- Microsoft.VC90.CRT.manifest

HP-UX

- liblog4cxx.sl, liblog4cxx.sl.10, liblog4cxx.sl.10.0
- libapr-1.sl.3, libapr-1.sl.3.3, libaprutil-1.sl.3.4
- libexpat-2.0.1.sl, libxerces-c.sl.28 , libcurl.sl.4, libpcre.sl.0, libexpat.sl.2 libz.sl, liblog4cxx.sl.10.0

Linux

- libxerces-c.so.28
- libcurl.so.4
- libexpat.so.2 für linux_k24 und libexpat-2.0.1.so für linux_26
- libpcre.so.0
- libz.so.1
- liblog4cxx.so.10.0.0

AIX

- libxerces-c28.a libcurl.4.so libexpat-2.0.1.a libpcre.a libz.so
- liblog4cxx.a

Sun Solaris

- libxerces-c.so.28
- libcurl.so.4 libpcre.so
- libexpat.so.2 libz.so
- liblog4cxx.so.10.0.0

Erstellen von Anwendungen mit Hilfe der neuen C++-Binärdateien

1. Nehmen Sie die dem CA EEM-SDK beiliegenden, neuen Bibliotheken auf, indem Sie die folgenden Zeilen zu Ihrer Makefile hinzufügen:

   ```
   -llog4cxx -llibexpat
   ```

2. Aktualisieren Sie Ihr Installationsprogramm, um die neuen Binärdateien, die Datei "eiam.config" und die Datei "eiam.log4cxx.config" zu packen.
3. Stellen Sie die neuen Binärdateien, die Datei "eiam.config" und die Datei "eiam.log4cxx.config" zusammen mit den CA EEM-SDK-Binärdateien bereit.

Hinweis: Die Logger-Header-Dateien müssen Sie nicht in Ihren Quellcode aufnehmen.

Dateien, die zum Erstellen und Ausführen der Anwendungen mit Hilfe von CA EEM C# SDK benötigt werden

Die folgenden Binärdateien werden benötigt, um mit Hilfe von CA EEM C# SDK Anwendungen einbetten und ausführen zu können:

- log4net.dll
- CPoz.dll
- iclient.dll
- CsharpSDK.dll

Hinweis: "CAPICOM.dll" und "InterOP.CAPICOM.dll" sind nicht notwendig, um Anwendungen mit Hilfe von C# SDK zu erstellen. Entfernen Sie diese dll-Dateien aus dem Paket.

Verpacken von CA EEM C# SDK mit Ihren Anwendungen

1. Fügen Sie die dlls vom folgenden Ordner als Referenz hinzu, wenn Sie Ihre Anwendung erstellen:

   %EIAM_SDK%\lib\csharp

2. Aktualisieren Sie Ihr Installationsprogramm, um die Referenz-Dlls sowie die Dateien "eiam.config" und "eiam.log4net.config" zu verpacken.

   Hinweis: Die Datei "eiam.config" und die Dateien "eiam.lognet.config" befinden sich im Ordner "%EIAM_SDK%\bin".

3. Stellen Sie die Referenz-Dlls sowie die Dateien "eiam.config" und "eiam.log4net.config" auf den Client-Computern bereit.

CA EEM-SDK-Konfiguration

Die folgenden Themen erläutern, wie CA EEM-SDKs mit Hilfe der "Safe::Configurator"-Klasse konfiguriert werden.

Weitere Informationen:

Wissenswertes über die "eiam.config"-Datei (siehe Seite 40)
CA EEM-SDK-Protokollierung (siehe Seite 89)
Konfigurieren von CA EEM-C++-SDKs (siehe Seite 47)
Konfigurieren von CA EEM-Java-SDKs mit SafeConfigurator (siehe Seite 46)

Wissenswertes über die "eiam.config"-Datei

Sie müssen die "eiam.config"-Datei verwenden, um CA EEM-SDK-Konfigurationsdaten zu steuern, z. B.:

- zyklische Puffer
- Logger-Konfigurationsdatei
- SAF-Ordner zum Speichern von Überwachungsdateien
- FIPS-kompatibler Modus

Die "eiam.config"-Datei besteht aus den folgenden konfigurierbaren Parametern:

CyclicBuffer size

Gibt die Anzahl der in einem zyklischen Puffer enthaltenen Protokollmeldungen an. Der zyklische Puffer legt eine festgelegte Anzahl von aktuellsten Protokollmeldungen im Speicher ab. Sobald der Puffer die festgelegte Größe erreicht, wird die älteste Protokollmeldung im Puffer durch eine neue Protokollmeldung ersetzt. Wenn die Anwendung abstürzt, können Sie die aktuellsten Protokollmeldungen wieder aus dem Core herstellen.

Standard: 500

Minimum: 0

Maximum: 1000

enable

Gibt an, ob der zyklische Puffer aktiviert ist. Wenn "enable" auf "false" gesetzt ist, ist der zyklische Puffer deaktiviert. Sie müssen daher keine Werte für die Parameter CyclicBuffer-Größe, Dump und Datei angeben.

Wert: [*true*|*false*] (wahr/falsch)

Standard: "true" (wahr)

Wichtig! Der zyklische Puffer ist standardmäßig aktiviert, unabhängig davon, ob Sie die Protokollierung aktiviert haben oder nicht. Wenn Sie den zyklischen Puffer aktivieren, wird dadurch die Leistung von CA EEM beeinträchtigt.

dump

Gibt an, ob die Inhalte des zyklischen Puffers in eine Datei geschrieben werden, wenn die "eiam.config"-Datei geändert oder aktualisiert wird.

Wert: [*true*|*false*] (wahr/falsch)

Standard: "false" (falsch)

file

Gibt den Dateinamen der Dump-Datei an. Wenn "Dump" auf "false" gesetzt ist, werden die Protokollmeldungen nicht in eine Dump-Datei geschrieben. "File" hat die Dateierweiterung ".log".

LoggerConfiguration file

Gibt den absoluten Pfad der Logger-Konfigurationsdateien für CA EEM-Java- und -C++-SDKs an. Die CA EEM-Protokollierungsdaten sind in den Logger-Konfigurationsdateien gespeichert. "eiam.log4cxx.config" und "eiam.log4j.config" sind die Logger-Konfigurationsdateien für CA EEM-C++-SDKs und CA EEM-Java-SDKs.

Saf directory

SAF-Ordner, in dem Überwachungsdateien zur Verarbeitung gespeichert werden.

Network sockettimeout

Gibt die Socket-Zeitüberschreitung in Millisekunden an.

Standard: 120000 (2 Sekunden)

Weitere Informationen:

CA EEM-SDK-Protokollierung (siehe Seite 89)

Beispiel für eine "eiam.config"-Datei

Das folgende Beispiel veranschaulicht den Inhalt der Datei "eiam.config":

```
<EiamConfiguration>
 <!-- EIAM-intern: zyklischen Puffer konfigurieren -->
 <CyclicBuffer size="500" dump="false" file="dump.log" enable="true" />
 <!-- Absoluter Dateipfad für Logger-Konfiguration, Bei Java:-  file="eiam.log4j.config" -->
 <LoggerConfiguration file="eiam.log4cxx.config"/>
 <!-- Absoluter Verzeichnispfad für SAF-Ordner, in dem Überwachungsdateien zur Verarbeitung gespeichert
werden-->
 <Saf directory="audit"/>
  <!-- Socket-Zeitüberschreitung in Millisekunden. Standardwert ist 2 Min. -->
 <Network sockettimeout="120000"/>
 <SDK type="Java">
        <iTechSDK>
            <FIPSMode>true</FIPSMode>
            <JCEProvider>JsafeJCE</JCEProvider>
            <Security>
                <digestAlgorithm>SHA1</digestAlgorithm>
            </Security>
            <Debug>
                <logLevel>trace</logLevel>
            </Debug>
        </iTechSDK>
 </SDK>
 <SDK type="C++">
        <iTechSDK>
            <FIPSMode></FIPSMode>
            <Commons>
                <etpkiCryptoLib></etpkiCryptoLib>
            </Commons>
            <TransportConfig>
                <!--mögliche Werte sind SSLV23 / SSLV3 / TLSV1-->
                <secureProtocol></secureProtocol>
            </TransportConfig>
            <Security>
                <!--mögliche Werte sind MD5/SHA1/SHA256/SHA384/SHA512-->
                <digestAlgorithm></digestAlgorithm>
            </Security>
            <Debug>
                <!--mögliche Werte sind ERROR/WARNING/TRACE/NOLEVEL-->
                <logLevel></logLevel>

                <!--mögliche Werte sind wahr/falsch -->
                <logToFile></logToFile>

                <!--Protokolldateiname-->
                <logFile></logFile>
```

```
                    <!--Protokolldateigröße in MB(positive Ganzzahl)-->
                    <maxLogSize></maxLogSize>
                </Debug>
            </iTechSDK>
        </SDK>
    </EiamConfiguration>
```

Aktivieren vom iTechnology SDK-Protokollierung

Sie können die iTechnology SDK-Protokollierung nur für CA EEM C++ SDK und CA EEM Java SDK aktivieren. Verwenden Sie für CA EEM C# SDK die Protokollkonfigurationsdatei.

Um die iTechnology SDK-Protokollierung zu aktivieren, öffnen Sie die Datei "eiam.config" und bearbeiten Sie die folgenden Kennungen:

- logLevel
- logToFile
- logFile
- maxLogSize

Bearbeiten Sie für CA EEM Java SDK die zuvor erwähnten Kennungen im Abschnitt <SDK type="Java">. Bearbeiten Sie für CA EEM C++ SDK die im Abschnitt <SDK type="C++"> erwähnten Kennungen.

Bevor Sie CA EEM Java SDK im Nur-FIPS-Modus konfigurieren

Um CA EEM Java SDK im Nur-FIPS-Modus zu konfigurieren, führen Sie die folgenden Aufgaben aus:

1. Konfigurieren Sie JRE, um JCE-Bibliotheken (Java Cryptography Extension) von Drittanbietern zu verwenden.
2. Fügen Sie die Crypto-J-Bibliotheken als ein JCE-Anbieter in der Datei "Java.security" hinzu.

   Hinweis: Weitere Informationen zur Konfiguration von JRE mit JCE finden Sie in der jeweiligen JCE-Dokumentation.
3. Aktivieren Sie den Nur-FIPS-Modus in der Datei "eiam.config".

Konfigurieren von CA EEM Java SDK im Nur-FIPS-Modus

Wenn Sie CA EEM-SDK im Nur-FIPS-Modus konfigurieren, verwendet CA EEM FIPS 140-2-kompatible kryptografische Bibliotheken, um vertrauliche Daten zu verschlüsseln und zu entschlüsseln.

So konfigurieren Sie CA EEM Java SDK im Nur-FIPS-Modus

1. Öffnen Sie die Datei "eiam.config" und bearbeiten Sie die folgenden Kennungen im Abschnitt <SDK type="Java">:
   - FIPSMode
   - JCEProvider
   - digestAlgorithm
2. Speichern und schließen Sie die Datei "eiam.config".
3. Starten Sie Ihre Anwendung neu.

   Der Nur-FIPS-Modus ist nun für CA EEM Java SDK konfiguriert.

Konfigurieren von CA EEM C++ SDK im Nur-FIPS-Modus

Wenn Sie CA EEM-SDK im Nur-FIPS-Modus konfigurieren, verwendet CA EEM FIPS 140-2-kompatible kryptografische Bibliotheken, um vertrauliche Daten zu verschlüsseln und zu entschlüsseln.

So konfigurieren Sie CA EEM C++ SDK im Nur-FIPS-Modus

1. Öffnen Sie die Datei "eiam.config" und bearbeiten Sie die folgenden Kennungen im Abschnitt <SDK type="C++">:
   - FIPSMode
   - etpkiCryptoLib
   - secureProtocol
   - digestAlgorithm
2. Speichern und schließen Sie die Datei "eiam.config".
3. Starten Sie Ihre Anwendung neu.

   Der Nur-FIPS-Modus ist nun für CA EEM C++ SDK konfiguriert.

Konfigurieren von CA EEM C# SDK im Nur-FIPS-Modus

Wenn Sie CA EEM-Server im Nur-FIPS-Modus konfigurieren, verwendet CA EEM ausschließlich FIPS 140-2-kompatible kryptografische Bibliotheken, um vertrauliche Daten zu verschlüsseln und zu entschlüsseln.

Hinweis: CA EEM C# SDK unterstützt keine P11-Zertifikate.

So konfigurieren Sie CA EEM C# SDK im Nur-FIPS-Modus

1. Öffnen Sie die Datei "eiam.config" und bearbeiten Sie die folgenden Kennungen im Abschnitt <SDK type="C#">:
   - FIPSMode
   - digestAlgorithm
2. Speichern und schließen Sie die Datei "eiam.config".
3. Starten Sie Ihre Anwendung neu.

   Der Nur-FIPS-Modus ist nun für CA EEM C# SDK konfiguriert.

Festlegen von SafeContext-Informationen

Die <SafeContext>-Kennung in der Datei "eiam.config" enthält Informationen, die für die Erstellung von "SafeContext" mit Hilfe der "SafeContextFactory"-Klasse erforderlich ist. Jede "SafeContext"-Kennung in der Datei "eiam.config" wird über eine eindeutige "refID"-Kennung identifiziert. Um "SafeContext" zu generieren, müssen Sie diese "refID" an "SafeContextFactory" weitergeben. Nachfolgend werden die Nutzen aufgelistet, wenn Informationen zu "SafeContext" in der Datei "eiam.config" angegeben werden:

So legen Sie Informationen zu "SafeContext" fest:

1. Öffnen Sie die Datei "eiam.config" und bearbeiten Sie den Abschnitt <SafeContext>, um die folgenden Kennungen festzulegen:
   - refID
   - Backend
   - Anwendung
   - Locale
   - Authentication Type
2. Speichern und schließen Sie die Datei "eiam.config".

Konfigurieren von CA EEM-Java-SDKs mit SafeConfigurator

Sie müssen ein CA EEM-SDK mit Hilfe der Klasse "Safe::Configurator" konfigurieren. Gehen Sie wie folgt vor, um das CA EEM-SDK zu konfigurieren:

Hinweis: Vor dem CA EEM-SDK müssen Sie zunächst die Datei "eiam.config" konfigurieren.

1. Fügen Sie die folgende API in Ihren Programmcode ein, um das CA EEM-SDK während dem Anwendungsstart zu initialisieren:

   ```
   SafeConfigurator.getInstance().init(filename);
   ```

   Wobei

   "filename"

   den absoluten Pfad der Datei "eiam.config" angibt, die Sie für Ihre Anwendung definiert haben.

   Hinweis: Alle CA EEM-SDK-Vorgänge nach dieser Zeile werden protokolliert. Dies erfolgt basierend auf den Protokollierungs-Detailebenen in der Logger-Konfiguration.

2. Fügen Sie die folgende API in die Sequenz Ihres Programmcodes ein, an der die Anwendung beendet wird.

   ```
   m_config.term();
   ```

   Hinweis: Bei jedem mit "m_config.init(*filename*)" durchgeführten Initialisierungsaufruf müssen Sie den Aufruf mit der entsprechenden API "m_config.term()" beendigen. Die Methoden "Init" und "Term" sind Thread-sicher und basieren auf der Zählung einzelner Verweise. Die Bibliothek "Safe" wird während des ersten Init()-Aufrufs initialisiert und, sobald der Verweiszähler Null angibt, beendet.

Weitere Informationen:

Wissenswertes über die "eiam.config"-Datei (siehe Seite 40)
Wissenswertes über Logger-Konfigurationsdateien (siehe Seite 90)

Konfigurieren von CA EEM-C++-SDKs

Sie müssen ein CA EEM-SDK mit Hilfe der Klasse "Safe::Configurator" konfigurieren. Gehen Sie wie folgt vor, um das CA EEM-SDK zu konfigurieren:

Hinweis: Vor dem CA EEM-SDK müssen Sie zunächst die Datei "eiam.config" konfigurieren.

1. Fügen Sie die folgende API in Ihren Programmcode ein, um das CA EEM-SDK während dem Anwendungsstart zu initialisieren:

   Safe::Configurator::getInstance()->init(*filename*);

   Wobei

   "filename"

   den absoluten Pfad der Datei "eiam.config" angibt, die Sie für Ihre Anwendung definiert haben.

2. Fügen Sie die folgende API in die Sequenz Ihres Programmcodes ein, an der die Anwendung beendet wird.

   Safe::Configurator::getInstance()->term();

   Hinweis: Bei jedem mit "Safe::Configurator::getInstance()->init(filename)" durchgeführten Initialisierungsaufruf müssen Sie den Aufruf mit der entsprechenden API "Safe::Configurator::getInstance()->term()" beendigen. Die Methoden "Init" und "Term" sind Thread-sicher und basieren auf der Zählung einzelner Verweise. Die Bibliothek "Safe" wird während des ersten Init()-Aufrufs initialisiert und, sobald der Verweiszähler Null angibt, beendet.

Weitere Informationen:

Wissenswertes über die "eiam.config"-Datei (siehe Seite 40)
Wissenswertes über Logger-Konfigurationsdateien (siehe Seite 90)

Initialisieren von CA EEM C# SDK

Konfigurieren Sie CA EEM-SDK über die "SafeConfigurator"-Klasse. Um CA EEM-SDK zu konfigurieren, führen Sie den folgenden Prozess aus:

Hinweis: Konfigurieren Sie die Datei "eiam.config", bevor Sie CA EEM-SDK konfigurieren. Wenn Sie die Datei "eiam.config" nicht zuerst konfigurieren, wird CA EEM-SDK mit der folgenden Standardkonfiguration initialisiert:

- Nicht-FIPS-Modus
- Protokollierung ist deaktiviert, nur die Konsolenprotokollierung ist aktiviert
- SAF-Speicherort wird deaktiviert

So initialisieren Sie CA EEM-SDK

1. Nehmen Sie die folgende API in Ihren Code auf, um CA EEM-SDK beim Anwendungsstart zu initialisieren:

   SafeConfigurator.getInstance().Init(*filename*);

   Wobei

   filename

   Gibt den absoluten Pfad der Datei "eiam.config" an, die Sie für Ihre Anwendung angegeben haben.

   Hinweis: Wenn Sie den Dateinamen nicht angeben, wird CA EEM-SDK mit den Standardwerten initialisiert.

2. Nehmen Sie die folgende API für das Herunterfahren Ihrer Anwendung in Ihren Code auf:

   SafeConfigurator.getInstance().term();

   Hinweis: Weitere Informationen zur "SafeConfigurator"-Klasse finden Sie im *Programmierhandbuch*.

Kapitel 5: Unterstützung von FIPS 140-2

Dieses Kapitel enthält folgende Themen:

FIPS 140-2 – Übersicht

In der FIPS 140-2-Veröffentlichung (Federal Information Processing Standards) sind die Anforderungen festgelegt, die erfüllt werden müssen, um innerhalb eines Sicherheitssystems zum Schutz von sensiblen, nicht klassifizierten Daten kryptografische Algorithmen zu verwenden. CA EEM-Server enthält die kryptografische Bibliothek Crypto-C ME v2.0 aus RSA ein, die offiziell den FIPS 140-2-*Sicherheitsanforderungen für kryptografische Module entspricht.* Die Nummer des Gültigkeitserklärungszertifikats für dieses Modul ist 608.

CA EEM Java SDK verwendet eine FIPS-kompatible kryptografische Bibliothek BSAFE Crypto-J 4.0 aus RSA. CA EEM C++ SDK enthält ETPKI 4.1.x, das kryptografische RSA-Bibliotheken verwendet.

CA EEM kann im Nicht-FIPS-Modus oder im Nur-FIPS-Modus ausgeführt werden. Die Verschlüsselungsgrenzen, d. h. die Art der Verschlüsselung in CA EEM, sind in beiden Modi die selben, doch die Algorithmen sind unterschiedlich.

Computerprodukte, die FIPS 140-2-zertifizierte kryptografische Module verwenden, können nur FIPS-genehmigte Sicherheitsfunktionen verwenden wie AES (Advanced Encryption Algorithm), SHA-1 (Secure Hash Algorithm) und Protokolle auf höherer Ebene wie TLS v1.0, die in den FIPS 140-2-Handbüchern ausdrücklich erlaubt sind.

Im Nur-FIPS-Modus verwendet CA EEM die folgenden Algorithmen:

- SHA-1 als standardmäßigen Digest Algorithm, um Kennwörter zu verschlüsseln und Serveranfragen zu signieren. Sie können aus den folgenden Algorithmen im Nur-FIPS-Modus auswählen:
  - SHA1
  - SHA256
  - SHA384
  - SHA512
- TLS v1.0 zur Kommunikation mit externen LDAP-Verzeichnissen, wenn die LDAP-Verbindung TLS verwendet.

Unterstützte Sicherheitsmodi in CA EEM

CA EEM unterstützt zwei Betriebsarten: Nicht-FIPS und Nur-FIPS. Die Funktionalität von CA EEM ist bei beiden Modi gleich. Der Unterschied zwischen diesen beiden Modi sind die kryptografischen Algorithmen, die zur Speicherung und Überprüfung von Kennwörtern sowie zur Übertragung vertraulicher Daten zwischen CA EEM und anderen Produkten wie LDAP-Verzeichnissen, CA SiteMinder usw. verwendet werden.

Nicht-FIPS

Beschreibt den Modus, der nicht mit FIPS kompatible kryptografische Verfahren verwendet. In diesem Modus wird MD5 als Standardalgorithmus verwendet, um vertrauliche Daten zu verschlüsseln und zu entschlüsseln. Neue Installationen oder Upgrades werden immer im Nicht-FIPS-Modus ausgeführt. Im Nicht-FIPS-Modus ist der CA EEM-Server abwärtskompatibel mit den CA EEM-Clients. Zum Beispiel können Sie CA EEM r8.4 SDK verwenden und eine Verbindung mit dem CA EEM r8.4 SP3-Server herstellen.

Nur-FIPS

Beschreibt den Modus, der nur FIPS-kompatible kryptografische Verfahren verwendet. Dieser Modus ist nicht kompatibel mit Clients, die im Nicht-FIPS-Modus ausgeführt werden. Sie können SDK-Clients mit CA EEM r8.4 SP3 im Nur-FIPS-Modus nur mit CA EEM r8.4 SP3-Servern im Nur-FIPS-Modus verwenden.

Konfigurieren von CA EEM-Servern im Nur-FIPS-Modus

Im Nur-FIPS-Modus muss CA EEM konfiguriert werden, um FIPS-kompatible Algorithmen zu verwenden. CA EEM-Server und CA EEM-SDK-Clients können nur kommunizieren, wenn beide im Nur-FIPS-Modus konfiguriert werden. Ähnlicherweise kann ein CA EEM-Server im Nur-FIPS-Modus nur mit einem LDAP-Verzeichnis kommunizieren, für das FIPS-kompatible Algorithmen konfiguriert wurden. Um die CA EEM-Umgebung in einem Nur-FIPS-Modus zu konfigurieren, führen Sie die folgenden Schritte aus:

- Überprüfen Sie die Voraussetzungen für die Konfiguration von CA EEM-Server im Nur-FIPS-Modus
- Konfigurieren Sie CA EEM-Server im Nur-FIPS-Modus

Überprüfen der Voraussetzungen für die Konfiguration von CA EEM-Server im Nur-FIPS-Modus

Dies sind die Voraussetzungen für das Konfigurieren des CA EEM-Servers im Nur-FIPS-Modus:

- Stellen Sie sicher, dass die anderen CA-Produkte (CA ITM, CA ELM usw.), die iGateway verwenden, im Nur-FIPS-Modus ausgeführt werden. iGateway kann nicht sowohl in Nur-FIPS-Modus als auch im Nicht-FIPS-Modus initialisiert werden. Wenn iGateway im Nur-FIPS-Modus initialisiert wird,- müssen alle Produkte iGateway im Nur-FIPS-Modus verwenden. Öffnen Sie die Datei "iGateway.conf" und überprüfen Sie den Wert für die folgende Kennung:

  FIPSMode

  Wenn der Wert dieser Kennung auf "False" gesetzt wurde, bedeutet dies, dass das Produkt iGateway im Nicht-FIPS-Modus ausführt. Im Hinblick auf die vorhandene Konfiguration von iGateway können Sie entscheiden, ob Sie CA EEM im Nur-FIPS-Modus aktivieren möchten.

- Überprüfen Sie die von den anderen CA-Produkten verwendeten Spindle-Versionen, öffnen Sie die Datei "spin.conf" und beachten Sie den Wert für die folgenden Kennungen: <Spindle-Name> und <Version>. Überprüfen Sie unter Verwendung der jeweiligen Produktdokumentation, ob diese Versionen FIPS-kompatibel sind.

  Hinweis: Die Datei "iGateway.conf" und die Datei "spin.conf" sind hier zu finden:

  - **Windows**: %IGW_LOC%
  - **Linux und UNIX**: /opt/CA/SharedComponents/iTechnology

Vor der Konfiguration von CA EEM im Nur-FIPS-Modus

Überprüfen Sie, ob Ihre Umgebung den Mindestanforderungen entspricht, bevor Sie die Umgebung in den Nur-FIPS-Modus wechseln. Drucken Sie sich die folgenden Punkte aus, um sie als Checkliste verwenden zu können:

- ☐ Führen Sie ein Upgrade für Ihren CA EEM-Server auf CA EEM r8.4 SP3 durch.
- ☐ Stellen Sie sicher, dass die mit CA EEM integrierten oder verbundenen Produkte für den Nur-FIPS-Modus konfiguriert werden.

Konfigurieren von CA EEM-Server im Nur-FIPS-Modus

Wenn Sie CA EEM-Server im Nur-FIPS-Modus konfigurieren, verwendet CA EEM ausschließlich FIPS 140-2-kompatible kryptografische Bibliotheken, um vertrauliche Daten zu verschlüsseln und zu entschlüsseln.

Hinweise:

- Verwenden Sie im Nur-FIPS-Modus IE7 (oder höher) oder Firefox 3.0 (oder höher), um die CA EEM-Admin-Benutzeroberfläche anzuzeigen. Weitere Informationen zur Konfiguration von Firefox im FIPS 140-2-Modus finden Sie auf der Firefox-Support-Webseite.
- Die folgende Prozedur gilt auch für die Änderung des Sicherheitsmodus des CA EEM-Servers von Nur-FIPS auf Nicht-FIPS oder umgekehrt.

So konfigurieren Sie CA EEM-Server im Nur-FIPS-Modus

1. Halten Sie den iGateway-Dienst an.
2. Halten Sie den CA Directory-Dienst mit Hilfe der folgenden Befehle an:

   Windows

   ```
   dxserver stop all
   ssld stop
   ```

   Linux und UNIX

   ```
   su - dsa -c "dxserver stop all"
   su - dsa -c "ssld stop"
   ```

3. Öffnen Sie die Datei "iGateway.conf" und stellen Sie die folgende Kennung auf "AN":

   ```
   <FIPSMode>AN<FIPSMode>
   ```

   Hinweis: Um den Modus von Nur-FIPS auf Nicht-FIPS zu verändern, stellen Sie die FIPS-Modus-Kennung auf "AUS".

4. Führen Sie den folgenden Befehl über die Eingabeaufforderung aus:

   Windows

   ssld remove iTechPoz-Server
   ssld install iTechPoz-Server -certfiles "%DXHOME%/config/ssld/personalities" -ca "%DXHOME%/config/ssld/iTechPoz-trusted.pem" -port 21847 -fips

   Linux und UNIX

   su - dsa
   ssld remove iTechPoz-Server
   ssld install iTechPoz-Server -certfiles $DXHOME/config/ssld/personalities -ca $DXHOME/config/ssld/iTechPoz-trusted.pem -port 21847 -fips

   Hinweis: Die Option "-port" gibt den ssld-Port an. Wenn Sie einen anderen ssld-Port konfiguriert haben, ersetzen Sie 21847 in den vorangehenden Befehlen durch die richtige Portnummer. Auch wenn Sie den Sicherheitsmodus von Nur-FIPS auf Nicht-FIPS verändern, verwenden Sie die Befehle in diesem Schritt ohne die Option "-fips".

5. Starten Sie den CA Directory-Dienst mit Hilfe der folgenden Befehle:

   Windows

   ssld start
   dxserver start all

   Linux und UNIX

   su - dsa -c "ssld start"
   su - dsa -c "dxserver start all"

6. Starten Sie den iGateway-Dienst.

   CA EEM ist nun im Nur-FIPS-Modus konfiguriert.

Überprüfen, ob CA EEM-Server im Nur-FIPS-Modus ausgeführt wird

Um zu überprüfen, ob der CA EEM-Server im Nur-FIPS-Modus ausgeführt wird, gehen Sie wie folgt vor:

1. Geben Sie die URL https://*Hostname* oder *IP-Adresse*:5250/spin/eiam/about.csp im Browser ein.

   Die Seite "Info" wird geöffnet.

2. Stellen Sie sicher, dass die Kennung FIPS: aktiviert ist.

   Wenn die Kennung aktiviert ist, bedeutet dies, dass der CA EEM-Server im Nur-FIPS-Modus ausgeführt wird.

Kommunikation zwischen CA EEM-Server und externen LDAP-Verzeichnissen

Die Kommunikation zwischen CA EEM-Server und einem externen Verzeichnis hängt vom Typ der LDAP-Verbindung zwischen den beiden ab: verschlüsselt oder nicht verschlüsselt. Im Folgenden werden die unterstützten Betriebsarten vom CA EEM-Server und externen Verzeichnissen, basierend auf der Verschlüsselung, beschrieben:

Verschlüsselung im CA EEM-Server für LDAP-Kommunikation ist aktiviert

Wenn die verschlüsselte Kommunikation mit einem externen LDAP-Verzeichnis für CA EEM-Server im FIPS-Modus konfiguriert wird, muss das LDAP-Verzeichnis auch konfiguriert werden, um den FIPS-Modus zu verwenden.

Konfigurieren des CA EEM-Servers zur Verwendung von Server-Zertifikaten in einem PKCS#11-Gerät

Um nCipher-PKCS#11-Geräte mit dem CA EEM-Server oder CA EEM-SDK zu verwenden, konfigurieren Sie das nCipher-Gerät und legen Sie die folgenden Eigenschaften folgendermaßen fest:

```
CKNFAST_OVERRIDE_SECURITY_ASSURANCES=all
```

Hinweis: Weitere Informationen zur Konfiguration des nCipher-Geräts mit einem Hardwaretoken finden Sie in der nCipher-Dokumentation.

So konfigurieren Sie den CA EEM-Server, um Zertifikate aus PKCS#11 Geräten zu verwenden:

1. Beenden Sie den iGateway-Service.
2. Öffnen Sie die Datei "iGateway.conf" und bearbeiten Sie die Kennungen <Connector name="Standardport"> CA Portal5250</port>, um die folgenden Werte festzulegen:

   certType

   Gibt den Typ des verwendeten Zertifikats an. Unterstützte Zertifikatstypen sind p12, pem, und p11.

   Standard: pem

   Typ: Untergeordneter Knoten

Verwendung des P11-Zertifikats

<pkcs11Lib/> – Pfad zur PKCS11-Bibliothek, bereitgestellt über Token

<token/> – Token-ID

<userpin/> – Verschlüsselter Benutzer-PIN

<id/> – ID des Zertifikats und privaten Schlüssels

<sensitive/> – Privater Schlüssel ist vertraulich Vertrauliche Schlüssel werden nicht wie Softwareschlüssel konvertiert und die Verschlüsselung funktioniert über Cryptopki-Hardware (nicht vertrauliche Schlüssel können wie vertrauliche Schlüssel behandelt werden, aber vertrauliche Schlüssel können nicht konvertiert oder als nicht vertrauliche Schlüssel betrachtet werden)

Standard: "False" (Falsch)

3. Speichern und schließen Sie die Datei "iGateway.conf".
4. Starten Sie den iGateway-Dienst.

Konfigurieren des CA EEM-Servers zum Speichern von Server-Zertifikaten in einem PKCS#11-Gerät

Um die CA EEM-Zertifikate in einem PKCS#11-Gerät zu speichern, gehen Sie wie folgt vor:

1. Beenden Sie den iGateway-Service.
2. Öffnen Sie die Datei "iGateway.conf" und bearbeiten Sie die <CertificateManager>-Kennungen, um die folgenden Werte festzulegen:

   certType

   Gibt den Typ des verwendeten Zertifikats an. Unterstützte Zertifikatstypen sind p12, pem, und p11.

   Standard: pem

   Typ: Untergeordneter Knoten

Verwendung des P11-Zertifikats

<pkcs11Lib><pkcs11Lib/> – Pfad zur PKCS11-Bibliothek, bereitgestellt über Token

<token><token/> – Token-ID

<userpin><userpin/> – Verschlüsselter Benutzer-PIN

<id><id/> – ID des Zertifikats und privaten Schlüssels

<sensitive><sensitive/> – Privater Schlüssel ist vertraulich Vertrauliche Schlüssel werden nicht wie Softwareschlüssel konvertiert und die Verschlüsselung funktioniert über Cryptopki-Hardware (nicht vertrauliche Schlüssel können wie vertrauliche Schlüssel behandelt werden, aber vertrauliche Schlüssel können nicht konvertiert oder als nicht vertrauliche Schlüssel betrachtet werden). Optional. Standardeinstellung ist "False" (Falsch).

3. Speichern und schließen Sie die Datei "iGateway.conf".
4. Starten Sie den iGateway-Dienst.

Konfigurieren Ihrer Anwendung im Nur-FIPS-Modus

Um Ihre Anwendung im Nur-FIPS-Modus zu konfigurieren, stellen Sie sicher, dass CA EEM-SDK im Nur-FIPS-Modus ist und CA EEM-SDK nur FIPS-kompatible Verfahren zur Verschlüsselung verwendet. Mit der Konfigurationsdatei "eiam.config" in CA EEM-SDK wird die sichere Betriebsart von CA EEM-SDK überprüft. Bevor Sie CA EEM-SDK im Nur-FIPS-Modus konfigurieren, überprüfen Sie das Folgende:

- Stellen Sie sicher, dass Sie über CA EEM-SDK, Version r8.4 SP3 verfügen.
- Migrieren Sie vorhandene P12-Zertifikate, die in CA EEM verwendet werden, in PEM-Zertifikate.
- Initialisieren Sie CA EEM-SDK im Nur-FIPS-Modus.

Migrieren von P12-Zertifikaten Ihrer Anwendung in PEM-Zertifikate.

CA EEM unterstützt P12-, PEM-, und PKCS#11-Zertifikate. Beachten Sie dazu Folgendes:

- P12-Unterstützung wird deaktiviert (nicht verfügbar) im Nur-FIPS-Modus. Als eine Alternative werden nun PEM- und PKCS#11- Zertifikate im Nur-FIPS-Modus unterstützt.

Hinweis: CA EEM C# SDK unterstützt nur PEM-Zertifikate im Nur-FIPS-Modus, P12- und PEM-Zertifikate im Nicht-FIPS-Modus.

Wenn Sie P12-Zertifikate verwenden, migrieren Sie diese in unterstützte Zertifikatsformate des Nur-FIPS-Modus. Verwenden Sie das Hilfsprogramm "igwCertUtil", um P12-Zertifikate in PEM-Zertifikate zu konvertieren. "igwCertUtil" ist ein Hilfsprogramm, um Zertifikate zu konvertieren, erstellen oder löschen. Das Hilfsprogramm befindet sich in dem folgenden Ordner:

Windows

%IGW_LOC%

UNIX und LINUX

$IGW_LOC

Hilfsprogramm "igwCertUtil" – Erstellen, Kopieren, Konvertieren und Löschen von Zertifikaten

Gültig für Windows, UNIX und Linux

Der Befehl zum Erstellen hat das folgende Format:

igwCertUtil -version *Version* -create -cert *inputcert*-Parameter -issuer issuercert-Parameter [-debug] [-silent]

Der Befehl zum Konvertieren hat das folgende Format:

igwCertUtil -version Version -conv -cert *inputcert*-Parameter -target newcert-Parameter [-debug] [-silent]

Der Befehl zum Kopieren hat das folgende Format:

igwCertUtil -version Version -copy -cert *inputcert*-Parameter -target newcert-Parameter [-debug] [-silent]

Der Befehl zum Löschen hat das folgende Format:

igwCertUtil -version Version -delete -cert *cert*-Parameter [-debug] [-silent]

-version *Version*

Gibt beim Erstellen, Konvertieren, Kopieren oder Löschen von Zertifikaten die Version von "igwCertUtil" an. Die Version wird zur Abwärtskompatibilität verwendet. Wenn "igwCertUtil" verändert wird, nimmt die Versionskennung das alte Verhalten an.

-cert *inputcert*-Parameter

Gibt das Zertifikat beim Erstellen, Konvertieren oder Kopieren von Zertifikaten als eine XML-Zeichenfolge an.

-issuer *issuercert*-Parameter

Gibt bei der Erstellung das Zertifikat an, das zur Signatur des neu generierten Zertifikats verwendet wird. Wenn kein Zertifikat angegeben wird, wird ein selbstsigniertes Zertifikat erstellt.

-target *newcert*-Parameter

Gibt die Konfiguration für das neue Zertifikat an, wenn ein vorhandenes Zertifikat konvertiert oder kopiert wird.

-cert *cert*-Parameter

-debug

(Optional) Aktiviert Debugging für "igwCertUtil".

-silent

(Optional) Aktiviert den automatischen Modus für "igwCertUtil".

Die folgenden Fehlercodes werden von "igwCertUtil" zurückgegeben:

- CERTUTIL_ERROR_UNKNOWN (-1): unbekannter oder undefinierter Fehler
- CERTUTIL_SUCCESS (0): erfolgreicher Vorgang
- CERTUTIL_ERROR_USAGE (1): falsche Befehlszeilenargumente übergeben
- CERTUTIL_ERROR_READCERT (2): Zertifikat kann nicht gelesen werden
- CERTUTIL_ERROR_WRITECERT (3): Zertifikat kann nicht geschrieben werden
- CERTUTIL_ERROR_DELETECERT (4): Zertifikat kann nicht gelöscht werden

Beispiel: Konvertieren von P12-Zertifikaten in PEM-Zertifikate

Das folgende Beispiel beschreibt, wie P12-Zertifikate in PEM-Zertifikate konvertiert werden:

```
igwCertUtil -version 4.6.0.0 -conv  -cert
"<Certificate><certType>p12</certType><certURI>testCert.p12</certURI><certPW>Kennwort</certPW></Certificate>" -target "<Certificate><certType>PEM</certType>
<CertURI>testCert.cer</certURI><keyURI>testCert.key</keyURI></Certificate>"
```

Beispiel: Konvertieren von P12-Zertifikaten in PKCS#11-Zertifikate:

```
igwCertUtil -version 4.6.0.0 -conv  -cert
"<Certificate><certType>p12</certType><certURI>testCert.p12</certURI><certPW>Kennwort</certPW></Certificate>" -target "<Certificate><certType>p11</certTyp
><pkcs11Lib>Pfad_zu_pkcs11_Bibliothek</pkcs11Lib><token>pkcs11_Token</token><userpin>Benutzer-PIN</userpin><id>Zertifikat-ID</id></Certificate>"
```

Initialisieren von CA EEM-SDK im Nur-FIPS-Modus

CA EEM-SDK kann im Nur-FIPS-Modus durch das Konfigurieren der Datei "eiam.config" initialisiert werden. Weitere Informationen zur Konfiguration von "eiam.config" finden Sie im Kapitel Konfigurieren von CA EEM-SDK (siehe Seite 35).

Kapitel 6: Sichern und Wiederherstellen von CA EEM Server

Dieses Kapitel enthält folgende Themen:

Sichern des Dateisystems

Es wird empfohlen, CA EEM-Server regelmäßig oder nach dem Vornehmen von Änderungen in den CA EEM-Server-Umgebungen zu sichern. Sie können die Sicherungskopien des CA EEM-Server zum Wiederherstellen von CA EEM-Servern verwenden, wenn dieser beschädigt sein sollte.

Sie müssen die folgenden CA EEM-Dateien und -Ordner sichern:

Datenbeschreibung	Dateinamen unter Windows	Dateinamen unter Linux
Konfigurationsdateien	■ iPoz.conf ■ Eiam.conf ■ iPoz.map ■ Spin.conf ■ iPozDsa.pem ■ iPozRouterDsa.pem ■ logDepot.conf ■ calmReporter.conf	■ iPoz.conf ■ Eiam.conf ■ iPoz.map ■ Spin.conf ■ iPozDsa.pem ■ iPozRouterDsa.pem ■ eiam-type ■ Sponsordateien ■ logDepot.conf ■ calmReporter.conf

Datenbeschreibung	Dateinamen unter Windows	Dateinamen unter Linux
Ereignisinformationen	■ logdepotdb ■ Ordner "calm_catalog" ■ Ordner "calm_archive"	■ logdepotdb ■ Ordner "calm_catalog" ■ Ordner "calm_archive"
Ordner	■ Systemregistrierung ■ iTechnology-Ordner	■ iTechnology-Ordner ■ Umgebungseinstellungen

Sichern von CA EEM-Server-Dateien und -Ordnern

Es wird empfohlen, CA EEM-Server regelmäßig oder nach dem Vornehmen von Änderungen in den CA EEM-Server-Umgebungen zu sichern. Sie können die Sicherungskopien des CA EEM-Servers zum Wiederherstellen des CA EEM-Servers verwenden, wenn dieser beschädigt sein sollte.

So sichern Sie CA EEM-Server-Dateien und -Ordner:

1. Stoppen Sie iGateway.
2. Sichern Sie die Konfigurationsdateien, Ereignisinformationen und Ordner von CA EEM.
3. Sichern Sie die in CA Directory gespeicherten CA EEM-Daten.

   Es werden Sicherungskopien der Server-Konfigurationsdateien, Ereignisse und Ordner von CA EEM erstellt.

Weitere Informationen:

Sichern des Dateisystems (siehe Seite 59)
Sichern der in CA Directory gespeicherten CA EEM-Daten (siehe Seite 63)

Verfahren zum Wiederherstellen

Sie müssen Ihre CA EEM-Daten wiederherstellen, um Folgendes durchführen zu können:

- Wiederherstellen einer beschädigten CA EEM-Installation
- Wiederherstellen einer CA EEM-Serverumgebung, die nicht wie gewünscht funktioniert

So stellen Sie CA EEM-Konfigurationsdateien und -Daten wieder her:

1. Stoppen Sie iGateway.
2. Benennen Sie alle gesicherten CA EEM-".conf"-Dateien in ".conf.merge" um, und kopieren Sie die umbenannten Konfigurationsdateien in den Ordner "iTechnology". Die ".conf.merge"-Dateien sind zum Zusammenführen der gesicherten Konfigurationsdateien mit den neuen Konfigurationsdateien erforderlich.
3. Stellen Sie die CA EEM-Daten wieder her.
4. Starten Sie iGateway.

Weitere Informationen:

Sichern der in CA Directory gespeicherten CA EEM-Daten (siehe Seite 63)

Starten des iGateway-Dienstes

Sie müssen die folgenden Befehle eingeben, um den iGateway-Dienst zu starten:

Windows

net start igateway

Linux und UNIX

$IGW_LOC/S99igateway start

Anhalten des iGateway-Dienstes

Sie müssen die folgenden Befehle eingeben, um den iGateway-Dienst anzuhalten:

Windows

net stop igateway

Linux und UNIX

$IGW_LOC/S99igateway stop

Kapitel 7: Sichern der in CA Directory gespeicherten CA EEM-Daten

Dieses Kapitel enthält folgende Themen:

Einführung in die CA Directory-Terminologie

In diesem Abschnitt wird die in diesem Dokument verwendete CA Directory-Terminologie erläutert:

DSA

Ein *DSA* ist ein Prozess, der den gesamten Namespace eines Verzeichnisses oder einen Teil davon verwaltet.

This also needs to be expanded to make it accessible to readers new to CA Directory. Bei der Installation von CA EEM Server können Sie die folgenden CA Directory-bezogenen Parameter konfigurieren:

DXmanager

DXmanager ist eine Web-Anwendung, mit deren Hilfe Sie Ihr Verzeichnis-Backbone erstellen, konfigurieren, überwachen und steuern können.

DSA-Konsole

Mit Hilfe der *DSA-Konsole* können Sie eine Verbindung zu einem DSA herstellen, um DXserver-Befehle zu geben, Protokollierungsinformationen zu empfangen und als Benutzeragent zu handeln.

DXtools

Die *DXtools* sind mehrere Befehlszeilen-Hilfsprogramme, die mit CA Directory ausgeliefert werden. Mit Hilfe dieser Tools können Sie Verzeichnisse verwalten, mit LDIF-Daten arbeiten, Daten in ein Verzeichnis laden und daraus entfernen und Schemata zur Verwendung mit CA Directory extrahieren und konvertieren.

LDIF (LDAP Data Interchange Format)

LDIF-Dateien sind Textdateien, die Verzeichnisinformationen in LDIF speichern. Sie können LDIF-Dateien dazu verwenden, Verzeichnisinformationen zwischen LDAP-Verzeichnisservern zu übertragen oder Änderungen zu beschreiben, die auf ein Verzeichnis angewendet werden sollen.

Verwenden der DXtools

Sie können die DXtools auf die folgenden Arten ausführen:

- Führen Sie die DXtools-Befehle mit Hilfe der DSA-Konsole auf dem Host aus.
- Führen Sie die DXtools-Befehle mit Hilfe der DSA-Konsole über ein TCP/IP-Netzwerk auf einem Remote-Host aus.
- Nehmen Sie die DXtools-Befehle in Ihre Skripte auf.

Alle Tools geben bei Erfolg den Wert 0 zurück und bei Auftreten eines Fehlers einen Wert ungleich 0.

Umgebungsvariable DXHOME

Für einige Tools muss die Umgebungsvariable DXHOME auf das Stammverzeichnis von DXserver gesetzt werden. Dies erfolgt automatisch bei der Installation von CA Directory.

Einige Tools erwarten, dass die DSA-Konfigurationsdateien sich im Ordner *config* unter dem Pfad in DXHOME befinden.

Beendigungsstatuscodes für die DXtools

Die DXtools verwenden gemeinsame Beendigungscodes, obgleich nicht alle Beendigungscodes für alle Tools gelten. Die Beendigungscodes lauten wie folgt:

0

Erfolg

1

Der entsprechende DSA wird ausgeführt.

2

Eine oder mehrere der Datenspeicherdateien sind bereits vorhanden.

3

Der angegebene Verzeichnisspeicherort existiert entweder nicht, oder es handelt sich nicht um ein Verzeichnis.

4

Die angegebene Datei besitzt den falschen Typ, beispielsweise handelt es sich um ein Verzeichnis.

5

Bei dieser Datei liegt ein Berechtigungsproblem vor.

6

Der vollständige Pfadname der Datenspeicherdatei ist zu lang. Dies kann daran liegen, dass der für das Datenspeicherverzeichnis angegebene Speicherort zu lang ist.

7

Bei dem Versuch, die alten Datenspeicherdateien zu entfernen, ist ein Fehler aufgetreten.

8

Bei dem Versuch, die alten Datenspeicherdateien umzubenennen, ist ein Fehler aufgetreten.

9

Bei dem Versuch, eine der Dateien zu erstellen oder aufzufüllen, ist ein Fehler aufgetreten.

10

Die Größe des Datenspeichers ist kleiner gleich 0.

11

Bei dem Versuch, die Datei zu erstellen, war nicht genug Speicherplatz auf dem Gerät vorhanden oder kein Arbeitsspeicher verfügbar.

12

Der Zugriff war nicht ausreichend, um die Datei zu erstellen oder den Zugriff auf die Datei festzulegen, da möglicherweise die Berechtigungen nicht ausreichten.

13

Die Umgebungsvariable DXHOME ist nicht festgelegt.

14

Die Umgebungsvariable DXHOME ist nicht gültig.

15

Der entsprechende DSA ist bereits vorhanden.

16

Der erstellte DSA konnte nicht gestartet werden. Einzelheiten finden Sie in den zugehörigen Protokolldateien.

17

Es wurden fehlerhafte oder unbekannte Befehlszeilenparameter angegeben.

18

Der entsprechende DSA existiert nicht.

Sichern von CA Directory-Daten

Gehen Sie zum Sichern von CA Directory-Daten wie folgt vor:

1. Stellen Sie eine Verbindung zu einem lokalen DSA her.
2. Erstellen Sie eine Snapshot-Kopie des Datenspeichers des Standard-DSAs, der ausgeführt wird. Dieser Vorgang wird Online-Dump genannt. Verwenden Sie zum Erstellen des Snapshots den folgenden Befehl:

   dump *dxgrid-db*

   Hinweis: Ersetzen Sie zum Sichern von CA EEM "dxgrid-db" durch den DSA-Namen "iTechPoz-ServerN".
3. Verwenden Sie das Tool "DXdumpdb", um den Online-Dump (ZDB-Dateien), d. h. die Snapshot-Kopie des Datenspeichers, in eine LDIF-Datei zu sichern.

Weitere Informationen:

Herstellen einer Verbindung zu einer lokalen DSA-Konsole (siehe Seite 66)
Online-Dump des Datenspeichers (siehe Seite 67)
Befehl "dump dxgrid-db" – Erstellen einer konsistenten Snapshot-Kopie eines Datenspeichers (siehe Seite 68)

Herstellen einer Verbindung zu einer lokalen DSA-Konsole

Sie können unter UNIX oder Windows eine lokale Verbindung zu einem DSA herstellen, wenn ein Konsolen-Port für den entsprechenden DSA festgelegt wurde.

So stellen Sie eine Verbindung zu einer lokalen DSA-Konsole her:

1. Öffnen Sie auf dem Host, auf dem der DSA ausgeführt wird, eine Befehlszeile.
2. Geben Sie folgenden Befehl ein:

   telnet localhost *Lokale-Portnummer*

   Lokale-Portnummer

   Gibt die Konsolen-Portnummer des DSA an, zu dem Sie eine Verbindung herstellen möchten.

Online-Dump des Datenspeichers

Sie können eine konsistente Snapshot-Kopie vom Datenspeicher eines DSA erstellen, der gerade ausgeführt wird (Online-Dump). Der DSA schließt vor der Ausführung des Online-Dumps alle Aktualisierungen ab und startet bis zur Fertigstellung der Kopie keine weiteren Aktualisierungen.

Die Datenspeicherdatei wird in eine Datei kopiert, deren Erweiterung mit ".z" beginnt. Die Datenbankdatei lautet also "*dxgrid-db*.zdb".

Hinweis: Durch jeden Dump wird die vorherige Sicherungsdatei überschrieben. Wenn die Sicherungsdatei erhalten bleiben soll, kopieren Sie sie vor dem nächsten Dump an einen anderen Speicherort.

Befehl "dump dxgrid-db" – Erstellen einer konsistenten Snapshot-Kopie eines Datenspeichers

Mit dem Befehl *dump dxgrid-db* wird eine konsistente Snapshot-Kopie vom Datenspeicher eines DSAs erstellt, der gerade ausgeführt wird (ein Online-Dump). Der DSA schließt vor der Ausführung dieses Befehls alle Aktualisierungen ab und startet bis zur Fertigstellung der Kopie keine weiteren Aktualisierungen.

Die Datenspeicherdatei wird in eine Datei kopiert, deren Erweiterung mit ".z" beginnt. Die Datenbankdatei lautet also "*dxgrid-db*.zdb".

Hinweis: Durch jeden Dump wird die vorherige Sicherungsdatei überschrieben. Wenn die Sicherungsdatei erhalten bleiben soll, kopieren Sie sie vor dem nächsten Dump an einen anderen Speicherort.

Das Tool "DXdumpdb" kann Daten aus einem Datenspeicher exportieren, der mit dem Dump-Befehl erstellt wurde.

Der Befehl besitzt das folgende Format:

dump *dxgrid-db* [period *start period*];

period *start period*

(Optional) Gibt an, dass der Online-Dump in regelmäßigen Intervallen ausgeführt wird.

start

Gibt die Anzahl der Sekunden seit Sonntag 00:00:00 Uhr GMT an.

Hinweis: Die Definition der Startzeit bezieht sich auf GMT und nicht auf Ihre Ortszeit.

period

Definiert die Anzahl der Sekunden zwischen Online-Dumps.

Beispiel: Stündliche Ausführung eines Online-Dumps

Mit dem folgenden Befehl wird stündlich eine Snapshot-Kopie des Datenspeichers erstellt:

dump *dxgrid-db* 0 3600

Hinweis: Stellen Sie sicher, dass Sie unter UNIX einen Cron-Job bzw. unter Windows eine geplante Aufgabe erstellen, um die gesicherte Datei an einen sicheren Speicherort zu kopieren. Durch jeden Dump werden die vorherigen Sicherungsdateien überschrieben.

Verwenden einer LDIF-Datei zum Sichern und Laden von Daten

LDIF-Dateien sind Textdateien, die Verzeichnisinformationen in LDIF speichern. Sie können LDIF-Dateien dazu verwenden, Verzeichnisinformationen zwischen LDAP-Verzeichnisservern zu übertragen oder Änderungen zu beschreiben, die auf ein Verzeichnis angewendet werden sollen.

CA Directory wird mit dem Tool DXdumpdb ausgeliefert, mit dem Sie Daten aus einem Datenspeicher in eine LDIF-Datei sichern können. Sie können dann zu einem späteren Zeitpunkt die Daten aus der LDIF-Datei in einen Datenspeicher laden, um den Verzeichnisinhalt wiederherzustellen.

Sichern eines Verzeichnisses in einer LDIF-Datei

So sichern Sie ein Verzeichnis in einer LDIF-Datei:

1. Melden Sie sich als Benutzer *dsa* (unter UNIX) bzw. als DXserver-Administrator (unter Windows) an.
2. Verwenden Sie den folgenden Befehl, um den Datenspeicher in der LDIF-Datei zu sichern:

   dxdumpdb -f *Dateiname* -z *Name des DSA*

   -f *Dateiname*

   Gibt den Dateipfad und den Dateinamen für den Daten-Dump an.

   -z

   Gibt an, dass DXdumpdb einen Dump aus der Kopie des Datenspeichers erstellt, der mit dem Konsolenbefehl "dump dxgrid-db" erzeugt wird.

   Name des DSA

   Gibt den Namen des DSA an.

Tool DXdumpdb – Exportieren von Daten aus einem Datenspeicher in eine LDIF-Datei

Mit dem Tool DXdumpdb können Sie Daten aus einem Datenspeicher in eine LDIF-Datei exportieren.

Hinweis: Eine Liste der Statuscodes, die von diesem und allen anderen DXtools-Befehlen zurückgegeben werden, finden Sie unter Beendigungsstatuscodes für die DXtools (siehe Seite 64).

Dieser Befehl hat folgendes Format:

dxdumpdb *Optionen DSA*

Optionen

Steht für eine oder mehrere der folgenden Optionen:

-f *Dateiname*

Gibt die Datei an, in die die exportierten Daten übertragen werden sollen. Wenn diese Option nicht angegeben ist, erfolgt die Ausgabe in die Standardausgabe oder auf den Bildschirm.

-v

Der Vorgang wird im ausführlichen Modus ausgeführt. Diese Option aktiviert die Fehler- und Statusverfolgung. Für die Option "-v" müssen Sie auch die Option "-f" angeben.

-z

Gibt an, dass DXdumpdb einen Dump aus der Kopie des Datenspeichers erstellt, der mit dem Konsolenbefehl *dump dxgrid-db* erzeugt wird.

DSA

Definiert den DSA. DXdumpdb sucht in den Konfigurationsdateien dieses DSAs nach dem Datenspeicher, der in eine LDIF-Datei exportiert werden soll.

Beispiel: Extrahieren von Democorp-Daten auf den Bildschirm

Im folgenden Beispiel werden die LDIF-Formatdaten aus dem Datenspeicher des DSAs *democorp* am Bildschirm ausgegeben:

```
dxdumpdb democorp
```

Beispiel: Sichern eines Online-Datenspeicher-Dumps

Im folgenden Beispiel wird ein Online-Datenspeicher-Dump in eine LDIF-Datei exportiert.

```
dxdumpdb -f eembackup -z iTechPoz-ServerN
```

Wiederherstellen von CA Directory-Daten

Gehen Sie zum Wiederherstellen von CA Directory wie folgt vor:

1. Halten Sie den DSA an.
2. Laden Sie mit Hilfe von DXloaddb einen Datenspeicher aus einer LDIF-Datei.

Tool DXloaddb – Laden eines Datenspeichers aus einer LDIF-Datei

Verwenden Sie DXloaddb, um einen Datenspeicher aus einer LDIF-Datei zu laden. Der Datenspeicher muss bereits vorhanden sein. Alle vorherigen Informationen im Datenspeicher werden gelöscht.

Hinweise zur Verwendung:

- Die LDIF-Datei muss nicht sortiert sein.
- DXloaddb erstellt Hash-Werte aus allen Kennworteinträgen in der LDIF-Datei, die in Klartext vorliegen.

  Wenn in der DSA-Konfiguration ein Hash-Algorithmus angegeben ist, wird dieser von DXloadbdb verwendet. Anderenfalls wird SHA-1 verwendet.
- Standardmäßig verwendet DXloaddb die DSA-Konfiguration zur Behandlung von Betriebsattributen:
  - Wenn *op-attrs* = *true* ist, werden alle Betriebsattribute in der LDIF-Datei in den Datenspeicher geladen.

    Für alle Einträge in der LDIF-Datei, die kein "createTimestamp"-Attribut besitzen, wird dieses Attribut zum Datenspeicher hinzugefügt.
  - Wenn *op-attrs* = *false* ist, werden die Betriebsattribute in der LDIF-Datei ignoriert, und DXloaddb erstellt keine Betriebsattribute.

Dieser Befehl hat folgendes Format:

dxloaddb [*Optionen*] *DSA LDIF-Datei*

Optionen

Steht für eine oder mehrere der folgenden Optionen:

-n

Gibt an, dass DXloaddb keine Aktionen ausführt.

-O

Gibt an, dass DXloaddb Standard-Betriebsattribute einschließt, z. B. Attribute für Kennwortrichtlinien (beispielsweise die Anzahl von Anmeldeversuchen) und Zeitstempel. Wenn diese Option angegeben ist, erstellt DXloaddb alle Betriebsattribute, die nicht in der LDIF-Datei definiert sind.

-s

Gibt an, dass DXloaddb für den Datenspeicher die folgenden Statistiken erstellt:

- Gesamtdatengröße in MB
- Gesamtanzahl von Einträgen
- Anzahl von ignorierten Einträgen
- Auffüllmenge in der Datenspeicherdatei in KB
- Durchschnittliche Anzahl von Einträgen pro MB

-v

Gibt die ausführliche Ausgabe an.

LDIF-Datei

Der Name der LDIF-Datei, die in den Datenspeicher geladen werden soll.

DSA

Definiert den DSA, dessen Datenspeicher geladen werden soll.

Beispiel: Erstellen und Laden eines Datenspeichers

Die korrekte Abfolge für das Erstellen und Laden eines Datenspeichers ist:

```
dxnewdb
dxloaddb
```

Beispiel: Laden von LDIF-Daten in einen Datenspeicher

Im folgenden Beispiel werden die Daten aus der Datei "democorp.ldif" in den Datenspeicher "democorp" geladen:

```
dxloaddb democorp democorp.ldif
```

Folgendes könnte ein Teil von "democorp.ldif" sein:

```
dn: o=Democorp, c=US
oc: organization
dn: ou=Administration, o=Democorp, c=US
oc: organizationalUnit
dn: cn=Fred Jones, ou=Administration, o=Democorp, c=US
oc: organizationalPerson
postalAddress: 11 Main Street $ Newtown
surname: Jones
title: Manager
telephonenumber: +1 (123) 456 7890
telephonenumber: +1 (987) 654 3210
dn: ou=Sales, o=democorp, c=US
oc: organizationalUnit
```

"Telephonenumber" ist zweimal vorhanden, weil es sich dabei um ein mehrwertiges Attribut handelt.

Kapitel 8: Konfigurieren des Failover

Dieses Kapitel enthält folgende Themen:

Failover

Die Failover-Funktion gewährleistet den ununterbrochenen Datenfluss und Betrieb, selbst wenn die Daten nicht mehr verfügbar sind.

Damit das Failover von CA EEM funktioniert, müssen Sie eine Anwendung an CA EEM anbinden, das auf einem Server installiert ist, um Informationen zu anderen Servern zu erlangen. Die Informationen über die andere Serverkonfiguration sind in der Datei "iPoz.conf" verfügbar, die für das Failover verwendet wird.

Sie können CA EEM zur Unterstützung von zwei Arten von Failover-Szenarios konfigurieren:

- Datenspeicher-Failover
- Server-Failover (siehe Seite 81)

Hinweis: Bei diesem Szenario gehen wir von den Hostnamen Server1, Server2 bis ServerN aus.

Failover in Anwendungsdatenspeicher

Der CA EEM-Server verwendet CA Directory als Anwendungsdatenspeicher. Dieser Anwendungsdatenspeicher stellt integrierten Support für Failover und Wiederherstellung zur Verfügung. Synchronisieren Sie die folgenden Punkte auf allen Servern in der Failover-Konfiguration:

1. Systemzeit
2. Sicherheitsmodus (Nicht-FIPS oder Nur-FIPS)
3. Anwendungsdatenspeicher
4. Stellen Sie sicher, dass die DNS-Suche korrekt konfiguriert ist

Wichtig! Sichern Sie die Anwendungsdatenspeicher, bevor Sie synchronisieren. Weitere Informationen zur Sicherung der Datenspeicher finden Sie in Sichern von CA EEM-Daten, die in CA Directory gespeichert sind (siehe Seite 63).

Konfigurieren von Failover in Anwendungsdatenspeicher

Hinweis: Führen Sie die Schritte des folgenden Ablaufs auf dem Primärserver aus. Auf dem Sekundärserver auszuführende Schritte werden ausdrücklich erwähnt.

Für diesen Vorgang wird vorausgesetzt, dass Sie den CA EEM-Server mit den folgenden Standardwerten installiert haben:

- DSA-Benutzer: dsa
- Daten-DSA-Port: 509
- Gruppenmitgliedschaft: etrdir

Wenn Sie einen dieser Parameter angepasst haben, ersetzen Sie die Standardwerte mit den benutzerdefinierten Werten.

So konfigurieren Sie den Failover im Anwendungsdatenspeicher

1. Halten Sie mit Hilfe der folgenden Befehle den CA EEM-Dienst auf allen Servern im Failover-Setup an:

   Windows

   net stop igateway
   dxserver stop all
   ssld stop

   Linux und UNIX

   $IGW_LOC/S99igateway stop
   su - dsa -c "dxserver stop all"
   su - dsa -c "ssld stop"

2. Kopieren Sie die folgenden Dateien von allen CA EEM-Sekundärserver auf den Primärserver, zum Beispiel Server1, und in die jeweiligen Ordner:

   Windows

   %DXHOME%\config\knowledge\iTechPoz-*HostnameOfServerN*.dxc
   %DXHOME%\config\knowledge\iTechPoz-*HostnameOfServerN-Router*.dxc
   %DXHOME%\config\ssld\personalities\iTechPoz-*HostnameOfServerN*.pem
   %DXHOME%\config\ssld\personalities\iTechPoz-*HostnameOfServerN*-Router.pem

   Linux und UNIX

   $DXHOME/config/knowledge/iTechPoz-*HostnameOfServerN*.dxc
   $DXHOME/config/knowledge/iTechPoz-*HostnameOfServerN-Router*.dxc
   $DXHOME/config/ssld/personalities/iTechPoz-*HostnameOfServerN*.pem
   $DXHOME/config/ssld/personalities/iTechPoz-*HostnameOfServerN*-Router.pem

3. Kopieren Sie die folgende Datei von den Sekundärservern in einen temporären Ordner auf dem Primärserver Server1:

   UNIX und Linux

   $DXHOME/config/ssld/iTechPoz-trusted.pem

   Windows

   %DXHOME%\config\ssld\iTechPoz-trusted.pem

4. Bearbeiten Sie die Konfigurationsdateien (iTechPoz-*HostnameOfServerN*.dxc) aller Server wie folgt auf Server1:

   Ändern Sie die folgende Zeile ab:

   address = tcp localhost port 509

   #address = tcp HostnameOfServerN port 509, tcp localhost port 509

   #dsa-flags = multi-write

   in:

   #address = tcp localhost port 509

   address = tcp HostnameOfServerN port 509, tcp localhost port 509

   dsa-flags = multi-write

   Hinweise:

   - CA EEM verwendet die Portnummer 509 als Standarddaten-DSA-Port. Wenn Sie für den CA EEM-Server einen benutzerdefinierten Daten-DSA-Port konfiguriert haben, ersetzen Sie 509 mit der benutzerdefinierten Portnummer.
   - Um IP-Adressen an Stelle von Hostnamen zu verwenden, setzen Sie doppelte Anführungszeichen für die IP-Adresse (" ").

5. Bearbeiten Sie "iTechPoz.dxg" von Server1, um Referenzen des Sekundärservers einzuschließen.

   Beispiel:

   ```
   # iTechPoz - iTechnology-Repository
   #Source - Wissensdateien von iTechPozRouter und iTechPoz DSAs.
   source "iTechPoz-HostnameofServer1-Router.dxc";
   source "iTechPoz-HostnameofServer1.dxc";
   source "iTechPoz-HostnameOfServer2-Router.dxc";
   source "iTechPoz-HostnameOfServer2.dxc";
   source "iTechPoz-ServerN-Router.dxc";
   source "iTechPoz-ServerN.dxc";
   ```

6. Erstellen Sie eine neue "iTechPoz-trusted.pem"-Datei durch Verketten der Inhalte von "iTechPoz-trusted.pem" aller Sekundärserver mit Server1.

   Windows

   type <absoluter Pfad zu iTechPoz-trusted.pem von Server2> >> <absoluter Pfad zu iTechPoz-trusted.pem von Server1>

   UNIX oder Linux

   cat <absoluter Pfad zu iTechPoz-trusted.pem von Server2> >> <absoluter Pfad zu iTechPoz-trusted.pem von Server1>

   Beispiel: type "C:\Programme\CA\Directory\dxserver\config\ssld\iTechPoz-trusted_2.pem" >> "C:\Programme\CA\Directory\dxserver\config\ssld\iTechPoz-trusted.pem

7. Verketten Sie den Inhalt von "iTechPoz-trusted.pem" aller Sekundärserver dann mit "iTechPoz-trusted.pem" von Server1.

8. Kopieren Sie die folgenden Dateien vom Primärserver in die jeweiligen Ordner auf allen Sekundärservern:

   Hinweis: Sichern Sie "iTechPoz-trusted.pem" sowie den Daten-DSA und die Routerdateien (iTechPoz*) der Sekundärserver vor dem Kopieren.

   UNIX und Linux

   $DXHOME/config/ssld/iTechPoz-trusted.pem
   $DXHOME/config/ssld/personalities/iTechPoz-*.pem
   $DXHOME/config/knowledge/iTechPoz*

   Windows

   %DXHOME%\config\ssld\iTechPoz-trusted.pem
   %DXHOME%\config\ssld\personalities\iTechPoz-*.pem
   %DXHOME%\config\knowledge\iTechPoz*

9. Bearbeiten Sie "iTechPoz.dxg" auf allen Sekundärservern. Die Datei "iTechPoz.dxg" muss Folgendes enthalten:

   # iTechPoz - iTechnology-Repository
   #Source - Wissensdateien von iTechPozRouter und iTechPoz DSAs.
   source "iTechPoz-HostnameOfServerN-Router.dxc";
   source "iTechPoz-HostnameOfServerN.dxc";
   source "iTechPoz-HostnameOfServer1-Router.dxc";
   source "iTechPoz-HostnameOfServer1.dxc";
   source "iTechPoz-HostnameOfServer2-Router.dxc";
   source "iTechPoz-HostnameOfServer2.dxc";
   source "iTechPoz-ServerKRouter.dxc";
   source "iTechPoz-ServerK.dxc";

   Hinweis: Die Eingaben für den Localhost müssen vor den Eingaben für andere Server angezeigt werden.

10. Ändern Sie die Eigentumsrechte bzw. Gruppenmitgliedschaft der folgenden Dateien auf "dsa" bzw. "etrdir" für alle CA EEM-Server, die auf UNIX oder Linux ausgeführt werden. Führen Sie die folgenden Befehle aus:

    chown dsa:etrdir /opt/CA/Directory/dxserver/config/ssld/iTechPoz-trusted.pem
    chown dsa:etrdir /opt/CA/Directory/dxserver/config/knowledge/iTechPoz*

11. Starten Sie mit Hilfe der folgenden Befehle den CA EEM-Dienst auf allen Servern:

    Windows

    ssld start
    dxserver start all
    net start igateway

    Linux und UNIX

    su - dsa -c "ssld start"
    su - dsa -c "dxserver start all"
    $IGW_LOC/S99igateway start

    Die Konfiguration des Failovers in Anwendungsdatenspeicher wurde gespeichert.

CA EEM-Server-Failover

Hinweis: Stellen Sie sicher, dass für alle Server im Failover-Setup (Server1, Server2 bis ServerN) dieselbe Version von CA EEM Server installiert ist, und synchronisieren Sie deren Systemzeit.

Sie können Server1 so konfigurieren dass er allen Sitzungen und Zertifikaten von allen Servern im Failover-Setup vertraut. Wiederholen Sie die folgende Prozedur für alle Server im Failover-Setup.

So konfigurieren Sie Server1 für das Failover:

1. Geben Sie die URL "https://server1:5250/spin" ein.
2. Wählen Sie "iTech Administrator", und klicken Sie auf "OK".

   Der Anmeldebildschirm wird angezeigt.
3. Geben Sie abhängig von Ihrer Auswahl für die Option "Typ" im Anmeldebildschirm die folgenden Anmeldeinformationen ein:

   Host

   Melden Sie sich als Root-Benutzer oder als Administrator an.
4. Klicken Sie auf die Registerkarte "Konfigurieren", fügen Sie "ServerN" als Hostname im Bereich "Vertrauenswürdige iAuthority Hosts (Trusted iAuthority Hosts)" hinzu, und klicken Sie auf "Vertrauen (Trust)".

   Der Datei "iControl.conf" wird ein Eintrag hinzugefügt, und Server1 behandelt Sitzungen von ServerN als vertrauenswürdig.

   Hinweis: Fügen Sie alle Server des Failover-Setups dem Bereich "Vertrauenswürdige iAuthority Hosts (Trusted iAuthority Hosts)" hinzu.
5. Klicken Sie auf die Registerkarte "iAuthority", geben Sie "Als ServerN bezeichnen (Label as ServerN)" ein, suchen Sie den Speicherort der PEM-Zertifikatsdatei im Bereich "Vertrauenswürdigen Root hinzufügen (Add Trusted Root)", und klicken Sie auf "Vertrauenswürdigen Root hinzufügen (Add Trusted Root)".

   Hinweis: Die PEM-Zertifikatsdatei (rootcert.pem) befindet sich im Verzeichnis "iTechnology" von ServerN.

   Der Datei "iAuthority.conf" wird ein Eintrag hinzugefügt, und Server1 behandelt Sitzungen von ServerN als vertrauenswürdig.

   Hinweis: Fügen Sie allen anderen Servern im Failover-Setup Zertifikatseinträge hinzu.

Konfigurieren von CA EEM-Dateien

Sie müssen CA EEM Server1 so konfigurieren, dass er die Liste der verfügbaren Fallback-Server empfängt, bei denen es sich um replizierte Versionen handelt.

So konfigurieren Sie CA EEM Server1:

1. Öffnen Sie das iTechnology-Verzeichnis von Server1.
   - **Windows**: %IGW_LOC%
   - **Linux und UNIX**: /opt/CA/SharedComponents/iTechnology (Standard)
2. Öffnen Sie die Datei "iPoz.conf", und fügen Sie das folgende Tag hinzu:

   ```
   <BackboneMember>Server2</BackboneMember>
   ```

3. Halten Sie iGateway an und starten Sie es.

   Windows

   ```
   net stop igateway
   net start igateway
   ```

   Linux und UNIX

   ```
   /opt/CA/SharedComponents/iTechnology/S99igateway stop
   /opt/CA/SharedComponents/iTechnology/S99igateway start
   ```

Sie müssen auch CA EEM Server2 so konfigurieren, dass er die Liste der verfügbaren Fallback-Server empfängt, bei denen es sich um replizierte Versionen handelt.

So konfigurieren Sie CA EEM Server2:

1. Öffnen Sie das iTechnology-Verzeichnis von Server2.
   - **Windows**: %IGW_LOC%
   - **Linux und UNIX**: /opt/CA/SharedComponents/iTechnology (Standard)
2. Öffnen Sie die Datei "iPoz.conf", und fügen Sie das folgende Tag hinzu:

   ```
   <BackboneMember>Server1</BackboneMember>
   ```

3. Halten Sie iGateway an und starten Sie es.

   Windows

   ```
   net stop igateway
   net start igateway
   ```

   Linux und UNIX

   ```
   /opt/CA/SharedComponents/iTechnology/S99igateway stop
   /opt/CA/SharedComponents/iTechnology/S99igateway start
   ```

Hinweis: Wiederholen Sie diese Prozedur für alle im Failover-Setup konfigurierten CA EEM-Server.

Kapitel 9: Artefakt-Föderation

Aktivieren der Artefakt-Föderation

Wenn Sie die Artefakt-Föderation verwenden möchten, führen Sie die folgenden Schritte für alle CA EEM-Server in einem Failover-Setup aus.

So aktivieren Sie die Artefakt-Föderation:

1. Beenden Sie den iGateway-Service.
2. Navigieren Sie zur Datei "Poz.conf", und öffnen Sie diese.
3. Bearbeiten Sie die folgende Kennung:

   <ArtifactManager SessionTimeout="10"
   RequestTimeout="30"ArtifactStore="local/federated"></ArtifactManager>

   Wobei

   SessionTimeOut

   die Ablaufzeit für eine exportierte Sitzung in Minuten angibt.

   Standard: 10 Minuten

   Bereich:

   RequestTimeOut

   die Ablaufzeit für eine Startanforderung in Minuten angibt.

   Standard: 30 Minuten

   Bereich:

   Store

   den Speicherort von Artefakten angibt. Wenn für den Wert "lokal" angegeben ist, sind die Artefakte aus einem CA EEM-Server nicht auf anderen CA EEM-Servern in einem Failover-Setup verfügbar. Damit Artefakte auf allen CA EEM-Servern verfügbar sind, muss der Wert dieses Parameters auf "föderiert" gesetzt werden.

   Wert: [lokal|föderiert]

   Standard: lokal

   Hinweis: Die Parameter "SessionTimeOut" und "RequestTimeOut" sind auch in der Datei "eiam.conf" enthalten. Wenn Sie diese Parameter in der Datei "eiam.conf" festlegen, haben die Werte der Datei "eiam.conf" Vorrang.
4. Speichern und schließen Sie die Datei.
5. Starten Sie den iGateway-Service neu.

   Die Artefakt-Föderation ist aktiviert.

Kapitel 10: Integration von CA SiteMinder

Dieses Kapitel enthält folgende Themen:

Integration von CA SiteMinder in CA EEM

Führen Sie im CA SiteMinder Administrator die folgenden Schritte durch, um CA SiteMinder in CA EEM zu integrieren:

- Erstellen Sie in CA SiteMinder einen Agenten für die Kommunikation zwischen CA EEM und dem CA SiteMinder-Richtlinienserver. Stellen sie sicher, dass der Agent 4.x-Agenten unterstützt.
- Richten Sie einen Administrator ein oder verwenden Sie den vorhandenen Standardadministrator "SiteMinder", der auf Systemebene gilt.
- Erstellen Sie ein CA SiteMinder-Benutzerverzeichnis für die Autorisierung, das von CA EEM zum Abrufen von LDAP-Attributen verwendet wird.
- Legen Sie das Feld "UniversalID" so fest, dass ein Benutzer im Verzeichnis eindeutig identifiziert werden kann, z. B. "sAMAccountName" oder Benutzer-ID. Sie können das Feld "UniversalID" über die SiteMinder-UI unter Benutzerverzeichnisse, Eigenschaften, Registerkarte "Benutzerattribute" festlegen.
- Legen Sie auf der Registerkarte "Benutzerattribut" für das Kennwortattribut (RW) "userPassword" fest.
- Erstellen Sie einen CA SiteMinder-Datenspeicher für die Authentifizierung, der von CA EEM zum Authentifizieren von Benutzern verwendet wird.

  Hinweis: Wenn der Autorisierungs- und der Authentifizierungs-Benutzerspeicher identisch sind, verwenden Sie den vorhandenen Benutzerspeicher, der für die Autorisierung erstellt wurde.
- Erstellen Sie einen Bereich (Realm) mit dem Ressourcenfilter (Resource Filter) "/iamt.html".
- Erstellen Sie eine CA SiteMinder-Domäne, und fügen Sie ihr Benutzerverzeichnisse, einen Administrator und einen Bereich (Realm) hinzu.

Weitere Informationen zu CA SiteMinder finden Sie in der Dokumentation zu CA SiteMinder.

Konfigurieren der Protokollierung in CA EEM-Server für CA SiteMinder-Module

So konfigurieren Sie Protokollebenen für die CA SiteMinder-Integration

1. Erstellen Sie eine Datei mit dem folgenden Inhalt, und speichern Sie die Datei mit dem Namen "sm_log.properties" ab:

   ```
   #Dateiname: sm_log.properties
   #Legen Sie die Standardprotokollebene für die Stammregistrierung fest
   .level = INFO
   #Legen Sie die Standardprotokollebene für den Protokollnamen "com.ca.eiam" fest
   com.ca.eiam.level = ALL
   ```

2. Ändern Sie die Protokollebene für "com.ca.eiam" in der Datei "sm.properties" auf einen der folgenden Werte:

   SEVERE

   Gibt die Meldungsebene für schwerwiegende Fehler an.

   WARNING

   Gibt eine Ebene für das Anzeigen von Warnungen an.

   INFO

   Gibt eine Ebene für informative Meldungen an.

   CONFIG

   Gibt eine Ebene für statische Konfigurationsmeldungen an.

   FINE

   Gibt eine Ebene für Informationen zur Ablaufverfolgung an.

   ALL

   Gibt an, dass alle Meldungsebenen protokolliert sind.

3. Speichern Sie die Datei am folgenden Speicherort:

   Windows

   %IGW_LOC%

   Linux und UNIX

   /opt/CA/SharedComponents/iTechnology

4. Beenden Sie den iGateway-Service.

5. Öffnen Sie die Datei "iGateway.conf" vom Speicherort, der in Schritt 3 angegeben wurde, und fügen Sie die folgenden Kennungen innerhalb der von "<JVMSettings></JVMSettings>" hinzu:

   ```
   <Properties name="eiam.sm">
   <system-properties>java.util.logging.config.file=sm_log.properties</system-properties>
   </Properties>
   ```

6. Speichern und schließen Sie die Datei.
7. Starten Sie den iGateway-Dienst.

Kapitel 11: CA EEM-SDK-Protokollierung

Bei Java- und C++-SDKs werden für den neuen Protokollierungsprozess in CA EEM jeweils log4j und log4cxx als Protokollierungs-Frameworks verwendet. Beim älteren Protokollierungsprozess wurde das Hilfsprogramm "safe::util logger" verwendet. Diese neue Funktion bringt Ihnen folgende Vorteile:

- Sie müssen Ihre Anwendung nicht neu starten, wenn Sie Protokollebenen aktualisieren oder ändern.
- Sie können Protokollierungseigenschaften wie Dateiname, Dateigröße, Anzahl der Backup-Protokolldateien usw. verwalten, indem sie die Parameter in der Logger-Konfigurationsdatei bearbeiten.
- Sie können CA EEM-SDK-Protokollmeldungen in Netzwerkaufrufe und Leistungsstatistiken kategorisieren.

Hinweis: Die Protokollierung in einem CA EEM-C#-SDK ist nicht aktualisiert. Sie müssen weiterhin "safe::util" verwenden, um Meldungen in CA EEM-C#-SDKs zu protokollieren.

Mit der Protokollierung können Sie Meldungen, Fehler und Informationen aufzeichnen, die von einem CA EEM-SDK generiert wurden. In einem CA EEM-SDK wird die Protokollierung von folgenden Dateien gesteuert.

- eiam.log4cxx.config
- eiam.log4j.config

Diese drei Dateien gehören zum CA EEM-SDK-Paket und werden standardmäßig im Verzeichnis "bin" abgelegt:

UNIX

/opt/CA/eIAMSDK/bin

Windows

C:\Programme\CA\Embedded IAM SDK\bin

Wissenswertes über Logger-Konfigurationsdateien

Die Logger-Konfigurationsdateien "eiam.log4cxx.config" und "eiam.log4j.config" werden zum Konfigurieren der CA EEM-SDK-Protokollierung verwendet. Diese Dateien enthalten folgende Hauptkomponenten:

- Appender
- Logger
- Root-Logger

Diese Komponenten enthalten konfigurierbare Parameter, mit denen Sie den Protokollierungsprozess je nach Ihren Geschäftsanforderungen anpassen können.

Appender

Ein Appender enthält Parameter, die die Protokollierung der einzelnen Logger steuern. Standardmäßig enthalten die Logger-Konfigurationsdateien folgende Appender:

SDK

Loggt die SDK-Meldungen in einer Protokolldatei. Gibt den Pfad einschließlich des Dateinamens der Protokolldatei an.

Standard: eiam.cppsdk.log

Hinweis: Wenn Sie Ihre Anwendung über den Tomcat-Server unter Windows bereitstellen, ist darauf zu achten, dass Sie anstelle des Rückwärtsschrägstrichs (\) den Vorwärtsschrägstrich (/) für den Pfad verwenden. Wenn Sie den Rückwärtsschrägstrich verwenden, wird die Protokolldatei nicht unter dem von Ihnen angegebenen Pfad, sondern im Apache Tomcat-Ordner erstellt.

Network

Loggt Meldungen in Bezug auf den Netzwerkaufruf in einer Protokolldatei.

Standard:eiam.network.cpp.log

Performance

Loggt Meldungen in Bezug auf den Performance-Aufruf in einer Protokolldatei.

Standard: eiam.performance.cpp.log

Console

Zeigt die Protokollmeldungen auf der Konsole an.

Der SDK-Appender ist standardmäßig aktiviert. Um andere Appender zu aktivieren, müssen die Kommentar-Strings (<!-- und -->) vom jeweils zugehörigen Code entfernt werden.

Ein Appender besteht aus den folgenden konfigurierbaren Parametern.

file

Gibt den Protokolldateinamen des Appenders an.

append

Gibt an, ob eine Gruppe von Protokollmeldungen an die Protokolldatei angehängt werden. Wird als Wert "wahr" angezeigt, wird die Gruppe von Protokollmeldungen an die letzte Protokollmeldung in der Protokolldatei angehängt.

BufferedIO

Gibt an, ob die aktuellste Protokollmeldung gepuffert wird. Wird als Wert "wahr" angezeigt, werden die aktuellsten Protokollmeldungen im Speicher abgelegt, bevor sie in die Protokolldatei geschrieben werden. Dies reduziert I/O-Operationen auf ein Minimum und ist von Vorteil bei einer höheren Protokollebene.

Wert: [*true*|*false*] (wahr/falsch)

Standard*:* "false" (falsch)

Hinweis: Die Standardgröße von "BufferedIO" lautet "8 KB"..

maxFileSize

Gibt die maximale Größe der Protokolldatei an. Überschreitet eine Protokolldatei die maximale Größe, wird ein neuer Protokolldateiname "log.1" angelegt und der Inhalt der Protokolldatei in die Datei "log.1" übertragen. Die Protokolldatei enthält nun die aktuellsten Protokollmeldungen. Überschreitet diese Datei ebenfalls die maximale Größe, wird ein neuer Protokolldateiname "log.2" angelegt. Der Inhalt aus "log.1" wird in die Datei "log.2" übertragen, und der Inhalt der Protokolldatei wird in die Datei "log.1" übertragen.

Standard: 10MB

Minimum: 10KB

Maximum: 2GB

Hinweis: Die Mindestgröße von "maxFileSize" muss größer oder gleich der Größe von "BufferedIO" sein.

maxBackupIndex

Gibt die maximale Anzahl von Sicherungsprotokolldateien an, die zur Aufbewahrung alter Protokolle verwendet werden. Wenn die Anzahl von Protokolldateien den maximalen Backup-Indexwert überschreitet, wird die Datei mit den ältesten Protokollmeldungen gelöscht.

Standard: 1

Minimum: 1

Maximum: 12

ConversionPattern

Gibt die Formatierung einer Protokollmeldung an. Konfigurieren Sie die Format-Modifikatoren und Umwandlungszeichen, um das Umwandlungsformat zu definieren.

Hinweis: Weitere Informationen zu Umwandlungsformaten finden Sie auf der Website www.apache.org unter dem Thema "log4j".

Beispiel: SDK-Appender

```
<appender name="SDK" class="org.apache.log4j.RollingFileAppender">
    <!-- The active sdk log file -->
    <param name="file" value="eiam.cppsdk.log" />
    <param name="append" value="true" />
    <param name="BufferedIO" value="false"/>
    <param name="maxFileSize" value="10000KB" />
    <param name="maxBackupIndex" value="1" />
    <layout class="org.apache.log4j.PatternLayout">
    <!-- The log message pattern -->
    <param name="ConversionPattern" value="%5p %d{ISO8601} [%t] [%c] %m%n"/>
    </layout>
</appender>
```

Appender in "eiam.log4net.config"

Ein Appender enthält Parameter zur Protokollierung. Standardmäßig enthält die Protokollkonfigurationsdatei die folgenden Appender:

SDK

Protokolliert die SDK-Meldungen in einer Protokolldatei. Gibt den Pfad einschließlich des Dateinamens der Protokolldatei an.

Standard: EIAM.C#SDK.log

Hinweis: Wenn Sie Ihre Anwendung unter Tomcat-Server auf Windows bereitstellen, stellen Sie sicher, dass Sie im Pfad Schrägstriche "/" und nicht umgekehrte Schrägstriche "\" verwenden. Wenn Sie umgekehrte Schrägstriche verwenden, wird die Protokolldatei nicht an dem Pfad erstellt, den Sie angegeben haben; stattdessen wird die Protokolldatei im Apache Tomcat-Ordner erstellt.

Netzwerk

Protokolliert die auf Netzwerkaufruf bezogene Meldungen in einer Protokolldatei.

Standard: EIAM.NETWORK.C#SDK.log

Leistung

Protokolliert die auf Leistungsaufruf bezogene Meldungen in einer Protokolldatei.

Standard: EIAM.PERFORMANCE.C#SDK.log

Konsole

Zeigt die Protokollmeldungen in der Konsole an.

SDK-Appender ist standardmäßig aktiviert. Um andere Appender zu aktivieren, entfernen Sie die Kommentare (<!-- und -->) aus dem jeweiligen Code.

Ein Appender besteht aus den folgenden konfigurierbaren Parametern:

file

Gibt den Protokolldateinamen des Appenders an.

appendToFile

Gibt an, ob eine Reihe von Protokollmeldungen an die Protokolldatei angehängt wird. Wenn der Wert "true" (wahr) ist, werden die Protokollmeldungen an die letzte Protokollmeldung der Protokolldatei angehängt.

maxSizeRollBackups

Gibt die Höchstanzahl an Sicherungsprotokolldateien an, die zur Aufbewahrung alter Protokolle dienen. Wenn die Anzahl an Protokolldateien den Wert für "maxSizeRollBackups" überschreitet, wird die Datei mit den ältesten Protokollmeldungen gelöscht.

Standard: 1

Minimum: 1

Maximum: 12

rollingStyle

Gibt an, ob die letzte Protokollmeldung zwischengespeichert wird. Wenn der Wert "true" (wahr) ist, werden die letzten Protokollmeldungen im Speicher behalten, bevor sie in die Protokolldatei geschrieben werden. Dies minimiert E/A-Vorgänge und ist nützlich, wenn die Protokollebene höher ist.

Wert: [*True (Wahr)*|*False (Falsch)*]

Standard*:* "False" (Falsch)

Hinweis: Der Standardwert für "BufferedIO" lautet "8 KB".

maximumFileSize

Gibt die maximale Größe der Protokolldatei an. Wenn eine Protokolldatei die maximale Größe überschreitet, wird eine neue Protokolldatei "log.1" erstellt und der Inhalt der Protokolldatei wird in die Datei "log.1" übertragen. Die Protokolldatei enthält jetzt die letzten Protokollmeldungen. Wenn diese Datei die maximale Größe ebenfalls überschreitet, wird eine neue Protokolldatei "log.2" erstellt, der Inhalt von "log.1" wird auf "log.2" übertragen, und der Inhalt der Protokolldatei wird auf "log.1" übertragen.

Standard: 10 MB

Minimum: 10 KB

Maximum: 2 GB

Hinweis: Die minimale Größe von "maxFileSize" muss größer oder gleich der Größe von "rollingStyle" sein.

ConversionPattern

Gibt das Format der Protokollmeldung an. Konfigurieren Sie die Formatmodifizierer und Umwandlungszeichen, um einen Wert für "ConversionPattern" anzugeben.

Hinweis: Weitere Informationen zu Konvertierungsmustern finden Sie im Abschnitt "log4net" unter www.apache.org.

Logger

Mit Loggern können Sie steuern, wie Netzwerk- und Performance-Protokollmeldungen nach Ebenen kategorisiert und während der Laufzeit angezeigt werden. Standardmäßig sind Netzwerk- und Performance-Logger deaktiviert. Um einen Logger zu aktivieren, müssen die Kommentar-Strings vom jeweils zugehörigen Code entfernt werden.

Ein Logger enthält die folgenden Parameter:

logger name

Gibt den Namen eines Loggers an.

additivity

Gibt an, ob Netzwerk- oder Performance-Protokollmeldungen in der SDK-Protokolldatei dupliziert werden.

Wert: [*true*|*false*] (wahr/falsch)

Standard: false (falsch)

level value

Gibt die Protokollebene eines Loggers an.

Wert: [*Trace*|*Debug*|*Info*|*Warn*|*Error*|*Fatal*|*Off*]

Im Folgenden sind die einzelnen Protokollebenen nach Vorrangigkeit aufgelistet.

Hinweis: Je höher die Protokollebene, desto geringer die Performance von CA EEM.

Trace

Bezeichnet Debugging auf niedriger Ebene. Diese Ebene beinhaltet den Steuerungsfluss und übergibt Argumente.

Debug

Gibt Meldungen an, die für die Problemdiagnose verwendet werden. Diese Ebene enthält Kontextinformationen.

Info

Gibt Kontextinformationen, mit denen die Ausführung in einer Produktionsumgebung auf grober Ebene protokolliert wird.

Warn

Zeigt ein potentielles Problem im System an. Entspricht eine Meldung zum Beispiel der Kategorie "Sicherheit", muss eine Warnmeldung angezeigt werden, wenn ein Wörterbuchangriff erkannt wird.

Error

Zeigt ein schwerwiegendes Problem im System an. Das Problem ist nicht zu beheben und bedarf eines manuellen Eingriffs.

Fatal

Zeigt einen schweren Anwendungsausnahmefehler an.

Off

Zeigt eine nicht vorhandene Protokollierung an.

Hinweis: Die Protokollebene des Standard-SDK-Appenders muss der Ebene "Error" entsprechen.

Beispiel: Performance-Logger

```
<logger name="Perform" additivity="false">
    <level value="trace"/>
    <appender-ref ref="Performance" />
</logger>
```

Root-Logger

Ein Root-Logger steuert die Protokollebene sämtlicher Appender. Wenn jedoch die Protokollebene des referenzierten Appenders im Root-Logger nicht mit der im übergeordneten Appender angegebenen Ebene übereinstimmt, wird die Protokollebene mit geringerer Priorität durch die Protokollebene mit höherer Priorität außer Kraft gesetzt.

Handelt es sich bei der Protokollebene eines Root-Loggers zum Beispiel um eine Error-Ebene und die Protokollebene des Netzwerk-Appenders ist mit "Trace" definiert, hat die Trace-Ebene Vorrang vor der Error-Ebene, und das System berücksichtigt während der Laufzeit Protokollmeldungen mit der Protokollebene "Trace".

Beispiel: Root-Logger

```
<root>
    <priority value="error" />
    <appender-ref ref="SDK" />
    <appender-ref ref="Console" />
</root>
```

Konfigurieren der Logger-Dateien

Mit CA EEM können Sie Protokollmeldungen in Bezug auf folgende Gesichtspunkte konfigurieren: Netzwerk, Performance, Konsole und SDK-Klassen.

So konfigurieren Sie Logger-Dateien:

1. Öffnen Sie die Logger-Konfigurationsdatei "eiam.log4cxx.config" oder "eiam.log4j.config" in einem Texteditor.
2. Aktivieren Sie die Logger und Appender.
3. Aktualisieren Sie die Appender-Parameter.
4. Speichern Sie die Logger-Konfigurationsdatei.

Beispiel für eine eiam.log4cxx.config-Datei:

Das folgende Beispiel veranschaulicht den Inhalt der Datei "eiam.log4cxx.config":

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE log4j:configuration SYSTEM "log4j.dtd">
<!-- Hinweis: Diese Datei wird alle 60 Sekunden vom SDK gelesen -->

<log4j:configuration xmlns:log4j="http://jakarta.apache.org/log4j/">

    <appender name="SDK" class="org.apache.log4j.RollingFileAppender">
        <!-- Die aktive SDK-Protokolldatei -->
        <param name="file" value="eiam.cppsdk.log" />
        <param name="append" value="true" />
        <param name="BufferedIO" value="false"/>
        <param name="maxFileSize" value="10000KB" />
        <param name="maxBackupIndex" value="1" />
        <layout class="org.apache.log4j.PatternLayout">
        <!-- Das Protokollmeldungsformat -->
        <param name="ConversionPattern" value="%5p %d{ISO8601} [%t] [%c] %m%n"/>
        </layout>
    </appender>

    <appender name="Network" class="org.apache.log4j.RollingFileAppender">
        <!-- Die Datei zur Protokollierung von Netzwerkaufrufen -->
        <param name="file" value="eiam.network.cpp.log" />
        <param name="append" value="true" />
        <param name="BufferedIO" value="true"/>
        <param name="maxFileSize" value="10000KB" />
        <param name="maxBackupIndex" value="1" />
        <layout class="org.apache.log4j.PatternLayout">
        <!-- Das Protokollmeldungsformat -->
        <param name="ConversionPattern" value="%5p %d{ISO8601} [%t] [%c] %m%n"/>
        </layout>
    </appender>

    <appender name="Performance" class="org.apache.log4j.RollingFileAppender">
        <!-- Die Datei zur Protokollierung von Performance-Aufrufen -->
        <param name="file" value="eiam.performance.cpp.log" />
        <param name="append" value="true" />
        <param name="BufferedIO" value="true"/>
        <param name="maxFileSize" value="10000KB" />
        <param name="maxBackupIndex" value="1" />
        <layout class="org.apache.log4j.PatternLayout">
        <!-- Das Protokollmeldungsformat -->
        <param name="ConversionPattern" value="%5p %d{ISO8601} [%t] [%c] %m%n"/>
        </layout>
    </appender>

    <appender name="Console" class="org.apache.log4j.ConsoleAppender">
        <!-- Protokollierung auf Konsole -->
```

```
            <layout class="org.apache.log4j.PatternLayout">
            <!-- Das Protokollmeldungsformat -->
            <param name="ConversionPattern" value="%5p %d{ISO8601} [%t] [%c] %m%n"/>
            </layout>
        </appender>

        <!-- Entfernen Sie den Kommentar, um den Performance-Logger zu aktivieren. -->
        <!--
        <logger name="Perform" additivity="false">
            <level value="trace"/>
            <appender-ref ref="Performance" />
        </logger>
        -->

        <!-- Entfernen Sie den Kommentar, um den Netzwerk-Logger zu aktivieren. -->
        <!--
        <logger name="Network" additivity="false">
            <level value="trace"/>
            <appender-ref ref="Network" />
        </logger>
        -->

        <root>
            <priority value="error" />
            <appender-ref ref="SDK" />
            <!-- <appender-ref ref="Console" /> -->
        </root>
    </log4j:configuration>
```

Beispiel einer "eiam.log4net.config"-Datei

Im Folgenden finden Sie ein Beispiel für die Datei "eiam.log4net.config":

```
<?xml version="1.0" encoding="utf-8" ?>

<log4net>
	<appender name="SDK" type="log4net.Appender.RollingFileAppender">
		<file value="EIAM.C#SDK.log" />
		<appendToFile value="true" />
		<maxSizeRollBackups value="1" />
		<maximumFileSize value="10000KB" />
		<rollingStyle value="Size" />
		<layout type="log4net.Layout.PatternLayout">
			<conversionPattern value="%date [%thread] %-5level %logger -
%message%newline" />
		</layout>
	</appender>

	<appender name="Network" type="log4net.Appender.RollingFileAppender">
		<file value="EIAM.NETWORK.C#SDK.log" />
		<appendToFile value="true" />
		<maxSizeRollBackups value="1" />
		<maximumFileSize value="10000KB" />
		<rollingStyle value="Size" />
		<layout type="log4net.Layout.PatternLayout">
			<conversionPattern value="%date [%thread] %-5level %logger -
%message%newline" />
		</layout>
	</appender>

	<appender name="Performance" type="log4net.Appender.RollingFileAppender">
		<file value="EIAM.PERFORMANCE.C#SDK.log" />
		<appendToFile value="true" />
		<maxSizeRollBackups value="1" />
		<maximumFileSize value="10000KB" />
		<rollingStyle value="Size" />
		<layout type="log4net.Layout.PatternLayout">
			<conversionPattern value="%date [%thread] %-5level %logger -
%message%newline" />
		</layout>
	</appender>

	<appender name="ConsoleAppender" type="log4net.Appender.ConsoleAppender">
		<layout type="log4net.Layout.PatternLayout">
			<conversionPattern value="%date [%thread] %-5level %logger -
%message%newline" />
		</layout>
	</appender>
```

```
        <!-- Uncomment to enable Performance Logging -->
        <!--
        <logger name="Perform" additivity="false">
                <level value="ERROR"/>
                <appender-ref ref="Performance" />
        </logger>-->

        <!-- Uncomment to enable Network Logging -->
        <!--<logger name="Network" additivity="false">
                <level value="ERROR"/>
                <appender-ref ref="Network" />
        </logger>-->

        <root>
                <level value="ERROR" />
                <appender-ref ref="SDK" />
                <!--        <appender-ref ref="ConsoleAppender" />      -->
        </root>
</log4net>
```

Beispiel einer "eiam.lo4j.config"-Datei

Im Folgenden finden Sie ein Beispiel für die Datei "eiam.log4cxx.config":

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE log4j:configuration SYSTEM "log4j.dtd">

<!-- Note that this file is read by the sdk every 60 seconds -->

<log4j:configuration xmlns:log4j="http://jakarta.apache.org/log4j/">

        <appender name="SDK" class="com.ca.eiam.log4j.RollingFileAppender">
                <!-- The active sdk log file -->
                <param name="file" value="eiam.javasdk.log" />
                <param name="append" value="true" />
                <param name="BufferedIO" value="false"/>
                <param name="maxFileSize" value="10000KB" />
                <param name="maxBackupIndex" value="1KB" />
                <layout class="com.ca.eiam.log4j.PatternLayout">
                        <!-- The log message pattern -->
                        <param name="ConversionPattern" value="%5p %d{ISO8601} [%t] [%c]
%m%n"/>
                </layout>
        </appender>

        <appender name="Network" class="com.ca.eiam.log4j.RollingFileAppender">
                <!-- The file to log Network calls -->
                <param name="file" value="eiam.network.java.log" />
                <param name="append" value="true" />
                <param name="BufferedIO" value="false"/>
                <param name="maxFileSize" value="10000KB" />
                <param name="maxBackupIndex" value="1KB" />
                <layout class="com.ca.eiam.log4j.PatternLayout">
                        <!-- The log message pattern -->
                        <param name="ConversionPattern" value="%5p %d{ISO8601} [%t] [%c]
%m%n"/>
                </layout>
        </appender>

        <appender name="Performance" class="com.ca.eiam.log4j.RollingFileAppender">
                <!-- The file to log Performance calls -->
                <param name="file" value="eiam.performance.java.log" />
                <param name="append" value="true" />
                <param name="BufferedIO" value="false"/>
                <param name="maxFileSize" value="10000KB" />
                <param name="maxBackupIndex" value="1KB" />
                <layout class="com.ca.eiam.log4j.PatternLayout">
                        <!-- The log message pattern -->
                        <param name="ConversionPattern" value="%5p %d{ISO8601} [%t] [%c]
%m%n"/>
```

```
                </layout>
        </appender>

        <appender name="Console" class="com.ca.eiam.log4j.ConsoleAppender">
                <!-- Logs to Console -->
                <layout class="com.ca.eiam.log4j.PatternLayout">
                        <!-- The log message pattern -->
                        <param name="ConversionPattern" value="%5p %d{ISO8601} [%t] [%c]
%m%n"/>
                </layout>
        </appender>

        <!-- Uncomment to enable Performance Logging -->
        <!--
        <logger name="Perform" additivity="false">
                <level value="trace"/>
                <appender-ref ref="Performance" />
        </logger>-->
        -->

        <!-- Uncomment to enable Network Logging -->
        <!--
        <logger name="Network" additivity="false">
                <level value="trace"/>
                <appender-ref ref="Network" />
        </logger>-->
        -->

        <root>
                <priority value="error" />
                <appender-ref ref="SDK" />
                <!-- <appender-ref ref="Console" /> -->
        </root>

</log4j:configuration>
```

Kapitel 12: Konfigurieren der Unterstützung für externe Verzeichnisserver

Dieses Kapitel enthält folgende Themen:

Konfigurieren eines externen Verzeichnisses mit CA EEM

Wenn Sie verschiedene externe Verzeichnisspeicher zur Authentifizierung und Autorisierung verwenden, konfigurieren Sie CA EEM folgendermaßen:

- Verwenden Sie die Datei "iPoz.conf", um das externe Authentifizierungsverzeichnis mit CA EEM zu konfigurieren
- Verwenden Sie die CA EEM-Admin-Benutzeroberfläche, um den CA EEM-Server mit dem externen Autorisierungsverzeichnis zu konfigurieren.

  Hinweis: Weitere Informationen zur Konfiguration von Referenzen zu externen Verzeichnissen finden Sie in der Online-Hilfe.

Um den CA EEM-Server für die Verwendung eines externen Verzeichnisses zur Authentifizierung zu festzulegen, konfigurieren Sie die folgenden Optionen in der Datei "iPoz.conf", die nach der Installation hier zu finden ist: /CA/SharedComponents/iTechnology.

Hinweis: Halten Sie "iGateway" an, bevor Sie die Datei "iPoz.conf", und starten Sie es danach neu.

UseExternalAuthDirectory

Gibt an, ob Sie ein anderes externes Verzeichnis für die Authentifizierung verwenden möchten. Geben Sie "Wahr (True)" ein, um ein anderes externes Verzeichnis zu verwenden. Die Standardeinstellung lautet "Falsch (False)".

ExternalAuthDirType:

Kennzeichnet den Typ des externen Verzeichnisses. Zu den gegenwärtig unterstützten Typen gehören CA Identity Manager, Custom Mapped Directory, Microsoft Active Directory, Novell eDirectory, Novell eDirectory-CN und Sun One Directory.

ExternalAuthDirUserDn

Legt die "UserDn" für den Typ des festgelegten externen Verzeichnisses fest.

ExternalAuthDirPassword

Legt das Benutzerkennwort im verschlüsselten Format fest.

Hinweis: Verschlüsseln Sie das Kennwort mit Hilfe des folgenden Befehls und fügen Sie es in die Datei "ipoz.conf" ein.

/iTechnology/safex -munge <*Kennwort in Klartext*>

ExternalAuthDirHost

Legt den Hostnamen fest, auf dem das externe Verzeichnis konfiguriert wurde.

ExternalAuthDirPort

Legt den Port fest, der das externe Verzeichnis überwacht.

ExternalAuthDirUserSearchPreFilter

Legt den Vorsuchfilter für das externe Verzeichnis fest. Sie können nach beliebigen Objektklassen suchen, wie beispielsweise Benutzern.

ExternalAuthDirUserSearchPostFilter

Legt den Nachsuchfilter für das externe Verzeichnis fest. Sie können nach beliebigen Objektklassen suchen, wie beispielsweise Benutzern.

ExternalDirCacheFolder

Gibt an, ob der CA EEM-Server die externen Verzeichnisordner zwischenspeichern muss. Wenn diese Kennung auf "True" (Wahr) festgelegt wird, speichert der CA EEM-Server die externen Ordner zwischen und Sie können auf diese Ordner über die CA EEM-Admin-Benutzeroberfläche zugreifen. Wenn diese Kennung auf "False" (Falsch) gesetzt wird, zeigt CA EEM die externen Verzeichnisordner nicht in der CA EEM-Admin-Benutzeroberfläche an.

Wert:[True (Wahr)|False (Falsch)].

Standard: **"True" (Wahr)**

Konfigurieren des CA EEM-Servers, um für Schrägstriche in von externen Verzeichnissen zurückgegebenen definierten Namen Escapezeichen zu setzen

Um den CA EEM-Server für die Verwendung eines externen Verzeichnisses zur Authentifizierung zu festzulegen, konfigurieren Sie die folgende Option in der Datei "iPoz.conf", die nach der Installation hier zu finden ist: /CA/SharedComponents/iTechnology.

Hinweis: Halten Sie "iGateway" an, bevor Sie die Datei "iPoz.conf", und starten Sie es danach neu.

ExternalDirEscapeSlash

Gibt an, ob in CA EEM in definierten Namen, die aus den externen Verzeichnissen zurückgegeben werden, für Schrägstriche "/" ein Escapezeichen gesetzt werden muss. Setzen Sie diese Kennung auf "True" (Wahr), wenn für Schrägstriche in CA EEM ein Escapezeichen gesetzt werden soll.

Hinweis: CA EEM muss konfiguriert werden, um für Schrägstriche in den definierten Namen Escapezeichen zu setzen, ansonsten kann CA EEM Objekte möglicherweise nicht ordnungsgemäß abrufen.

Wert: [True (Wahr)|False (Falsch)]

Standard: **"False" (Falsch)**

Failover-Unterstützung für externe Verzeichnisserver konfigurieren

Sie können die Fallback-Funktion von CA EEM erweitern, damit ein Fallback auf einen anderen externen Verzeichnisserver durchgeführt wird, bei dem es sich um eine replizierte Version des Servers handelt.

Nehmen Sie hierzu die Zuordnung in der Datei "iPoz.conf" vor.

Hinweis: Sie müssen iGateway anhalten, bevor Sie Änderungen an der Datei "iPoz.conf" vornehmen und den Dienst danach erneut starten.

ExternalDirHostBackup

Gibt den Hostnamen des replizierten externen Verzeichnisservers an.

ExternalAuthDirHostBackup

Gibt den Hostnamen des anderen externen Verzeichnisservers an, der für die Benutzerauthentifizierung verwendet werden soll.

Verbinden mit LDAP-Servern über TLS

Um eine TLS-Verbindung zum LDAP-Server herzustellen, müssen Sie den LDAP-Server so konfigurieren, dass er anonyme Zertifikate akzeptiert. Um EEM so zu konfigurieren, dass eine Verbindung mit LDAP über TLS hergestellt wird, gehen Sie wie folgt vor:

So konfigurieren Sie CA EEM, um eine Verbindung mit LDAP über TLS herzustellen

1. Melden Sie sich an der CA EEM-Benutzeroberfläche an:
2. Klicken Sie auf "Konfigurieren" und "EEM-Server".
3. Klicken Sie auf "Globale Benutzer/Globale Gruppen".

   Der Bereich "EEM-Server-Konfiguration" wird angezeigt.
4. Klicken Sie auf die Option "Von externem Verzeichnis referenzieren".
5. Geben Sie die Konfigurationsdetails ein.

   Hinweis: Weitere Informationen zu Konfigurationsdetails finden Sie in der Online-Hilfe.
6. Wählen Sie "TLS (Transport Layer Security) verwenden" aus.
7. Klicken Sie auf "Speichern".

Verbinden mit LDAP-Servern über SSL

Sie benötigen die folgenden Zertifikate, um eine SSL-Verbindung zu LDAP-Servern herzustellen:

Zertifizierungsstellenzertifikat

Sie erhalten dieses Zertifikat von einer Zertifizierungsstelle, z. B. von Verisign oder Thwate. Dieses Zertifikat gibt an, dass von dieser Zertifizierungsstelle ausgestellte Zertifikate gültig und vertrauenswürdig sind.

LDAP-Server-Zertifikat

Sie müssen dieses Zertifikat bei einer vertrauenswürdigen Zertifizierungsstelle anfordern. Dieses Zertifikat enthält Informationen zum LDAP-Server und identifiziert den LDAP-Server beim Client.

Hinweis: CA EEM unterstützt nur ".pem"-Zertifikate für SSL-Verbindungen.

Verbinden von CA EEM mit dem LDAP-Server über SSL

Der folgende Prozess erläutert, wie der CA EEM-Server und der LDAP-Server über SSL kommunizieren.

1. Der CA EEM-Server stellt mit Hilfe eines Zertifizierungsstellenzertifikats eine Verbindung zum LDAP-Server her.
2. Der LDAP-Server verifiziert das Zertifizierungsstellenzertifikat, und wenn das Zertifikat gültig ist, baut er einen Handshake mit dem CA EEM-Server auf.
3. Der LDAP-Server sendet während des Handshakes seinen öffentlichen Schlüssel an den CA EEM-Server. Der öffentliche Schlüssel wird zur Verschlüsselung von Daten verwendet, die an den LDAP-Server gesendet werden.
4. Der CA EEM-Server verschlüsselt Daten mit Hilfe des öffentlichen Schlüssels und sendet die Daten an den LDAP-Server.
5. Der CA EEM-Server sendet den Benutzernamen und das Kennwort zur Authentifizierung mit dem LDAP-Server.

Konfigurieren der SSL-Verbindungen

Führen Sie die folgenden Schritte aus, um die SSL-Kommunikation zwischen dem LDAP-Server und dem CA EEM-Server zu konfigurieren:

1. Konfigurieren Sie den LDAP-Server für die Verwendung von Zertifikaten.
2. Konfigurieren Sie den CA EEM-Server für die Kommunikation über SSL.

Konfigurieren Sie den LDAP-Server für die Verwendung von SSL-Zertifikaten

Führen Sie die folgenden Schritte aus, um den LDAP-Server für die Verwendung von SSL zu konfigurieren:

1. Beschaffen Sie ein Zertifizierungsstellenzertifikat, und installieren Sie das Zertifikat im Speicher für vertrauenswürdige Zertifikate auf Ihrem LDAP-Server.
2. Fordern Sie bei der Zertifizierungsstelle ein Server-Zertifikat an, und installieren Sie das Zertifikat im Server-Zertifikatspeicher Ihres LDAP-Servers.
3. Aktivieren Sie für den LDAP-Server das Annehmen von SSL-Verbindungen.

Aktivieren von SSL in CA EEM Server

So aktivieren Sie SSL im Server:

1. Kopieren Sie das Zertifikat der Zertifizierungsstelle vom LDAP-Server, und speichern Sie es auf dem Computer, auf dem CA EEM Server ausgeführt wird.
2. Öffnen Sie die Datei "ipoz.conf", und bearbeiten Sie die folgenden Tags:

   <ExternalDirSSL>

   Gibt an, ob die SSL-Kommunikation aktiviert oder deaktiviert ist. Sie müssen dieses Tag auf "true" setzen, um die SSL-Kommunikation zu aktivieren.

   <ExternalDirCACertPath>

   Gibt den Pfad an, unter dem das Zertifikat der Zertifizierungsstelle auf dem Computer gespeichert wird, auf dem CA EEM Server ausgeführt wird.
3. Starten Sie iGateway neu.

Kapitel 13: Konfigurieren der Unterstützung für eine große Anzahl an Richtlinien

Dieses Kapitel enthält folgende Themen:

Unterstützung für eine große Anzahl an Richtlinien

Hinweis: CA EEM unterstützt nur in C++-SDK-fähigen Client-Umgebungen eine große Anzahl an Richtlinien.

Sie müssen CA EEM Server und die Clients konfigurieren, bevor Sie Anwendungen registrieren, die eine große Anzahl an Richtlinien verwenden.

Hinweis: CA EEM unterstützt auf der HP-UX-Plattform bis zu 20.000 Richtlinien.

Konfigurieren zusätzlicher Einstellungen für CA EEM Server unter AIX

Führen Sie die folgenden zusätzlichen Schritte aus, um CA EEM Server für die Unterstützung einer großen Anzahl an Richtlinien unter AIX zu konfigurieren.

So konfigurieren Sie CA EEM Server unter AIX:

1. Ändern Sie die Netzwerkeinstellungen, indem Sie in der AIX-Befehlszeile den folgenden Befehl eingeben:

   ```
   no -o tcp_nodelayack=1
   ```

2. Erhöhen Sie das Prozesslimit, indem Sie in der AIX-Befehlszeile den folgenden Befehl eingeben:

   ```
   ulimit -d unlimited
   ulimit -f unlimited
   ```

Client-Konfiguration

Sie müssen den Client für die Unterstützung einer großen Anzahl an Richtlinien konfigurieren.

Konfigurieren von Clients für alle Betriebssysteme

Damit die Bereitstellung einer großen Anzahl an Richtlinien unterstützt wird, müssen Sie Clients für alle Betriebssysteme konfigurieren:

- Verlängern Sie die Aktualisierungszeit des Anwendungs-Caches, um Cache-Aktualisierungen während der Registrierung von Anwendungen mit Safex zu vermeiden.

  Weitere Informationen zur Cache-Aktualisierung finden Sie im *Programmierhandbuch*.

  Hinweis: Es wird empfohlen, die Cache-Aktualisierungszeit während der Registrierung auf 3600 Sekunden zu setzen, um Cache-Aktualisierungen während der Registrierung zu vermeiden. Ändern Sie die Cache-Aktualisierungszeit nach der Registrierung auf 30 Sekunden. Dies ist die Standardeinstellung.

- Aktivieren Sie die zuverlässige Ereigniszustellung.

  Weitere Informationen zur zuverlässigen Ereigniszustellung finden Sie im *Programmierhandbuch*.

Kapitel 14: Archivierung von Ereignissen

Dieses Kapitel enthält folgende Themen:

Übersicht

Mit Hilfe von CA EEM können Sie Berichte über durch CA EEM Server generierte Ereignisse generieren und verwalten. Das Archivierungssystem sortiert die archivierten Dateien nach den folgenden drei Status:

Warme Datenbankdateien

Die Archivdateien, die erstellt werden, sobald die Anzahl der Ereignisse die maximale Anzahl an Zeilen in einer Ereignisdatenbank überschreitet. Warme Archivdateien stehen für Abfragen und Berichte aus CA EEM Server zur Verfügung. In eine warme Datenbankdatei können keine Daten eingefügt werden. Warme Datenbankdateien sind in CA EEM Server nur für die Anzahl an Tagen verfügbar, die in den Ereignisprotokolleinstellungen unter "Max. Anzahl an Archivierungstagen" angegeben ist.

Kalte Datenbankdateien

Die Archivdateien mit dem warmen Status, die manuell an einem anderen Speicherort gesichert werden. Bei einer kalten Datenbankdatei können keine Abfragen durchgeführt und keine Berichte erstellt werden. Kalte Datenbankdateien müssen "entfrostet" werden, bevor sie für Abfragen oder Berichte verwendet werden können.

Entfrostete Datenbankdateien

Die Archivdateien mit dem kalten Status, die wiederhergestellt werden, so dass Benutzer mit CA EEM Server Abfragen durchführen oder Berichte erstellen können. Entfrostete Datenbankdateien sind im Archivverzeichnis nur für die Anzahl von Stunden verfügbar, die in den Ereignisprotokolleinstellungen unter "Ereignisrichtlinie" angegeben ist.

So ändern Sie die Ereignisprotokolleinstellungen:

1. Melden Sie sich bei CA EEM an.

   Die Startseite von CA EEM wird angezeigt.

2. Klicken Sie auf "Berichte verwalten", "Konfiguration", "Dienste", "Ereignisprotokolleinstellungen".

   Die Ereignisprotokolleinstellungen werden angezeigt.

Hinweis: Weitere Informationen zum Konfigurieren von Diensten für die Verwaltung von Berichten finden Sie in der *Online-Hilfe*.

Hilfsprogramm zum Entfrosten von kalten Datenbankdateien

CA EEM bietet ein Hilfsprogramm zum Entfrosten von kalten Datenbankdateien. Sie müssen die Dateien wiederherstellen und entfrosten, so dass sie vom kalten in den warmen Status übergehen, bevor Sie Abfragen für die Dateien ausführen und Live-Berichte anzeigen können. Das Hilfsprogramm "sem" bietet die entsprechende Funktionalität. Sie können das Hilfsprogramm "sem" beim Support unter http://supportconnect.ca.com herunterladen.

So richten Sie das Hilfsprogramm "sem" ein:

1. Extrahieren Sie die komprimierten Dateien für das Hilfsprogramm "sem".
2. Legen Sie abhängig von Ihrem Betriebssystem die Umgebungsvariablen fest:

   Linux oder Solaris

   Export LD_LIBRARY_PATH = <*sem-Extrahierungsordner*>:$LD_LIBRARY_PATH

   AIX

   Export LIBPATH = <*sem-Extrahierungsordner*> :$LIBPATH

   HP-UX

   Export SHLIB_PATH= <*sem-Extrahierungsordner*> :$SHLIB_PATH

Hinweis: Unter Windows müssen Sie zum Einrichten des Hilfsprogramms "sem" über die Befehlszeile in den extrahierten Ordner wechseln und die Datei "sem.exe" ausführen.

Syntax des Hilfsprogramms "sem"

Das Hilfsprogramm "sem" besitzt die folgende Syntax:

sem -h <Hostname> -u <Benutzer> -p <Kennwort> -listcolddb | -defrost <Archiv>

-h

Gibt den Hostnamen des Computers an, auf dem die kalten Datenbankdateien gespeichert werden.

-u

Gibt den Benutzernamen an, der für die Authentifizierung mit CA EEM Server verwendet wird.

-p

Gibt das Kennwort für einen Benutzernamen an, der für die Authentifizierung mit CA EEM Server verwendet wird.

-listcolddb

Listet alle kalten Datenbankdateien auf, die auf dem Host-Computer gespeichert sind.

-defrost <Archiv>

Entfrostet die angegebene Archivdatei.

-fips

Gibt an, dass das Hilfsprogramm "sem" FIPS-kompatible Algorithmen verwendet.

Hinweis: Das Hilfsprogramm "sem" muss mit der Option "-fips" verwendet werden, wenn der CA EEM-Server in einem Nur-FIPS-Modus konfiguriert ist.

In der folgenden Tabelle werden die Rückgabewerte des Hilfsprogramms "sem" erläutert:

Rückgabewert	Beschreibung
0	Erfolg
1	Ungültige Argumente
2	Ungültiger Benutzername
3	Authentifizierung fehlgeschlagen
4	Kalte Datenbankdateien konnten nicht aufgelistet werden
5	Kalte Datenbankdatei konnte nicht entfrostet

Rückgabewert	Beschreibung
	werden
6	Fehler bei der Initialisierung

Entfrosten von kalten Datenbankdateien

Sie müssen die Dateien wiederherstellen und entfrosten, so dass sie vom kalten in den warmen Status übergehen, bevor Sie Abfragen für die Dateien ausführen und Live-Berichte anzeigen können.

Hinweis: Vor dem Entfrosten müssen die kalten Datenbankdateien in das Archivverzeichnis "iTechnology\calm_archive" kopiert werden.

So können Sie kalte Datenbankdateien wiederherstellen und verfügbar machen:

1. Kopieren Sie den gesicherten Ordner "calm_archive" in den derzeitigen Ordner "calm_archive".
2. Führen Sie das Hilfsprogramm "sem" über die Befehlszeile aus, um eine Liste aller kalten Datenbankdateien abzurufen.

   sem -h <*Hostname*> -u <*Benutzername*> -p <*Kennwort*> -listcolddb

   CA EEM-Server im Nur-FIPS-Modus

   sem -h <*Hostname*> -u <*Benutzername*> -p <*Kennwort*> -fips -listcolddb
3. Führen Sie das Hilfsprogramm "sem" aus, um die kalten Datenbankdateien zu entfrosten.

   sem -h <*Hostname*> -u <*Benutzername*> -p <*Kennwort*> -defrost <*Archiv*>

   CA EEM-Server im Nur-FIPS-Modus

   sem -h <*Hostname*> -u <*Benutzername*> -p <*Kennwort*> -fips -defrost <*Archiv*>

   Die kalten Datenbankdateien werden wiederhergestellt und entfrostet.