WebSphere Application Server

Caching Proxy Administratorhandbuch

Version 6.0.1

GC12-3421-01

WebSphere Application Server

Caching Proxy Administratorhandbuch

Version 6.0.1

GC12-3421-01

Anmerkung

Vor Verwendung dieser Informationen und des darin beschriebenen Produkts sollten die allgemeinen Informationen unter „Bemerkungen" auf Seite 291 gelesen werden.

- Die IBM Homepage finden Sie im Internet unter: **ibm.com**
- IBM und das IBM Logo sind eingetragene Marken der International Business Machines Corporation.
- Das e-business-Symbol ist eine Marke der International Business Machines Corporation.
- Infoprint ist eine eingetragene Marke der IBM.
- ActionMedia, LANDesk, MMX, Pentium und ProShare sind Marken der Intel Corporation in den USA und/oder anderen Ländern.
- C-bus ist eine Marke der Corollary, Inc. in den USA und/oder anderen Ländern.
- Java und alle auf Java basierenden Marken und Logos sind Marken der Sun Microsystems, Inc. in den USA und/oder anderen Ländern.
- Microsoft Windows, Windows NT und das Windows-Logo sind Marken der Microsoft Corporation in den USA und/oder anderen Ländern.
- PC Direct ist eine Marke der Ziff Communications Company in den USA und/oder anderen Ländern.
- SET und das SET-Logo sind Marken der SET Secure Electronic Transaction LLC.
- UNIX ist eine eingetragene Marke der Open Group in den USA und/oder anderen Ländern.
- Marken anderer Unternehmen/Hersteller werden anerkannt.

Zeite Ausgabe (März 2005)

Diese Ausgabe gilt für:
WebSphere Application Server Version 6.0.1

Außerdem gilt diese Ausgabe für alle nachfolgenden Releases und Änderungen, sofern in den neuen Ausgaben keine anderen Hinweise enthalten sind.

Diese Veröffentlichung ist eine Übersetzung des Handbuchs
IBM WebSphere Application Server Caching Proxy Administration Guide,
IBM Form GC31-6857-01,
herausgegeben von International Business Machines Corporation, USA

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

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

Änderung des Textes bleibt vorbehalten.

Herausgegeben von:
SW TSC Germany
Kst. 2877
März 2005

Inhaltsverzeichnis

Abbildungsverzeichnis

Inhalt des Handbuchs

Dieses Vorwort beschreibt Zielgruppe, Zweck und Aufbau des Handbuchs, Eingabehilfen, Konventionen und Terminologie sowie Referenzliteratur.

Zielgruppe

Das *Caching Proxy Administratorhandbuch* wurde für erfahrene Netz- und Systemadministratoren geschrieben, die mit ihrem Betriebssystem und mit der Bereitstellung von Internet-Services vertraut sind. Vorkenntnisse in Caching Proxy sind nicht erforderlich.

Dieses Handbuch bietet keine Unterstützung für frühere Releases von Caching Proxy.

Konventionen und Terminologie in diesem Handbuch

Diese Dokumentation richtet sich in Bezug auf Typografie und Hervorhebung nach den folgenden Konventionen.

Tabelle 1. In diesem Handbuch verwendete Konventionen

Konvention	Bedeutung
Fettschrift	Bei Bezugnahme auf grafische Benutzerschnittstellen (GUIs) werden Menüs, Menüpunkte, Titel, Knöpfe (Schaltflächen), Symbole und Ordner in Fettschrift dargestellt. Die Fettschrift kann auch zum Hervorheben von Befehlsnamen verwendet werden, die sonst mit dem Begleittext verwechselt werden könnten.
`Monospace-Schrift`	Kennzeichnet Text, der an der Eingabeaufforderung eingegeben werden muss. In Monospace-Schrift werden auch Anzeigetexte, Codebeispiele und Dateiauszüge hervorgehoben.
Kursivschrift	Kennzeichnet Variablenwerte, die eingegeben werden müssen (z. B. müssen Sie *Dateiname* durch den Namen einer Datei ersetzen). Kursivschrift wird außerdem für Hervorhebungen und Handbuchtitel verwendet.
Strg-*x*	Diese Angabe kennzeichnet eine Tastenkombination mit der Steuerungstaste. *x* steht hier für eine Tastenbezeichnung. Beispielsweise bedeutet die Angabe <**Strg-c**>, dass Sie die Taste **Strg** gedrückt halten und gleichzeitig die Taste **c** drücken sollen.
Eingabetaste	Bei dieser Angabe müssen Sie die Eingabetaste drücken.
`%`	Steht in der Linux- und UNIX-Befehlsumgebung für die Aufforderung zur Eingabe eines Befehls, der keine Root-Berechtigung erfordert.
`#`	Steht in der Linux- und UNIX-Befehlsumgebung für die Aufforderung zur Eingabe eines Befehls, der Root-Berechtigung erfordert.
`C:\`	Steht für die Eingabeaufforderung in der Windows-Umgebung.
Befehlseingabe	Wenn Sie aufgefordert werden, einen Befehl einzugeben, geben Sie den Befehl ein und drücken Sie dann die Eingabetaste. Beispiel: Die Anweisung "Geben Sie den Befehl **ls** ein" bedeutet, dass Sie an der Eingabeaufforderung **ls** eingeben und anschließend die Eingabetaste drücken müssen.

Tabelle 1. In diesem Handbuch verwendete Konventionen (Forts.)

Konvention	Bedeutung
[ ]	In eckigen Klammern werden optionale Elemente in Syntaxbeschreibungen angegeben.
{ }	In geschweiften Klammern werden Listen in Syntaxbeschreibungen angegeben, aus denen Sie einen Eintrag auswählen können.
|	Dieses Zeichen trennt in Syntaxbeschreibungen die Einträge in einer Auswahlliste, die in geschweiften Klammern ({ }) angegeben ist.
...	Drei Punkte in Syntaxbeschreibungen bedeuten, dass der vorherige Eintrag einmal oder mehrmals wiederholt werden kann. Drei Punkte in Beispielen bedeuten, dass Informationen weggelassen wurden, um die Darstellung zu vereinfachen.

Zugriffsmöglichkeiten

Features zur Erleichterung des Zugriffs helfen körperbehinderten Benutzern (mit eingeschränkter Beweglichkeit oder Sehschwäche), Softwareprodukte erfolgreich anzuwenden. WebSphere Application Server Version 6.0.1 bietet im Wesentlichen die folgenden Features für verbesserte Zugriffsmöglichkeiten an:

- Sie können ein Bildschirmleseprogramm und einen digitalen Sprachsynthesizer verwenden, um zu hören, was auf dem Bildschirm angezeigt wird. Für die Dateneingabe und die Navigation in der Benutzerschnittstelle können Sie Spracherkennungssoftware wie IBM ViaVoice einsetzen.
- Für die Ausführung von Funktionen können Sie an Stelle der Maus die Tastatur benutzen.
- Für die Konfiguration und Verwaltung der Features von Application Server können Sie an Stelle der bereitgestellten grafischen Oberflächen auch Standardtexteditoren oder Befehlszeilenschnittstellen verwenden. Nähere Informationen zu den Zugriffsmöglichkeiten für bestimmte Features finden Sie in der Dokumentation zu diesen Features.

Senden von Kommentaren

Ihre Rückmeldung ist uns wichtig, damit wir möglichst genaue und hochwertige Informationen bieten können. Wenn Sie Kommentare zu diesem Handbuch oder anderen Dokumentationen zu Edge Components von WebSphere Application Server abgeben möchten, gehen Sie wie folgt vor:

- Senden Sie Ihren Kommentar per E-Mail an fsdoc@us.ibm.com. Geben Sie dabei Folgendes an: Handbuchtitel, Teilenummer des Handbuchs, Version von WebSphere Application Server und wenn möglich die Position der Textstelle, auf die sich Ihr Kommentar bezieht (z. B. Seitenzahl oder Tabellennummer).

Referenzinformationen

- *Edge Components Konzepte, Planung und Installation*, IBM Form GC12-3420-00
- *Programming Guide for Edge Components*, IBM Form GC31-6856-00
- *Load Balancer Administratorhandbuch*, IBM Form GC12-3422-00
- *IBM WebSphere Edge Services Architecture*
- IBM Home-Website: *www.ibm.com/*
- Produkt-Website zu IBM WebSphere Application Server: *www.ibm.com/software/webservers/appserv/*
- Bibliotheks-Website zu IBM WebSphere Application Server: *www.ibm.com/software/webservers/appserv/library.html*
- Support-Website zu IBM WebSphere Application Server: *www.ibm.com/software/webservers/appserv/support.html*
- IBM WebSphere Application Server Information Center: *www.ibm.com/software/webservers/appserv/infocenter.html*
- IBM WebSphere Application Server Edge Components Information Center: *www.ibm.com/software/webservers/appserv/ecinfocenter.html*

Teil 1. Erste Schritte mit Caching Proxy

Dieser Teil enthält eine Übersicht über die Komponente Caching Proxy, Anweisungen zur Verwendung der Konfigurations- und Verwaltungsformulare und des Konfigurationsassistenten, Anweisungen zum manuellen Editieren der Datei ibmproxy.conf und Prozeduren zum Starten und Stoppen des Proxy-Servers.

Dieser Teil enthält die folgenden Kapitel:

Kapitel 1. Übersicht

Die Komponente Caching Proxy fängt Datenanforderungen von Clients ab, ruft die angeforderten Informationen von den Maschinen ab, die Inhalte bereitstellen, und liefert die abgerufenen Inhalte an die Clients. Meistens beziehen sich die Anforderungen auf Dokumente, die auf Webserver-Maschinen (auch Ursprungsserver oder Inhaltshosts genannt) gespeichert und mit Hypertext Transfer Protocol (HTTP) zugestellt werden. Sie können Caching Proxy jedoch auch für die Verwendung anderer Protokolle wie File Transfer Protocol (FTP) und Gopher konfigurieren.

Caching Proxy legt Inhalte, die zwischengespeichert werden können, vor der Zustellung an den Requester in einem lokalen Cache ab. Beispiele für solche im Cache speicherbaren Inhalte sind statische Webseiten und JSPs mit dynamisch generierten, aber sich nur selten ändernden Fragmenten. Durch die Zwischenspeicherung (oder Caching) ist Caching Proxy in der Lage, nachfolgende Anforderungen für denselben Inhalt direkt aus dem Cache bereitzustellen, was sehr viel schneller ist als ein erneuter Abruf des Inhalts vom Inhaltshost.

Anmerkung: Caching Proxy ist auf allen unterstützten Plattformen mit Ausnahme von 64-Bit-Systemen mit Itanium-2- oder AMD-Opteron-Prozessor verfügbar.

Neue Features

Das *Caching Proxy Administratorhandbuch* beschreibt neue Features, neu unterstützte Plattformen und Programmkorrekturen für Version 6.0 und Releases nach Version 5.0 (5.0.1, 5.0.2, 5.1 und 5.1.1). Die wichtigsten dieser neuen Features (für Version 6.0 und Releases nach Version 5.0) sind im Folgenden aufgelistet.

Anmerkungen:

1. Nähere Informationen zur unterstützten Hardware und den Softwareanforderungen finden Sie im Kapitel "Voraussetzungen für Edge Components" in der Veröffentlichung *Edge Components Konzepte, Planung und Installation*.
2. *Topaktuelle* Informationen zu den Systemvoraussetzungen für Edge Components Version 6 finden Sie auf der folgenden Webseite zu WebSphere Application Server:
   `http://www.ibm.com/software/webservers/appserv/doc/latest/prereq.html`

- **Unterstützung für Linux für S/390, zSeries, iSeries und pSeries**

  Der Proxy-Server wird jetzt nicht nur unter Linux für Intel, sondern auch unter Linux für S/390, zSeries, iSeries und pSeries unterstützt.
- **Unterstützung für AIX 5.2 und AIX 5.3**

  Caching Proxy unterstützt jetzt neben AIX 5.1 auch AIX 5.2 und AIX 5.3.
- **Unterstützung für Solaris 9**

  Caching Proxy unterstützt jetzt neben Solaris 8 auch Solaris 9.
- **Unterstützung für Windows Server 2003**

  Caching Proxy unterstützt jetzt neben Windows 2000 auch Windows Server 2003.
- **Unterstützung für HP-UX Version 11i**

  Caching Proxy unterstützt jetzt neben AIX-, Linux-, Solaris- und Windows-Systemen auch HP-UX.

- **Unterstützung für JDK 1.4.2**

  Es wird eine neue Version von JDK (32-Bit) unterstützt: JDK 1.4.2.
- **Unterstützung für GSKit 7**

  Mit Caching Proxy wird standardmäßig eine neue Version von GSKit bereitgestellt und installiert: GSKit 7.
- **Funktionale Erweiterungen und neue Anweisungen für JunctionRewrite**

  Es werden funktionale Erweiterungen und neue Anweisungen für JunctionRewrite bereitgestellt. Nähere Informationen hierzu finden Sie in den folgenden Abschnitten:
  - „Umschreiben von Junctions aktivieren (optional)" auf Seite 48
  - „JunctionReplaceUrlPrefix — Bei Verwendung des Plug-in JunctionRewrite den URL ersetzen, anstatt Präfix einzufügen" auf Seite 230
  - „JunctionSkipUrlPrefix — Bei Verwendung des Plug-in JunctionRewrite das Überschreiben von URLs überspringen, die das Präfix bereits enthalten" auf Seite 231
  - „JunctionRewriteSetCookiePath — Bei Verwendung des Plug-in JunctionRewrite die Pfadoption im Header Set-Cookie umschreiben" auf Seite 230
  - „RewriteSetCookieDomain — Umzuschreibendes Domänenmuster angeben" auf Seite 271
  - „Proxy - Proxy-Protokolle oder einen Reverse Proxy angeben" auf Seite 261

  Außerdem wird eine Alternative zur Verwendung des Plug-in JunctionRewrite beschrieben. Nähere Informationen hierzu finden Sie in den folgenden Abschnitten:
  - „UseCookie als Alternative zu JunctionRewrite" auf Seite 50
  - „JunctionRewrite - Umschreiben von URLs aktivieren" auf Seite 230
- **Zusätzliche neue Anweisungen**

  Die folgenden neuen Anweisungen werden unterstützt:
  - „CacheMatchLanguage — Sprachvorgabe für zurückgegebene Cache-Inhalte festlegen" auf Seite 194
  - „EnableTcpNodelay — Socket-Option TCP NODELAY aktivieren" auf Seite 213
  - Anweisungen für Einschränkung von Anforderungen:
    - „LimitRequestBody — Maximale Größe des Hauptteils für PUT- und POST-Anforderungen festlegen" auf Seite 232
    - „LimitRequestFields - Maximale Anzahl Header in Clientanforderungen festlegen" auf Seite 233
    - „LimitRequestFieldSize - Maximale Header-Länge und maximale Länge der Anforderungszeile festlegen" auf Seite 233
  - Anweisungen für das Speichern und Laden des Cache auf die Platte:
    - „ExportCacheImageTo - Cache-Speicher auf die Platte exportieren" auf Seite 218
    - „ImportCacheImageFrom - Cache-Speicher aus einer Datei importieren" auf Seite 229

- **Funktionale Erweiterungen für vorhandene Anweisungen**

  Es wurden funktionale Erweiterungen an den folgenden vorhandenen Anweisungen vorgenommen:

  – BindSpecific: Mit der Option `OutgoingSrcIp` kann der Proxy-Server eine bestimmte Quellen-IP-Adresse zum Herstellen abgehender Verbindungen verwenden.

  – ReversePass: Mit der Option `Host:Port` kann der Proxy-Server basierend auf dem Hostnamen und Port des Back-End-Servers verschiedene ReversePass-Regeln anwenden.

- **Beispiel-Plug-in Transmogrifier zur Erweiterung der Funktionalität von JunctionRewrite**

  Für JunctionRewrite wird jetzt anpassbarer Mustercode bereitgestellt, der JavaScript- (SCRIPT) und Applet-Tag-Blöcke (APPLET) in HTML-Dateien umschreibt und syntaktisch analysiert. Das Plug-in JunctionRewrite ist allein nicht in der Lage, die Ressourcenverknüpfungen in JavaScript oder in Parameterwerten von Java zu verarbeiten. Nähere Informationen hierzu finden Sie im Abschnitt „Beispiel-Plug-in Transmogrifier zur Erweiterung der Funktionalität von JunctionRewrite" auf Seite 51.

- **Änderungen an der Konfiguration des Dämons PACD**

  Informationen zum Aktivieren der Unterstützung für anonyme Bindungen finden Sie im Abschnitt „pac_ldap.cred erstellen" auf Seite 144.

  Wenn Sie zwischen einem Proxy-Server und einem LDAP-Server eine SSL-Verbindung herstellen möchten, müssen Sie das Kennwort für die Schlüsseldatenbank in der Datei pac_keyring.pwd speichern. Nähere Informationen hierzu finden Sie im Abschnitt „Eine neue Schlüsseldatenbank, ein neues Kennwort und eine neue Datei zur Kennwortspeicherung erstellen" auf Seite 123.

- **Änderungen an der Standardkonfiguration für mehr Sicherheit**

  In der Konfigurationsdatei (ibmproxy.conf) wurden Änderungen an den Standardeinstellungen vorgenommen, um mehr Sicherheit zu erreichen. Beispielsweise wurden Änderungen vorgenommen, um die Einstellung HTTP CONNECTION und SSL-Tunnelung inaktivieren zu können. Es sind keine neuen Anweisungen für diese Erweiterungen verfügbar.

Kapitel 2. Konfigurations- und Verwaltungsformulare verwenden

Caching Proxy stellt HTML-Formulare bereit, die an die anfragenden Clients gesendet und zur Konfiguration des Proxy-Servers verwendet werden können. Diese Formulare führen CGI-Programme aus, die die lokale Konfigurationsdatei des Proxy-Servers, ibmproxy.conf, editieren. Zur Verwendung dieser Formulare muss der Proxy-Server aktiv und so konfiguriert sein, dass er die Formulare aus dem lokalen Verzeichnis, in dem sie gespeichert sind, bereitstellt.

Anmerkung: Caching Proxy ist auf allen unterstützten Plattformen mit Ausnahme von 64-Bit-Systemen mit Itanium-2- oder AMD-Opteron-Prozessor verfügbar.

Standardmäßig wird Caching Proxy so installiert, dass die Datei ibmproxy.conf Pass-Anweisungen enthält, die den Zugriff auf die Konfigurations- und Verwaltungsformulare ermöglichen. Wenn ein Client die Standard-Homepage von diesem Proxy-Server anfordert, wird die Datei Frntpage.html zurückgegeben. Diese Seite enthält einen Hypertext-Link auf die Startseite der Konfigurations- und Verwaltungsformulare (wte.html).

Die Konfigurations- und Verwaltungsformulare sind geschützt und erfordern eine Clientauthentifizierung, bevor sie den Clients bereitgestellt werden. Anweisungen zur Konfiguration von ID und Kennwort des Administrators finden Sie im Abschnitt „Das Administratorkennwort festlegen” auf Seite 10.

Browser-Anforderungen

Ein Webbrowser, der für den Zugriff auf die Konfigurations- und Verwaltungsformulare verwendet wird, muss Folgendes unterstützen:

- *HTML 4.0*: Alle Formulare entsprechen der HTML-Spezifikation 4.0. Ihr Webbrowser muss HTML 4.0 und Framesets unterstützen.
- *Java 1.1 und JavaScript*: Die Applets entsprechen der Java-Spezifikation 1.1. Der Webbrowser muss eine Java Virtual Machine (JVM) unterstützen, die mit Java 1.1 kompatibel ist. Die Applets sind nicht für JVMs geeignet, die der Java-Spezifikation 2.0 entsprechen. JavaScript und Java müssen aktiviert sein.
- *256 Farben:* Die Workstation, auf der der Webbrowser ausgeführt wird, muss mindestens 256 Farben unterstützen.

Die **empfohlenen** Browser sind Mozilla 1.4 und Mozilla 1.7 für Linux- und UNIX-Systeme und Internet Explorer für Windows-Systeme. Nähere Informationen zur Verwendung von Browsern für die Anzeige der Konfigurations- und Verwaltungsformulare finden Sie in der Veröffentlichung *Edge Components Konzepte, Planung und Installation*.

Anmerkungen:

1. Auf Linux-Systemen des Typs 64-Bit-PowerPC ist es nicht möglich, mit dem Browser Mozilla auf die Konfigurations- und Verwaltungsformulare zuzugreifen, weil kein JDK für diese Architektur verfügbar ist. Sie können jedoch von einer anderen Maschine mit einem unterstützten Webbrowser auf die Konfigurations- und Verwaltungsformulare zugreifen.

2. Wenn Sie beim Starten der Administrationskonsole zwei Mal zur Anmeldung aufgefordert werden, ist Ihre Java-Einstellung im Internet Explorer nicht ordnungsgemäß definiert. Zum Korrigieren dieser Einstellung in Internet Explorer klicken Sie auf **Extras > Internetoptionen > Erweitert** und wählen das Markierungsfeld **Verwenden Sie Java 2 v1.4.X** ab.

Die Konfigurations- und Verwaltungsformulare aufrufen

Gehen Sie wie folgt vor, um die Konfigurations- und Verwaltungsformulare aufzurufen:

1. Vergewissern Sie sich, dass der Proxy-Server aktiv ist. Anweisungen zum Starten des Proxy-Servers finden Sie in Kapitel 5, „Caching Proxy starten und stoppen", auf Seite 15.
2. Rufen Sie in einem HTTP-Browser die Homepage des Proxy-Servers (Frntpage.html) oder die Startseite der Konfigurations- und Verwaltungsformulare (wte.html) auf.

   Anmerkung: Welche Seite angezeigt wird, ist von den verwendeten Zuordnungsregeln des Proxy-Servers abhängig und kann von den in Klammern angegebenen Standardseiten abweichen.

   `http://Name.Ihres.Servers[:Port][/Verzeichnis][/Seite.html]`

   Die Parameter sind im Folgenden erläutert:

   - *Name.Ihres.Servers* steht für den vollständigen Pfadnamen Ihres Hosts (z. B. `http://www.ibm.com/`).
   - [:*Port*] - Wenn der Server Verwaltungsanforderungen an einem anderen Port als Port 80 empfängt, fügen Sie diese Port-Nummer hinter dem Servernamen ein: `http://Name.Ihres.Servers:Port`
   - [*/Verzeichnis*] - Ob ein Verzeichnis im URL hinzugefügt werden muss, ist abhängig von den Zuordnungsregeln.
   - [*/Seite.html*] - Die HTML-Seite muss nur für den Fall angegeben werden, dass sie nicht als Begrüßungsseite aufgelistet ist. Informationen zu Begrüßungsseiten finden Sie im Abschnitt „Standardbegrüßungsseiten definieren" auf Seite 54.
3. Klicken Sie auf **Konfigurations- und Verwaltungsformulare**, um die Formulare für die Serverkonfiguration aufzurufen. Daraufhin werden Sie zur Eingabe des Benutzernamens und Kennworts des Administrators aufgefordert. Geben Sie einen berechtigten Benutzernamen und das zugehörige Kennwort ein. Das Clientfenster für die Konfiguration von Caching Proxy wird geöffnet.

   Anmerkungen:

   a. Nach Anzeige der Hauptseite kann es noch einige Sekunden dauern, bis der Inhalt in den Navigationsrahmen auf der linken Seite geladen wird.
   b. Auf Systemen mit Windows 2003 ist es möglich, dass Verbindungen abgebrochen werden, wenn Verwaltungsformulare (CGI-Scripts) angefordert werden. Als direkte Folge davon melden Browser, dass keine Daten empfangen wurden oder die Seite nicht verfügbar ist. Sie können dieses Problem vermeiden, indem Sie für MaxActiveThreads einen größeren Wert als 200 oder für ConnThreads einen größeren Wert als 50 angeben. Auf diese Weise wird das Problem mit den Verbindungsabbrüchen behoben. Nähere Informationen zu diesen Anweisungen finden Sie in den Abschnitten „MaxActiveThreads - Die maximale Anzahl aktiver Threads angeben" auf Seite 238 und „ConnThreads — Anzahl der Verbindungs-Threads für die Verbindungsverwaltung festlegen" auf Seite 203.

4. Im Navigationsrahmen auf der linken Seite werden die fünf Hauptkategorien der Konfigurationsformulare angezeigt:
   - **Konfiguration des Proxy-Servers**
   - **Cache-Konfiguration**
   - **Serverkonfiguration**
   - **Überwachung der Serveraktivität**
   - **Plug-in-Konfiguration**

   Wenn Sie auf das Dreieck links von einer Überschrift klicken, werden die Konfigurationsformulare für diese Kategorie angezeigt. Klicken Sie auf ein Formular, um es zu öffnen. In den Eingabefeldern des Formulars werden die aktuellen Konfigurationswerte (falls vorhanden) angezeigt. Falls Sie die Konfiguration seit der Installation nicht geändert haben, sind dies die Standardwerte.
5. Geben Sie in den Formularen die Konfigurationsdaten für diese bestimmte Funktion ein. Jedes Formular enthält Anweisungen, die Sie bei der Auswahl der richtigen Einstellungen unterstützen. Wenn Sie weiterführende Informationen benötigen, klicken Sie das Hilfesymbol, das Fragezeichen (**?**) oben in jedem Formular an. Es werden folgende Links angezeigt:
   - **Hilfe für Feld** - Beschreibungen der Felder in den einzelnen Anzeigen
   - **Vorgehensweise** - Detaillierte Schritte zur Verwendung des Formulars, um bestimmte Aufgaben auszuführen
   - **Index** - Ein Index mit Hilfeinformationen
6. Nachdem Sie ein Formular ausgefüllt haben, klicken Sie auf **Übergeben**, um die Serverkonfiguration mit den Änderungen, die Sie vorgenommen haben, zu aktualisieren. Jedes Formular enthält unterhalb der Eingabefelder die Schaltfläche **Übergeben**. Wenn Sie die Änderungen im Formular nicht anwenden möchten, klicken Sie auf **Zurücksetzen**, um die Felder im Formular auf ihre ursprünglichen Werte zurückzusetzen.
7. Wenn Sie auf **Übergeben** klicken und die Eingabe akzeptiert wird, erscheint im oberen Rahmen folgende Nachricht:

   ```
   Die gewünschten Konfigurationsänderungen wurden erfolgreich durchgeführt.
   ```

   Wird die Eingabe nicht akzeptiert, erscheint im oberen Rahmen eine Fehlernachricht, die die ungültigen Einstellungen enthält.
8. Klicken Sie im oberen Rahmen auf das Symbol für Serverneustart (|), um den Proxy-Server erneut zu starten. Wenn der Proxy-Server den Befehl zum Neustart empfängt, akzeptiert er keine weiteren Anforderungen von Clients, beantwortet jedoch alle Anforderungen, deren Verarbeitung bereits gestartet wurde. Nach dem erneuten Laden der geänderten Konfigurationsdatei akzeptiert der Proxy-Server wieder Clientanforderungen.

   Anmerkung: Wenn Sie bestimmte Anweisungen in den Konfigurations- und Verwaltungsformularen oder in der Datei ibmproxy.conf ändern, reicht ein Neustart des Server nicht aus. Sie müssen den Server vollständig stoppen und anschließend erneut starten, damit die Änderungen wirksam werden. Die betroffenen Anweisungen können Sie der Tabelle 6 auf Seite 175 entnehmen.

Das Administratorkennwort festlegen

Nach der Installation der Caching-Proxy-Pakete müssen Sie eine Administrator-ID und ein Administratorkennwort für den Zugriff auf die Konfigurations- und Verwaltungsformulare erstellen. In der Standardkonfiguration des Proxy-Servers werden Benutzer, die die Konfigurations- und Verwaltungsformulare anfordern, anhand der Kennwortdatei webadmin.passwd im Verzeichnis /opt/ibm/edge/cp/server_root/protect/ (Linux- und UNIX-Systeme) bzw. im Verzeichnis \Programme\IBM\edge\cp\etc\ (Windows-Systeme) authentifiziert. Wenn die Datei webadmin.passwd vorhanden ist, wird diese bei der Installation eines Pakets nicht überschrieben.

Mit den folgenden Befehlen können Sie der Datei webadmin.passwd einen Administratoreintrag hinzufügen:

- Linux- und UNIX-Systeme:

  ```
  # htadm -adduser /opt/ibm/edge/cp/server_root/protect/webadmin.passwd
  ```

  Geben Sie auf Anforderung den Benutzernamen, das Kennwort und den realen Namen des Administrators für das Programm **htadm** ein.

- Auf Windows-Systemen:

  ```
  cd "\Programme\IBM\edge\cp\server_root\protect\"
  htadm -adduser webadmin.passwd"
  ```

  Geben Sie auf Anforderung den Benutzernamen, das Kennwort und den realen Namen des Administrators für das Programm **htadm** ein.

Anmerkung: Beim Benutzernamen und Kennwort des Administrators wird zwischen Groß- und Kleinschreibung auch dann unterschieden, wenn das Betriebssystem diese Unterscheidung nicht macht. Geben Sie Benutzernamen und Kennwort mit dem Befehl `htadm` exakt ein, wenn Sie auf die Konfigurations- und Verwaltungsformulare zugreifen.

Eine detaillierte Beschreibung des Befehls **htadm** finden Sie im Abschnitt „Befehl htadm" auf Seite 167.

Kapitel 3. Den Konfigurationsassistenten verwenden

Mit dem Konfigurationsassistenten von Caching Proxy können Sie einen installierten Caching-Proxy-Server schnell konfigurieren. Dieses Programm legt nur die grundlegenden Anweisungen fest, die erforderlich sind, um das Verhalten von Caching Proxy in der Weise zu ändern, dass dieser als Ersatzserver verwendet wird. Möglicherweise müssen für den Proxy-Server noch weitere Konfigurationseinstellungen vorgenommen werden.

Gehen Sie wie folgt vor, um den Konfigurationsassistenten von Caching Proxy zu verwenden:

1. Starten Sie den Konfigurationsassistenten.

   Klicken Sie unter Windows auf **Start –> Programme –> IBM WebSphere –> Edge Components –> Caching Proxy –> Konfigurationsassistent**.

   Geben Sie auf Linux- und UNIX-Systemen den Befehl `/opt/ibm/edge/cp/cpwizard/cpwizard.sh` ein.
2. Wählen Sie den Netz-Port aus, an dem der Proxy-Server HTTP-Anforderungen empfangen soll.
3. Geben Sie den Namen des Zielinhaltsservers ein.
4. Geben Sie die Benutzer-ID und das Kennwort für den Administrator des Proxy-Servers ein.

Anmerkungen:

1. Der Konfigurationsassistent definiert die folgenden Anweisungen:

   ```
   Port Port-Nummer
   Proxy /* http://Inhaltsserver :Port
   ```
2. Wenn Sie den Proxy-Server mit dem Konfigurationsassistenten konfigurieren, müssen Sie zum Aktivieren von SSL eine Zuordnungsregel erstellen, damit die an Port 443 empfangenen Anforderungen weitergeleitet werden. Nähere Informationen hierzu finden Sie im Abschnitt „Zuordnungsregeln definieren” auf Seite 45.

   Beispiele:

   ```
   Proxy /* http://Inhaltsserver :443
   ```

   oder

   ```
   Proxy /* https://Inhaltsserver :443
   ```

Einschränkungen: Auf Linux-Systemen funktionieren die Tastaturkurzbefehle für den Konfigurationsassistenten von Caching Proxy nicht.

Kapitel 4. Die Datei ibmproxy.conf manuell editieren

Caching Proxy kann durch Editieren der Konfigurationsdatei ibmproxy oder mit den Konfigurations- und Verwaltungsformularen manuell konfiguriert werden.

- Auf Linux- und UNIX-Systemen ist die Datei ibmproxy.conf im Verzeichnis /etc/ gespeichert.
- Auf Windows-Systemen finden Sie die Datei ibmproxy.conf im Verzeichnis C:\Programme\IBM\edge\cp\etc\de_DE\.

Die Konfigurationsdatei setzt sich aus Anweisungen zusammen. Zum Ändern der Konfiguration müssen Sie die Konfigurationsdatei in einem Editor öffnen, die Anweisungen ändern und Ihre Änderungen anschließend speichern. Sie können beinahe jeden Texteditor, z. B. emacs und vi, zum Editieren der Konfigurationsdatei verwenden.

Anmerkung: Der Textdateieditor, der mit Solaris Common Desktop Environment (CDE) bereitgestellt wird, sollte nicht verwendet werden. Der Solaris-Editor ändert manchmal die Eignergruppe der Datei und die Merkmale der Dateiverbindung, so dass die Konfigurations- und Verwaltungsformulare nicht in die Konfigurationsdatei schreiben können.

Die in der Konfigurationsdatei vorgenommenen Änderungen werden nach dem Neustart des Servers wirksam, sofern Sie keine der Anweisungen geändert haben, die in der Tabelle 6 auf Seite 175 aufgeführt sind. Wenn Sie eine der Anweisungen aus dieser Liste geändert haben, müssen Sie den Server stoppen und anschließend erneut starten. Diesbezügliche Anweisungen finden Sie in Kapitel 5, „Caching Proxy starten und stoppen", auf Seite 15.

Anhang B, „Anweisungen in der Konfigurationsdatei", auf Seite 175 beschreibt jede der Anweisungen in der Konfigurationsdatei und enthält detaillierte Angaben zu deren Syntax.

Kapitel 5. Caching Proxy starten und stoppen

Die Komponente Caching Proxy ist so konzipiert, dass sie fortlaufend als Hintergrundprozess ausgeführt wird und nur sehr wenige Bedienereingriffe erfordert. In der Regel wird der Proxy-Server während des Boot-Zyklus der Maschine gestartet und nur gestoppt, wenn Wartungsmaßnahmen durchgeführt werden müssen. Der Proxy-Server kann bei Bedarf manuell gestartet werden. Außerdem kann dem Proxy-Server eine Neustartanweisung übergeben werden, die den Proxy-Server stoppt und dann erneut startet, ohne aktive Clientverbindungen zu unterbrechen.

Anmerkung: Caching Proxy ist auf allen unterstützten Plattformen mit Ausnahme von 64-Bit-Systemen mit Itanium-2- oder AMD-Opteron-Prozessor verfügbar.

Automatisches Starten und Stoppen auf Linux- und UNIX-Systemen

Auf Linux- und UNIX-Systemen werden bei der Installation von Caching Proxy ein Initialisierungs-Script mit dem Namen **ibmproxy** und die zugehörigen symbolischen Verbindungen in den entsprechenden /etc/-Verzeichnissen gespeichert. Die Scripts werden dann in die Start- und Systemabschlussroutinen des Betriebssystems integriert. Sie können die Konfigurationseinstellungen für automatischen Neustart ändern, indem Sie das Script **ibmproxy** editieren und die Optionen des Befehls **ibmproxy** ändern.

Anmerkung: Grenzwert für Dateideskriptoren unter Solaris

Es ist möglich, dass das Initialisierungs-Script von Caching Proxy wegen des systemweiten Grenzwerts für Dateideskriptoren unter Solaris nicht die gewünschte maximale Anzahl von Dateideskriptoren festlegen kann. Ist der systemweite Maximalwert niedriger als die Einstellung im Initialisierungs-Script von Caching Proxy, wird der systemweite Grenzwert verwendet. Um Probleme mit der Proxy-Leistung zu vermeiden, die aus einem zu niedrigen Wert (unter 1024) resultieren, können Sie den Grenzwert für die Dateideskriptoren ändern. Führen Sie den Befehl **ulimit** aus, um die Anzahl der Deskriptoren anzuzeigen, die derzeit verfügbar sind. Liegt der Wert unter 1024, erhöhen Sie den Grenzwert. Um den Grenzwert für Dateideskriptoren auf 1024 zu erhöhen, fügen Sie der Datei /etc/system die folgende Zeile hinzu:

```
set rlim_fd_cur=0x400
```

Automatisches Starten und Stoppen inaktivieren

Gehen Sie wie folgt vor, um das automatische Starten und Stoppen zu inaktivieren:

- Entfernen Sie auf AIX-Systemen den Befehl **ibmproxy** aus der Initialisierungsdatei.
- Entfernen Sie auf HP-UX-Systemen die folgenden Verbindungen zur Datei **ibmproxy**:
  - /sbin/rc1.d/K154ibmproxy
  - /sbin/rc2.d/S880ibmproxy

- Entfernen Sie auf Linux-Systemen die symbolischen Verbindungen zum Script **/etc/rc.d/init.d/ibmproxy** in den Unterverzeichnissen der Ausführungsebene.

  Entfernen Sie unter SuSE Linux die folgenden Verbindungen zur Datei **ibmproxy**:
  - /etc/rc.d/rc3.d/S20ibmproxy
  - /etc/rc.d/rc3.d/K20ibmproxy
  - /etc/rc.d/rc4.d/S20ibmproxy
  - /etc/rc.d/rc4.d/K20ibmproxy
  - /etc/rc.d/rc5.d/S20ibmproxy
  - /etc/rc.d/rc5.d/K20ibmproxy

  Entfernen Sie unter Red Hat Linux die folgenden Verbindungen zur Datei **ibmproxy**:
  - /etc/rc.d/rc0.d/K54ibmproxy
  - /etc/rc.d/rc1.d/K54ibmproxy
  - /etc/rc.d/rc2.d/K54ibmproxy
  - /etc/rc.d/rc6.d/K54ibmproxy
  - /etc/rc.d/rc3.d/S88ibmproxy
  - /etc/rc.d/rc5.d/S88ibmproxy
- Entfernen Sie auf Solaris-Systemen den Befehl **ibmproxy start** und die beiden zugehörigen Beendigungs-Scripts wie folgt:
  - Löschen Sie S88ibmproxy aus dem Verzeichnis /etc/rc2.d.
  - Löschen Sie K54ibmproxy aus dem Verzeichnis /etc/rc0.d.
  - Löschen Sie K54ibmproxy aus dem Verzeichnis /etc/rc1.d.

Manuelles Starten auf Linux- und UNIX-Systemen

Der Befehl **ibmproxy** wird unabhängig von der Startmethode entweder direkt von der Eingabeaufforderung oder aus einem Script heraus aufgerufen. Eine detaillierte Beschreibung des Befehls **ibmproxy** finden Sie im Abschnitt „Befehl ibmproxy" auf Seite 172. Beispiele für die am häufigsten verwendeten Argumente folgen.

Unter AIX:

- Wenn Sie den Proxy-Server für die Standard-Locale mit dem Befehl **startsrc** starten möchten, geben Sie Folgendes ein:

  ```
  startsrc -s ibmproxy
  ```

- Wenn Sie den Proxy-Server für eine andere Locale mit dem Befehl **startsrc** starten möchten, geben Sie Folgendes ein:

  ```
  startsrc -s ibmproxy -e "LC_ALL=Locale"
  ```

- Wenn Sie den Proxy-Server mit den Standardlaufzeiteinstellungen ohne den Befehl **startsrc** starten möchten, geben Sie Folgendes ein:

  ```
  ibmproxy
  ```

Unter HP-UX:

- Wenn Sie den Proxy-Server mit dem Initialisierungs-Script starten möchten, geben Sie im Stammverzeichnis an einer Eingabeaufforderung Folgendes ein:

  ```
  /sbin/init.d/ibmproxy start
  ```

- Wenn Sie den Proxy-Server als Hintergrundprozess ohne das Initialisierungs-Script starten möchten, geben Sie im Stammverzeichnis an einer Eingabeaufforderung Folgendes ein:

  ```
  /usr/sbin/ibmproxy
  ```

- Wenn Sie den Proxy-Server als Vordergrundprozess ohne das Initialisierungs-Script starten möchten, geben Sie im Stammverzeichnis an einer Eingabeaufforderung Folgendes ein:

  ```
  /usr/sbin/ibmproxy -nobg
  ```

Unter Linux:

- Wenn Sie den Proxy-Server mit dem Initialisierungs-Script starten möchten, geben Sie im Stammverzeichnis an einer Eingabeaufforderung Folgendes ein:

  ```
  /etc/rc.d/init.d/ibmproxy start
  ```

- Wenn Sie den Proxy-Server als Hintergrundprozess ohne das Initialisierungs-Script starten möchten, geben Sie im Stammverzeichnis an einer Eingabeaufforderung Folgendes ein:

  ```
  /usr/sbin/ibmproxy
  ```

- Wenn Sie den Proxy-Server als Vordergrundprozess ohne das Initialisierungs-Script starten möchten, geben Sie im Stammverzeichnis an einer Eingabeaufforderung Folgendes ein:

  ```
  /usr/sbin/ibmproxy -nobg
  ```

- Wenn Sie den Proxy-Server mit einer bereits vorhandenen SQUID-Konfigurationsdatei, squidConfig.file, starten möchten, geben Sie im Stammverzeichnis an einer Eingabeaufforderung Folgendes ein:

  ```
  squidConfig.file -r /etc/errors_icons.conf
  ```

  Dabei legt die Datei *errors_icons.conf* die Symbole fest, die beim Durchsuchen von Verzeichnissen für die zugeordneten Dateitypen verwendet werden sollen.

Unter Solaris:

- Wenn Sie den Proxy-Server mit dem Initialisierungs-Script starten möchten, geben Sie im Stammverzeichnis an einer Eingabeaufforderung Folgendes ein:

  ```
  /etc/init.d/ibmproxy start
  ```

- Wenn Sie den Proxy-Server als Hintergrundprozess ohne das Initialisierungs-Script starten möchten, geben Sie im Stammverzeichnis an einer Eingabeaufforderung Folgendes ein:

  ```
  /usr/sbin/ibmproxy
  ```

- Wenn Sie den Proxy-Server als Vordergrundprozess ohne das Initialisierungs-Script starten möchten, geben Sie im Stammverzeichnis an einer Eingabeaufforderung Folgendes ein:

  ```
  /usr/sbin/ibmproxy -nobg
  ```

Als Windows-Dienst starten

Wird Caching Proxy als Windows-Dienst installiert, wird er wie jeder andere Windows-Dienst gestartet:

1. Klicken Sie auf **Start –> Einstellungen (für Windows 2000) –> Systemsteuerung**.
2. Klicken Sie in der **Systemsteuerung** doppelt auf **Verwaltung –> Dienste**.
3. Heben Sie im Fenster **Dienste** den Eintrag **Caching Proxy** hervor.
4. Klicken Sie auf **Starten**, um Caching Proxy als Dienst zu starten.

Ist Caching Proxy als Dienst installiert, kann er so konfiguriert werden, dass er automatisch beim Start von Windows gestartet wird. In diesem Fall müssen Sie sich nicht anmelden, damit der Proxy-Server Anforderungen beantworten kann. Führen Sie folgende Schritte aus, damit der Proxy-Server automatisch gestartet wird:

1. Klicken Sie auf **Start –> Einstellungen (für Windows 2000) –> Systemsteuerung**.
2. Klicken Sie in der **Systemsteuerung** doppelt auf **Verwaltung –> Dienste**.
3. Heben Sie im Fenster **Dienste** den Eintrag **Caching Proxy** hervor.
4. Klicken Sie auf den Radioknopf **Automatisch** und anschließend auf **Starten**, damit der Caching-Proxy-Dienst beim Start von Windows automatisch gestartet wird.

Umgebungsvariable PATH aktualisieren

Wird für Caching Proxy in der Anzeige **Dienste** der Status `Gestartet` angezeigt, obwohl der Proxy-Server nicht aktiv ist, wurde die Maschine eventuell nach der Installation des Proxy-Servers nicht neu gestartet. Wurde die Komponente Caching Proxy so definiert, dass sie interaktiv mit dem Desktop arbeitet, und wurde sie danach nicht neu gestartet, wird eventuell folgende Fehlernachricht in einem Dialogfenster angezeigt: `Message catalog error: the message catalog could not be loaded or is invalid` (Nachrichtenkatalogfehler: Der Nachrichtenkatalog konnte nicht geladen werden oder ist ungültig).

Die Maschine muss jedoch neu gestartet werden, damit der Wert der Umgebungsvariablen PATH in der Windows-Registry aktualisiert wird. Wird die Registry nicht aktualisiert, zeigt die Variable PATH möglicherweise die richtigen Pfade für Caching Proxy und GSK7 an, funktioniert aber nicht ordnungsgemäß.

Anmerkung: Auf Windows-Systemen kann ein Konflikt auftreten, wenn Caching Proxy und eine andere Anwendung, z. B. ein Netzdateisystem, als Dienst ausgeführt werden. Pfade, die ein fernes Laufwerk enthalten, deren Eigner eine ebenfalls als Dienst ausgeführte Dateisystemanwendung ist, können von Caching Proxy manchmal nicht verwendet werden.

Dieses Problem kann auftreten, wenn in der Windows-Umgebungsvariablen PATH der Pfad für den Dateisystemdienst vor dem Pfad für den Caching-Proxy-Dienst angegeben ist. Sie können das Problem lösen, indem Sie Dateisystemdienste in der Variablen PATH möglichst am Ende angeben.

Dieses Problem hat keine Auswirkung auf ferne Laufwerke, die von Anwendungen bereitgestellt werden, die nicht als Windows-Dienste ausgeführt werden. Beispielsweise kann Caching Proxy auf freigegebene Laufwerke auf anderen Windows-Maschinen zugreifen, die in einem lokalen Netz (LAN) sichtbar sind.

Als Windows-Anwendung starten

Das Menü 'Start' verwenden

Bei der Installation von Caching Proxy als Windows-Anwendung wird im Menü **Start** ein Untermenü **Caching Proxy** erstellt. Zum Starten von Caching Proxy als Anwendung klicken Sie auf **Start –> Programme –> IBM WebSphere –> Edge Components –> Caching Proxy**.

Diese Startprozedur führt den Proxy-Server mit den aktuellen Konfigurationseinstellungen aus. Wenn Sie beim Start andere Einstellungen festlegen möchten, müssen Sie den Server von der Eingabeaufforderung aus starten. Nähere Informationen hierzu finden Sie im folgenden Abschnitt.

Eingabeaufforderung verwenden

Wenn Sie den Server von einer Windows- oder DOS-Eingabeaufforderung aus starten möchten, verwenden Sie den Befehl **ibmproxy**. Sollten Sie Windows seit der Installation des Servers noch nicht heruntergefahren und erneut gestartet haben, müssen Sie (standardmäßig) mit dem Befehl den vollständigen Pfadnamen angeben. Beispiel:

```
c:\Programme\IBM\edge\cp\bin\ibmproxy.exe
```

Der Befehl **ibmproxy** startet den Server mit den aktuellen Konfigurationseinstellungen. Wenn Sie die Serverkonfiguration seit der Installation nicht geändert haben, basiert die aktuelle Konfiguration auf den Informationen, die Sie während der Installation angegeben haben, und auf den Standardoptionen.

Mit dem Befehl **ibmproxy** wird der Server als Anwendung gestartet, selbst wenn Sie Caching Proxy als Dienst installiert haben. Damit der Server als Anwendung ausgeführt wird, können Sie auch die Befehlsoption **-noservice** angeben. Mit anderen Befehlsoptionen können Sie die Konfigurationseinstellungen während der Laufzeit ändern.

Mehrere Proxy-Server starten

Sie können mehrere Instanzen des Proxy-Servers parallel ausführen, aber jede Instanz muss an einem gesonderten Port empfangsbereit sein. Auf AIX-Systemen kann mit SRC nur eine Instanz gestartet werden. Für alle Instanzen des Servers müssen eindeutige Konfigurationsdateien angegeben werden, weil in der Konfigurationsdatei eine Port-Nummer festgelegt ist und diese Nummer für jeden Server auf einer bestimmten Maschine eindeutig sein muss. Zum Starten einer weiteren Instanz des Servers (wenn mindestens eine Serverinstanz aktiv ist) geben Sie an der Eingabeaufforderung folgenden Befehl ein:

- Unter Linux und UNIX:

  ```
  ibmproxy -r andere_Konfigurationsdatei
  ```

- Unter Windows:

  ```
  ibmproxy -noservice -r andere_Konfigurationsdatei
  ```

andere_Konfigurationsdatei steht für eine eindeutige Konfigurationsdatei.

Beim Starten mehrerer Instanzen des Servers müssen Sie die Prozess-ID notieren, die für jede Instanz angezeigt wird. Sie benötigen diese IDs, damit Sie bestimmte Instanzen des Servers stoppen können.

Anmerkung: Auf Linux-Systemen, auf denen mehrere Instanzen des Servers ausgeführt werden, können Sie mit dem folgenden Befehl den zuletzt gestarteten Server stoppen: **/etc/rc.d/init.d/ibmproxy stop**. Die übrigen Instanzen müssen manuell gestoppt werden. Nähere Informationen hierzu finden Sie im Abschnitt „Manuelles Stoppen auf Linux- und UNIX-Systemen" auf Seite 21.

Manuelles Stoppen auf Linux- und UNIX-Systemen

Gehen Sie wie folgt vor, um den Server zu stoppen:

- Sie müssen entweder der Benutzer sein, der den Prozess gestartet hat, oder der Superuser root.
- Sie müssen zum Stoppen dieselbe Methode verwenden, mit der der Server gestartet wurde. Die folgende Tabelle enthält die Startmethoden und die zugehörigen Stoppmethoden.

Tabelle 2. Start- und Stoppmethoden für Linux- und UNIX-Systeme

Startmethode	**Stoppmethode**
In /etc/inittab (unter AIX)	Geben Sie Folgendes ein: `stopsrc -s ibmproxy`
In /sbin/init.d (unter HP-UX)	Geben Sie Folgendes ein: `/sbin/init.d/ibmproxy stop`
In /etc/rc.d/init.d (unter Linux)	Geben Sie Folgendes ein: `/etc/rc.d/init.d/ibmproxy stop`
`ibmproxy`	1. Ermitteln Sie die Prozess-ID von **ibmproxy**. Geben Sie unter AIX den Befehl `ps -aef | grep "ibmproxy"` ein, unter Linux den Befehl `ps -aux | grep ibmproxy | grep` *`Server-ID`*, und unter Solaris und HP-UX den Befehl `ps -ef | grep "ibmproxy"`. 2. Stoppen Sie den **ibmproxy**-Prozess, indem Sie Folgendes eingeben: `kill` *`Prozess-ID`* Zum Stoppen aller Server auf dieser Maschine geben Sie Folgendes ein: `killall ibmproxy`
`ibmproxy -nobg`	Drücken Sie die Tastenkombination `Strg-c`.
`ibmproxy -r -`*`andere_Konfigurationsdatei`* (unter AIX)	Geben Sie Folgendes ein: `stopsrc -s ibmproxy -p` *`Prozess-ID`*
`ibmproxy -r -`*`andere_Konfigurationsdatei`* (unter Linux)	1. Ermitteln Sie die Prozess-ID von **ibmproxy**. Geben Sie Folgendes ein: `ps aux | grep ibmproxy | grep` *`Prozess-ID`* 2. Stoppen Sie den **ibmproxy**-Prozess, indem Sie Folgendes eingeben: `kill` *`Prozess-ID`*

Zum Stoppen des Servers melden Sie sich als root an und geben dann an der Eingabeaufforderung Folgendes ein:

- Unter AIX: `stopsrc -s ibmproxy`
- Unter HP-UX: `/sbin/init.d/ibmproxy stop`
- Unter Linux: `/etc/rc.d/init.d/ibmproxy stop`
- Unter Solaris: `/etc/init.d/ibmproxy stop`

Einschränkungen für die Stoppbefehle

Bei Verwendung der Stoppbefehle stoßen Sie möglicherweise auf die folgenden Einschränkungen:

- **AIX, HP-UX und Linux**

  Auf AIX-, HP-UX- und Linux-Systemen beenden die Befehle zum Stoppen des Caching-Proxy-Systems manchmal nur den Caching-Proxy-Prozess. Dieses Verhalten ist beim AIX-Befehl **stopsrc -s ibmproxy** zu beobachten. Dasselbe gilt für den Befehl **ibmproxy -stop** unter HP-UX und Linux.

  Der vom LDAP-Server verwendete PACD-Prozess bleibt möglicherweise auch nach dem Stoppen des Proxy-Servers aktiv. Der PACD-Prozess kann mit dem Befehl **kill** wie folgt zuverlässig gestoppt werden:

  ```
  kill -15 PACD-Prozess-ID
  ```

- **Solaris**

  Wird der Befehl **ibmproxy -stop** auf einem Solaris-System ausgeführt, hat dies nicht dasselbe Ergebnis wie die Ausführung dieses Befehls auf anderen Betriebssystemen. Aufgrund einer Einschränkung im Solaris-Code wird der Schritt zur Serverbeendigung nicht ausgeführt, wenn der Befehl **ibmproxy -stop** auf einer Solaris-Plattform ausgeführt wird.

  Diese Einschränkung hat sowohl Auswirkungen auf die Proxy-Server-Software als auch auf die Plug-ins, die vom Kunden implementiert werden.

  Es ist möglich, dass der vom LDAP-Server verwendete PACD-Prozess auch nach dem Stoppen des Proxy-Servers aktiv bleibt. Der PACD-Prozess kann mit dem Befehl **kill** wie folgt zuverlässig gestoppt werden:

  ```
  kill -15 PACD-Prozess-ID
  ```

Manuelles Stoppen auf einem Windows-System

Sie können den Caching-Proxy-Server auf dieselbe Art und Weise stoppen wie andere Windows-Programme.

Ist der Proxy-Server als Dienst installiert, gehen Sie wie folgt vor:

1. Klicken Sie auf **Start –> Einstellungen (für Windows 2000) –> Systemsteuerung**.
2. Klicken Sie in der **Systemsteuerung** doppelt auf **Verwaltung –> Dienste**.
3. Heben Sie im Fenster **Dienste** den Eintrag **Caching Proxy** hervor.
4. Klicken Sie auf **Stoppen**, um den Caching-Proxy-Dienst zu stoppen.

Ist der Proxy-Server nicht als Dienst installiert, führen Sie einen der folgenden Schritte aus, um Caching Proxy zu stoppen:

- Klicken Sie in der rechten oberen Ecke auf das Symbol **x**.
- Wählen Sie im Menü **Datei** den Menüpunkt **Beenden** aus.
- Drücken Sie die Tasten **Alt + F4**.

Neustart nach Konfigurationsänderungen

Nachdem die Serverkonfiguration (mit den Konfigurations- und Verwaltungsformularen oder durch Editieren der Datei ibmproxy.conf) geändert wurde, müssen Sie den Server erneut starten, damit die Änderungen wirksam werden. In den meisten Fällen können Sie den Server erneut starten, ohne ihn zuerst zu stoppen. Einige Einstellungen werden jedoch durch einen einfachen Neustart nicht aktualisiert. Nähere Informationen hierzu finden Sie in Tabelle 6 auf Seite 175.

Um für den Server einen Neustart durchzuführen, ohne ihn vorher zu stoppen, klicken Sie in einem Konfigurations- und Verwaltungsformular auf **Neustart** oder geben Sie folgenden Befehl ein: `ibmproxy -restart`

Teil 2. Caching-Proxy-Prozess konfigurieren und verwalten

In diesem Teil wird die Zusammenarbeit zwischen der Komponente Caching Proxy und dem Betriebssystem, der Computerhardware und dem Netzwerk beschrieben. Außerdem beschreibt dieser Teil die Prozeduren zum Konfigurieren dieser Zusammenarbeit. Diese Elemente der Proxy-Server-Konfiguration werden in der Regel vom Systemadministrator verwaltet und müssen sorgfältig auf die Netzressourcen wie IP-Adressen und Hostnamen sowie auf die Systemressourcen wie den verfügbaren Speicher und die CPU-Zyklen abgestimmt werden.

Dieser Teil enthält die folgenden Kapitel:

Kapitel 6. Den Server definieren

Caching Proxy wird in der Regel als Hintergrundprozess auf einem Hostcomputersystem ausgeführt, das als Netzserver konfiguriert ist. Dieser Prozess wird einer oder allen aktiven IP-Adressen (Internet Protocol) auf dem Hostcomputersystem zugeordnet (*an diese gebunden*). Caching Proxy wartet an angegebenen Ports auf verschiedene Internet-Protokolle wie FTP und HTTP und führt gemäß der Verhaltenskonfiguration bestimmte Aktionen für diese Anforderungen aus. (Nähere Informationen hierzu finden Sie in Teil 3, „Das Verhalten von Caching Proxy konfigurieren", auf Seite 41.)

Anmerkung: Caching Proxy ist auf allen unterstützten Plattformen mit Ausnahme von 64-Bit-Systemen mit Itanium-2- oder AMD-Opteron-Prozessor verfügbar.

Standardmäßig setzt Caching Proxy den Namen des Hostcomputersystems voraus. Sie können dieses Standardverhalten jedoch außer Kraft setzen, indem Sie explizit einen Hostnamen für den Proxy-Server angeben. Wenn Sie Caching Proxy an eine bestimmte IP-Adresse binden möchten, müssen Sie den dieser IP-Adresse entsprechenden Hostnamen für den Proxy-Server wählen.

Anmerkung: Wenn der Proxy-Server versucht, eine Bindung zu einer IP-Adresse herzustellen, und der Hostname keiner verfügbaren IP-Adresse entspricht, schlägt der Bindungsversuch fehl. In diesem Fall ist der Proxy-Server an allen verfügbaren IP-Adressen empfangsbereit.

Der Hostname des Proxy-Servers hat keinen Einfluss darauf, wie der Clientdatenverkehr aufgelöst wird. Der Proxy-Server vergleicht seinen Hostnamen nicht mit dem Wert des Arguments Hostname im Header der HTTP-Anforderung. Der Hostname des Proxy-Servers wird gelegentlich in dynamisch generierte lokale Inhaltsseiten wie Fehlernachrichten eingebunden. Er wird außerdem als Wert des Arguments "Via" im HTTP-Header an den Anforderungsclient zurückgegeben.

Der Proxy-Server kann so konfiguriert werden, dass der Hostname des Anforderungsclients durch den Hostnamen des Proxy-Servers ersetzt wird, bevor die Anforderung an den Zielserver übergeben wird. Auf diese Weise wird der Zielserver gezwungen, den Übertragungskanal über den Proxy-Server aufrecht zu erhalten, anstatt eine Direktverbindung zum Client aufzubauen.

Geben Sie zur Definition des Proxy-Server-Prozesses mit den Anweisungen ServerRoot, Hostname und Port die physischen Adressen der Proxy-Server-Dateien auf dem Hostcomputersystem, den Namen, den der Proxy-Server verwendet, um auf sich selbst zu verweisen, und die Ports an, an denen der Proxy-Server empfangsbereit ist. Falls der Host mehrere IP-Adressen besitzt, kann der Proxy-Server an eine bestimmte Adresse gebunden werden. Setzen Sie hierfür die Anweisung BindSpecific auf "On" und die Anweisung "Hostname" auf die IP-Adresse.

Ein Verwaltungs-Port ist eine Methode für den Zugriff auf die Konfigurations- und Verwaltungsformulare und die Verwaltung des Servers. Wenn Sie den Zugriff auf den Proxy-Server über einen Verwaltungs-Port zulassen möchten, geben Sie mit der Anweisung AdminPort einen Wert an. Am Verwaltungs-Port empfangene Anforderungen werden nicht in dieselbe Warteschlange eingereiht wie Anforderungen, die am Standard-Port empfangen werden. Sie können Zuordnungsregeln schreiben, um den Zugriff auf die Konfigurations- und Verwaltungsformulare über diesen Port zuzulassen.

Wenn die Anweisung BindSpecific aktiviert ist, wird Caching Proxy mit der IP-Adresse, die aus dem Wert der Anweisung Hostname abgeleitet wird, an den mit der Anweisung Port angegebenen Port gebunden. Der mit der Anweisung AdminPort angegebene Port wird an alle auf dem System verfügbaren IP-Adressen gebunden.

Wenn Sie den Standardnamen des aktiven Servers, z. B. IBM-PROXY oder IBM_HTTP_SERVER, ändern möchten, geben Sie einen Wert für die Anweisung HeaderServerName an. Dieser Wert wird in das Serverfeld der HTTP-Antwort übernommen.

Zur Steigerung der Proxy-Leistung können Sie die Anweisung PureProxy auf "On" setzen. Mit dieser Einstellung werden alle Cache-Funktionen inaktiviert.

Zugehörige Anweisungen

Der Proxy-Server-Prozess kann mit den folgenden Anweisungen definiert werden:

- „Hostname - Den vollständig qualifizierten Domänennamen oder die IP-Adresse für den Server angeben" auf Seite 224
- „ServerRoot - Das Verzeichnis angeben, in dem das Serverprogramm installiert ist" auf Seite 276
- „HeaderServerName - Namen des Proxy-Servers angeben, der im HTTP-Header zurückgegeben wird" auf Seite 224
- „BindSpecific - Angeben, ob der Server an eine oder an alle IP-Adressen gebunden wird" auf Seite 189
- „Port - Den Port angeben, an dem der Server Anforderungen empfängt" auf Seite 251
- „AdminPort - Port für die Anforderung von Verwaltungsseiten oder -formularen angeben" auf Seite 185
- „PureProxy - Dedizierten Proxy inaktivieren" auf Seite 267

Nähere Informationen hierzu finden Sie in Kapitel 4, „Die Datei ibmproxy.conf manuell editieren", auf Seite 13.

Konfigurations- und Verwaltungsformulare

Mit den folgenden Konfigurations- und Verwaltungsformularen können Sie die Werte der zugehörigen Anweisungen ändern:

- **Serverkonfiguration –> Basiseinstellungen –> Hostname**
- **Serverkonfiguration –> Basiseinstellungen –> Serverstammverzeichnis**
- **Serverkonfiguration –> Basiseinstellungen –> Standard-Port-Nummer(n)**
- **Serverkonfiguration –> Basiseinstellungen –> Administrator-Port-Nummer**
- **Serverkonfiguration –> Basiseinstellungen –> Bindungsoptionen**
- **Proxy-Konfiguration –> Leistung des Proxy-Servers –> Als reinen Proxy-Server ausführen**

Anmerkung: Die Anweisung HeaderServerName kann nicht mit den Konfigurations- und Verwaltungsformularen geändert werden.

Nähere Informationen hierzu finden Sie in Kapitel 2, „Konfigurations- und Verwaltungsformulare verwenden”, auf Seite 7.

Kapitel 7. Das Eigentumsrecht an Prozessen festlegen

Wenn ein anderer Benutzer als der Superuser root die Komponente Caching Proxy startet, hat dieser Benutzer das Eigentumsrecht an allen Prozessen, die dem Proxy-Server zugeordnet sind. Wird Caching Proxy jedoch vom Superuser root gestartet, liest eine Proxy-Server-Funktion zum Festlegen der Benutzer-IDs die Anweisungen UserId und GroupId aus der Datei ibmproxy.conf und erteilt dem angegebenen Benutzer und der angegebenen Gruppe das Eigentumsrecht an den Prozessen zu. Auf diese Weise wird der Dateizugriff beschränkt und das Computersystem geschützt. Wenn Sie die Anweisungen UserId und GroupId ändern, müssen Sie das Eigentumsrecht und die Berechtigungen für die Protokollverzeichnisse und andere Dateien wie ACLs (Access Control List, Zugriffssteuerungsliste), die der Proxy-Server verwendet, aktualisieren.

Anmerkung: Auf Linux-Systemen wird das Eigentumsrecht nur für Prozesse und Threads geändert, die für den Empfang von Verbindungen verantwortlich sind. Der Eigner von Prozessen und Threads, die für andere Aktivitäten im Arbeitsablauf zuständig sind, ist weiterhin root. Alle Prozesse und Threads erhalten eine Prozess-ID (PID). Der Befehl **ps** listet alle Prozess-IDs auf, unabhängig davon, ob sie einem Prozess oder einem Thread zugeordnet sind.

Sie legen das Eigentumsrecht für den Proxy-Server-Prozess fest, indem Sie mit den Anweisungen UserID, GroupID und PidFile die Benutzer-ID, die Gruppen-ID und die Position der Datei angeben, in der die Prozess-ID aufgezeichnet ist.

Wenn Sie die Ausführung des Proxy-Server-Prozesses im Vordergrund erzwingen möchten, setzen Sie die Anweisung NoBG auf "On".

Zugehörige Anweisungen

Mit den folgenden Anweisungen legen Sie das Eigentumsrecht für den Proxy-Server-Prozess fest:

- „UserId - Standard-Benutzer-ID angeben" auf Seite 285
- „GroupId - Gruppen-ID angeben" auf Seite 223
- „NoBG - Den Caching-Proxy-Prozess im Vordergrund ausführen" auf Seite 243
- „PidFile (nur Linux und UNIX) - Die Datei angeben, in der die Prozess-ID von Caching Proxy gespeichert werden soll" auf Seite 250

Nähere Informationen hierzu finden Sie in Kapitel 4, „Die Datei ibmproxy.conf manuell editieren", auf Seite 13.

Konfigurations- und Verwaltungsformulare

Mit den folgenden Konfigurations- und Verwaltungsformularen können Sie die Werte der zugehörigen Anweisungen ändern:

- **Serverkonfiguration –> Basiseinstellungen –> Benutzer-ID**
- **Serverkonfiguration –> Basiseinstellungen–> Gruppen-ID**
- **Serverkonfiguration –> Basiseinstellungen –> Dateiadresse der Prozess-ID**

Anmerkung: Die Anweisung NoBG kann nicht mit den Konfigurations- und Verwaltungsformularen geändert werden.

Nähere Informationen hierzu finden Sie in Kapitel 2, „Konfigurations- und Verwaltungsformulare verwenden”, auf Seite 7.

Kapitel 8. Verbindungen verwalten

Caching Proxy erzeugt für die Bearbeitung jeder Clientanforderung einen neuen Thread. Wenn keine Threads verfügbar sind, bewahrt der Proxy-Server die Anforderungen so lange auf, bis wieder Threads verfügbar sind. Mit zunehmender Anzahl aktiver Threads verbraucht der Proxy-Server auch mehr Speicher. Sie legen die maximale Anzahl aktiver Threads mit dem Wert für die Anweisung MaxActiveThreads fest.

Die Einstellung "ListenBacklog" gibt die maximale Anzahl anstehender Anforderungen für Clientverbindungen an, die der Server protokolliert, bevor er Verbindungen mit neuen Clients zurückweist. Legen Sie diesen Wert unter Berücksichtigung der Anzahl von Anforderungen fest, die der Server in wenigen Sekunden verarbeiten kann. Ein Server muss auf eine Clientverbindung reagieren, bevor das Clientzeitlimit abläuft. Geben Sie mit der Anweisung ListenBacklog die maximale Anzahl von Verbindungen an, die aufbewahrt werden soll.

Der Proxy-Server kann persistente Client/Server-Verbindungen verwalten. Bei der Verwendung einer persistenten Verbindung nimmt der Server mehrere Anforderungen des Clients entgegen und sendet seine Antworten über dieselbe TCP/IP-Verbindung. Persistente Verbindungen verringern die Latenzzeit für Clients und die CPU-Last auf dem Proxy-Server und verbrauchen nur geringfügig mehr Systemspeicher. Der Gesamtdurchsatz erhöht sich, wenn der Server nicht für jede Anforderung und Antwort eine gesonderte TCP/IP-Verbindung herstellen muss, und die TCP/IP-Verbindung kann am effektivsten genutzt werden, wenn sie persistent ist.

Durch die Bündelung von Verbindungen (Verbindungs-Pooling) auf Serverseite kommen die Vorteile persistenter Verbindungen auf Serverseite zum Tragen, weil vorhandene Verbindungen zwischen einem Proxy-Server und den Ursprungsservern wiederverwendet werden können. Bei jeder wiederverwendeten Verbindung werden drei TCP-Pakete eingespart (zwei Three-Way-Handshake-Pakete zum Herstellen der Verbindung und eines zum Schließen der Verbindung). Im Folgenden sind einige Vorteile der Bündelung von Verbindungen auf Serverseite aufgeführt:

- weniger Netzüberlastungen (durch Minimierung der Öffnungs- und -Schließvorgänge für Verbindungen),
- weniger CPU-Zeit, die von Routern, Clients und Servern beansprucht wird,
- niedrigere Speicherbelegung durch Clients und Server,
- bei fehlenden Cache-Einträgen schnellere Proxy-Antwort (weil das Öffnen und Schließen von Verbindungen vermieden wird).

Anmerkung: Die Bündelung von Verbindungen wird nur in einer kontrollierten Umgebung empfohlen. In einer Umgebung, in der die Ursprungsserver nicht mit HTTP 1.1 kompatibel sind, könnte sich die Leistung durch die Bündelung von Verbindungen verschlechtern. Außerdem müssen die Ursprungsserver ordnungsgemäß konfiguriert sein. Im Folgenden sehen Sie einen Beispielauszug aus der Konfigurationsdatei von Apache 1.3.19:

- `#KeepAlive: Legt fest, ob persistente Verbindungen zugelassen werden sollen (mehrere Anforderungen pro Verbindung). Zum Inaktivieren auf Off setzen.#`

- `KeepAlive On`
- `#MaxKeepAliveRequests: Die maximale Anzahl der Anforderungen, die während einer persistenten Verbindung zulässig sind. Auf 0 setzen, um eine unbegrenzte Anzahl Verbindungen zuzulassen. Für eine maximale Leistung sollte eine hohe Zahl gewählt werden.#`
- `Max KeepAliveRequests 0`
- `#KeepAliveTimeout: Die in Sekunden angegebene Wartezeit für die nächste Anforderung von demselben Client in derselben Verbindung#`
- `KeepAliveTimeout 240`

Diese Einstellungen halten Verbindungen zu den Webservern so lange geöffnet, wie sie verwendet werden, und erlauben dem Proxy-Server, die Verbindungen anstelle des Ursprungsservers zu verwalten. Daher werden die Verbindungen nur in der benötigten Menge im Pool verwaltet.

Wenn die Bündelung von Verbindungen auf Serverseite aktiviert ist, werden HTTP-Verbindungen zu den Ursprungsservern im Pool gespeichert. SSL-Verbindungen werden in Konfigurationen, in denen die Anweisung SSLEnable für den Proxy-Server auf "on" gesetzt ist, ebenfalls im Pool gespeichert.

Sie können den Verbindungspool durch Angabe der folgenden Einstellungen konfigurieren: maximale Anzahl nicht verwendeter Sockets, die pro Server bereitgehalten werden sollen, die Wartezeit für den Server, bevor dieser eine nicht genutzte persistente Verbindung beendet, das Zeitintervall, in dem der Garbage-Collection-Thread den Pool auf verfallene Verbindungen überprüft (das Standardzeitlimit für Verbindungen sind zwei Minuten).

Sie können den Zeitraum definieren, für den verschiedene Verbindungen geöffnet bleiben, indem Sie Werte für die Anweisungen InputTimeout, OutputTimeout, PersistTimeout, ReadTimeout und ScriptTimeout festlegen.

Zugehörige Anweisungen

Mit den folgenden Anweisungen können Sie Verbindungen mit dem Proxy-Server-Prozess verwalten:

- „MaxActiveThreads - Die maximale Anzahl aktiver Threads angeben" auf Seite 238
- „ConnThreads — Anzahl der Verbindungs-Threads für die Verbindungsverwaltung festlegen" auf Seite 203
- „ListenBacklog - Den für den Server zulässigen Empfangsrückstand (listen-Backlog) für Clientverbindungen angeben" auf Seite 233
- „ProxyPersistence - Persistente Verbindungen zulassen" auf Seite 265
- „MaxPersistRequest - Die maximale Anzahl Anforderungen angeben, die über eine persistente Verbindung empfangen werden können" auf Seite 240
- „ServerConnPool - Das Pooling von Verbindungen zum Ursprungsserver festlegen" auf Seite 275
- „MaxSocketPerServer - Die maximale Anzahl offener Sockets für den Server angeben" auf Seite 241
- „ServerConnTimeout - Maximale Zeit der Inaktivität festlegen" auf Seite 275

- „ServerConnGCRun - Intervall für die Ausführung des Garbage-Collection-Thread festlegen" auf Seite 274
- „PersistTimeout - Wartezeit zwischen Clientanforderungen angeben" auf Seite 249
- „InputTimeout - Zeitlimit für Eingabe festlegen" auf Seite 229
- „ReadTimeout - Zeitlimit für eine Verbindung angeben" auf Seite 269
- „OutputTimeout - Zeitlimit für die Ausgabe angeben" auf Seite 247
- „ScriptTimeout - Zeitlimiteinstellung für Scripts angeben" auf Seite 273

Nähere Informationen hierzu finden Sie in Kapitel 4, „Die Datei ibmproxy.conf manuell editieren", auf Seite 13.

Konfigurations- und Verwaltungsformulare

Mit den folgenden Konfigurations- und Verwaltungsformularen können Sie die Werte der zugehörigen Anweisungen ändern:

- **Serverkonfiguration –> Systemverwaltung –> Leistung –> Maximale Anzahl aktiver Threads**
- **Serverkonfiguration –> Systemverwaltung –> Leistung –> Größe des listen-Backlog**
- **Proxy-Konfiguration –> Leistung des Proxy-Servers –> Persistente Verbindungen zulassen**
- **Serverkonfiguration –> Systemverwaltung –> Leistung –> Maximale Anzahl der Anforderungen**
- **Serverkonfiguration –> Systemverwaltung –> Leistung –> Zeitlimit für persistente Verbindung**
- **Serverkonfiguration –> Systemverwaltung –> Zeitlimits –> Zeitlimit für Eingabe**
- **Serverkonfiguration –> Systemverwaltung –> Zeitlimits –> Zeitlimit für Lesen**
- **Serverkonfiguration –> Systemverwaltung –> Zeitlimits –> Zeitlimit für Ausgabe**
- **Serverkonfiguration –> Systemverwaltung –> Zeitlimits –> Zeitlimit für Scripts**
- **Serverkonfiguration –> Systemverwaltung –> Zeitlimits –> Zeitlimit für persistente Verbindung**

Anmerkungen:

1. Die Anweisungen ServerConnPool, MaxsocketPerServer, ServerConnTimeout und ServerConnGCRun können nicht mit den Konfigurations- und Verwaltungsformularen geändert werden.
2. Die Anweisung PersistTimeout kann mit dem Formular **Serverkonfiguration –> Systemverwaltung –> Leistung** und mit dem Formular **Serverkonfiguration –> Systemverwaltung –> Zeitlimits** geändert werden.

Nähere Informationen hierzu finden Sie in Kapitel 2, „Konfigurations- und Verwaltungsformulare verwenden", auf Seite 7.

Kapitel 9. Proxy-Server-Prozess optimieren

Sie können die Leistung von Caching Proxy merkbar verbessern, indem Sie das System ordnungsgemäß konfigurieren und optimieren. Nachfolgend sind Empfehlungen zur Verbesserung der Konfiguration und zur Optimierung aufgeführt.

Leistungsbezogene Anweisungen definieren

Die folgenden Anweisungen wirken sich erheblich auf die Leistung des Proxy-Server-Prozesses aus:

- „PureProxy - Dedizierten Proxy inaktivieren" auf Seite 267. Diese Funktion verbessert die Systemleistung durch vollständiges Inaktivieren des Caching.
- „ProxyPersistence - Persistente Verbindungen zulassen" auf Seite 265. Diese Funktion ermöglicht Clients und Servern die Verwaltung offener Verbindungen. Durch persistente Verbindungen wird die Zeitverzögerung für Dokumentanforderungen vom Proxy-Server verringert, allerdings erfordern sie eine höhere Netzbandbreite sowie einen dedizierten Server-Thread für jede Verbindung. Lassen Sie keine persistenten Verbindungen zu, wenn die Anzahl der verfügbaren Threads durch die Konfiguration eingeschränkt ist.

In den folgenden Feldern der Konfigurations- und Verwaltungsformulare können Sie die Werte der zugehörigen Anweisungen ändern:

- **Proxy-Konfiguration –> Leistung des Proxy-Servers: Als reinen Proxy-Server ausführen**
- **Proxy-Konfiguration –> Leistung des Proxy-Servers: Persistente Verbindungen zulassen**

Andere Anwendungen überprüfen

Überprüfen Sie die Service- und Dämonprozesse, die auf dem System ausgeführt werden, und entfernen Sie die nicht benötigten Prozesse, um Speicher freizugeben und die CPU zu entlasten. Wenn auf dem System beispielsweise ein Webserver ausgeführt wird, der nur wenige Webseiten bereitstellt, sollten Sie in Erwägung ziehen, Caching Proxy als einzigen Webserver einzusetzen. Inaktivieren Sie andere Webserver wie folgt:

- Unter AIX: Entfernen Sie die Einträge für die Webserver in /etc/inittab.
- Unter Linux: Entfernen Sie die Verknüpfungen zu den Webservern im Verzeichnis /etc/rc.d/rc*x*.d der Standardausführungsebene Ihres Systems (normalerweise 2).
- Unter HP-UX und Solaris: Entfernen Sie die Verknüpfungen zu den Webservern im Verzeichnis /etc/rc*x*.d der Standardausführungsebene Ihres Systems (normalerweise 2).
- Auf Windows-Systemen:
  1. Klicken Sie auf **Start –> Einstellungen (für Windows 2000) –> Systemsteuerung –> Verwaltung –> Dienste**.
  2. Überprüfen Sie die Dienste, die nicht erforderlich, aber auf `Automatisch` eingestellt sind.
  3. Ändern Sie die Startart für diese Dienste von `Automatisch` in `Manuell`.

Paging-Bereich prüfen

Stellen Sie sicher, dass Ihr System über einen ausreichenden Paging-Bereich verfügt, um einen ordnungsgemäßen Betrieb zu gewährleisten. Das System benötigt einen Paging-Bereich, der doppelt so groß ist wie der Arbeitsspeicher. Verteilen Sie den Paging-Bereich, sofern möglich, auf mehrere physische Laufwerke. Beispielsweise ist für einen Server des Typs Netfinity 5000 mit 512 MB Arbeitsspeicher und fünf SCSI-Laufwerken ein Gesamt-Paging-Bereich von 1 GB mit ungefähr 200 MB auf jedem Laufwerk erforderlich.

Dateisystem optimieren

Caching Proxy erstellt und löscht während seiner Ausführung viele Dateien. Wenn der Proxy-Server Zugriffe protokolliert (im Zugriffsprotokoll, im Proxy-Zugriffsprotokoll oder im Cache-Zugriffsprotokoll), sollte jedem Protokoll ein eigenes Dateisystem zugewiesen werden. Auf diese Weise können Sie verhindern, dass bei einem unerwarteten Anschwellen der Protokolle kein Speicher belegt wird, der für eine andere Funktion vorgesehen ist (z. B. für den Cache).

TCP/IP-Konfiguration optimieren

Caching Proxy reagiert sensibel auf Änderungen der TCP/IP-Konfiguration. Wenn Sie die TCP/IP-Werte in einem Betriebssystem verringern, kann dies ein unerwartetes Verhalten des Proxy-Servers zur Folge haben. Falls zu niedrige TCP/IP-Werte verwendet werden, kann es vorkommen, dass Verbindungen von den Clients, die Verbindungen zum Proxy-Server herstellen, oder vom Ursprungsserver, zu dem der Proxy-Server eine Verbindung aufbaut, zurückgesetzt werden. Dies gilt insbesondere für Clients, die mit niedriger Bandbreite (56700 bps oder weniger) mit dem Proxy-Server verbunden sind. Gehen Sie deshalb vorsichtig vor, wenn Sie die TCP/IP-Parameterwerte herabsetzen müssen.

TCP-Warteintervall für Umgebungen mit hoher Arbeitslast optimieren (HP-UX, Linux, Solaris, Windows)

Das TCP-Warteintervall gibt an, wie lange ein Socket auf ein FIN-Paket vom Sender wartet, bevor er das Schließen der Verbindung erzwingt. In Umgebungen mit hoher Arbeitslast scheint der Proxy-Server zu blockieren, wenn viele Sockets im Status TIME_WAIT verbleiben, nachdem ihre Verbindungen geschlossen wurden. Die Wahl eines kürzeren TCP-Warteintervalls verringert die Anzahl der Sockets im Wartezustand und kann in Umgebungen mit hoher Arbeitslast das scheinbare Blockieren des Proxy-Servers verhindern. Es wird empfohlen, dieses Intervall auf 5 Sekunden einzustellen.

Gehen Sie wie folgt vor, um das TCP-Warteintervall auf 5 Sekunden einzustellen:

- Unter HP-UX:

  Geben Sie den folgenden Befehl ein:

  ```
  ndd /dev/tcp -set tcp_time_wait_interval 5000
  ```

  Verwenden Sie das Dienstprogramm ″sam″, um den Kernel-Parameter max-_thread_proc auf 2048 oder höher zu setzen.

Anmerkung: Sie können außerdem die folgenden Kernel-Parameter anpassen: maxfiles, maxfiles_lim, maxproc, shmem, tcp_conn_request_max, tcp_ip_abort_interval, tcp_keepalive_interval, tcp_rexmit_interval_initial, tcp_rexmit_interval_max, tcp_rexmit_interval_min, tcp_xmit_hiwater_def, tcp_recv_hiwater_def.

- Unter Linux:

  Geben Sie die folgenden Befehle ein:

  ```
  echo "1024 61000" > /proc/sys/net/ipv4/ip_local_port_range
  echo "5" > /proc/sys/net/ipv4/tcp_fin_timeout
  ```

- Unter Solaris:

  Geben Sie den folgenden Befehl ein:

  ```
  ndd /dev/tcp –set tcp_time_wait_interval 5000
  ```

  Öffnen Sie die Datei /etc/system in einem Editor und ändern Sie sie wie folgt:

  ```
  set tcp:tcp_conn_hash_size=8129
  ```

- Unter Windows:

  Sie müssen einen Eintrag in der Registrierungsdatenbank erstellen, um das TCP-Warteintervall festzulegen. Nähere Informationen hierzu finden Sie in der Windows-Dokumentation.

Linux-Kernel anpassen

Einige Grenzwerte im Linux-Kernel sind niedrig und können geändert werden. Einige Grenzwerte können über das Dateisystem /proc geändert werden, andere erfordern ein erneutes Kompilieren des Kernel.

Anmerkung: Das Dateisystem /proc ist virtuell, d. h., es existiert nicht physisch auf dem Datenträger. Es ist vielmehr eine Schnittstelle zum Linux-Kernel. Da es physisch nicht existiert, gehen Ihre Eingabewerte bei einem Neustart verloren. Daher sollten Änderungen am Dateisystem /proc unter RedHat in der Datei /etc/rc.d/rc.local und unter SuSE in der Datei /etc/rc.config vorgenommen werden. In diesem Fall werden die Änderungen beim Neustart aktiviert.

Im Folgenden sind einige Empfehlungen aufgelistet:

- Die Maximalzahl für Dateideskriptoren ist standardmäßig 4096. Dieser Wert kann geändert werden, indem folgende Zeile in die Datei rc.local aufgenommen wird:

  ```
  echo 32768 > /proc/sys/fs/file-max
  ```

- Die Maximalanzahl für I-Nodes ist standardmäßig 16384. Dieser Wert kann geändert werden, indem folgende Zeile in die Datei rc.local aufgenommen wird:

  ```
  echo 65536 > /proc/sys/fs/inode-max
  ```

- Die TCP- und UDP-Ports liegen standardmäßig im Bereich zwischen 1024 und 4999. Dieser Bereich kann durch Hinzufügen der folgenden Zeile zur Datei rc.local in 32768 - 61000 geändert werden:

  ```
  echo 32768 61000 > /proc/sys/net/ipv4/ip_local_port_range
  ```

- Der Standardwert für die zulässige Task-Anzahl ist 512. Wenn zu viele Tasks aktiv sind, wirkt sich dies auf die maximale Anzahl Threads für einen Prozess aus. Dieser Grenzwert kann mit der Einstellung NR_TASKS in der Datei *Ihre_Kernel-Quelle*/include/linux/tasks.h auf 2048 erhöht werden.
- Ändern Sie außerdem den Wert von MIN_TASKS_LEFT_FOR_ROOT in 24. Sie müssen den Kernel erneut kompilieren, damit diese Änderung wirksam wird.

Wenn Sie den Kernel neu generieren möchten, aktivieren Sie nur Optionen, die Sie wirklich benötigen. Wenn Sie einen bestimmten Dämon nicht benötigen, führen Sie ihn nicht aus.

Die Variablen für die Thread-Optimierung unter AIX anpassen

Auf AIX-Systemen kann die Leistung von Caching Proxy durch Verwendung von System-Threads und mehrerer Heap-Speicher für die Threads verbessert werden. Die Leistung ist von der Mehrprozessorunterstützung des Betriebssystems und der Thread-Planung des Betriebssystems abhängig. Sie können eine Leistungsverbesserung erzielen, indem Sie die Variablen für die Thread-Optimierung unter AIX wie folgt definieren:

```
export AIXTHREAD_SCOPE=S
export SPINLOOPTIME=500
export YIELDLOOPTIME=100
export MALLOCMULTIHEAP=1
```

Sie können diese Umgebungsvariablen vor dem Start von `/usr/sbin/ibmproxy` setzen oder der Datei `/etc/rc.ibmproxy` hinzufügen, falls Sie den Proxy-Server mit **startsrc -s ibmproxy** starten. Nach der Anpassung dieser Variablen für die Thread-Optimierung ist auf SMP-Systemen eine deutliche Leistungsverbesserung erkennbar. In manchen Fällen kann auch auf Einzelprozessorsystemen eine Leistungsverbesserung festgestellt werden.

Anmerkung: Nähere Informationen zu den Variablen für die Thread-Optimierung finden Sie in der Dokumentation zum Betriebssystem AIX.

Teil 3. Das Verhalten von Caching Proxy konfigurieren

Dieser Teil erläutert, wie die Komponente Caching Proxy Clientanforderungen beantwortet, und beschreibt die Prozeduren zum Konfigurieren dieses Verhaltens. Diese Elemente der Proxy-Server-Konfiguration werden in der Regel von einem Webadministrator verwaltet und haben keine Auswirkung auf andere Prozesse auf dem Hostcomputersystem oder andere Computersysteme im Netzwerk.

Dieser Teil enthält die folgenden Kapitel:

Kapitel 10. Verarbeitung von Anforderungen verwalten

Wenn Caching Proxy eine Clientanforderung empfängt, führt er die im Methodenfeld angegebene Aktion für das im URL-Feld angegebene Objekt aus, sofern die angeforderte Methode aktiviert ist. Der Proxy-Server löst den URL gemäß einer Reihe vom Administrator definierten Zuordnungsregeln auf. Diese Zuordnungsregeln können Caching Proxy beispielsweise anweisen, als Webserver aufzutreten und das Objekt aus dem lokalen Dateisystem abzurufen oder als Proxy-Server aufzutreten und das Objekt von einem Ursprungsserver abzurufen.

Dieses Kapitel beschreibt, wie Sie Methoden aktivieren, Zuordnungsregeln definieren und einen Ersatz-Proxy-Server konfigurieren.

HTTP/FTP-Methoden aktivieren

Clientanforderungen an den Server enthalten ein Methodenfeld, das die Aktion angibt, die der Server für das angegebene Objekt ausführen soll.

In der folgenden Liste sind die Methoden aufgeführt, die vom Proxy-Server unterstützt werden. Außerdem wird beschrieben, wie der Proxy-Server auf eine Clientanforderung, die die Methode enthält, reagiert, wenn die Methode aktiviert ist.

Anmerkung: Einige Methoden gelten sowohl für HTTP- als auch für FTP-Anforderungen. Wenn Sie diese Methoden für HTTP aktivieren, werden sie auch für FTP aktiviert.

DELETE
Der Proxy-Server löscht das im URL angegebene Objekt. Mit der Methode DELETE können Clients Dateien von Ihrem Caching-Proxy-Server löschen. Definieren Sie mit den Zugriffsschutzkonfigurationen für den Server, wer die Methode DELETE für welche Dateien verwenden darf. Ausführliche Informationen hierzu finden Sie in Kapitel 25, „Zugriffsschutzkonfigurationen für den Server", auf Seite 115.

GET Der Proxy-Server gibt die mit dem URL angegebenen Daten zurück. Wenn der URL auf ein ausführbares Programm verweist, gibt der Proxy-Server die Ausgabe des Programms zurück. Diese Methode kann in persistenten Verbindungen verwendet werden.

HEAD
Der Proxy-Server gibt nur den im URL angegebenen HTTP-Dokument-Header ohne den Dokumenthauptteil zurück.

OPTIONS
Der Proxy-Server gibt Informationen zu den Übertragungsoptionen in der Anforderungs-/Antwortkette zurück, die im URL angegeben ist. Mit dieser Methode kann ein Client die einem Objekt zugeordneten Optionen und Voraussetzungen oder das Leistungsspektrum eines Servers bestimmen, ohne das Objekt bearbeiten oder abrufen zu müssen.

POST Die Anforderung enthält Daten und einen URL. Der Proxy-Server empfängt die in der Anforderung enthaltenen Daten als neue untergeordnete Instanz für die im URL angegebene Ressource, die die Daten verarbeitet.

Die Ressource kann ein Datenempfangsprogramm, ein Gateway für ein anderes Protokoll oder ein gesondertes Programm für den Empfang von Anmerkungen bzw. Anhängen sein.

Die Methode POST ist so konzipiert, dass sie Anhänge vorhandener Ressourcen bearbeiten kann. Beispiele hierfür sind das Senden einer Nachricht an ein Schwarzes Brett, eine Newsgroup, eine Mailing-Liste oder eine ähnliche Gruppe von Ressourcen, die Bereitstellung eines Datenblocks, z. B. aus einem Formular an ein Datenverarbeitungsprogramm, oder die Erweiterung einer Datenbank durch eine Anfügeoperation (**append**). In Caching Proxy selbst wird die Methode POST für die Verarbeitung der Konfigurations- und Verwaltungsformulare verwendet.

Die Methode kann in persistenten Verbindungen verwendet werden.

PUT Die Anforderung enthält Daten und einen URL. Der Proxy-Server speichert die Daten in der im URL angegebenen Ressource. Wenn die Ressource bereits vorhanden ist, ersetzt PUT sie durch die in der Anforderung enthaltenen Daten. Falls die Ressource noch nicht vorhanden ist, wird sie erstellt und mit den in der Anforderung enthaltenen Daten gefüllt. Die Methode kann in persistenten Verbindungen verwendet werden.

Wenn Sie die Methode PUT aktivieren, können Dateien mit HTTP und FTP auf den Caching Proxy geschrieben werden. Da die Clients mit der Methode PUT auf den Caching Proxy schreiben können, müssen Sie mit den Zugriffsschutzkonfigurationen für den Server definieren, wer die Methode PUT für welche Dateien verwenden darf. (Nähere Informationen hierzu finden Sie in Kapitel 25, „Zugriffsschutzkonfigurationen für den Server", auf Seite 115.)

TRACE

Der Proxy-Server meldet die vom Client gesendete Anforderungsnachricht zurück. Mit dieser Methode ist der Client in der Lage zu erkennen, was am anderen Ende der Anforderungskette empfangen wird, und diese Daten für Test- oder Diagnosezwecke zu verwenden. Der Inhaltstyp der Proxy-Antwort ist message/http.

Zugehörige Anweisungen

Mit den folgenden Anweisungen können Sie die HTTP/FTP-Methoden aktivieren:

- „Enable - HTTP-Methoden aktivieren" auf Seite 213
- „Disable - HTTP-Methoden inaktivieren" auf Seite 211

Nähere Informationen hierzu finden Sie in Kapitel 4, „Die Datei ibmproxy.conf manuell editieren", auf Seite 13.

Konfigurations- und Verwaltungsformulare

In den folgenden Konfigurations- und Verwaltungsformularen können Sie die Werte der zugehörigen Anweisungen ändern:

- **Serverkonfiguration –> Anforderungsverarbeitung –> HTTP-Methoden –> GET**
- **Serverkonfiguration –> Anforderungsverarbeitung –> HTTP-Methoden –> HEAD**
- **Serverkonfiguration –> Anforderungsverarbeitung –> HTTP-Methoden –> POST**
- **Serverkonfiguration –> Anforderungsverarbeitung –> HTTP-Methoden –> PUT**
- **Serverkonfiguration –> Anforderungsverarbeitung –> HTTP-Methoden –> DELETE**
- **Serverkonfiguration –> Anforderungsverarbeitung –> HTTP-Methoden –> OPTIONS**
- **Serverkonfiguration –> Anforderungsverarbeitung –> HTTP-Methoden –> TRACE**

Anmerkung: Wenn Sie die Methode POST inaktivieren, ist eine Konfiguration von Caching Proxy mit den Konfigurations- und Verwaltungsformularen nicht möglich.

Nähere Informationen hierzu finden Sie in Kapitel 2, „Konfigurations- und Verwaltungsformulare verwenden", auf Seite 7.

Zuordnungsregeln definieren

Zuordnungsregeln sind Konfigurationsanweisungen, die dafür sorgen, dass Clientanforderungen für Caching Proxy auf eine bestimmte Weise verarbeitet werden, z. B. dass die Anforderungen (über einen Proxy) an einen Ursprungsserver weitergeleitet, umgeleitet oder zurückgewiesen werden. Die korrekte Definition von Zuordnungsregeln ist für eine ordnungsgemäße Funktionsweise von Caching Proxy von entscheidender Bedeutung. Zuordnungsregeln wirken sich auf Folgendes aus:

- die grundlegende Ausführung des Proxy-Servers,
- den Zugriff auf die Browser-basierten Konfigurations- und Verwaltungsformulare,
- die Möglichkeit, Servlet-Ergebnisse und andere dynamisch generierte Inhalte im Cache zu speichern.

Die Anweisungen für Zuordnungsregeln haben das folgende Format:

```
Regel Schablone Ziel [IP-Adresse | Hostname]:[Port]
```

Nur die Anforderungen, die der angegebenen Schablone und der IP/Port-Kombination entsprechen, unterliegen dieser Regel. Eine Schablone kann Platzhalterzeichen enthalten, z. B. https://**/*.asp.

Die Reihenfolge, in der die Regeln in der Konfigurationsdatei angegeben sind, ist von entscheidender Bedeutung. Wenn eine Anforderung einer Schablone entspricht, wird sie sofort verarbeitet, und alle nachfolgenden Regeln werden nicht mehr ausgewertet. Dies gilt jedoch nicht für Map-Anweisungen. Die Anweisung Map ersetzt den URL in der Anforderung. Diese neue Anforderung wird anschließend mit den verbleibenden Zuordnungsregeln verglichen.

Zuordnungsregeln

Die folgenden Zuordnungsregeln gelten für Clientanforderungen, die der angegebenen Schablone entsprechen:

- **Map** — Die Anforderung wird umgeschrieben. Die Regel Map ersetzt einen Anforderungs-URL (Schablone) durch eine andere URL-Zeichenfolge (Ziel). Nach dieser Ersetzung wird die Anforderung, die jetzt die neue Zeichenfolge enthält, mit den verbleibenden Zuordnungsregeln verglichen.
- **Pass, Exec** — Die Anforderung wird lokal bearbeitet. Die Regeln Pass und Exec verarbeiten die Anforderung auf dem Proxy-Server. Die Regel Pass ordnet einen Anforderungs-URL (Schablone) einer Datei zu, die der Proxy-Server (Ziel) bereitstellt. Die Regel Exec ordnet einen Anforderungs-URL einem CGI-Programm zu, das auf dem Proxy-Server ausgeführt wird.
- **Fail** — Die Anforderung wird zurückgewiesen. Die Regel Fail weist eine Anforderung (Schablone) auf dem Proxy-Server zurück. Alle Anforderungen, die der Schablone einer Fail-Regel entsprechen, werden nicht weiter verarbeitet. Fail-Regeln haben keine Zielargumente.
- **Redirect** — Die Anforderung wird weitergeleitet. Die Regel Redirect leitet eine Anforderung (Schablone) an einen anderen Webserver (Ziel) weiter. Da ein vollständiger URL (einschließlich des Übertragungsprotokolls) das Ziel dieser Regel ist, kann das Protokoll während der Umleitung geändert werden, um beispielsweise einer HTTP-Anforderung SSL-Verschlüsselung hinzuzufügen. Bei einer Umleitung wird der Cache vor der Bearbeitung der Anforderung nicht überprüft.
- **Proxy, ProxyWAS** — Die Anforderung wird (über einen Proxy-Server) weitergeleitet. Die Regeln Proxy und ProxyWAS übergeben Anforderungen (Schablonen) an einen anderen Server (Ziel). Anders als einfache Redirect-Regeln erlauben Proxy-Regeln dem Proxy-Server, für die Bearbeitung von Anforderungen den Cache zu überprüfen, Inhalte von Ursprungsservern im Cache zu speichern und HTTP-Header zu schreiben, die erweiterte Funktionen unterstützen. Verwenden Sie die Regel ProxyWAS anstelle der Regel Proxy, wenn es sich bei dem Ursprungsserver um einen WebSphere Application Server handelt.

Die folgende Zuordnungsregel gilt für Antworten des Ursprungsservers:

- **ReversePass** — Automatisch umgeleitete Anforderungen werden abgefangen. Die Regel ReversePass vergleicht die Antwort des Ursprungsservers mit der Schablone, wenn sie über den Proxy-Server an den Client übergeben wird. Mit der Regel ReversePass können Umleitungsstatuscodes ermittelt werden, die dazu führen würden, dass ein Client in direkten Kontakt mit dem Ursprungsserver tritt. Der Client wird angewiesen, eine Verbindung zu dem im Zielargument definierten Server herzustellen.

Die folgenden Zuordnungsregeln gelten für API-Anwendungen:

- **nameTrans** — Empfängt die Anforderung und führt für die Namensumsetzung während der Anforderungsverarbeitung die im Ersetzungsdateipfad angegebene API-Anwendung aus.
- **service** — Empfängt die Anforderung und führt die für die Bereitstellung während der Anforderungsverarbeitung im Ersetzungsdateipfad angegebene API-Anwendung aus.

Ersatzserver konfigurieren

Gehen Sie wie folgt vor, um einen Standardersatzserver zu konfigurieren:

- Legen Sie Port 80 als Port für den Proxy-Server fest.

  ```
  Port 80
  ```

- Fügen Sie vor allen anderen Regeln eine Regel Proxy hinzu, die alle an Port 80 empfangenen Anforderungen an den Ursprungsserver weiterleitet.

  ```
  Proxy /* http://der.Inhalts.Server.com/* :80
  ```

- Aktivieren Sie den Verwaltungs-Port an einem anderen Port als Port 80.

  ```
  AdminPort 8080
  ```

Damit wird der gesamte HTTP-Datenverkehr an Port 80 an den Ursprungsserver weitergeleitet. Datenverkehr, der am Verwaltungs-Port empfangen wird, entspricht nicht der ersten Proxy-Regel mit Platzhalterzeichen und bleibt deshalb von dieser Regel unberührt. Die verbleibenden Zuordnungsregeln werden für die Bearbeitung der Anforderung verwendet.

Zugehörige Anweisungen

Mit den folgenden Anweisungen definieren Sie die Zuordnungsregeln:

- „Map - Übereinstimmende Anforderungen in eine neue Anforderungszeichenfolge ändern" auf Seite 237
- „Pass - Die Schablone zum Akzeptieren von Anforderungen angeben" auf Seite 247
- „Exec - Für übereinstimmende Anforderungen ein CGI-Programm ausführen" auf Seite 217
- „Redirect - Schablone für Anforderungen angeben, die an einen anderen Server gesendet werden" auf Seite 269
- „Proxy - Proxy-Protokolle oder einen Reverse Proxy angeben" auf Seite 261
- „ProxyWAS - Festlegen, dass Anforderungen an WebSphere Application Server gesendet werden" auf Seite 266
- „ReversePass - Automatisch umadressierte Anforderungen abfangen" auf Seite 270

Nähere Informationen hierzu finden Sie in Kapitel 4, „Die Datei ibmproxy.conf manuell editieren", auf Seite 13.

Konfigurations- und Verwaltungsformulare

Mit den folgenden Konfigurations- und Verwaltungsformularen können Sie die Werte der zugehörigen Anweisungen ändern:

- **Serverkonfiguration –> Anforderungsverarbeitung –> Routing von Anforderungen**

Anmerkung: Die Konfigurations- und Verwaltungsformulare bieten keine Unterstützung für das Argument Port-Nummer.

Nähere Informationen hierzu finden Sie in Kapitel 2, „Konfigurations- und Verwaltungsformulare verwenden", auf Seite 7.

Umschreiben von Junctions aktivieren (optional)

Die Anweisung JunctionRewrite aktiviert die Routine für das Umschreiben von Junctions in Caching Proxy, die die Antworten von den Ursprungsservern so umschreibt, dass URLs, die relativ zum Server angegeben sind, dem richtigen Ursprungsserver zugeordnet werden, wenn Junctions verwendet werden. Das Plug-in JunctionRewrite muss ebenfalls aktiviert sein. Junctions werden mit den Proxy-Zuordnungsregeln definiert.

Wenn Sie die Junction mit den Proxy-Zuordnungsregeln definieren, können Sie die Anweisung Proxy mit oder ohne die Option JunctionPrefix verwenden.

Junction ohne die Option JunctionPrefix definieren

Die folgenden Beispiele sind gültige Junctions, die von der Routine für das Umschreiben von Junctions bearbeitet werden können:

- `Proxy /shop/* http://shopserver.acme.com/*`
- `Proxy /auth/* http://authserver.acme.com/*`

Das folgende Beispiel zeigt eine gültige Junction, die von der Routine für das Umschreiben von Junctions *nicht* bearbeitet wird:

- `Proxy /* http://defaultserver.acme.com/*`

Im Folgenden sehen Sie Beispiele für ungültige Junctions:

- `Proxy /images/*.gif http://imageserver.acme.com/images/*.gif`
- `Proxy /cgi-bin/* http://cgiserver.acme.com/cgi/perl/*`

Diese Zuordnungsregeln erstellen Junctions für shopserver, authserver, und b2bserver. shopserver gibt ein HTML-Dokument mit den folgenden URLs zurück, die in die entsprechenden HTML-Tags eingeschlossen sind:

- /index.html (Referenz relativ zum Server)
- /images/shop.gif (Referenz relativ zum Server)
- buy/buy.jsp (Referenz relativ zum Verzeichnis)
- http://ebay.com (absolute Referenz)

Die Routine für das Umschreiben von Junctions schreibt die relativ zum Server angegebenen Referenzen gemäß den Proxy-Zuordnungsregeln wie folgt um:

- /shop/index.html (geändert)
- /shop/images/shop.gif (geändert)
- buy/buy.jsp (unverändert)
- http://ebay.com (unverändert)

Junction mit der Option JunctionPrefix definieren (empfohlene Methode)

Wenn Sie die Option JunctionPrefix mit der Anweisung Proxy verwenden, anstatt das Junction-Präfix aus dem ersten URL-Muster in der Proxy-Regel abzuleiten, können Sie das Junction-Präfix mit dem folgenden Format in der Proxy-Regel deklarieren:

`Proxy` *`URL-Muster1 URL-Muster2`* `JunctionPrefix:`*`URL-Präfix`*

Wenn Sie die Option JunctionPrefix verwenden, gibt es bezüglich des Formats des ersten URL-Musters keine Einschränkungen. Damit das Umschreiben von Junctions unterstützt wird, auch wenn die Option JunctionPrefix *nicht* verwendet wird, muss der Proxy-URL das folgende Format haben: `Proxy /market/* http://b2bserver/*`. Wenn Sie die Option JunctionPrefix jedoch verwenden, ist die folgende Proxy-Regel für das Umschreiben von Junctions gültig:

```
Proxy  /market/partners/*.html http://b2bserver.acme.com/*.html
       junctionprefix:/market/partners
```

Die Routine für das Umschreiben von Junctions wirkt sich auf die folgenden Tags aus:

Tabelle 3. Tags, die von der Routine für das Umschreiben von Junctions betroffen sind

Tag	**Attribute**
!—	URL
a	href
applet	archive, codebase
area	href
base	href
body	background
del	cite
embed	pluginspage
form	action
input	src
frame	src, longdesc
iframe	src, longdesc
ilayer	src, background
img	src, usemap, lowsrc, longdesc, dynsrc
layer	src, background
link	href
meta	url
object	data, classid, codebase, codepage
script	src
table	background
td	background
th	background
tr	background

Anmerkung: Die Routine für das Umschreiben von Junctions hat keine Auswirkungen auf Tags, die von JavaScript oder von Plug-ins im Browser generiert werden.

Zugehörige Anweisungen

Sie können zum Aktivieren der Routine für das Umschreiben von Junctions und des zugehörigen Plug-in die folgenden Anweisungen verwenden:

- „ServerInit - Schritt ″Server Initialization″ anpassen” auf Seite 275
- „Transmogrifier - Schritt ″Data Manipulation″ anpassen” auf Seite 283
- „JunctionRewrite - Umschreiben von URLs aktivieren” auf Seite 230
- „JunctionRewriteSetCookiePath — Bei Verwendung des Plug-in JunctionRewrite die Pfadoption im Header Set-Cookie umschreiben” auf Seite 230
- „JunctionReplaceUrlPrefix — Bei Verwendung des Plug-in JunctionRewrite den URL ersetzen, anstatt Präfix einzufügen” auf Seite 230
- „JunctionSkipUrlPrefix — Bei Verwendung des Plug-in JunctionRewrite das Überschreiben von URLs überspringen, die das Präfix bereits enthalten” auf Seite 231

Nähere Informationen hierzu finden Sie in Kapitel 4, „Die Datei ibmproxy.conf manuell editieren”, auf Seite 13.

Konfigurations- und Verwaltungsformulare

Verwenden Sie das folgende Konfigurations- und Verwaltungsformular, um das Plug-in JunctionRewrite zu aktivieren:

- **Serverkonfiguration –> Anforderungsverarbeitung –> Verarbeitung von API-Anforderungen**

Anmerkung: Die Konfigurations- und Verwaltungsformulare bieten keine Unterstützung für die Anweisung JunctionRewrite.

Nähere Informationen hierzu finden Sie in Kapitel 2, „Konfigurations- und Verwaltungsformulare verwenden”, auf Seite 7.

UseCookie als Alternative zu JunctionRewrite

Mit Cookies können Informationen zum Back-End-Server gespeichert werden. Es wird ein Cookie an den Client-Browser gesendet. Wenn der Browser Ressourcen in der HTML-Seite anfordert, hängt er ein Cookie an diese Anforderungen an, damit Caching Proxy die Anforderungen an den richtigen Back-End-Server weiterleitet.

Wenn Sie Cookies als Alternative zu JunctionRewrite verwenden möchten, müssen Sie die folgenden Änderungen in der Datei ibmproxy.conf vornehmen:

1. Ersetzen Sie **JunctionRewrite on** durch **JunctionRewrite on UseCookie**.
2. Setzen Sie das Plug-in JunctionRewrite auf Kommentar.

Es folgt ein Vergleich zwischen dem Plug-in JunctionRewrite und der Cookie-Implementierung.

- Plug-in JunctionRewrite
  - Die HTML-Seite wird umgeschrieben.
  - Das Plug-in unterstützt kein Umschreiben von Script-basierten Sprachen und Applets, sofern das Plug-in Transmogrifier nicht verwendet wird. Nähere Informationen hierzu finden Sie im Abschnitt „Beispiel-Plug-in Transmogrifier zur Erweiterung der Funktionalität von JunctionRewrite” auf Seite 51.
  - Das Plug-in bietet eine geringere Leistung.

- Es sind keine Einschränkungen bezüglich der Konfiguration von Back-End-Servern vorhanden. Der Client kann in einer Sitzung auf mehrere Back-End-Server zugreifen.

- Cookie-Implementierung
  - Die HTML-Seite wird *nicht* umgeschrieben. Es wird ein Cookie an den Clientbrowser gesendet.
  - Im Clientbrowser muss die Cookie-Unterstützung aktiviert sein.
  - Die Cookie-Implementierung bietet eine bessere Leistung.
  - Es gibt Einschränkungen bezüglich der Konfiguration von Back-End-Servern. Die Cookie-Implementierung kann nur verwendet werden, wenn ein Client in einer Sitzung nur auf einen Back-End-Server zugreift.

Anmerkung: Für die Verwendung von JunctionRewrite mit der Option Use-Cookie ist eine Einschränkung zu beachten. Die URLs aller Anforderungen werden falsch umgesetzt, obwohl das Cookie nur für ein Unterverzeichnis des Hosts gilt. Mit den folgenden beiden Methoden können Sie dafür sorgen, dass URLs im Stammverzeichnis, die keine Junction erfordern, ordnungsgemäß bearbeitet werden:

- Kopieren Sie die Proxy-Regeln in der Datei ibmproxy.conf vor die Anweisung JunctionRewrite. (Alle Proxy-Regeln, die vor der Anweisung JunctionRewrite stehen, werden nicht umgeschrieben.)
- Ordnen Sie jeden URL explizit zu, d. h. verwenden Sie keine Platzhalterzeichen (*). Beispiel:

```
Proxy /no-juntion.jpg  http://login-server/no-junction.jpg
```

Beispiel-Plug-in Transmogrifier zur Erweiterung der Funktionalität von JunctionRewrite

Es wir ein anpassbarer Beispielcode bereitgestellt, der JavaScript™- (SCRIPT) und Applet-Tag-Blöcke (APPLET) in HTML-Dateien umschreibt und syntaktisch analysiert. Das Plug-in JunctionRewrite ist allein nicht in der Lage, die Ressourcenverknüpfungen in JavaScript oder in Parameterwerten von Java™ zu verarbeiten.

Nach der Installation von Caching Proxy können Sie denselben Code kompilieren und für die Ausführung mit JunctionRewrite konfigurieren.

Die folgenden Beispieldateien sind im Unterverzeichnis ...samples/cp/ des Verzeichnisses gespeichert, in das Sie den Fixpack heruntergeladen haben.

- Makefile (Makefile für dieses Beispiel-Plug-in)
- junctionRewrite2.h (Interface für angepasste Parser-Steuerroutine)
- junctionRewrite2.c (Implementierung für oben genanntes Interface)
- scriptHandler.c (Beispielsteuerroutine für das Umschreiben von JavaScript)
- appletHandler.c (Beispielsteuerroutine für Applet-Blöcke)
- junctionRewrite2.def (Plug-in-Definitionsdatei für Windows)
- junctionRewrite2.exp (Plug-in-Exportdatei für Linux und UNIX)

Kapitel 11. Bereitstellung lokaler Inhalte verwalten

Die Zuordnungsregeln Pass und Exec werden verwendet, um einem anfordernden Client lokale Inhalte bereitzustellen. Standardmäßig wird eine Pass-Regel mit einer Schablone mit Platzhalterzeichen als letzte Zuordnungsregel angegeben. Diese Regel weist alle Anforderungen, die nicht mit vorherigen Schablonen übereinstimmen, an, Dateien aus einem Zielverzeichnis abzurufen, das im Allgemeinen als Dokumentstammverzeichnis bezeichnet wird.

Wenn ein URL empfangen wird, der keinen Dateinamen enthält, durchsucht Caching Proxy das angegebene Verzeichnis bzw., sofern kein Verzeichnis angegeben ist, das Dokumentstammverzeichnis nach einer Datei, die der Liste der Begrüßungsseiten entspricht, die in der Konfigurationsdatei angegeben sind. Wenn mehrere Begrüßungsseiten definiert sind, sucht der Proxy-Server die Seiten in der Reihenfolge, in der sie definiert worden sind. Die erste gefundene Begrüßungsseite wird bereitgestellt.

Die Server-Homepage ist die Webseite, die der Server standardmäßig bereitstellt, wenn er eine Anforderung empfängt, die nur den URL des Servers ohne Angabe eines Verzeichnisses oder eines Dateinamens enthält. Wie zuvor erläutert, erfordert die Standardzuordnungsregel mit Platzhalterzeichen, dass die Server-Homepage im Dokumentstammverzeichnis gespeichert ist und dass der Dateiname der Homepage einer definierten Begrüßungsseite entspricht.

Anmerkung: Einige Webbrowser verwenden den Begriff *Homepage* für die erste Seite, die der Browser beim Start lädt. In diesem Dokument wird der Begriff nur für die Server-Homepage verwendet.

Dieses Kapitel beschreibt, wie Sie das Dokumentstammverzeichnis und Begrüßungsseiten definieren.

Dokumentstammverzeichnis definieren

Folgende Verzeichnisse werden standardmäßig als Dokumentstammverzeichnisse verwendet:

- Unter Linux und UNIX: /opt/ibm/edge/cp/server_root/pub/*Sprache*/
- Unter Windows:
  Laufwerk:\Programme\IBM\edge\cp\server_root\pub*Sprache*\ bzw. das Verzeichnis, das Sie bei der Installation als HTML-Verzeichnis angegeben haben.

Zugehörige Anweisungen

Mit der folgenden Anweisung definieren Sie das Dokumentstammverzeichnis:

- „Pass - Die Schablone zum Akzeptieren von Anforderungen angeben” auf Seite 247

Nähere Informationen hierzu finden Sie in Kapitel 4, „Die Datei ibmproxy.conf manuell editieren”, auf Seite 13.

Konfigurations- und Verwaltungsformulare

Gehen Sie wie folgt vor, um das Dokumentstammverzeichnis mit den Konfigurations- und Verwaltungsformularen zu ändern:

1. Wählen Sie **Serverkonfiguration –> Anforderungsverarbeitung –> Routing von Anforderungen** aus.
2. Suchen Sie in der Tabelle für das Routing von Anforderungen die Zeile, die die Zeichenfolge /* (Schrägstrich Stern) in der Spalte **Anforderungsschablone** enthält. Dies ist das Dokumentstammverzeichnis. Klicken Sie im Feld **Index** unterhalb der Tabelle auf die Zahl, die in der Spalte **Index** für diese Zeile angegeben ist.
3. Klicken Sie auf **Ersetzen**.
4. Klicken Sie in der Dropdown-Liste **Aktion** auf **Pass**.
5. Geben Sie im Feld "URL-Anforderungsschablone" die Zeichenfolge /* ein.
6. Geben Sie das neue Dokumentstammverzeichnis im Feld **Ersetzungsdateipfad** ein.
7. Klicken Sie auf **Übergeben**.
8. Wenn Ihre Änderungen akzeptiert wurden, klicken Sie auf das Symbol für **Serverneustart** (|) im oberen Rahmen.

Nach dem Neustart verwendet der Server das neue Dokumentstammverzeichnis.

Nähere Informationen hierzu finden Sie in Kapitel 2, „Konfigurations- und Verwaltungsformulare verwenden", auf Seite 7.

Standardbegrüßungsseiten definieren

Der Server sucht im Dokumentstammverzeichnis nach der Homepage. Welche Datei jedoch zurückgegeben wird, bestimmt die Liste der Begrüßungsseiten.

Informationen zu Begrüßungsseiten

Wenn der Server eine URL-Anforderung empfängt, die keinen Dateinamen enthält, versucht er, die Anforderung anhand einer Liste von Begrüßungsseiten zu bearbeiten, die in der Konfigurationsdatei des Servers definiert ist. Diese Liste definiert die Dateien, die als Standard-Homepages zu verwenden sind. Der Server ermittelt Ihre Homepage, indem er die Liste der Begrüßungsseiten mit den Dateien in Ihrem Dokumentstammverzeichnis auf Entsprechungen abgleicht. Die erste gefundene Übereinstimmung ist die Datei, die als Homepage zurückgegeben wird. Wird keine Übereinstimmung gefunden, zeigt der Server eine Auflistung der Dokumente im Dokumentstammverzeichnis an.

Soll eine bestimmte Datei als Homepage des Servers verwendet und zurückgegeben werden, wenn eine Anforderung kein Verzeichnis oder keinen Dateinamen enthält, müssen Sie die Datei im Dokumentstammverzeichnis speichern und sicherstellen, dass ihr Name mit einem in der Liste der Begrüßungsseiten gespeicherten Dateinamen übereinstimmt.

Die Standardkonfigurationsdatei legt fest, dass folgende Dateinamen in der angegebenen Reihenfolge als Begrüßungsseiten verwendet werden sollen:

1. welcome.html oder welcome.htm
2. index.html oder index.htm
3. Frntpage.html

Der Server gibt die erste Datei zurück, die mit einem Dateinamen in der Liste übereinstimmt. Solange Sie keine Datei mit dem Namen welcome.html oder index.html erstellt und im Dokumentstammverzeichnis gespeichert haben, verwendet der Server die Datei Frntpage.html als Homepage.

Wenn Sie beispielsweise die Standardkonfiguration verwenden und Ihr Dokumentstammverzeichnis keine Datei welcome.html, aber Dateien mit den Namen index.html und FrntPage.html, wird die Datei index.html als Homepage verwendet.

Falls der Server keine Homepage findet, wird der Inhalt des Dokumentstammverzeichnisses als Verzeichnis angezeigt.

Zugehörige Anweisungen

Mit der folgenden Anweisung definieren Sie die Begrüßungsseiten:

- „Welcome - Die Namen von Begrüßungsdateien angeben" auf Seite 287

Nähere Informationen hierzu finden Sie in Kapitel 4, „Die Datei ibmproxy.conf manuell editieren", auf Seite 13.

Konfigurations- und Verwaltungsformulare

Mit den folgenden Konfigurations- und Verwaltungsformularen definieren Sie die Begrüßungsseiten:

- **Serverkonfiguration –> Verzeichnisse und Begrüßungsseite –> Begrüßungsseite**

Nähere Informationen hierzu finden Sie in Kapitel 2, „Konfigurations- und Verwaltungsformulare verwenden", auf Seite 7.

Kapitel 12. FTP-Verbindungen verwalten

Caching Proxy leitet Anforderungen für FTP-URLs an den entsprechenden FTP-Server weiter, kann jedoch nicht dazu verwendet werden, Anforderungen von einem FTP-Client weiterzuleiten. Caching Proxy unterstützt nur FTP-Anforderungen, die von einem HTTP-Client (mit dem Protokollschema ftp://) empfangen werden.

Es werden nur die Methoden GET, PUT und DELETE für Anforderungen für FTP-Dateien unterstützt. Für Anforderungen, die sich auf FTP-Verzeichnislisten beziehen, wird nur die Methode GET unterstützt. Standardmäßig sind die Methoden PUT und DELETE in Caching Proxy inaktiviert. Nähere Informationen hierzu finden Sie im Abschnitt „HTTP/FTP-Methoden aktivieren" auf Seite 43.

Dieses Kapitel beschreibt, wie Sie FTP-Dateien schützen und Anmeldungen am FTP-Server, Verzeichnispfade und Verkettung verwalten.

FTP-Dateien schützen

Wenn Sie die Methode PUT zum Hochladen von FTP-Dateien oder die Methode DELETE zum Löschen von FTP-Dateien aktiviert haben, müssen Sie den FTP-Proxy-Zugriffsschutz zumindest für die Anforderungen PUT und DELETE definieren, um zu verhindern, dass unberechtigte Benutzer die Dateien auf Ihrem FTP-Server aktualisieren.

Zum Festlegen des Zugriffsschutzes für FTP-Proxy-Anforderungen wählen Sie in den Konfigurations- und Verwaltungsformularen **Serverkonfiguration –> Zugriffsschutz für Dokumente** aus. Um eine Zugriffsschutzkonfiguration für FTP-Dateianforderungen zu erstellen, muss die Anforderungsschablone mit `ftp://` beginnen. Verwenden Sie z. B. zum Schutz der Dateien im Verzeichnis exams die Schablone `ftp://exams/*`.

Nähere Informationen zum Erstellen von Zugriffsschutzkonfigurationen finden Sie in Kapitel 25, „Zugriffsschutzkonfigurationen für den Server", auf Seite 115.

Anmeldungen am FTP-Server verwalten

Falls im Anforderungs-URL keine Benutzer-ID und kein Kennwort angegeben sind, versucht Caching Proxy, sich anonym beim angefragten FTP-Server anzumelden (mit der Benutzer-ID ANONYMOUS). Viele FTP-Server erfordern die Angabe einer E-Mail-Adresse als Kennwort für die anonyme FTP-Anmeldung. Wenn der FTP-Server nach einem Kennwort für die anonyme Anmeldung fragt, sendet Caching Proxy die E-Mail-Adresse, die in der Anweisung WebmasterEmail in der Konfigurationsdatei angegeben ist.

Zum Festlegen der E-Mail-Adresse des Webmaster wählen Sie in den Konfigurations- und Verwaltungsformularen **Serverkonfiguration –> Systemverwaltung –> SNMP-MIB** aus. Die E-Mail-Adresse kann auch mit der Anweisung WebmasterEmail festgelegt werden.

Einzelheiten hierzu finden Sie im Abschnitt „WebMasterEMail - Eine E-Mail-Adresse für den Empfang von ausgewählten Serverberichten festlegen" auf Seite 287.

Wenn der FTP-Server im Anforderungs-URL eine bestimmte Benutzer-ID und ein bestimmtes Kennwort für die Anmeldung erfordert, kann der Benutzer die Benutzer-ID und das Kennwort im Anforderungs-URL eingeben. Beispiel:
`ftp://Benutzer-ID:Kennwort@ftpserverhost/`

Wenn Sie das Kennwort für die FTP-Benutzer-ID nicht im Anforderungs-URL angeben möchten, kann der Benutzer auch nur die Benutzer-ID in die URL-Adresse aufnehmen: `ftp://Benutzer-ID@ftpserverhost`. Caching Proxy versucht zuerst, sich mit der angegebenen Benutzer-ID und ohne Kennwort beim FTP-Server anzumelden. Sollte die Anmeldung ohne Kennwort nicht erfolgreich sein, fordert der Browser die Eingabe des Kennworts für die angegebene Benutzer-ID an.

Für alle anderen Anmeldungen als die anonyme Anmeldung muss mindestens die Benutzer-ID im URL angegeben werden. Wenn die Benutzer-ID nicht angegeben ist, wird versucht, eine anonyme Anmeldung durchzuführen, und der Client wird nicht zur Eingabe der Benutzer-ID aufgefordert.

FTP-Verzeichnispfade verwalten

Sie müssen Caching Proxy mitteilen, ob die Pfadnamen in FTP-URLs relativ zum Arbeitsverzeichnis des Benutzers oder relativ zum Stammverzeichnis interpretiert werden sollen. Wenn ein Benutzer, der an einem FTP-Server angemeldet ist, beispielsweise ein Standardarbeitsverzeichnis mit dem Namen /export/home/user1 hat und eine Datei mit dem Namen test1.exe aus einem Unterverzeichnis mit dem Namen test abrufen möchte, verwendet der Proxy-Server, abhängig davon, wie die FTP-URLs zu interpretieren sind, die folgenden URLs für den Abruf der Datei vom FTP-Server:

- Wenn *absolute* Pfadnamen festgelegt wurden, verwendet der FTP-Server folgenden URL: `ftp://user1:user1pw@FTPhost/export/home/user1/test/test1.exe`
- Wenn *relative* Pfadnamen festgelegt wurden, verwendet der FTP-Server folgenden URL: `ftp://user1:user1pw@FTPhost/test/test1.exe`

Wenn relative FTP-URL-Pfade festgelegt wurden, können Benutzer trotzdem einen absoluten Pfadnamen angeben, indem sie der Konvention folgend als erstes Zeichen `%2F` anstelle des Anfangsschrägstrichs (/) für das Stammverzeichnis angeben. Falls user1, der das Arbeitsverzeichnis /export/home/user1 hat, beispielsweise auf eine Datei im Arbeitsverzeichnis von user2, /export/home/user2, zugreifen möchte, wird die Anforderung `ftp://user1:user1pw@FTPhost/%2Fexport/home/user2/Datei` ordnungsgemäß als relativer URL zum Stammverzeichnis / (d. h. als absoluter Pfadname) interpretiert, auch wenn relative FTP-URL-Pfadnamen ausgewählt wurden.

Sie können die Interpretation von FTP-URL-Pfaden im Konfigurations- und Verwaltungsformular **Proxy-Konfiguration –> Leistung des Proxy-Servers** definieren. Wählen Sie im unteren Teil des Formulars unter **Pfade der FTP URL:** entweder **Absolute Pfade** aus, wenn Sie das Stammverzeichnis des Servers angeben möchten, oder wählen Sie **Relative Pfade** aus, wenn Sie das Arbeitsverzeichnis des Benutzers als Pfadanfang festlegen möchten.

Sie können diese Einstellung auch in der Konfigurationsdatei des Proxy-Servers ändern. Nähere Informationen hierzu finden Sie im Abschnitt „FTPUrlPath - Angeben, wie FTP-URLs interpretiert werden sollen” auf Seite 221.

FTP-Verkettung verwalten

Wenn Sie mehrere Web-Proxy-Server verketten, können Sie festlegen, dass Anforderungen, die FTP-URLs enthalten, an einen verketteten Web-Proxy-Server und nicht direkt an den FTP-Server gesendet werden. Wählen Sie in den Konfigurations- und Verwaltungsformularen **Proxy-Konfiguration –> Proxy-Kettung und Nicht-Proxy-Domänen** aus, wenn Sie einen verketteten Proxy-Server für FTP-Anforderungen festlegen möchten. Für die Angabe des URL eines verketteten Proxy-Servers wird das Protokollschema http:// verwendet, auch wenn Anforderungen für ein Protokollschema ftp:// verkettet werden.

Informationen zur Konfiguration der FTP-Verkettung in der Konfigurationsdatei des Proxy-Servers finden Sie im Abschnitt „ftp_proxy - Für FTP-Anforderungen einen anderen Proxy-Server angeben” auf Seite 221.

Kapitel 13. Serververarbeitung anpassen

In diesem Kapitel wird beschrieben, wie Sie mit SSI-Anweisungen (Server-Side Includes) Informationen in CGI-Programme und HTML-Dokumente einfügen, die an einen Client übermittelt werden. Außerdem werden die Anpassung der Fehlernachrichten des Servers sowie die Zuordnung von Ressourcen beschrieben.

Server-Side Includes

Mit Server-Side Includes können Sie CGI-Programmen und HTML-Dokumenten Informationen hinzufügen, die der Server an den Client sendet, wenn er als Ursprungsserver auftritt (weitergeleitete und zwischengespeicherte Objekte sind nicht betroffen). Das aktuelle Datum, die Dateigröße und das Datum der letzten Änderung einer Datei sind Beispiele für die Art von Informationen, die an den Client gesendet werden können. Dieser Abschnitt beschreibt das Befehlsformat für Server-Side Includes und erläutert die Schritte, die erforderlich sind, damit Server-Side-Include-Befehle in CGI-Programmen und HTML-Dokumenten funktionieren. Sie können Server-Side Includes auch zum Anpassen von Fehlerseiten verwenden.

Hinweise zu Server-Side Includes

Bevor Sie Server-Side Includes auf Ihrem Server verwenden, sollten Sie folgende Aspekte bezüglich Leistung, Sicherheit und Risiken bedenken:

- Die Leistung kann erheblich beeinträchtigt werden, wenn der Server Dateien verarbeitet, während er sie sendet.
- Die Sicherheit ist gefährdert, wenn Sie den Standardbenutzern die Ausführung von Befehlen auf Ihrem Server erlauben. Gehen Sie bei der Auswahl der Verzeichnisse, in denen Server-Side Includes und der Befehl **exec** abgelegt werden, sorgfältig vor. Das Sicherheitsrisiko ist nur sehr gering, wenn Sie den Befehl **exec** nicht aktivieren.
- Bei Verwendung von Server-Side Includes können Probleme auftreten. Beispielsweise sind rekursive Verweise auf Dateien nicht möglich. Wenn Sie z. B. die Datei sleepy.html ausführen und das Programm `<-- !#include file="sleepy.html" -->` findet, erkennt der Server den Fehler nicht und schlägt fehl. (Die Verwendung nicht rekursiver Dateiverweise in anderen Dateien ist kein Problem.)

Server-Side Includes konfigurieren

Wählen Sie in den Konfigurations- und Verwaltungsformularen **Serverkonfiguration –> Basiseinstellungen** aus, um die Unterstützung für Server-Side Includes zu aktivieren. In diesem Formular können Sie angeben, welche der folgenden Arten von Server-Side Includes akzeptiert werden:

- CGI-Scripts,
- Dateien,
- alle außer CGI-Scripts, die den Befehl **exec** verwenden,
- keine.

Außerdem können Sie in diesem Formular festlegen, ob die Verarbeitung von Server-Side Includes zusätzlich zu anderen Dateitypen auch für Text- und HTML-Dokumente vorgenommen werden soll.

Stellen Sie außerdem sicher, dass die Dateierweiterung, die Sie für Server-Side Includes verwenden, erkannt wird. Wählen Sie in den Konfigurations- und Verwaltungsformularen **Serverkonfiguration –> MIME-Typen und -Codierung** aus und rufen Sie das Formular **MIME-Typen** auf. Standardmäßig werden die Erweiterungen shtml und htmls erkannt.

Die folgenden Referenzabschnitte beschreiben, wie Sie durch Ändern der Anweisungen in der Konfigurationsdatei Ihren Proxy-Server für die Unterstützung von Server-Side Includes konfigurieren:

- „AddType - Datentyp für Dateien mit bestimmten Suffixen angeben" auf Seite 184
- „imbeds - Angeben, ob Server-Side Includes verarbeitet werden" auf Seite 227

Format für Server-Side Includes

Include-Befehle müssen in Form von Kommentaren in das HTML-Dokument oder CGI-Programm eingefügt werden. Die Befehle müssen das folgende Format haben:

```
<!--#Anweisung Tag=Wert ... -->
oder
<!--#Anweisung Tag="Wert" ... -->
```

Die Werte müssen nur dann in Anführungszeichen gesetzt werden, wenn sie Leerzeichen enthalten.

Anweisungen für Server-Side Includes

In diesem Abschnitt werden die Anweisungen beschrieben, die der Server für Server-Side Includes akzeptiert. (Verwechseln Sie diese Anweisungen nicht mit den Anweisungen für die Konfigurationsdatei des Proxy-Servers, die in Anhang B, „Anweisungen in der Konfigurationsdatei", auf Seite 175 beschrieben werden.)

config - Dateiverarbeitung steuern

Mit dieser Anweisung können Sie bestimmte Aspekte der Dateiverarbeitung steuern. Die gültigen Tags sind `cmntmsg`, `errmsg`, `sizefmt` und `timefmt`.

cmntmsg

Mit diesem Tag können Sie eine Nachricht angeben, die den Kommentaren vorangestellt werden soll, die von anderen Anweisungen hinzugefügt werden. Wenn eine Anweisung Text zwischen der Anweisungsspezifikation und `"-->"` enthält, wird dieser Text als Kommentar behandelt und der Datei hinzugefügt, die der Server an den Client sendet.

Beispiel:

```
<!--#config cmntmsg="[Dies ist ein Kommentar]" -->
<!-- #echo var="  " zusätzlicher Text -->
```

Ergebnis: `<!--[Dies ist ein Kommentar] zusätzlicher Text -->`

Standard: `[Folgendes war zusätzlich in der Anweisung]`

errmsg

Mit diesem Tag können Sie die Nachricht angeben, die an den Client gesendet werden soll, wenn beim Verarbeiten einer Datei ein Fehler auftritt. Die Nachricht wird im Fehlerprotokoll des Servers aufgezeichnet.

Beispiel:

```
<!-- #config errmsg="[Es ist ein Fehler aufgetreten]" -->
```

Standard: `"[Fehler beim Verarbeiten dieser Anweisung]"`

sizefmt

Mit diesem Tag können Sie das Format angeben, in dem die Dateigröße angezeigt wird. In den folgenden Beispielen ist `bytes` der Wert, der zum Anzeigen der Anzahl Bytes verwendet wird, und `abbrev` der Wert, der zum Anzeigen der Anzahl Kilobytes bzw. Megabytes verwendet wird.

Beispiel 1:

```
<!--#config sizefmt=bytes -->
<!--#fsize file=foo.html -->
```

Ergebnis: `1024`

Beispiel 2:

```
<!--#config sizefmt=abbrev -->
<!--#fsize file=foo.html -->
```

Ergebnis: `1 KB`

Standard: `"abbrev"`

timefmt

Mit diesem Tag können Sie das Format festlegen, in dem Datums- und Zeitangaben angezeigt werden.

Beispiel:

```
<!--#config timefmt="%D %T" -->
<!--#flastmod file=foo.html -->
```

Ergebnis: `"18/10/95 12:05:33"`

Standard: `"%a, %d %b %Y %T %Z"`

Die folgenden Formate für `strftime()` können für das Tag `timefmt` verwendet werden:

Kennung	Bedeutung
%%	Durch % ersetzen
%a	Durch den abgekürzten Namen des Wochentags ersetzen
%A	Durch den vollständigen Namen des Wochentags ersetzen
%b	Durch den abgekürzten Namen des Monats ersetzen
%B	Durch den vollständigen Namen des Monats ersetzen
%c	Durch Datum und Uhrzeit ersetzen
%C	Durch die Jahreszahl ersetzen (Jahr dividiert durch 100 und abgeschnitten)
%d	Durch den Tag des Monats ersetzen (01-31)
%D	Datum im Format *%m*/*%t*/*%j* einfügen
%e	Monat als Dezimalzahl einfügen (01-12) (unter C POSIX ist dies ein zweistelliges, rechtsbündiges Feld mit Leerzeichen)

Kennung	Bedeutung
%E[cCxyY]	Wenn das alternative Datum/Uhrzeit-Format nicht verfügbar ist, werden die %E-Deskriptoren ihren nicht erweiterten Entsprechungen zugeordnet (%EC wird z. B. %C zugeordnet).
%Ec	Durch die alternative Darstellung von Datum und Uhrzeit ersetzen
%EC	Durch den Namen des Basisjahrs (Zeitraum) in der alternativen Darstellung ersetzen
%Ex	Durch die alternative Darstellung des Datums ersetzen
%EX	Durch die alternative Darstellung der Uhrzeit ersetzen
%Ey	Durch die Jahreszahl ohne Jahrhundert (%EC) in der alternativen Darstellung ersetzen
%EY	Durch die vollständige alternative Jahresdarstellung ersetzen
%h	Durch den abgekürzten Namen des Monats ersetzen (wie %b)
%H	Durch die Stunde (23-Stunden-Format) in Dezimalformat (00-23) ersetzen
%I	Durch die Stunde (12-Stunden-Format) in Dezimalformat (00-12) ersetzen
%j	Durch den Tag des Jahres (001-366) ersetzen
%m	Durch den Monat (01-12) ersetzen
%M	Durch die Minute (00-59) ersetzen
%n	Durch eine neue Zeile ersetzen
%O[deHlmMSUwWy]	Wenn das alternative Datum/Uhrzeit-Format nicht verfügbar ist, werden die %E-Deskriptoren ihren nicht erweiterten Entsprechungen zugeordnet (%Od wird z. B. %d zugeordnet).
%Od	Durch den Tag des Monats unter Verwendung der alternativen numerischen Symbole, nach Bedarf aufgefüllt mit führenden Nullen ersetzen, falls es ein alternatives Symbol für Null gibt, andernfalls mit führenden Leerzeichen
%Oe	Durch den Tag des Monats unter Verwendung der alternativen numerischen Symbole, nach Bedarf aufgefüllt mit führenden Leerzeichen ersetzen
%OH	Durch die Stunde (24-Stunden-Format) unter Verwendung der alternativen numerischen Symbole ersetzen
%OI	Durch die Stunde (12-Stunden-Format) unter Verwendung der alternativen numerischen Symbole ersetzen
%Om	Durch den Monat unter Verwendung der alternativen numerischen Symbole ersetzen
%OM	Durch die Minuten unter Verwendung der alternativen numerischen Symbole ersetzen
%OS	Durch die Sekunden unter Verwendung der alternativen numerischen Symbole ersetzen
%OU	Durch die Kalenderwoche (Sonntag als erster Tag der Woche, Regeln wie bei %U) unter Verwendung der alternativen numerischen Symbole ersetzen
%Ow	Durch den Wochentag (Sonntag=0) unter Verwendung der alternativen numerischen Symbole ersetzen

Kennung	Bedeutung
%OW	Durch die Kalenderwoche (Montag als erster Tag der Woche) unter Verwendung der alternativen numerischen Symbole ersetzen
%Oy	Durch das Jahr (ohne Jahrhundert (%C)) in der alternativen Darstellung und unter Verwendung der alternativen numerischen Symbole ersetzen
%p	Durch die lokale Entsprechung für AM und PM ersetzen
%r	Durch die Zeichenfolgeentsprechung für %I:%M:%S %p ersetzen
%R	Durch die Uhrzeit im 24-Stunden-Format (%H:%M) ersetzen
%S	Durch die Sekunden (00-61) ersetzen
%t	Durch einen Tabulator ersetzen
%T	Durch die Zeichenfolgeentsprechung für %H:%M:%S ersetzen
%u	Durch den Wochentag im Dezimalformat (1-7) ersetzen, wobei die 1 für Montag steht
%U	Durch die Kalenderwoche (00-53) ersetzen, wobei Sonntag der erste Tag der Woche ist
%V	Durch die Kalenderwoche (01-53) ersetzen, wobei Montag der erste Tag der Woche ist
%w	Durch den Wochentag (0-6) ersetzen, wobei die 0 für Sonntag steht
%W	Durch die Kalenderwoche (00-53) ersetzen, wobei Montag der erste Tag der Woche ist
%x	Durch die entsprechende Datumsdarstellung ersetzen
%X	Durch die entsprechende Uhrzeitdarstellung ersetzen
%y	Durch die zweistellige Jahresangabe (ohne Jahrhundert) ersetzen
%Y	Durch die vollständige vierstellige Jahresangabe ersetzen
%Z	Durch den Namen der Zeitzone bzw. eine leere Zeichenfolge ersetzen, falls die Zeitzone unbekannt ist

Die Konfiguration des Betriebssystems legt die vollständigen und abgekürzten Angaben für Monate und Jahre fest.

echo - Variablenwerte anzeigen

Mit dieser Anweisung können Sie den Wert von Umgebungsvariablen anzeigen, die mit dem Tag `var` angegeben werden. Wenn eine Variable nicht gefunden wurde, wird (`None`) angezeigt. Außerdem können Sie mit **echo** einen Wert anzeigen, der mit der Anweisung **set** oder **global** festgelegt wurde. Folgende Umgebungsvariablen können angezeigt werden:

DATE_GMT
Das aktuelle Datum und die Uhrzeit in westeuropäischer Zeit (GMT). Das Format dieser Variablen wird mit der Anweisung **config timefmt** festgelegt.

DATE_LOCAL
Das aktuelle Datum und die Ortszeit. Das Format dieser Variablen wird mit der Anweisung **config timefmt** festgelegt.

DOCUMENT_NAME
Der Name des Ausgangsdokuments. Falls das HTML-Dokument von einem CGI-Programm generiert wurde, enthält diese Variable den Namen des CGI-Programms.

DOCUMENT_URI
Der vollständige, vom Client angeforderte URL ohne die Abfragezeichenfolge.

LAST_MODIFIED
Das Datum und die Uhrzeit der letzten Änderung des aktuellen Dokuments. Das Format dieser Variablen wird mit der Anweisung **config timefmt** festgelegt.

QUERY_STRING_UNESCAPED
Die vom Client gesendete Suchabfrage (nicht definiert, sofern ein HTML-Dokument nicht von einem CGI-Programm generiert wurde).

SSI_DIR
Der Pfad der aktuellen Datei, relativ zu SSI_ROOT. Wenn die aktuelle Datei in SSI_ROOT gespeichert ist, lautet dieser Wert "/".

SSI_FILE
Der Dateiname der aktuellen Datei.

SSI_INCLUDE
Der Wert in dem include-Befehl, mit dem die aktuelle Datei abgerufen wurde. Diese Variable ist für die Ausgangsdatei nicht definiert.

SSI_PARENT
Der Pfad und Dateiname der Datei, die den include-Befehl enthält, mit dem die aktuelle Datei abgerufen wurde, relativ zu SSI_ROOT.

SSI_ROOT
Der Pfad der Ausgangsdatei. Alle include-Anforderungen müssen sich in diesem Verzeichnis oder in einem seiner Unterverzeichnisse befinden.

Beispiel:

```
<!--#echo var=SSI_DIR -->
```

exec - CGI-Programme angeben

Mit dieser Anweisung können Sie die Ausgabe eines CGI-Programms einfügen. Die Anweisung exec verwirft alle HTTP-Header, die vom CGI-Programm ausgegeben werden, *mit Ausnahme* der folgenden:

Content-type
Legt fest, ob der Hauptteil der Ausgabe für andere Includes syntaktisch analysiert werden soll.

Content-encoding
Legt fest, ob die Umsetzung von EBCDIC in ASCII durchgeführt werden muss.

Last-modified
Ersetzt den aktuellen Wert des Header "last-modified", sofern der aktuelle Wert nicht größer ist als der angegebene Wert.

cgi - URL-Adresse des CGI-Programms angeben

Mit dieser Anweisung können Sie den URL eines CGI-Programms angeben.

In diesem Beispiel ist **program** das CGI-Programm, das ausgeführt werden soll. **path_info** und **query_string** stehen für einen oder mehrere Parameter, die als Umgebungsvariablen an das Programm übergeben werden.

```
<!--#exec cgi="/cgi-bin/program/path_info?query_string" -->
```

Dieses Beispiel zeigt die Verwendung der Variablen:

```
<!--#exec cgi="&path;&cgiprog;&pathinfo;&querystring;" -->
```

flastmod - Datum und Uhrzeit der letzten Dokumentänderung anzeigen

Mit dieser Anweisung können Sie Datum und Uhrzeit der letzten Dokumentänderung anzeigen. Das Format dieser Variablen wird mit der Anweisung **config timefmt** festgelegt. Die gültigen Tags für diese Anweisung sind **file** und **virtual**, die im Folgenden beschrieben werden.

Anweisungsformate:

```
<!--#flastmod file="/Pfad/Datei" -->
<!--#flastmod virtual="/Pfad/Datei" -->
```

file Mit diesem Tag können Sie den Namen einer Datei angeben. Bei **flastmod**, **fsize** und **include** wird **file** relativ zu **SSI_ROOT** interpretiert, wenn ein Schrägstrich (**'/'**) vorangestellt ist. Andernfalls wird das Tag relativ zu **SSI_DIR** interpretiert. Die angegebene Datei muss im Verzeichnis **SSI_ROOT** oder in einem seiner untergeordneten Verzeichnisse gespeichert sein. Beispiel:

```
<!--#flastmod file="/Pfad/Datei" -->
```

virtual
Mit diesem Tag können Sie die URL-Adresse des virtuellen Pfads zu einem Dokument angeben. Bei **flastmod**, **fsize** und **include** wird **virtual** immer durch Zuordnungsanweisungen des Servers übergeben. Beispiel:

```
<!--#flastmod virtual="/Pfad/Datei" -->
```

Beispiel:

```
<!--#flastmod file="foo.html" -->
```

Ergebnis: `12May96`

fsize - Dateigröße anzeigen

Mit dieser Anweisung können Sie die Größe der angegebenen Datei anzeigen. Das Format dieser Variablen wird mit der Anweisung **config sizefmt** festgelegt. Die gültigen Tags für diese Anweisung sind **file** und **virtual**, die im Abschnitt zur Anweisung **flastmod** beschrieben sind.

Beispiel:

```
<!--#fsize file="/Pfad/Datei" -->
<!--#fsize virtual="/Pfad/Datei" -->
```

Ergebnis: `1K`

global - Globale Variablen definieren

Mit dieser Anweisung können Sie globale Variablen definieren, die zu einem späteren Zeitpunkt von dieser Datei oder anderen eingefügten Dateien zurückgemeldet werden können.

Beispiel:

```
<!--#global var=Variablenname value="einWert" -->
```

Wenn Sie beispielsweise auf ein übergeordnetes Dokument außerhalb der virtuellen Grenzen verweisen möchten, müssen Sie die globale Variable DOCUMENT_URI definieren. Außerdem müssen Sie im untergeordneten Dokument auf die globale Variable verweisen. Dieses Beispiel zeigt die HTML-Codierung, die Sie in das übergeordnete Dokument einfügen müssen:

```
<!--#global var="PARENT_URI" value=&DOCUMENT_URI; -->
```

Dieses Beispiel zeigt die HTML-Codierung, die Sie in das untergeordnete Dokument einfügen müssen:

```
<!--#flastmod virtual=&PARENT_URI; -->
```

include - Ein Dokument in die Ausgabe einfügen

Mit dieser Anweisung können Sie den Text eines Dokuments in die Ausgabe einfügen. Die gültigen Tags für diese Anweisung sind **file** und **virtual**, die im Abschnitt zur Anweisung **flastmod** beschrieben werden.

set - Zurückzumeldende Variablen festlegen

Mit dieser Anweisung können Sie eine Variable definieren, die zu einem späteren Zeitpunkt von dieser Datei (ausschließlich) zurückgemeldet werden kann.

Beispiel:

```
 <!--#set var="Variable 2" value="AndererWert" -->
```

Wenn Sie eine Anweisung definieren, können Sie innerhalb von `value` eine Zeichenfolge zurückgeben. Beispiel:

```
<!--#include file="&Dateiname;" -->
```

Variablen: Einer auf Serverseite definierten Anweisung set folgt im Allgemeinen eine Anweisung echo, d. h. die definierte Variable wird gesucht und, sofern möglich, zurückgemeldet. Anschließend wird die Ausführung der Funktion fortgesetzt. Die Anweisung kann mehrere Verweise auf Variablen enthalten. Außerdem können Sie mit set-Anweisungen auf Serverseite eine bereits definierte Variable zurückmelden. Falls keine definierte Variable gefunden wird, wird nichts angezeigt.

Wenn eine auf Serverseite definierte Anweisung set in einer SSI-Anweisung einen Variablenverweis findet, versucht sie, diesen Verweis auf *Serverseite* aufzulösen. In der zweiten Zeile des folgenden Beispiels bildet die Servervariable `&index;` zusammen mit der Zeichenfolge `var` den Variablennamen `var1`. Anschließend wird der Variablen `&var1;` ein Wert zugeordnet, indem dem Et-Zeichen (&) im Ausdruck `&ecirc;` ein Escape-Zeichen vorangestellt wird, damit dieser Ausdruck nicht als Variable interpretiert wird. Er wird vielmehr als Zeichenfolge verwendet, um den Wert `fr&ecirc;d` bzw. *fred* mit einem Zirkumflex über dem e zu bilden. Die Variable `&ecirc;` ist eine clientseitige Variable.

```
<!--#set var="index" value="1" -->
<!--#set var="var&index;" value="fr\&ecirc;d" -->
<!--#echo var="var1" -->
```

Die folgenden Zeichen können durch Voranstellen eine Backslash (\) geschützt werden:

Zeichen	Bedeutung
\a	Alert (Signalton)
\b	Rückschritt
\f	Formularvorschub (Seitenvorschub)
\n	Neue Zeile
\r	Zeilenschaltung
\t	Horizontaltabulator
\v	Vertikaltabulator
\'	Einfaches Anführungszeichen
\"	Doppeltes Anführungszeichen
\?	Fragezeichen
\\	Umgekehrter Schrägstrich
\-	Silbentrennungsstrich
\.	Punkt
\&	Et-Zeichen

Fehlernachrichten anpassen

Sie können die von Caching Proxy zurückgegebenen Fehlernachrichten anpassen und für bestimmte Fehlerbedingungen spezielle Nachrichten definieren. Wählen Sie in den Konfigurations- und Verwaltungsformularen **Serverkonfiguration –> Anpassung von Fehlernachrichten** aus. In diesem Formular können Sie eine Fehlerbedingung auswählen und eine bestimmte HTML-Datei angeben, die für diese Bedingung verwendet werden soll.

Informationen zum Anpassen von Fehlermeldungen durch Editieren der Anweisungen in der Konfigurationsdatei des Proxy-Servers finden Sie im Abschnitt zur Anweisung „ErrorPage - Für eine bestimmte Fehlerbedingung eine angepasste Nachricht angeben” auf Seite 214.

RTSP-Umleitung (Real Time Streaming Protocol)

Mit RTSP Redirector, einem neuen Feature von WebSphere Application Server Version 6.0.1, wird jetzt das Streaming von Multimediadateien unterstützt. Mit RTSP ist Caching Proxy in der Lage, Kontakt zu Media-Playern herzustellen und deren Anforderungen an einen geeigneten Proxy-Server oder Content-Server weiterzuleiten, der die angeforderten Multimediainhalte bereitstellt.

RTSP (Real Time Streaming Protocol) ist in RFC 2326 definiert und ein Internet-Standardprotokoll für die Steuerung von Datenströmen. Obwohl RTSP keine Technologie zum *Senden* von Datenströmen beinhaltet, ist es so flexibel, dass es zur Steuerung von Datenströmen eingesetzt werden kann, die sich nicht auf die Video- oder Audiowiedergabe beschränken.

Informationen zur RTSP-Umleitung

Mit der Umleitungsfunktion von RTSP kann Caching Proxy Anforderungen für jede von RTSP gesteuerte Multimediasitzung umleiten, die Streaming unterstützt. Dies betrifft folgende Multimediatypen:

- RealNetworks-Audiodateien
- RealNetworks-Videodateien
- RealNetworks-Live-Streams (Audio und Video)
- Dateien von Microsoft Media Player
- Multimediadateien von Apple Quicktime

Jede Wiedergabeeinheit (Player), deren Konfiguration die Kontaktaufnahme zu einem Proxy-Server am RTSP-Port (in der Regel 554) unterstützt, kann dieses Framework in Caching Proxy verwenden, um ihre Anforderungen von RTSP Redirector bearbeiten zu lassen.

Multimediapräsentationen werden von RTSP Redirector weder zwischengespeichert noch direkt umgeleitet. Für die Unterstützung dieser beiden Funktionen muss RTSP Redirector zusammen mit einem Multimediaserver eines Fremdanbieters eingesetzt werden, der Streaming unterstützt. Wenn RTSP Redirector in Caching Proxy verwendet wird, muss der Zugriff auf einen oder mehrere RTSP-Proxy-Server über das Netz möglich sein.

Einschränkung von RTSP

Für diese Funktion gilt folgende Einschränkung:

Derzeit werden nur RealNetworks-Technologien unterstützt. Dazu gehören der RealProxy-Proxy-Server, der RealServer-Ursprungsserver sowie der RealPlayer.

Verbesserungen von RTSP

In früheren Versionen unterlag RTSP Redirector der Einschränkung, dass alle Anforderungen an denselben Ursprungsserver für alle URLs auf dieselbe Weise umgeleitet wurden. Eine Umleitung basierend auf Dateinamen oder anderen Teilen des angeforderten URL war nicht möglich. Diese Einschränkung gilt nicht mehr. RTSP Redirector verwendet jetzt den vollständigen URL aus den empfangenen Anforderungen zusammen mit dem Schwellenwert (rtsp_proxy_threshold), der in der Konfigurationsdatei von Caching Proxy festgelegt ist, um zu bestimmen, ob die Clientanforderungen an den Ursprungsserver oder an einen Proxy-Server umgeleitet werden. Anforderungen an denselben Ursprungsserver werden individuell behandelt.

RTSP-Umleitung konfigurieren

Mit den folgenden Anweisungen in der Konfigurationsdatei können Sie die RTSP-Umleitung steuern. Die Einstellungen für diese Anweisungen werden bei einem Neustart des Servers nicht aktualisiert. Sie müssen den Server vollständig stoppen und anschließend erneut starten, damit die an diesen Anweisungen vorgenommenen Änderungen wirksam werden.

- „RTSPEnable - RTSP-Umleitung aktivieren" auf Seite 272
- „rtsp_proxy_server - Server für die Umleitung angeben" auf Seite 272
- „rtsp_proxy_threshold - Anzahl Anforderungen vor der Umleitung in einen Cache angeben" auf Seite 272
- „rtsp_url_list_size - Anzahl URLs im Proxy-Speicher angeben" auf Seite 273

Kapitel 14. Header-Optionen konfigurieren

Beim Anfordern von Dokumenten senden Web-Clients Header, die zusätzliche Informationen zum Browser oder zur Anforderung enthalten. Header werden automatisch generiert, wenn eine Anforderung gesendet wird.

Caching Proxy unterstützt verschiedene Optionen zur Anpassung von Header-Informationen, damit diese dem Zielserver verborgen bleiben. Obwohl das Ersetzen des aktuellen Header durch einen allgemeineren Header die Anonymität der Clients erhöht, hat dies den Nachteil, dass dadurch Header-basierte Seitenanpassungen, die in manche Webseiten geschrieben werden, inaktiviert werden.

In der Regel haben Header das folgende Format:

```
User-Agent: Mozilla 2.02/OS2
Client-IP: 45.37.192.3
Referer: http://www.bigcompany.com/WebTrafficExpress/main.html
```

Dieser Header enthält die folgenden Felder:

- **User-Agent:** Enthält Informationen zu Browser und Betriebssystem.
- **Client-IP:** Enthält die IP-Adresse des Client, der den URL anfordert.
- **Herkunftsadresse:** Liefert dem Zielserver den URL des Link, der auf diese Seite verweist.

Die meisten Header können durch entsprechende Einstellungen in der Proxy-Konfiguration blockiert werden. Einige Header-Felder werden jedoch von den Ursprungsservern benötigt. Werden diese Felder blockiert, werden Webseiten daher nicht ordnungsgemäß angezeigt. (Beispielsweise können manchmal falsche Webseiten angezeigt werden, wenn das Header-Feld ″host″ blockiert ist.) Nähere Informationen zu den Header-Feldern finden Sie in der Spezifikation von HTTP Version 1.1.

Zugehörige Anweisungen

Informationen zum Ändern von Header-Optionen durch Editieren der Konfigurationsdatei des Proxy-Servers finden Sie in den Referenzabschnitten zu den folgenden Anweisungen:

- „NoProxyHeader - Die Client-Header angeben, die blockiert werden sollen” auf Seite 245
- „ProxyFrom - Einen Client mit einem Header des Typs From: angeben” auf Seite 264
- „ProxyIgnoreNoCache - Anforderung zum erneuten Laden ignorieren” auf Seite 264
- „ProxySendClientAddress - Den Header ″Client IP:″ generieren” auf Seite 265
- „ProxyUserAgent - Die Zeichenfolge ″User Agent″ ändern” auf Seite 265
- „ProxyVia - Format des HTTP-Headers angeben” auf Seite 266

Nähere Informationen hierzu finden Sie in Kapitel 4, „Die Datei ibmproxy.conf manuell editieren”, auf Seite 13.

Konfigurations- und Verwaltungsformulare

Die Header-Optionen können in zwei Konfigurations- und Verwaltungsformularen definiert werden:

- Wählen Sie **Proxy-Konfiguration –> Vertraulichkeitseinstellungen** aus. Definieren Sie im Formular **Vertraulichkeitseinstellungen** die folgenden Einstellungen:
  - **IP-Adresse des Client an Zielserver weiterleiten**

    Wählen Sie dieses Markierungsfeld aus, wenn die IP-Adresse des anfragenden Clients an den Zielserver (Inhaltsserver) weitergeleitet werden soll. Wenn Sie dieses Feld nicht auswählen, empfängt der Zielserver die IP-Adresse des Proxy-Servers. Wird dieses Feld nicht ausgewählt, erhöht sich außerdem die Anonymität des Client beim Surfen im Web.
  - **Benutzeragent-Zeichenfolge**

    Geben Sie die Zeichenfolge ein, die im Header an den Zielserver gesendet werden soll. Diese Zeichenfolge ersetzt den Browser-Typ und das Betriebssystem, die ein Client verwendet. Beispiel: Die Angabe `Caching Proxy4.0` ersetzt im folgenden Header `Mozilla 2.02/OS2`:

    ```
    Content-Type:MIME
    User-Agent: Mozilla 2.02/OS2
    Referer: http://www.ics.raleigh.ibm.com/WebTrafficExpress/main.html
    Pragma:no-cache
    ```
  - **Von:**

    Geben Sie die E-Mail-Adresse ein, die der Zielserver bei der Syntaxanalyse des Header "From:" liest. In der Regel wird die E-Mail-Adresse des Proxy-Administrators angegeben, da der Administrator die Berichte zu Problemen und Fehlern erhalten sollte.
  - Klicken Sie auf **Übergeben**, um die Änderungen in der Konfigurationsdatei vorzunehmen.
- Wählen Sie **Proxy-Konfiguration –> Filtern von Proxy-Headern** aus. In diesem Formular können Sie die HTTP-Header auflisten, die blockiert werden sollen:
  1. Klicken Sie auf **Hinzufügen** oder **Entfernen** und geben Sie eine Indexposition für den blockierten Header an.
  2. Geben Sie den Client-HTTP-Header ein, der blockiert werden soll. (Eine vollständige Liste und Beschreibungen der Header finden Sie in der Spezifikation von HTTP 1.1.)
  3. Klicken Sie auf **Übergeben**, um die Änderungen in der Konfigurationsdatei vorzunehmen.

Nähere Informationen hierzu finden Sie in Kapitel 2, „Konfigurations- und Verwaltungsformulare verwenden”, auf Seite 7.

Kapitel 15. Informationen zur Anwendungsprogrammierschnittstelle

Eine vollständige Beschreibung der Anwendungsprogrammierschnittstelle (API) finden Sie in der Veröffentlichung *Programming Guide for Edge Components*. API-Anweisungen in der Konfigurationsdatei aktivieren Plug-in-Routinen, die in bestimmten Schritten während der Anforderungsverarbeitung aufgerufen werden. Diese Plug-in-Routinen können integrierte Routinen ersetzen oder zusätzlich zu diesen ausgeführt werden.

Zugehörige Anweisungen

Im Folgenden sind die API-Anweisungen aufgeführt:

- „Authentication - Schritt ″Authentication″ anpassen″ auf Seite 187
- „Authorization - Schritt ″Authorization″ anpassen″ auf Seite 188
- „Error - Schritt ″Error″ anpassen″ auf Seite 213
- „Log - Schritt ″Log″ anpassen″ auf Seite 234
- „Midnight - API-Plug-in für die Archivierung von Protokollen angeben″ auf Seite 242
- „NameTrans - Schritt ″Name Translation″ anpassen″ auf Seite 243
- „ObjectType - Schritt ″Object Type″ anpassen″ auf Seite 246
- „PostAuth — Schritt ″PostAuth″ anpassen″ auf Seite 252
- „PostExit - Schritt ″PostExit″ anpassen″ auf Seite 252
- „PreExit - Schritt ″PreExit″ anpassen″ auf Seite 253
- „ServerInit - Schritt ″Server Initialization″ anpassen″ auf Seite 275
- „ServerTerm - Schritt ″Server Termination″ anpassen″ auf Seite 276
- „Service - Schritt ″Service″ anpassen″ auf Seite 277
- „Transmogrifier - Schritt ″Data Manipulation″ anpassen″ auf Seite 283
- „TransmogrifiedWarning - Warnung an Client senden″ auf Seite 284

Nähere Informationen hierzu finden Sie in Kapitel 4, „Die Datei ibmproxy.conf manuell editieren″, auf Seite 13.

Konfigurations- und Verwaltungsformulare

Im folgenden Konfigurations- und Verwaltungsformular können Sie die Werte der zugehörigen Anweisungen ändern:

- **Serverkonfiguration –> Anforderungsverarbeitung –> Verarbeitung von API-Anforderungen**

Nähere Informationen hierzu finden Sie in Kapitel 2, „Konfigurations- und Verwaltungsformulare verwenden″, auf Seite 7.

Teil 4. Proxy-Server-Caching konfigurieren

Dieser Teil beschreibt den Proxy-Cache und erläutert, wie Sie ihn konfigurieren. Der Cache kann so konfiguriert werden, dass Dateien im Hauptspeicher (Speicher-Cache) oder auf einer oder mehreren Speichereinheiten (Platten-Cache) gespeichert werden. Sie können einen Agenten für die Cache-Aktualisierung konfigurieren, der häufig angeforderte Dateien vorab in den Cache lädt. Darüber hinaus sind verschiedene URL-Filter für das Caching verfügbar. In diesem Teil wird ferner beschrieben, wie der Cache über fernen Cache-Zugriff oder das Plug-in ICP (Internet Caching Protocol) gemeinsam genutzt werden kann, wie veraltete Dateien mit der Garbage-Collection aus dem Cache gelöscht und wie dynamisch generierte Dateien im Cache gespeichert werden können.

Dieser Teil enthält die folgenden Kapitel:

Kapitel 16. Übersicht über das Proxy-Server-Caching

Beim Caching speichert der Proxy-Server lokale Kopien der von den Clients angeforderten Dateien im Cache, so dass er bei erneuter Anforderung von demselben oder anderen Clients diese Dateien direkt aus dem Cache bereitstellen kann.

Caching Proxy ist mit HTTP 1.1 kompatibel und folgt im Allgemeinen dem Protokoll HTTP 1.1 für das Caching und für die Überprüfung der Dokumentaktualität.

Dieses Kapitel beschreibt einige Funktionen des Proxy-Server-Cache. In den folgenden Kapiteln wird beschrieben, wie Sie die richtigen Konfigurationswerte für diese Funktionen definieren.

Cache-Speicher

Der Proxy-Server kann den Cache auf einer physischen Speichereinheit oder im Systemspeicher speichern. Welche Art von Cache-Speicher für Ihr System besser geeignet ist, hängt vom Leistungsspektrum Ihrer Hardware und davon ab, ob Ihnen kürzere Cache-Antwortzeiten oder eine höhere Maximalanzahl von Elementen im Cache wichtiger ist. Die Antwortzeiten eines Speicher-Cache sind in der Regel kürzer als die eines Platten-Cache, aber die Größe des Speicher-Cache wird durch die RAM-Kapazität der Proxy-Server-Maschine begrenzt. Die Größe eines Platten-Cache wird durch die Kapazität der Speichereinheit beschränkt, die in der Regel sehr viel höher ist als die RAM-Kapazität.

Für Platten-Caches verwendet Caching Proxy reines Platten-Caching, d. h. der Proxy-Server schreibt direkt auf die Cache-Einheit, ohne die Lese- und Schreibprotokolle des Betriebssystems zu verwenden. Die Speichereinheit für einen Platten-Cache muss mit dem Befehl **htcformat** vorbereitet werden. Nähere Informationen zum Befehl **htcformat** finden Sie in Kapitel 17, „Basiseinstellungen für das Caching", auf Seite 81.

Cache-Index

Sowohl beim Speicher-Cache als auch beim Platten-Cache legt Caching Proxy einen Cache-Index im Systemspeicher ab, der die Verarbeitungszeit für das Suchen der im Cache gespeicherten Dateien verringert.

Die Cache-Verzeichnisstruktur und die Suchfunktionen von Caching Proxy unterscheiden sich von denen anderer Proxy-Server. Caching Proxy verwaltet im Hauptspeicher einen Index mit Informationen zu den Dateien im Cache. Wenn der RAM anstelle einer Platte oder eines anderen Datenträgers zum Suchen verwendet wird, sind Such- und Abrufoperationen schneller.

Der Index enthält URLs, Cache-Positionen und Verfallsinformationen für Objekte im Cache. Aus diesem Grund verhält sich die für den Index erforderliche Speicherkapazität proportional zur Anzahl der Objekte im Cache.

Wenn eine Anforderung von einem Client empfangen wird, prüft der Proxy-Server, ob dieser URL in dem im Systemspeicher abgelegten Cache-Index enthalten ist.

- Wenn sich die Datei nicht im Index befindet, wird die Anforderung an den Zielserver weitergeleitet.

  - Der URL wird dann geprüft, um festzustellen, ob die abgerufene Datei im Cache gespeichert werden kann. Fällt das Ergebnis der Prüfung positiv aus, speichert Proxy-Server die abgerufene Datei im Cache.
  - Der Cache-Index wird dann mit dem URL, der Position und den Verfallsinformationen für das neu im Cache gespeicherte Objekt aktualisiert.
- Wenn sich die Datei im Index befindet, geht der Proxy-Server wie folgt vor:
  - Die Verfallsinformationen werden geprüft, um festzustellen, ob die Datei im Cache aktuell ist.
    - Wenn die Verfallszeit des Objekts abgelaufen ist, wird Kontakt mit dem Zielserver aufgenommen, und das verfallene Objekt wird durch das neu abgerufene Dokument ersetzt. Die Verfallsinformationen im Cache-Index werden aktualisiert.
    - Wenn die Verfallszeit des Objekts nicht abgelaufen ist, wird das Dokument aus dem Cache des Proxy-Servers zurückgegeben.

FTP-Caching

Wenn im Proxy-Server das Zwischenspeichern (Caching) von Anforderungen konfiguriert ist, kann er sowohl FTP- als auch HTTP-Dateianforderungen zwischenspeichern. Da FTP-Dateien jedoch nicht dieselben Header-Informationen wie HTTP-Dateien enthalten, wird das Verfallsdatum für zwischengespeicherte FTP-Dateien anders als für andere zwischengespeicherte Dateien berechnet.

Wenn der FTP-Server eine Anforderung zum Abrufen einer Datei empfängt, sendet der Proxy-Server zunächst eine LIST-Anforderung für die Datei an den FTP-Server, um FTP-Verzeichnisinformationen zu dieser Datei zu erhalten. Falls der FTP-Server auf die LIST-Anforderung hin mit einer Ausführungsbestätigung und die Verzeichnisinformationen zu der Datei sendet, erstellt der Proxy-Server einen HTTP-Header ″Last-Modified″ mit dem Datum, das aus diesen FTP-Verzeichnisinformationen ermittelt wird. Die Caching-Funktion des Proxy-Servers verwendet diesen Header ″Last-Modified″ anschließend zusammen mit dem Wert der Anweisung CacheLastModifiedFactor aus der Konfigurationsdatei, um die maximale Verweildauer der FTP-Datei im Cache-Speicher zu bestimmen.

Nähere Informationen zur Verwendung des Header ″Last-Modified″ und der Anweisung CacheLastModifiedFactor zur Bestimmung der Verweildauer einer Datei im Cache-Speicher finden Sie in Kapitel 19, „Cache-Inhalt verwalten”, auf Seite 89.

FTP-Dateien, die nicht durch anonyme Anmeldung, sondern für eine bestimmte Benutzer-ID abgerufen werden, werden als private Dateien behandelt und nicht zwischengespeichert.

DNS-Caching

Zusätzlich zum Caching von Webinhalten führt der Proxy-Server DNS-Caching (DNS = Domänennamensserver) durch. Wenn ein Client beispielsweise einen URL von der Website www.meineWebsite.com anfordert, fordert der Proxy-Server seinen DNS-Server auf, den Hostnamen www.meineWebsite.com in eine IP-Adresse aufzulösen. Die IP-Adresse wird dann in den Cache gestellt, um die Antwortzeit bei zukünftigen Anforderungen für diesen Hostnamen zu verbessern. DNS-Caching wird automatisch durchgeführt, und die Konfiguration kann diesbezüglich nicht geändert werden.

Cache-Ausschlüsse

Einige Dateien und Dokumente werden nie im Cache gespeichert. Dazu gehören die folgenden Dateien:

- Dateien, die auf Anforderungen zurückgegeben werden, die andere HTTP-Methoden als GET verwenden, wie zum Beispiel POST und PUT.
- Dokumente, die eine Authentifizierung erfordern, sofern das Caching dieser Dokumente nicht explizit vom Ursprungsserver zugelassen wird.
- Die dynamische Ausgabe eines beliebigen CGI-Scripts (da diese bei jeder Anforderung verschieden ist). Dynamisch generierte Ergebnisse von Servlets und JavaServer Pages (JSP), die von IBM WebSphere Application Server ausgeführt werden, können im Cache gespeichert werden, wenn dynamisches Caching aktiviert ist. Nähere Informationen hierzu finden Sie in Kapitel 22, „Caching von dynamisch erstelltem Inhalt", auf Seite 105.
- Alle Dateien, die von einem URL zurückgegeben werden, der ein Fragezeichen (?) enthält, falls das Caching von Abfragen nicht explizit zugelassen ist. (Nähere Informationen zum Konfigurieren des Caching von Abfrageergebnissen finden Sie in Kapitel 18, „Zwischenzuspeichernde Inhalte steuern", auf Seite 85.)

Sie können die im Cache gespeicherten Elemente noch weiter einschränken, indem Sie Caching-Filter definieren. Beispielsweise könnten Sie festlegen, dass der Proxy-Server keine Dateien im Cache speichern soll, die lokal vom Proxy-Server bereitgestellt werden. Nähere Informationen hierzu finden Sie in Kapitel 18, „Zwischenzuspeichernde Inhalte steuern", auf Seite 85.

Cache-Verwaltung

Zum Verwalten eines Cache gehören viele Faktoren. Als Serveradministrator können Sie Folgendes festlegen:

- Welche Dokumente im Cache gespeichert werden. Nähere Informationen hierzu finden Sie in Kapitel 18, „Zwischenzuspeichernde Inhalte steuern", auf Seite 85.
- Wie viele Dokumente im Cache gespeichert werden können. Nähere Informationen hierzu finden Sie in Kapitel 17, „Basiseinstellungen für das Caching", auf Seite 81.
- Wie lange im Cache gespeicherte Dokumente als aktuell gelten. Nähere Informationen hierzu finden Sie in Kapitel 19, „Cache-Inhalt verwalten", auf Seite 89.
- Wie häufig der Cache gelöscht werden soll (Garbage-Collection) und welche Art von Dateien eher beibehalten werden sollen. Nähere Informationen hierzu finden Sie in Kapitel 19, „Cache-Inhalt verwalten", auf Seite 89.
- Wie im Cache gespeicherte Dokumente indexiert werden sollen. Nähere Informationen hierzu finden Sie in Kapitel 17, „Basiseinstellungen für das Caching", auf Seite 81.
- Wann der Cache aktualisiert wird. Nähere Informationen hierzu finden Sie in Kapitel 20, „Den Cache-Agenten für automatische Aktualisierung und Vorabladen konfigurieren", auf Seite 95.
- Ferner Cache-Zugriff. Nähere Informationen hierzu finden Sie in Kapitel 21, „Gemeinsamen Cache verwenden", auf Seite 101.
- Wie Protokolle aufbewahrt und archiviert werden. Nähere Informationen hierzu finden Sie in Kapitel 17, „Basiseinstellungen für das Caching", auf Seite 81.

Zusätzlich können Anpassungen der Cache-Konfiguration vorgenommen werden, um die Gesamtleistung von Caching Proxy zu verbessern. Nähere Informationen zur Leistungsoptimierung finden Sie in Kapitel 23, „Proxy-Server-Cache optimieren”, auf Seite 109.

Kapitel 17. Basiseinstellungen für das Caching

Wenn Sie zum Installieren von Caching Proxy im Produktinstallationsprogramm von Edge Components die Standardeinstellungen verwendet haben, ist das Caching aktiviert, und der Cache wird im Hauptspeicher gespeichert. Mit den folgenden Basiseinstellungen für das Caching können Sie den Cache an die Anforderungen Ihres Systems anpassen.

Falls Sie das Installationsprogramm nicht verwendet haben, konfigurieren Sie diese Einstellungen, um das Caching zu aktivieren.

Im Folgenden sind die grundlegenden Schritte für die Konfiguration des Caching aufgeführt:

1. Caching aktivieren
2. Cache-Speicher konfigurieren

Nach der Konfiguration der Basiseinstellungen für den Cache können Sie die Einstellungen für die folgenden Funktionen hinzufügen oder ändern:

- Cache-Speicher anpassen
- Cache-Speicher auf Platte speichern oder laden
- Mit URL-Filtern die zwischenzuspeichernden Elemente einschränken
- Durch die Aktivierung des Caching von Abfrageergebnissen oder dynamisch generierten Dateien den Umfang der zwischenzuspeichernden Elemente erweitern
- Verfallsdatum für zwischengespeicherte Dateien und Garbage-Collection konfigurieren
- Automatisches Aktualisieren und Vorabladen des Cache konfigurieren
- Gemeinsame Nutzung des Cache mit RCA (Remote Cache Access, ferner Cache-Zugriff) oder ICP (Internet Caching Protocol) konfigurieren
- Protokollierung konfigurieren

Anleitungen zum Ändern dieser Einstellungen finden Sie in diesem Kapitel.

1. Caching aktivieren

Zum Aktivieren des Caching setzen Sie die Anweisung Caching auf "on" oder wählen Sie im Konfigurationsformular **Cache-Konfiguration –> Cache-Einstellungen** das Feld **Proxy-Caching aktivieren** aus. Wenn Sie keine Cache-Einheit angeben, wird der Cache im Speicher abgelegt. Zum Erstellen eines Platten-Cache führen Sie die Schritte im Abschnitt „2. Cache-Speicher konfigurieren" aus.

2. Cache-Speicher konfigurieren

Die Aufgaben zum Konfigurieren eines Cache-Speichers richten sich danach, ob Sie einen Speicher-Cache oder einen Platten-Cache verwenden möchten.

Wenn Sie einen Speicher-Cache verwenden möchten, müssen Sie den Wert für die Einstellung "Cache-Speicher" so wählen, dass der gesamte Inhalt eines Cache aufgenommen werden kann. Informationen zu den empfohlenen Größen für den Cache-Speicher finden Sie im Abschnitt „Cache-Speicher festlegen" auf Seite 83.

Wenn Sie einen Platten-Cache verwenden möchten, gehen Sie wie folgt vor:

1. Bereiten Sie eine Speichereinheit für den Cache vor.

   Für den Cache ist eine speziell formatierte Einheit erforderlich. Es empfiehlt sich, eine vollständige Einheit oder Plattenpartition für den Cache zu reservieren. Die Mindestgröße für einen Cache sind 16.392 KB.

   Gehen Sie wie folgt vor, um die Cache-Einheit zu formatieren:

   a. Wählen Sie eine Einheit für den Cache aus. Stellen Sie sicher, dass kein anderes Programm diesen Speicherbereich verwendet und dass die Einheit als unformatierte Einheit (Raw Device) zur Verfügung steht.

   b. Formatieren Sie die Einheit mit dem Befehl **htcformat**. Die Syntax ist wie folgt:

      ```
      htcformat Pfad_der_unformatierten_Einheit [-blocksize Blockgröße]
      [-blocks Anzahl_Blöcke]
      ```

      Die Argumente -blocksize und -blocks sind optional. Die Standardblockgröße sind 8192 Bytes. Wenn die Anzahl der Blöcke nicht angegeben ist, wird die maximal mögliche Anzahl von Blöcken auf die Plattenpartition geschrieben.

      Achten Sie bei der Angabe des Einheitenpfades darauf, dass Sie den Pfad der unformatierten Einheit angeben.

      - Auf AIX-Plattformen ist der Pfad der unformatierten Einheit für einen logischen Datenträger, der als /dev/lv02 definiert ist, beispielsweise /dev/rlv02.
      - Auf Linux-Plattformen müssen Sie vor der Ausführung des Befehls **htcformat** zuerst den Befehl **raw** ausführen, um den Pfad der unformatierten Einheit dem realen SCSI-Laufwerk sdb1 zuzuordnen.

        ```
        raw /dev/raw/raw1 dev/sdb1
        ```

      - Auf HP-UX- und Solaris-Plattformen ist der Pfad der unformatierten Einheit für eine Partition, die als /dev/dsk/c0t0d0s0 definiert ist, beispielsweise /dev/rdsk/c0t0d0s0.
      - Auf Windows-Plattformen ist der Pfad der unformatierten Einheit für eine Einheit, die als e: definiert ist, beispielsweise \\.\e:.

      Zusätzliche Informationen zum Zugriff auf unformatierte Einheiten finden Sie im Referenzmaterial zu Ihrem Dateisystem.

2. Geben Sie die Cache-Einheit mit der Anweisung CacheDev oder im Konfigurationsformular **Cache-Einstellungen** an. Sie können auch mehrere Einheiten angeben.

Achtung:
Auf Windows-Systemen markiert der Befehl htcformat die Cache-Einheit nicht automatisch als nicht beschreibbar.

Wenn das Betriebssystem versucht, auf die Cache-Einheit zu schreiben, können zwischengespeicherte Daten verloren gehen. Sie können dies verhindern, indem Sie mit dem Windows-Dienstprogramm "Datenträgerverwaltung" die Platte vorbereiten, bevor Sie den Befehl htcformat ausführen. Löschen Sie zur Vorbereitung der Platte mit dem Plattendienstprogramm die Einheit oder Partition, die Sie verwenden möchten, und erstellen Sie sie anschließend erneut, ohne sie zu formatieren. Dieser Schritt bewirkt, dass das System die Einheit für den Systemspeicher als nicht verfügbar einstuft.

Optionale Anpassungen

Cache-Speicher festlegen

Legen Sie den Wert gemäß den folgenden Richtlinien mit der Anweisung CacheMemory (oder im Feld **Cache-Speicher** des Konfigurationsformulars **Cache-Einstellungen**) fest. Die mit diesem Wert festgelegte Speicherkapazität wird für die Unterstützung der Cache-Infrastruktur einschließlich des Cache-Index und, sofern das Speicher-Caching konfiguriert wurde, für das Speichern des Cache-Inhalts verwendet.

Mindestwerte

Damit Platten-Caches mit optimaler Leistung arbeiten können, wird für die Einstellung ″Cache-Speicher″ ein Mindestwert von 64 MB für die Unterstützung der Cache-Infrastruktur einschließlich Cache-Index empfohlen. Mit zunehmender Größe des Cache wächst auch der Cache-Index an, und es wird zusätzlicher Cache-Speicher für das Speichern des Index benötigt. 64 MB für den Cache-Speicher reichen aus, um die Unterstützung der Cache-Infrastruktur zu gewährleisten und den Cache-Index für einen Platten-Cache mit einer Größe von bis zu 6,4 GB zu speichern. Für größere Platten-Caches sollte der Cache-Speicher 1 % der Cache-Größe betragen.

Bei Speicher-Caches entspricht der Wert für den Cache-Speicher der Speicherkapazität, die für die Unterstützung der Cache-Infrastruktur und den Cache selbst reserviert wurde. Für den Cache-Speicher wird ein Mindestwert von 64 MB empfohlen.

Maximalwert

Wenn zu viel physischer Speicher für den Speicher-Cache reserviert wird, können unerwünschte Folgen auftreten, z. B. Fehler wegen mangelnder Speicherkapazität oder Fehler des Proxy-Servers. Die Einschränkungen bezüglich der Einstellung des Cache-Speichers sind auf die Einschränkungen von 32-Bit-Anwendungen zurückzuführen. Da Caching Proxy eine 32-Bit-Anwendung ist, können maximal 2 GB Hauptspeicher verwendet werden.

Caching Proxy reserviert den mit der Anweisung CacheMemory definierten Hauptspeicher und verwendet diesen als Cache für das Speichern von Objekten. Unabhängig davon, ob der Cache ein Speicher-Cache oder ein Platten-Cache ist, müssen Sie zusätzlichen Speicher für die Cache-Datenstrukturen, die Netz-E/A- und Verbindungspuffer, die Sitzungspuffer und den Hauptprozess und alle Threads reservieren. Darüber hinaus kann es erforderlich sein, für Anforderungen bestimmter Clients einen größeren Speicherpoolblock zu reservieren als standardmäßig vorgesehen. Wenn mit der Anweisung CacheMemory ein Wert knapp unterhalb des 2-GB-Grenzwerts festgelegt wurde, steht Caching Proxy möglicherweise nicht genügend Speicher für die Ausführung zur Verfügung, insbesondere dann, wenn die Arbeitslast sehr hoch ist.

Es wird empfohlen, die Anweisung CacheMemory auf einen Wert kleiner-gleich 1.600 MB zu setzen. Falls Sie einen höheren Wert als 1.600 MB angeben, geht dies zu Lasten des Speichers, den Caching Proxy für den normalen Betrieb benötigt, und verursacht nachteilige Nebeneffekte. Diese Nebeneffekte äußern sich normalerweise, aber nicht ausschließlich in einer erhöhten CPU-Belastung (möglicherweise bis zu einer Belastung von 100 %), im Auftreten von Fehlern wegen mangelnder Speicherkapazität und in einer schlechteren Leistung. Ist insgesamt ein größerer Cache erforderlich, sollten Sie Cache-Einheiten verwenden oder eine Konfiguration implementieren, die einen gemeinsam genutzten Cache mit RCA oder ICP vorsieht.

Cache-Speicher auf Platte speichern oder laden

Sie können den Cache-Inhalt aus einer Speicherauszugsdatei importieren oder in eine Speicherauszugsdatei exportieren. Dies ist nützlich, wenn bei einem Neustart Cache-Speicher verloren geht oder wenn derselbe Cache für mehrere Proxy-Server implementiert wird.

Caching-Filter definieren

Sie können den Umfang der zwischenzuspeichernden Inhalte einschränken, indem Sie Filter für den Abgleich von URL-Anforderungsformaten verwenden. Nähere Informationen zum Konfigurieren von Filtern finden Sie in Kapitel 18, „Zwischenzuspeichernde Inhalte steuern", auf Seite 85.

Caching für Abfrageergebnisse und dynamisch generierte Dateien konfigurieren

Sie können den Proxy-Server so konfigurieren, dass die Ergebnisse von Abfrageanforderungen zwischengespeichert werden. Standardmäßig werden URLs, die ein Fragezeichen (?) enthalten, nicht zwischengespeichert. Nähere Informationen hierzu finden Sie im Abschnitt „Abfrageantworten zwischenspeichern" auf Seite 86.

Es können auch die Ergebnisse einer Servlet- oder JSP-Ausführung in einem IBM WebSphere Application Server zwischengespeichert werden. Nähere Informationen hierzu finden Sie in Kapitel 22, „Caching von dynamisch erstelltem Inhalt", auf Seite 105.

Verfallsdatum für Dateien und Garbage-Collection konfigurieren

In Kapitel 19, „Cache-Inhalt verwalten", auf Seite 89 wird beschrieben, wie Sie das Verfallsdatum für Dateien im Cache konfigurieren und festlegen können, wie veraltete Dateien gelöscht werden sollen.

Automatisches Vorabladen konfigurieren

Der Cache kann so konfiguriert werden, dass die am häufigsten verwendeten Dateien täglich aktualisiert werden, bevor sie angefordert werden. Nähere Informationen hierzu finden Sie in Kapitel 20, „Den Cache-Agenten für automatische Aktualisierung und Vorabladen konfigurieren", auf Seite 95.

Gemeinsame Nutzung des Cache konfigurieren

In bestimmten Situationen kann durch die Verwendung eines gemeinsam genutzten Cache die Wahrscheinlichkeit erhöht werden, dass eine angeforderte Datei im Cache gefunden wird. Nähere Informationen hierzu finden Sie in Kapitel 21, „Gemeinsamen Cache verwenden", auf Seite 101.

Protokollierung konfigurieren

Die Verwaltung exakter Protokolle ist für die Verwaltung von Caching Proxy von entscheidender Bedeutung. In Teil 6, „Caching Proxy überwachen", auf Seite 147 finden Sie Informationen zum Konfigurieren und Verwenden der Proxy-Server-Protokolle.

Kapitel 18. Zwischenzuspeichernde Inhalte steuern

Caching Proxy stellt mehrere Filtermethoden zur Verfügung, mit denen Sie steuern können, welche Dateien, Dokumente und anderen Objekte zwischengespeichert werden:

- URL-basierte Caching-Filter
- Caching von Abfrageantworten
- Caching lokal bereitgestellter Dateien
- Caching von Dateien basierend auf einem Teil-URL
- Caching von Dateien basierend auf einem Teil des Anforderungs-URL
- Caching dynamisch generierter Dateien (siehe Kapitel 22, „Caching von dynamisch erstelltem Inhalt", auf Seite 105)

Anmerkung: Das Konfigurations- und Verwaltungsformular **Cache-Konfiguration –> Cache-Verhalten** enthält die Option **Cache-Namen basierend auf ankommendem URL festlegen**. (Die entsprechende Anweisung in der Konfigurationsdatei heißt CacheByIncomingURL.) Diese Anweisung verweist auf den Dateinamen der Datei im Cache. Wählen Sie dieses Markierungsfeld aus, damit der Dateiname der Cache-Datei basierend auf dem ankommenden URL festgelegt wird. Wenn das Markierungsfeld nicht ausgewählt ist, wird der Dateiname basierend auf dem abgehenden URL festgelegt.

Filter für URL-basiertes Caching konfigurieren

Der Proxy-Server kann so konfiguriert werden, dass er Anforderungen mit einer URL-Schablone vergleicht, um festzustellen, ob eine Datei zwischengespeichert wird. Zur Konfiguration dieser Funktion werden Schablonen für Anforderungen definiert, deren Dateien *immer* zwischengespeichert werden. Außerdem können separate Schablonen für die Anforderungen definiert werden, deren Dateien *nicht* zwischengespeichert werden dürfen. Es können mehrere Schablonen verwendet werden. Ein ähnliches Verfahren wird verwendet, um das Caching für Abfrageantworten zu aktivieren. Nähere Informationen hierzu finden Sie im Abschnitt „Abfrageantworten zwischenspeichern" auf Seite 86.

Wenn Sie die URL-Caching-Filter in der Datei ibmproxy.conf definieren möchten, finden Sie diesbezügliche Anleitungen in den Abschnitten „CacheOnly - Nur Dateien im Cache speichern, deren URLs mit einer Schablone übereinstimmen" auf Seite 197 und „NoCaching - Dateien, deren URLs mit einer Schablone übereinstimmen, nicht im Cache speichern" auf Seite 244.

Wenn Sie die URL-Caching-Filter in den Konfigurations- und Verwaltungsformularen definieren möchten, verwenden Sie hierfür das Feld **Filter für Zwischenspeicherung nach URL** im Formular **Cache-Konfiguration –> Cache-Verhalten**. In diesem Abschnitt geben Sie die URLs, deren Dateien immer zwischengespeichert werden, oder die URLs an, deren Dateien nicht zwischengespeichert werden. Wenn Sie zwei Listen anlegen möchten, eine mit den Dateien, die immer zwischengespeichert werden sollen, und eine andere mit Dateien, die nicht zwischengespeichert werden sollen, müssen Sie zuerst die eine Liste erstellen und dann auf die Schaltfläche **Übergeben** klicken, bevor Sie die andere Liste erstellen.

Abfrageantworten zwischenspeichern

Die Antworten, die auf Abfragen (d. h. auf URL-Anforderungen, die ein Fragezeichen (?) enthalten), zurückgegeben werden, können unter Verwendung von Caching-Filtern zwischengespeichert werden. Dies kann in Szenarios mit einem Reverse Proxy (Ersatz) hilfreich sein, wenn viele Clients dieselbe Abfrage senden.

Sie können das Caching für Abfragen konfigurieren, indem Sie die Anweisung CacheQueries in der Konfigurationsdatei ibmproxy.conf editieren. Die Anweisung CacheQueries unterstützt folgende Optionen:

- Always - Alle Abfrageantworten von den Hosts, die mit der Schablone übereinstimmen, werden zwischengespeichert, falls sie gemäß den Standards von HTTP 1.1 zwischenspeicherbar sind.
- Public - Die Abfrageantworten von Hosts, die mit der Schablone übereinstimmen, werden zwischengespeichert, falls sie den Header ″Cache-control: public″ oder einen Header für erzwungene Neuüberprüfung (Forced Revalidation) enthalten und falls sie gemäß den Standards von HTTP 1.1 zwischenspeicherbar sind.

Zusätzliche Informationen zu diesen Optionen finden Sie im Abschnitt „CacheQueries - Cache-Antworten auf URLs festlegen, die ein Fragezeichen (?) enthalten" auf Seite 197.

Wenn Sie das Caching für Abfrageantworten mit den Konfigurations- und Verwaltungsformularen konfigurieren möchten, verwenden Sie hierfür das Feld **Abfrage/Antwortfilter für Zwischenspeicherung nach URL** im Formular **Cache-Konfiguration** –> **Cache-Verhalten**. Wenn Sie zwei Listen anlegen möchten, müssen Sie zuerst die eine Liste erstellen und dann auf **Übergeben** klicken, bevor Sie die andere Liste erstellen.

Weitere Voraussetzungen für das Caching von Abfrageantworten

Nachdem Sie die Einstellung für das Caching von Abfragen konfiguriert haben, müssen Sie sich vergewissern, ob die folgenden Einstellungen für das Caching von Abfrageantworten richtig konfiguriert sind. Informationen zum Festlegen dieser Optionen in den Konfigurations- und Verwaltungsformularen finden Sie im Abschnitt „Cache-Aktualität konfigurieren" auf Seite 92.

- CacheTimeMargin - Diese Anweisung legt die Mindestverfallszeit fest. Dateien, deren Verfallszeit kleiner ist als dieser Mindestwert, werden nicht zwischengespeichert. Weil Abfrageantworten manchmal sehr kurze Verfallszeiten besitzen, wird eine größere Anzahl von Abfrageantworten zwischengespeichert, wenn Sie diese Anweisung auf einen niedrigen Wert setzen. Ziehen Sie den Abschnitt „CacheTimeMargin - Mindestlebensdauer für das Caching einer Datei angeben" auf Seite 199 zu Rate oder verwenden Sie das Formular **Einstellungen für Verfallszeit des Cache**, das im Abschnitt „Cache-Aktualität konfigurieren" auf Seite 92 beschrieben wird.
- CacheDefaultExpiry - Diese Anweisung legt die Verfallszeit für Dateien fest, die kein explizites Verfallsdatum oder kein letztes Änderungsdatum besitzen, auf dessen Basis die Verfallszeit berechnet werden kann. Wird diese Einstellung für HTTP-Anforderungen auf einen höheren Wert als den Standardwert 0 gesetzt, wird eine größere Anzahl von Abfrageantworten zwischengespeichert. Allerdings erhöht sich dadurch auch das Risiko, dass veraltete Inhalte aus dem Cache zurückgegeben werden.

Ziehen Sie den Abschnitt „CacheDefaultExpiry - Standardverfallszeit für Dateien angeben" auf Seite 191 zu Rate oder verwenden Sie das Formular **Einstellungen für Verfallszeit des Cache**, das im Abschnitt „Cache-Aktualität konfigurieren" auf Seite 92 beschrieben wird.

- CacheLastModifiedFactor - Mit dieser Anweisung wird ein Verfallsdatum für Dateien berechnet, die ein letztes Änderungsdatum besitzen, aber kein explizites Verfallsdatum. Wird dieser Faktor für HTTP-Dateien auf einen höheren Wert gesetzt, verlängert sich die Verweildauer einer HTTP-Datei im Cache ohne erneute Überprüfung. Allerdings erhöht sich dadurch auch das Risiko, dass veraltete Inhalte aus dem Cache zurückgegeben werden. Ziehen Sie den Abschnitt „CacheLastModifiedFactor - Wert zur Berechnung der Verfallsdaten angeben" auf Seite 193 zu Rate oder verwenden Sie das Formular **Faktor letzte Änderung**, das im Abschnitt „Cache-Aktualität konfigurieren" auf Seite 92 beschrieben wird.
- Optional können Sie die Anweisung SignificantUrlTerminator oder die Anweisung AggressiveCaching definieren. Nähere Informationen hierzu finden Sie in den Abschnitten „SignificantURLTerminator - Abschlusscode für URL-Anforderungen festlegen" auf Seite 278 und „AggressiveCaching - Caching für Dateien festlegen, die als nicht zwischenspeicherbar gekennzeichnet sind" auf Seite 186.

Caching lokal bereitgestellter Dateien

Da eine Zwischenspeicherung der vom Proxy-Server bereitgestellten Dateien in der Regel nicht effizient ist, werden Dateien, die aus der lokalen Domäne des Servers stammen, standardmäßig nicht zwischengespeichert. Wenn Sie Objekte, die aus der lokalen Domäne des Servers stammen, zwischenspeichern möchten, wählen Sie das Feld **Lokalen Domänendateien zwischenspeichern** im Konfigurations- und Verwaltungsformular **Cache-Konfiguration** –> **Cache-Verhalten** aus. Alternativ können Sie die Anweisung CacheLocalDomain in der Konfigurationsdatei des Proxy-Servers auf `on` setzen.

Caching von Dateien basierend auf Teil-URL

Für die Entscheidung, ob ein Objekt zwischengespeichert wird, kann anstelle des vollständigen URL auch nur ein bestimmter (signifikanter) Teil des eingehenden URL herangezogen werden. Diese Art von Filter ist für die transaktionsgesteuerte Bereitstellung von Webinhalten und für dynamisches Caching nützlich, da für diverse eingehende Anforderungen häufig dieselbe Antwort zurückgegeben wird, wenn signifikante Teile der URLs eingehender Anforderungen identisch sind.

Das auf Teil-URLs basierende Caching kann nicht in den Konfigurations- und Verwaltungsformularen festgelegt werden. Sie müssen die Anweisung SignificantUrlTerminator in der Konfigurationsdatei des Proxy-Servers verwenden, um einen Beendigungscode für URL-Anforderungen anzugeben. Diese Spezifikation bewirkt, dass Caching Proxy bei der Verarbeitung der Anforderung nur die Zeichen vor dem Beendigungscode auswertet und anhand des Ergebnisses bestimmt, ob die angeforderte Datei zwischengespeichert wird. Falls mehrere Beendigungscodes definiert sind, vergleicht Caching Proxy die eingehenden URLs in der Reihenfolge mit den Beendigungscodes, in der diese in der Datei ibmproxy.conf definiert sind. Nähere Informationen hierzu finden Sie im Abschnitt „SignificantURLTerminator - Abschlusscode für URL-Anforderungen festlegen" auf Seite 278.

Zugehörige Anweisungen in der Konfigurationsdatei

Die Referenzabschnitte zu den folgenden Anweisungen beschreiben, wie Sie Caching-Filter direkt in der Konfigurationsdatei des Proxy-Servers definieren können:

- „NoCaching - Dateien, deren URLs mit einer Schablone übereinstimmen, nicht im Cache speichern" auf Seite 244
- „CacheOnly - Nur Dateien im Cache speichern, deren URLs mit einer Schablone übereinstimmen" auf Seite 197
- „CacheQueries - Cache-Antworten auf URLs festlegen, die ein Fragezeichen (?) enthalten" auf Seite 197
- „CacheLocalDomain - Angeben, ob die lokale Netzdomäne im Cache gespeichert werden soll" auf Seite 194
- „SignificantURLTerminator - Abschlusscode für URL-Anforderungen festlegen" auf Seite 278

Informationen zu Dokumenten, die nicht zwischengespeichert werden können, finden Sie in Kapitel 16, „Übersicht über das Proxy-Server-Caching", auf Seite 77.

Kapitel 19. Cache-Inhalt verwalten

Da beim Caching eine Kopie der bereitgestellten Datei erstellt und gespeichert wird, müssen routinemäßig Pflegemaßnahmen ergriffen werden, damit der Cache ordnungsgemäß funktioniert. Die Dateien im Cache müssen auf *Aktualität* geprüft und ungültig gemacht werden, wenn sie nicht mehr mit den Dateien auf dem Ursprungsserver konsistent sind. Dieser Dateiverfallsprozess wird im Abschnitt „Dateiverfall" beschrieben. Außerdem müssen ungültige und nicht mehr verwendete Dateien aus dem Cache gelöscht werden, um Platz für neue Dateien zu machen. Dieser Cache-Bereinigungsprozess wird im Abschnitt „Garbage-Collection" auf Seite 94 beschrieben.

Dateiverfall

Die Aufgabe, die Objekte im Cache mit den Originalobjekten auf dem Inhaltsserver konsistent zu halten, wird als Aufrechterhalten der Cache-Aktualität bezeichnet. Für jedes Dokument oder andere Objekt, das Caching Proxy im Cache speichert, wird ein Zeitpunkt errechnet, zu dem das Dokument verfällt.

Für HTTP-Seiten enthält der Header des Dokuments, der vom Inhaltsserver generiert wird, die Informationen zum Verfallszeitpunkt.

Weil das Protokoll FTP keine äquivalenten Informationen zur Verfallszeit enthält, generiert Caching Proxy für FTP-Dateien einen eigenen Header `Last-Modified:`, der auf den FTP-Verzeichnisinformationen der Dateien basiert, und berechnet anhand dieser Informationen die Verfallszeit. Kann der Proxy-Server für eine Datei auf dem FTP-Server keine Verzeichnisinformation abrufen, wird der für den FTP-URL passende Standardwert verwendet. Weil für FTP-Server kein Standarddatumsformat existiert, kann Caching Proxy die von einigen FTP-Servern gesendeten Angaben zu Datum und Uhrzeit möglicherweise nicht interpretieren. In diesen Fällen wird die Standardverfallszeit des Proxy-Servers verwendet. Durch Verwendung dieser Prozedur kann der Proxy-Server das Caching von HTTP-Seiten und FTP-Dateien auf ähnliche Weise durchführen.

Der Verfallszeitpunkt kann von einem Inhaltsserver auf eine der folgenden Arten angegeben werden (in der gewünschten Reihenfolge):

1. Der Inhaltsserver gibt den Header `Cache-control: s-maxage=`*n* an. Damit wird dem Proxy-Server mitgeteilt, dass das Objekt für den Zeitraum von *n* Sekunden nach dem Empfang aktuell bleibt.
2. Der Inhaltsserver gibt den Header `Cache-control: max-age=` *n* an. Damit wird dem Proxy-Server mitgeteilt, dass das Objekt für den Zeitraum von *n* Sekunden nach dem Empfang aktuell bleibt.
3. Der Inhaltsserver gibt den Header `Expires:` *n* an. Damit wird dem Proxy-Server mitgeteilt, dass das Objekt für den mit *n* angegebenen Zeitraum aktuell bleibt.
4. Der Inhaltsserver gibt mit dem Header `Last-Modified:`*n* den Zeitpunkt an, zu dem das Dokument zuletzt geändert wurde. Der Proxy-Server berechnet die Zeitspanne seit der letzten Änderung des Dokuments, multipliziert diese mit dem Faktor letzte Änderung für den Cache (Cache Last Modified), der in der Konfigurationsdatei des Proxy-Servers definiert ist, und geht dann davon aus, dass das Dokument für die errechnete Zeit gültig bleibt. Wenn der Inhaltsserver beispielsweise feststellt, dass das Dokument vor einer Woche (sieben Tage)

zuletzt geändert wurde und der Faktor letzte Änderung für den Cache auf 0.14 gesetzt ist, geht der Proxy-Server davon aus, dass das Dokument ungefähr einen Tag lang gültig ist. Anweisungen zum Definieren des Faktors letzte Änderung für den Cache enthält der Abschnitt „Cache-Aktualität konfigurieren" auf Seite 92.

5. Wird keine dieser Informationen vom Inhaltsserver angegeben, sucht Caching Proxy nach der Einstellung für die Standardverfallszeit des Cache (Cache Default Expiry), die für den aktuellen URL gilt, und verwendet diese Einstellung als Verfallszeit. Anweisungen zum Definieren der Standardverfallszeit für den Cache enthält der Abschnitt „Cache-Aktualität konfigurieren" auf Seite 92.

Nachdem die Verfallszeit wie oben beschrieben berechnet wurde, prüft Caching Proxy, ob ein Wert für die Mindesthaltezeit für diesen URL existiert. Ist dies der Fall und ist die angegebene Zeit länger als die errechnete Verfallszeit, wird die angegebene Mindesthaltezeit als Verfallszeit des Objekts verwendet. Dies gilt auch, wenn Caching Proxy für ein Dokument eine Verfallszeit von 0 Minuten berechnet. Sie müssen deshalb bei der Einstellung der Mindesthaltezeit sorgfältig vorgehen, um zu verhindern, dass veraltete Inhalte bereitgestellt werden. (Verwenden Sie zum Festlegen der Mindesthaltezeit die Anweisung CacheMinHold oder die Einstellung **URL-Verfallszeit** im Formular **Cache-Konfiguration –> Einstellungen für Verfallszeit des Cache**. Nähere Informationen hierzu finden Sie im Abschnitt „Cache-Aktualität konfigurieren" auf Seite 92.)

Der endgültige Wert für die Verfallszeit wird mit der Zeit verglichen, die die Einstellung **Zeitlimit** angibt. Ist die Verfallszeit größer als der Wert für **Zeitlimit**, wird das Dokument zwischengespeichert. Andernfalls wird es dem Cache nicht hinzugefügt. (Anweisungen zum Festlegen des Werts für **Zeitlimit** finden Sie im Abschnitt „Cache-Aktualität konfigurieren" auf Seite 92.)

Wenn das Dokument im Cache gefunden wird, die Verfallszeit aber überschritten ist, sendet Caching Proxy eine spezielle Anforderung, *if-modified-since*, an den Inhaltsserver. Diese Anforderung veranlasst den Inhaltsserver, das Dokument nur in dem Fall zu senden, wenn es seit dem letzten Empfang auf dem Proxy-Server geändert wurde. Wenn das Dokument nicht geändert wurde, sendet der Inhaltsserver nur eine entsprechende Nachricht und sendet die Seite nicht erneut. In diesem Fall ruft der Proxy-Server das Dokument aus dem Cache ab. Für FTP-Dateien simuliert der Proxy-Server diesen "if-modified-since"-Prozess. Wenn festgestellt wird, dass die Datei auf dem FTP-Server nicht geändert wurde, wird die Datei aus dem Cache abgerufen. Andernfalls wird die neuere Version vom FTP-Server abgerufen.

Zusätzliche Informationen zur Aktualität des Cache

- Fast alle statischen Webdokumente (im Gegensatz zu dynamisch generierten Dokumenten) enthalten den Header `Last-Modified:`. Proxy-Server verwenden diesen Header sehr häufig, um die Verfallszeit für Dokumente zu berechnen. Auch Caching Proxy verwendet diese Methode vorrangig für FTP-Dateien. Falls diese Methode fehlschlägt, greift der Proxy-Server auf die Werte für die Standardverfallszeit zurück.
- Die Header `Cache-control: s-maxage`, `Cache-control: max-age` und `Expires:` werden in nur sehr wenigen Dokumenten verwendet.
- Dynamisch generierte Seiten, die häufig nicht zwischengespeichert werden können, können einen Header `Expires: 0` oder `Cache-control: no-cache` enthalten. Diese Header geben an, dass das Dokument sofort verfällt.

Nähere Informationen zum Caching von Dateien, die dynamisch von IBM WebSphere Application Server generiert werden, finden Sie in Kapitel 22, „Caching von dynamisch erstelltem Inhalt", auf Seite 105.

- Beim Festlegen eines anderen Wertes als 0 Minuten für die Standardverfallszeit von URLs mit der Syntax HTTP: ist mit Vorsicht zu verfahren. Viele dynamisch generierte Seiten enthalten keine Header zur Verfallszeit. Für diese wird deshalb die Standardverfallszeit verwendet. Wenn Sie eine Standardverfallszeit von mehr als 0 Minuten festlegen, kann der Proxy-Server diese Objekte zwischenspeichern. Dies kann jedoch bedeuten, dass Benutzer veraltete Daten abrufen (oder unerwartete Ergebnisse von CGI-Programmen oder Servlets erhalten).
- In den folgenden Fällen überprüft der Proxy-Server die Dokumente bei jeder Anforderung erneut auf dem Server, unabhängig davon, ob das Dokument im Cache verfallen ist oder nicht:
  - Das Dokument enthält einen der folgenden Header:
    - Cache-control: s-maxage
    - Cache-control: must-revalidate
    - Cache-control: proxy-revalidate
  - Das Dokument erfordert eine Benutzerberechtigung, darf aber vom Server im Cache gespeichert werden.
  - Das Dokument enthält den Header `Cache-Control: no-cache`, wird aber trotzdem (durch aggressives Caching) im Cache gespeichert.

Informationen zu Datumsangaben in FTP

Da das Protokoll FTP nicht über so strenge Definitionen für Datum und Uhrzeit verfügt wie das Protokoll HTTP, können einige Faktoren dazu führen, dass der Header "Last-Modified", den der Proxy-Server für FTP-Dateien generiert, geringfügig vom tatsächlichen Datum der Datei abweicht. Dazu zählen folgende Faktoren:

- Anders als das HTTP-Protokoll legt das FTP-Protokoll nicht fest, dass zurückgegebene Daten im GMT-Format (westeuropäische Zeit) angegeben werden müssen. Das vom FTP-Server zurückgegebene Datum hat voraussichtlich das lokale Zeitformat des FTP-Servers. Da der Proxy-Server die Zeitzone des FTP-Servers nicht bestimmen kann, interpretiert er die Zeit als Zeit seiner eigenen Zeitzone. Davon ausgenommen ist der Windows-FTP-Server, der die Daten in westeuropäischer Zeit zurückgibt. Wenn der Proxy-Server feststellt, dass der FTP-Server unter Windows ausgeführt wird, geht er davon aus, dass das Verzeichnisdatum in westeuropäischer Zeit angegeben ist.
- Einige FTP-Server verwenden für das Datum in den zurückgegebenen Verzeichnisinformationen lediglich das Format *Monat Tag Jahr* ohne Angabe von Stunden oder Minuten. Wenn der FTP-Server keine Angaben zu Stunden und Minuten für die Datei zurückgibt, geht der Proxy-Server davon aus, dass die Datei zuletzt zur spätest möglichen Uhrzeit des vom FTP-Server zurückgegebenen Datums geändert wurde. Beispiel: Wenn die vom FTP-Server für eine Datei zurückgegebenen Verzeichnisinformationen anzeigen, dass die Datei zuletzt am 13. Oktober 1998 geändert wurde, jedoch keine Angaben zu Stunde oder Minute enthalten, geht der Proxy-Server davon aus, dass die Datei zuletzt am 13. Oktober 1998 um 23:59:59 Uhr geändert wurde. Falls der FTP-Server nicht unter Windows ausgeführt wird, wandelt der Proxy-Server dieses Datum aus seiner lokalen Zeitzone in die entsprechende westeuropäische Zeit um.

Wenn eine FTP-Datei im Cache verfällt, simuliert der Proxy-Server die HTTP-Gültigkeitsüberprüfung ″if modified since″ für die FTP-Datei. Dazu wird erneut der FTP-Befehl LIST für die angeforderte Datei ausgeführt, wobei das Dateidatum aus der vom FTP-Server zurückgegebenen Antwort abgerufen und mit dem Datum verglichen wird, das der Proxy-Server beim ursprünglichen Anfordern der Datei aus dem Header ″Last-Modified″ generiert hat. Ist das Dateidatum unverändert, kennzeichnet der Proxy-Server die FTP-Datei im Cache als neu überprüft, setzt eine neue Verfallszeit für die Datei und gibt die Datei aus dem Cache zurück, anstatt sie erneut vom FTP-Server abzurufen. Weichen die beiden Dateidaten voneinander ab, ruft der Proxy-Server die Datei erneut vom FTP-Server ab und speichert die neue Kopie mit dem neuen Dateidatum im Cache.

Die Verzeichnisinformationen zu der Datei können nicht immer vom FTP-Server abgerufen werden. Falls der Proxy-Server das Dateidatum für die FTP-Datei nicht bestimmen kann, generiert er für die Datei keinen Header ″Last-Modified″. Stattdessen verwendet er den mit der Anweisung CacheDefaultExpiry angegebenen Wert für den URL, um zu bestimmen, wie lange die Datei zwischengespeichert bleiben soll. Nach Ablauf dieser Zeit ruft der Proxy-Server die Datei erneut vom FTP-Server ab. Wenn es den Anschein hat, dass die Anweisung CacheDefaultExpiry für bestimmte FTP-Dateien im Cache sehr häufig verwendet wird und diese Dateien häufig abgerufen werden (und auf diese Weise einen hohen Datenverkehr im Netz verursachen), sollten Sie in Erwägung ziehen, für diese speziellen Dateien eine differenziertere Angabe der Anweisung CacheDefaultExpiry zu verwenden. Auf diese Weise werden diese Daten länger zwischengespeichert.

Wenn Sie die Einstellungen für die Cache-Verfallszeiten mit den Konfiguration- und Verwaltungsformularen festlegen möchten, verwenden Sie das Formular **Cache-Konfiguration** –> **Einstellungen für Verfallszeit des Cache** –> **Zeitlimit für Dateien im Cache**. Ausführliche Informationen zum Festlegen von Verfallsdaten für Dateien im Cache finden Sie im Abschnitt „Dateiverfall” auf Seite 89.

Cache-Aktualität konfigurieren

Wählen Sie in den Konfigurations- und Verwaltungsformularen **Cache-Konfiguration –> Einstellungen für Verfallszeit des Cache** aus, um die Verfallszeiten für Dateien im Cache festzulegen. Die folgenden Formulare sind hilfreich.

Verfallszeit auf URL-Basis

Legen Sie in diesem Formular die Mindestzeit fest, für die Dateien basierend auf ihren URLs im Cache gespeichert bleiben. Sie können für verschiedene URL-Anforderungsschablonen ein unterschiedliches Cache-Verhalten festlegen.

Die Referenzabschnitte zu den folgenden Anweisungen in Anhang B, „Anweisungen in der Konfigurationsdatei”, auf Seite 175 beschreiben, wie Sie die Verfallszeit für Dateien auf URL-Basis in der Konfigurationsdatei des Proxy-Servers festlegen:

- „CacheMinHold - Angeben, wie lange Dateien verfügbar bleiben sollen” auf Seite 196

Standardeinstellungen für Verfallszeiten

Im Formular **Einstellungen für Verfallszeit des Cache** können Sie die Standardeinstellungen für die Verfallszeiten von verwendeten oder nicht verwendeten Dateien definieren. Sie können für HTTP-, FTP- und Gopher-Dateien und für verwendete und für nicht verwendete Dateien verschiedene Werte festlegen.

Außerdem enthält dieses Formular weitere Optionen für die Verfallszeit von Dateien:

- **Verfall zwischengespeicherter Datei überprüfen aktivieren**: Dieses Markierungsfeld ist standardmäßig ausgewählt. Im Allgemeinen empfiehlt es sich, diese Option auszuwählen, um zu verhindern, dass der Server veraltete Inhalte zurückgibt.
- **Dateiabruf von fernen Servern inaktivieren**: Wählen Sie diese Option aus, wenn der Server keine Dateien von fernen Servern abrufen soll.
- **Keine Dateien im Cache speichern, die verfallen innerhalb**: Legen Sie mit dieser Option eine Zeitperiode fest, um zu verhindern, dass Dateien zwischengespeichert werden, die bereits nach kurzer Zeit verfallen. Standardmäßig werden Dateien, die bereits nach 10 Minuten verfallen, nicht zwischengespeichert.

Die Referenzseiten zu den folgenden Anweisungen beschreiben, wie Sie die Einstellungen für die Standardverfallszeit in der Konfigurationsdatei des Proxy-Servers festlegen:

- „CacheDefaultExpiry - Standardverfallszeit für Dateien angeben" auf Seite 191
- „CacheExpiryCheck - Angeben, ob der Server verfallene Dateien zurückgeben soll" auf Seite 192
- „CacheTimeMargin - Mindestlebensdauer für das Caching einer Datei angeben" auf Seite 199
- „CacheUnused - Zeitlimit für ungenutzte Dateien im Cache angeben" auf Seite 199
- „CacheNoConnect - Eigenständigen Cache-Modus angeben" auf Seite 197

Einstellungen für Faktor letzte Änderung

Im Formular **Faktor letzte Änderung** wird der Wert festgelegt, mit dem der Proxy-Server das Verfallsdatum für Cache-Dateien berechnet, die keine Verfallsdaten im Header enthalten. Sie können für Dateien, die mit verschiedenen Anforderungsschablonen übereinstimmen, unterschiedliche Werte definieren. Die erste Schablone, mit der eine Übereinstimmung erzielt wird, wird zur Berechnung des Verfallsdatums verwendet.

Informationen zum Festlegen des Faktors letzte Änderung (Last Modified) durch direktes Editieren der Konfigurationsdatei des Proxy-Servers finden Sie im Abschnitt „CacheLastModifiedFactor - Wert zur Berechnung der Verfallsdaten angeben" auf Seite 193.

Cache-Zeitlimit

Im Konfigurationsformular **Zeitlimit für Dateien im Cache** können Sie die maximale Verweildauer einer Datei im Cache definieren. Die Zeitlimits werden basierend auf den Anforderungsschablonen definiert. Sie können festlegen, dass Dateien nach Ablauf des Zeitlimits gelöscht oder neu überprüft werden sollen. Mit diesen Einstellungen können Dateien, deren Verfallsdaten ungültig sind, oder Dateien mit sehr langen Verfallszeiten verwaltet werden.

Informationen zum Festlegen der maximalen Verfallszeit für Dateien im Cache durch Editieren der Konfigurationsdatei des Proxy-Servers finden Sie in den folgenden Abschnitten:

- „CacheMaxExpiry - Maximale Lebensdauer für Cache-Dateien angeben" auf Seite 195
- „CacheClean - Zeitlimit für Dateien im Cache angeben" auf Seite 191

Garbage-Collection

Zu den Strategien, häufig aufgerufene URLs im Cache zu speichern und die Verwendung der Systemressourcen zu minimieren, gehört, dass Caching Proxy einen Bereinigungsprozess, die so genannte *Garbage-Collection* durchführt, bei dem alte oder nicht verwendete Dateien aus dem Cache entfernt werden, um Platz für aktuelle Dateien zu schaffen.

Bei der Garbage-Collection werden die Dateien im Cache-Verzeichnis zunächst überprüft. Die verfallenen Dateien werden gelöscht, um den Cache zu verkleinern und Platz für neue Dateien zu schaffen. Die Garbage-Collection wird automatisch durchgeführt. Sie können jedoch einige Einstellungen konfigurieren, um den Prozess an Ihre Voraussetzungen anzupassen.

Garbage-Collection konfigurieren

Zum Konfigurieren der Garbage-Collection wählen Sie in den Konfigurations- und Verwaltungsformularen **Cache-Konfiguration –> Einstellungen für Garbage-Collection** aus. In diesem Formular können Sie den *oberen Grenzwert* und den *unteren Grenzwert* definieren. Diese Werte legen fest, wann die Garbage-Collection gestartet und gestoppt wird. Wenn der vom Cache belegte Speicherplatz den als oberen Grenzwert festgelegten Prozentsatz erreicht oder überschreitet, wird die Garbage-Collection gestartet. Die Garbage-Collection wird so lange fortgesetzt, bis der Prozentsatz des durch den Cache belegten Speicherplatzes kleiner-gleich dem unteren Grenzwert ist.

Sie können zwischen zwei Algorithmen für die Garbage-Collection wählen. Der Algorithmus **responsetime** (Antwortzeit) optimiert die Antwortzeiten, indem vorzugsweise große Dateien aus dem Cache entfernt werden. Der Algorithmus **bandwidth** (Bandbreite) optimiert die Nutzung der Netzbandbreite, indem vorzugsweise kleinere Dateien aus dem Cache entfernt werden. Sie können einen der beiden Algorithmen oder eine Kombination aus beiden auswählen.

Informationen zum Konfigurieren der Garbage-Collection durch Editieren der Konfigurationsdatei des Proxy-Servers finden Sie in den Referenzabschnitten zu den folgenden Anweisungen:

- „Gc - Garbage-Collection angeben” auf Seite 222
- „GcHighWater - Den Beginn der Garbage-Collection festlegen” auf Seite 222
- „GcLowWater - Das Ende der Garbage-Collection festlegen” auf Seite 223
- „CacheAlgorithm - Den Cache-Algorithmus angeben” auf Seite 190

Kapitel 20. Den Cache-Agenten für automatische Aktualisierung und Vorabladen konfigurieren

Die meisten Proxy-Server, die Caching unterstützen, speichern eine Datei nur dann im Cache, wenn ein Benutzer sie anfordert. Caching Proxy verfügt über einen Cache-Agenten, der ein automatisches Vorabladen in den Cache durchführt. Sie können festlegen, dass der Cache-Agent bestimmte URLs automatisch abruft und/oder dass er die am häufigsten angeforderten URLs abruft und diese im Cache speichert, bevor sie angefordert werden.

In einigen Fällen müssen der Hostname des Proxy-Servers und das Cache-Zugriffsprotokoll angegeben werden, damit der Cache vorab geladen werden kann. Wählen Sie zum Konfigurieren des Cache-Agenten in den Konfigurations- und Verwaltungsformularen **Cache-Konfiguration** aus und rufen Sie dann die Formulare **Vorabladen des Cache** und **Cache-Aktualisierung** auf. Dateien, die Abfrageergebnisse enthalten (d. h. deren URLs das Fragezeichen (?) enthalten), werden nur dann zwischengespeichert, wenn das Caching von Abfragen aktiviert ist.

Die automatische Cache-Aktualisierung und das automatische Vorabladen des Cache bieten folgende Vorteile:

- Bestimmte URLs werden in den Cache geladen, bevor ein Benutzer die Seiten anfordert.
- Der Cache wird gefüllt, bevor der Server durch Benutzeraktivitäten in Anspruch genommen wird.
- Die Bereitstellung aktueller Dateien aus dem Cache ist schneller als beim Abruf der Dateien bei der ersten Anforderung.

Die Nachteile sind:

- Der Proxy-Server ist selbst zu Zeiten geringer Benutzeraktivität mit dem Caching von Seiten beschäftigt.
- Sie müssen steuern, welche Dateien automatisch geladen werden. Wenn verlinkte Dateien aus Seiten höherer Ebenen geladen werden, wie z. B. Webindizes und Suchseiten, kann dies dazu führen, dass eine große Anzahl Seiten angefordert wird.

Aus Gründen der Effizienz sollte der Agent zur Cache-Aktualisierung so konfiguriert werden, dass er ausgeführt wird, wenn der Server freie Kapazitäten aufweist und bevor der Server durch Clientanforderungen in Anspruch genommen wird. In diesem Fall stehen die Dateien im Cache bereit und können schnell abgerufen werden, wenn der Benutzer diese zum ersten Mal anfordert. Standardmäßig wird der Cache-Agent jede Nacht um 3 Uhr Ortszeit gestartet.

Anmerkung: Wenn der Cache-Agent zur Aktualisierung des Cache ausgeführt wird, müssen Sie in der Datei ibmproxy.conf in der Zeile ″Proxy http:*″ das Kommentarzeichen entfernen. Andernfalls erscheint im Fehlerprotokoll der Fehler ″403 Forbidden By Rule Error″, und die Cache-Aktualisierung wird nicht durchgeführt.

Hostnamen des Servers festlegen

Legen Sie auf Linux- und UNIX-Plattformen den Hostnamen des Proxy-Servers fest, dessen Cache vorab geladen oder aktualisiert werden soll. Legen Sie auf Windows-Plattformen den Hostnamen nur dann fest, wenn der zu aktualisierende Proxy-Server nicht auf der lokalen Maschine ausgeführt wird. (Beachten Sie, dass das Aktualisieren des Cache eines fernen Servers basierend auf den Dateien, auf die am häufigsten zugegriffen wird, nicht möglich ist, da der lokale Cache-Agent keinen Zugriff auf das Cache-Zugriffsprotokoll eines fernen Servers hat.)

Zum Festlegen des Hostnamens für den Proxy-Server wählen Sie in den Konfigurations- und Verwaltungsformularen **Cache-Konfiguration –> Cache-Aktualisierung: Cache-Zielserver angeben** aus.

Bestimmte Dateien vorab in den Cache laden

Um den Inhalt bestimmter URLs vorab in den Cache zu laden, wählen Sie in den Konfigurations- und Verwaltungsformularen **Cache-Konfiguration –> Vorabladen des Cache** aus. In diesem Formular können Sie die URLs angeben, die der Cache-Agent laden soll. Der Proxy-Server ruft diese Seiten ab, wenn der Cache-Agent gestartet wird, unabhängig davon, ob diese bereits im Cache enthalten waren. (Diese URLs werden in der Konfigurationsdatei des Proxy-Servers mit der Anweisung LoadURL angegeben.) Dieses Formular kann auch verwendet werden, um URLs zu definieren, deren Inhalt nicht im Cache gespeichert werden soll. Für diese Art des Vorabladens in den Cache ist kein Zugriff auf ein Cache-Zugriffsprotokoll erforderlich.

Mit dem Formular **Vorabladen des Cache** können folgende Optionen konfiguriert werden:

- **Cache täglich aktualisieren**: Wählen Sie dieses Markierungsfeld aus, wenn der Cache-Agent den Cache jede Nacht aktualisieren soll. Soll der Cache-Agent nicht gestartet werden, darf dieses Markierungsfeld nicht ausgewählt sein.
- **Cache-Aktualisierungszeit**: Wenn der Cache-Agent zu einem anderen Zeitpunkt als um 3:00 Uhr Ortszeit ausgeführt werden soll, geben Sie an, wann er gestartet werden soll.
- **Cache-Inhalte** - Geben Sie im Feld **URL oder IP-Adresse** die URLs an, die geladen werden sollen. Um URLs vom Vorabladen auszuschließen, geben Sie diese URLs an und klicken Sie im Feld **Cache-Status** auf **Ignorieren**.

Häufig aufgerufene Dateien vorab in den Cache laden

Damit die am häufigsten aufgerufenen Seiten automatisch vorab geladen werden, verwenden Sie das Formular **Cache-Konfiguration –> Cache-Aktualisierung**. Diese Funktion setzt voraus, dass der Proxy-Server über ein Cache-Zugriffsprotokoll verfügt. (Position und Name des Protokolls können geändert werden. Nähere Informationen finden Sie in Teil 6, „Caching Proxy überwachen”, auf Seite 147.) Die am häufigsten aufgerufenen URLs werden automatisch anhand des Cache-Zugriffsprotokolls ermittelt. Der Administrator kann darüber hinaus eine Anzahl von häufig verwendeten Seiten festlegen, die in den Cache geladen werden sollen. (Diese Anzahl wird in der Konfigurationsdatei des Proxy-Servers mit der Anweisung LoadTopCached festgelegt.)

Mit dem Formular **Cache-Aktualisierung** können Sie folgende Optionen konfigurieren:

- **Cache täglich aktualisieren**: Wählen Sie dieses Markierungsfeld aus, wenn der Cache-Agent den Cache jede Nacht aktualisieren soll. Soll der Cache-Agent nicht gestartet werden, darf dieses Markierungsfeld nicht ausgewählt sein.
- **Cache-Aktualisierungszeit**: Wenn der Cache-Agent zu einem anderen Zeitpunkt als um 3:00 Uhr Ortszeit ausgeführt werden soll, geben Sie die Zeit, zu der er gestartet werden soll, in Stunden und Minuten an.
- **Cache-Zielserver angeben**: Verwenden Sie diese Option, wenn ein anderer Server als die lokale Maschine aktualisiert werden soll. (Beachten Sie, dass häufig aufgerufene Seiten eines fernen Servers nicht vorab in den Cache geladen werden können.)
- **Zwischenspeichern der am häufigsten aufgerufenen URLs**: Geben Sie die Anzahl der URLs an, die aus dem Cache-Zugriffsprotokoll der letzten Nacht zwischengespeichert werden sollen.
- **Mit zwischengespeicherten URLs verknüpfte Seiten**: Verwenden Sie diese Einstellung, um die Delving-Funktion zu konfigurieren. (Eine Beschreibung der Delving-Funktion finden Sie im nächsten Abschnitt.) Geben Sie die Anzahl der Ebenen an, für die die Delving-Funktion durchgeführt werden soll, und legen Sie fest, ob die Delving-Funktion für alle Seiten (**always**), für keine Seiten (**never**), nur für vom Administrator angegebene Seiten (**admin**) oder nur für häufig aufgerufene Seiten (**topn**) durchgeführt werden soll. Legen Sie außerdem fest, ob die Delving-Funktion hostübergreifend durchgeführt werden soll, ob eine Verzögerung zwischen den einzelnen Anforderungen eingefügt werden soll und ob Inline-Grafiken im Cache gespeichert werden sollen.
- **Anzahl der Threads**: Legen Sie die maximale Anzahl Threads fest, die für die Aktualisierung des Cache verwendet werden soll.
- **Maximale Länge der Bearbeitungswarteschlange**: Legen Sie die maximale Größe der Warteschlange für anzufragende URLs fest.
- **Maximal anzufragende URLs**: Legen Sie die maximale Anzahl Seiten fest, die geladen werden soll. Diese Anzahl wird geprüft, bevor Seiten über die Delving-Funktion abgerufen werden.
- **Maximale Zeit**: Legen Sie die maximale Ausführungszeit für den Cache-Agenten fest. Wenn als Zeitspanne 0 Stunden 0 Minuten festgelegt werden, wird der Cache-Agent vollständig ausgeführt.

Delving

Delving ist eine optionale Teilfunktion der Funktion für automatische Cache-Aktualisierung. Die meisten Webseiten enthalten Links zu anderen Seiten mit zugehörigen Informationen. Benutzer folgen oft den Link-Pfaden von einer Seite zu einer anderen und von einer Site zu einer anderen. Mit Delving können diese logischen Informationspfade im Cache gespeichert werden. Beim Delving folgt der Cache-Agent einer festgelegten Anzahl von HTML-Links auf den Seiten, die er lädt, und lädt auch alle diese über Links verknüpften Seiten in den Cache. Die über Links verknüpften Seiten können sich auf demselben Host wie die Ursprungsseite oder auf anderen Hosts befinden. Abb. 1 auf Seite 98 zeigt eine grafische Darstellung dieser Links.

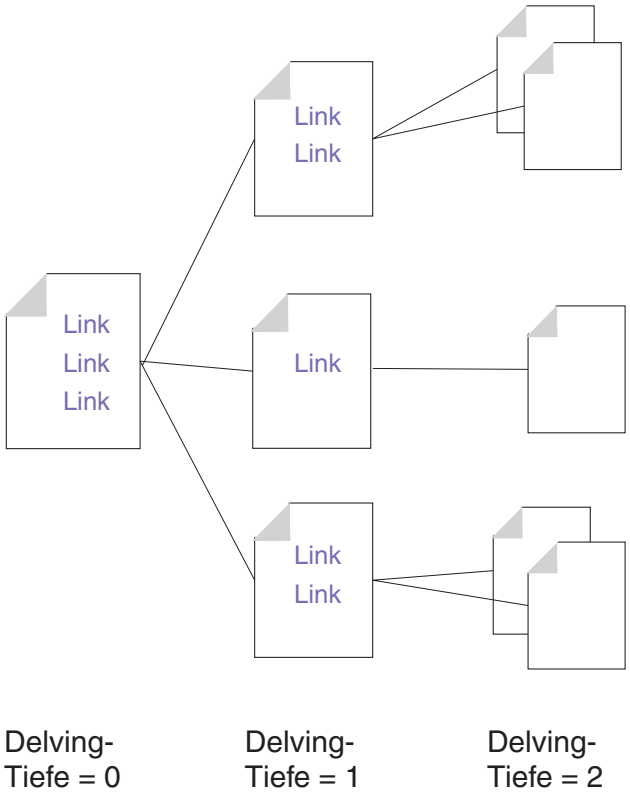

Abbildung 1. Delving

Zum Steuern des Delving-Prozesses legt der Administrator für den Cache-Agenten eine maximale Anzahl URLs fest, die dieser laden kann (die Standardeinstellung ist 2000), sowie eine maximale Zeitdauer für die Ausführung (die Standardeinstellung ist zwei Stunden) und eine maximale Anzahl Threads, die verwendet werden kann (die Standardeinstellung ist vier). Darüber hinaus kann der Administrator weitere Steuerelemente konfigurieren. Standardmäßig ist Delving für zwei Hierarchieebenen aktiviert, aber nicht hostübergreifend möglich. Zusätzlich dazu wird zwischen den einzelnen Anforderungen eine Verzögerung eingefügt. Informationen zum Ändern dieser Einstellungen enthält der Abschnitt „Zugehörige Anweisungen in der Konfigurationsdatei des Proxy-Servers” auf Seite 99.

Der Cache-Agent lädt und aktualisiert den Cache anschließend in dieser Reihenfolge:

1. Er lädt spezielle Seiten, die vom Administrator angegeben wurden.
2. Er lädt häufig aufgerufene Seiten aus dem Cache-Zugriffsprotokoll.
3. Wenn die maximale Anzahl Seiten dadurch nicht erreicht wird, werden zusätzliche Seiten durch Delving geladen.

Beachten Sie, dass der Cache-Agent erst beim Starten des Link-übergreifenden Delving prüft, ob die maximale Anzahl Seiten erreicht wurde. Wenn der Wert für die maximale Anzahl Seiten (in der Konfigurationsdatei des Proxy-Servers mit MaxURLs festgelegt) kleiner ist als die Anzahl Seiten, die in den Schritten 1 und 2 bereits abgerufen wurden, werden keine weiteren über Links verknüpften Seiten abgerufen.

Die folgenden Beispiele veranschaulichen, wie der Cache-Agent die Prioritäten für die Cache-Aktualisierung und die Delving-Funktion abhängig von der festgelegten maximalen Anzahl URLs behandelt. (Dabei wird angenommen, dass die Delving-Funktion für alle diese Beispiele konfiguriert ist.)

Einstellung in der Konfigurationsdatei	Ergebnis

`LoadURL`
` http://www.getthis.com/main.html`
`LoadURL`
` http://www.getmetoo.com/welcome.htm`
`LoadTopCached 30`
`MaxURLs 50`
	Wenn das Cache-Zugriffsprotokoll mehr als 30 eindeutige URLs enthält, ruft der Cache-Agent die Seiten main.html, welcome.htm und die 30 URLs ab, die gemäß dem Cache-Zugriffsprotokoll am häufigsten angefordert wurden. Da der Wert für MaxURLs nicht erreicht wurde, ruft der Cache-Agent ausgehend von den Seiten, die bereits im Cache gespeichert wurden, bis zu 18 über Links verknüpfte URLs ab und lädt sie.

`LoadURL`
` http://ww.joesmith.edu/favorites.html`
`LoadURL`
` http://www.janesmith.edu/dislikes.html`
`LoadTopCached 30`
`MaxURLs 25`
	Wenn das Cache-Zugriffsprotokoll mehr als 30 eindeutige URLs enthält, ruft der Cache-Agent die Seiten favorites.html und dislikes.html sowie die 30 URLs ab, die gemäß dem Cache-Zugriffsprotokoll am häufigsten angefordert wurden. Es werden keine anderen Dateien abgerufen, da der Wert von MaxURLs erreicht wurde.

`LoadURL http://www.hello.com/hi.htm`
`LoadURL`
` http://www.ballyhoo.com/index.html`
`LoadTopCached 20`
`MaxURLs 25`
	Wenn das Cache-Zugriffsprotokoll mehr als 20 eindeutige URLs enthält, ruft der Cache-Agent die Seiten **hi.htm** und **index.html**, die 20 URLs, die gemäß dem Cache-Zugriffsprotokoll am häufigsten angefordert wurden, und 3 über Links verknüpfte URLs von bereits abgerufenen Seiten ab. Es werden keine anderen Dateien abgerufen, da der Wert von MaxURLs erreicht wurde.

Zugehörige Anweisungen in der Konfigurationsdatei des Proxy-Servers

Der Cache-Agent kann auch durch direktes Editieren der entsprechenden Anweisungen in der Konfigurationsdatei des Proxy-Servers konfiguriert werden. Informationen zu den Anweisungen in der Konfigurationsdatei des Proxy-Servers, die sich auf den Cache-Agenten beziehen, finden Sie auf den folgenden Referenzseiten in Anhang B, „Anweisungen in der Konfigurationsdatei", auf Seite 175:

- „AutoCacheRefresh - Angeben, ob Cache-Aktualisierung verwendet werden soll" auf Seite 188
- „CacheAccessLog - Pfad für Cache-Zugriffsprotokolldateien angeben" auf Seite 189
- „CacheRefreshTime - Startzeitpunkt für Cache-Agenten angeben" auf Seite 198
- „DelayPeriod - Verzögerungen zwischen Anforderungen angeben" auf Seite 207
- „DelveAcrossHosts - Domänenübergreifendes Caching angeben" auf Seite 207
- „DelveDepth - Angeben, wie weit den Links beim Caching gefolgt werden soll" auf Seite 207
- „DelveInto - Angeben, ob der Cache-Agent den Links folgen soll" auf Seite 208
- „IgnoreURL - URLs angeben, die nicht aktualisiert werden sollen" auf Seite 227

- „LoadInlineImages - Aktualisierung eingebetteter Grafiken steuern” auf Seite 233
- „LoadTopCached - Die Anzahl der am häufigsten angeforderten Seiten angeben, die aktualisiert werden sollen” auf Seite 234
- „LoadURL - Die URLs angeben, die aktualisiert werden sollen” auf Seite 234
- „MaxUrls - Die maximale Anzahl URLs angeben, die aktualisiert werden sollen” auf Seite 241

Den Cache-Agenten manuell starten

Wenn die automatische Cache-Aktualisierung aktiviert ist, führt der Cache-Agent zur angegebenen Zeit automatisch eine Cache-Aktualisierung durch. Sie können den Cache-Agenten jedoch auch zu einem beliebigen Zeitpunkt von einer Befehlszeile aus starten.

Verwenden Sie hierfür die folgende ausführbare Datei:

- Auf Linux- und UNIX-Plattformen: usr/sbin/cacheagt
- Auf Windows-Plattformen: *Serverstammverzeichnis*\bin\cacheagt.exe

  Dabei steht *Serverstammverzeichnis* für das Laufwerk und das Verzeichnis, in dem Caching Proxy installiert wurde (z. B. C:\Programme\IBM\edge\cp).

Auf Linux- und UNIX-Plattformen können Sie den Cache-Agenten über den Dämon **cron** automatisch zu verschiedenen Zeitpunkten ausführen. Über **cron** gesteuerte Jobs werden durch Hinzufügen einer Zeile zur Systemdatei crontab angegeben. Beispiel für einen Eintrag in der Befehlsdatei unter Linux und UNIX:

```
45 16 * * * /usr/sbin/cacheagt
```

Mit diesem Befehl wird der Cache-Agent jeden Tag um 16:45 Ortszeit gestartet. Wenn der Cache-Agent mehrmals täglich ausgeführt werden soll, können Sie mehrere Einträge verwenden. Nähere Informationen hierzu finden Sie in der Dokumentation zum Betriebssystem zum Thema Dämon **cron**.

Wenn Sie den Dämon **cron** verwenden, um den Cache-Agenten auszuführen, achten Sie darauf, dass Sie die Option für automatische Aktualisierung entweder über das Konfigurationsformular **Cache-Konfiguration –> Cache-Aktualisierung** oder durch Editieren der Konfigurationsdatei des Proxy-Servers inaktivieren. Andernfalls wird der Cache-Agent mehr als einmal täglich ausgeführt.

Kapitel 21. Gemeinsamen Cache verwenden

Es ist nicht ungewöhnlich, dass ein Zugangspunkt (Point of Presence) im Web mehr Datenverkehr verarbeiten muss, als ein Server bewältigen kann. Dieses Problem kann durch das Hinzufügen von Servern gelöst werden. Bei der Verwendung mehrerer Proxy-Server, die Caching unterstützen, können sich die Inhalte der Caches überlappen. Die Überlappung hat neben unnötigen Redundanzen zur Folge, dass keine maximalen Bandbreiteneinsparungen erzielt werden, weil Dateien im Cache erneut von den Ursprungsservern abgerufen werden müssen, wenn ein Proxy-Server Anforderungen für diese Dateien erhält und diese nicht aus seinem eigenen Cache bedienen kann. Obwohl mehrfaches Caching durch Verwendung einer hierarchischen Kette von Proxy-Servern minimiert werden kann, erfordert dies zusätzliche Datenübertragungen über einen bestimmten Server, und jeder zusätzliche Link in der Kette bedeutet eine zusätzliche Latenzzeit.

Diese Probleme können durch die Verwendung eines gemeinsamen Cache gelöst werden, indem jedem Cache erlaubt wird, seine Inhalte anderen Caches zur Verfügung zu stellen. Bandbreite kann durch folgende Tatsachen eingespart werden:

- Objekte werden nicht mehrfach abgerufen.
- Mit einem größeren kombinierten logischen Cache wird eine höhere Trefferquote erzielt.

Es werden zwei Methoden bereitgestellt, mit denen mehrere Caches so verwendet werden können, als wären sie ein einziger logischer Cache:

- Remote Cache Access (RCA) ist eine Funktion von Caching Proxy, die einen Bereich von Member-Caches definiert. Gemäß einer internen Logik wird eine Datei in genau einem dieser Caches gespeichert.
- Es wird ein Plug-in für Caching Proxy bereitgestellt, das dem Proxy-Server die Verwendung von Internet Caching Protocol (ICP) ermöglicht. Sie können das Plug-in ICP anstelle von RCA verwenden, wenn Sie Daten auf Caching-Proxy-Maschinen und Nicht-Caching-Proxy-Maschinen gemeinsam benutzen möchten.

RCA und ICP können zusammen verwendet werden.

Ferner Cache-Zugriff

Beachten Sie bei der Planung eines fernen Cache-Zugriffs die folgenden Empfehlungen:

- Die beteiligten Proxy-Server sollten sich nah beieinander befinden und über Verbindungen mit hoher Bandbreite miteinander verbunden sein (z. B. FDDI, SP2-Bus).
- Die Zugehörigkeit zum RCA-Bereich muss langfristig sein, damit die Konfiguration so stabil wie möglich ist.
- Proxy-Server müssen ein ähnliches Leistungsspektrum aufweisen (z. B. CPU, Größe des Hauptspeichers und des Cache).
- Netzausfälle dürfen nur selten auftreten.
- Es müssen weniger als 100 Member in einem Bereich sein.
- Alle Member des Bereichs müssen dieselbe Version der Software Caching Proxy verwenden.

Anmerkung: Wenn die Proxy-Server im RCA-Bereich (RCA = Remote Cache Access, ferner Cache-Zugriff) unterschiedliche Linux-Betriebssysteme verwenden (z. B. SuSE und Red Hat), stellen Sie sicher, dass der Benutzer ″nobody″ auf allen Peers dieselbe UID hat. Prüfen Sie das Kennwort und die Einträge für die Gruppendatei im Verzeichnis /etc/ auf jedem Computer und ordnen Sie ″nobody″ dieselbe UID zu.

Es sollte auf den fernen Cache-Zugriff verzichtet werden, falls eine dieser Bedingungen nicht erfüllt ist oder unterschiedliche Organisationen unterschiedliche Server verwalten, die Member des RCA-Bereichs sind.

Fernen Cache-Zugriff konfigurieren

Zum Konfigurieren des fernen Cache-Zugriffs wählen Sie in den Konfigurations- und Verwaltungsformularen **Cache-Konfiguration –> Ferner Cache-Zugriff** aus. Die Felder in diesem Formular definieren einen benannten Bereich, der einen einzigen logischen Cache verwendet. Geben Sie die erforderlichen Informationen für jedes Member des Bereichs ein.

Informationen zum Konfigurieren des fernen Cache-Zugriffs durch Editieren der Konfigurationsdatei des Proxy-Servers finden Sie in Anhang B, „Anweisungen in der Konfigurationsdatei”, auf Seite 175 in den Referenzabschnitten zu folgenden Anweisungen:

- „ArrayName - Name des fernen Cache-Bereichs” auf Seite 187
- „Member - Ein Member eines Bereichs angeben” auf Seite 241

Plug-in Internet Caching Protocol (ICP) konfigurieren

Das Plug-in Internet Caching Protocol (ICP) erlaubt Caching Proxy, bei der Suche nach HTML-Seiten oder anderen Ressourcen, die im Cache gespeichert werden können, ICP-kompatible Caches abzufragen. Wenn der Proxy-Server eine HTTP-Anforderung empfängt, durchsucht er seinen eigenen Cache nach der Ressource. Wird die Ressource im lokalen Cache nicht gefunden und ist das Plug-in ICP aktiviert, packt der Proxy-Server den URL in ein ICP-Abfragepaket und übermittelt dieses Paket an alle identifizierten ICP-Peer-Caches. Falls ein Peer-Cache antwortet, dass er die Ressource besitzt, ruft der Proxy-Server die Ressource aus dem Cache dieses Peer ab. Antworten zwei oder mehr Peers positiv, wird die erste Antwort verarbeitet. Falls keiner der Peers mit einem Treffer antwortet, wird die Anforderung vom ursprünglichen Server gemäß seines definierten Arbeitsablaufs verarbeitet. Der Proxy-Server kann beispielsweise ein anderes Plug-in aufrufen, mit der RCA-Routine fortfahren (sofern RCA aktiviert ist) oder die angeforderte Ressource selbst abrufen.

Plug-in ICP konfigurieren

Sie können das Plug-in ICP in der Konfigurationsdatei des Proxy-Servers, ibmproxy.conf, aktivieren und konfigurieren. Die Anweisung ServerInit und/oder die Anweisung PreExit muss in der Konfigurationsdatei zum Abschnitt mit den API-Anweisungen hinzugefügt werden, damit das Plug-in ICP verwendet werden kann. Welche Anweisung verwendet wird, ist vom Aufgabenbereich von Caching Proxy im ICP-System abhängig:

- Damit Caching Proxy als ICP-Server eingesetzt wird, verwenden Sie die Anweisung ServerInit, um das Modul icpServer aufzurufen.
- Damit Caching Proxy als ICP-Client eingesetzt wird, verwenden Sie die Anweisung PreExit, um das Modul icpClient aufzurufen.

- Damit Caching Proxy als ICP-Client und als ICP-Server eingesetzt wird, verwenden Sie beide Anweisungen.
- Konfigurieren Sie mit den Anweisungen icpAddress, icpMaxThreads, icpPeer, icpPort und icpTimeout die Einstellungen, die das Plug-in verwenden soll.

Zum Erstellen dieser Anweisungen können Sie die Datei ibmproxy.conf manuell editieren oder, falls der Proxy-Server bereits aktiv ist, eine Verbindung zum Konfigurations- und Verwaltungsformular **Serverkonfiguration–> Anforderungsverarbeitung –> Verarbeitung von API-Anforderungen** herstellen.

Beachten Sie, dass die Prototypanweisungen (in Form von Kommentaren) zum API-Abschnitt der Datei ibmproxy.conf hinzugefügt wurden. Diese API-Anweisungen sind in einer zweckmäßigen Reihenfolge angeordnet. Wenn Sie API-Anweisungen hinzufügen, um neue Funktionen und Plug-in-Module zu aktivieren, sollten Sie die Anweisungen wie im Prototypabschnitt der Konfigurationsdatei gezeigt anordnen. Alternativ dazu können Sie, falls erforderlich, die Kommentarzeichen für API-Anweisungen entfernen und API-Anweisungen editieren, um die Unterstützung für jede gewünschte Funktion oder jedes gewünschte Plug-in hinzuzufügen.

Die Anweisungen ServerInit und PreExit unterstützen zwei Argumente: (1) den vollständig qualifizierten Pfad der gemeinsam benutzten Bibliothek und (2) den Funktionsaufruf. Diese Argumente werden durch einen Doppelpunkt (:) getrennt. Das erste Argument ist systemspezifisch und abhängig davon, wo die Plug-in-Komponenten installiert sind. Das zweite Argument ist in der gemeinsam genutzten Bibliothek fest codiert und muss wie gezeigt eingegeben werden.

Jede Anweisung in der Konfigurationsdatei des Proxy-Servers muss in einer gesonderten Zeile stehen.

```
ServerInit Pfad_der_gemeinsam_benutzten_Bibliothek:icpServer
```

Linux- und UNIX-Systeme:

```
ServerInit /opt/ibm/edge/cp/internet/lib/plugins/icp/libicp_plugin.so:icpServer
```

Beispiel für Windows:

```
ServerInit C:\Programme\IBM\edge\cp\Bin\plugins\icp\icpplugin.dll:icpServer
PreExit Pfad_der_gemeinsam_benutzten_Bibliothek:icpClient
```

Linux- und UNIX-Systeme:

```
PreExit /opt/ibm/edge/cp/internet/lib/plugins/icp/libicp_plugin.so:icpClient
```

Beispiel für Windows:

```
PreExit C:\Programme\IBM\edge\cp\Bin\plugins\icp\icpplugin.dll:icpClient
```

Zum Konfigurieren der Plug-in-Einstellungen können Sie die ICP*-Anweisungen in der Konfigurationsdatei des Proxy-Servers hinzufügen oder ändern. Nähere Informationen hierzu finden Sie in den Beschreibungen der folgenden Anweisungen.

- „ICP_Address — IP-Adresse für ICP-Abfragen angeben" auf Seite 225
- „ICP_MaxThreads - Maximale Anzahl Threads für ICP-Abfragen angeben" auf Seite 226
- „Occupier - Member eines ICP-Cluster angeben" auf Seite 226
- „ICP_Port — Port-Nummer für ICP-Abfragen angeben" auf Seite 227
- „ICP_Timeout — Maximale Wartezeit für ICP-Abfragen angeben" auf Seite 227
- „PreExit - Schritt "PreExit" anpassen" auf Seite 253
- „ServerInit - Schritt "Server Initialization" anpassen" auf Seite 275

Kapitel 22. Caching von dynamisch erstelltem Inhalt

Die Funktion für dynamisches Caching ermöglicht Caching Proxy, dynamisch generierten Inhalt in Form von Antworten von JavaServer Pages (JSP) und Servlets, die von einem IBM WebSphere Application Server erstellt wurden, zwischenzuspeichern. Im Anwendungsserver wird ein Caching-Proxy-Adaptermodul verwendet, damit die Antworten so abgeändert werden, dass sie sowohl im Cache des Proxy-Servers als auch im dynamischen Cache des Anwendungsservers gespeichert werden können. Auf diese Weise ist es möglich, dynamisch erstellten Inhalt am Rand des Netzes im Cache zu speichern und den Inhaltshost von der Aufgabe zu befreien, wiederholt Anforderungen an den Anwendungsserver zu senden, wenn mehrere Clients denselben Inhalt anfordern.

Anmerkung: Die Funktion für dynamisches Caching aktiviert den Proxy-Server nicht für die Zwischenspeicherung der Ergebnisse von URL-Abfragen. Das Caching von Abfrageergebnissen muss mit Caching-Filtern konfiguriert werden, die in Kapitel 18, „Zwischenzuspeichernde Inhalte steuern", auf Seite 85 und in der Referenzdokumentation zur Anweisung „CacheQueries - Cache-Antworten auf URLs festlegen, die ein Fragezeichen (?) enthalten" auf Seite 197 beschrieben sind. Die Abfrageergebnisse von Ursprungsservern, die keine IBM WebSphere Application Server sind, können zwischengespeichert werden.

Manchmal muss das Caching von Abfragen aktiviert werden, damit die Funktion für dynamisches Caching funktioniert, z. B. wenn die Servlets URLs in Form von Abfragen verwenden. Der Proxy-Server betrachtet jeden URL, der ein Fragezeichen (?) enthält, als Abfrage.

Das Caching von dynamisch generiertem Inhalt bietet folgende Vorteile:

- Die Arbeitslast der Inhaltshosts wird reduziert.
- Die Arbeitslast der Anwendungsserver wird reduziert.
- Die angeforderten Ressourcen werden den Endbenutzern schneller bereitgestellt.
- Die Bandbreitenbelegung zwischen Servern wird reduziert.
- Die Skalierbarkeit von Websites, die dynamisch generierten Inhalt erstellen oder bereitstellen, wird erhöht.

Der Anwendungsserver exportiert nur vollständig erstellte öffentliche Seiten für das Caching durch den Proxy-Server. Private Seiten werden vom Proxy-Server nicht im Cache gespeichert. Beispielsweise kann eine dynamisch generierte Seite von einer öffentlichen Site, die Wettervorhersagen enthält, von IBM WebSphere Application Server exportiert und von Caching Proxy im Cache gespeichert werden. Andererseits kann eine dynamisch generierte Seite, die den Inhalt des elektronischen Warenkorbs eines Benutzers enthält, vom Proxy-Server nicht im Cache gespeichert werden. Damit eine dynamisch generierte Seite im Cache gespeichert werden kann, müssen auch alle Teilkomponenten dieser Seite im Cache speicherbar sein.

Dynamische Dateien im Cache verfallen nicht wie reguläre Dateien, sondern müssen von dem Anwendungsserver, von dem sie generiert wurden, ungültig gemacht (invalidiert) werden.

Die Einträge im dynamischen Cache werden in folgenden Fällen ungültig gemacht:

- Der Garbage Collector für den dynamischen Cache entfernt einen Eintrag aufgrund einer Cache-Überlastung.
- Das im Servlet-Eintrag (servletcache.xml) oder in der Anweisung ExernalCacheManager des Proxy-Servers festgelegte Zeitlimit läuft ab.
- Ein externer Agent oder eine externe Anwendung ruft die APIs des dynamischen Cache zwecks Invalidierung von Cache-Einträgen auf.

Einträge des dynamischen Cache werden ungültig gemacht, indem für die spezielle Instanz des Caching-Proxy-Plug-in für dynamisches Caching eine Invalidierungsnachricht generiert wird. Caching Proxy empfängt diese Nachricht als Post-Nachricht an /WES_External_Adapter. Daraufhin löscht Caching Proxy die ungültigen Einträge aus seinem Cache.

Für das dynamische Caching sind folgende Konfigurationsschritte erforderlich.

- Konfiguration von IBM WebSphere Application Server
  - Konfigurieren Sie jeden Anwendungsserver für lokales dynamisches Caching.
  - Konfigurieren Sie jeden Anwendungsserver für die Verwendung des Adapters für externen Cache.
  - Legen Sie fest, welche externen Caches für jedes im Cache speicherbare Servlet und jede im Cache speicherbare JSP verwendet werden können.
- Konfiguration von Caching Proxy
  - Aktivieren Sie Caching Proxy für die Verwendung des Plug-in für dynamisches Caching.
  - Legen Sie die Quellen fest, aus denen dynamischer Inhalt zwischengespeichert werden soll.

IBM WebSphere Application Server für Proxy-Caching konfigurieren

Dynamisches Caching im Anwendungsserver konfigurieren

Befolgen Sie die Anweisungen in der Dokumentation zu IBM WebSphere Application Server, um Ihren Anwendungsserver für die Verwendung des lokalen dynamischen Cache (auch Cache für dynamische Fragmente genannt) zu konfigurieren. Der Cache für dynamische Fragmente arbeitet mit dem externen Cache von Application Server Caching Proxy zusammen.

Adapter für den Anwendungsserver konfigurieren

IBM WebSphere Application Server kommuniziert mit Caching Proxy über ein Softwaremodul, das als Adapter für externen Cache (External Cache Adapter) bezeichnet und mit Application Server installiert wird.

Anmerkung: Auf der Support-Website zu IBM WebSphere Application Server finden Sie technische Hinweise zum Konfigurieren von dynamischem Caching.

Caching Proxy für dynamisches Caching konfigurieren

Damit der Proxy-Server dynamisch generierten Inhalt (Ergebnisse von Servlets und JSPs) zwischenspeichern kann, müssen Sie in der Konfigurationsdatei des Proxy-Servers, ibmproxy.conf, zwei Änderungen vornehmen. Die erste Änderung aktiviert das Plug-in-Modul für dynamisches Caching, und die zweite Änderung konfiguriert dieses Plug-in-Modul in der Weise, dass es die Quellen des dynamischen Inhalts, der zwischengespeichert werden kann, erkennt.

Mit der Anweisung Service das Plug-in für dynamisches Caching aktivieren

Das Plug-in für dynamisches Caching wird mit einer API-Anweisung für den Schritt "Service" aktiviert. Zum Erstellen dieser Anweisung können Sie entweder die Datei ibmproxy.conf manuell editieren, oder, falls der Proxy-Server bereits aktiv ist, in den Konfigurations- und Verwaltungsformularen Folgendes auswählen: **Serverkonfiguration –> Anforderungsverarbeitung –> Verarbeitung von API-Anforderungen**. Der Inhalt der Anweisung wird in Beispielen weiter hinten in diesem Abschnitt gezeigt.

Eine Prototypanweisung Service zur Aktivierung des dynamischen Caching ist als Kommentar im API-Abschnitt der Datei ibmproxy.conf enthalten. Sie hat den Titel `JSP Plug-in`. Beachten Sie, dass die API-Prototypanweisungen in einer zweckmäßigen Reihenfolge angeordnet sind. Wenn Sie API-Anweisungen hinzufügen, um neue Funktionen und Plug-in-Module zu aktivieren, sollten Sie die Anweisungen wie im Prototypabschnitt der Konfigurationsdatei gezeigt anordnen. Alternativ dazu können Sie die Kommentarzeichen vor den API-Prototypanweisungen entfernen und sie gegebenenfalls editieren, um die Unterstützung für eine gewünschte Funktion oder ein Plug-in hinzuzufügen.

Definieren Sie die Anweisung Service wie in den folgenden Beispielen gezeigt. (Beachten Sie, dass jede Anweisung in der Konfigurationsdatei des Proxy-Servers in einer Zeile stehen muss, auch wenn die folgenden Beispiele aus Gründen der besseren Lesbarkeit Zeilenumbrüche enthalten.)

- Für AIX:

```
Service  /WES_External_Adapter  /opt/ibm/edge/cp/lib/plugins/
dynacache/libdyna_plugin.o:exec_dynacmd
```

- Für Solaris:

```
Service  /WES_External_Adapter  /opt/ibm/edge/cp/lib/plugins/
dynacache/libdyna_plugin.so:exec_dynacmd
```

- Für Linux:

```
Service  /WES_External_Adapter  /usr/lib/libdyna_plugin.so:exec_dynacmd
```

- Für Windows:

```
Service  /WES_External_Adapter  C:\Programme\IBM\edge\cp\bin\plugins\
dynacache\dyna_plugin.dll:exec_dynacmd
```

Falls Sie die Software Caching Proxy nicht im Standardverzeichnis installiert haben, verwenden Sie Ihren Installationspfad.

Mit der Anweisung ExternalCacheManager die Dateiquellen angeben

Jeder Caching-Proxy-Server muss so konfiguriert sein, dass er die Quelle der dynamisch generierten Dateien erkennt. Fügen Sie für jeden Anwendungsserver, der dynamisch generierten Inhalt auf dem Proxy-Server zwischenspeichert, eine Anweisung ExternalCacheManager zur Datei ibmproxy.conf hinzu. Diese Anweisung gibt einen WebSphere Application Server an, der Ergebnisse auf dem Proxy-Server zwischenspeichert, und definiert eine maximale Verfallszeit für den Inhalt, der von diesem Server stammt. Ausführliche Informationen hierzu finden Sie im Abschnitt „ExternalCacheManager - Caching Proxy für dynamisches Caching vom IBM WebSphere Application Server konfigurieren" auf Seite 219.

Die Server-ID, die in der Anweisung ExternalCacheManager verwendet wird, muss mit der Gruppen-ID übereinstimmen, die in der Zeilengruppe für externe Cache-Gruppen in der Datei dynacache.xml des Anwendungsservers angegeben ist.

Fügen Sie für das vorherige Beispiel folgenden Eintrag zur Datei ibmproxy.conf eines jeden Proxy-Servers hinzu:

```
ExternalCacheManager   IBM-edge-cp-XYZ-1  20 seconds
```

Caching Proxy speichert nur den Inhalt eines IBM WebSphere Application Server im Cache, dessen Gruppen-ID mit einem ExternalCacheManager-Eintrag in der Datei ibmproxy.conf übereinstimmt.

Kapitel 23. Proxy-Server-Cache optimieren

Wenn das Caching aktiviert ist, ist die Geschwindigkeit der Cache-Speichereinheiten für die Leistung von Caching Proxy von entscheidender Bedeutung. Dieser Abschnitt enthält Empfehlungen zur Auswahl der Cache-Speicherart und zur optimalen Konfiguration der Cache-Speichereinheiten.

Speichermedium für den Cache auswählen

Caching Proxy kann zwei verschiedene Arten von Cache-Speichermedien verwenden:

- Hauptspeicher
- Unformatierte Plattenpartitionen

Ein Speicher-Cache ist die schnellste Methode zum Abrufen von Dateien, aber die Größe eines solchen Cache wird durch die Größe des verfügbaren Speichers auf der Proxy-Server-Maschine beschränkt. Ein Platten-Cache, der sich aus einer oder mehreren unformatierten Plattenpartitionen zusammensetzt, ist langsamer als ein Speicher-Cache, unterstützt jedoch in den meisten Fällen größere Caches.

Die Leistung des Platten-Cache optimieren

Die für das Platten-Caching verwendeten Einheitenpartitionen müssen für den Cache reserviert werden, d. h. verwenden Sie diese physischen Platten nicht für andere Dateisysteme und für keinen anderen Zweck als zum Speichern des Proxy-Cache. Verwenden Sie außerdem auf keiner der für den Proxy-Cache vorgesehenen Platten Datenkomprimierung. Dadurch wird die Leistung beeinträchtigt.

Jede Cache-Speichereinheit (ob Platte oder Datei) bedeutet zusätzlichen Hauptspeicherbedarf für den Proxy-Server. Im Allgemeinen wird bei Verwendung eines vollständigen physischen Datenträgers als Cache-Einheit die beste Leistung erzielt. Der Einsatz von RAID oder anderen Methoden zum Kombinieren mehrerer physischer Datenträger zu einem einzelnen logischen Datenträger kann eine Verschlechterung der Leistung zur Folge haben. Wenn Sie mehrere Platten verwenden möchten, definieren Sie diese im Konfigurationsformular **Cache-Einstellungen** oder durch Editieren der Anweisung CacheDev in der Konfigurationsdatei des Proxy-Servers einzeln als Cache-Einheiten. Diese Methode ermöglicht dem Proxy-Server, die gleichzeitigen Lese- und Schreibvorgänge auf mehreren Datenträgern zu steuern, ohne dabei von der Leistung des Betriebssystems oder eines Plattensubsystems abhängig zu sein.

Garbage-Collection im Cache

Bei der Garbage-Collection des Proxy-Server-Cache werden verfallene Dateien aus dem Cache gelöscht, wodurch Speicher für das Caching von Dateien für neue Anforderungen freigemacht wird. Die Garbage-Collection wird automatisch ausgelöst, wenn die Auslastung des Speichers den vom Administrator festgelegten Grenzwert erreicht, der als *oberer Grenzwert* bezeichnet wird. Sie wird fortgesetzt, bis der *untere Grenzwert* für die Auslastung des Speichers erreicht ist.

Da bei der Garbage-Collection ein Minimum an CPU-Ressourcen verbraucht wird und die Verfügbarkeit der noch nicht verfallenen, zwischengespeicherten Dateien nicht beeinträchtigt wird, ist es nicht erforderlich, die Garbage-Collection zu bestimmten Zeiten durchzuführen.

Zur Verbesserung der Leistung der Garbage-Collection können Sie einen oberen Grenzwert und einen unteren Grenzwert festlegen. Sie können außerdem den Algorithmus konfigurieren, den die Garbage-Collection verwenden soll. Nähere Informationen zum Ändern der Garbage-Collection finden Sie im Abschnitt „Garbage-Collection" auf Seite 94.

Plattformspezifische Optimierungen

Nachfolgend sind zusätzliche Vorschläge für die Optimierung der Cache-Leistung auf jeder Plattform aufgeführt.

AIX

Erstellen Sie einen logischen Datenträger auf einer Platte, vorzugsweise unter Einbeziehung aller verfügbaren physischen Partitionen. Erstellen Sie beispielsweise auf einer 9-GB-Platte einen logischen Datenträger mit einer Größe von 9 GB und dem Namen cpcache1. Formatieren Sie ihn und legen Sie ihn als Proxy-Cache-Einheit fest. Verwenden Sie dazu den unformatierten logischen Datenträger /dev/rcpcache1.

HP-UX und Solaris

Erstellen Sie auf der Cache-Einheit eine Partition (oder einen Sektor) in der Gesamtgröße des Datenträgers. Beispiel: Erstellen Sie auf einer 9-GB-Platte eine 9-GB-Partition mit dem Namen c1t3d0s0. Formatieren Sie die Partition und legen Sie sie als Proxy-Cache-Einheit fest. Verwenden Sie dazu die unformatierte Einheit (Raw Device) /dev/rdsk/c1t3d0s0.

Windows

Erstellen Sie eine Partition in der Gesamtgröße des Datenträgers. Beispiel: Erstellen Sie auf einer 9-GB-Platte eine 9-GB-Partition mit dem Namen i:. Formatieren Sie die Partition und legen Sie sie als Proxy-Cache-Einheit fest. Verwenden Sie dazu die unformatierte Einheit \\.\i:.

Informationen zum Konfigurieren des Proxy-Server-Cache und zum Formatieren und Definieren der Cache-Einheiten finden Sie in Teil 4, „Proxy-Server-Caching konfigurieren", auf Seite 75.

Teil 5. Sicherheit für Caching Proxy konfigurieren

Dieser Teil enthält Informationen zur Basissicherheit, zur Verwendung von SSL in Caching Proxy, zum Aktivieren von Verschlüsselungshardware sowie zur Verwendung des Plug-in IBM Tivoli Access Manager (früher Tivoli Policy Director) und des Autorisierungsmoduls PAC-LDAP.

Dieser Teil enthält die folgenden Kapitel:

Kapitel 24. Informationen zur Sicherheit von Proxy-Servern

Jeder über das Internet zugängliche Server ist dem Risiko ausgesetzt, dass dem System, auf dem er ausgeführt wird, unerwünschtes Interesse geschenkt wird. Nicht autorisierte Personen könnten versuchen, Kennwörter zu erraten, Dateien zu aktualisieren oder auszuführen oder vertrauliche Daten zu lesen. Teil der Anziehungskraft des World Wide Web ist seine Zugänglichkeit für jedermann. Diese Zugänglichkeit ist jedoch nicht nur von Vorteil, sondern fördert auch den Missbrauch.

In den folgenden Abschnitten wird beschrieben, wie Sie den Zugriff auf Dateien kontrollieren können, die sich auf dem Caching-Proxy-Server befinden.

Caching Proxy unterstützt SSL-Verbindungen (Secure Sockets Layer), in denen Daten zwischen dem Client-Browser und dem Zielserver (ein Inhaltsserver oder ein Ersatz-Proxy-Server) durch Ver- und Entschlüsselung sicher übertragen werden.

Ist Caching Proxy als Ersatzserver definiert, kann er sichere Verbindungen zu Clients und/oder zu Inhaltsservern aufbauen. Wenn Sie SSL-Verbindungen mit den Konfigurations- und Verwaltungsformularen aktivieren möchten, wählen Sie **Proxy-Konfiguration –> SSL-Einstellungen** aus. Wählen Sie in diesem Formular das Markierungsfeld **SSL aktivieren** aus und geben Sie eine Schlüsselringdatenbank und eine Kennwortdatei für die Schlüsselringdatenbank an.

Sie können mehrere grundlegende Vorsichtsmaßnahmen zum Schutz Ihres Systems treffen:

- Verwenden Sie einen für den öffentlichen Zugriff bestimmten Server in einem Netz, das von Ihrem lokalen oder internen Netz getrennt ist.
- Deaktivieren Sie Dienstprogramme, mit denen ferne Benutzer auf die internen Verarbeitungsprozesse des Servers zugreifen können. Darunter fallen insbesondere **telnet**-, **TN3270**-, **rlogin**- und **finger**-Clients auf dem System, auf dem der Server aktiv ist.
- Verwenden Sie Paketfilter und Firewalls.

  Mit Paketfiltern können Sie die Quelle und den Zielort von Daten definieren. Sie können Ihr System so konfigurieren, dass bestimmte Kombinationen von Quelle und Ziel zurückgewiesen werden.

  Eine Firewall trennt das interne Netz von einem öffentlich zugänglichen Netz, wie z. B. dem Internet. Die Firewall kann eine Gruppe von Computern oder ein einzelner Computer sein, der als Gateway in beide Richtungen agiert und den Datenverkehr reguliert und verfolgt. Als Firewall-Software können Sie z. B. IBM Firewall einsetzen.
- Kontrollieren Sie die Verwendung von CGI-Scripts. Die Verwendung von CGI-Scripts auf einem Webserver kann ein Sicherheitsrisiko darstellen, da diese Scripts Umgebungsvariablen anzeigen können, die sensible Daten wie beispielsweise Benutzer-IDs und Kennwörter enthalten. Machen Sie sich mit den Funktionen des CGI-Programms genau vertraut, bevor Sie es auf Ihrem Server ausführen, und kontrollieren Sie, wer auf die CGI-Scripts auf Ihrem Server zugreifen kann.

Anmerkung: Wenn Sie den Proxy-Server mit dem Konfigurationsassistenten konfigurieren, müssen Sie zum Aktivieren von SSL eine Zuordnungsregel erstellen, damit die an Port 443 empfangenen Anforderungen weitergeleitet werden. Nähere Informationen hierzu finden Sie im Abschnitt „Zuordnungsregeln definieren" auf Seite 45.

Beispiele:

```
Proxy /* http://content server :443
```

oder

```
Proxy /* https://content server :443
```

Kapitel 25. Zugriffsschutzkonfigurationen für den Server

Dieses Kapitel beschreibt, wie Sie die Daten und Dateien auf Ihrem Server mit Zugriffsschutzkonfigurationen schützen können. Zugriffsschutzkonfigurationen werden auf der Grundlage der Anforderung ausgelöst, die der Server empfängt. Dabei spielen insbesondere das angeforderte Verzeichnis, die Datei oder der Dateityp, die in der Anforderung angegeben sind, eine Rolle. In einer Zugriffsschutzkonfiguration steuern untergeordnete Anweisungen die Erteilung bzw. Verweigerung von Zugriffsrechten auf der Grundlage der Merkmale der geschützten Verzeichnisse oder Dateien.

Den Zugriffsschutz mit den Konfigurations- und Verwaltungsformularen festlegen

Um eine Zugriffsschutzkonfiguration zu definieren und ihre Anwendung festzulegen, wählen Sie in den Konfigurations- und Verwaltungsformularen **Serverkonfiguration –> Zugriffsschutz für Dokumente** aus. Legen Sie in diesem Formular Folgendes fest:

1. Legen Sie die Position dieser Zugriffsschutzregel fest.

   Regeln für den Zugriffsschutz werden in der Reihenfolge angewendet, in der sie in der Tabelle im Konfigurationsformular aufgeführt sind. Im Allgemeinen sind in der Tabelle zuerst die speziellen und dann die generischen Regeln aufgelistet.

   Legen Sie mit dem Dropdown-Menü und den Schaltflächen im Formular die Position einer Zugriffsschutzregel fest.

2. Definieren Sie eine Anforderungsschablone.

   Der Zugriffsschutz wird auf der Grundlage der Anforderungsschablonen aktiviert, die mit dem Inhalt der Anforderungen verglichen werden, die die Clients an Ihren Proxy-Server senden.

   Eine *Anforderung* ist der Teil eines vollständigen URL, der dem Hostnamen des Servers folgt. Wenn Ihr Server beispielsweise den Namen fine.feathers.com hat und ein Browser den URL `http://fine.feathers.com/waterfowl/schedule.html` sendet, empfängt der Server die Anforderung /waterfowl/schedule.html. Anforderungsschablonen spezifizieren Verzeichnis- und/oder Dateinamen, die dem Zugriffsschutz unterliegen. Beispiele für Anforderungen, die den Zugriffsschutz auf der Basis der gerade beschriebenen Schablone (/waterfowl/schedule.html) aktivieren, sind /waterfowl/* und /*schedule.html.

   Geben Sie im Feld **URL-Anforderungsschablone** die Anforderungsschablone ein.

3. Definieren Sie eine Zugriffsschutzkonfiguration.

   Eine Zugriffsschutzkonfiguration weist den Caching-Proxy-Server an, wie er vorgehen soll, wenn eine Anforderung mit einer Anforderungsschablone übereinstimmt. Sie können eine benannte Zugriffsschutzkonfiguration verwenden oder im Formular **Zugriffsschutz für Dokumente** eine neue Konfiguration definieren.

   Wenn Sie eine benannte Konfiguration verwenden möchten, klicken Sie auf den Radioknopf **Benannter Zugriffsschutz** und geben Sie den Namen im dafür vor-

gesehenen Feld ein. Wenn Sie eine neue Zugriffsschutzkonfiguration definieren möchten, klicken Sie auf den Radioknopf **Inline** und befolgen Sie die angezeigten Anweisungen (siehe Schritt 6).

4. Wählen Sie eine Requester-Adresse aus (optional).

   Auf Anforderungen von unterschiedlichen Serveradressen können unterschiedliche Regeln angewendet werden. Beispielsweise könnten Sie für Anforderungen von Protokolldateien eine andere Zugriffsschutzkonfiguration anwenden, wenn diese Anforderungen von IP-Adressen Ihres Unternehmens stammen.

   Anmerkung: Zur Filterung von Requester-Adressen müssen Sie die DNS-Suchfunktion (DNS Lookup) aktivieren. Nähere Informationen hierzu finden Sie im Abschnitt „DNS-Lookup - Angeben, ob der Server die Hostnamen von Clients suchen soll" auf Seite 212.

   Wenn die Requester-Adresse in der Regel enthalten sein soll, geben Sie sie im Feld **IP-Adresse oder Hostname des Servers** ein.

5. Klicken Sie auf **Übergeben**.

   Wenn Sie eine benannte Zugriffsschutzkonfiguration verwenden, sind keine weiteren Eingaben erforderlich. Haben Sie eine Inline-Zugriffsschutzkonfiguration ausgewählt oder eine benannte Konfiguration angegeben, die nicht existiert, werden vom System weitere Formulare angezeigt.

6. Legen Sie die Einzelheiten des Zugriffsschutzes fest.

   Wenn Sie keine vorhandene Zugriffsschutzkonfiguration angegeben haben, wird ein weiteres Formular geöffnet, in dem Sie festlegen können, welche Benutzer auf die Dokumente oder Verzeichnisse zugreifen dürfen, die mit der Anforderungsschablone übereinstimmen, und für welche Aktionen die Benutzer berechtigt sind.

   - **Einstellungen der Kennwortauthentifizierung** - Geben Sie die Kennwort- und/oder Gruppendatei für die Benutzerauthentifizierung an. Geben Sie außerdem den Namen an, mit dem der Server identifiziert wird, wenn dieser Namen und Kennwort eines Requester anfordert.

     Anmerkung: Einige Browser speichern Benutzer-IDs und Kennwörter im Cache und ordnen sie einer Server-ID zu. Ihre Benutzer finden es möglicherweise angenehmer, wenn Sie immer dieselbe Server-ID mit derselben Kennwortdatei verwenden.

   - **Berechtigungen** - Geben Sie die Benutzer oder Gruppen an, die zum Lesen, Schreiben oder Löschen von geschützten Dateien berechtigt sind.

7. Klicken Sie auf **Übergeben**.
8. Starten Sie den Server erneut.

Den Zugriffsschutz mit den Anweisungen in der Konfigurationsdatei festlegen

Wenn Sie den Zugriffsschutz durch Editieren der Konfigurationsdatei von Caching Proxy definieren möchten, muss Ihnen Folgendes bekannt sein:

- Der Unterschied zwischen den Anweisungen Protect, defProt und Protection:
  - Die Protect-Anweisung definiert den Zugriffsschutz, indem sie eine Anforderungsschablone mit einer Zugriffsschutzkonfiguration verbindet. Nähere Informationen finden Sie im Abschnitt „Protect - Eine Zugriffsschutzkonfiguration für Anforderungen aktivieren, die mit einer Schablone übereinstimmen" auf Seite 253.

- Mit der defProt-Anweisung wird eine Standardzugriffsschutzkonfiguration für eine bestimmte Anforderungsschablone festgelegt. Nähere Informationen finden Sie im Abschnitt „DefProt - Standardzugriffsschutzkonfiguration für Anforderungen angeben, die mit einer Schablone übereinstimmen" auf Seite 204.
- Mit der Protection-Anweisung wird eine benannte Zugriffsschutzkonfiguration definiert. Nähere Informationen finden Sie im Abschnitt „Protection - In der Konfigurationsdatei eine benannte Zugriffsschutzkonfiguration definieren" auf Seite 258.

- Die Art und Weise, wie Zugriffsschutz und Anforderungs-Routing zusammenarbeiten:

  Mit Anweisungen für Anforderungs-Routing wie Map, Exec, Pass und Proxy steuern Sie, welche Anforderungen Ihr Server akzeptiert und wie er Anforderungen an die Dateiadressen weiterleitet. Die Anweisungen für Anforderungs-Routing verwenden dieselben Anforderungsschablonen wie Zugriffsschutzanweisungen. Da die Anweisungen ausgeführt werden, die mit der ersten übereinstimmenden Schablone für jede Anforderung verknüpft sind, müssen die Zugriffsschutzanweisungen vor den Routing-Anweisungen in der Konfigurationsdatei stehen, damit der Zugriffsschutz ordnungsgemäß funktioniert. Nähere Informationen hierzu finden Sie im Abschnitt „Protect - Eine Zugriffsschutzkonfiguration für Anforderungen aktivieren, die mit einer Schablone übereinstimmen" auf Seite 253.
- Der Unterschied zwischen Inline- und benannten Zugriffsschutzkonfigurationen:

  Mit der Protect-Anweisung können Sie eine Inline-Zugriffsschutzkonfiguration angeben oder auf eine vorhandene benannte Konfiguration verweisen. Die Syntax der beiden Angaben unterscheidet sich nur minimal. Nähere Informationen finden Sie im Abschnitt „Protect - Eine Zugriffsschutzkonfiguration für Anforderungen aktivieren, die mit einer Schablone übereinstimmen" auf Seite 253.
- Das Erstellen einer Zugriffsschutzkonfiguration:

  Eine Zugriffsschutzkonfiguration besteht aus einer Folge von Angaben, die die untergeordneten Anweisungen für Zugriffsschutz enthalten. Syntax und Referenzinformationen zum Schreiben von Zugriffsschutzkonfigurationen finden Sie in Anhang B, „Anweisungen in der Konfigurationsdatei", auf Seite 175 in den folgenden Abschnitten:
  - „Protect - Eine Zugriffsschutzkonfiguration für Anforderungen aktivieren, die mit einer Schablone übereinstimmen" auf Seite 253
  - „Protection - In der Konfigurationsdatei eine benannte Zugriffsschutzkonfiguration definieren" auf Seite 258
  - „Untergeordnete Anweisungen für den Zugriffsschutz - Angeben, wie eine Gruppe von Ressourcen geschützt wird" auf Seite 259

Standardeinstellungen für Zugriffsschutz

Die Standardkonfigurationsdatei des Proxy-Servers enthält eine Zugriffsschutzkonfiguration, die für den Zugriff auf die Dateien im Verzeichnis /admin-bin/ die Angabe einer Administrator-ID und eines Kennworts erfordert. Diese Einstellung schränkt den Zugriff auf die Konfigurations- und Verwaltungsformulare ein.

Kapitel 26. Secure Sockets Layer (SSL)

SSL (Secure Sockets Layer) ist ein System, mit dem Informationen, die über das Internet gesendet werden, automatisch verschlüsselt und beim Empfänger vor ihrer Verwendung wieder entschlüsselt werden. Auf diese Weise werden sensible Informationen wie Kreditkartennummern während der Übertragung über das Internet geschützt.

Caching Proxy verwendet SSL für den Schutz von Ersatzservern und für eine sichere Fernverwaltung. Nähere Informationen hierzu finden Sie in den folgenden Abschnitten. Mit SSL können auch Verbindungen zu Back-End-Servern (z. B. Inhalts- oder Anwendungsservern) und Datenübertragungen zwischen dem Proxy-Server und seinen Clients geschützt werden.

Der SSL-Handshake

Der SSL-Zugriffsschutz wird aktiviert, wenn eine Anforderung für eine sichere Verbindung von einer Maschine an eine andere gesendet wird, z. B. wenn ein Browser eine Anforderung an einen Ersatz-Proxy-Server sendet. Die Anforderungssyntax https:// anstelle von http:// teilt dem Browser mit, dass die Anforderung an Port 443 gesendet werden soll. Dies ist der Port, an dem der Server geschützte Verbindungsanforderungen empfängt (im Unterschied zu Port 80, an dem Routineanforderungen empfangen werden). Zum Aufbau einer sicheren Sitzung zwischen Browser und Server führen die beiden Maschinen einen Informationsaustausch aus, den so genannten *SSL-Handshake*. Mit diesem Austausch legen die beiden Parteien eine Verschlüsselungsspezifikation fest und wählen den Schlüssel aus, mit dem die Informationen ver- und entschlüsselt werden sollen. Die Schlüssel werden automatisch generiert und verfallen mit Beendigung der Sitzung. Im Folgenden wird ein typisches Szenario (mit SSL Version 3) beschrieben:

1. Hello-Nachricht des Clients

   Der Client leitet eine SSL-Sitzung mit Caching Proxy ein, indem er eine Hello-Nachricht sendet, die die Verschlüsselungsfunktionen des Clients beschreibt.

2. Hello-Nachricht des Servers

   Der Server sendet sein Zertifikat an den Client und wählt die Cipher Suite aus, die zur Datenverschlüsselung verwendet werden soll.

3. Abschlussnachricht des Clients

   Der Client sendet die Informationen zum Chiffrierschlüssel, mit dem die symmetrischen Chiffrierschlüssel für die verschlüsselten Daten erstellt werden. Diese Schlüsselinformationen werden als *Premaster Secret* bezeichnet und sind mit dem öffentlichen Schlüssel des Servers (aus dem Serverzertifikat, siehe „Schlüssel- und Zertifikatverwaltung" auf Seite 120) verschlüsselt. Server und Client können die symmetrischen Chiffrierschlüssel zum Lesen und Schreiben aus dem Premaster Secret abrufen.

4. Abschlussnachricht des Servers

   Der Server sendet die Abschlussbestätigung und einen Nachrichtenauthentifizierungscode (Message Authentication Code, MAC) für das gesamte Handshake-Protokoll.

5. Clientüberprüfung

   Der Client sendet eine Nachricht, um die Abschlussnachricht des Servers zu überprüfen.

6. Sicherer Datenfluss

   Nachdem der Client die Server-Finish-Nachricht überprüft hat, kann der verschlüsselte Datenfluss beginnen.

Mit einem Caching-Proxy-Server als Endpunkt für sichere Verbindungen können Sie die Arbeitslast auf Ihren Inhaltsservern und Anwendungsservern verringern. Wenn ein Caching-Proxy-Server eine sichere Verbindung verwaltet, ist er für die Verschlüsselung, Entschlüsselung und die Generierung der Schlüssel zuständig. Bei allen Vorgängen handelt es sich um CPU-intensive Operationen. Außerdem können Sie in Caching Proxy SSL-Sitzungszeitlimits konfigurieren, um eine möglichst lange Verwendung jedes Schlüssels zu gewährleisten.

Einschränkungen von SSL

In WebSphere Application Server Caching Proxy gelten folgende SSL-Einschränkungen:

- Caching Proxy selbst kann nicht als Zertifizierungsstelle verwendet werden (siehe „Schlüssel- und Zertifikatverwaltung").
- Einige Browser unterstützen eventuell nicht die vollständige Verschlüsselungstechnologie, die in Caching Proxy verwendet wird.

Sichere Fernverwaltung konfigurieren

Die Fernverwaltung von Caching Proxy kann mit den SSL-Sicherheitsfunktionen (Secure Sockets Layer) und der Kennwortauthentifizierung durchgeführt werden. Dadurch verringert sich das Risiko, dass nicht autorisierte Personen auf den Proxy-Server zugreifen.

Wenn Sie für die Fernverwaltung Ihres Servers SSL einsetzen möchten, verwenden Sie zum Öffnen der Konfigurations- und Verwaltungsformulare eine https://-Anforderung anstelle einer http://-Anforderung. Beispiel:

`https://Name.Ihres.Servers/IhreFrontPage.html`

Schlüssel- und Zertifikatverwaltung

Wie bereits erwähnt, müssen Sie vor der Konfiguration von SSL eine Schlüsseldatenbank einrichten und ein Zertifikat erstellen oder abrufen. Zertifikate dienen der Authentifizierung von Servern. Verwenden Sie zum Definieren Ihrer Zertifizierungsdateien das Dienstprogramm IBM Key Management (auch iKeyman genannt). Dieses Dienstprogramm ist Teil der GSKit-Software, die im Produktumfang von Application Server enthalten ist. Außerdem enthält GSKit eine Java-basierte Grafikschnittstelle zum Öffnen von Zertifikatdateien.

Nachfolgend sind die grundlegenden Schritte zum Definieren der SSL-Schlüssel und Zertifikate beschrieben.

1. Stellen Sie sicher, dass GSKit installiert ist. Auf den meisten Plattformen wird das Programm automatisch mit der Komponente Caching Proxy installiert. Der Name des Pakets ist gsk7ikm (und gsk7ikm_gcc295 auf Linux-Systemen für i386). GSKit wird normalerweise im Verzeichnis ibm/gsk7/ (auf AIX-Systemen im Verzeichnis ibm/gskit/) installiert. Auf Windows-Plattformen kann das Programm auch über das Menü **Start** aufgerufen werden.

   Anmerkung: Falls das GSKit unter Windows mit InstallShield nicht installiert wird, stellen Sie sicher, dass der Pfad zum Installationsverzeichnis keine Leerzeichen enthält.

2. Verwenden Sie IBM Key Management, um einen Schlüssel für sichere Netzübertragungen zu erstellen und das Zertifikat einer Zertifizierungsstelle zu empfangen. Während Sie auf dieses Zertifikat warten, können Sie ein selbstsigniertes Zertifikat erstellen.
3. Erstellen Sie eine Schlüsseldatenbank und legen Sie ein Kennwort für diese Datenbank fest.

Anmerkung: Die Schlüssel- und Schlüsselspeicherdateien werden deinstalliert, wenn die Komponente Caching Proxy von Application Server deinstalliert wird. Damit Sie nicht ein neues Zertifikat bei einer Zertifizierungsstelle anfordern müssen, sollten Sie Sicherungskopien dieser beiden Dateien in einem anderen Verzeichnis speichern, bevor Sie die Proxy-Software deinstallieren.

Zertifizierungsstellen

Ihrem öffentlichen Schlüssel muss ein digital signiertes Zertifikat einer Zertifizierungsstelle (CA) zugeordnet sein, die als vertrauenswürdige Stamm-CA auf Ihrem Server festgelegt ist. Sie können ein signiertes Zertifikat erwerben, indem Sie eine Zertifikatanforderung an den Provider der Zertifizierungsstelle (CA) senden. Caching Proxy unterstützt die folgenden externen Zertifizierungsstellen:

- VeriSign
- Thawte

Standardmäßig sind die folgenden Zertifizierungsstellen als Trusted-CAs definiert:

- Verisign Class 1 Individual Subscriber CA - Persona Not Validated
- Verisign Class 2 Individual Subscriber CA - Persona Not Validated
- Verisign Class 3 Individual Subscriber CA - Persona Not Validated
- VeriSign Class 3 International Server CA
- VeriSign Class 2 OnSite Individual CA
- VeriSign Class 1 Public Primary CA
- VeriSign Class 2 Public Primary CA
- VeriSign Class 3 Public Primary CA
- VeriSign Class 1 Public Primary CA - G2
- VeriSign Class 2 Public Primary CA - G2
- RSA Secure Server CA (from VeriSign)
- Thawte Personal Basic CA
- Thawte Personal Freemail CA
- Thawte Personal Premium CA
- Thawte Premium Server CA
- Thawte Server CA

Das Dienstprogramm IBM Key Management verwenden

Dieser Abschnitt enthält eine Kurzübersicht über die Verwendung des Dienstprogramms IBM Key Management (iKeyman). Mit diesem Dienstprogramm erstellen Sie Ihre SSL-Schlüsseldatenbankdatei, öffentlich-private Schlüsselpaare und Zertifikatanforderungen. Nachdem Sie das von der Zertifizierungsstelle signierte Zertifikat empfangen haben, legen Sie es mit IBM Key Management in der Schlüsseldatenbank ab, in der Sie die ursprüngliche Zertifikatanforderung erstellt haben.

Weitere ausführliche Informationen finden Sie in der Dokumentation zu IBM Key Management und GSKit, die der GSKit-Software beiliegt.

Das System für die Ausführung von IBM Key Management konfigurieren

Gehen Sie vor dem Start der GUI von IKeyman wie folgt vor:

1. Installieren Sie IBM Java 2 Technology Version 1.4.2 (32-Bit) oder eine kompatible Java-Version.
2. Setzen Sie die Variable JAVA_HOME auf das Java-Verzeichnis. Beispiel:
   - Windows: `set JAVA_HOME=C:\Programme\IBM\Java142`
   - Linux und UNIX: `export JAVA_HOME=/usr/opt/IBMJava2-142`
3. Entfernen Sie die Dateien ibmjsse.jar und gskikm.jar (falls vorhanden) sowie die Datei ibmjcaprovider.jar aus dem Verzeichnis JAVA_HOME/jre/lib/ext.

   Anmerkung: Für Sun müssen Sie das Verzeichnis JAVA_HOME/jre/lib/ext durch JAVA_HOME/lib/ext/ ersetzen.
4. Die folgenden JAR-Dateien sind derzeit im Verzeichnis *GSKit-Installationspfad*/classes/jre/lib/ext/ gespeichert.
   - Kopieren Sie die angegebenen JAR-Dateien nach JAVA_HOME/jre/lib/:

     ```
     ibmjcefw.jar
     ibmpkcs11.jar
     ```
   - Kopieren Sie die angegebenen JAR-Dateien nach JAVA_HOME/jre/lib/ext:

     ```
     ibmjceprovider.jar
     ibmpkcs.jar
     ```
   - Kopieren Sie die angegebenen JAR-Dateien nach JAVA_HOME/jre/lib/security:

     ```
     local_policy.jar
     US_export_policy.jar
     ```
5. Registrieren Sie die Serviceprovider IBM JCE, IBM CMS und/oder IBMJCEFIPS:

   Aktualisieren Sie die Datei JAVA_HOME/jre/lib/security/java.security, indem Sie hinter dem Sun-Provider die Provider IBM CMS und IBM JCE hinzufügen. Beispiel:

   ```
   security.provider.1=sun.security.provider.Sun
   security.provider.2=com.ibm.spi.IBMCMSProvider
   security.provider.3=com.ibm.crypto.provider.IBMJCE
   ```

   Eine Beispieldatei java.security ist im folgenden Verzeichnis enthalten: *GSKit-Installationspfad*/classes/gsk_java.security
   - Zur Unterstützung von FIPS müssen Sie auch die Datei JAVA_HOME/jre/lib/security/java.security editieren und hinter dem Sun-Provider IBMJCEFIPS hinzufügen. Vergewissern Sie sich, dass der Provider IBMJCEFIPS mit einer höheren Priorität registriert wird als IBMJCE. Beispiel:

     ```
     security.provider.1=sun.security.provider.Sun
     security.provider.2=com.ibm.spi.IBMCMSProvider
     security.provider.3=com.ibm.crypto.fips.provider.IBMJCEFIPS
     security.provider.4=com.ibm.crypto.provider.IBMJCE
     ```
6. (Optional) Wenn Sie mit JSSE arbeiten und JSSE für den Zugriff auf die Verschlüsselungshardware verwenden, installieren Sie die Datei ibmpkcs11.jar im Verzeichnis JAVA_HOME/jre/lib und folgen Sie den Anweisungen in der Datei *GSKit-Installationspfad*/classes/native/native-support.zip, um die gemeinsam genutzten Bibliotheken für die Verschlüsselungshardware zu konfigurieren.

Anmerkung: Die Datei ibmpkcs11.jar ist auch im JSSE-Paket vom 5. August 2002 enthalten. Zum Registrieren von IBMPKCS11 muss die Datei JAVA_HOME/jre/lib/security/java.security wie folgt aktualisiert werden:

```
security.provider.1=sun.security.provider.Sun
security.provider.2=com.ibm.crypto.provider.IBMJCE
security.provider.3=com.ibm.crypto.pkcs11.provider.IBMPKCS11
```

IBM Key Management starten

Starten Sie wie folgt die grafische Benutzerschnittstelle von IBM Key Management:

- Geben Sie auf Linux- und UNIX-Plattformen an einer Eingabeaufforderung `gsk7ikm` ein.
- Klicken Sie unter Windows auf **Start –> Programme –> IBM WebSphere –> Edge Components –> Caching Proxy –> Dienstprogramm Key Management starten.**

Wenn Sie während dieser Sitzung eine neue Schlüsseldatenbankdatei erstellen, wird diese Datei in dem Verzeichnis gespeichert, in dem Sie IBM Key Management gestartet haben.

Eine neue Schlüsseldatenbank, ein neues Kennwort und eine neue Datei zur Kennwortspeicherung erstellen

Bei einer Schlüsseldatenbank handelt es sich um eine Datei, in der der Server ein oder mehrere Schlüsselpaare und Zertifikate speichert. Für sämtliche Schlüsselpaare und Zertifikate können Sie entweder eine Schlüsseldatenbank verwenden oder mehrere Datenbanken erstellen. Mit dem Dienstprogramm IBM Key Management können Sie neue Schlüsseldatenbanken erstellen und die zugehörigen Kennwörter und Dateien zur Kennwortspeicherung festlegen.

So erstellen Sie eine Schlüsseldatenbank und eine Datei zur Kennwortspeicherung:

1. Starten Sie IBM Key Management.
2. Wählen Sie im Hauptmenü **Schlüsseldatenbankdatei –> Neu** aus.
3. Im Dialogfenster **Neu** muss der Dateityp **CMS-Schlüsseldatenbank** ausgewählt sein. Geben Sie den Namen Ihrer Schlüsseldatenbank und die Dateiadresse ein oder bestätigen Sie die Standardeinstellung **key.kdb**. Klicken Sie auf **OK**.
4. Geben Sie im Dialogfenster **Aufforderung zur Kennworteingabe** das Kennwort für diese Datenbank ein und bestätigen Sie es. Klicken Sie auf **OK**.
5. Wählen Sie das Markierungsfeld zum Speichern der Kennwortdatei aus. Geben Sie bei entsprechender Aufforderung ein Kennwort ein und bestätigen Sie es. Die folgende Nachricht wird angezeigt: `DB-Typ: CMS-Schlüsseldatenbankdatei` *`Name_der_Schlüsseldatenbank`*

   Anmerkung: Wenn Sie die Kennwortdatei nicht speichern, wird der Server zwar gestartet, aber er nimmt keine Anforderungen an Port 443 entgegen.

Das Kennwort, das Sie beim Erstellen einer neuen Schlüsseldatenbank angeben, schützt den privaten Schlüssel. Der private Schlüssel ist der einzige Schlüssel, der Dokumente signieren oder Nachrichten entschlüsseln kann, die mit dem öffentlichen Schlüssel verschlüsselt wurden.

Beachten Sie bei der Festlegung des Kennworts folgende Richtlinien:

- Das Kennwort muss aus Zeichen bestehen, die im englischen Zeichensatz enthalten sind.
- Das Kennwort muss mindestens sechs Zeichen lang sein und mindestens zwei nicht aufeinanderfolgende Zahlen enthalten. Stellen Sie sicher, dass das Kennwort keine öffentlich zugänglichen persönlichen Daten enthält, z. B. Ihren Namen oder den Namen eines Familienangehörigen, Initialen oder Geburtsdaten.
- Speichern Sie das Kennwort versteckt.

Es wird empfohlen, das Kennwort der Schlüsseldatenbank regelmäßig zu ändern. Wenn Sie ein Verfallsdatum für das Kennwort festlegen, sollten Sie sich auf jeden Fall notieren, wann Sie das Kennwort ändern müssen. Sollten Sie das Verfallsdatum des Kennworts verpassen, wird eine Nachricht in das Fehlerprotokoll geschrieben. Der Server wird dann zwar gestartet, kann aber keine sicheren Netzverbindungen herstellen.

So ändern Sie das Kennwort für die Schlüsseldatenbank:

1. Wählen Sie im Hauptmenü die Optionen **Schlüsseldatenbankdatei –> Öffnen** aus.
2. Geben Sie im Dialogfenster **Öffnen** den Namen Ihrer Schlüsseldatenbank ein oder übernehmen Sie die Standardeinstellung **key.kdb**. Klicken Sie auf **OK**.
3. Geben Sie im Dialogfenster **Aufforderung zur Kennworteingabe** Ihr definiertes Kennwort ein und klicken Sie auf **OK**.
4. Wählen Sie im Hauptmenü die Optionen **Schlüsseldatenbankdatei –> Kennwort ändern** aus.
5. Geben Sie im Dialogfenster **Kennwort ändern** das Kennwort für diese Datenbank ein und bestätigen Sie es. Klicken Sie auf **OK**.

Wenn Sie zwischen einem Proxy-Server und einem LDAP-Server eine SSL-Verbindung herstellen möchten, müssen Sie das Kennwort für die Schlüsseldatenbank in der Datei pac_keyring.pwd speichern. (Die Datei pac_keyring.pwd ist nicht die verdeckte Datei, die von iKeyman generiert wird.)

Ein neues Schlüsselpaar und eine neue Zertifikatanforderung erstellen

In der Schlüsseldatenbank sind Schlüsselpaare und Zertifikatanforderungen gespeichert. So erstellen Sie ein öffentlich-privates Schlüsselpaar und eine Zertifikatanforderung:

1. Wenn Sie die Schlüsseldatenbank noch nicht erstellt haben, befolgen Sie die Anweisungen im Abschnitt „Eine neue Schlüsseldatenbank, ein neues Kennwort und eine neue Datei zur Kennwortspeicherung erstellen" auf Seite 123.
2. Klicken Sie im Hauptmenü von IBM Key Management auf **Schlüsseldatenbank –> Datei –> Öffnen**.
3. Geben Sie im Dialogfenster **Öffnen** den Namen Ihrer Schlüsseldatenbank ein (oder klicken Sie auf **key.kdb**, falls Sie die Standardeinstellung verwenden möchten). Klicken Sie auf **OK**.
4. Geben Sie im Dialogfenster **Aufforderung zur Kennworteingabe** Ihr Kennwort ein und klicken Sie auf **OK**.
5. Wählen Sie im Hauptmenü die Optionen **Erstellen –> Neue Zertifikatanfrage** aus.

6. Geben Sie im Dialogfenster **Neue Schlüssel- und Zertifikatanfrage** Folgendes an:
   - **Schlüsselname:** Geben Sie einen Namen ein, der den Schlüssel und das Zertifikat in der Datenbank kennzeichnet, z. B. `mein selbstsigniertes Zertifikat` oder `www.UnternehmenA.com`.
   - **Schlüsselgröße:** Größe des Schlüssels, z. B. `1024`. (Zur Verwendung der 128-Bit-Verschlüsselung wird die Schlüsselgröße 1024 empfohlen.)
   - **Organisationsname:** Name der Organisation, die dem Schlüssel zugeordnet werden soll, z. B. `UnternehmenA`.
   - **Organisationseinheit** (optional)
   - **Standort** (optional)
   - **Bundesland** (optional)
   - **Postzustellbezirk** (optional)
   - **Land:** Ihr Landescode. Sie müssen mindestens zwei Zeichen angeben, z. B. `DE`.
   - **Name der Zertifikatanforderungsdatei:** Der Name der Anforderungsdatei. Sie können wahlweise einen Standardnamen verwenden.
7. Klicken Sie auf **OK**. Eine Bestätigungsnachricht wie die Folgende wird angezeigt:

   ```
   Eine neue Zertifikatanfrage wurde erfolgreich
   in der Datei Name_der_Schlüsseldatenbankname erstellt.
   ```
8. Klicken Sie auf **OK**. Der von Ihnen eingegebene Name wird daraufhin unter der Überschrift **Persönliche Zertifikatanfrage** angezeigt.
9. Klicken Sie im Dialogfenster **Information** auf **OK**. Sie werden daran erinnert, die Datei an eine Zertifizierungsstelle zu senden.
10. Wenn Sie kein selbstsigniertes Zertifikat erstellt haben (ausführliche Informationen dazu finden Sie im nächsten Abschnitt, "Selbstsigniertes Zertifikat erstellen"), senden Sie die Zertifikatanforderung an die Zertifizierungsstelle:
    - IBM Key Management muss aktiv sein.
    - Starten Sie einen Webbrowser und geben Sie die URL-Adresse der Zertifizierungsstelle ein, von der Sie das Zertifikat anfordern möchten.
    - Folgen Sie den Anweisungen der Zertifizierungsstelle, um Ihre Zertifikatanforderung zu senden.

    Die Bearbeitung von Zertifikatanforderungen kann unter Umständen zwei bis drei Wochen dauern. Während Sie auf die Bearbeitung Ihrer Zertifikatanforderung durch die Zertifizierungsstelle warten, können Sie als Ihre eigene Zertifizierungsstelle auftreten und mit iKeyman ein selbstsigniertes Serverzertifikat erstellen, damit Sie SSL-Sitzungen zwischen Clients und dem Caching-Proxy-Server aufbauen können.

Selbstsigniertes Zertifikat erstellen

Erstellen Sie mit IBM Key Management ein selbstsigniertes Serverzertifikat, um damit SSL-Sitzungen zwischen Clients und dem Proxy-Server zu aktivieren, während Sie auf die Ausstellung des angeforderten Zertifikats warten. Außerdem können Sie selbstsignierte Zertifikate zu Testzwecken verwenden.

Gehen Sie wie folgt vor, um ein selbstsigniertes Zertifikat zu erstellen:

1. Wenn Sie die Schlüsseldatenbank noch nicht erstellt haben, befolgen Sie die Anweisungen im Abschnitt „Eine neue Schlüsseldatenbank, ein neues Kennwort und eine neue Datei zur Kennwortspeicherung erstellen" auf Seite 123.

2. Klicken Sie im Hauptmenü von IBM Key Management auf **Schlüsseldatenbank –> Datei –> Öffnen**.
3. Geben Sie im Dialogfenster **Öffnen** den Namen Ihrer Schlüsseldatenbank ein (oder übernehmen Sie die Standardeinstellung **key.kdb**). Klicken Sie auf **OK**.
4. Geben Sie im Dialogfenster **Aufforderung zur Kennworteingabe** Ihr Kennwort ein und klicken Sie auf **OK**.
5. Wählen Sie im Inhaltsrahmen **Schlüsseldatenbank** die Option **Persönliche Zertifikate** aus und klicken Sie auf **Neues selbstsigniertes Zertifikat erstellen**.
6. Legen Sie im Fenster **Neues selbstsigniertes Zertifikat erstellen** Folgendes fest:
   - **Schlüsselname:** Geben Sie einen Namen ein, mit dem der Schlüssel und das Zertifikat in der Datenbank identifiziert werden, z. B. `mein selbstsigniertes Zertifikat`.
   - **Schlüsselgröße:** Größe des Schlüssels, z. B. 512.
   - **Allgemeiner Name:** Der vollständige Hostname des Servers, z. B. `www.meinserver.com`
   - **Organisationsname:** Name der Organisation, die dem Schlüssel zugeordnet werden soll, z. B. `Unternehmen A`
   - **Organisationseinheit** (optional)
   - **Standort** (optional)
   - **Bundesland** (optional)
   - **Postzustellbezirk** (optional)
   - **Land:** Ihr Landescode. Sie müssen mindestens zwei Zeichen angeben, z. B. DE.
   - **Gültigkeitszeitraum:** Die Dauer, für die das Zertifikat gültig ist.
7. Klicken Sie auf **OK**.
8. Registrieren Sie die Schlüsseldatenbank bei dem Server, indem Sie die Schlüsseldatei und die Datei zur Kennwortspeicherung (Stash-Datei) zu den Konfigurationseinstellungen hinzufügen (siehe Abschnitt „Eine neue Schlüsseldatenbank, ein neues Kennwort und eine neue Datei zur Kennwortspeicherung erstellen" auf Seite 123).

Schlüssel exportieren

So exportieren Sie Schlüssel in eine andere Schlüsseldatenbank:

1. Starten Sie IBM Key Management.
2. Wählen Sie im Hauptmenü die Optionen **Schlüsseldatenbankdatei –> Öffnen** aus.
3. Geben Sie im Dialogfenster **Öffnen** den Namen Ihrer Schlüsseldatenbank ein (oder übernehmen Sie die Standardeinstellung **key.kdb**). Klicken Sie auf **OK**.
4. Geben Sie im Dialogfenster **Aufforderung zur Kennworteingabe** Ihr Kennwort ein und klicken Sie auf **OK**.
5. Wählen Sie im Inhaltsrahmen **Schlüsseldatenbank** die Option **Persönliche Zertifikate** aus und klicken Sie anschließend auf **Export/Import**.
6. Führen Sie im Fenster **Schlüssel exportieren/importieren** folgende Schritte aus:
   - Wählen Sie **Schlüssel exportieren** aus.
   - Wählen Sie die Art der Zieldatenbank aus (z. B. **PKCS12**).
   - Geben Sie den Dateinamen ein oder wählen Sie ihn aus, indem Sie auf **Durchsuchen** klicken.
   - Geben Sie die richtige Adresse an.
7. Klicken Sie auf **OK**.

8. Geben Sie im Dialogfenster **Aufforderung zur Kennworteingabe** das richtige Kennwort ein. Geben Sie es anschließend zur Bestätigung erneut ein und klicken Sie dann auf **OK**, um den ausgewählten Schlüssel in eine andere Schlüsseldatenbank zu exportieren.

Schlüssel importieren

So importieren Sie Schlüssel aus einer anderen Schlüsseldatenbank:

1. Starten Sie IBM Key Management.
2. Wählen Sie im Hauptmenü die Optionen **Schlüsseldatenbankdatei –> Öffnen** aus.
3. Geben Sie im Dialogfenster **Öffnen** den Namen Ihrer Schlüsseldatenbank ein (oder übernehmen Sie die Standardeinstellung **key.kdb**). Klicken Sie auf **OK**.
4. Geben Sie im Dialogfenster **Aufforderung zur Kennworteingabe** Ihr Kennwort ein und klicken Sie auf **OK**.
5. Wählen Sie im Inhaltsrahmen **Schlüsseldatenbank** die Option **Persönliche Zertifikate** aus und klicken Sie anschließend auf **Export/Import**.
6. Führen Sie im Fenster **Schlüssel exportieren/importieren** folgende Schritte aus:
   - Wählen Sie **Schlüssel importieren** aus.
   - Wählen Sie die Art der Schlüsseldatenbankdatei aus (z. B. **PKCS12**).
   - Geben Sie den Dateinamen ein oder wählen Sie ihn aus, indem Sie auf **Durchsuchen** klicken.
   - Wählen Sie die richtige Adresse aus.
7. Klicken Sie auf **OK**.
8. Geben Sie im Dialogfenster **Aufforderung zur Kennworteingabe** Ihr Kennwort ein und klicken Sie auf **OK**.
9. Wählen Sie in der Liste **Schlüsselname** den richtigen Namen aus und klicken Sie auf **OK**.

Zertifizierungsstellen auflisten

So können Sie eine Liste mit vertrauenswürdigen Zertifizierungsstellen (Trusted-CAs) in einer Schlüsseldatenbank anzeigen:

1. Starten Sie IBM Key Management.
2. Wählen Sie im Hauptmenü die Optionen **Schlüsseldatenbankdatei –> Öffnen** aus.
3. Geben Sie im Dialogfenster **Öffnen** den Namen Ihrer Schlüsseldatenbank ein (oder übernehmen Sie die Standardeinstellung **key.kdb**). Klicken Sie auf **OK**.
4. Geben Sie im Dialogfenster **Aufforderung zur Kennworteingabe** Ihr Kennwort ein und klicken Sie auf **OK**.
5. Wählen Sie im Inhaltsrahmen **Schlüsseldatenbank** die Option **Zertifikate des Unterzeichners** aus.
6. Klicken Sie auf **Zertifikate des Unterzeichners**, **Persönliche Zertifikate** oder **Zertifikatanforderungen**, um im Fenster **Schlüsselinformationen** eine Liste der Zertifizierungsstellen anzuzeigen.

Ein von einer Zertifizierungsstelle signiertes Zertifikat empfangen

Gehen Sie wie folgt vor, um ein Zertifikat zu empfangen, das Sie per E-Mail von einer Zertifizierungsstelle erhalten haben, die standardmäßig als Trusted-CA definiert ist (siehe die Liste im Abschnitt „Zertifizierungsstellen" auf Seite 121). Wenn es sich bei der Zertifizierungsstelle (CA), die Ihr Zertifikat ausstellt, nicht um eine in der Schlüsseldatenbank gespeicherte vertrauenswürdige Zertifizierungsstelle handelt, müssen Sie zunächst das Zertifikat dieser Instanz speichern und die CA als vertrauenswürdige CA festlegen. Anschließend können Sie Ihr von der CA signiertes Zertifikat in die Datenbank laden. Sie können kein signiertes Zertifikat von einer Zertifizierungsstelle empfangen, die nicht als vertrauenswürdig eingestuft wurde (siehe Abschnitt „Ein CA-Zertifikat speichern").

So laden Sie das von der Zertifizierungsstelle signierte Zertifikat in eine Schlüsseldatenbank:

1. Starten Sie IBM Key Management.
2. Wählen Sie im Hauptmenü die Optionen **Schlüsseldatenbankdatei –> Öffnen** aus.
3. Geben Sie im Dialogfenster **Öffnen** den Namen Ihrer Schlüsseldatenbank ein (oder übernehmen Sie die Standardeinstellung **key.kdb**). Klicken Sie auf **OK**.
4. Geben Sie im Dialogfenster **Aufforderung zur Kennworteingabe** Ihr Kennwort ein und klicken Sie auf **OK**.
5. Vergewissern Sie sich, dass der Dateiname in der Liste der Datenbanktypen richtig ist.
6. Wählen Sie im Fenster **Schlüsseldatenbank** die Option **Persönliche Zertifikate** aus und klicken Sie anschließend auf **Empfangen**.
7. Geben Sie im Dialogfenster **Zertifikat aus Datei empfangen** im Textfeld **Dateiname des Zertifikats** den Namen einer gültigen Datei mit 64-Bit-Verschlüsselung ein. Klicken Sie auf **OK**.
8. Zum Schließen des Dienstprogramms IBM Key Management klicken Sie im Hauptmenü auf **Schlüsseldatenbankdatei –> Beenden**.

Ein CA-Zertifikat speichern

Zum Aufbau sicherer Verbindungen werden nur Zertifikate akzeptiert, die von einer vertrauenswürdigen Zertifizierungsstelle signiert wurden. Wenn Sie der Liste der vertrauenswürdigen Stellen eine Zertifizierungsstelle hinzufügen möchten, müssen Sie ihr Zertifikat abrufen und als vertrauenswürdig abspeichern. Gehen Sie wie folgt vor, um ein Zertifikat von einer neuen Zertifizierungsstelle zu speichern:

1. Starten Sie IBM Key Management.
2. Wählen Sie im Hauptmenü die Optionen **Schlüsseldatenbankdatei –> Öffnen** aus.
3. Geben Sie im Dialogfenster **Öffnen** den Namen Ihrer Schlüsseldatenbank ein (oder übernehmen Sie die Standardeinstellung **key.kdb**). Klicken Sie auf **OK**.
4. Geben Sie im Dialogfenster **Aufforderung zur Kennworteingabe** Ihr Kennwort ein und klicken Sie auf **OK**.
5. Wählen Sie im Inhaltsrahmen **Schlüsseldatenbank** die Option **Zertifikate des Unterzeichners** aus und klicken Sie auf **Hinzufügen**.

6. Wählen Sie im Dialogfenster **CA-Zertifikat aus Datei hinzufügen** den Namen der 64-Bit-verschlüsselten ASCII-Zertifikatdatei aus oder verwenden Sie dazu die Option **Durchsuchen**. Klicken Sie auf **OK**.
7. Geben Sie im Dialogfenster **Name** einen Namen ein und klicken Sie auf **OK**.
8. Kennzeichnen Sie das Zertifikat als vertrauenswürdig, indem Sie das Markierungsfeld auswählen (Standardeinstellung).

   Anmerkung: Sie können das Markierungsfeld *nach* dem Erstellen des Zertifikats anzeigen, indem Sie die Schaltfläche "Anzeigen/Bearbeiten" anklicken. Das Markierungsfeld ist in der Anzeige enthalten, wird jedoch erst nach dem Hinzufügen des Zertifikats angezeigt.

Standardschlüssel in einer Schlüsseldatenbank anzeigen

Gehen Sie wie folgt vor, um den Eintrag für den Standardschlüssel anzuzeigen:

1. Starten Sie IBM Key Management.
2. Wählen Sie im Hauptmenü die Optionen **Schlüsseldatenbankdatei –> Öffnen** aus.
3. Geben Sie im Dialogfenster **Öffnen** den Namen Ihrer Schlüsseldatenbank ein (oder übernehmen Sie die Standardeinstellung **key.kdb**). Klicken Sie auf **OK**.
4. Geben Sie im Dialogfenster **Aufforderung zur Kennworteingabe** Ihr Kennwort ein und klicken Sie auf **OK**.
5. Wählen Sie im Inhaltsrahmen **Schlüsseldatenbank** die Option **Persönliche Zertifikate** aus und wählen Sie den Namen des CA-Zertifikats aus.
6. Klicken Sie im Fenster **Schlüsselinformationen** auf **Anzeigen/Bearbeiten**, um die Standardschlüsselinformationen des Zertifikats anzuzeigen.

Unterstützte Verschlüsselungsspezifikationen

In der folgenden Tabelle sind die Verschlüsselungsalgorithmen und Hash-Verfahren aufgeführt, die für die SSL-Versionen 2 und 3 verwendet werden.

Generierung von Schlüsselpaaren: RSA 512–1024 (Größe von privaten Schlüsseln)

SSL Version 2

US-Version	Exportversion
RC4 US	RC4 Export
RC2 US	RC2 Export
DES 56-Bit	*nicht zutreffend*
Triple DES US	*nicht zutreffend*
RC4 Export	*nicht zutreffend*
RC2 Export	*nicht zutreffend*

SSL Version 3

US-Version	Exportversion
Triple DES SHA US	DES SHA Export
DES SHA Export	RC2 MD5 Export
RC2 MD5 Export	RC4 MD5 Export
RC4 SHA US	NULL SHA
RC4 MD5 US	NULL MD5
RC4 MD5 Export	NULL NULL
RC4 SHA 56-Bit	*nicht zutreffend*
DES CBC SHA	*nicht zutreffend*
NULL SHA	*nicht zutreffend*
NULL MD5	*nicht zutreffend*
NULL NULL	*nicht zutreffend*

Diese SSL-Spezifikationen können auch direkt durch Editieren der Konfigurationsdatei des Proxy-Servers konfiguriert werden. Einzelheiten hierzu finden Sie in Anhang B, „Anweisungen in der Konfigurationsdatei", auf Seite 175 in den Referenzabschnitten zu den folgenden Anweisungen:

- „V2CipherSpecs - Unterstützte Verschlüsselungsspezifikationen für SSL Version 2 auflisten" auf Seite 285
- „V3CipherSpecs - Unterstützte Verschlüsselungsspezifikationen für SSL Version 3 auflisten" auf Seite 286

128-Bit-Verschlüsselung für Caching Proxy

Es wird nur eine 128-Bit-Verschlüsselungsversion von Caching Proxy bereitgestellt. Die 56-Bit-Version ist nicht mehr verfügbar. Wenn Sie eine ältere Version aktualisieren, können Sie Caching Proxy direkt in Ihrer derzeit installierten 128-Bit- oder 56-Bit-Version installieren. Falls Sie vorher einen 56-Bit-Browser (Export) verwendet haben, müssen Sie einen Upgrade auf einen 128-Bit-Browser durchführen, damit Sie die 128-Bit-Verschlüsselung im Proxy-Server nutzen können.

Falls nach dem Upgrade von einer 56-Bit-Version von Caching Proxy auf die 128-Bit-Version die für die Verschlüsselung von Zertifikaten verwendete Schlüsselgröße auf 1024 gesetzt ist, müssen keine Änderungen an der Konfiguration vorgenommen werden. Ist die Schlüsselgröße auf 512 gesetzt, müssen Sie neue Zertifikate mit einer Schlüsselgröße von 1024 erstellen, damit Sie die 128-Bit-Verschlüsselung des Proxy-Servers nutzen können. Erstellen Sie neue Schlüssel mit dem Dienstprogramm IBM Key Management (iKeyman).

1. Starten Sie IBM Key Management.
   - Geben Sie auf Linux- und UNIX-Plattformen an einer Eingabeaufforderung `gsk7ikm` ein.
   - Klicken Sie unter Windows auf **Start –> Programme –> IBM WebSphere –> Edge Components –> Dienstprogramm Key Management starten**.
2. Wählen Sie im Hauptmenü die Optionen **Schlüsseldatenbankdatei –> Öffnen** aus.

3. Geben Sie im Dialogfenster **Öffnen** den Namen Ihrer Schlüsseldatenbank ein (oder klicken Sie auf **key.kdb**, wenn Sie die Standardeinstellung übernehmen möchten). Klicken Sie dann auf **OK**.
4. Wird das Dialogfenster **Aufforderung zur Kennworteingabe** angezeigt, geben Sie Ihr Kennwort ein und klicken Sie auf **OK**.
5. Wählen Sie im Hauptmenü die Optionen **Erstellen –> Neue Zertifikatanforderung erstellen** aus.
6. Legen Sie im Fenster **Neuen Schlüssel und neue Zertifikatanforderung erstellen** Folgendes fest:
   - **Schlüsselname**: Geben Sie einen Namen ein, mit dem Schlüssel und Zertifikat in der Datenbank gekennzeichnet werden.
   - **Schlüsselgröße**: Wählen Sie **1024** aus.
   - **Name der Organisation**: Geben Sie den Namen der Organisation ein, die dem Schlüssel zugeordnet werden soll.
   - **Land**: Geben Sie Ihren Landescode ein. Sie müssen mindestens zwei Zeichen angeben, z. B. `DE`.
   - **Name der Zertifikatanforderungsdatei**: Geben Sie den Namen der Anforderungsdatei ein oder verwenden Sie einen Standardnamen.
7. Klicken Sie auf **OK**.

Eine detaillierte Beschreibung des Dienstprogramms IBM Key Management finden Sie im Abschnitt „Schlüssel- und Zertifikatverwaltung” auf Seite 120.

Beachten Sie, dass diese Produktversion unter SuSE Linux keine Verschlüsselung unterstützt.

Kapitel 27. Unterstützung für Verschlüsselungshardware aktivieren

Gehen Sie wie folgt vor, wenn die SSL-Handshake-Routine auf eine Hardwareverschlüsselungskarte ausgelagert werden soll:

1. Installieren Sie die Hardwareverschlüsselungskarte gemäß den Anweisungen des Herstellers.
2. Aktivieren Sie SSL für den Caching Proxy. Nähere Informationen finden Sie in Kapitel 26, „Secure Sockets Layer (SSL)", auf Seite 119.
3. Editieren Sie die Anweisung SSLCryptoCard manuell in der Konfigurationsdatei ibmproxy.conf. In den Konfigurations- und Verwaltungsformularen wird für diese Anweisung kein Eintrag angezeigt. Nähere Informationen finden Sie in der Beschreibung der Anweisung SSLCryptoCard im Abschnitt „SSLCryptoCard - Die installierte Verschlüsselungskarte angeben" auf Seite 280.

Kapitel 28. Das Plug-in Tivoli Access Manager verwenden

Mit Tivoli Access Manager (früher Tivoli Policy Director) wird ein Plug-in für Caching Proxy bereitgestellt. Über dieses Plug-in kann Caching Proxy Access Manager zur Authentifizierung und Autorisierung verwenden. Dieses Plug-in bietet Unternehmen, die Access Manager für die Webzugriffssteuerung verwenden, die Möglichkeit, mit der Edge-Technologie zu arbeiten, ohne für den Proxy-Server separate Autorisierungsschemata einrichten zu müssen.

Nähere Informationen zu Tivoli Access Manager finden Sie auf der Produkt-Website http://www.ibm.com/software/tivoli/products/. Nähere Informationen zu den Software- und Hardwarevoraussetzungen und zur Installation des Plug-in finden Sie in der Dokumentation, die mit Tivoli Access Manager geliefert wird.

Anmerkung: Das Plug-in Tivoli Access Manager wird unter Red Hat Linux eventuell nicht unterstützt. Aktuelle Informationen zur Unterstützung des Plug-in auf Linux-Plattformen erhalten Sie von Tivoli.

Konfiguration

Zusammen mit dem Plug-in Access Manager wird ein Konfigurations-Script für Caching Proxy bereitgestellt.

Vor Verwendung des Konfigurations-Script auszuführende Schritte

Gehen Sie vor dem Ausführen des Script wie folgt vor:

- Installieren Sie die erforderliche Software.
- Vergewissern Sie sich, dass der Proxy-Server für die Verwendung von Port 80 (Standardwert) eingestellt ist.
- Konfigurieren Sie die Komponenten LDAP und Access Manager und achten Sie darauf, dass diese Komponenten aktiv sind, wenn Sie das Plug-in Access Manager konfigurieren.
- Vergewissern Sie sich, dass die Administrator-ID für Access Manager und der LDAP-Administratorname vorliegen. Diese Werte benötigen Sie für die Konfiguration des Proxy-Servers.

Konfigurations-Script verwenden

Das Konfigurations-Script hat den Namen **wslconfig.sh** und ist im Verzeichnis /opt/pdweb-lite/bin/ enthalten. Geben Sie bei entsprechender Aufforderung die Administrator-ID für Access Manager und die Administrator-ID für LDAP ein.

Das Konfigurations-Script führt die folgenden Schritte automatisch aus:

- Die Benutzer-ID von Caching Proxy wird auf `root` und die Gruppen-ID auf `other` gesetzt.
- Die Anweisung noLog wird auf `*` gesetzt, so dass keine Einträge in das Zugriffsprotokoll von Caching Proxy geschrieben werden.
- Eine ServerInit-Anweisung mit folgenden Informationen wird erstellt:

```
ServerInit /opt/pdweb-lite/lib/wesauth.so:WTESeal_Init
/opt/pdweb-lite/etc/ibmwesas.conf
```

- Eine PreExit-Anweisung mit folgenden Informationen wird erstellt:

  ```
  PreExit /opt/pdweb-lite/lib/wesauth.so:WTESeal_PreExit
  ```

- Eine Authorization-Anweisung mit folgenden Informationen wird erstellt:

  ```
  Authorization * /opt/pdweb-lite/lib/wesauth.so:WTESeal_Authorize
  ```

- Eine ServerTerm-Anweisung mit folgenden Informationen wird erstellt:

  ```
  ServerTerm /opt/pdweb-lite/lib/wesauth.so:WTESeal_Term
  ```

  Eine Protect-Anweisung und eine Zugriffsschutzkonfiguration (Protection) werden erstellt, die alle Anforderungen an den Authentifizierungsprozess von Access Manager wie folgt weiterleiten:

  ```
  Protection PROXY-PROT {
          ServerId WebSEAL-Lite
          Mask All@(*)
          AuthType Basic
          }
  Protect * PROXY-PROT
  ```

Caching Proxy und Plug-in Access Manager starten

Nachdem der Proxy-Server und das Plug-in Access Manager konfiguriert wurden, führen Sie zum Starten des Proxy-Servers den Befehl **wslstartwte** anstelle des Befehls **ibmproxy start** aus. Mit dem Befehl **wslstartwte** werden automatisch Umgebungsvariablen geladen, die das Plug-in Access Manager zur Initialisierung benötigt. Wenn Sie den Befehl **wslstartwte** nicht zum Starten des Proxy-Servers verwenden, erscheinen Fehlernachrichten zum Plug-in Access Manager. Der entsprechende Befehl zum Stoppen, **ibmproxy stop**, ist auch bei Verwendung des Plug-in gültig.

Kapitel 29. PAC-LDAP-Autorisierungsmodul verwenden

Übersicht

Das PAC-LDAP-Autorisierungsmodul erlaubt Caching Proxy bei Ausführung von Autorisierungs- und Authentifizierungsroutinen den Zugriff auf einen LDAP-Server (Lightweight Directory Access Protocol). Das Modul besteht aus zwei Komponentengruppen: einem Paar gemeinsam genutzter Bibliotheken, durch die die LDAP-Funktionalität zur API von Caching Proxy hinzugefügt wird, und einem PAC-Dämon (Policy Authentication Control). Eine Anweisung ServerInit in der Datei ibmproxy.conf weist die gemeinsam genutzte Bibliothek an, einen oder mehrere PAC-Dämonprozesse zu starten, wenn Caching Proxy gestartet wird. Die gemeinsam genutzten Bibliotheken lesen die Datei paccp.conf, um die Anzahl und Merkmale der PAC-Dämonprozesse zu bestimmen. Während der Initialisierung liest der Dämon die Konfigurationsanweisungen in der Datei pac.conf und die Policy-Informationen in der Datei pacpolicy.conf. Anschließend weist entweder eine Authentication-Anweisung in der Datei ibmproxy.conf den Proxy-Server an, die gemeinsam genutzte Bibliothek jedesmal aufzurufen, wenn eine Authentifizierung erforderlich ist, oder eine Authorization-Anweisung kontrolliert während der Verarbeitung von Standard-HTTP-Anforderungen den Arbeitsablauf in Caching Proxy.

Authentifizierung

Bei der Authentifizierung wird ermittelt, ob die übergebenen Berechtigungsnachweise (Benutzername und Kennwort) gültig sind. Es wird geprüft, ob ein Benutzer in der Registrierungsdatenbank enthalten ist und ob das angegebene Kennwort mit dem in der Registrierungsdatenbank gespeicherten Kennwort übereinstimmt. Diese Aktionen werden ausgeführt, wenn das Modul PAC-LDAP während der Authentifizierung verwendet wird.

Wird das PAC-LDAP-Autorisierungsmodul für die Authentifizierung aktiviert, wird es zum Standard-Repository, aus dem die Benutzer-IDs, Kennwörter und Gruppen abgerufen werden. Wenn eine HTTP-Anforderung den Caching-Proxy-Arbeitsablauf durchläuft, vergleicht jede Protect-Anweisung den angeforderten URL mit ihrer Anforderungsschablone. Bei Übereinstimmung ruft die Protect-Anweisung ein Zugriffsschutzschema auf, das die Server-ID, die Art der zu verwendenden Authentifizierung, die Maskierungsregeln, die auf den Anforderungsclient angewendet werden sollen, sowie die Positionen der Kennwörter und Gruppendateien umfasst. Ist keine Kennwortdatei definiert, werden Benutzer-ID und Kennwort über das PAC-LDAP-Autorisierungsmodul abgerufen. Die Policys des Typs 0, 1, 2 und 3 definieren Authentifizierungsschemata. Bei erfolgreicher Authentifizierung wird die Anforderung bearbeitet. Andernfalls gibt Caching Proxy den Fehler 401 an den Client zurück.

Autorisierung

Bei der Autorisierung wird ermittelt, ob ein Benutzer die erforderliche Berechtigung für den Zugriff auf eine geschützte Ressource besitzt. Wenn das Modul PAC-LDAP verwendet wird, werden während dieses Prozesses Autorisierungsregeln angewendet, die in der Datei pacpolicy.conf für die HTTP-Anforderung enthalten sind.

Ist das PAC-LDAP-Autorisierungsmodul für die Autorisierung aktiviert, werden auf die HTTP-Anforderung Autorisierungsregeln aus der Datei pacpolicy.conf angewendet. Wenn eine HTTP-Anforderung den Caching-Proxy-Arbeitsablauf durchläuft, vergleicht jede Protect-Anweisung den angeforderten URL mit ihrer Anforderungsschablone. Bei Übereinstimmung ruft die Protect-Anweisung ein Zugriffsschutzschema auf. In diesem Fall entspricht das Zugriffsschutzschema der vom PAC-LDAP-Autorisierungsmodul kontrollierten Autorisierungsroutine. Die Authorization-Anweisung vergleicht den angeforderten URL mit ihrer Anforderungsschablone. Bei Übereinstimmung wird das PAC-LDAP-Authorisierungsmodul aufgerufen. Policys vom Typ 4, die in der Datei pacpolicy.conf definiert sind, grenzen die Authentifizierung weiter ein, die für verschiedene URL-Anforderungen erforderlich ist.

Lightweight Directory Access Protocol (LDAP)

LDAP erlaubt den interaktiven Zugriff auf X.500-Verzeichnisse bei minimalem Verbrauch von Systemressourcen. IANA hat LDAP den TCP-Port 389 und den UDP-Port 389 zugeordnet. Nähere Informationen finden Sie in RFC 1777, der LDAP definiert.

Installation

Alle Komponenten des PAC-LDAP-Autorisierungsmoduls werden automatisch installiert, wenn das Caching-Proxy-System von WebSphere Application Server Version 6.0.1 installiert wird. Auf Linux- und UNIX-Systemen werden ein Verzeichnis für die Caching-Proxy-Bibliothek (./lib/), ein Verzeichnis für die Bibliothek des PAC-LDAP-Autorisierungsmoduls (./lib/plugins/pac/), ein Verzeichnis für die Binärdateien (./bin/) und ein Verzeichnis für die Konfiguration (./etc/) im Verzeichnis /opt/ibm/edge/cp/ erstellt. Anschließend werden von den Verzeichnissen /usr/lib/, /usr/sbin/ und /etc aus symbolische Verbindungen zu diesen produktspezifischen Verzeichnissen erstellt.

Verzeichnisstruktur

Linux- und UNIX-Verzeichnis	**Windows-Verzeichnis**	**Inhalt**
/opt/ibm/edge/cp/	\Programme\IBM\edge\cp\	CP-Basisverzeichnis (*cp_root*)
cp_root/sbin/	\Programme\IBM\edge\cp\Bin\	CP-Binärdateien und -Scripts
/usr/sbin/		Symb. Verbindungen zu *cp_root*/sbin/
cp_root/etc/	\Programme\IBM\edge\cp\etc\	CP-Konfigurationsdatei
/etc/		Symb. Verbindungen zu *cp_root*/etc/
cp_root/lib/	\Programme\IBM\edge\cp\lib\ plugins\	CP-Bibliotheken
cp_root/lib/ plugins/pac/	\Programme\IBM\edge\ cp\lib\plugins\pac\	Bibliotheken des PAC-LDAP-Autorisierungsmoduls

Linux- und UNIX-Verzeichnis	Windows-Verzeichnis	Inhalt
/usr/lib/		Symb. Verbindungen zu *cp_root*/lib/ und *cp_root*/lib/plugins/pac/
cp_root/server_root/pac/data/	\Programme\IBM\edge\cp\server_root\pac\data\	Datenspeicher für PAC-LDAP-Autorisierungsmodul
cp_root/server_root/pac/creds/	\Programme\IBM\edge\cp\server_root\pac\creds\	Berechtigungsnachweise für PAC-LDAP-Autorisierungsmodul

Dateien des LDAP-Plug-in

Linux- und UNIX-Dateiname	Windows-Dateiname	Beschreibung
libpacwte.so	pacwte.dll	gemeinsam genutzte Bibliothek
libpacman.so	pacman.dll	gemeinsam genutzte Bibliothek
pacd_restart.sh	pacd_restart.bat	Script für Neustart des PAC-Dämons
paccp.conf, pac.conf, pacpolicy.conf	paccp.conf, pac.conf, pacpolicy.conf	Konfigurations- und Policy-Dateien

Zusätzliche Voraussetzungen für sichere PACD-LDAP-Serververbindungen

Vom LDAP-Clientpaket vorausgesetztes GSKit installieren

Damit zwischen dem PACD-Dämon und dem LDAP-Server sichere SSL-Verbindungen (SSL = Secure Sockets Layer) hergestellt werden können, müssen Sie das vom LDAP-Clientpaket vorausgesetzte GSKit-Paket installieren. GSKit 7 ist die erforderliche Version und wird standardmäßig auf der Caching-Proxy-Maschine bereitgestellt. Möglicherweise ist diese jedoch nicht die Version, die der LDAP-Client auf der Maschine voraussetzt. Es ist möglich, verschiedene GSKit-Versionen für verschiedene Prozesse auf einer Maschine zu verwenden.

Kopieren Sie die GSKit-Schlüsseldatei nach `$pacd_creds_dir/pac_keyring.kdb` und das Kennwort in die Datei `$pacd_creds_dir/pac_keyring.pwd`.

Anmerkung: Informationen zu den Voraussetzungen für GSKit auf dem LDAP-Server finden Sie in der Dokumentation zu IBM Directory Server auf der folgenden Website:
http://www.ibm.com/software/tivoli/products/directory-server/

Definition der Umgebungsvariablen LD_PRELOAD für Linux-Systeme

Auf Linux-Systemen muss die Umgebungsvariable LD_PRELOAD wie folgt konfiguriert werden, damit SSL-Verbindungen zwischen dem PACD-Dämon und dem LDAP-Server hergestellt werden können. Setzen Sie die Variable auf den folgenden Wert:

```
LD_PRELOAD=/usr/lib/libstdc++-libc6.1-1.so.2
```

Die in diesem Abschnitt beschriebenen GSKit-Voraussetzungen gelten auch für Linux-Systeme.

Die Datei ibmproxy.conf editieren, um das PAC-LDAP-Autorisierungsmodul zu aktivieren

Drei Anweisungen, ServerInit, Authorization oder Authentication und ServerTerm, müssen zum Abschnitt mit den API-Anweisungen der Datei ibmproxy.conf hinzugefügt werden, damit das PAC-LDAP-Autorisierungsmodul initialisiert wird. Zum Erstellen dieser Anweisungen müssen Sie entweder die Datei ibmproxy.conf manuell editieren, oder, falls der Proxy-Server bereits aktiv ist, die Konfigurations- und Verwaltungsformulare in einem Internet-Browser aufrufen und das Formular ″Verarbeitung von API-Anforderungen″ öffnen. (Klicken Sie nacheinander auf **Serverkonfiguration** –> **Anforderungsverarbeitung**–> **Verarbeitung von API-Anforderungen**.) Jede Anweisung in der Konfigurationsdatei des Proxy-Servers muss in einer gesonderten Zeile stehen, auch wenn die hier aufgeführten Beispiele aus Gründen der besseren Lesbarkeit Zeilenumbrüche enthalten.

Beachten Sie die Prototypanweisungen (in Form von Kommentaren) im API-Abschnitt der Datei ibmproxy.conf. Diese API-Anweisungen sind in einer zweckmäßigen Reihenfolge angeordnet. Wenn Sie API-Anweisungen hinzufügen, um neue Funktionen und Plug-in-Module zu aktivieren, sollten Sie die Anweisungen wie im Prototypabschnitt der Konfigurationsdatei gezeigt anordnen. Alternativ dazu können Sie, falls erforderlich, die Kommentarzeichen für API-Anweisungen entfernen und API-Anweisungen editieren, um die Unterstützung für jede gewünschte Funktion oder jedes gewünschte Plug-in hinzuzufügen.

Die Anweisung ServerInit unterstützt drei Argumente: (1) den vollständig qualifizierten Pfad der gemeinsam genutzten Bibliothek, (2) den Funktionsaufruf und (3) den vollständig qualifizierten Pfad der Datei paccp.conf. Das erste und zweite Argument werden durch einen Doppelpunkt (:) getrennt. Das zweite und dritte Argument werden durch ein Leerzeichen getrennt. Das erste Argument und das dritte Argument sind systemspezifisch und abhängig davon, wo die Plug-in-Komponenten installiert sind. Das zweite Argument ist in der gemeinsam genutzten Bibliothek fest codiert und muss wie gezeigt eingegeben werden. Beim Erstellen einer Anweisung ServerInit im Formular ″Verarbeitung von API-Anforderungen″ müssen das zweite und dritte Argument im Feld **Funktionsname** eingegeben werden. Das dritte Argument wird in der Spalte **IP-Schablone** angezeigt.

Die Anweisung Authorization unterstützt drei Argumente: (1) eine Anforderungsschablone, (2) den vollständig qualifizierten Pfad der gemeinsam genutzten Bibliothek und (3) den Funktionsnamen. Die HTTP-Anforderungen werden mit der Anforderungsschablone verglichen, um zu bestimmen, ob eine Anwendungsfunktion aufgerufen werden muss. Die Anforderungsschablone kann ein Protokoll, eine Domäne und einen Host enthalten. Sie darf als vorangestelltes Zeichen einen Schrägstrich (/) und als Platzhalterzeichen einen Stern (*) verwenden.

Beispielsweise sind folgende Schablonen gültig: /front_page.html , http://www.ics.raleigh.ibm.com, /pub*, /* und *. Der Funktionsname ist der Name, den Sie Ihrer Anwendungsfunktion im Programm zugewiesen haben. Dieser Name ist fest codiert und muss wie gezeigt eingegeben werden. Die ersten zwei Argumente werden durch ein Leerzeichen getrennt. Die letzten beiden Argumente werden durch einen Doppelpunkt (:) getrennt.

Die Anweisung Authentication unterstützt zwei Argumente: (1) den vollständig qualifizierten Pfad der gemeinsam genutzten Bibliothek und (2) den Funktionsnamen. Diese Argumente werden durch einen Doppelpunkt (:) getrennt. Das erste Argument ist systemspezifisch und abhängig davon, wo die gemeinsam genutzte Bibliothek installiert ist. Die URL-Schablone für das erste Argument muss mit dem Dokumentstammverzeichnis (/) beginnen, wenn Caching Proxy als Reverse Proxy verwendet wird. Das zweite Argument ist in der gemeinsam benutzten Bibliothek fest codiert und muss wie angezeigt eingegeben werden.

Die Anweisung ServerTerm unterstützt zwei Argumente: (1) den vollständig qualifizierten Pfad der gemeinsam genutzten Bibliothek und (2) den Funktionsnamen. Diese Argumente werden durch einen Doppelpunkt (:) getrennt. Das erste Argument ist systemspezifisch und abhängig davon, wo die gemeinsam genutzte Bibliothek installiert ist. Das zweite Argument ist in der gemeinsam genutzten Bibliothek fest codiert und muss wie gezeigt eingegeben werden. Diese Anweisung beendet den PAC-Dämon, wenn der Proxy-Server beendet wird. Ist der Dämon einem anderen Eigner zugeordnet als der Proxy-Server, kann der Proxy-Server den Dämon möglicherweise nicht stoppen. In diesem Fall muss der Dämon vom Administrator manuell gestoppt werden.

```
ServerInit Pfad_der_gemeinsamen Bibliothek:pacwte_auth_init
Pfad_der_Konfigurations-/Policy-Datei
```

Linux- und UNIX-Systeme:

```
ServerInit /usr/lib/libpacwte.so:pacwte_auth_init /etc/pac.conf
```

Beispiel für Windows:

```
ServerInit C:\Progra ~1\IBM\edge\cp\lib\plugins\
  pac\pacwte.dll:pacwte_auth_init C:\Progra ~1\IBM\edge\cp
Authorization Anforderungsschablone Pfad_der_gemeinsamen_Bibliothek:pacwte_auth_policy
```

Linux- und UNIX-Systeme:

```
Authorization http://* /usr/lib/libpacwte.so:pacwte_auth_policy
```

Beispiel für Windows:

```
Authorization http://* C:\Programme\IBM\edge\cp\lib\plugins\
  pac\pacwte.dll:pacwte_auth_policy
Authentication BASIC Pfad_der_gemeinsamen_Bibliothek:pacwte_auth_policy
```

Linux- und UNIX-Systeme:

```
Authentication BASIC /usr/lib/plugins/pac/libpacwte.so:pacwte_auth_policy
```

Beispiel für Windows:

```
Authentication BASIC C:\Programme\IBM\edge\cp\lib\plugins\
  pac\pacwte.dll:pacwte_auth_policy
ServerTerm Pfad_der_gemeinsamen_Bibliothek:pacwte_shutdown
```

Linux- und UNIX-Systeme:

```
ServerTerm /usr/lib/libpacwte.so:pacwte_shutdown
```

Beispiel für Windows:

```
ServerTerm BASIC C:\Programme\IBM\edge\cp\lib\plugins\
  pac\bin\pacwte.dll:pacwte_shutdown
```

Die Konfigurationsdateien des PAC-LDAP-Autorisierungsmoduls editieren

Die Konfigurations- und Policy-Dateien des PAC-LDAP-Autorisierungsmoduls müssen in einem Texteditor manuell editiert werden. Der Name einer Anweisung muss vom ersten Argument durch einen Doppelpunkt (:) getrennt werden. Mehrere Argumente werden durch Kommas (,) voneinander getrennt. Die Konfigurations- und Policy-Datei enthält Anmerkungen, die Sie beim Editieren unterstützen. Wichtige Policy-Anweisungen sind unten aufgeführt.

paccp.conf

Die Datei paccp.conf wird während der Initialisierung von Caching Proxy von den gemeinsam genutzten Bibliotheken gelesen und enthält Definitionen (Zeilengruppe [PAC_MAN_SERVER]) für jeden PAC-Dämon, der gestartet wird. Für jeden PAC-Dämon muss eine eigene Zeilengruppe [PAC_MAN_SERVER] vorhanden sein.

```
[PAC_MAN_SERVER]
hostname:                    # Name des PAC-Dämons
port:                        # Port, an dem pacd Anforderungen empfängt.

[PACWTE_PLUGIN]
hostname_check:[true|false]  # Aktiviert die DNS-Suche. Damit ibmproxy
                             # funktioniert, muss die DNS-Suche aktiviert sein.
```

pac.conf

Die Datei pac.conf legt den LDAP-Server fest, zu dem der PAC-Dämon eine Verbindung herstellt.

```
[PAC_MAN_SERVER]
hostname:                    # Name des PAC-Dämons
port:                        # Port, an dem pacd Anforderungen empfängt.
conn_type:ssl                # Setzen Sie dies auf Kommentar,
                             # wenn SSL nicht verwendet wird.
authentication_sequence: [primary|secondary|none]
authorization_sequence:  [primary|secondary|none]

[LDAP_SERVER]
hostname:                    # Hostname des LDAP-Servers.
port:389                     # Der Port, an dem LDAP Anforderungen empfängt.
ssl_port:636                 # SSL-Port, der vom LDAP-Server verwendet wird.
admin_dn:                    # Benutzer mit Zugriffsberechtigung für LDAP-Server.
                             # Geben Sie admin_dn:NULL für die Unterstützung
                             # anonymer Bindungen an.
search_base:                 # Der Abschnitt der LDAP-Baumstruktur, der nach
                             # Policy-Informationen durchsucht werden soll.
                             # Sofern nicht erforderlich, geben
                             # Sie search_base:NULL an.
search_key:                  # ID-Feld, das gesucht werden soll.

[CACHE]
cred_cache_enabled [TRUE|FALSE] # Berechtigungs-Cache aktivieren.
cred_cache_min_size:100      # Mindestanzahl an Berechtigungen, die im Cache
                             # von pacd gespeichert werden sollen.
cred_cache_max_size:64000    # Maximale Anzahl an Berechtigungen, die im Cache
```

```
                            # von pacd gespeichert werden sollen.
cred_cache_expiration:86400  # Das Verfallsdatum einer Berechtigung.
policy_cache_enabled:[TRUE|FALSE] # Aktiviert/inaktiviert den Policy-Cache.
policy_cache_min_size:100    # Mindestanzahl an Policy-Einträgen, die im Cache
                            # gespeichert werden sollen.
policy_cache_max_size:64000  # Maximale Anzahl an Policy-Einträgen, die im Cache
                            # gespeichert werden sollen.
policy_cache_expiration:86400 # Verfallsdatum eines Policy-Eintrags.
```

pacpolicy.conf

Jede LDAP-Policy verwendet die folgende Schablone in der Konfigurations- und Policy-Datei. Jede Policy muss mit dem Schlüsselwort POLICY beginnen, das in Großbuchstaben geschrieben und in eckige Klammern eingeschlossen werden muss.

```
[POLICY]
default_policy:[grant|deny]  # Beschreibt die Standard-Policy für Benutzer,
                            # die nicht im Abschnitt POLICY beschrieben sind.
pac_client_hostname:         # Die Instanzen von Caching-Proxy, die
                            # eine Policy-Liste verwenden dürfen.
id:                          # Die ID für den LDAP-Eintrag oder IP/Hostnamen
                            # (Platzhalterzeichen werden unterstützt,
                            # z. B. *.ibm.com).
grant:[true|false]           # true bedeutet den Zugriff erlauben, false bedeutet
                            # den Zugriff verweigern.
type:[0|1|2|3|4]             # 0 ist ein LDAP-Eintrag, der eine Gruppe ist
                            # 1 ist ein LDAP-Eintrag, der keine Gruppe ist
                            # 2 ist eine IP-Adresse
                            # 3 ist ein Hostname
                            # 4 ist ein URL
propagate:[true|false]       # true bedeutet, dass die Zugriffsrechte (Zugriff
                            # erteilen oder verweigern) an alle
                            # untergeordneten Einträge oder Member
                            # weitergegeben werden.
stop_entry:[entry|NULL]      # Die Weitergabe der Zugriffsrechte endet
                            # bei diesem Eintrag. Ist die ID eine Gruppe,
                            # muss stop_entry auf NULL gesetzt werden.
                            # stop_entry kann auf eine IP-Adresse oder einen
                            # Hostnamen angewendet werden. Jede Anweisung der
                            # Art stop_entry muss in einer separaten Zeile stehen.
exception_entry:[entry|NULL] # Beim Zuordnen der Zugriffsrechte werden diese
                            # Einträge übersprungen, jedoch nicht die
                            # untergeordneten Baumstrukturen. Hierbei kann
                            # es sich um eine Liste von Einträgen handeln.
                            # exception_entry kann auf eine Gruppe, eine
                            # IP-Adresse oder einen Hostnamen angewendet werden.
                            # Jede Anweisung vom Typ exception_entry muss in
                            # einer separaten Zeile stehen.
Exception_type:
Exception:
```

Platzhalterzeichen (*) werden nur für den letzten Teil einer IP-Adresse oder für den ersten Teil eines Hostnamens in den Anweisungen `id` und `stop_entry` akzeptiert. Für Anweisungen vom Typ `exception_entry` werden keine Platzhalterzeichen unterstützt. Für alle Felder in LDAP-Einträgen werden ebenfalls keine Platzhalterzeichen unterstützt.

Die Angabe mehrerer Policy-Anweisungen wird unterstützt, wobei in den Fällen, in denen Policy-Anweisungen in Konflikt miteinander stehen, der Wert "false" immer Priorität hat, d. h., eine einzige Verweigerung in einer beliebigen Policy blockiert den Zugriff. Die Reihenfolge, in der die Policy-Anweisungen in der Konfigurations- und Policy-Datei aufgelistet sind, ist nicht relevant und stellt keine Priorität her.

Eine Reihe von Policy-Beispielen finden Sie in der Datei pacpolicy.conf im Verzeichnis mit den Konfigurationsdateien.

Anmerkung: Verschachtelte Gruppen können Policy-Anweisungen nicht von ihren Elterngruppen übernehmen. Nur Policys, für die die Gruppe als explizites Member definiert ist, werden auf eine Gruppe angewendet.

pac_ldap.cred erstellen

Erstellen Sie eine Textdatei mit dem Namen pac_ldap.cred im Verzeichnis */cp_root*/server_root/pac/creds. Diese Datei enthält das Kennwort für den Benutzernamen, der mit der Anweisung admin_dn in der Datei pac.conf angegeben wurde.

Anmerkung: Für die Unterstützung von anonymen Bindungen müssen Sie die Anweisung admin_dn in der Datei pac.conf in admin_dn:NULL ändern und eine Pseudozeichenfolge in der Datei pac_ldap.cred angeben.

Der PAC-Dämon verschlüsselt das Kennwort, wenn er die Datei zum ersten Mal liest.

Setzen Sie auf Linux- und UNIX-Plattformen die folgenden Befehle ab, um die Datei pac_ldap.cred zu erstellen:

```
cd cp_root/server_root/pac/creds
echo "password" > pac_ldap.cred
chown nobody pac_ldap.cred
chgrp nobody pac_ldap.cred
(unter SuSE Linux muss chgrp nogroup pac_ldap.cred verwendet werden)
```

Zum Erstellen der Datei auf einer Windows-Plattform geben Sie das Kennwort in einer Textdatei ein und speichern Sie diese Datei im Verzeichnis server_root\pac\creds\.

pacd starten und stoppen

Der LDAP-Berechtigungsdämon wird als pacd-Prozess ausgeführt. Sie können den LDAP-Berechtigungsdämon mit den bereitgestellten Scripts starten, ohne Caching Proxy zu unterbrechen. Führen Sie das Script pacd wie folgt aus:

- Auf Linux- und UNIX-Plattformen:

  ```
  /usr/sbin/pacd_restart.sh Benutzer-ID_für_pacd
  ```

- Auf Windows-Plattformen:

  ```
  C:\Programme\IBM\edge\cp\Bin\pacd_restart.bat CP-Installationsstammverzeichnis
  ```

Anmerkung: Der pacd-Prozess kann auch ausgeführt werden, nachdem der Caching-Proxy-Server heruntergefahren wurde. Dazu muss auf AIX-Systemen der Befehl **stopsrc -ibmproxy** und auf HP-UX-, Linux- und Solaris-Systemen der Befehl **ibmproxy -stop** ausgeführt werden. Der Prozess pacd kann mit dem Befehl **kill** wie folgt sicher beendet werden:

```
kill -15 pacd-Prozess-ID
```

Unter HP-UX: Möglicherweise laden das Plug-in PAC-LDAP und pacd zur Laufzeit nicht alle abhängigen gemeinsam genutzten Bibliotheken. Prüfen Sie daher vor ihrer Verwendung, ob die Systemvariablen wie folgt festgelegt sind:

```
SHLIB_PATH=/usr/lib:/usr/IBMldap/lib
PATH=/usr/IBMldap/bin:$PATH
PATH=/usr/IBMldap/bin
```

`/usr/IBMldap/` ist der Standardinstallationspfad für den LDAP-Client unter HP-UX. Ändern Sie die Angaben für PATH und SHLIB_PATH entsprechend ab, falls der LDAP-Client in einem anderen Verzeichnis installiert ist. Falls diese Variablen *nicht* festgelegt werden, können folgende Fehler auftreten:

- Nachdem das Plug-in PAC-LDAP aktiviert wurde, wird die folgende Fehlernachricht in das Fehlerprotokoll geschrieben:

  ```
  "Fehler bei der Serverinitialisierung: Einige Funktionen aus
   dem DLL-Modul /opt/ibm/edge/cp/lib/plugins/pac/libpacwte.sl
   wurden nicht geladen."
  ```

- Beim Versuch, /usr/sbin/pacd zu starten, wird der folgende Verknüpfungsfehler angezeigt:

  ```
  "/usr/lib/dld.sl: Can't find path for shared library: libibmldap.sl
   /usr/lib/dld.sl: No such file or directory
   Abort"
  ```

Unter AIX: Wenn Sie pacd mit IBM Tivoli Directory Server 5.2 starten, kann das Modul PAC-LDAP möglicherweise nicht geladen werden. In diesem Fall wird der folgende Fehler angezeigt:

```
exec(): 0509-036 Das Programm /usr/sbin/pacd kann aufgrund der folgenden
                     Fehler nicht geladen werden:
        0509-022 Das Modul /usr/lib/libpacman.a kann nicht geladen werden.
        0509-150 Das abhängige Modul libldap.a konnte nicht geladen werden.
        0509-022 Das Modul libldap.a kann nicht geladen werden.
```

Sie können dieses Problem umgehen, indem Sie die folgende symbolische Verbindung erstellen: `ln -s /usr/lib/libibmldap.a /usr/lib/libldap.a`

Anmerkung: Nach der Konfiguration der LDAP-Authentifizierung in Caching Proxy wird der folgende Fehler angezeigt:

```
Für Uid konnte kein Wert extrahiert werden. Rückkehrcode: 3
```

Dieser Fehler wird auch angezeigt, wenn die LDAP-Authentifizierung ordnungsgemäß funktioniert, und kann deshalb ignoriert werden.

Teil 6. Caching Proxy überwachen

Dieser Teil enthält Anweisungen zum Überwachen von Caching Proxy mit Protokollen und dem Überwachungsprogramm für Serveraktivitäten (Server Activity Monitor).

Dieser Teil enthält die folgenden Kapitel:

Kapitel 30. Protokollierung konfigurieren

Zum Anpassen der Protokollierung können Sie entweder die Konfigurations- und Verwaltungsformulare verwenden oder Anweisungen in der Konfigurationsdatei des Proxy-Servers editieren. Sie können folgende Optionen festlegen:

- Pfade und Dateinamen zum Speichern der Protokolldateien
- Filter zum Ein- und Ausschließen von Informationen in den Zugriffsprotokolldateien
- Verwaltungsoptionen zum Archivieren und Löschen von Protokollen

Informationen zu Protokollen

Caching Proxy kann drei Arten von Zugriffsprotokollen sowie ein Ereignisprotokoll und ein Fehlerprotokoll erstellen:

- Zugriffsprotokolle:
  - **Zugriffsprotokoll** - Protokolliert lokale Verwaltungsanforderungen an Caching Proxy selbst.
  - **Cache-Zugriffsprotokoll** - Protokolliert Anforderungen für Objekte, die sich im Cache befinden.
  - **Proxy-Zugriffsprotokoll** - Protokolliert Anforderungen, die von Ursprungsservern weitergeleitet werden.
- **Ereignisprotokoll** - Protokolliert Cache-Informationsnachrichten.
- **Fehlerprotokoll** - Protokolliert Fehlernachrichten zu Caching Proxy.

Caching Proxy beginnt jeden Tag um 00:00 Uhr mit der Erstellung neuer Protokolldateien. Ist der Proxy-Server um 00:00 Uhr nicht aktiv, werden die neuen Protokolle beim ersten Start an diesem Tag erstellt. Sie können für jede Protokolldatei das Verzeichnis sowie das Dateinamenspräfix angeben. Jede Protokolldatei, die erstellt wird, enthält außerdem ein Datumssuffix im Format *.Mmmttjjjj* (z. B. .Apr142000).

Da Protokolle sehr viel Plattenspeicherplatz belegen können, sollten Sie, um Fehler zu vermeiden, die Protokolldateien nicht auf der Speichereinheit speichern, auf der sich das Betriebssystem und der Cache befinden, sondern auf einer separaten Speichereinheit. Darüber hinaus sollten Sie die Routinen zur Protokollverwaltung, wie im Abschnitt „Protokolle verwalten und archivieren" auf Seite 153 beschrieben, konfigurieren.

Protokolldateinamen und grundlegende Optionen

Zum Festlegen der Basiskonfiguration für die Protokolle des Proxy-Servers wählen Sie in den Konfigurations- und Verwaltungsformularen **Serverkonfiguration –> Protokollierung –> Protokolldateien** aus. Geben Sie für jede Protokolldatei, die Sie verwenden möchten, den Pfad und Dateinamen an. Der aktuelle Dateiname für jedes Protokoll wird im zugehörigen Textfeld angezeigt. Wenn Sie keinen Pfad angegeben haben, wird der Standardwert angezeigt.

Die in den Protokollen des Proxy-Servers aufgezeichneten Informationen werden nicht automatisch in das Systemprotokoll geschrieben. Caching Proxy kann jedoch so konfiguriert werden, dass die Protokolleinträge zusätzlich oder ausschließlich in

das Systemprotokoll geschrieben werden. Wählen Sie im Formular **Protokolldateien** das Markierungsfeld **Information in Syslog protokollieren** aus. Das Systemprotokoll muss vor Auswahl dieser Option bereits erstellt worden sein.

Wenn die Informationen des Proxy-Server-Protokolls nur in das Systemprotokoll geschrieben werden sollen, müssen Sie die Konfigurationsdatei des Proxy-Servers editieren. Informationen hierzu finden Sie im zugehörigen Abschnitt „LogToSyslog - Angeben, ob Zugriffsinformationen an das Systemprotokoll gesendet werden sollen (nur Linux und UNIX)" auf Seite 236.

Zugehörige Anweisungen in der Konfigurationsdatei

Wenn Sie die Protokolle in der Konfigurationsdatei des Proxy-Servers definieren möchten, lesen Sie in Anhang B, „Anweisungen in der Konfigurationsdatei", auf Seite 175 die zugehörigen Abschnitte zu folgenden Anweisungen:

- „AccessLog - Pfad für die Zugriffsprotokolldatei angeben" auf Seite 177
- „CacheAccessLog - Pfad für Cache-Zugriffsprotokolldateien angeben" auf Seite 189
- „ErrorLog - Die Datei zur Protokollierung von Serverfehlern angeben" auf Seite 214
- „EventLog - Den Pfad der Ereignisprotokolldatei angeben" auf Seite 216
- „LogToSyslog - Angeben, ob Zugriffsinformationen an das Systemprotokoll gesendet werden sollen (nur Linux und UNIX)" auf Seite 236
- „ProxyAccessLog - Name und Pfad für die Proxy-Zugriffsprotokolldatei angeben" auf Seite 263

Filter für Zugriffsprotokolle

In Zugriffsprotokollen werden die Aktivitäten der Hostmaschine, des Proxy-Servers und des Cache aufgezeichnet. Für jede Zugriffsanforderung, die der Proxy-Server empfängt, wird im entsprechenden Zugriffsprotokoll ein Eintrag eingefügt, der folgende Angaben enthält:

- Inhalt der Anforderung
- Zeitpunkt der Anforderung
- Ursprung der Anforderung
- Methode der Anforderung
- Art der Datei, die der Server als Antwort auf die Anforderung gesendet hat
- Rückkehrcode, der anzeigt, ob die Anforderung erfolgreich war
- Größe der gesendeten Daten

Zugriffsfehler werden im Fehlerprotokoll des Servers protokolliert.

Gründe für die Steuerung der Protokollierung

Es gibt mehrere Gründe, weshalb die Informationen, die protokolliert werden, eingeschränkt werden sollten:

- Protokollgröße verkleinern:

  Die Protokolldateien eines ausgelasteten Servers können so groß werden, dass sie den gesamten Plattenspeicherplatz des Servers belegen. Standardmäßig werden alle Zugriffsanforderungen protokolliert, was bedeutet, dass nicht nur für eine HTML-Seite, sondern auch für jedes Bild, das diese Seite enthält, Protokolleinträge geschrieben werden. Werden nur wichtige Zugriffsanforderungen protokolliert, reduziert sich die Anzahl der Einträge im Protokoll erheblich.

Beispielsweise können Sie die Zugriffsprotokolle so konfigurieren, dass Zugriffsanforderungen für HTML-Seiten protokolliert werden, jedoch keine Anforderungen für GIF-Bilder.

- Spezielle Informationen erfassen:

  Wenn Sie beispielsweise wissen möchten, wer von außerhalb Ihres Unternehmens auf den Server zugreift, können Sie Zugriffsanforderungen, die von IP-Adressen innerhalb Ihres Unternehmens stammen, herausfiltern. Wenn Sie die Größe der Zielgruppe für eine bestimmte Website ermitteln möchten, können Sie ein Zugriffsprotokoll erstellen, in dem nur die Zugriffsanforderungen für diesen URL aufgezeichnet werden.

Aus Zugriffsprotokollen ausgeschlossene Informationen werden in keinem Zugriffsbericht aufgezeichnet und sind für eine spätere Verwendung nicht verfügbar. Wenn Sie aus diesem Grund nicht sicher sind, wie viele Zugriffsinformationen Sie protokollieren müssen, sollten Sie beim Herausfiltern von Informationen vorsichtig sein, bis Sie mehr Erfahrung mit der Überwachung des Servers gesammelt haben.

Filter für Zugriffsprotokoll konfigurieren

Zugriffsprotokolleinträge können basierend auf einem der folgenden Attribute gefiltert werden:

- URL (Dateien oder Verzeichnisse)
- IP-Adresse oder Hostname
- Benutzeragenten
- Methode
- MIME-Typ
- Rückkehrcode

Zum Festlegen von Filtern wählen Sie in den Konfigurations- und Verwaltungsformularen **Serverkonfiguration –> Protokollierung –> Ausschlüsse aus dem Zugriffsprotokoll** aus. Legen Sie nur die gewünschten Ausschlüsse fest. Sie müssen nicht alle Kategorien verwenden.

- Listen Sie im Abschnitt **Anforderungen für die folgenden Verzeichnisse oder Dateien nicht im Zugriffsprotokoll protokollieren** die URL-Pfade auf, für die keine Protokolleinträge erstellt werden sollen.
- Listen Sie im Abschnitt **Anforderungen von den folgenden Benutzeragenten nicht protokollieren** die Proxy-Agenten auf, für die keine Protokolleinträge erstellt werden sollen.
- Listen Sie im Abschnitt **Anforderungen von den folgenden Hostnamen oder IP-Adressen nicht protokollieren** die Hostnamen oder IP-Adressen auf, für die keine Protokolleinträge erstellt werden sollen.
- Wählen Sie im Abschnitt **Keine Anforderungen protokollieren, für die die folgenden Methoden verwendet wurden:** die Markierungsfelder der Methoden aus, für die keine Protokolleinträge erstellt werden sollen.
- Wählen Sie im Abschnitt **Anforderungen für Dateien mit den folgenden MIME-Typen nicht protokollieren** die Markierungsfelder der MIME-Typen aus, für die keine Protokolleinträge erstellt werden sollen.

  Anmerkung: Diese Anweisung betrifft nur das Proxy-Zugriffsprotokoll. Es ist nicht möglich, ein Protokoll zu filtern, das diese zwischengespeicherten Objekte auflistet, jedoch nicht deren MIME-Typen. Verwenden Sie dazu AccessLogExcludeURL.

- Wählen Sie im Abschnitt **Anforderungen mit den folgenden Rückkehrcodes nicht protokollieren** die Markierungsfelder der Rückkehrcodes aus, für die keine Protokolleinträge erstellt werden sollen.

Klicken Sie auf **Übergeben**.

Zugehörige Anweisungen in der Konfigurationsdatei

Wenn Sie die Filter für das Zugriffsprotokoll in der Konfigurationsdatei des Proxy-Servers definieren möchten, lesen Sie in Anhang B, „Anweisungen in der Konfigurationsdatei", auf Seite 175 die zugehörigen Abschnitte zu folgenden Anweisungen:

- „AccessLogExcludeMethod - Protokolleinträge für Dateien oder Verzeichnisse unterdrücken, die mit einer bestimmten Methode angefordert werden" auf Seite 178
- „AccessLogExcludeMimeType - Einträge für bestimmte MIME-Typen im Zugriffsprotokoll unterdrücken" auf Seite 179
- „AccessLogExcludeReturnCode - Protokolleinträge für bestimmte Rückkehrcodes unterdrücken" auf Seite 179
- „AccessLogExcludeURL - Protokolleinträge für bestimmte Dateien oder Verzeichnisse unterdrücken" auf Seite 180
- „AccessLogExcludeUserAgent - Protokolleinträge für bestimmte Browser unterdrücken" auf Seite 180
- „NoLog - Protokolleinträge für bestimmte Hosts oder Domänen, die mit einer Schablone übereinstimmen, unterdrücken" auf Seite 244

Standardeinstellungen der Protokolle

- **Standardpfade**

  Alle Protokolle sind in der Standardkonfiguration von Caching Proxy aktiviert. Sie sind im Unterverzeichnis logs/ des Installationsverzeichnisses gespeichert. Die Standardpfade lauten wie folgt:

  – Lokale Zugriffsprotokolle (für Verwaltungszugriffe):
    - Linux und UNIX: /opt/ibm/edge/cp/server_root/logs/local
    - Windows: *Laufwerk*:\Programme\IBM\edge\cp\logs\local
  – Cache-Zugriffsprotokoll:
    - Linux and UNIX: /opt/ibm/edge/cp/server_root/logs/cache
    - Windows: *Laufwerk*:\Programme\IBM\edge\cp\logs\cache
  – Proxy-Zugriffsprotokoll:
    - Linux und UNIX: /opt/ibm/edge/cp/server_root/logs/proxy
    - Windows: *Laufwerk*:\Programme\IBM\edge\cp\logs\proxy
  – Fehlerprotokoll:
    - Linux und UNIX: /opt/ibm/edge/cp/server_root/logs/error
    - Windows: *Laufwerk*:\Programme\IBM\edge\cp\logs\error
  – Ereignisprotokoll:
    - Linux und UNIX: /opt/ibm/edge/cp/server_root/logs/event
    - Windows: *Laufwerk*:\Programme\IBM\edge\cp\logs\event

  Jeder Protokolldateiname ist eine Kombination aus dem Basisdateinamen und einem Datumssuffix im Format *.Mmmttjjjj*, z. B. proxy.Feb292000.

- **Standardformate**

  Protokolle werden standardmäßig im einheitlichen Dateiformat (Common File Format) gespeichert. Ein kombiniertes Protokollformat ist auch verfügbar und kann durch Einfügen der folgenden Zeile zur Konfigurationsdatei des Proxy-Servers (ibmproxy.conf) festgelegt werden:

  ```
  LogFileFormat combined
  ```

  Das kombinierte Protokollformat ähnelt dem einheitlichen Format, verfügt jedoch über zusätzliche Felder mit Informationen zu Herkunftsadressen, Benutzeragenten und Cookies. Die Ortszeit ist das Standardzeitformat.
- **Standardinhalt**

  Standardmäßig werden alle Zugriffsanforderungen im entsprechenden Zugriffsprotokoll aufgezeichnet, während im Systemprotokoll keine Informationen zu Zugriffen protokolliert werden. Informationen zu Fehlern werden nur im Fehlerprotokoll und Informationen zu Ereignissen nur im Ereignisprotokoll aufgezeichnet.
- **Standardverwaltung**

  In der Standardkonfiguration werden Protokolle weder archiviert noch gelöscht.

Protokolle verwalten und archivieren

Caching Proxy verwaltet die Protokolle mit Hilfe eines Plug-in. Nähere Informationen finden Sie in Anhang B, „Anweisungen in der Konfigurationsdatei", auf Seite 175 auf der zugehörigen Seite zur Anweisung „Midnight - API-Plug-in für die Archivierung von Protokollen angeben" auf Seite 242 in der Konfigurationsdatei.

Sie können festlegen, wie die Tagesprotokolle archiviert und entfernt werden sollen. Grundlegende Optionen:

- Protokolle, die ein angegebenes Alter überschreiten, komprimieren und entfernen.
- Protokolle, die ein angegebenes Alter erreichen oder deren Protokollkategorie eine angegebene Gesamtgröße erreicht, entfernen.
- Jeden Tag um Mitternacht ein eigenes Programm zur Verwaltung und Archivierung der Protokolle ausführen.

Standardmäßig werden die Protokolle des aktuellen und des vorangegangenen Tages von den Verwaltungsagenten niemals entfernt. Alle Protokolle des aktuellen Tages und alle Cache-Zugriffsprotokolle des vorangegangenen Tages werden von den Verwaltungsagenten niemals komprimiert.

Zum Konfigurieren der Protokollverwaltung wählen Sie in den Konfigurations- und Verwaltungsformularen **Serverkonfiguration –> Protokollierung –> Protokollarchivierung** aus. Legen Sie in diesem Formular mit dem Dropdown-Fenster die Verwaltungsmethode fest.

- Wenn Sie **Löschen** ausgewählt haben, legen Sie die Werte für das Alter und/oder die Dateigröße fest, auf deren Basis ermittelt wird, welche Protokolle gelöscht werden müssen. Werden Dateien nach Alter und Größe gelöscht, werden zuerst alle Dateien gelöscht, die das maximale Alter überschreiten, bevor die Dateien gelöscht werden, die die maximale Größe überschreiten. Werden Dateien nach Größe gelöscht, werden ältere Protokolle zuerst gelöscht.

- Wenn Sie **Komprimieren** ausgewählt haben, legen Sie das Alter der zu komprimierenden Protokolle und den Befehl fest, der für die Komprimierung der Protokolldateien verwendet werden soll (einschließlich aller Parameter). Legen Sie zudem das maximale Alter der Protokolle fest. Nach dem Komprimieren der Protokolle löscht der Verwaltungsagent komprimierte Protokolle, die das maximale Alter überschreiten.

Zugehörige Anweisungen in der Konfigurationsdatei

Informationen zum Konfigurieren der Protokollarchivierung unter Verwendung der Konfigurationsdatei des Proxy-Servers finden Sie in Anhang B, „Anweisungen in der Konfigurationsdatei", auf Seite 175 auf den Seiten zu folgenden Anweisungen:

- „CompressAge - Zeitpunkt für die Komprimierung von Protokollen angeben" auf Seite 201
- „CompressDeleteAge - Zeitpunkt zum Löschen von Protokollen angeben" auf Seite 202
- „CompressCommand - Komprimierungsbefehl und zugehörige Parameter angeben" auf Seite 201
- „LogArchive - Verhalten der Protokollarchivierung angeben" auf Seite 235
- „Midnight - API-Plug-in für die Archivierung von Protokollen angeben" auf Seite 242
- „PurgeAge - Altersgrenze für ein Protokoll angeben" auf Seite 267
- „PurgeSize - Begrenzung für die Größe des Protokollarchivs angeben" auf Seite 267.

Szenario einer Protokolldatei

Das folgende Beispiel zeigt, wie Sie die Protokollierung an Ihre Erfordernisse anpassen können. Angenommen, Sie haben Caching Proxy gerade erst erworben und installiert. Sie möchten den Server so einrichten, dass Zugriffs- und Fehlerinformationen unter Berücksichtigung folgender Kriterien protokolliert werden:

- Die Protokolle müssen eine im Ortszeitformat angegebene Zeitmarke und ein einheitliches Protokolldateiformat (Common Log File Format) verwenden.
- Die Zugriffsprotokolle müssen gelöscht werden, wenn sie älter als 30 Tage sind oder eine Gesamtgröße von 25 MB erreichen.
- Folgende Anforderungsarten sollen nicht in den Zugriffsprotokollen protokolliert werden:
  - Anforderungen für GIF-Bilder
  - Anforderungen für Hosts, deren IP-Adresse mit dem Muster 130.128.*.* übereinstimmt
  - Umleitungsanforderungen (diese Anforderungen geben einen Rückkehrcode zwischen 300 und 399 zurück)

Um den Caching-Proxy-Server so zu konfigurieren, dass er die Protokolle gemäß diesen Kriterien verwaltet, wählen Sie in den Konfigurations- und Verwaltungsformularen zunächst **Serverkonfiguration –> Protokollieren** aus.

1. Falls erforderlich, rufen Sie das Formular **Protokolldateien** auf, um die Pfade zu den Zugriffsprotokolldateien anzugeben. (Standardpfade sind vorgegeben.)
2. Legen Sie im Formular **Protokollarchivierung** fest, wie die Dateien archiviert werden sollen:
   - Legen Sie als Archivierungsmethode **Löschen** fest.
   - Füllen Sie die Felder unter **Wenn Löschen verwendet wird** wie folgt aus:
     – **Protokolle löschen, die älter sind als** `30` **Tage**
     – **Protokolle löschen, die größer sind als** `25` **MB**
3. Legen Sie im Formular **Ausschlüsse aus dem Zugriffsprotokoll** folgende Filter für die Protokolleinträge fest:
   - Fügen Sie in der Liste **Anforderungen von den folgenden Hostnamen oder IP-Adressen nicht protokollieren** im Feld **Ausgeschlossener Host** den Eintrag `130.128.*.*` hinzu.
   - Wählen Sie unter **Anforderungen für Dateien mit den folgenden MIME-Typen nicht protokollieren** das Markierungsfeld **image/gif** aus.
   - Wählen Sie unter **Anforderungen mit den folgenden Rückkehrcodes nicht protokollieren** das Markierungsfeld **(3xx) Umleitung** aus.

Nach Ausführung dieser Schritte werden in der Konfigurationsdatei des Proxy-Servers folgende Zeilen erzeugt:

```
LogArchive purge
PurgeAge 30
PurgeSize 25
AccessLogExcludeURL *.gif
NoLog 130.128.*.*
AccessLogExcludeReturnCode 300
```

Kapitel 31. Überwachung der Serveraktivität

Die Überwachung der Serveraktivität (Server Activity Monitor) von Caching Proxy zeigt Statistiken zu Server- und Netzleistung, den Server- und Netzstatus sowie die Zugriffsprotokolleinträge an. Diese Überwachungsfunktion kann fern genutzt werden und muss nicht auf derselben Maschine installiert sein, auf der der Server ausgeführt wird. Die Überwachung der Serveraktivität ist standardmäßig aktiviert und erfordert keine Konfiguration.

Die Funktion zur Überwachung der Serveraktivität (Server Activity Monitor) kann auf zwei Arten aufgerufen werden:

- Geben Sie in einem Webbrowser den folgenden URL ein. Ersetzen Sie "Name.Ihres.Servers" durch den vollständigen Namen Ihres Servers:

  `http://Name.Ihres.Servers/Usage/Initial`
- Wählen Sie in den Konfigurations- und Verwaltungsformularen **Überwachung der Serveraktivität** aus.

Anders als bei anderen Formularen im Konfigurationsclient werden mit den Formularen dieser Kategorie keine Konfigurationen für den Server festgelegt, sondern Daten zur Serverauslastung angezeigt. Diese Formulare enthalten wesentlich mehr Informationen, als in einem Konsolfenster angezeigt werden können.

In den folgenden Abschnitten werden die Informationen erläutert, die die **Überwachung der Serveraktivität** liefert. Außerdem werden Empfehlungen gegeben, wie Sie diese Informationen zur Leistungsoptimierung nutzen können.

Unter **Überwachung der Serveraktivität** sind mehrere Seiten verfügbar:

- **Aktivitätsstatistik**
- **Netzstatistik**
- **Zugriffsstatistik**
- **Statistik über Proxyzugriff**
- **Cache-Statistik**
- **Übersicht über die Cache-Aktualisierung**

Auf jeder dieser Seiten wird die Schaltfläche **Aktualisieren** angezeigt. Wenn Sie diese Schaltfläche anklicken, werden die Informationen aktualisiert.

Aktivitätsstatistik

Tabelle 4 zeigt ein Beispiel der Seite **Aktivitätsstatistik**.

Tabelle 4. Aktivitätsstatistik

Aktivitätsstatistik	
Verbindungen	1 aktiv, 431 maximal
Antwortzeit	Nicht verfügbar
Durchsatz	0 Verbindungen/Sekunde
Heute verarbeitete Anforderungen	0
Summe verarbeiteter Anforderungen	114
Anforderungsfehler	3

Mit diesen Statistiken zur Serveraktivität kann der Datenverkehr auf dem Server unter den Aspekten Anzahl der Zugriffsanforderungen, Antwortzeit, Durchsatz, Anzahl Anforderungen, die heute verarbeitet wurden, Gesamtzahl der verarbeiteten Anforderungen und Fehler überwacht werden. Die folgenden Konfigurationsänderungen haben Auswirkungen auf die Seite **Aktivität**.

- **Anzahl der aktiven Threads** - Damit wird festgelegt, wie viele Threads für Serveranforderungen eingesetzt werden sollen. Sie können die Anzahl der verfügbaren Threads abhängig von der Größe des vorhandenen Arbeitsspeichers erhöhen bzw. verringern. Um die Anzahl der aktiven Threads zu ändern, wählen Sie in den Konfigurations- und Verwaltungsformularen **Serverkonfiguration –> Systemverwaltung –> Leistung** aus oder editieren Sie in der Konfigurationsdatei die Anweisung MaxActiveThreads. (Siehe Abschnitt „MaxActiveThreads - Die maximale Anzahl aktiver Threads angeben" auf Seite 238.)
- **Persistente Verbindungen** - Gibt an, ob der Proxy-Server persistente Verbindungen mit einem Client zulässt. Diese Einstellung kann sich auf den Netzdurchsatz auswirken. Um diese Einstellung in den Konfigurations- und Verwaltungsformularen zu ändern, wählen Sie **Proxy-Konfiguration –> Leistung des Proxy-Servers** aus, um persistente Verbindungen zu aktivieren oder zu inaktivieren. Wählen Sie **Serverkonfiguration –> Systemverwaltung** aus, um festzulegen, wie die Verbindungen verwaltet werden sollen. Informationen zum Ändern dieser Einstellungen in der Konfigurationsdatei finden Sie in den Referenzabschnitten zu den folgenden Anweisungen:
  - „MaxPersistRequest - Die maximale Anzahl Anforderungen angeben, die über eine persistente Verbindung empfangen werden können" auf Seite 240
  - „PersistTimeout - Wartezeit zwischen Clientanforderungen angeben" auf Seite 249
  - „ProxyPersistence - Persistente Verbindungen zulassen" auf Seite 265

Netzstatistik

Tabelle 5 zeigt ein Beispiel der Seite **Netzstatistik**.

Tabelle 5. Netzstatistik

Netzstatistik	
Abgehende Daten:	1KB/Sekunde
Ankommende Daten:	1KB/Sekunde
Eingesparte Bandbreite	3 KB (0KB/Sekunde)
Heute eingesparte Bandbreite:	0 KB (0 KB/Sekunde)

Das Formular **Netzstatistik** enthält Informationen zu dem Netz, in dem der Proxy-Server läuft, einschließlich der Datenübertragungsgeschwindigkeiten für gesendete und empfangene Bytes.

Zugriffsstatistik

Auf der Seite **Zugriffsstatistik** werden die letzten 20 Einträge aus den Zugriffsprotokollen angezeigt. Die Seite zeigt die aktuellen Einträge aus dem Zugriffsprotokoll des Proxy-Servers (in schwarzer Schrift) und aus dem Cache-Zugriffsprotokoll (in blauer Schrift) an. Sie können die Anzeige anpassen, indem Sie die zu protokollierenden Objekte festlegen. Nähere Informationen zu den Zugriffsprotokollstatistiken finden Sie im Abschnitt „Filter für Zugriffsprotokolle" auf Seite 150.

Statistik über Proxyzugriff

Das Formular **Statistik über Proxyzugriff** enthält Informationen zur Aktivität des Proxy-Servers, z. B. welche URLs angefordert wurden und ob diese aus dem Cache bereitgestellt wurden. Nach den URLs werden die an die Clients zurückgegebenen Rückkehrcodes sowie die Dateigröße in Byte angezeigt. Die folgenden Einstellungen können die Proxy-Zugriffsstatistik verbessern:

- Verwenden Sie die automatische Cache-Aktualisierung, um die Wahrscheinlichkeit zu erhöhen, dass ein angefordertes Dokument im Cache enthalten ist. Nähere Informationen hierzu finden Sie in Kapitel 20, „Den Cache-Agenten für automatische Aktualisierung und Vorabladen konfigurieren", auf Seite 95.
- Erhöhen Sie die Mindesthaltezeit für Dateien im Cache. Nähere Informationen hierzu finden Sie im Abschnitt „Cache-Aktualität konfigurieren" auf Seite 92.
- Speichern Sie keine Dateien im Cache, die von der lokalen Domäne bereitgestellt werden. Obwohl diese Einstellung die Anzahl der Anforderungen, die aus dem Cache beantwortet werden, eher reduziert, hat sie keine negativen Auswirkungen auf die Leistung, wenn die Dateien aus dem lokalen Intranet genauso schnell zurückgegeben werden wie aus dem Cache (manchmal sogar schneller). Nähere Informationen hierzu finden Sie in Kapitel 18, „Zwischenzuspeichernde Inhalte steuern", auf Seite 85.

Cache-Statistik

Ist das Caching aktiviert, zeigt die Seite **Cache-Statistik** aktuelle Informationen zu den Cache-Zugriffen an. Unter anderem werden die folgenden Informationen zu Cache und Index angezeigt:

- Ob der Cache derzeit funktionsbereit ist oder durch einen Serverstart neu indexiert wird
- Ob eine Garbage-Collection ausgeführt wird
- Anzahl der Cache-Treffer

Viele Cache-Konfigurationsoptionen wirken sich auf die Ergebnisse der Cache-Statistik aus (siehe Teil 4, „Proxy-Server-Caching konfigurieren", auf Seite 75).

Übersicht über die Cache-Aktualisierung

Wurde der Cache-Agent so konfiguriert, dass Dateien vorab in den Cache geladen werden, zeigt die Seite **Übersicht über die Cache-Aktualisierung** Informationen zur letzten Ausführung des Cache-Agenten an. Der Cache-Agent muss mindestens einmal ausgeführt werden, damit Informationen angezeigt werden können. Wenn Sie die Arbeitsweise des Cache-Aktualisierungsagenten ändern möchten, müssen Sie Folgendes berücksichtigen:

- Wenn der größte Teil des Datenverkehrs in Ihrem Intranet nicht für lokale Websites bestimmt ist, sollten Sie unter Umständen das Caching in der lokalen Domäne inaktivieren. Nähere Informationen hierzu finden Sie in Kapitel 18, „Zwischenzuspeichernde Inhalte steuern", auf Seite 85.
- Wenn viele Clients eine Seite anfordern, die nicht im Cache-Zugriffsprotokoll angezeigt wird, können Sie den zu ladenden URL manuell festlegen. Anweisungen hierzu finden Sie im Abschnitt „Zugehörige Anweisungen in der Konfigurationsdatei des Proxy-Servers" auf Seite 99.
- Passen Sie die Anzahl der am häufigsten aufgerufenen URLs an, die vorab geladen werden sollen. Anweisungen hierzu finden Sie im Abschnitt „Zugehörige Anweisungen in der Konfigurationsdatei des Proxy-Servers" auf Seite 99.

- Geben Sie die maximale Ausführungsdauer für den Cache-Agenten an. Anweisungen hierzu finden Sie im Abschnitt „Zugehörige Anweisungen in der Konfigurationsdatei des Proxy-Servers" auf Seite 99.

Anhang A. Caching-Proxy-Befehle verwenden

Dieser Anhang enthält eine Referenz für die Caching-Proxy-Befehle.

Befehl cgiparse

Zweck

Verwenden Sie den Befehl **cgiparse**, um die Umgebungsvariable QUERY_STRING für CGI-Scripts syntaktisch zu analysieren (Parsing). Wenn die Umgebungsvariable QUERY_STRING nicht definiert ist, liest der Befehl die CONTENT_LENGTH-Zeichen aus der Standardeingabe. Die gesamte zurückgegebene Ausgabe wird in die Standardausgabe geschrieben.

Format

```
cgiparse -Flag [Wert]
```

Parameter

Im Folgenden sind die gültigen Flags, die zugehörigen Optionen (-k -f -v -r -i -s -p -c -q -P) und ihre Funktionen aufgeführt:

-keywords | -k
Führt eine syntaktische Analyse von QUERY_STRING nach Schlüsselwörtern durch. Schlüsselwörter werden entschlüsselt und in die Standardausgabe geschrieben - ein Schlüsselwort pro Zeile.

-form | -f
Führt eine syntaktische Analyse von QUERY_STRING als Formularanforderung durch. Gibt eine Zeichenfolge zurück, die bei der Auswertung durch die Shell Shell-Variablen mit dem Präfix FORM_, gefolgt von einem Feldnamen, definiert. Feldwerte sind der Inhalt der Variablen.

-value *Feldname* **| -v** *Feldname*
Führt eine syntaktische Analyse von QUERY_STRING als Formularanforderung durch. Gibt nur den Wert von *Feldname* zurück.

-read | -r
Liest die CONTENT_LENGTH-Zeichen aus der Standardeingabe und schreibt diese in die Standardausgabe.

-init | -i
Falls QUERY_STRING nicht definiert ist, wird der Wert aus der Standardeingabe gelesen und eine Anweisung SET zurückgegeben, die QUERY_STRING auf diesen Wert setzt. Kann mit den Methoden GET und POST verwendet werden. Eine typische Verwendung ist:

```
eval 'cgiparse -init'
```

Dieser Befehl setzt die Umgebungsvariable QUERY_STRING, unabhängig davon, ob die Methode GET oder POST verwendet wurde.

Bei Verwendung der Methode GET kann **cgiparse** in einem Script mehrfach aufgerufen werden. cgiparse darf bei Verwendung der Methode POST nur einmal angegeben werden. Wenn die Methode POST verwendet wird und die Standardeingabe gelesen wurde, findet der nächste Befehl **cgiparse** die Standardeingabe leer vor und wartet unbegrenzt.

-sep *Trennzeichen* **| -s** *Trennzeichen*
Gibt die Zeichenfolge an, die zum Trennen mehrerer Werte verwendet wird. Wenn Sie das Flag **-value** verwenden, ist das Standardtrennzeichen ein Zeilenvorschubzeichen. Wenn Sie das Flag **-form** verwenden, ist das Standardtrennzeichen ein Komma (,).

-prefix *Präfix* | **-p** *Präfix*
Wenn dieses Flag zusammen mit den Flags **-POST** und **-form** verwendet wird, gibt es das Präfix an, das zum Erstellen von Namen für Umgebungsvariablen verwendet werden soll. Der Standardwert ist ″FORM_″.

-count | -c
Wenn dieses Flag zusammen mit den Flags **-keywords**, **-form** und **-value** verwendet wird, gibt es die Anzahl der Elemente zurück, die sich auf diese Flags beziehen.

-keywords | -k
Gibt die Anzahl Schlüsselwörter zurück.

-form | -f
Gibt die Anzahl der eindeutigen Felder zurück (mehrere Werte werden als ein Feld gezählt).

-value *Feldname* | **-v** *Feldname*
Gibt die Anzahl der Werte für *Feldname* zurück (wenn es kein Feld mit dem Namen *Feldname* gibt, ist die Ausgabe `0`).

-*Zahl*
Wird zusammen mit **-keywords**, **-form** und **-value** verwendet und gibt die angegebene Anzahl von Vorkommen in Bezug auf diese Flags zurück.

-keywords
Gibt das *n*te Schlüsselwort zurück. (Zum Beispiel wird bei der Angabe **-2 -keywords** das zweite Schlüsselwort zurückgegeben.)

-form
Gibt alle Werte des *n*ten Felds zurück. (Zum Beispiel werden bei der Angabe **-2 -form** alle Werte des zweiten Felds zurückgegeben.)

-value *Feldname*
Gibt den *n*ten der Werte von Feld *Feldname* zurück. (Zum Beispiel wird bei der Angabe **-2 -value -whatsit** der zweite Wert des Felds **whatsit** als Ausgabe ausgegeben).

-quiet | -q
Unterdrückt alle Fehlernachrichten. (Ein Exit-Status ungleich Null weist auf einen Fehler hin.)

-POST | -P
Informationen aus der Standardeingabe (oder, falls ein Dateiname verwendet werden soll, die Datei stdin) werden direkt entschlüsselt und in Shell-Variablen aufgelöst. QUERY_STRING wird nicht verwendet. Die Verwendung von **-POST** entspricht der Verwendung der Optionen **-init** und **-form** nacheinander.

Beispiele

In den folgenden Beispielen wird die Tatsache ignoriert, dass QUERY_STRING bereits vom Server definiert wurde. $ steht in den folgenden Beispielen für die Eingabeaufforderung der Bourne-Shell.

- Schlüsselwortsuche

```
$ QUERY_STRING="is+2%2B2+really+four%3F"
$ export QUERY_STRING
$ cgiparse -keywords
is
2+2
really
four?
$
```

- Syntaxanalyse für alle Formularfelder

```
$ export QUERY_STRING="name1=Value1&name2=Value2%3f+That%27s+right%21";
$ cgiparse -form
FORM_name1='Value1'; FORM_name2='Value2? That'\''s right!'
$ eval `cgiparse -form`
$ set | grep FORM
FORM_name1="Value1"
FORM_name2="Value2? That's right!"
$
```

- Extrahieren von nur einem Feldwert

```
$ QUERY_STRING="name1=value1&name2=Second+value%3F+That'\''s%27s
$ cgiparse -value name1
value1
$ cgiparse -value name2
Second value? That's right!
$
```

Ergebnisse

0 Erfolgreich

1 Ungültige Befehlszeile

2 Umgebungsvariablen nicht korrekt definiert

3 Die angeforderten Informationen konnten nicht abgerufen werden (beispielsweise ist kein solches Feld vorhanden, oder QUERY_STRING enthält Schlüsselwörter, wenn Werte für das Formularfeld angefordert werden.)

Anmerkung: Wenn einer dieser Fehlercodes angezeigt wird, werden möglicherweise noch weitere Informationsnachrichten ausgegeben. Die Nachricht hängt vom ausgeführten Befehl ab.

Befehl cgiutils

Zweck

Verwenden Sie den Befehl **cgiutils** in NPH-Programmen (No-Parse Header), um eine vollständige HTTP-1.0-Anwort zu generieren.

Anmerkung: Wenn Sie Ihre eigenen NPH-Programme zur Verfügung stellen möchten, damit eigene Rückgabewerte zurückgegeben werden, muss der Name des Programms mit **nph-** beginnen. Auf diese Weise wird verhindert, dass der Server-Header Ihren Rückgabewert mit dem Standardrückgabewert des Servers überschreibt

Format

```
cgiutils -Flag [Wert]
```

Wenn der *Wert* Leerzeichen enthält, setzen Sie ihn in Anführungszeichen (″″).

Parameter

-version
Gibt die Versionsnummer zurück.

-nodate
Gibt den Header **Date:** nicht zurück.

-noel
Gibt nach Headern keine Leerzeile zurück. Dies ist nützlich, wenn den ersten Header-Zeilen weitere MIME-Header folgen sollen.

-status *nnn*
Gibt die vollständige HTTP-Antwort mit dem Statuscode *nnn* anstelle einer Teilmenge der HTTP-Header zurück. Verwenden Sie dieses Flag nicht, wenn Sie nur den Header **Expires:** anzeigen möchten.

-reason *Erklärung*
Gibt die Ursachenzeile für die HTTP-Antwort an. Sie können dieses Flag nur zusammen mit dem Flag **-status** *nnn* verwenden.

-ct [*Typ/Subtyp***]**
Gibt den MIME-Header für den Inhaltstyp an. In diesem Beispiel wird der MIME-Inhaltstyp text/html definiert:

```
cgiutils -ct text/html
```

Wenn Sie den *Typ/Subtyp* weglassen, wird der MIME-Inhaltstyp auf den Standardwert text/plain festgelegt. In diesem Beispiel wird der MIME-Inhaltstyp auf text/plain festgelegt.

```
cgiutils -ct
```

-ce *Codierung*
Gibt den MIME-Header für die Inhaltscodierung an. Beispiel:

```
cgiutils -ce  x-compress
```

-cl *Sprachencode*
Gibt den MIME-Header für die Inhaltssprache an. Beispiel:

```
cgiutils -cl en_UK
```

-length *nnn*
Gibt den MIME-Header für die Inhaltslänge an.

-expires *Zeitangabe*
Gibt den MIME-Header für die Verfallszeit (**Expires:**) an. Dieses Flag gibt die Lebensdauer (das Verfallsdatum eines Dokuments) in einer beliebigen Kombination von Tagen, Stunden, Minuten und Sekunden an. Dies ist die Zeitdauer, die ein Dokument als gültig angesehen wird. Beispiel:

```
cgiutils -expires 2 days 12 hours
```

Mit dem Befehl **cgiutils** wird die Zeitspanne, die Sie angeben, zur aktuellen Westeuropäischen Zeit addiert, um das Verfallsdatum zu errechnen. Das Verfallsdatum wird im HTTP-Format in den Header **Expires:** gestellt.

-expires now
Generiert den Header **Expires:**, der mit dem Header **Date:** übereinstimmt.

-uri *URI*
Gibt den URI (Universal Resource Identifier) für das zurückgegebene Dokument an. URI ist synonym mit URL zu verstehen.

-extra *xxx: yyy*
Gibt einen zusätzlichen Header an, der für den Befehl **cgiutils** nicht anders angegeben werden kann.

Beispiele

- In diesem Beispiel wird das Verfallsdatum für den Header **Expires:** errechnet.

  ```
  cgiutils -expires "1 year 3 months 2 weeks 4 days 12 hours 30 mins"
  ```

- Im folgenden Beispiel werden ein Statuscode und eine Ursache angegeben, und der Header **Expires:** wird gleich dem Header **Date:** gesetzt.

  ```
  cgiutils -status 200 -reason "Virtual doc follows" -expires now
  ```

  Dadurch können Header wie dieser hier generiert werden:

  ```
  HTTP/1.0 200 Virtual doc follows
  MIME-Version: 1.0
  Server: IBM-ICS
  Date: Tue, 05 Jan 1996 03:43:46 GMT
  Expires: Tue, 05 Jan 1996 03:43:46 GM
  ```

  Der Befehl cgiutils generiert automatisch den Header **Server:**, da dieser in der CGI-Umgebung zur Verfügung steht. Das Feld **Date:** wird ebenfalls automatisch generiert, es sei denn, das Flag **-nodate** ist angegeben.

  Hinter der Ausgabe wird eine Leerzeile eingefügt, um das Ende des Abschnitts mit den MIME-Headern zu markieren. Falls Sie hinter diesem Abschnitt weitere Header einfügen möchten, verwenden Sie das Flag **-noel** (NO-Empty-Line), wie im nächsten Beispiel gezeigt.

- Wenn hinter der Header-Zeile keine Leerzeile eingefügt werden soll, verwenden Sie das Flag **-noel**:

  ```
  cgiutils -noel -expires "2 days" -nodate
  ```

  ```
  HTTP/1.0 200 Virtual doc follows
  MIME-Version: 1.0
  Server: IBM-ICS
  Expires: Tue, 07 Jan 1996 03:43:46 GMT
  ```

Befehl htadm

Zweck

Verwenden Sie den Befehl **htadm**, um die Kennwortdateien des Servers zu verwalten. Der Server verwendet Kennwortdateien, um den Zugriff auf Ihre Dateien zu steuern. Mit diesem Befehl können Sie einer Kennwortdatei einen Benutzernamen hinzufügen, einen Benutzer aus einer Kennwortdatei löschen, das Kennwort eines Benutzers überprüfen und eine leere Kennwortdatei erstellen. Außerdem können Sie das Kennwort eines Benutzers ändern, indem Sie zuerst das Kennwort des Benutzers löschen und anschließend ein neues Kennwort erstellen.

Anmerkung: Wenn Sie den Befehl **htadm** verwenden, um einen Benutzer hinzuzufügen, ein Kennwort zu ändern oder zu überprüfen, müssen Sie das Kennwort in der Befehlszeile eingeben. Da der Befehl das Kennwort in der Befehlszeile so schnell wie möglich löscht, ist es sehr unwahrscheinlich, dass Sie ein Kennwort eines Benutzer sehen können, indem Sie die Prozessliste auf der Maschine anzeigen (z. B. mit dem Befehl **ps**).

Format

```
htadm -Flag  [Wert]
```

Parameter

-adduser *Kennwortdatei Benutzername* **[***Kennwort* **[***echter Name***]]**

Fügt der Kennwortdatei einen Benutzer und ein Kennwort hinzu. Wenn Sie den Befehl nur mit dem Parameter *Kennwortdatei* eingeben, werden Sie zur Eingabe der anderen Parameter aufgefordert.

Kennwortdatei

Pfad und Name der Kennwortdatei, die dem Benutzer hinzugefügt werden soll.

Benutzername

Der Name des Benutzers, der hinzugefügt werden soll.

Verwenden Sie nur alphabetische und numerische Zeichen für den Benutzernamen. Verwenden Sie keine Sonderzeichen.

Der Befehl schlägt fehl, wenn es bereits einen Benutzer mit demselben Namen in der Kennwortdatei gibt.

Kennwort

Das Kennwort, das Sie für den Benutzernamen definieren möchten.

Kennwörter können bis zu 32 Zeichen lang sein. Verwenden Sie nur alphabetische und numerische Zeichen für das Kennwort. Verwenden Sie keine Sonderzeichen.

Anmerkungen:

1. Einige Browser können keine Kennwörter lesen und senden, die länger als acht Zeichen sind. Auf Grund dieser Begrenzung erkennt der Server, wenn Sie ein Kennwort definieren, das länger als acht Zeichen ist, entweder das vollständige Kennwort oder nur die ersten acht Zeichen des Kennworts als gültige Zeichen.

2. Beim Benutzernamen und Kennwort des Administrators wird zwischen Groß- und Kleinschreibung auch dann unterschieden, wenn das Betriebssystem diese Unterscheidung nicht macht. Geben Sie Benutzernamen und Kennwort mit dem Befehl `htadm` exakt ein, wenn Sie auf die Konfigurations- und Verwaltungsformulare zugreifen.

echter Name
Ein Kommentar oder Name zur Identifizierung des Benutzernamens, den Sie hinzufügen. Die Eingabe wird in die Kennwortdatei geschrieben.

-deluser *Kennwortdatei* **[***Benutzername***]**
Löscht einen Benutzer aus der Kennwortdatei. Wenn Sie den Befehl nur mit dem Parameter *Kennwortdatei* eingeben, werden Sie zur Eingabe des Parameters *Benutzername* aufgefordert.

Kennwortdatei
Der Pfad und Name der Kennwortdatei, in der ein Benutzer gelöscht werden soll.

Benutzername
Der Name des Benutzers, der gelöscht werden soll. Der Befehl schlägt fehl, wenn der Benutzername, den Sie angeben, nicht in der Kennwortdatei vorhanden ist.

-passwd *Kennwortdatei* **[***Benutzername* **[***Kennwort***]]**
Ändert das Kennwort für einen Benutzernamen, der bereits in der Kennwortdatei definiert ist. Wenn Sie den Befehl nur mit dem Parameter *Kennwortdatei* eingeben, werden Sie zur Eingabe der anderen Parameter aufgefordert.

Kennwortdatei
Pfad und Name der Kennwortdatei, die den Benutzernamen enthält, dessen Kennwort geändert werden soll.

Benutzername
Der Benutzername, dessen Kennwort geändert werden soll. Der Befehl schlägt fehl, wenn der Benutzername, den Sie angeben, nicht in der Kennwortdatei vorhanden ist.

Kennwort
Das neue Kennwort, das Sie für den Benutzernamen definieren möchten.

Kennwörter können bis zu 32 Zeichen lang sein. Verwenden Sie nur alphabetische und numerische Zeichen für das Kennwort. Verwenden Sie keine Sonderzeichen.

Anmerkungen:

1. Einige Browser können keine Kennwörter lesen und senden, die länger als acht Zeichen sind. Auf Grund dieser Begrenzung erkennt der Server, wenn Sie ein Kennwort definieren, das länger als acht Zeichen ist, entweder das vollständige Kennwort oder nur die ersten acht Zeichen des Kennworts als gültige Zeichen.
2. Beim Benutzernamen und Kennwort des Administrators wird zwischen Groß- und Kleinschreibung auch dann unterschieden, wenn das Betriebssystem diese Unterscheidung nicht macht. Geben Sie Benutzernamen und Kennwort mit dem Befehl `htadm` exakt ein, wenn Sie auf die Konfigurations- und Verwaltungsformulare zugreifen.

-check *Kennwortdatei* **[***Benutzername* **[***Kennwort***]]**
Prüft das Kennwort für einen Benutzernamen, der bereits in der Kennwortdatei definiert ist, und gibt die Information aus, ob das Kennwort korrekt ist oder nicht. Wenn Sie den Befehl nur mit dem Parameter *Kennwortdatei* eingeben, werden Sie zur Eingabe der anderen Parameter aufgefordert.

Kennwortdatei
Pfad und Name der Kennwortdatei, die den Benutzernamen enthält, dessen Kennwort geprüft werden soll.

Benutzername
Der Benutzername, dessen Kennwort geprüft werden soll. Der Befehl schlägt fehl, wenn der Benutzername, den Sie angeben, nicht in der Kennwortdatei vorhanden ist.

Kennwort
Das Kennwort, das geprüft werden soll. Wenn das Kennwort, das Sie eingeben, das Kennwort ist, das für den Benutzernamen definiert ist, schreibt der Befehl `Correct` in die Standardausgabe und gibt am Ende der Prüfung den Rückkehrcode `0` aus. Wenn das Kennwort, das Sie eingeben, nicht das Kennwort ist, das für den Benutzernamen definiert ist, schreibt der Befehl `Incorrect` in die Standardausgabe.

-create *Kennwortdatei*
Erstellt eine leere Kennwortdatei.

Kennwortdatei
Pfad und Name der Kennwortdatei, die erstellt werden soll.

Beispiele

- Zum Hinzufügen eines Benutzers in eine Kennwortdatei:
  - Linux- und UNIX-Systeme:

    ```
    htadm -adduser /opt/ibm/edge/cp/server_root/protect/heroes.pwd
    clark superman "Clark Kent"
    ```

  - Windows-Systeme:

    ```
    htadm -adduser "C:\Programme\IBM\edge\cp\server_root\protect\
         heroes.pwd" clark superman "Clark Kent"
    ```

    Anmerkung: Der Befehl **htadm** muss in einer Zeile eingegeben sein. Er wird hier nur zur besseren Lesbarkeit in zwei Zeilen angezeigt. Am realen System würde der Befehl in einer Zeile mit mindestens einem Leerzeichen zwischen `clark` und `superman` eingegeben werden.
- Zum Löschen eines Benutzers aus der Kennwortdatei:
  - Linux- und UNIX-Systeme:

    ```
    htadm -deluser /opt/ibm/edge/cp/server_root/protect/
         heroes.pwd clark superman "Clark Kent"
    ```

  - Windows-Systeme:

    ```
    htadm -deluser "C:\Programme\IBM\edge\cp\server_root\protect\
         heroes.pwd" clark superman "Clark Kent"
    ```

Befehl htcformat

Zweck

Verwenden Sie den Befehl **htcformat**, um eine unformatierte Einheit oder eine Datei für die Speicherung des Proxy-Server-Cache vorzubereiten. Dieser Formatierungsbefehl muss verwendet werden, bevor die Einheit für den Proxy-Server-Cache angegeben wird.

Im Einheitenpfad muss die unformatierte Einheit angegeben sein. Nähere Informationen zum Zugriff auf unformatierte Einheiten finden Sie in der Dokumentation zu Ihrem Dateisystem. Beispiele sind in Teil 4, „Proxy-Server-Caching konfigurieren", auf Seite 75 enthalten.

Anmerkung: Linux-2.2-Kernel unterstützen kein Caching auf unformatierten Einheiten. Auf Linux-Plattformen können nur Dateien und Hauptspeicher für den Cache verwendet werden.

Die Mindestgröße für einen Caching-Proxy-Cache sind 16.392 KB. Dies entspricht 2.049 Blöcken.

Format

```
htcformat Einheit [-blocksize <Blockgröße>] [-blocks Anzahl Blöcke]
htcformat -file Dateipfad [-blocksize Blockgröße] -blocks Anzahl Blöcke
```

Parameter

-blocksize
Definiert die Blockgröße der Cache-Einheit. Die Blockgröße wird in Bytes angegeben. Der Standardwert ist 8.192 und sollte für alle Situationen verwendet werden.

-blocks
Anzahl der Blöcke, die auf der Einheit oder in der Datei erstellt werden sollen. Beim Formatieren einer Datei muss mit diesem Argument die Dateigröße angegeben werden. Dieses Argument kann auch verwendet werden, um die Größe einer bestimmten Einheit oder Partition einzuschränken, die für die Speicherung des Cache verwendet wird. Wenn das Argument -blocks nicht angegeben ist, werden so viele Blöcke erstellt, wie auf die Partition passen.

-file
Formatiert eine Datei anstelle einer Speichereinheit.

Verwendung

Das Caching-System teilt die Cache-Dateien oder -Einheiten zusätzlich in Container für Indexierung und Garbage-Collection auf. Für die Größe der Container wird eine bestimmte Anzahl von Blöcken festgelegt. Die Größe eines Containers kann nicht konfiguriert werden. Damit die Funktion Garbage-Collection ausgeführt werden kann, sind mindestens zwei Container erforderlich. Die Mindestgröße des Cache beträgt 16.392 KB.

Der Befehl **htcformat** weist Formatierungsanforderungen für Cache-Einheiten mit weniger als zwei Containern zurück.

Beispiele

Der folgende Beispielbefehl formatiert eine Plattenpartition mit dem Namen c0t0d0s0 unter Solaris.

```
htcformat /dev/rdsk/c0t0d0s0
```

Der folgende Beispielbefehl formatiert eine Plattenpartition mit dem Namen lv02 unter AIX.

```
htcformat /dev/rlv02
```

Der folgende Beispielbefehl formiert eine Plattenpartition mit dem Namen d: unter Windows.

```
htcformat \\.\d:
```

Der folgende Beispielbefehl erstellt die Datei filecache mit einer Größe von ca. 1 GB.

```
htcformat -file /opt/ibm/edge/cp/filecache -blocks 131072
```

Befehl ibmproxy

Zweck

Verwenden Sie den Befehl **ibmproxy**, um den Server zu starten.

Sie können alle Flags (außer **-r**) mit den Anweisungen in der Konfigurationsdatei des Servers definieren.

Es ist üblich, eine Readme-Datei mit Anweisungen und Hinweisen zu erstellen, die von allen Benutzern, die anfangen, mit dem Verzeichnis arbeiten, gelesen werden sollte. Standardmäßig bettet der Befehl **ibmproxy** alle Readme-Dateien in die Hypertextversion eines Verzeichnisses ein. Die Anweisungen für die Readme-Datei können auch mit der Konfigurationsanweisung DirReadme definiert werden.

Format

```
ibmproxy [-Flag [-Flag [-Flag..]]]
```

Parameter

-nobg
Führt den Server als Vordergrundprozess und nicht als Hintergrundprozess aus. Standardmäßig wird der Prozess im Hintergrund ausgeführt.

-nosnmp
Inaktiviert die SNMP-Unterstützung.

-p *Port-Nummer*
Ist auf dieser Port-Nummer empfangsbereit. Die Standard-Port-Nummer ist 80. Mit diesem Flag wird die Anweisung Port außer Kraft gesetzt, die in der Konfigurationsdatei angegeben ist. Wenn der Standardwert oder der in der Konfigurationsdatei angegebene Wert verwendet werden soll, lassen Sie dieses Flag weg.

-r *Konfigurationsdatei*
Gibt die Datei an, die als Konfigurationsdatei verwendet werden soll. Sie müssen dieses Flag verwenden, wenn der Server mit einer anderen Konfigurationsdatei als der Standardkonfigurationsdatei gestartet werden soll. Das Flag ermöglicht die Verwendung mehrerer Konfigurationsdateien.

-restart
Startet einen Server erneut, der derzeit ausgeführt wird. Der Befehl **ibmproxy** ruft die Prozessnummer des aktiven Servers aus der PID-Datei ab und sendet dem Prozess ein HUP-Signal (HangUP). Anschließend werden die Konfigurationsdateien des Servers erneut geladen und die Protokolldateien erneut geöffnet. Es dürfen nicht zwei Instanzen des Servers zur selben Zeit dieselbe PID-Datei, dieselben Protokolldateien und denselben Proxy-Cache verwenden. Dies würde zu einer Beschädigung der Daten führen.

Da der **http**-Dämon die vom Server derzeit verwendete Konfigurationsdatei lesen muss, um auf die PID-Datei zuzugreifen, müssen Sie beim Neustart dieselbe Konfigurationsdatei angeben. Wenn Sie beim Start des Servers das Flag **-r** und eine bestimmte Konfigurationsdatei verwendet haben, müssen Sie dieses Flag und diese Datei mit **-restart** angeben.

-snmp
Aktiviert die SNMP-Unterstützung.

-unload
Unter AIX wird hiermit die transparente Kernel-Erweiterung des Proxy-Servers entladen. Unter Linux werden hiermit die zugehörigen Firewall-Regeln entfernt.

Optionen für die Signalverarbeitung sind ebenfalls nur auf Linux- und UNIX-Plattformen verfügbar. Auf Linux- und UNIX-Plattformen stehen folgende Optionen zur Verfügung.

SIGTERM
Der Befehl **ibmproxy** wird nach Abschluss der aktuellen Verarbeitung beendet. Mit SIGKILL oder CANCEL können Sie die Verarbeitung sofort beenden.

SIGHUP
Falls der Befehl **ibmproxy** gerade ausgeführt wird, wird er erneut gestartet, und die Konfigurationsdatei wird erneut geladen. Danach wird die Verarbeitung fortgesetzt.

Beispiele

- Geben Sie Folgendes ein, wenn Sie den Server an Port 8080 mit der Konfigurationsdatei /usr/etc/ibmproxy.conf anstelle der Standardkonfigurationsdatei /etc/ibmproxy.conf starten möchten:

  ```
  ibmproxy -p 8080 -r /usr/etc/ibmproxy.conf
  ```

- Unter AIX müssen Sie Folgendes eingeben, damit ein Server mit der Standardkonfigurationsdatei unter Verwendung des System Resource Controller gestartet wird:

  ```
  startsrc -s ibmproxy
  ```

  Falls die Standardkonfigurationsdatei nicht vorhanden ist, exportiert der Befehl **ibmproxy** die Verzeichnisstruktur /Public. Diese Verzeichnisstruktur kann Softlinks zu anderen Verzeichnisstrukturen enthalten.

Anhang B. Anweisungen in der Konfigurationsdatei

In diesem Anhang werden die Anweisungen beschrieben, die in der Konfigurationsdatei ibmproxy.conf enthalten sind.

- **Auf Linux- und UNIX-Systemen:** Diese Anweisungen befinden sich in der Konfigurationsdatei ibmproxy.conf im Verzeichnis /etc/.
- **Auf Windows-Systemen:** Diese Anweisungen befinden sich normalerweise im Verzeichnis C:\Programme\IBM\edge\cp\.

Verwenden Sie diese Informationen als Referenz, wenn Sie den Server durch Editieren der Datei ibmproxy.conf konfigurieren. Falls Sie die Konfigurations- und Verwaltungsformulare verwenden, können Sie dieses Kapitel ignorieren.

Die Anweisungen sind in alphabetischer Reihenfolge aufgeführt.

Anweisungen, die bei einem Neustart nicht geändert werden

Einige Anweisungen werden bei einem Neustart des Servers nicht aktualisiert. Wenn Sie die folgenden Anweisungen ändern, während der Server aktiv ist, müssen Sie den Server manuell stoppen und anschließend erneut starten. (Nähere Informationen hierzu finden Sie in Kapitel 5, „Caching Proxy starten und stoppen", auf Seite 15.)

Tabelle 6. Anweisungen, die bei einem Neustart nicht aktualisiert werden

Anweisungsgruppe	Anweisungen
CGI	DisinheritEnv, InheritEnv
Caching	Caching
Protokollierung	AccessLog, CacheAccessLog, ErrorLog, ProxyAccessLog, ServerRoot
Netzzugriff	BindSpecific, Hostname, ListenBacklog, Port
Leistung	MaxActiveThreads
RTSP	Alle RTSP-Anweisungen
SSL	Alle SSL-Anweisungen
Prozesssteuerung unter Linux und UNIX	GroupId, UserId
Verschiedenes	TransparentProxy

Übersicht über die Anweisungen

Dieser Anhang enthält zu jeder Anweisung folgende Informationen:

- Eine Überschrift mit dem Namen und einer Kurzbeschreibung der Anweisung
- Angaben zur Verwendung der Anweisung
- Das Format der Anweisung, das der folgenden allgemeinen Syntax folgt:

  ```
  Anweisungsname Wert
  ```

- Sofern angebracht, ein Beispiel für die mögliche Einstellung der Anweisung in der Konfigurationsdatei

Anmerkung: Beispiele für Anweisungen mit Windows-spezifischen Pfaden enthalten manchmal die Angabe *server_root*. Diese Angabe steht stellvertretend für das Stammverzeichnis des bei der Installation ausgewählten Servers.

- Der Standardwert bzw. die Standardwerte der Anweisung

  Dies sind die in der Standardkonfigurationsdatei ursprünglich codierten Werte. Ändern Sie nur die Teile der Konfigurationsdatei, für die Sie von den Standardeinstellungen abgeweichende Werte verwenden möchten. Eine Anweisung, für die kein Standardwert definiert ist, ist in der Datei auf Kommentar gesetzt (# ist das erste Zeichen in der entsprechenden Zeile). Wenn Sie für eine solche Anweisung einen Wert festlegen möchten, entfernen Sie das Kommentarzeichen und fügen Sie den Wert zu dieser Zeile der Konfigurationsdatei hinzu.

Gültige Werte

Die folgende Liste enthält Werte, die in der Konfigurationsdatei angegeben werden können:

- In den Referenzinformationen zu einigen Anweisungen enthält der Teil *Wert* Schablonen für Anforderungen, Pfadnamen oder Hostnamen. Sofern nicht anders angegeben, kann in Schablonen ein Stern (*) verwendet werden. Der Stern dient als Platzhalter für eine beliebige Zeichenfolge oder ein beliebiges Zeichen.
- Gültige Werte für Konfigurationsanweisungen, die bejaht werden können, sind:
  - `Yes`
  - `On`
  - `OK`
  - `Enable`
- Gültige Werte für Konfigurationsanweisungen, die verneint werden können, sind:
  - `No`
  - `Off`
  - `None`
  - `Disable`
- Gültige Angaben für Konfigurationsanweisungen, mit denen eine Zeit angegeben werden kann, sind beliebige Kombinationen von:
  - *hh* - Stunden
  - *hh*:*mm* - Stunden und Minuten
  - *hh*:*mm*:*ss* - Stunden, Minuten und Sekunden
  - *n* `years` - Anzahl der Jahre mit 365 Tagen
  - *n* `months` - Anzahl der Monate mit 30 Tagen
  - *n* `weeks` - Anzahl der Wochen mit 7 Tagen
  - *n* `days` - Anzahl der Tage mit 24 Stunden
  - *n* `hours` - Anzahl der Stunden mit 60 Minuten
  - *n* `minutes` - Anzahl der Minuten mit 60 Sekunden
  - *n* `seconds` - Anzahl der Sekunden
  - *n* `fortnights` - Anzahl der 14-tägigen Intervalle

  Alle Einträge werden in Sekunden umgewandelt und addiert.
- Leerzeichen in einem in der Konfigurationsdatei angegebenen Dateinamen sind nicht zulässig. Leerzeichen werden als Begrenzer behandelt.

Syntax der Datensätze in der Konfigurationsdatei

Beachten Sie beim Editieren der Konfigurationsdatei Folgendes:

- Jede Anweisung muss in einer neuen Zeile beginnen.
- Werte werden durch ein oder mehrere Leerzeichen getrennt. Zwischen Leerzeichen und Tabulatorzeichen wird nicht unterschieden.
- Der Anfang eines Kommentars wird durch ein Nummernzeichen (#) angezeigt. Alle Zeichen ab dem Symbol # bis zum Ende der Zeile werden ignoriert.
- Falls mit einer Anweisung ein Nummernzeichen oder ein Leerzeichen angegeben werden muss, geben Sie vor diesem Zeichen einen Backslash (\) als Escape-Zeichen an. Ein Escape-Zeichen zeigt an, dass das nachfolgende Zeichen als Zeichen und nicht als Befehl interpretiert werden muss. Wird beispielsweise das Zeichen \# in einer Zeile gefunden, interpretiert der Server dies als Symbol # und nicht als den Anfang eines Kommentars und liest die darauf folgenden Zeichen. Wird das Zeichen \ in einer Zeile gefunden, interpretiert der Server dies als Leerzeichen, nicht als Wertbegrenzer und liest die darauf folgenden Zeichen, um den Wert zu bilden.

Anweisungen von Caching Proxy

Nachfolgend werden die Anweisungen von Caching Proxy beschrieben.

AcceptAnything - Alle Dateien bereitstellen

Mit dieser Anweisung können Sie festlegen, dass Dateien einem Client auch dann bereitgestellt werden, wenn der MIME-Typ der Datei nicht mit dem vom Client gesendeten Header `ACCEPT:` übereinstimmt. Wenn diese Anweisung auf `OFF` gesetzt ist, werden Dateien, deren MIME-Typ nicht mit den vom Client akzeptierten Typen übereinstimmt, nicht angezeigt. Stattdessen wird eine Fehlerseite angezeigt.

Format

```
AcceptAnything  {off | on}
```

Beispiel

```
AcceptAnything off
```

Standardwert

```
AcceptAnything on
```

AccessLog - Pfad für die Zugriffsprotokolldatei angeben

Mit dieser Anweisung können Sie das Verzeichnis und die Datei angeben, in der der Server Zugriffsstatistiken protokollieren soll. Standardmäßig schreibt der Server einen Eintrag in das Protokoll, wenn ein Client eine Datenanforderung für auf dem lokalen Server gespeicherte Daten an den Server sendet. Normalerweise gehören zu diesen Einträgen ausschließlich Anforderungen des Konfigurationsclients und Zugriffsanforderungen, wenn die Caching-Proxy-Maschine als Ursprungsserver eingesetzt wird. Dieses Protokoll enthält keine Informationen zu Proxy- oder Cache-Zugriffen.

Mit der Anweisung NoLog können Sie Clients angeben, deren Anforderungen nicht protokolliert werden sollen. Eine Beschreibung der Anweisung NoLog finden Sie im Abschnitt „NoLog - Protokolleinträge für bestimmte Hosts oder Domänen, die mit einer Schablone übereinstimmen, unterdrücken” auf Seite 244.

Wenn der Server aktiv ist, startet er täglich um 00:00 Uhr eine neue Protokolldatei. Andernfalls wird eine neue Protokolldatei gestartet, wenn der Server zum ersten Mal am Tag gestartet wird. Beim Erstellen der Datei verwendet der Server den von Ihnen angegebenen Dateinamen und hängt ein Datumssuffix an. Das Datumssuffix hat das Format *TTMmmJJJJ*, wobei *Mmm* für die ersten drei Buchstaben des Monats, *TT* für den Tag des Monats und *JJJJ* für das Jahr stehen.

Anmerkung: Wenn Sie die Standardeinstellungen des Servers für Benutzer-ID, Gruppen-ID oder Protokollverzeichnispfade ändern, müssen Sie die neuen Verzeichnisse erstellen und die für diese Verzeichnisse erforderlichen Berechtigungen und Eigentumsrechte aktualisieren. Damit der Server Daten in ein benutzerdefiniertes Protokollverzeichnis schreiben kann, müssen Sie die Berechtigung für dieses Verzeichnis auf 755 setzen und als Eigner die benutzerdefinierte Benutzer-ID für den Server angeben. Wenn Sie beispielsweise die Benutzer-ID des Servers vom Standardwert in jdoe und das Standardprotokollverzeichnis in server_root/account ändern, muss das Verzeichnis server_root/account die Berechtigung 755 besitzen. Der Eigner muss jdoe sein.

Es empfiehlt sich, alte Protokolldateien zu löschen, weil sie erheblich viel Speicherplatz auf dem Festplattenlaufwerk belegen können.

Format

```
AccessLog  /Verzeichnispfad/Name_der_Protokolldatei
```

Beispiel

```
AccessLog /logs/accesslog
```

Standardwerte

- **Linux- und UNIX-Systeme:** `AccessLog /opt/ibm/edge/cp/server_root/logs/local`
- **Windows-Systeme:** `AccessLog Laufwerk:\Programme\IBM\edge\cp\logs\local`

AccessLogExcludeMethod - Protokolleinträge für Dateien oder Verzeichnisse unterdrücken, die mit einer bestimmten Methode angefordert werden

Mit dieser Anweisung können Sie verhindern, dass mit einer bestimmten Methode angeforderte Zugriffe auf Dateien oder Verzeichnisse protokolliert werden. Sie könnten beispielsweise festlegen, dass DELETE-Anforderungen für Dateien und Verzeichnisse nicht protokolliert werden.

Es können mehrere Vorkommen dieser Anweisung in der Konfigurationsdatei enthalten sein. Sie können mit einer Anweisung auch mehrere Methoden angeben, wenn Sie sie durch Leerzeichen voneinander trennen.

Format

```
AccessLogExcludeMethod Methode  [...]
```

Beispiele

```
AccessLogExcludeMethod GET
AccessLogExcludeMethod PUT
AccessLogExcludeMethod POST
AccessLogExcludeMethod DELETE
AccessLogExcludeMethod GET   PUT
```

Standardwert
Keiner. Der Server zeichnet unabhängig von der Methode alle Anforderungen für Dateien und Verzeichnisse im Zugriffsprotokoll auf.

AccessLogExcludeMimeType - Einträge für bestimmte MIME-Typen im Zugriffsprotokoll unterdrücken

Mit dieser Anweisungen können Sie verhindern, dass Zugriffsanforderungen für Verzeichnisse und Dateien eines bestimmten MIME-Typs im Proxy-Zugriffsprotokoll aufgezeichnet werden. (Beispiele für MIME-Typen sind text/html, image/gif und image/jpeg.) Sie könnten beispielsweise festlegen, dass keine Zugriffsanforderungen für Grafiken im GIF-Format protokolliert werden.

Es können mehrere Vorkommen dieser Anweisung in der Konfigurationsdatei enthalten sein. Sie können mit einer Anweisung auch mehrere MIME-Typen angeben, indem Sie sie durch mindestens ein Leerzeichen voneinander trennen.

Anmerkung: Diese Anweisung betrifft nur das Proxy-Zugriffsprotokoll. Protokolle, in denen diese zwischengespeicherten Objekte nach MIME-Typ aufgelistet werden, können nicht gefiltert werden. Für solche Protokolle müssen Sie die Anweisung AccessLogExcludeURL verwenden.

Format
```
AccessLogExcludeMimeType MIME-Typ  [...]
```

Beispiel
```
AccessLogExcludeMimeType image/gif
AccessLogExcludeMimeType text/html
AccessLogExcludeMimeType image/gif   text/html
```

Standardwert
Keiner. Der Server zeichnet unabängig vom MIME-Typ alle Anforderungen für Dateien und Verzeichnisse im Zugriffsprotokoll auf.

AccessLogExcludeReturnCode - Protokolleinträge für bestimmte Rückkehrcodes unterdrücken

Mit dieser Anweisung können Sie festlegen, dass Zugriffsanforderungen, denen ein Fehlercode aus einem bestimmten Bereich zugeordnet ist, nicht protokolliert werden. Diese Fehlercodenummern sind Statuscodes des Proxy-Servers. Es können keine einzelnen Codes angegeben werden. Wenn Sie beispielsweise 300 angeben, werden die Zugriffsanforderungen mit Umleitungsrückkehrcodes (301, 302, 303 und 304) von der Protokollierung ausgeschlossen.

Es können mehrere Vorkommen dieser Anweisung in der Konfigurationsdatei enthalten sein. Sie können mit einer Anweisung auch mehrere Rückkehrcodes angeben, indem Sie sie durch mindestens ein Leerzeichen voneinander trennen.

Format
```
AccessLogExcludeReturnCode Bereich
```

Beispiel
```
AccessLogExcludeReturnCode 300
```

Standardwert
Keiner. Der Server zeichnet unabängig vom Rückkehrcode alle Anforderungen im Zugriffsprotokoll auf.

AccessLogExcludeURL - Protokolleinträge für bestimmte Dateien oder Verzeichnisse unterdrücken

Mit dieser Anweisung können Sie festlegen, dass Zugriffsanforderungen für bestimmte Dateien oder Verzeichnisse, die einer bestimmten URL-Schablone entsprechen, nicht protokolliert werden. Sie könnten beispielsweise festlegen, dass Zugriffsanforderungen für GIF-Dateien oder Zugriffsanforderungen für eine bestimmte Datei oder ein bestimmtes Verzeichnis auf Ihrem Server nicht protokolliert werden.

Es können mehrere Vorkommen dieser Anweisung in der Konfigurationsdatei enthalten sein. Sie können mit einer Anweisung auch mehrere Einträge angeben, indem Sie sie durch mindestens ein Leerzeichen voneinander trennen.

Format

```
AccessLogExcludeURL  Datei_oder_Typ [...]
```

Beispiele

```
AccessLogExcludeURL *.gif
AccessLogExcludeURL /Freebies/*
AccessLogExcludeURL  *.gif   /Freebies/*
```

Standardwert

Keiner. Der Server protokolliert die Zugriffanforderungen für alle Dateien und Verzeichnisse.

AccessLogExcludeUserAgent - Protokolleinträge für bestimmte Browser unterdrücken

Mit dieser Anweisung können Sie festlegen, dass Zugriffsanforderungen von bestimmten Benutzeragenten (z. B. Internet Explorer 5.0) nicht protokolliert werden.

Es können mehrere Vorkommen dieser Anweisung in der Konfigurationsdatei enthalten sein. Sie können mit einer Anweisung auch mehrere Einträge angeben, indem Sie sie durch mindestens ein Leerzeichen voneinander trennen.

Format

```
AccessLogExcludeUserAgent Benutzeragent [...]
```

Beispiel

```
AccessLogExcludeUserAgent *Mozilla/2.0
AccessLogExcludeUserAgent *MSIE 5*
```

Standardwert

Standardmäßig enthält die Datei ibmproxy.conf die folgenden Definitionen für die Anweisung AccessLogExcludeUserAgent:

```
AccessLogExcludeUserAgent IBM_Network_Dispatcher_HTTP_Advisor
AccessLogExcludeUserAgent IBM_Network_Dispatcher_WTE_Advisor
```

Die aufgelisteten Benutzeragenten sind für bestimmte Load-Balancer-Advisor definiert, die dem Caching-Proxy-Server normalerweise vorgeschaltet sind. Anforderungen dieser Benutzeragenten werden standardmäßig nicht protokolliert, um die Schreibzugriffe auf das Protokoll zu verringern und dadurch einen höheren Durchsatz zu erzielen. Die Zugriffsanforderungen aller anderen Benutzeragenten werden standardmäßig vom Server protokolliert.

AddBlankIcon - Symbol-URL für die Ausrichtung von Überschriften in Verzeichnislisten angeben

Mit dieser Anweisung können Sie ein Symbol angeben, das zum Ausrichten der Überschriften in Verzeichnislisten verwendet werden soll, die zurückgegeben werden, wenn der Server als Proxy-Server für FTP-Anforderungen auftritt. Die Symbole erscheinen neben den zugehörigen Dateien, um dem Benutzer die Unterscheidung zwischen den Dateien zu erleichtern.

Sie können ein leeres oder jedes andere Symbol angeben. Für eine ordnungsgemäße Ausrichtung muss das verwendete Symbol dieselbe Größe besitzen wie die anderen Symbole in den Verzeichnislisten.

Format

`AddBlankIcon` *`Symbol-URL`* *`alternativer_Text`*

`Symbol-URL`

Gibt den letzten Teil des URL für das Symbol an. Der Server hängt diesen Wert an das Verzeichnis /icons/ an, um die vollständige URL-Anforderung zu bilden. Falls sich die Anforderung auf eine lokale Datei bezieht, übersetzt der Server sie gemäß den Zuordnungsanweisungen. Damit das Symbol abgerufen werden kann, müssen die Zuordnungsanweisungen die Übergabe der Anforderung zulassen.

Wenn Sie den Server als Proxy-Server verwenden, muss die vollständige Anforderung ein vollständig qualifizierter URL sein, der auf Ihren Server verweist.

`alternativer_Text`

Gibt den alternativen Text an, der anstelle des Symbols verwendet wird, falls der anfordernde Browser keine Grafiken anzeigt.

Beispiel

```
AddBlankIcon logo.gif Logo
```

Standardwerte

- **Linux- und UNIX:** `AddBlankIcon blank.m.pm.gif`
- **Windows:** `AddBlankIcon blank.gif`

Standardmäßig ist kein alternativer Text angegeben, weil das Symbol leer ist.

AddDirIcon - Symbol-URL für die Darstellung von Verzeichnissen in Verzeichnislisten angeben

Mit dieser Anweisung können Sie ein Symbol für die Darstellung eines Verzeichnisses in einer Verzeichnisliste angeben.

Format

`AddDirIcon` *`Symbol-URL`* *`alternativer_Text`*

`Symbol-URL`

Gibt den letzten Teil des URL für das Symbol an. Der Server hängt diesen Wert an das Verzeichnis /icons/ an, um die vollständige URL-Anforderung zu bilden. Falls sich die Anforderung auf eine lokale Datei bezieht, übersetzt der Server sie gemäß den Zuordnungsanweisungen. Damit das Symbol abgerufen werden kann, müssen die Zuordnungsanweisungen die Übergabe der Anforderung zulassen.

Wenn Sie den Server als Proxy-Server verwenden, muss die vollständige Anforderung ein vollständig qualifizierter URL sein, der auf Ihren Server verweist. Sie müssen den URL einer lokalen Datei zuordnen und sicherstellen, dass die Zuordnungsanweisungen die Übergabe des URL zulassen.

alternativer_Text
Gibt den alternativen Text an, der anstelle des Symbols verwendet wird, falls der anfordernde Browser keine Grafiken anzeigt.

Beispiel

```
AddDirIcon direct.gif DIR
```

Standardwerte

- **Linux und UNIX:** `AddDirIcon dir.m.pm.gif  DIR`
- **Windows:** `AddDirIcon dir.gif DIR`

AddEncoding - MIME-Inhaltscodierung für Dateien mit bestimmten Suffixen angeben

Mit dieser Anweisung können Sie Dateien mit einem bestimmten Suffix an einen MIME-Codierungstyp binden. Diese Anweisung wird selten verwendet.

Format

`AddEncoding` *.Erweiterung Codierung*

.Erweiterung
Das Muster für das Dateisuffix.

Codierung
Der MIME-Codierungstyp, den Sie an Dateien binden möchten, die dem angegebenen Suffixmuster entsprechen.

Beispiel

```
AddEncoding .qp quoted_printable
```

Standardwert

```
AddEncoding .Z x-compress
```

AddIcon - Symbol an einen MIME-Inhaltstyp oder -Codierungstyp binden

Mit dieser Anweisung können Sie Symbole für die Darstellung von Dateien mit einem bestimmten MIME-Inhaltstyp oder MIME-Codierungstyp angeben. Der Server verwendet diese Symbole in Verzeichnislisten, einschließlich FTP-Verzeichnislisten.

Format

`AddIcon` *Symbol-URL alternativer_Text Schablone_für_MIME-Typ*

Symbol-URL

Gibt den letzten Teil des URL für das Symbol an. Der Server hängt diesen Wert an das Verzeichnis /icons/ an, um die vollständige URL-Anforderung zu bilden. Falls sich die Anforderung auf eine lokale Datei bezieht, übersetzt der Server sie gemäß den Zuordnungsanweisungen. Damit das Symbol abgerufen werden kann, müssen die Zuordnungsanweisungen die Übergabe der Anforderung zulassen.

Wenn Sie den Server als Proxy-Server verwenden, muss die vollständige Anforderung ein vollständig qualifizierter URL sein, der auf Ihren Server verweist. Sie müssen den URL einer lokalen Datei zuordnen und sicherstellen, dass die Zuordnungsanweisungen die Übergabe des URL zulassen.

alternativer_Text
Gibt den alternativen Text an, der anstelle des Symbols verwendet wird, falls der anfordernde Browser keine Grafiken anzeigt.

Schablone_für_Typ
Gibt eine Schablone für einen MIME-Inhaltstyp oder -Codierungstyp an. Schablonen für den Inhaltstyp enthalten immer einen Schrägstrich (/). Schablonen für Codierungstypen enthalten niemals einen Schrägstrich.

Beispiel

```
AddIcon video_file.m.pm.gif MOV video/*
```

Standardwerte

In der Konfigurationsdatei ibmproxy.conf sind mehrere Standardwerte für die Anweisung AddIcon gesetzt.

AddParentIcon - Symbol-URL für die Darstellung eines Elternverzeichnisses in Verzeichnislisten angeben

Mit dieser Anweisung können Sie ein Symbol für die Darstellung eines Elternverzeichnisses in Verzeichnislisten angeben.

Format

```
AddParentIcon   Symbol-URL   alternativer_Text
```

Symbol-URL

Gibt den letzten Teil des URL für das Symbol an. Der Server hängt diesen Wert an das Verzeichnis /icons/ an, um die vollständige URL-Anforderung zu bilden. Falls sich die Anforderung auf eine lokale Datei bezieht, übersetzt der Server sie gemäß den Zuordnungsanweisungen. Damit das Symbol abgerufen werden kann, müssen die Zuordnungsanweisungen die Übergabe der Anforderung zulassen.

Wenn Sie den Server als Proxy-Server verwenden, muss die vollständige Anforderung ein vollständig qualifizierter URL sein, der auf Ihren Server verweist. Sie müssen den URL einer lokalen Datei zuordnen und sicherstellen, dass die Zuordnungsanweisungen die Übergabe des URL zulassen.

alternativer_Text
Gibt den alternativen Text an, der anstelle des Symbols verwendet wird, falls der anfordernde Browser keine Grafiken anzeigt.

Beispiel

```
AddParentIcon parent.gif UP
```

Standardwert

```
AddParentIcon   dir-up.gif    UP
```

AddType - Datentyp für Dateien mit bestimmten Suffixen angeben

Mit dieser Anweisung können Sie Dateien mit einem bestimmten Suffix an einen MIME-Typ oder -Subtyp binden. Es können mehrere Vorkommen dieser Anweisung in der Konfigurationsdatei enthalten sein. Für die gebräuchlichsten Suffixe stellt der Server Standardwerte bereit.

Format

`AddType` *`.Erweiterung Typ/Subtyp Codierung`* [*`Qualität`*[*`Zeichensatz`*]]

.Erweiterung
Das Muster für das Dateisuffix. Das Platzhalterzeichen (*) kann nur in den beiden folgenden speziellen Suffixmustern verwendet werden:

. Findet alle Dateinamen, die einen Punkt (.) enthalten und mit keiner anderen Regel übereinstimmen.

***** Findet alle Dateinamen, die keinen Punkt (.) enthalten und mit keiner anderen Regel übereinstimmen.

Typ/Subtyp
Der MIME-Typ und -Subtyp, den Sie an Dateien binden möchten, die dem angegebenen Suffixmuster entsprechen.

Codierung
Die MIME-Inhaltscodierung, die für die Konvertierung der Dateien verwendet wurde. Anhand der Codierung bestimmt ein FTP-Proxy-Server außerdem, ob die Datei im Binärmodus abgerufen werden soll. Die gebräuchlichsten Codierungen sind `7bit`, `8bit` und `binary`, die folgendermaßen bestimmt werden:

7bit Die Daten werden als kurze Zeilen (weniger als 1000 Zeichen) mit ASCII-Daten vom Typ 8859-1 dargestellt. Normalerweise fallen Quellcode- und Textdateien in diese Kategorie. Ausnahmen sind Dateien, die Grafikzeichen oder Zeichen mit Akzenten enthalten.

8bit Die Daten werden als kurze Zeilen dargestellt, können aber Zeichen der oberen Hälfte des Zeichensatzes enthalten (zum Beispiel Grafikzeichen oder Zeichen mit Akzent). Normalerweise fallen PostScript-Dateien und Textdateien aus dem europäischen Raum in diese Kategorie.

binary Diese Codierung kann für alle Datentypen verwendet werden. Die Daten können außer Nicht-ASCII-Zeichen auch lange Zeilen (mit mehr als 1000 Zeichen) enthalten. Fast jede Datei des Typs `image/*`, `audio/*` und `video/*` fällt in diese Kategorie, ebenso binäre Datendateien des Typs `application/*`.

Jeder andere Codierungswert wird als Binärtyp behandelt und in MIME-Headern als MIME-Header für Inhaltscodierung übergeben. Die Spezifikationen `7bit` und `8bit` werden nicht in MIME-Headern gesendet.

Qualität
Gibt einen optionalen Anzeiger für den relativen Wert (auf einer Skala von 0.0 bis 1.0) für den Inhaltstyp an. Der Qualitätswert wird verwendet, wenn einer Anforderung mehrere Darstellungen einer Datei entsprechen. Der Server wählt die Datei aus, die den höchsten Qualitätswert besitzt. Wenn beispielsweise die Datei internet.ps angefordert wird und für den Server die folgenden AddType-Anweisungen festgelegt sind, verwendet der Server die Zeile `application/postscript`, weil ihr Qualitätswert höher ist.

```
AddType .ps application/postscript 8bit 1.0
AddType *.*  application/binary       binary   0.3
```

Zeichensatz
Ein wahlfreier Anzeiger für den Zeichensatz, den Sie Textdateien zuordnen möchten. Für die Dateien, denen Sie einen Zeichensatz zuordnen, teilt der Server den Client-Browsern mit, welcher Zeichensatz zum Anzeigen der Datei verwendet werden muss. Wenn Sie für das Feld *Zeichensatz* einen Wert festlegen, müssen Sie auch einen Wert für *Qualität* angeben.

Beispiel

```
AddType .bin application/octet-stream binary 0.8
```

Standardwerte

In der Konfigurationsdatei ibmproxy.conf sind mehrere Standardwerte für die Anweisung AddType gesetzt.

AddUnknownIcon - Symbol-URL für die Darstellung unbekannter Dateitypen in Verzeichnislisten angeben

Mit dieser Anweisung können Sie ein Symbol für die Darstellung von Dateien mit unbekannten Dateitypen in der Verzeichnisliste angeben.

Format

`AddUnknownIcon` *Symbol-URL* *alternativer_Text*

Symbol-URL

Gibt den letzten Teil des URL für das Symbol an. Der Server hängt diesen Wert an `/icons/` an, um die vollständige URL-Anforderung zu bilden. Falls sich die Anforderung auf eine lokale Datei bezieht, übersetzt der Server sie gemäß den Zuordnungsanweisungen. Damit das Symbol abgerufen werden kann, müssen die Zuordnungsanweisungen die Übergabe der Anforderung zulassen.

Wenn Sie den Server als Proxy-Server verwenden, muss die vollständige Anforderung ein vollständig qualifizierter URL sein, der auf Ihren Server verweist. Sie müssen den URL einer lokalen Datei zuordnen und sicherstellen, dass die Zuordnungsanweisungen die Übergabe des URL zulassen.

alternativer_Text
Gibt den alternativen Text an, der anstelle des Symbols verwendet wird, falls der anfordernde Browser keine Grafiken anzeigt.

Beispiel

```
AddUnknownIcon saywhat.gif  unknown
```

Standardwerte

- **Linux und UNIX:** `AddUnknownIcon  unknown.gif   ???`
- **Windows:** `AddUnknownIcon  unknown.gif   ???`

AdminPort - Port für die Anforderung von Verwaltungsseiten oder -formularen angeben

Mit dieser Anweisung können Sie einen Port angeben, den Administratoren für den Zugriff auf Statusseiten oder Konfigurationsformulare des Servers verwenden können. Anforderungen an diesen Port werden nicht zusammen mit den anderen eingehenden Anforderungen am Standard-Port oder an Ports, die mit der Anweisung Port definiert wurden, in einer Warteschlange gespeichert. Auf die Anforderungen an den Verwaltungs-Port werden jedoch dieselben Zugriffssteuerungs- und Anforderungszuordnungsregeln (z. B. Pass, Exec, Protect) angewendet.

Anmerkung: Der Verwaltungs-Port darf *nicht* mit dem Standard-Port oder den mit der Anweisung Port definierten Ports identisch sein.

Format
`AdminPort` *`Port-Nummer`*

Beispiel
```
AdminPort 2001
```

Standardwert
Keiner.

AggressiveCaching - Caching für Dateien festlegen, die als nicht zwischenspeicherbar gekennzeichnet sind

Mit dieser Anweisung können Sie festlegen, ob vom Ursprungsserver zurückgegebene Dateien, die als nicht zwischenspeicherbar gekennzeichnet sind, trotzdem im Cache gespeichert werden sollen. Dateien, die als nicht zwischenspeicherbar gekennzeichnet sind und von dieser Anweisung erfasst werden, werden mit der Angabe `must revalidate` (müssen überprüft werden) gekennzeichnet. Jedesmal, wenn die Datei angefordert wird, sendet der Proxy-Server eine Anforderung `If-Modified-Since` (Anfrage, ob die Datei in der Zwischenzeit geändert wurde) an den Ursprungsserver, damit die Antwort überprüft wird, bevor sie aus dem Cache bereitgestellt wird. Derzeit sind die einzigen durch diese Anweisung betroffenen Dateien die vom Ursprungsserver zurückgegebenen Antworten, die den Cache Header `cache-control: no-cache` enthalten. Sie können mehrere dieser Anweisungen definieren.

Format
`AggressiveCaching` *`URL-Muster`*

Beispiele
```
AggressiveCaching http://www.hosta.com/*
AggressiveCaching http://www.hostb.com/*
```

Aus Gründen der Abwärtskompatibilität wird die frühere Syntax dieser Anweisung (`AggressiveCaching {on | off}`) jetzt wie folgt behandelt:

`AggressiveCaching on` wird behandelt als `AggressiveCaching * .`

`AggressiveCaching off` wird ignoriert.

Anmerkung: Wenn `AggressiveCaching off` und `AggressiveCaching` *`URL-Muster`* angegeben sind, wird `AggressiveCaching off` ignoriert, und es wird eine Warnung angezeigt.

Standardwert
Keiner.

AlwaysWelcome - Festlegen, ob das angeforderte Verzeichnis nach Begrüßungsdateien durchsucht werden soll

Bei Anforderungen, die einen Verzeichnisnamen, aber keinen Dateinamen enthalten, steuert die Anweisung AlwaysWelcome, ob der Server in diesem Verzeichnis nach einer Begrüßungsdatei sucht. Standardmäßig hat AlwaysWelcome den Wert `on`, d. h. der Server sucht immer im angeforderten Verzeichnis nach einer Datei, deren Name mit dem in einer Welcome-Anweisung angegebenen Namen übereinstimmt. Wenn der Server eine Übereinstimmung findet, gibt er die Datei an den

Anfordernden zurück. Falls der Server in einem Verzeichnis mehrere Dateien findet, die den in Welcome-Anweisungen angegebenen Dateinamen entsprechen, bestimmt die Reihenfolge der Welcome-Anweisungen, welche Datei zurückgegeben wird. Der Server verwendet die Welcome-Anweisung verwendet, die in der Konfigurationsdatei zuerst angegeben ist.

Format

```
AlwaysWelcome  on | off
```

Standardwert

```
AlwaysWelcome  on
```

Zugehörige Anweisungen

- „Welcome - Die Namen von Begrüßungsdateien angeben" auf Seite 287

appendCRLFtoPost - CRLF in POST-Anforderungen einfügen

Mit dieser Anweisung können Sie URLs angeben, für die Caching Proxy die Zeichen für Zeilenschaltung und Zeilenvorschub (CRLF, Carriage Return Line Feed) am Ende des Hauptteils einer POST-Anforderung einfügen soll. Sie können mehrere dieser Anweisungen definieren.

Anmerkung: Definieren Sie diese Anweisung nur für URLs, bei denen während der Verarbeitung von POST-Anforderungen ein Problem auftritt.

Format

```
appendCRLFtoPost  URL-Muster
```

Beispiel

```
appendCRLFtoPost http://www.hosta.com/
```

Standardwert

Keiner.

ArrayName - Name des fernen Cache-Bereichs

Mit dieser Anweisung können Sie den fernen Cache-Bereich konfigurieren, der von den Servern gemeinsam genutzt werden soll.

Anmerkung: Beim Einrichten eines Bereichs muss die Anweisung Hostname für alle Member des Bereichs gleich konfiguriert werden.

Format

```
ArrayName Bereichsname
```

Standardwert

Keiner.

Authentication - Schritt "Authentication" anpassen

Mit dieser Anweisung können Sie eine angepasste Anwendungsfunktion angeben, die der Server während der Bearbeitung einer Anforderung im Schritt "Authentication" aufrufen soll. Dieser Code wird entsprechend dem Authentifizierungsschema ausgeführt. Es wird nur die `Basisauthentifizierung` unterstützt.

Anmerkung: Die Authentifizierung ist Teil des Autorisierungsprozesses. Sie wird nur dann durchgeführt, wenn eine Autorisierung erforderlich ist.

Format

Authentication *Schema /Pfad/Datei:Funktionsname*

Schema
Gibt ein Authentifizierungsschema an, das genauer bestimmt, ob die Anwendungsfunktion aufgerufen wird. Gültige Werte sind der Stern (*) und BASIC.

/Pfad/Datei
Der vollständig qualifizierte Dateiname des kompilierten Programms einschließlich der Erweiterung.

Funktionsname
Der Name der Anwendungsfunktion in Ihrem Programm.

Beispiel

```
Authentication BASIC /ics/api/bin/icsextpgm.so:basic_authentication
```

Standardwert

Keiner.

Authorization - Schritt "Authorization" anpassen

Mit dieser Anweisung können Sie eine angepasste Anwendungsfunktion angeben, die der Server während der Bearbeitung einer Anforderung im Schritt "Authorization" aufrufen soll. Dieser Code prüft, ob das angeforderte Objekt dem Client bereitgestellt werden kann.

Format

Authorization *Anforderungsschablone /Pfad/Datei:Funktionsname*

Anforderungsschablone
Eine Schablone für Anforderungen, die genauer bestimmt, ob eine Anwendungsfunktion aufgerufen wird. Die Spezifikation kann das Protokoll, die Domäne und den Host, einen vorangestellten Schrägstrich (/) und einen Stern (*) als Platzhalterzeichen enthalten. Beispiele für gültige Schablonen sind /front_page.html, http://www.ics.raleigh.ibm.com, /pub*, /* und *. Die Anforderungsschablone muss mit dem Dokumentstammverzeichnis (/) beginnen, wenn Caching Proxy als Reverse Proxy eingesetzt wird.

/Pfad/Datei
Der vollständig qualifizierte Dateiname des kompilierten Programms einschließlich der Erweiterung.

Funktionsname
Der Name der Anwendungsfunktionen in Ihrem Programm.

Beispiel

```
Authorization /index.html /api/bin/icsextpgm.so:auth_url
```

Standardwert

Keiner.

AutoCacheRefresh - Angeben, ob Cache-Aktualisierung verwendet werden soll

Mit dieser Anweisung können Sie die Cache-Aktualisierung aktivieren und inaktivieren. Wenn die Cache-Aktualisierung aktiviert ist, wird der Cache-Inhalt automatisch aktualisiert. Ist die Cache-Aktualisierung inaktiviert, wird der Cache-Agent nicht aufgerufen, und alle Einstellungen werden ignoriert. Wenn Sie den Cache-

Agenten mit einer anderen Methode starten, z. B. mit einem **cron**-Job auf Linux- und UNIX-Systemen, setzen Sie diese Anweisung auf `off` (Aktualisierung inaktivieren).

Format

```
AutoCacheRefresh {on | off}
```

Standardwert

```
AutoCacheRefresh On
```

BindSpecific - Angeben, ob der Server an eine oder an alle IP-Adressen gebunden wird

Mit dieser Anweisung können Sie auf einem System mit mehreren Schnittstellenadressen (Multihomed) angeben, ob der Server eine einzige Netzadresse überwachen soll. Wenn Sie diese Anweisung auf `On` setzen, wird der Server an die mit der Anweisung Hostname angegebene IP-Adresse anstatt an alle lokalen IP-Adressen gebunden.

Falls Sie diese Anweisung nicht angeben, wird der Server an den Standardhostnamen gebunden.

Sollten Sie diese Anweisung ändern, müssen Sie den Server manuell stoppen und anschließend erneut starten. Der Server führt die Änderungen erst dann aus, wenn Sie ihn erneut starten. (Nähere Informationen hierzu finden Sie in Kapitel 5, „Caching Proxy starten und stoppen", auf Seite 15.)

Format

```
BindSpecific {on | off}  [OutgoingSrcIp IP-Adresse | Hostname]
```

[OutgoingSrcIp *IP-Adresse* | *Hostname***]**
Mit der Option OutgoingSrcIp können Sie festlegen, dass Caching Proxy eine bestimmte Quellen-IP-Adresse für abgehende Verbindungen verwenden soll. Dies ist für die Caching-Proxy-Einstellungen in DMZ und für spezielle Firewall-Regeln hilfreich.

Standardwert

```
BindSpecific Off
```

BlockSize - Größe der Blöcke im Cache festlegen

Mit dieser Anweisung wird die Größe (in Byte) der Blöcke auf dem Datenträger der Cache-Einheit festgelegt. Der Standardert ist 8192. Ändern Sie diesen Wert nicht, da dies die einzige unterstützte Größe ist. Nähere Informationen finden Sie im Abschnitt „Befehl htcformat" auf Seite 170.

Format

```
BlockSize Größe
```

Standardwert

Standardmäßig ist in der Konfigurationsdatei keine Einstellung für BlockSize enthalten. (Der Standardwert ist 8192.)

CacheAccessLog - Pfad für Cache-Zugriffsprotokolldateien angeben

Mit dieser Anweisung können Sie den Pfad und die Datei angeben, in der der Server ein Protokoll zu den Zugriffen auf den Proxy-Cache speichern soll. Diese

Anweisung ist nur zulässig, wenn der Server als Proxy-Server ausgeführt wird. Nähere Informationen finden Sie im Abschnitt „CacheRefreshTime - Startzeitpunkt für Cache-Agenten angeben” auf Seite 198.

Wenn Sie die Protokollierung von Anforderungen an den Proxy-Cache aktivieren möchten, müssen Sie die Anweisung Caching auf `ON` setzen und Wert für die Anweisungen CacheMemory und CacheAccessLog definieren. Es können wahlweise eine oder mehrere Cache-Einheiten mit der Anweisung CacheDev definiert werden.

Der Wert von CacheAccessLog kann ein absoluter Pfad oder ein Pfad relativ zum Serverstammverzeichnis (ServerRoot) sein. (Für jeden Pfad wird ein Beispiel gezeigt.)

Format

```
CacheAccessLog  Pfad/Datei
```

Beispiele

```
CacheAccessLog /absolute/path/logfile
CacheAccessLog /logs/logfile
```

Standardwerte

- **Linux- und UNIX-Systeme:** `CacheAccessLog /opt/ibm/edge/cp/server_root/logs/cache`
- **Windows-Systeme:** `CacheAccessLog Laufwerk:\Programme\IBM\edge\cp\logs\cache`

CacheAlgorithm - Den Cache-Algorithmus angeben

Mit dieser Anweisung können Sie den Cache-Algorithmus angeben, den der Server für die Garbage-Collection verwenden soll.

Format

```
CacheAlgorithm  {bandwidth | responsetime | blend}
```

bandwidth
Versucht, so viel Netzbandbreite wie möglich einzusparen.

responsetime
Versucht, die Antwortzeit für den Benutzer zu minimieren.

blend
Verwendet eine ausgewogene Kombination von `bandwidth` und `responsetime`.

Standardwert

```
CacheAlgorithm bandwidth
```

CacheByIncomingUrl - Basis für die Generierung von Dateinamen im Cache angeben

Mit dieser Anweisung können Sie festlegen, ob der eingehende URL der Anforderung als Basis für die generierten Namen der Dateien im Cache verwendet werden soll.

Wenn Sie diese Anweisung auf `on` setzen, werden die Namen für die Dateien im Cache auf der Basis des eingehenden URL generiert. Wenn Sie die Anweisung auf `off` setzen, wird der eingehende URL zuerst an alle gültigen Plug-ins für Namens-

übersetzung, MAP-Regeln und PROXY-Regeln übergeben. Der daraufhin generierte URL wird dann als Basis für die Namen der Dateien im Cache verwendet.

Anmerkung: Verwenden Sie für die Definition von Cache-Filtern in einem Szenario mit Reverse Proxy für URL-basierte Cache-Filter ein Format, das mit dem Dokumentstammverzeichnis / (Schrägstrich) beginnt. Beispiel: `/test/index.html`. Das Format darf *kein* Protokoll enthalten, wie z. B. *http://*.

Format

```
CacheByIncomingUrl {on | off}
```

Standardwert

```
CacheByIncomingURL off
```

CacheClean - Zeitlimit für Dateien im Cache angeben

Mit dieser Anweisung können Sie angeben, wie lange der Server Dateien im Cache halten soll. Wenn die Garbage-Collection ausgeführt wird, löscht der Server die Dateien im Cache, die dieses Zeitlimit überschreiten, ungeachtet ihres Verfallsdatums. Wird eine Datei angefordert, die das mit dieser Anweisung festgelegte Zeitlimit überschritten hat, überprüft der Server die Datei, um sicherzustellen, dass sie gültig ist, bevor er sie zurückgibt.

Format

```
CacheClean  Zeitangabe
```

Beispiel

```
CacheClean 2 weeks
```

Standardwert

```
CacheClean 1 month
```

CacheDefaultExpiry - Standardverfallszeit für Dateien angeben

Mit dieser Anweisung können Sie eine Standardverfallszeit für Dateien festlegen, für die der Server keinen Header des Typs `Expires` oder `Last-Modified` bereitstellt. Geben Sie eine URL-Schablone und die Verfallszeit für die Dateien an, deren URLs mit der Schablone übereinstimmen. Es können mehrere Vorkommen dieser Anweisung in der Konfigurationsdatei enthalten sein. Fügen Sie für jede Schablone eine eigene Anweisung ein. Die URL-Schablone muss das Protokoll enthalten. Geben Sie die Zeit in einer beliebigen Kombination aus Monaten, Wochen, Tagen und Stunden an.

Format

```
CacheDefaultExpiry URL-Schablone Verfallszeit
```

Standardwerte

```
CacheDefaultExpiry ftp:*  1 day
CacheDefaultExpiry gopher:*  2 days
CacheDefaultExpiry http:*  0 days
```

Anmerkung: Die Standardverfallszeit für das Protokoll HTTP sind 0 Tage (`0 days`). Es empfiehlt sich, diesen Wert zu übernehmen, weil viele Script-Programme kein Verfallsdatum angeben, obwohl ihre Ausgabe sofort verfällt. Ein anderer Wert als `0` kann bewirken, dass den Clients veraltete Inhalte angezeigt werden.

CacheDev - Speichereinheit für den Cache angeben

Mit dieser Anweisung können Sie eine Speichereinheit für den Cache angeben. Sie können eine Datei oder eine unformatierte Plattenpartition angeben. Auf AIX-Plattformen kann ein logischer Datenträger ohne Dateisystem angegeben werden. (Wenn kein Hauptspeicher-Cache verwendet wird, erzielen Sie durch die Verwendung eines Datenträgers ohne Dateisystem für den Cache die beste Leistung.)

Cache-Einheiten müssen vorbereitet werden, bevor sie als solche festgelegt werden. Verwenden Sie den Befehl **htcformat**, um eine Cache-Einheit vorzubereiten. Nähere Informationen hierzu finden Sie im Abschnitt „Befehl htcformat" auf Seite 170.

Es können mehrere Cache-Einheiten angegeben werden. Jeder Einheit werden dieselben Werte für CacheMemory und BlockSize zugewiesen. Jedoch belegt jede Cache-Einheit auf der Proxy-Server-Maschine einen Speicher von etwa 8 MB. Weniger, aber dafür größere Einheiten sind effektiver als eine größere Anzahl kleinerer Einheiten. Die beste Effizienz wird bei der Verwendung eines vollständigen Datenträgers als eine große Partition erreicht, die für keinen anderen Zweck vorgesehen ist. Nähere Informationen zum Cache-Speicher finden Sie im Abschnitt „Die Leistung des Platten-Cache optimieren" auf Seite 109.

Format

```
CacheDev  {unformatierte_Plattenpartition | Datei}
```

Beispiele

AIX: `CacheDev /dev/rlv02`

HP-UX: `CacheDev /dev/rdsk/c1t15d0`

Linux: `CacheDev /opt/IBMWTE/filecache1`

Solaris: `CacheDev /dev/rdsk/c1t3d0s0`

Windows: `CacheDev \\.\E:`

Standardwert

Keiner.

CacheExpiryCheck - Angeben, ob der Server verfallene Dateien zurückgeben soll

Mit dieser Anweisung können Sie angeben, ob der Server Dateien im Cache, deren Verfallszeit überschritten ist, zurückgeben soll. Setzen Sie den Wert auf `Off`, wenn der Server auch verfallene Dateien zurückgeben soll. Verwenden Sie den Standardwert `On`, falls der Proxy-Server bei Anforderung einer verfallenen Datei durch einen Client auf dem Ursprungsserver nach einer neueren Version der Datei suchen soll. Im Allgemeinen lehnen Administratoren die Rückgabe verfallener Dateien ab und machen nur für Demonstrationszwecke eine Ausnahme, wenn der zurückgegebene Inhalt nicht von Bedeutung ist.

Format

```
CacheExpiryCheck {on | off}
```

Standardwert

```
CacheExpiryCheck On
```

CacheFileSizeLimit - Maximale Größe für Dateien im Cache angeben

Mit dieser Anweisung können Sie die maximale Größe von Dateien im Cache angeben. Dateien, deren Größe diesen Wert überschreitet, werden nicht in den Cache gestellt. Der Wert kann in Byte (B), Kilobyte (K), Megabyte (M) oder Gigabyte (G) angegeben werden. Zwischen Zahl und Maßeinheit (B, K, M, G) kann ein Leerzeichen eingefügt werden.

Format

```
CacheFileSizeLimit Maximum {B | K | M | G}
```

Standardwert

```
CacheFileSizeLimit 4000 K
```

CacheLastModifiedFactor - Wert zur Berechnung der Verfallsdaten angeben

Mit dieser Anweisung können Sie den Wert angeben, der zur Berechnung der Verfallsdaten für bestimmte URLs oder für alle URLs, die mit einer Schablone übereinstimmen, verwendet werden soll.

HTTP-Server liefern für eine Datei anstelle eines Verfallsdatums häufig die Zeit der letzten Änderung (last-modified). In ähnlicher Weise können FTP-Dateien anstelle eines Verfallsdatums eine Zeitmarke für die letzte Änderung besitzen. Caching Proxy berechnet für diese Dateien auf der Basis der Zeit der letzten Änderung ein Verfallsdatum. Dabei wird der Zeitpunkt der letzten Änderung verwendet, um die Zeitspanne seit der letzten Änderung zu ermitteln, die anschließend mit dem in der Anweisung CacheLastModifiedFactor angegebenen Wert multipliziert wird. Das Ergebnis dieser Berechnung ist die Lebensdauer dieser Datei bzw. die Zeitperiode, nach deren Ablauf die Datei als veraltet gilt.

Sie können auch `off` oder `-1` angeben, um die Anweisung zu inaktivieren und kein Verfallsdatum zu berechnen. Der Proxy-Server liest die CacheLastModifiedFactor-Anweisungen in der Reihenfolge, in der sie in der Konfigurationsdatei angegeben sind. Er verwendet die erste Anweisung, die auf die zwischengespeicherte Datei angewendet werden kann.

Format

```
CacheLastModifiedFactor URL Faktor
```

URL
: Der vollständige URL der Datei, die im Cache gespeichert werden soll, einschließlich des Protokolls. Sie können eine URL-Schablone mit Sternen (`*`) als Platzhalterzeichen verwenden, um eine Maske einzusetzen.

Faktor
: Der Faktor, der in der Berechnung verwendet werden soll. Die Werte `off` oder `-1` können ebenfalls angegeben werden.

Beispiele

```
CacheLastModifiedFactor *://hosta/* off
CacheLastModifiedFactor ftp://hostb/* 0.30
CacheLastModifiedFactor ftp://* 0.25
CacheLastModifiedFactor http://* 0.10
CacheLastModifiedFactor * 0.50
```

Standardwerte

```
CacheLastModifiedFactor http://*/ 0.10
CacheLastModifiedFactor http://*.htm* 0.20
CacheLastModifiedFactor http://*.gif 1.00
CacheLastModifiedFactor http://*.jpg 1.00
CacheLastModifiedFactor http://*.jpeg 1.00
CacheLastModifiedFactor http://*.png 1.00
CacheLastModifiedFactor http://*.tar 1.00
CacheLastModifiedFactor http://*.zip 1.00
CacheLastModifiedFactor http:* 0.15
CacheLastModifiedFactor ftp:* 0.50
CacheLastModifiedFactor * 0.10
```

Der Standardwert von 0,14 führt dazu, dass eine Woche zuvor geänderte Dateien nach einem Tag verfallen.

CacheLocalDomain - Angeben, ob die lokale Netzdomäne im Cache gespeichert werden soll

Mit dieser Anweisung können Sie festlegen, ob URLs von Hosts aus derselben Domäne wie der Proxy-Server in den Cache gestellt werden sollen. Lokale Sites in einem Intranet müssen normalerweise nicht in den Cache gestellt werden, weil die interne Bandbreite ausreicht, um die URLs schnell zu laden. Werden lokale Sites nicht in den Cache gestellt, steht für URLs, deren Abrufen längere Zeit in Anspruch nimmt, mehr Cache-Speicher zu Verfügung.

Format

```
CacheLocalDomain {on | off}
```

Standardwert

```
CacheLocalDomain on
```

CacheMatchLanguage — Sprachvorgabe für zurückgegebene Cache-Inhalte festlegen

Wenn der Back-End-Server in der Lage ist, Inhalte für denselben URL in mehreren Sprachen zurückzugeben, können Sie diese Anweisung für die Unterstützung des Caching von Inhalten für URLs in mehreren Sprachen verwenden. Mit dieser Anweisung kann Caching Proxy die Sprachvorgabe in den Anforderungen mit der Sprache der Antworten im Cache vergleichen.

Wenn CacheMatchLanguage aktiviert ist, vergleicht Caching Proxy vor dem Laden des Inhalts aus dem Cache die Sprachvorgabe im Header Accept-Language Ihrer Anforderung mit der Sprache des Inhalts im Cache. Außerdem stellt Caching Proxy die Abweichung von der Vorgabe fest. Falls die Abweichung von der Vorgabe kleiner als der angegebene Grenzwert ist, wird die Kopie aus dem Cache zurückgegeben. Andernfalls leitet der Proxy-Server die Anforderung an den Back-End-Server weiter, um eine aktuelle Kopie in der angeforderten Sprache abzurufen.

Format

```
CacheMatchLanguage {on | off}  Abweichung_Sprachvorgabe Sonder-ID_für_alle_Sprachen
```

Abweichung_Sprachvorgabe
Geben Sie einen Wert zwischen 0,001 und 0,9999 an.

Sonder-ID_für_alle_Sprachen
Geben Sie eine Sprachenzeichenfolge an, die der Server im Header Content-Language zurückgibt, um dem Proxy-Server mitzuteilen, dass die Antwort für alle Sprachvorgaben verwendet werden kann.

Beispiele

Das folgende Beispiel zeigt die Konfiguration der Anweisung, des Cache-Objekts und der Anforderung:

```
CacheMatchLanguage On 0.2
```

Das Cache-Objekt ist vereinfachtes Chinesisch (zh_cn), und die Anforderung ist wie folgt:

```
GET / HTTP/1.1
...
Accept-Language: en_US;q=1.0, zh_cn;q=0.7, ja;q=0.3
....
```

In diesem Beispiel fordert der Benutzer eine Seite in Englisch (mit Code und Qualität en_US/1.0), dann in vereinfachtem Chinesisch (mit Code und Qualität zh_cn/0.7) und dann in Japanisch (mit Code und Qualität ja/0.3) an. Das Cache-Objekt liegt in vereinfachtem Chinesisch vor. Die Abweichung von der Vorgabe zwischen höchster erwarteter Qualität und gefundener Sprachqualität ist 1.0 - 0.7 = 0.3. Da in der Anweisung CacheMatchLanguage ein Grenzwert von 0.2 angegeben ist und 0.3 größer als der Grenzwert ist, fordert der Proxy-Server eine neue Kopie dieses URL vom Server an, anstatt das Cache-Objekt zurückzugeben.

Falls der Server beim Zurückgegeben der Antwort keine Sprache und auch keine Sonder-ID für alle Sprachen im Header Content-Language zurückgibt, vergleicht der Proxy-Server bei der nächsten eingehenden Anforderung die Sprachvorgabe nicht und gibt die Kopie aus dem Cache zurück.

Standardwert

```
CacheMatchLanguage off
```

CacheMaxExpiry - Maximale Lebensdauer für Cache-Dateien angeben

Mit dieser Anweisung können Sie die maximale Verweildauer für Dateien im Cache festlegen. Die Lebensdauer einer Cache-Datei entspricht der Zeitdauer, während der die Datei aus dem Cache bereitgestellt wird, ohne dass der Ursprungsserver auf Aktualisierungen überprüft wird. In einigen Fällen kann die für eine Cache-Datei berechnete Lebensdauer länger sein als die gewünschte Dauer der Speicherung. Die Lebensdauer der Datei, entweder durch den Ursprungsserver vorgegeben oder von Caching Proxy berechnet, darf nicht größer sein als der mit der Anweisung CacheMaxExpiry angegebene Grenzwert.

Es können mehrere Vorkommen dieser Anweisung in der Konfigurationsdatei enthalten sein. Fügen Sie für jede Schablone eine eigene Anweisung ein.

Format

```
CacheMaxExpiry URL Lebensdauer
```

URL
: Der vollständige URL der Datei, die im Cache gespeichert werden soll, einschließlich des Protokolls. Sie können eine URL-Schablone mit Sternen (*) als Platzhalterzeichen verwenden, um eine Maske einzusetzen.

Lebensdauer
: Die maximale Lebensdauer für Cache-Dateien, die mit der URL-Schablone übereinstimmen. Die Zeit kann in einer beliebigen Kombination aus Monaten, Wochen, Tagen, Stunden, Minuten und Sekunden angegeben werden.

Beispiele

```
CacheMaxExpiry ftp:* 1 month
CacheMaxExpiry http://www.santaclaus.np/* 2 days 12 hours
```

Standardwert

```
CacheMaxExpiry 1 month
```

CacheMemory - Den Cache-RAM angeben

Mit dieser Anweisung können Sie die für den Cache zu reservierende Speicherkapazität festlegen. Für eine optimale Leistung der Platten-Caches wird für den Cache-Speicher ein Mindestwert von 64 MB zur Unterstützung der Cache-Infrastruktur und des Cache-Indexes empfohlen. Mit zunehmender Größe des Cache wächst auch der Cache-Index an, und es wird zusätzlicher Cache-Speicher für das Speichern des Index benötigt. Ein Cache-Speicherwert von 64 MB ist groß genug zur Unterstützung der Cache-Infrastruktur und zum Speichern eines Cache-Indexes für einen Platten-Cache mit ca. 6,4 GB. Für größere Platten-Caches sollte der Cache-Speicher 1 % der Cache-Größe betragen.

Wird ein Hauptspeicher-Cache verwendet, sollte diese Anweisung auf einen Wert gesetzt werden, der die Cache-Größe plus die für den Cache-Index benötigte Speicherkapazität umfasst.

Der empfohlene Maximalwert für diese Anweisung sind 1600 MB. Dieser Grenzwert ergibt sich aus der Tatsache, dass Caching Proxy als 32-Bit-Anwendung maximal 2 GB Speicher verwenden kann. Wenn sich der Speicherbereich, der sich aus dem Speicher für den Cache plus dem Speicher für die Routineverarbeitung ergibt, dem Grenzwert von 2 GB nähert oder diesen überschreitet, arbeitet Caching Proxy nicht mehr ordnungsgemäß.

Die Speicherkapazität kann in Byte (B), Kilobyte (K), Megabyte (M) oder Gigabyte (G) angegeben werden.

Format

```
CacheMemory amount {B | K | M | G}
```

Standardwert

```
CacheMemory 64 M
```

CacheMinHold - Angeben, wie lange Dateien verfügbar bleiben sollen

Geben Sie mit dieser Anweisung die URLs für Dateien an, deren Verfallszeiten außer Kraft gesetzt werden sollen. Einige Sites legen als Verfallszeit von Dateien einen Zeitpunkt fest, der vor Ablauf ihrer Lebensdauer liegt, so dass der Server diese Dateien häufiger abrufen muss. Die Anweisung CacheMinHold bewirkt, dass die verfallene Datei für die angegebene Zeit im Cache gespeichert bleibt, bevor sie erneut abgerufen wird. Sie können mehrere dieser Anweisungen definieren.

Anmerkung: Wenn Verfallsdaten außer Kraft gesetzt werden, kann es passieren, dass die Dateien im Cache veraltet und nicht mehr auf dem neuesten Stand sind.

Beispiel

```
CacheMinHold http://www.cachebusters.com/* 1 hour
```

Standardwert
Keiner.

CacheNoConnect - Eigenständigen Cache-Modus angeben

Geben Sie mit dieser Anweisung an, ob der Proxy-Server Dateien von fernen Servern abruft. Bei Verwendung des Standardwerts (`Off`) kann der Server Dateien von fernen Servern abrufen. Bei Verwendung des Werts `On` wird der Server im eigenständigen Cache-Modus ausgeführt. Das bedeutet, dass der Server nur die Dateien zurückgeben kann, die sich bereits in seinem Cache befinden. Wenn der Server in diesem Modus arbeitet, sollte die Anweisung CacheExpiryCheck auf zusätzlich auf `Off` gesetzt werden.

Die Ausführung des Servers im eigenständigen Cache-Modus kann sinnvoll sein, wenn Sie den Server für Demonstrationszwecke verwenden. Wenn Sie sicher sind, dass im Cache alle für die Demonstration benötigten Dateien gespeichert sind, ist keine Netzverbindung erforderlich.

Format
```
CacheNoConnect  {on | off}
```

Standardwert
```
CacheNoConnect Off
```

CacheOnly - Nur Dateien im Cache speichern, deren URLs mit einer Schablone übereinstimmen

Mit dieser Anweisung können Sie festlegen, dass nur die Dateien in den Cache gestellt werden sollen, deren URLs mit einer bestimmten Schablone übereinstimmen. Diese Anweisung kann in der Konfigurationsdatei mehrfach angegeben werden. Fügen Sie für jede Schablone eine eigene Anweisung ein. Die URL-Schablone muss das Protokoll enthalten. Wird für diese Anweisung kein Wert angegeben, können alle URLs im Cache gespeichert werden, die nicht durch eine NoCaching-Anweisung ausgeschlossen werden. Enthält die Konfigurationsdatei weder eine CacheOnly-Anweisung noch eine NoCaching-Anweisung, können alle URLs im Cache gespeichert werden.

Format
```
CacheOnly  URL-Muster
```

Beispiel
```
CacheOnly http://realstuff/*
```

Standardwert
Keiner.

CacheQueries - Cache-Antworten auf URLs festlegen, die ein Fragezeichen (?) enthalten

Geben Sie mit dieser Anweisung die URLs an, für die die Antworten auf Abfragen im Cache gespeichert werden sollen. Bei Verwendung des Werts `PUBLIC` *URL-Muster* werden Antworten auf GET-Anforderungen, die ein Fragezeichen im URL enthalten, dann im Cache gespeichert, wenn der Ursprungsserver den Header `cache-control: public` angibt und ein Caching der Antwort möglich ist. Bei Verwendung des Werts `ALWAYS` *URL-Muster* werden Antworten auf GET-Anforderungen, die ein Fragezeichen im URL enthalten, immer im Cache gespeichert, sofern das Caching der Antworten generell möglich ist.

Sie können mehrere dieser Anweisungen definieren.

```
CacheQueries {ALWAYS | PUBLIC} URL-Muster
```

Beispiele

```
CacheQueries ALWAYS http://www.hosta.com/*
CacheQueries PUBLIC http://www.hostb.com/*
```

Anmerkung: Aus Gründen der Abwärtskompatibilität wird die frühere Syntax von `CacheQueries {ALWAYS | PUBLIC | NEVER}` wie folgt behandelt:

- Die Angaben `CacheQueries ALWAYS` und `CacheQueries PUBLIC` werden wie `CacheQueries ALWAYS *` und `CacheQueries PUBLIC *` behandelt.
- `CacheQueries NEVER` wird ignoriert.
- Wenn sowohl `CacheQueries NEVER` als auch `CacheQueries URL-Muster` angegeben wird, wird die Anweisung `CacheQueries NEVER` ignoriert, aber eine Warnung wird zurückgegeben.

Standardwert

Keiner.

CacheRefreshInterval - Zeitintervall für erneut Überprüfung von Cache-Objekten angeben

Legen Sie mit dieser Anweisung das Zeitintervall fest, in dem auf dem Ursprungsserver überprüft werden soll, ob eine im Cache gespeicherte Datei in der Zwischenzeit geändert wurde.

Obwohl die Anweisung CacheClean eine Ähnlichkeit mit dieser Anweisung aufweist, besteht doch ein Unterschied. Die Anweisung CacheRefreshInterval legt lediglich fest, dass der Proxy-Server die Gültigkeit einer Datei überprüfen soll, bevor er sie verwendet, während die Anweisung CacheClean bewirkt, dass die Datei nach einer angegebenen Zeitperiode aus dem Cache gelöscht wird.

Format

- Mit dem folgenden Format wird das Aktualisierungsintervall für alle Dokumente angegeben, die mit dem URL-Muster übereinstimmen:

  ```
  CacheRefreshInterval URL_Muster Zeitperiode
  ```

- Mit dem folgenden Format wird das Aktualisierungsintervall für alle Dokumente angegeben, die *nicht* mit einem URL-Muster übereinstimmen. Es wird lediglich ein Aktualisierungsintervall angegeben.

  ```
  CacheRefreshInterval Zeitperiode
  ```

Beispiele

```
CacheRefreshInterval *.gif 8 hours
CacheRefreshInterval 1 week
```

Standardwert

```
CacheRefreshInterval 2 weeks
```

CacheRefreshTime - Startzeitpunkt für Cache-Agenten angeben

Geben Sie mit dieser Anweisung den Startzeitpunkt des Cache-Agenten an. Der Cache-Agent kann zu einer bestimmten Uhrzeit gestartet werden.

Format

```
CacheRefreshTime HH:MM
```

Standardwert

```
CacheRefreshTime 03:00
```

CacheTimeMargin - Mindestlebensdauer für das Caching einer Datei angeben

Mit der Anweisung CacheTimeMargin wird festgelegt, wie lang die Lebensdauer einer Datei mindestens sein muss, damit sie im Cache gespeichert wird.

Caching Proxy ermittelt für jede Datei eine Verfallszeit. Wenn es unwahrscheinlich ist, dass die Datei vor Ablauf ihrer Verfallszeit von einer anderen Anforderung abgerufen wird, betrachtet Caching Proxy die Lebensdauer der Datei als zu kurz für das Caching. Standardmäßig werden von Caching Proxy keine Dateien im Cache gespeichert, deren Lebensdauer kürzer ist als 10 Minuten. Sollte der Cache noch nicht nahe an seiner maximalen Kapazitätsgrenze arbeiten, verwenden Sie für diese Anweisung den Anfangswert. Ist die Kapazitätsgrenze des Cache fast erreicht, sollten Sie den Wert für die Mindestlebensdauer eventuell erhöhen.

Format

```
CacheTimeMargin Mindestlebensdauer
```

Standardwert

```
CacheTimeMargin 10 minutes
```

Anmerkung: Wird diese Anweisung auf einen Wert über vier Stunden gesetzt, wird die Effizienz des Cache drastisch vermindert.

CacheUnused - Zeitlimit für ungenutzte Dateien im Cache angeben

Geben Sie mit dieser Anweisung die maximale Zeit an, während der der Server ungenutzte Dateien, die mit einer angegebenen Schablone übereinstimmen, im Cache speichern soll. Der Server löscht die ungenutzten Dateien, deren URLs mit der Schablone übereinstimmen, ungeachtet ihres Verfallsdatums nach der angegebenen Speicherzeit aus dem Cache. Sie können mehrere dieser Anweisungen in der Konfigurationsdatei definieren. Fügen Sie für jede Schablone eine eigene Anweisung ein. Die URL-Schablone muss das Protokoll enthalten. Geben Sie die Zeit in einer beliebigen Kombination aus Monaten, Wochen, Tagen und Stunden an.

Format

```
CacheUnused URL-Schablone Zeitperiode
```

Beispiele

```
CacheUnused ftp:* 3 weeks
CacheUnused gopher:* 3 days 12 hours
CacheUnused * 4 weeks
```

Standardwerte

```
CacheUnused ftp:*  3 days
CacheUnused gopher:*   12 hours
CacheUnused http:* 2 days
```

Caching - Proxy-Caching aktivieren

Mit dieser Anweisung können Sie das Caching für Dateien aktivieren. Wenn das Caching aktiviert ist, stellt der Proxy-Server die Dateien, die er von anderen Servern abruft, in einen lokalen Cache. Der Proxy-Server kann nachfolgende Anforderungen für dieselben Dateien aus dem Cache bedienen, ohne die Dateien von anderen Servern abrufen zu müssen.

Format

```
Caching {on | off}
```

Standardwert

```
Caching On
```

Anmerkung: Wenn Sie die Anweisung Caching ändern, müssen Sie den Server manuell stoppen und anschließend erneut starten. (Nähere Informationen hierzu finden Sie in Kapitel 5, „Caching Proxy starten und stoppen", auf Seite 15.)

CdfAware - Instanz von Caching Proxy als Komponente des Content Distribution Framework festlegen

Mit dieser Anweisung können Sie festlegen, ob Caching Proxy eine Komponente eines Content Distribution Framework ist.

Format

```
CdfAware {yes | no}
```

Standardwert

```
CdfAware no
```

CdfRestartFile - Datei zum Speichern der Zuordnung von Dateiname zu URL angeben

Mit dieser Anweisung können Sie den Namen der Datei angeben, in der die Daten für die Zuordnung von Dateinamen zu URLs gespeichert sind, so dass diese Daten über mehrere Instanzen von ibmproxy hinweg in einem Content Distribution Framework festgeschrieben werden. Caching Proxy verwaltet eine Zuordnungstabelle, in der ein angeforderter URL einem Dateinamen auf dem Webserver zugeordnet ist. In dieser Datei wird diese Tabelle persistent gespeichert, so dass die Daten nach den Neustarts nicht verloren gehen. Verwenden Sie diese Anweisung nur dann, wenn die Anweisung CdfAware auf `yes` gesetzt ist.

Format

```
CdfRestartFile Pfad/Dateiname
```

Beispiel

- **Linux und UNIX:** `CdfRestartFile /opt/ibm/edge/cd/cdfRestartFile`
- **Windows:** `CdfRestartFile C:\progra~1\ibm\edge\cd\cdfRestartFile.txt`

Standardwert

Keiner.

CompressAge - Zeitpunkt für die Komprimierung von Protokollen angeben

Mit dieser Anweisung können Sie festlegen, wann Protokolle komprimiert werden sollen. Wenn die Protokolle älter sind als der mit CompressAge gesetzte Wert, werden sie komprimiert. Wenn CompressAge den Wert `0` hat, werden keine Protokolle komprimiert. Die Protokolle für den aktuellen Tag und den Tag davor werden nicht komprimiert.

Format

`CompressAge` *`Anzahl_Tage`*

Standardwert

```
CompressAge 1
```

Zugehörige Anweisungen

- „CompressDeleteAge - Zeitpunkt zum Löschen von Protokollen angeben" auf Seite 202
- „CompressCommand - Komprimierungsbefehl und zugehörige Parameter angeben"
- „LogArchive - Verhalten der Protokollarchivierung angeben" auf Seite 235
- „Midnight - API-Plug-in für die Archivierung von Protokollen angeben" auf Seite 242
- „PurgeAge - Altersgrenze für ein Protokoll angeben" auf Seite 267
- „PurgeSize - Begrenzung für die Größe des Protokollarchivs angeben" auf Seite 267

CompressCommand - Komprimierungsbefehl und zugehörige Parameter angeben

Mit dieser Anweisung können Sie einen Befehl erstellen, der das Dienstprogramm zum Komprimieren der Protokolle festlegt und Parameter an dieses Dienstprogramm übergibt. Geben Sie auch den Pfad für die archivierten Protokolle mit an.

Das Komprimierungsdienstprogramm muss in einem Verzeichnis installiert werden, das im Pfad für diese Maschine enthalten ist.

Format

`CompressCommand` *`Befehl`*

`Befehl`
Der Befehl und die Parameter, die verwendet werden sollen, angegeben in einer Zeile. Normalerweise werden die Parameter `%%LOGFILES%%` und `%%DATE%%` immer verwendet.

`%%LOGFILES%%`
Die Liste der Protokolldateien, die für ein bestimmtes Datum (`%%DATE%%`) verfügbar sind.

`%%DATE%%`
Die Datumsmarke in einer Protokolldatei.

Beispiele

- **Linux und UNIX:**

```
CompressCommand tar -cf /logarchs/log%%DATE%%.tar %%LOGFILES%% ;
                    gzip /logarchs/log%%DATE%%.tar
CompressCommand tar -cf /logarchs/log%%DATE%%.tar %%LOGFILES%% ;
                    compress /logarchs/log%%DATE%%.tar
CompressCommand zip -q /logarchs/log%%DATE%%.zip %%LOGFILES%%
```

  Anmerkung: Der Befehl und alle seine Parameter müssen in einer Zeile stehen. In den vorangehenden Beispielen wurden die beiden ersten Befehle aus Gründen der Lesbarkeit auf mehrere Zeilen verteilt.

- **Windows:**

```
CompressCommand pkzip -q d:\logarchs\log%%DATE%%.tar
%%LOGFILES%%
```

Standardwert

Keiner.

Zugehörige Anweisungen

- „CompressAge - Zeitpunkt für die Komprimierung von Protokollen angeben" auf Seite 201
- „CompressDeleteAge - Zeitpunkt zum Löschen von Protokollen angeben"
- „LogArchive - Verhalten der Protokollarchivierung angeben" auf Seite 235
- „Midnight - API-Plug-in für die Archivierung von Protokollen angeben" auf Seite 242
- „PurgeAge - Altersgrenze für ein Protokoll angeben" auf Seite 267
- „PurgeSize - Begrenzung für die Größe des Protokollarchivs angeben" auf Seite 267

CompressDeleteAge - Zeitpunkt zum Löschen von Protokollen angeben

Mit dieser Anweisung können Sie angeben, wann ein komprimiertes Protokoll gelöscht werden soll. Wenn ein Protokoll älter ist als die für CompressDeleteAge angegebenen Tage, wird es gelöscht. Ist die Anweisung CompressDeleteAge auf `0` gesetzt oder ist ihr Wert niedriger als der Wert für die Anweisung CompressAge, wird ein Protokoll nicht gelöscht.

Anmerkung: Das Plug-in für Komprimierung löscht niemals die Protokolle für den aktuellen oder den vergangenen Tag.

Format

```
CompressDeleteAge Anzahl_Tage
```

Standardwert

```
CompressDeleteAge 7
```

Zugehörige Anweisungen

- „CompressAge - Zeitpunkt für die Komprimierung von Protokollen angeben" auf Seite 201
- „CompressCommand - Komprimierungsbefehl und zugehörige Parameter angeben" auf Seite 201
- „LogArchive - Verhalten der Protokollarchivierung angeben" auf Seite 235

- „Midnight - API-Plug-in für die Archivierung von Protokollen angeben" auf Seite 242
- „PurgeAge - Altersgrenze für ein Protokoll angeben" auf Seite 267
- „PurgeSize - Begrenzung für die Größe des Protokollarchivs angeben" auf Seite 267

ConfigFile — Name einer weiteren Konfigurationsdatei angeben

Mit dieser Anweisung können Sie Namen und Position einer zusätzlichen Konfigurationsdatei angeben. Die in der angegebenen Konfigurationsdatei enthaltenen Anweisungen werden im Anschluss an die aktuelle Konfigurationsdatei abgearbeitet.

Anmerkung: Stellen Sie sicher, dass die Berechtigung der zusätzlichen Konfigurationsdatei für den Benutzer `nobody` auf `Read` gesetzt ist, damit der Cache-Agent diese Datei lesen kann.

Beispiele

- **Linux und UNIX:** `ConfigFile /etc/rca.conf`
- **Windows:** `ConfigFile c:\WINNT\rca.conf`

Standardwert

Keiner.

ConnThreads — Anzahl der Verbindungs-Threads für die Verbindungsverwaltung festlegen

Mit dieser Anweisung können Sie die Anzahl der Verbindungs-Threads festlegen, die für die Verwaltung von Verbindungen zu verwenden sind.

Format

`ConnThreads` *`Anzahl`*

Standardwert

`ConnThreads 5`

Zugehörige Anweisungen

- „MaxActiveThreads - Die maximale Anzahl aktiver Threads angeben" auf Seite 238

ContinueCaching - Angeben, wie viel Prozent einer Datei für das Caching erforderlich sind

Mit dieser Anweisung können Sie angeben, wie viel Prozent der angeforderten Datei übertragen werden müssen, damit Caching Proxy die Cache-Datei vollständig erstellen kann, selbst wenn die Clientverbindung beendet wird. Gültige Werte für diese Variable sind ganze Zahlen im Bereich von 0 - 100.

Wird beispielsweise `ContinueCaching 75` angegeben, setzt Caching Proxy die Übertragung der Datei vom Inhaltsserver fort und generiert die Cache-Datei, wenn mindestens 75 % der Datei bereits übertragen wurden, bevor Caching Proxy feststellt, dass die Clientverbindung beendet wurde.

Format

```
ContinueCaching Prozentsatz
```

Standardwert

```
ContinueCaching 75
```

DefinePicsRule - Regel für Inhaltsfilterung angeben

Legen Sie mit dieser Anweisung die Informationen für den Proxy-Server fest, die erforderlich sind, damit URLs auf Inhalte gefiltert werden, die Informationen zum Bewertungsservice enthalten. Sie können mehrere dieser Anweisungen angeben.

Format

```
DefinePicsRule "Filtername"  {
```

Standardwert

```
DefinePicsRule "RSAC-Beispiel" {
```

DefProt - Standardzugriffsschutzkonfiguration für Anforderungen angeben, die mit einer Schablone übereinstimmen

Mit dieser Anweisung können Sie den Anforderungen, die mit einer Schablone übereinstimmen, eine Standardzugriffsschutzkonfiguration zuordnen.

Anmerkung: Damit der Zugriffsschutz ordnungsgemäß funktioniert, müssen die Anweisungen DefProt und Protect in der Konfigurationsdatei vor allen Pass- oder Exec-Anweisungen stehen.

Format

```
DefProt Anforderungsschablone  Zugriffsschutzkonfiguration
[FOR Server-IP-Adresse | Hostname]
```

Anforderungsschablone
Eine Schablone für Anforderungen, denen eine Standardzugriffsschutzkonfiguration zugeordnet werden soll. Der Server vergleicht die eingehenden Clientanforderungen mit der Schablone und ordnet ihnen bei Übereinstimmung eine Zugriffsschutzkonfiguration zu.

Der Zugriffsschutz wird jedoch bei Übereinstimmung von Anforderungen mit der Schablone noch nicht aktiviert, es sei denn, die Anforderung stimmt ebenfalls mit der Schablone einer nachfolgenden Protect-Anweisung überein. Eine Beschreibung dazu, wie die Anweisung Protect mit DefProt verwendet wird, finden Sie im Abschnitt „Protect - Eine Zugriffsschutzkonfiguration für Anforderungen aktivieren, die mit einer Schablone übereinstimmen" auf Seite 253.

Zugriffschutzkonfiguration
Die benannte Zugriffsschutzkonfiguration, die in der Konfigurationsdatei definiert ist und die den Anforderungen zugeordnet werden soll, die mit der *Anforderungsschablone* übereinstimmen. Die Zugriffsschutzkonfiguration wird mit untergeordneten Anweisungen für den Zugriffsschutz definiert. Dieser Parameter kann eines von drei Formaten verwenden:

- Ein vollständiger Pfad und Dateiname einer separaten Datei, die die untergeordneten Anweisungen für den Zugriffsschutz enthält.
- Ein Kennsatzname für eine Zugriffsschutzkonfiguration, der mit dem Namen übereinstimmt, der zuvor in der Anweisung Protection definiert wurde. Die Anweisung Protection enthält die untergeordneten Anweisungen für den Zugriffsschutz.

- Die eigentlichen untergeordneten Anweisungen für den Zugriffsschutz. Die untergeordneten Anweisungen müssen in geschweifte Klammern ({}) eingeschlossen werden. Die linke geschweifte Klammer muss in der Zeile, in der die Anweisung DefProt enthalten ist, das letzte Zeichen sein. Jede untergeordnete Anweisung steht in einer separaten Zeile. Die rechte geschweifte Klammer muss im Anschluss an die letzte untergeordnete Anweisung in einer separaten Zeile stehen. Zwischen den geschweiften Klammern dürfen keine Kommentarzeilen enthalten sein. Beschreibungen der untergeordneten Anweisungen für den Zugriffsschutz finden Sie in den folgenden Abschnitten:
  - „AuthType - Authentifizierungsart angeben" auf Seite 259
  - „DeleteMask - Die Benutzernamen, Gruppen und Adressen angeben, denen das Löschen von Dateien erlaubt ist" auf Seite 259
  - „GetMask - Die Benutzernamen, Gruppen und Adressen angeben, denen das Abrufen von Dateien erlaubt ist" auf Seite 259
  - „GroupFile - Die Position der zugeordneten Gruppendatei angeben" auf Seite 260
  - „Mask - Die Benutzernamen, Gruppen und Adressen angeben, die HTTP-Anforderungen stellen dürfen" auf Seite 260
  - „PasswdFile - Die Position der zugeordneten Kennwortdatei angeben" auf Seite 260
  - „PostMask - Die Benutzernamen, Gruppen und Adressen angeben, denen das Versenden von Dateien erlaubt ist" auf Seite 261
  - „PutMask - Die Benutzernamen, Gruppen und Adressen angeben, denen das Ablegen von Dateien erlaubt ist" auf Seite 261
  - „ServerID - Einen Namen angeben, der der Kennwortdatei zugeordnet werden soll" auf Seite 261

[FOR *Server-IP-Adresse* | *Hostname*]

Falls Sie mehrere IP-Adressen oder virtuelle Hosts verwenden, geben Sie mit diesem Parameter eine IP-Adresse oder einen Hostnamen an. Der Server verwendet die Anweisung nur für Anforderungen an diese IP-Adresse oder für diesen Host. Bei einer IP-Adresse ist dies die Adresse der Netzverbindung des Servers und nicht die Adresse des Clients, von dem die Anforderung stammt.

Sie können eine IP-Adresse angeben (z. B. `FOR 240.146.167.72`) oder einen Hostnamen (z. B. `FOR hostA.bcd.com`).

Dieser Parameter ist optional. Ohne Angabe dieses Parameters verwendet der Server die Anweisung für alle Anforderungen ungeachtet der IP-Adresse, unter der die Anforderungen ankommen, oder des Hostnamens im URL.

Anmerkungen:

1. Dieser Parameter darf nur verwendet werden, wenn der Parameter *Zugriffsschutzkonfiguration* als Pfad und Dateiname oder als Kennsatz einer Zugriffsschutzkonfiguration angegeben wurde. Wurden für den Parameter *Zugriffsschutzkonfiguration* die tatsächlich verwendeten und in geschweifte Klammern eingeschlossenen untergeordneten Anweisungen für Zugriffsschutz angegeben, darf dieser Parameter nicht verwendet werden.
2. Damit dieser Parameter verwendet werden kann, muss die Zeichenfolge `FOR` oder eine andere Zeichenfolge (ohne Leerzeichen) zwischen den Parameter *Zugriffsschutzkonfiguration* und den Parameter *IP-Adresse* oder *Hostname* gesetzt werden.

Für die IP-Adresse eines Servers kann kein Platzhalterzeichen verwendet werden.

Anmerkung: Die Anweisung muss in einer Zeile eingegeben werden.

Beispiele

- Das folgende Beispiel gibt eine separate Datei an, die die untergeordneten Anweisungen für den Zugriffsschutz enthält.

```
DefProt /secret/* /server/protect/setup1.acc
```

- Im folgenden Beispiel wird ein Kennsatzname verwendet, der auf die untergeordneten Anweisungen für den Zugriffsschutz verweist. Dieser Kennsatzname muss mit einem Kennsatznamen einer Protection-Anweisung übereinstimmen. Dabei muss die Protection-Anweisung vor der Anweisung DefProt stehen.

```
DefProt /secret/* SECRET-PROT
```

- Das folgende Beispiel enthält die untergeordneten Anweisungen für den Zugriffsschutz als Teil der Anweisung DefProt.

```
DefProt {
  AuthType Basic
   ServerID restricted
  PasswdFile /docs/etc/WWW/restrict.password
  GroupFile /docs/etc/WWW/restrict.group
   GetMask authors
   PutMask authors
  }
```

- Die folgenden Beispiele verwenden den optionalen Parameter IP-Adresse. Wenn der Server Anforderungen empfängt, die mit `/secret/` beginnen, wird diesen Anforderungen je nach der IP-Adresse der Netzverbindung, über die die Anforderungen empfangen werden, eine andere Standardzugriffsschutzkonfiguration zugeordnet. Den Anforderungen, die an der Adresse 0.67.106.79 empfangen werden, ordnet der Server den Standardzugriffsschutz zu, der in einer Protection-Anweisung mit dem Kennsatz `CustomerA-PROT` definiert ist. Den Anforderungen, die unter der Adresse 0.83.100.45 empfangen werden, ordnet der Server den Standardzugriffsschutz zu, der in einer Protection-Anweisung mit dem Kennsatz `CustomerB-PROT` definiert ist.

```
DefProt   /secret/*   CustomerA-PROT   0.67.106.79
DefProt   /secret/*   CustomerB-PROT   0.83.100.45
```

- In den folgenden Beispielen wird der optionale Parameter Hostname verwendet. Wenn Ihr Server Anforderungen empfängt, die mit `/secret/` beginnen, wird diesen Anforderungen auf der Basis des Hostnamens im URL eine andere Standardzugriffsschutzkonfiguration zugeordnet. Den Anforderungen, die für hostA empfangen werden, ordnet der Server den Standardzugriffsschutz zu, der in einer Protection-Anweisung mit dem `CustomerA-PROT` definiert ist. Den Anforderungen, die für hostB empfangen werden, ordnet der Server den Standardzugriffsschutz zu, der in einer Protection-Anweisung mit dem Kennsatz `CustomerB-PROT` definiert ist.

```
DefProt /secret/* CustomerA-PROT hostA.bcd.com
DefProt /secret/* CustomerB-PROT hostB.bcd.com
```

Standardwert

Keiner.

DelayPeriod - Verzögerungen zwischen Anforderungen angeben

Geben Sie mit dieser Anweisung an, ob der Cache-Agent zwischen dem Senden von Anforderungen an die Zielserver warten soll. Durch Angabe einer Verzögerung zwischen den Anforderungen wird die Last auf der Proxy-Maschine und der Netzverbindung sowie auf den Zielservern reduziert. Wird keine Verzögerung angegeben, wird der Cache-Agent mit maximaler Geschwindigkeit ausgeführt. Bei langsamen Internet-Verbindungen empfiehlt es sich, keine Verzögerung festlegen, um das Netz maximal auszulasten.

Anmerkung: Bei einer Internet-Verbindung, die schneller ist als 128 kbps, sollten Sie die Anweisung DelayPeriod auf `On` setzen, damit nicht zu viele Anforderungen zu schnell an Sites gesendet werden, die gerade aktualisiert werden.

Format

```
DelayPeriod {on | off}
```

Standardwert

```
DelayPeriod On
```

DelveAcrossHosts - Domänenübergreifendes Caching angeben

Geben Sie mit dieser Anweisung an, ob der Cache-Agent hostübergreifenden Hypertext-Links folgen soll. Wenn ein zwischengespeicherter URL Links zu anderen Servern enthält, kann der Server sie entweder ignorieren oder ihnen folgen. Wird die Anweisung DelveInto auf `never` gesetzt, wird sie nicht angewendet.

Format

```
DelveAcrossHosts {on | off}
```

Standardwert

```
DelveAcrossHosts Off
```

DelveDepth - Angeben, wie weit den Links beim Caching gefolgt werden soll

Geben Sie mit dieser Anweisung die Anzahl der Link-Ebenen an, denen gefolgt werden soll, wenn der Server nach Seiten sucht, die in den Cache geladen werden sollen. Ist die Anweisung DelveInto auf `never` gesetzt, wird sie nicht angewendet.

Format

```
DelveDepth Anzahl_Link-Ebenen
```

Standardwert

```
DelveDepth 2
```

DelveInto - Angeben, ob der Cache-Agent den Links folgen soll

Geben Sie mit dieser Anweisung an, ob der Cache-Agent Seiten laden soll, die über Links mit den URLs im Cache verknüpft sind.

Format

```
DelveInto {always | never | admin | topn}
```

always
Der Cache-Agent folgt den Links von allen zuvor im Cache gespeicherten URLs.

never
Der Cache-Agent ignoriert alle Links in URLs.

admin
Der Cache-Agent folgt nur den Links der URLs, die in mit LoadURL-Anweisungen angegeben sind.

topn
Der Cache-Agent folgt nur den Links der am häufigsten abgerufenen Dateien im Cache.

Standardwert

```
DelveInto always
```

DirBackgroundImage - Ein Hintergrundbild für Verzeichnislisten angeben

Mit dieser Anweisung können Sie für Verzeichnislisten, die vom Proxy-Server generiert werden, ein Hintergrundbild festlegen. Verzeichnislisten werden generiert, wenn mit dem Proxy-Server FTP-Sites durchsucht werden.

Geben Sie einen absoluten Pfad für das Hintergrundbild an. Befindet sich das Bild auf einem anderen Server, müssen Sie das Hintergrundbild als vollständigen URL angeben. Wird kein Hintergrundbild angegeben, wird ein einfacher weißer Hintergrund verwendet.

Format

```
DirBackgroundImage /Pfad/Datei
```

Beispiele

```
DirBackgroundImage /images/corplogo.png
DirBackgroundimage http://www.somehost.com/graphics/embossed.gif
```

Standardwert

Keiner.

DirShowBytes - Für kleine Dateien in Verzeichnislisten die Bytezahl anzeigen

Legen Sie mit dieser Anweisung fest, ob in Verzeichnislisten für Dateien, die kleiner sind als 1 KB, die exakte Bytezahl angezeigt werden soll. Bei der Angabe von `Off` wird für alle Dateien, die bis zu 1 KB groß sind, eine Dateigröße von 1 KB angezeigt.

Format

```
DirShowBytes {on | off}
```

Standardwert

```
DirShowBytes Off
```

DirShowCase - Behandlung von Groß-/Kleinschreibung beim Sortieren von Dateien in Verzeichnislisten festlegen

Legen Sie mit dieser Anweisung fest, ob beim Sortieren der Dateien in Verzeichnislisten zwischen Groß- und Kleinbuchstaben unterschieden werden soll.

Bei der Angabe von `On` werden Namen in Großbuchstaben in der Dateiliste vor den Namen in Kleinbuchstaben angezeigt.

Format

```
DirShowCase {on | off}
```

Standardwert

```
DirShowCase On
```

DirShowDate - In Verzeichnislisten das Datum der letzten Änderung anzeigen

Legen Sie mit dieser Anweisung fest, ob in Verzeichnislisten für jede Datei das Datum der letzten Änderung angezeigt werden soll.

Format

```
DirShowDate {on | off}
```

Standardwert

```
DirShowDate On
```

DirShowDescription - In Verzeichnislisten Beschreibungen zu den Dateien anzeigen

Legen Sie mit dieser Anweisung fest, ob in Verzeichnislisten Beschreibungen für HTML-Dateien angezeigt werden sollen. Die Beschreibungen werden aus den HTML-Tags `<title>` übernommen.

Beschreibungen für FTP-Verzeichnislisten enthalten die MIME-Typen der Dateien, falls diese ermittelt werden können.

Format

```
DirShowDescription {on | off}
```

Standardwert

```
DirShowDescription On
```

DirShowHidden - In Verzeichnislisten verdeckte Dateien anzeigen

Legen Sie mit dieser Anweisung fest, ob in Verzeichnislisten verdeckte Dateien im Verzeichnis angezeigt werden sollen. Der Server betrachtet jede Datei, deren Name mit einem Punkt (.) beginnt, als verdeckte Datei.

Format

```
DirShowHidden {on | off}
```

Standardwert

```
DirShowHidden On
```

DirShowIcons - In Verzeichnislisten Symbole anzeigen

Geben Sie mit dieser Anweisung an, ob der Server Symbole in Verzeichnislisten anzeigt. Symbole können verwendet werden, um in der Liste eine grafische Darstellung des Inhaltstyps von Dateien zu erhalten. Die Symbole selbst werden mit den Anweisungen AddBlankIcon, AddDirIcon, AddIcon, AddParentIcon und AddUnknownIcon definiert.

Format

```
DirShowIcons {on | off}
```

Standardwert

```
DirShowIcons On
```

DirShowMaxDescrLength - Maximale Länge für Beschreibungen in Verzeichnislisten angeben

Mit dieser Anweisung können Sie die maximale Anzahl der Zeichen festlegen, die im Beschreibungsfeld von Verzeichnislisten dargestellt werden.

Format

```
DirShowMaxDescrLength Anzahl_Zeichen
```

Standardwert

```
DirShowMaxDescrLength 25
```

DirShowMaxLength - Maximale Länge der Dateinamen in Verzeichnislisten angeben

Geben Sie mit dieser Anweisung die maximale Anzahl Zeichen an, die für Dateinamen in Verzeichnislisten verwendet werden kann.

Format

```
DirShowMaxDescrLength Anzahl_Zeichen
```

Standardwert

```
DirShowMaxLength 25
```

DirShowMinLength - Mindestlänge der Dateinamen in Verzeichnislisten angeben

Geben Sie mit dieser Anweisung die Mindestanzahl Zeichen an, die in Verzeichnislisten immer für Dateinamen reserviert ist. Diese Feldlänge kann durch die Dateinamen im Verzeichnis überschritten werden. Jedoch können die Dateinamen nicht länger sein, als in der Anweisung `DirShowMaxLength` festgelegt.

Format

```
DirShowMinLength Anzahl_Zeichen
```

Standardwert

```
DirShowMinLength 15
```

DirShowSize - In Verzeichnislisten die Dateigröße anzeigen

Legen Sie mit dieser Anweisung fest, ob in Verzeichnislisten die Größe jeder Datei angezeigt werden soll.

Format

```
DirShowSize {on | off}
```

Standardwert

```
DirShowSize On
```

Disable - HTTP-Methoden inaktivieren

Legen Sie mit dieser Anweisung fest, welche HTTP-Methoden der Server nicht akzeptiert. Geben Sie für jede Methode, die vom Server zurückgewiesen werden soll, eine separate Disable-Anweisung ein.

In der Standardkonfigurationsdatei sind die Methoden GET, HEAD, OPTIONS, POST und TRACE aktiviert und alle anderen unterstützten HTTP-Methoden inaktiviert. Um eine aktivierte Methode zu inaktivieren, löschen Sie sie aus der Anweisung Enable und fügen Sie sie zur Anweisung Disable hinzu.

Format

```
Disable Methode
```

Anmerkung: In den Konfigurations- und Verwaltungsformularen wird zum Aktualisieren der Serverkonfiguration die Methode POST verwendet. Wenn Sie die Methode POST inaktivieren, können die Konfigurations- und Verwaltungsformulare nicht mehr verwendet werden.

Standardwerte

```
Disable PUT
Disable DELETE
Disable   CONNECT
```

DisInheritEnv - Die Umgebungsvariablen angeben, die durch CGI-Programme nicht übernommen werden sollen

Geben Sie mit dieser Anweisung die Umgebungsvariablen an, die durch Ihre CGI-Programme nicht übernommen werden sollen (andere Variablen als die CGI-Umgebungsvariablen, die speziell für die CGI-Verarbeitung vorgesehen sind).

Standardmäßig werden durch CGI-Programme alle Umgebungsvariablen übernommen. Mit dieser Anweisung können Sie einzelne Umgebungsvariablen von der Übernahme durch GGI-Programme ausschließen.

Format

```
DisInheritEnv Umgebungsvariable
```

Beispiele

```
DisInheritEnv PATH
DisInheritEnv LANG
```

In diesem Beispiel werden alle Umgebungsvariablen außer PATH und LANG durch CGI-Programme übernommen.

Standardwert

Keiner.

DNS-Lookup - Angeben, ob der Server die Hostnamen von Clients suchen soll

Geben Sie mit dieser Anweisung an, ob der Server die Hostnamen der anfragenden Clients suchen soll.

Format

```
DNS-Lookup {on | off}
```

Der verwendete Wert beeinflusst folgende Faktoren des Servers:

- Die Leistung des Servers. Wird der Standardwert `Off` verwendet, werden Leistung und Antwortzeit des Servers verbessert, da für die Suche der Hostnamen keine Ressourcen eingesetzt werden müssen.
- Die Informationen, die der Server zu den Clients in den Protokolldateien aufgezeichnet.
  `Off` - Clients werden nach der IP-Adresse identifiziert.
  `On` - Clients werden nach dem Hostnamen identifiziert.
- Der Wert bestimmt, ob in Zugriffsschutzkonfigurationen, Servergruppendateien und ACL-Dateien (Access Control List = Zugriffssteuerungsliste) Hostnamen in Adressenschablonen verwendet werden können.
  `Off`- In Adressenschablonen dürfen keine Hostnamen verwendet werden, stattdessen müssen IP-Adressen angegeben werden.
  `On` - In Adressenschablonen dürfen Hostnamen verwendet werden, es dürfen keine IP-Adressen angegeben werden.

Anmerkung: Damit in den Zugriffsschutzregeln Domänennamen verwendet werden können, muss die Anweisung DNS-Lookup auf `On` gesetzt sein.

Standardwert

```
DNS-Lookup Off
```

Enable - HTTP-Methoden aktivieren

Legen Sie mit dieser Anweisung fest, welche HTTP-Methoden der Server akzeptiert.

Es können alle benötigten HTTP-Methoden aktiviert werden. Geben Sie für jede Methode, die der Server akzeptieren soll, eine separate Enable-Anweisung ein.

Format

```
Enable Methode
```

Wenn für einen bestimmten URL keine Service-Anweisung vorhanden ist, kann mit der Anweisung Enable eine angepasste Programmierung für eine HTTP-Methode erfolgen. Das mit dieser Anweisung angegebene Programm setzt die Standardverarbeitung für diese Methode außer Kraft.

```
Enable Methode /Pfad/DateiDLL:Funktionsname
```

Standardwerte

```
Enable GET
Enable HEAD
Enable POST
Enable TRACE
Enable OPTIONS
```

EnableTcpNodelay — Socket-Option TCP NODELAY aktivieren

Mit dieser Anweisung können Sie die Socket-Option TCP NODELAY aktivieren.

Die Anweisung EnableTcpNodelay verbessert die Leistung, wenn kleine IP-Paket wie SSL-Handshake- oder HTTP-Kurzantworten zwischen Caching Proxy und dem Client übertragen werden. Die Option TCP NODELAY ist standardmäßig für alle Sockets aktiviert.

Format

```
EnableTcpNodelay {All | HTTP | HTTPS | None}
```

Standardwert

```
EnableTcpNodelay  All
```

Error - Schritt ″Error″ anpassen

Mit dieser Anweisung können Sie eine angepasste Anwendungsfunktion angeben, die der Server während des Schritts ″Error″ aufrufen soll. Dieser Code wird ausgeführt, um beim Auftreten eines Fehlers angepasste Fehlerroutinen bereitzustellen.

Format

```
Error  Anforderungsschablone /Pfad/Datei:Funktionsname
```

Anforderungsschablone
: Eine Schablone für Anforderungen, die festlegt, ob eine Anwendungsfunktion aufgerufen wird. Die Schablone kann als Angaben das Protokoll, die Domäne und den Host enthalten, sie darf als vorangestelltes Zeichen einen Schrägstrich (/) verwenden sowie als Platzhalterzeichen einen Stern (*). Beispielsweise sind folgende Schablonen gültig: `/front_page.html`, `http://www.ics.raleigh.ibm.com`, `/pub*`, `/*` und `*`.

/Pfad/Datei
Der vollständig qualifizierte Dateiname des kompilierten Programms einschließlich der Erweiterung.

Funktionsname
Der Name der Anwendungsfunktion in Ihrem Programm.

Beispiel
```
Error /index.html /ics/api/bin/icsext05.so:error_rtns
```

Standardwert
Keiner.

ErrorLog - Die Datei zur Protokollierung von Serverfehlern angeben

Mit dieser Anweisung können Sie den Pfad und den Namen der Datei angeben, in der der Server die internen Fehler protokollieren soll.

Anmerkung: Wenn Sie die Standardeinstellungen des Servers für Benutzer-ID, Gruppen-ID oder Protokollverzeichnispfade ändern, müssen Sie die neuen Verzeichnisse erstellen und die für diese Verzeichnisse erforderlichen Berechtigungen und Eigentumsrechte aktualisieren. Damit der Server Daten in ein benutzerdefiniertes Protokollverzeichnis schreiben kann, muss die Berechtigung für dieses Verzeichnis auf 755 gesetzt werden und als Eigner die benutzerdefinierte Benutzer-ID des Servers festgelegt werden. Wenn Sie beispielsweise die Benutzer-ID des Servers vom Standardwert in jdoe und das Standardprotokollverzeichnis in server_root/account ändern, muss das Verzeichnis server_root/account die Berechtigung 755 besitzen, und als Eigner muss jdoe festgelegt sein.

Wenn der Server aktiv ist, startet er täglich um 00:00 Uhr eine neue Protokolldatei. Andernfalls startet der Server eine neue Protokolldatei, wenn er zum ersten Mal am Tag gestartet wird. Beim Erstellen der Datei verwendet der Server den durch Sie angegebenen Dateinamen und hängt an diesen ein Suffix für das Datum an. Für das Datumssuffix wird das Format *TTMmmJJJJ* verwendet, wobei *Mmm* die ersten drei Buchstaben des Monatsnamens, *TT* der Tag des Monats und *JJJJ* das Jahr sind.

Format
```
ErrorLog  /Pfad/logs-Verzeichnis/Dateiname
```

Standardwerte
- **Linux- und UNIX-Systeme:** `ErrorLog /opt/ibm/edge/cp/server_root/logs/error`
- **Windows-Systeme:** `ErrorLog Laufwerk:\Programme\IBM\edge\cp\logs\error`

ErrorPage - Für eine bestimmte Fehlerbedingung eine angepasste Nachricht angeben

Legen Sie mit dieser Anweisung den Namen einer Datei fest, die an den anfragenden Client gesendet werden soll, wenn der Server eine bestimmte Fehlerbedingung feststellt. Die Konfigurationsdatei ibmproxy.conf enthält ErrorPage-Anweisungen, in denen den Fehlerschlüsselwörtern bestimmte Fehlernachrichtendateien zugeordnet sind.

Zur Anpassung von Fehlernachrichten können Sie die ErrorPage-Anweisungen in der Weise ändern, dass sie den Fehlerschlüsselwörtern andere Dateien zuordnen, oder aber Sie können die bereitgestellten Fehlernachrichtendateien ändern. Zum Beispiel kann eine Nachricht so geändert werden werden, dass sie mehr Informationen zur Fehlerursache und Vorschläge zur Fehlerbehebung enthält. In internen Netzen könnten Sie den Benutzern eine Kontaktperson angeben.

ErrorPage-Anweisungen können in der Konfigurationsdatei an beliebiger Stelle stehen. Tritt der Fehler auf, wird die Datei gemäß den in der Konfigurationsdatei definierten Zuordnungsregeln verarbeitet. Deshalb muss die Datei, die gesendet werden soll, an einer Position befinden, die durch die mit den Anweisungen Fail, Map, NameTrans, Pass, Redirect und Service definierten Zuordnungsregeln erreicht werden kann. Es wird mindestens eine Pass-Anweisung benötigt, damit der Server die Fehlernachrichtendatei übergeben kann.

Format

`ErrorPage` *`Schlüsselwort /Pfad/Dateiname.html`*

`Schlüsselwort`
Ein Schlüsselwort, das einer Fehlerbedingung zugeordnet ist. Die Schlüsselwörter werden in den ErrorPage-Anweisungen in der Datei ibmproxy.conf aufgelistet. Die Schlüsselwörter dürfen nicht geändert werden.

`/Pfad/Dateiname.html`
Der vollständig qualifizierte Webname der Fehlerdatei, wie sie sich im Web einem Client darstellt. Vorgegebene Fehlernachrichtendateien finden Sie unter /HTML/errorpages/.

Beispiel

```
ErrorPage scriptstart /HTML/errorpages/scriptstart.htmls
```

In diesem Beispiel sendet der Server, wenn die Bedingung `scriptstart` festgestellt wird, die Datei scriptstart.htmls, die sich im Verzeichnis /HTML/errorpages/ befindet, an den Client.

Der folgende HTML-Text ist ein Beispiel für den Inhalt einer solchen Datei:

```
<HTML>
<HEAD>
<TITLE>Nachricht für Bedingung SCRIPTSTART</TITLE>
</HEAD>
<BODY>
Das CGI-Programm konnte nicht gestartet werden.
<P>
<A HREF="mailto:admin@websvr.com">Benachrichtigen Sie den Administrator</A>
über dieses Problem.
</BODY>
</HTML>
```

Falls in der Konfigurationsdatei `PASS /* /wwwhome/*` die Anweisung ist, die mit dem Pfad oben übereinstimmt, wird als vollständiger Pfad für diese Nachrichtendatei `/wwwhome/HTML/errorpages/scriptstart.htmls` eingesetzt.

Vom Server zurückgegebene Fehlernachrichten anpassen

Jede Fehlerbedingung wird durch ein Schlüsselwort identifiziert. Bevor Sie entscheiden, welche Fehlernachrichten angepasst werden sollen, lassen Sie sich zuerst die mit Caching Proxy gelieferten Fehlernachrichtendateien, die sich im Verzeichnis /HTML/errorpages befinden, anzeigen. Die Fehlerseite enthält die Fehlernummer, die Standardnachricht, eine Erklärung der Ursache sowie eine geeignete Maßnahme zur Fehlerbehebung.

Führen Sie in diesem Fall einen der folgenden Schritte aus, um eine Fehlernachricht zu ändern:

- Ändern Sie die vorhandene HTML- oder HTMLS-Datei (nachdem Sie zuerst ein Sicherungskopie erstellt haben) oder erstellen Sie eine neue HTML- oder HTMLS-Datei mit dem gewünschten Text. Dazu kann ein HTML-Editor oder ein ASCII-Editor verwendet werden. Eine HTMLS-Datei muss verwendet werden, wenn Includes des Servers genutzt werden sollen.
- Falls Sie eine Fehlernachrichtendatei mit einem anderen Namen (oder in einem anderen Pfad) erstellt haben, ändern Sie die ErrorPage-Anweisung für dieses Schlüsselwort, so dass sie auf diese Datei zeigt.

Fehlerbedingungen, Fehlerursachen und Standardnachrichten

Alle Fehlerschlüsselwörter und Standardfehlernachrichtendateien sind in der Datei ibmproxy.conf im Abschnitt zur Anweisung ErrorPage aufgelistet. Die Fehlernachrichtendateien enthalten Nummer und Schlüsselwort für die Fehlernachricht, die Standardnachricht, eine Erklärung sowie eine Benutzeraktion.

Standardwerte

In der Datei ibmproxy.conf sind mehrere Standardwerte gesetzt.

Falls Sie eine ErrorPage-Anweisung für eine Fehlerbedingung nicht ändern, wird die für diese Bedingung festgelegte Standardfehlerseite des Servers gesendet.

EventLog - Den Pfad der Ereignisprotokolldatei angeben

Mit dieser Anweisung können Sie den Pfad und den Namen der Datei für das Ereignisprotokoll angeben. Im Ereignisprotokoll werden Informationsnachrichten über den Cache selbst aufgezeichnet.

Anmerkung: Wenn Sie die Standardeinstellungen des Servers für Benutzer-ID, Gruppen-ID oder Protokollverzeichnispfade ändern, müssen Sie die neuen Verzeichnisse erstellen und die für diese Verzeichnisse erforderlichen Berechtigungen und Eigentumsrechte aktualisieren. Damit der Server Daten in ein benutzerdefiniertes Protokollverzeichnis schreiben kann, muss die Berechtigung für dieses Verzeichnis auf 755 gesetzt werden und als Eigner die benutzerdefinierte Benutzer-ID des Servers festgelegt werden. Wenn Sie beispielsweise die Benutzer-ID des Servers vom Standardwert in jdoe und das Standardprotokollverzeichnis in server_root/account ändern, muss das Verzeichnis server_root/account die Berechtigung 755 besitzen, und als Eigner muss jdoe festgelegt sein.

Wenn der Server aktiv ist, startet er täglich um 00:00 Uhr eine neue Protokolldatei. Andernfalls startet der Server eine neue Protokolldatei, wenn er zum ersten Mal am Tag gestartet wird. Beim Erstellen der Datei verwendet der Server den durch Sie angegebenen Dateinamen und hängt an diesen ein Suffix für das Datum an. Für das Datumssuffix wird das Format *TTMmmJJJJ* verwendet, wobei *Mmm* die ersten drei Buchstaben des Monatsnamens, *TT* der Tag des Monats und *JJJJ* das Jahr sind.

Format

```
EventLog  /Pfad/logs-Verzeichnis/Dateiname
```

Standardwerte

- **Linux- und UNIX-Systeme:** `EventLog /opt/ibm/edge/cp/server_root/logs/event`
- **Windows-Systeme:** `EventLog Laufwerk:\Programme\IBM\edge\cp\logs\event`

Exec - Für übereinstimmende Anforderungen ein CGI-Programm ausführen

Mit dieser Anweisung können Sie eine Schablone für Anforderungen festlegen, auf die mit der Ausführung eines CGI-Programms reagiert werden soll. Wenn eine Anforderung mit einer Schablone für eine Exec-Anweisung übereinstimmt, wird diese Anforderung nicht mehr mit der Anforderungsschablone nachfolgender Anweisungen verglichen.

Format

```
Exec  Anforderungsschablone Programmpfad [Server-IP-Adresse | Hostname]
```

Anforderungsschablone
Eine Schablone für Anforderungen, die der Server akzeptieren und auf die er mit der Ausführung eines CGI-Programms reagieren soll.

Als Platzhalterzeichen muss sowohl in der `Anforderungsschablone` als auch im `Programmpfad` ein Stern (`*`) verwendet werden. Der Teil der Anforderung, der mit `Anforderungsschablone` übereinstimmt, muss mit dem Namen der Datei beginnen, die das CGI-Programm enthält.

Die Anforderung kann außerdem zusätzliche Daten enthalten, die an das CGI-Programm in der Umgebungsvariable PATH_INFO übergeben werden. Die zusätzlichen Daten befinden sich hinter dem ersten Schrägstrich (/), der in der Anforderung nach dem Namen der CGI-Programmdatei steht. Die Daten werden entsprechend den CGI-Spezifikationen übergeben.

Programmpfad
Der Pfad zu der Datei, die das CGI-Programm enthält, das der Server für die Anforderung ausführen soll. Der *Programmpfad* muss außerdem ein Platzhalterzeichen enthalten. Das Platzhalterzeichen wird durch den Namen der Datei ersetzt, die das CGI-Programm enthält.

Die Anweisung Exec ist rekursiv und gilt für alle Unterverzeichnisse. Für die Verzeichnisse unter cgi-bin und admin-bin ist keine separate Exec-Anweisung notwendig.

[*Server-IP-Adresse* | *Hostname*]

Falls Sie mehrere IP-Adressen oder virtuelle Hosts verwenden, geben Sie mit diesem Parameter eine IP-Adresse oder einen Hostnamen an. Der Server verwendet die Anweisung nur für Anforderungen an diese IP-Adresse oder für diesen Host. Bei einer IP-Adresse ist dies die Adresse der Netzverbindung des Servers und nicht die Adresse des Clients, von dem die Anforderung stammt.

Sie können eine IP-Adresse (z. B. `240.146.167.72`) oder einen Hostnamen (z. B. `hostA.bcd.com`) angeben.

Dieser Parameter ist optional. Ohne diesen Parameter verwendet der Server die Anweisung für alle Anforderungen ungeachtet der IP-Adresse, unter der die Anforderungen eingehen, oder des Hostnamens im URL.

Platzhalterzeichen dürfen zur Angabe von Server-IP-Adressen nicht verwendet werden.

Beispiele

In diesem Beispiel führt der Server beim Empfang einer Anforderung von `/idd/depts/plan/c92` das CGI-Programm in /depts/bin/plan.exe aus, wobei `c92` als Eingabe an das Programm übergeben wird.

Im folgenden Beispiel wird der optionale Parameter IP-Adresse verwendet. Wenn der Server Anforderungen empfängt, die mit `/cgi-bin/`, beantwortet er die Anforderung aus einem anderen Verzeichnis, basierend auf der IP-Adresse der Netzverbindung, in der die Anforderung empfangen wurde. Für Anforderungen, die an der Adresse 130.146.167.72 empfangen werden, verwendet der Server das Verzeichnis /CGI-BIN/customerA. Für Anforderungen, die über eine Verbindung zur Adresse 0.83.100.45 empfangen werden, verwendet der Server das Verzeichnis /CGI-BIN/customerB.

```
Exec    /cgi-bin/*    /CGI-BIN/customerA/*   130.129.167.72
Exec    /cgi-bin/*    /CGI-BIN/customerB/*   0.83.100.45
```

Im folgenden Beispiel wird der optionale Parameter Hostname verwendet. Wenn der Server Anforderungen empfängt, die mit `/cgi-bin` beginnen, beantwortet er die Anforderung aus einem anderen Verzeichnis, basierend auf dem Hostnamen im URL. Für Anforderungen, die für hostA.bcd.com empfangen werden, verwendet der Server das Verzeichnis /CGI-BIN/customerA. Bei für hostB.bcd.com eingehenden Anforderungen verwendet der Server das Verzeichnis /CGI-BIN/customerB.

```
Exec /cgi-bin/* /CGI-BIN/customerA/* hostA.bcd.com
Exec /cgi-bin/* /CGI-BIN/customerB/* hostB.bcd.com
```

Standardwerte

- **Linux- und UNIX-Systeme**

  ```
  Exec  /cgi-bin/*      /opt/ibm/edge/cp/server_root/cgi-bin/*
  Exec  /admin-bin/*    /opt/ibm/edge/cp/server_root/admin-bin/*
  ```

- **Windows-Systeme**

  ```
  Exec  server_root/cgi-bin/*
  Exec  server_root/admin-bin/*
  Exec  server_root/DOCS/admin-bin/*
  ```

ExportCacheImageTo - Cache-Speicher auf die Platte exportieren

Mit dieser Anweisung können Sie den Cache-Inhalt in eine Speicherauszugsdatei exportieren. Dies ist nützlich, wenn bei einem Neustart Cache-Speicher verloren geht oder wenn derselbe Cache für mehrere Proxys implementiert wird.

Format

```
ExportCacheImageTo Name_der_Exportdatei
```

Standardwert

Keiner.

ExternalCacheManager - Caching Proxy für dynamisches Caching vom IBM WebSphere Application Server konfigurieren

Mit dieser Anweisung können Sie den Caching-Proxy-Server so konfigurieren, dass er einen IBM WebSphere Application Server (der mit einem Caching-Proxy-Adaptermodul konfiguriert ist) erkennt, von dem er dynamisch erstellte Ressourcen in den Cache abrufen kann. Caching Proxy speichert Kopien der JSP-Ergebnisse, die auch im dynamischen Cache von Application Server gespeichert sind. Caching Proxy speichert nur Inhalte eines IBM WebSphere Application Server im Cache, dessen Gruppen-ID mit einem ExternalCacheManager-Eintrag übereinstimmt.

Beachten Sie, dass Sie außerdem eine Service-Anweisung zur Konfigurationsdatei von Caching Proxy hinzufügen müssen, um diese Funktion zu aktivieren. Außerdem sind im Anwendungsserver zusätzliche Konfigurationsschritte erforderlich. Nähere Informationen finden Sie in Kapitel 22, „Caching von dynamisch erstelltem Inhalt", auf Seite 105.

Format

```
ExternalCacheManager ID_des_externen_Cache-Managers  Maximale_Verfallszeit
```

ID_des_externen_Cache_Managers
: Die ID, die dem IBM WebSphere Application Server zugeordnet ist, der den Proxy-Server bedient. Diese ID muss mit der ID übereinstimmen, die im Attribut `externalCacheGroup: group id` in der Datei dynacache.xml des Anwendungsservers definiert ist.

Maximale_Verfallszeit
: Die Standardverfallszeit, die für Ressourcen festgelegt ist, die auf Anweisung des externen Cache-Managers in den Cache abgerufen werden. Wird eine im Cache gespeicherte Ressource vom externen Cache-Manager innerhalb der angegebenen Zeit nicht ungültig gemacht, läuft ihre Lebensdauer nach der festgelegten Verfallszeit ab. Die Zeit kann in Minuten oder Sekunden angegeben werden.

Beispiel

Der folgende Eintrag definiert einen externen Cache-Manager (einen IBM WebSphere Application Server) in der Domäne www.xyz.com, dessen Ressourcen nach maximal 20 Sekunden verfallen.

```
ExternalCacheManager   IBM-CP-XYZ-1  20 seconds
```

Standardwert

Keiner.

Fail - Übereinstimmende Anforderungen zurückweisen

Mit dieser Anweisung können Sie für Anforderungen, die der Server nicht verarbeiten soll, eine Schablone festlegen. Wenn eine Anforderung mit einer Schablone für eine Fail-Anweisung übereinstimmt, wird diese Anforderung nicht mehr mit der Anforderungsschablone nachfolgender Anweisungen verglichen.

Format

```
Fail Anforderungsschablone [Server-IP-Adresse | Hostname]
```

Anforderungsschablone
Eine Schablone für Anforderungen, die der Server zurückweisen soll. Falls eine Anforderung mit der Schablone übereinstimmt, sendet der Server dem Requester eine Fehlernachricht.

In der Schablone kann als Platzhalterzeichen der Stern verwendet werden. Für das Tilde-Zeichen (~) direkt nach dem Schrägstrich (/) muss eine exakte Übereinstimmung bestehen. Ein Platzhalterzeichen wird nicht als Übereinstimmung gewertet.

[*Server-IP-Adresse* | *Hostname*]
Falls Sie mehrere IP-Adressen oder virtuelle Hosts verwenden, geben Sie mit diesem Parameter eine IP-Adresse oder einen Hostnamen an. Der Server verwendet die Anweisung nur für Anforderungen an diese IP-Adresse oder für diesen Host. Bei einer IP-Adresse ist dies die Adresse der Netzverbindung des Servers und nicht die Adresse des Clients, von dem die Anforderung stammt.

Sie können eine IP-Adresse (z. B. `240.146.167.72`) oder einen Hostnamen (z. B. `hostA.bcd.com`) angeben.

Dieser Parameter ist optional. Ohne diesen Parameter verwendet der Server die Anweisung für alle Anforderungen ungeachtet der IP-Adresse, unter der die Anforderungen eingehen, oder des Hostnamens im URL.

Für die IP-Adresse eines Servers kann kein Platzhalterzeichen verwendet werden.

Beispiele

Im folgenden Beispiel weist der Server alle Anforderungen zurück, die mit `/usr/local/private/` beginnen.

```
Fail /usr/local/private/*
```

Die folgenden Beispiele verwenden den optionalen Parameter IP-Adresse. Der Server weist alle Anforderungen zurück, die mit `/customerB/` beginnen, wenn diese in einer Netzverbindung mit der IP-Adresse 240.146.167.72 empfangen werden. Der Server weist alle Anforderungen zurück, die mit `/customerA/` beginnen, wenn diese an einer Netzverbindung mit der IP-Adresse 0.83.100.45 empfangen werden.

```
Fail    /customerB/*    240.146.167.72
Fail    /customerA/*    0.83.100.45
```

In den folgenden Beispielen wird der optionale Parameter Hostname verwendet. Der Server weist alle Anforderungen zurück, die mit `/customerB/` beginnen, wenn diese für hostA.bcd.com empfangen werden. Der Server weist alle Anforderungen zurück, die mit `/customerA/` beginnen, wenn diese für hostB.bcd.com empfangen werden.

```
Fail /customerB/* hostA.bcd.com
Fail /customerA/* hostB.bcd.com
```

Standardwert

Keiner.

flexibleSocks - Flexible SOCKS-Implementierung aktivieren

Mit dieser Anweisung können Sie den Proxy-Server anweisen, die Art der Verbindung, die hergestellt werden soll, anhand der SOCKS-Konfigurationsdatei zu bestimmen.

Format

```
flexibleSocks {on | off}
```

Standardwert

```
flexibleSocks on
```

FTPDirInfo - Für ein Verzeichnis eine Begrüßungs- oder eine beschreibende Nachricht generieren

Mit dieser Anweisung können Sie festlegen, dass FTP-Server für ein Verzeichnis eine Begrüßungs- oder eine beschreibende Nachricht generieren sollen. Diese Nachricht kann wahlweise als Bestandteil von FTP-Auflistungen angezeigt werden. Mit der Anweisung FTPDirInfo kann auch gesteuert werden, wo diese Nachricht angezeigt wird.

Format

```
FTPDirInfo  {top | bottom | off}
```

top
Die Begrüßungsnachricht am Seitenanfang vor der Auflistung der Dateien im Verzeichnis anzeigen.

bottom
Die Begrüßungsnachricht am Seitenende nach der Auflistung der Dateien im Verzeichnis anzeigen.

off
Die Begrüßungsseite nicht anzeigen.

Standardwert

```
FTPDirInfo top
```

ftp_proxy - Für FTP-Anforderungen einen anderen Proxy-Server angeben

Falls Ihr Proxy-Server Teil einer Kette von Proxy-Servern ist, geben Sie mit dieser Anweisung den Namen eines anderen Proxy-Servers an, den dieser Server für FTP-Anforderungen kontaktieren soll. Sie müssen einen vollständigen URL einschließlich des nachgestellten Schrägstrichs (/) angeben. Nähere Informationen zur Verwendung eines optionalen Domänennamens oder einer Schablone finden Sie im Abschnitt „no_proxy - Schablonen für Direktverbindungen zu Domänen angeben" auf Seite 245.

Format

```
ftp_proxy vollständiger_URL [Domänenname_oder_Schablone]
```

Beispiel

```
ftp_proxy http://outer.proxy.server/
```

Standardwert

Keiner.

FTPUrlPath - Angeben, wie FTP-URLs interpretiert werden sollen

Mit dieser Anweisung können Sie angeben, ob die Pfadinformationen in den FTP-URLs relativ zum Arbeitsverzeichnis des angemeldeten Benutzers oder als relativ zum Stammverzeichnis interpretiert werden sollen.

Format

```
FTPUrlPath  {relative | absolute}
```

Wird die Anweisung FTPUrlPath auf `absolute` gesetzt, muss FTP-Arbeitsverzeichnis des angemeldeten Benutzers in den FTP-URL-Pfad aufgenommen werden. Wird `FTPUrlPath Relative` angegeben, darf das FTP-Arbeitsverzeichnis des angemeldeten Benutzers nicht im FTP-URL-Pfad angegeben werden. Beispielsweise müssen abhängig von der Einstellung der Anweisung FTPUrlPath folgende URL-Pfade angegeben werden, um auf die Datei test1.html im Arbeitsverzeichnis /export/home/user1 des angemeldeten Benutzers zuzugreifen:

- Ist als Einstellung `FTPUrlPath absolute` festgelegt, ist folgender URL-Pfad erforderlich: `ftp://ftphost/export/home/user1/test1.html`.
- Ist als Einstellung `FTPUrlPath relative` festgelegt, ist folgender URL-Pfad erforderlich: `ftp://ftphost/test1.html`.

Standardwert

Keiner.

Gc - Garbage-Collection angeben

Geben Sie mit dieser Anweisung an, ob die Garbage-Collection verwendet werden soll. Ist das Caching aktiviert, verwendet der Server den Prozess der Garbage-Collection zum Löschen von Dateien, die nicht mehr im Cache gespeichert werden dürfen. Die Dateien werden auf Basis ihres Verfallsdatums und anderer Werte von Proxy-Anweisungen gelöscht. Im Allgemeinen wird die Garbage-Collection durchgeführt, wenn das Caching aktiviert ist. Wird keine Garbage-Collection verwendet, wird der Proxy-Cache nicht effizient genutzt.

Format

```
Gc {on | off}
```

Standardwert

```
Gc On
```

GCAdvisor - Prozess der Garbage-Collection anpassen

Mit dieser Anweisung können Sie eine angepasste Anwendung angeben, die der Server für die Garbage-Collection verwenden soll.

Format

`GCAdvisor` */Pfad/Datei*`:`*Funktionsname*

/Pfad/Datei
: Der vollständig qualifizierte Dateiname des kompilierten Programms einschließlich der Erweiterung.

Funktionsname
: Der Name der Anwendungsfunktion in Ihrem Programm.

Beispiel

```
GCAdvisor /api/bin/customadvise.so:gcadv
```

GcHighWater - Den Beginn der Garbage-Collection festlegen

Legen Sie mit dieser Anweisung den Prozentsatz der Gesamtkapazität des Cache fest, der erreicht werden muss, damit die Garbage-Collection ausgelöst wird. Dieser Prozentsatz wird als *oberer Grenzwert* bezeichnet. Dieser obere Grenzwert wird als Prozentsatz der gesamten Kapazität des Cache angegeben. Die Garbage-Collec-

tion wird so lange fortgesetzt, bis der untere Grenzwert erreicht ist. Informationen zu dieser Einstellung finden Sie im Abschnitt „GcLowWater - Das Ende der Garbage-Collection festlegen". Der Prozentsatz für den oberen Grenzwert kann auf eine Zahl zwischen 50 und 95 gesetzt werden.

Format

```
GcHighWater Prozentsatz
```

Standardwert

```
GcHighWater 90
```

GcLowWater - Das Ende der Garbage-Collection festlegen

Legen Sie mit dieser Anweisung den Prozentsatz der Gesamtkapazität des Cache fest, der erreicht werden muss, damit der Prozess der Garbage-Collection beendet wird. Dieser Prozentsatz wird als *unterer Grenzwert* bezeichnet. Dieser untere Grenzwert wird als Prozentsatz der gesamten Kapazität des Cache angegeben. Er muss auf einen niedrigeren Wert als der obere Grenzwert gesetzt werden. Informationen zum oberen Grenzwert enthält der Abschnitt „GcHighWater - Den Beginn der Garbage-Collection festlegen" auf Seite 222.

Format

```
GcLowWater Prozentsatz
```

Standardwert

```
GcLowWater 60
```

gopher_proxy - Für Gopher-Anforderungen einen anderen Proxy-Server angeben

Falls Ihr Proxy-Server Teil einer Kette von Proxy-Servern ist, geben Sie mit dieser Anweisung den Namen eines anderen Proxy-Servers an, den dieser Server für Gopher-Anforderungen kontaktieren soll. Sie müssen einen vollständigen URL einschließlich des nachgestellten Schrägstrichs (/) angeben. Nähere Informationen zur Verwendung eines optionalen Domänennamens oder einer Schablone finden Sie im Abschnitt „no_proxy - Schablonen für Direktverbindungen zu Domänen angeben" auf Seite 245.

Format

```
gopher_proxy vollständiger_URL[Domänenname_oder_Schablone]
```

Beispiel

```
gopher_proxy http://outer.proxy.server/
```

Standardwert

Keiner.

GroupId - Gruppen-ID angeben

Mit dieser Anweisung können Sie den Gruppennamen oder die Gruppennummern angeben, die der Server annehmen soll, bevor er auf Dateien zugreift.

Wenn Sie diese Anweisung ändern, müssen Sie den Server manuell stoppen und anschließend erneut starten, damit die Änderung wirksam wird. Die Änderung wird erst wirksam, wenn Sie den Server erneut starten. (Nähere Informationen hierzu finden Sie in Kapitel 5, „Caching Proxy starten und stoppen", auf Seite 15.)

Anmerkung: Wenn Sie die Standardeinstellungen des Servers für Benutzer-ID, Gruppen-ID oder Protokollverzeichnispfade ändern, müssen Sie die neuen Verzeichnisse erstellen und die für diese Verzeichnisse erforderlichen Berechtigungen und Eigentumsrechte aktualisieren. Damit der Server Daten in ein benutzerdefiniertes Protokollverzeichnis schreiben kann, muss die Berechtigung für dieses Verzeichnis auf 755 gesetzt werden und als Eigner die benutzerdefinierte Benutzer-ID des Servers festgelegt werden. Wenn Sie beispielsweise die Benutzer-ID des Servers vom Standardwert in jdoe und das Standardprotokollverzeichnis in server_root/account ändern, muss das Verzeichnis server_root/account die Berechtigung 755 besitzen, und als Eigner muss jdoe festgelegt sein.

Format

```
GroupId { Gruppenname | Gruppennummer}
```

Standardwerte

AIX: `GroupId nobody`

HP-UX: `GroupId other`

Linux:

Red Hat: `GroupId nobody`

SuSE: `GroupId nogroup`

Solaris: `GroupId nobody`

HeaderServerName - Namen des Proxy-Servers angeben, der im HTTP-Header zurückgegeben wird

Mit dieser Anweisung können Sie den Namen des Proxy-Servers angeben, der im HTTP-Header zurückgegeben wird.

Format

```
HeaderServerName Name
```

Standardwert

Keiner.

Hostname - Den vollständig qualifizierten Domänennamen oder die IP-Adresse für den Server angeben

Mit dieser Anweisung können Sie den Domänennamen oder eine IP-Adresse angeben, die für Dateianforderungen an die Clients zurückgegeben werden soll. Wird ein Domänenname angegeben, muss ein Domänennamensserver in der Lage sein, den Namen in eine IP-Adresse aufzulösen. Bei Angabe einer IP-Adresse wird der Domänennamensserver nicht benötigt.

Anmerkung: Wird ein Bereich eingerichtet, muss die Anweisung Hostname für alle Member des Bereichs mit demselben Wert konfiguriert werden.

Format

```
Hostname {Name | IP-Adresse}
```

Standardwert

Diese Anweisung ist in der ursprünglichen Konfigurationsdatei standardmäßig nicht angegeben. Falls Sie diese Anweisung in der Konfigurationsdatei nicht angeben, wird standardmäßig der Hostname verwendet, der im Domänennamensserver definiert ist.

http_proxy - Für HTTP-Anforderungen einen anderen Proxy-Server angeben

Falls Ihr Proxy-Server Teil einer Kette von Proxy-Servern ist, geben Sie mit dieser Anweisung den Namen eines anderen Proxy-Servers an, den dieser Server für HTTP-Anforderungen kontaktieren soll. Sie müssen einen vollständigen URL einschließlich des nachgestellten Schrägstrichs (/) angeben. Nähere Informationen zur Verwendung eines optionalen Domänennamens oder einer Schablone finden Sie im Abschnitt „no_proxy - Schablonen für Direktverbindungen zu Domänen angeben" auf Seite 245.

Format

`http_proxy` *`vollständiger_URL`*[*`Domänenname_oder_Schablone`*]

Beispiel

```
http://outer.proxy.server/
```

Standardwert

Keiner.

HTTPSCheckRoot - HTTPS-Anforderungen filtern

Geben Sie mit dieser Anweisung an, ob Caching Proxy die nicht geschützte Homepage für den URL abruft und versucht, Kennsätze darin zu finden. Werden Kennsätze gefunden, werden sie auf die sichere Anforderung angewendet.

Beispiel: Bei Anforderung von `https://www.ibm.com/` ruft Caching Proxy den URL http://www.ibm.com/ ab, durchsucht ihn nach Kennsätzen und verwendet die gefundenen Kennsätze als Filter für https://www.ibm.com/.

Ist HTTPSCheckRoot auf `off` gesetzt, ruft Caching Proxy keine ungeschützten Homepage ab, um sie nach Kennsätzen zu durchsuchen.

Format

```
HTTPSCheckRoot {on | off}
```

Standardwert

```
HTTPSCheckRoot  on
```

ICP_Address — IP-Adresse für ICP-Abfragen angeben

Legen Sie mit dieser untergeordneten Anweisung eine IP-Adresse fest, die zum Senden und Empfangen von ICP-Abfragen verwendet wird. Die Anweisung muss in die Anweisungen **<MODULEBEGIN> ICP** und **<MODULEEND>** eingeschlossen werden.

Format

`ICP_Address` *`IP-Adresse`*

Standardwert

Diese Anweisung ist in der ursprünglichen Konfigurationsdatei standardmäßig nicht angegeben. Falls Sie diese Anweisung in der Konfigurationsdatei nicht angeben, werden ICP-Abfragen standardmäßig akzeptiert und an jede Schnittstelle gesendet.

ICP_MaxThreads - Maximale Anzahl Threads für ICP-Abfragen angeben

Legen Sie mit dieser untergeordneten Anweisung die Anzahl der Threads fest, die zum Empfangen von ICP-Abfragen generiert wurden. Die Anweisung muss in die Anweisungen **<MODULEBEGIN> ICP** und **<MODULEEND>** eingeschlossen werden.

Anmerkung: Unter Redhat Linux 6.2 und niedrigeren Versionen müssen Sie für diese Anweisung eine niedrige Zahl angeben, weil die maximale Anzahl Threads, die pro Prozess erstellt werden kann, klein ist. Wird eine große Anzahl Threads für die ICP-Verwendung angegeben, steht für die Beantwortung von Anforderungen eventuell nur eine beschränkte Anzahl Threads zur Verfügung.

Format

```
ICP_MaxThreads Anzahl_Threads
```

Standardwert

```
ICP_MaxThreads   5
```

Occupier - Member eines ICP-Cluster angeben

Ist der Proxy-Server Teil eines ICP-Cluster, geben Sie mit dieser untergeordneten Anweisung die ICP-Peers an. Die Anweisung muss in die Anweisungen **<MODULEBEGIN> ICP** und **<MODULEEND>** eingeschlossen werden.

Wird ein neuer Peer zum ICP-Cluster hinzugefügt, müssen die Informationen zum ICP-Peer zur Konfigurationsdatei aller vorhandenen Peers hinzugefügt werden. Geben Sie jeden Peer in einer separaten Zeile an. Beachten Sie, dass der aktuelle Host ebenfalls in die Peer-Liste aufgenommen werden kann. Beim Initialisieren ignoriert ICP den Eintrag des aktuellen Host. Daher können Sie eine einzelne Konfigurationsdatei verwenden und diese auf andere Peer-Maschinen kopieren, ohne dass es erforderlich wäre, diese Datei zu editieren und den aktuellen Host aus der Datei zu löschen.

Format

```
ICP_Peer Hostname HTTP-Port ICP-Port
```

Hostname
Der Name des Peer.

HTTP-Port
Der Proxy-Port des Peer.

ICP-Port
Der ICP-Server-Port des Peer.

Beispiel

Mit der folgenden Zeile wird der Host abc.xcompany.com mit dem Proxy-Port 80 und dem ICP-Port 3128 als Peer hinzugefügt.

```
ICP_Peer  abc.xcompany.com  80  3128
```

Standardwert
Keiner.

ICP_Port — Port-Nummer für ICP-Abfragen angeben

Legen Sie mit dieser untergeordneten Anweisung die Port-Nummer fest, an der der ICP-Server ICP-Anforderungen empfängt. Die Anweisung muss in die Anweisungen **<MODULEBEGIN> ICP** und **<MODULEEND>** eingeschlossen werden.

Format
```
ICP_Port Port-Nummer
```

Standardwert
```
ICP_Port 3128
```

ICP_Timeout — Maximale Wartezeit für ICP-Abfragen angeben

Legen Sie mit dieser untergeordneten Anweisung die maximale Zeit fest, während der Caching Proxy auf Antworten auf ICP-Abfragen warten soll. Die Zeit wird in Millisekunden angegeben. Die Anweisung muss in die Anweisungen **<MODULEBEGIN> ICP** und **<MODULEEND>** eingeschlossen werden.

Format
```
ICP_Timeout  Zeitlimit_in_Millisekunden
```

Standardwert
```
ICP_Timeout  2000
```

IgnoreURL - URLs angeben, die nicht aktualisiert werden sollen

Mit dieser Anweisung können Sie URLs angeben, die vom Cache-Agenten nicht geladen werden sollen. Diese Anweisung ist nützlich, wenn der Cache-Agent Seiten lädt, die über Links mit den URLs im Cache verknüpft sind. Sie können die Anweisung IgnoreURL mehrfach angeben, um unterschiedliche URLs oder URL-Masken festzulegen. Der Wert für diese Anweisung kann Sterne (*) als Platzhalterzeichen enthalten und ist auf diese Weise als Maske anwendbar.

Format
```
IgnoreURL URL
```

Beispiele
```
IgnoreURL http://www.yahoo.com/
IgnoreURL http://*.ibm.com/*
```

Standardwert
```
IgnoreURL */cgi-bin/*
```

imbeds - Angeben, ob Server-Side Includes verarbeitet werden

Mit dieser Anweisung können Sie angeben, ob Server-Side Includes für Dateien, die vom Dateisystem und/oder von CGI-Programmen bereitgestellt werden, verarbeitet werden sollen. Die Verarbeitung von Server-Side Includes erfolgt für Dateien mit dem Inhaltstyp `ext/x-ssi-html`. Sie können wahlweise festlegen, dass die Verarbeitung von Server-Side Includes auch für Dateien mit dem Inhaltstyp `text/html` erfolgen soll. Nähere Informationen zu den Inhaltstypen finden Sie im Abschnitt „AddType - Datentyp für Dateien mit bestimmten Suffixen angeben” auf Seite 184.

Darüber kann die Verarbeitung von Server-Side Includes auch dazu verwendet werden, Informationen dynamisch in die Datei einzufügen, die zurückgegeben wird. Dazu gehören Informationen wie das Datum, die Größe einer Datei, das Datum der letzten Änderung einer Datei, Umgebungsvariablen für CGI oder Server-Side Includes oder Textdokumente. Die Verarbeitung von Server-Side Includes wird nur für Dateien durchgeführt, die lokal erstellt wurden. Caching Proxy führt keine Verarbeitung von Server-Side Includes für Proxy- oder Cache-Objekte durch.

Bei der Verarbeitung von Server-Side Includes durchsucht der Server die Dateien jedesmal, wenn sie ihm bereitgestellt werden, auf spezielle Befehle. Dies kann die Leistung des Servers beeinflussen und die Antwortzeit gegenüber Clients erhöhen.

Format

```
imbeds {on | off | files | cgi | noexec}  {SSIOnly | html}
```

on
Die Verarbeitung von Server-Side Includes erfolgt für Dateien aus dem Dateisystem oder für Dateien von CGI-Programmen.

off
Die Verarbeitung von Server-Side Includes findet für keine Dateien statt.

files
Die Verarbeitung von Server-Side Includes erfolgt ausschließlich für Dateien aus dem Dateisystem.

cgi
Die Verarbeitung von Server-Side Includes erfolgt ausschließlich für Dateien, die von CGI-Programmen zurückgegeben wurden.

noexec

SSIOnly
Die Verarbeitung von Server-Side Includes erfolgt für Dateien mit dem Inhaltstyp `text/x-ssi-html`.

html
Die Verarbeitung von Server-Side Includes erfolgt für Dateien mit dem Inhaltstyp text/html und dem Inhaltstyp `text/x-ssi-html`.

Der Server überprüft den Inhaltstyp jeder empfangenen Datei und die Ausgabe jedes abgearbeiteten CGI-Programms.

Die Verarbeitung von Server-Side Includes erfolgt normalerweise nur für Dateien mit dem Inhaltstyp `text/x-ssi-html`. Allerdings können Sie festlegen, dass Dateien mit dem Inhaltstyp `text/html` für Server-Side Includes verarbeitet werden sollen.

Anmerkung: Der Server behandelt html, .html und .htm als `html`. Jeder andere Wert wird als `SSIOnly` behandelt.

Jeder Suffix muss eine AddType-Anweisung mit dem richtigen Inhaltstyp besitzen. Bei Verwendung anderer Suffixe als .htm oder .html sollten Sie sicherstellen, dass eine AddType-Anweisung mit dem Inhaltstyp `text/x-ssi/html` definiert wurde.

Standardwert

```
imbeds on SSIOnly
```

ImportCacheImageFrom - Cache-Speicher aus einer Datei importieren

Mit dieser Anweisung können Sie den Cache-Inhalt aus einer Speicherauszugsdatei importieren. Dies ist nützlich, wenn bei einem Neustart Cache-Speicher verloren geht oder wenn derselbe Cache für mehrere Proxys implementiert wird.

Format

```
ImportCacheImageFrom Name_der_Importdatei
```

Standardwert

Keiner.

InheritEnv - Angeben, welche Umgebungsvariablen durch CGI-Programme übernommen werden sollen

Mit dieser Anweisung können Sie die Umgebungsvariablen angeben, die Ihre CGI-Programme übernehmen sollen (andere Variablen als die CGI-Umgebungsvariablen, die speziell für die CGI-Verarbeitung vorgesehen sind).

Wird keine InheritEnv-Anweisung verwendet, werden durch CGI-Programme alle Umgebungsvariablen übernommen. Bei Verwendung von InheritEnv-Anweisungen werden nur die in diesen InheritEnv-Anweisungen angegebenen Umgebungsvariablen gemeinsam mit den speziellen CGI-Umgebungsvariablen übernommen. Die Anweisung ermöglicht es, den Wert übernommener Variablen wahlfrei zu initialisieren.

Format

```
InheritEnv Umgebungsvariable
```

Beispiele

```
InheritEnv PATH
InheritEnv LANG=ENUS
```

In diesem Beispiel werden durch CGI-Programme nur die Umgebungsvariablen PATH und LANG übernommen. Dabei wird die Umgebungsvariable LANG mit dem Wert ENUS initialisiert.

Standardwert

Keiner. Standardmäßig werden durch CGI-Programme alle Umgebungsvariablen übernommen.

InputTimeout - Zeitlimit für Eingabe festlegen

Mit dieser Anweisung können Sie eine Zeitspanne angeben, die einem Client nach dem Aufbau einer Verbindung zum Server zugestanden wird, um eine Anforderung zu senden. Ein Client stellt zuerst die Verbindung zum Server her und sendet dann eine Anforderung. Sendet der Client innerhalb der in dieser Anweisung angegebenen Zeitspanne keine Anforderung, schließt der Server die Verbindung. Geben Sie die Zeitperiode in einer beliebigen Kombination aus Stunden, Minuten und Sekunden an.

Format

```
InputTimeout Zeit
```

Beispiel

```
InputTimeout 3 mins 30 secs
```

Standardwert

```
InputTimeout 2 minutes
```

JunctionReplaceUrlPrefix — Bei Verwendung des Plug-in JunctionRewrite den URL ersetzen, anstatt Präfix einzufügen

Diese Anweisung überschreibt die Standardaktion des Plug-in JunctionRewrite und erlaubt damit dem Proxy-Server, bestimmte URL-Links in der HTML-Seite zu korrigieren. Sie wird in Kombination mit der Anweisung JunctionRewrite verwendet.

Die Anweisung JunctionReplaceUrlPrefix weist das Plug-in JunctionRewrite an, den URL aus *URL-Muster_1* durch *URL-Muster_2* zu ersetzen, anstatt ein Präfix am Anfang des URL einzufügen.

Format

```
JunctionReplaceUrlPrefix URL-Muster_1 URL-Muster_2
```

Beispiel

```
JunctionReplaceUrlPrefix /server1.internaldomain.com/*  /server1/*
```

In diesem Beispiel wird angenommen, dass der URL /server1.internaldomain.com/notes.nsf und das Präfix /server1 ist. Anstatt das Präfix einzufügen, um den URL in /server1/server1.internaldomain.com/notes.nsf umzuschreiben, ändert das Plug-in JunctionRewrite den URL in /server1/notes.nsf.

Standardwert

Keiner.

JunctionRewrite - Umschreiben von URLs aktivieren

Diese Anweisung aktiviert die Routine für das Umschreiben von Junctions im Caching Proxy, die die Antworten von den Ursprungsservern so umschreibt, dass URLs, die relativ zum Server angegeben werden, dem richtigen Ursprungsserver zugeordnet werden, wenn Junctions verwendet werden. Das Plug-in für das Umschreiben von Junctions (JunctionRewrite) muss ebenfalls aktiviert sein, wenn **JunctionRewrite on** ohne die Option UseCookie festgelegt wird. Junctions werden durch die Proxy-Zuordnungsregeln definiert.

Nähere Informationen zu JunctionRewrite enthalten die Abschnitte „UseCookie als Alternative zu JunctionRewrite" auf Seite 50 und „Beispiel-Plug-in Transmogrifier zur Erweiterung der Funktionalität von JunctionRewrite" auf Seite 51.

Format

```
JunctionRewrite {on | on UseCookie | off}
```

Standardwert

```
JunctionRewrite off
```

JunctionRewriteSetCookiePath — Bei Verwendung des Plug-in JunctionRewrite die Pfadoption im Header Set-Cookie umschreiben

Mit dieser Anweisung kann der Proxy-Server die Pfadoption im Header Set-Cookie umschreiben, wenn eine Übereinstimmung mit dem Cookie-Namen gefunden wird. Wenn die Antwort eine Junction erfordert und ein Junction-Präfix definiert ist,

wird das Präfix vor jedem Pfad eingefügt. Die Anweisung kann für das Plug-in JunctionRewrite und zusammen mit der Anweisung RewriteSetCookieDomain verwendet werden.

Format

```
JunctionRewriteSetCookiePath Cookie-Name1 Cookie-Name2...
```

Cookie-Name
Ein Cookie-Name im Header Set-Cookie.

Standardwert

Keiner.

JunctionSkipUrlPrefix — Bei Verwendung des Plug-in JunctionRewrite das Überschreiben von URLs überspringen, die das Präfix bereits enthalten

Diese Anweisung überschreibt die Standardaktion des Plug-in JunctionRewrite und überspringt das Umschreiben von URLs, falls eine Übereinstimmung mit dem URL-Muster gefunden wird. Die Anweisung kann für das Plug-in JunctionRewrite verwendet werden und bietet die Möglichkeit, einige URL-Links in der HTML-Seite zu korrigieren. Normalerweise wird die Anweisung verwendet, um die URLs zu überspringen, die bereits ein Präfix enthalten.

Format

```
JunctionSkipUrlPrefix URL-Muster
```

Beispiel

```
JunctionSkipUrlPrefix  /server1/*
```

In diesem Beispiel wird angenommen, dass der URL /server1/notes.nsf und das Junction-Präfix /server1/ ist. Anstatt den URL in /server1/server1/notes.nsf umzuschreiben, überspringt das Plug-in JunctionRewrite das Umschreiben des URL, d. h. der URL /server1/notes.nsf wird weiterhin verwendet.

Standardwert

Keiner.

KeepExpired - Garbage-Collection für den Proxy-Cache ändern

Mit dieser Anweisung können Sie festlegen, wie Dateien mit abgelaufener Verfallszeit während der Garbage-Collection für den Proxy-Cache behandelt werden. Gültige Werte sind `on` und `off`. Wird diese Anweisung auf `on` gesetzt, werden die Dateien, deren Verfallsdatum überschritten wurde, zum Löschen markiert und während der Garbage-Collection zuerst gelöscht. Ist der Cache voll, werden die Dateien mit abgelaufener Verfallszeit zuerst gelöscht. Wird diese Anweisung auf `off` gesetzt, werden Dateien mit abgelaufener Verfallszeit während der Garbage-Collection auch dann gelöscht, wenn der Proxy-Cache nicht voll ist.

Format

```
KeepExpired {on | off}
```

Standardwert

```
KeepExpired off
```

KeyRing - Den Dateipfad zur Schlüsselringdatenbank angeben

Mit dieser Anweisung können Sie den Dateipfad der Schlüsselringdatenbank angeben, die der Server für SSL-Anforderungen verwendet. Schlüsselringdateien werden mit dem Schlüsselverwaltungsprogramm iKeyman generiert.

Anmerkung: Die SSL-Anweisungen werden unter SuSE Linux nicht unterstützt.

Format

```
KeyRing Dateiname
```

Beispiele

Windows:
```
KeyRing c:\Programme\IBM\edge\cp\\key.kdb
```

Linux und UNIX:
```
KeyRing /etc/key.kdb
```

Standardwert

Keiner.

KeyRingStash - Den Pfad zur Kennwortdatei der Schlüsselringdatenbank angeben

Mit dieser Anweisung können Sie den Dateipfad der Kennwortdatei der Schlüsselringdatenbank angeben. Die Kennwortdatei wird beim Erstellen einer Schlüsselringdatenbankdatei vom Schlüsselverwaltungsprogramm iKeyman generiert.

Anmerkung: Die SSL-Anweisungen werden unter SuSE Linux nicht unterstützt.

Format

```
KeyRingStash Dateipfad
```

Beispiele

Windows:
```
KeyRingStash c:\Programme\IBM\edge\cp\key.sth
```

Linux und UNIX:
```
KeyRingStash /etc/key.sth
```

Standardwert

Keiner.

LimitRequestBody — Maximale Größe des Hauptteils für PUT- und POST-Anforderungen festlegen

Mit dieser Anweisung können Sie die maximale Größe des Hauptteils in PUT- oder POST-Anforderungen festlegen. Die LimitRequest-
Anweisungen werden dazu verwendet, den Proxy vor Attacken zu schützen.

Der Wert kann in Kilobyte (K), Megabyte (M) und Gigabyte (G) angegeben werden.

Format

```
LimitRequestBody maximale_Größe_des_Hauptteils {K | M | G}
```

Standardwert

```
LimitRequestBody 10 M
```

LimitRequestFields - Maximale Anzahl Header in Clientanforderungen festlegen

Geben Sie mit dieser Anweisung die maximale Anzahl Header an, die in Clientanforderungen gesendet werden dürfen. Die LimitRequest-Anweisungen werden dazu verwendet, den Proxy vor Attacken zu schützen.

Format

`LimitRequestFields` *`Anzahl_Header`*

Standardwert

`LimitRequestFields 32`

LimitRequestFieldSize - Maximale Header-Länge und maximale Länge der Anforderungszeile festlegen

Legen Sie mit dieser Anweisung die maximale Länge der Anforderungszeile fest und die maximale Länge jedes Header in einer Anforderung. Die LimitRequest-Anweisungen werden dazu verwendet, den Proxy vor Attacken zu schützen.

Der Wert kann in Byte (B) und Kilobyte (K) angegeben werden.

Format

`LimitRequestFieldSize` *`maximale_Header-Länge`* `{B | K}`

Standardwert

`LimitRequestFieldSize 4096 B`

ListenBacklog - Den für den Server zulässigen Empfangsrückstand (listen-Backlog) für Clientverbindungen angeben

Geben Sie mit dieser Anweisung die zulässige Anzahl der Clientverbindungen an, die sich im Empfangsrückstand (listen-Backlog) befinden dürfen, bevor der Server an die Clients eine Nachricht über die Ablehnung ihrer Verbindungsanforderungen sendet. Diese Zahl ist abhängig von der Anzahl Anforderungen, die der Server in wenigen Sekunden verarbeiten kann. Legen Sie keinen höheren Wert fest als die Zahl, die der Server verarbeiten kann, bevor das Zeitlimit für die Verbindung vom Client überschritten wird und die Verbindung abgebrochen wird.

Anmerkung: Ist der Wert für ListenBacklog größer als der von TCP/IP unterstützte SOMAXCONN-Wert, wird stattdessen der SOMAXCONN-Wert verwendet.

Format

`ListenBacklog` *`Anzahl_Anforderungen`*

Standardwert

`ListenBacklog 128`

LoadInlineImages - Aktualisierung eingebetteter Grafiken steuern

Legen Sie mit dieser Anweisung fest, ob der Cache-Agent eingebettete Grafiken abrufen soll. Ist LoadInlineImages auf `on` gesetzt, werden Grafiken, die in einer im Cache gespeicherten Seite eingebettet sind, ebenfalls im Cache gespeichert. Ist die Anweisung auf `off` gesetzt, werden eingebettete Grafiken nicht im Cache gespeichert.

Format
`LoadInlineImages {on | off}`

Standardwert
`LoadInlineImages on`

LoadTopCached - Die Anzahl der am häufigsten angeforderten Seiten angeben, die aktualisiert werden sollen

Mit dieser Anweisung können Sie den Cache-Agenten anweisen, das Cache-Zugriffsprotokoll der vergangenen Nacht zu lesen und die am häufigsten angeforderten URLs zu laden.

Wenn Sie für die Anweisung LoadTopCached einen Wert definieren, muss die Anweisung Caching auf `On` gesetzt und für die Anweisung CacheAccessLog ein Wert festgelegt sein.

Format
`LoadTopCached` *`Anzahl_Seiten`*

Standardwert
`LoadTopCached 100`

LoadURL - Die URLs angeben, die aktualisiert werden sollen

Mit dieser Anweisung können Sie URLs angeben, die der Cache-Agent in den Cache laden soll. In der Konfigurationsdatei können mehrere LoadURL-Anweisungen enthalten sein, die Verwendung von Platzhalterzeichen ist jedoch nicht zulässig.

Format
`LoadURL` *`URL`*

Beispiel
`LoadURL http://www.ibm.com/`

Standardwert
Keiner.

Log - Schritt ″Log″ anpassen

Mit dieser Anweisung können Sie eine angepasste Anwendungsfunktion angeben, die der Server im Schritt ″Log″ aufrufen soll. Dieser Code unterstützt die Protokollierung und andere Verarbeitungsfunktionen, die nach Beendigung der Verbindung ausgeführt werden.

Format
`Log` *`Anforderungsschablone /Pfad/Datei:Funktionsname`*

`Anforderungsschablone`
Eine Schablone für Anforderungen, die festlegt, ob eine Anwendungsfunktion aufgerufen wird. Die Schablone kann als Angaben das Protokoll, die Domäne und den Host enthalten, sie darf als vorangestelltes Zeichen einen Schrägstrich (/) verwenden sowie als Platzhalterzeichen einen Stern (*). Beispielsweise sind folgende Schablonen gültig: `/front_page.html`, `http://www.ics.raleigh.ibm.com`, `/pub*`, `/*` und `*`.

/Pfad/Datei
Der vollständig qualifizierte Dateiname des kompilierten Programms einschließlich der Erweiterung.

Funktionsname
Der Name der Anwendungsfunktion in Ihrem Programm. Sie müssen die Namen der Funktionen für das Öffnen, das Schreiben und das Schließen angeben.

Beispiel

```
Log /index.html /api/bin/icsextpgm.so:log_url
```

Standardwert

Keiner.

LogArchive - Verhalten der Protokollarchivierung angeben

Legen Sie mit dieser Anweisung das Verhalten der Archivierungsroutine fest. Diese Anweisung gilt für alle Protokolle mit globalen Einstellungen. Sie legt fest, ob Protokolle komprimiert oder gelöscht werden sollen oder ob keine Aktion stattfinden soll.

Bei Angabe von `Compress` legen Sie mit den Anweisungen CompressAge und CompressDeleteAge fest, wann die Protokolle komprimiert oder gelöscht werden sollen. Geben Sie mit der Anweisung CompressCommand den Befehl und die zugehörigen Parameter an, die verwendet werden sollen.

Bei Angabe von `Purge` legen Sie mit den Anweisungen PurgeAge und PurgeSize fest, wann die Protokolle gelöscht werden sollen.

Format

```
LogArchive {Compress | Purge | none}
```

Compress
Legt fest, dass die Archivierungsroutine die Protokolle komprimieren soll.

Purge
Legt fest, dass die Archivierungsroutine die Protokolle löschen soll.

none
Legt fest, dass die Archivierungsroutine keine Aktion ausführen soll.

Standardwert

```
LogArchive Purge
```

Zugehörige Anweisungen

- „CompressAge - Zeitpunkt für die Komprimierung von Protokollen angeben" auf Seite 201
- „CompressDeleteAge - Zeitpunkt zum Löschen von Protokollen angeben" auf Seite 202
- „CompressCommand - Komprimierungsbefehl und zugehörige Parameter angeben" auf Seite 201
- „Midnight - API-Plug-in für die Archivierung von Protokollen angeben" auf Seite 242
- „PurgeAge - Altersgrenze für ein Protokoll angeben" auf Seite 267
- „PurgeSize - Begrenzung für die Größe des Protokollarchivs angeben" auf Seite 267

LogFileFormat — Format des Zugriffsprotokolls angeben

Mit dieser Anweisung können Sie das Dateiformat für die Zugriffsprotokolldateien angeben.

Format

```
LogFileFormat  {common | combined}
```

Standardmäßig werden Protokolle im NCSA Common Log Format angezeigt. Geben Sie `combined` an, um Protokolle stattdessen im ″NCSA Combined Log Format″ anzuzeigen. In diesem Format werden die Felder Referring URL, User Agent und Cookie hinzugefügt (soweit in der Anforderung vorhanden).

Standardwert

```
LogFileFormat common
```

LogToGUI (nur Windows) - Nur Protokolleinträge im Serverfenster anzeigen

Nur für Windows-Systeme: Wenn Sie den Proxy-Server in der Befehlszeile ausführen, können Sie mit dieser Anweisung die Ausgabe in das Zugriffsprotokoll umleiten. Zur Optimierung der Serverleistung ist diese Anweisung standardmäßig auf `off` (inaktiviert) gesetzt.

Anmerkung: Diese Anweisung hat keine Auswirkungen, wenn der Proxy als Dienst ausgeführt wird.

Format

```
LogToGUI  {on | off}
```

Standardwert

```
LogToGUI off
```

LogToSyslog - Angeben, ob Zugriffsinformationen an das Systemprotokoll gesendet werden sollen (nur Linux und UNIX)

Diese Anweisung ist nur für **Linux- und UNIX-Systeme gültig**. Mit dieser Anweisung können Sie angeben, ob der Server Zugriffsanforderungen und Zugriffsfehler nicht nur in den Zugriffs- und den Fehlerprotokolldateien, sondern zusätzlich auch im Systemprotokoll protokollieren soll.

Format

```
LogToSyslog {on | off}
```

Die Systemprotokolldatei muss sich auf Ihrem Server befinden, bevor Sie festlegen, dass die Fehlerprotokollinformationen in diese Datei geschrieben werden sollen. Sie können wählen, ob Zugriffs- oder Fehlerinformationen oder beide Arten von Informationen protokolliert werden sollen.

Damit nur Fehlerinformationen an das Systemprotokoll gesendet werden, fügen Sie folgende Zeile zur Datei /etc/syslog.conf hinzu:

```
user.err syslog-Ausgabedatei_für_Fehlerinformationen
```

Damit nur Zugriffsinformationen an das Systemprotokoll gesendet werden, fügen Sie folgende Zeile zur Datei /etc/syslog.conf hinzu:

```
user.info syslog-Infodatei_für_Zugriffsinformationen
```

Damit sowohl Fehler- als auch Zugriffsinformationen an das Systemprotokoll gesendet werden, fügen Sie beide Zeilen zur Datei /etc/syslog.conf hinzu:

Geben Sie die *syslog-Ausgabedatei* und die *syslog-Infodatei* im folgenden Format an:

- **AIX:** `/var/adm/Name_der_syslog-Datei`
- **HP-UX:** `/var/adm/syslog/syslog.log`
- **Linux:** `/var/adm/messages`
- **Solaris:** `/var/adm/messages`

Nachdem Sie die Systemprotokolldatei erstellt haben, können Sie sie mit dem folgenden Befehl neu starten:

```
kill -HUP 'cat /etc/syslog.pid'
```

Standardwert

```
LogToSyslog Off
```

Map - Übereinstimmende Anforderungen in eine neue Anforderungszeichenfolge ändern

Mit dieser Anweisung können Sie die Anforderungen, die in eine neue Anforderungszeichenfolge geändert werden sollen, eine Schablone angeben. Nachdem der Server die Anforderung geändert hat, gleicht er die neue Anforderungszeichenfolge mit den Anforderungsschablonen nachfolgender Anweisungen ab.

Format

`Map` *`Anforderungsschablone neue_Anforderung`* [*`Server-IP-Adresse`* | *`Hostname`*]

`Anforderungsschablone`
Eine Schablone für die Anforderungen, die der Server ändert und anschließend mit anderen Schablonen abgleicht.

In der Schablone kann ein Stern (*) als Platzhalterzeichen verwendet werden. Für das Tilde-Zeichen (~) direkt nach dem Schrägstrich (/) muss eine exakte Übereinstimmung bestehen. Ein Platzhalterzeichen wird nicht als Übereinstimmung gewertet.

`neue_Anforderung`
Die neue Anforderungszeichenfolge, die der Server für den Abgleich mit den Anforderungsschablonen nachfolgender Anweisungen verwendet. Die als *`neue_Anforderung`* festgelegte Zeichenfolge darf ein Platzhalterzeichen enthalten, wenn die *`Anforderungsschablone`* ebenfalls ein Platzhalterzeichen verwendet. Der Teil der Anforderung, der mit dem Platzhalterzeichen in der *`Anforderungsschablone`* übereinstimmt, wird an Stelle des Platzhalterzeichens in der *`neuen_Anforderung`* eingesetzt.

[*`Server-IP-Adresse | Hostname`*]
Falls Sie mehrere IP-Adressen oder virtuelle Hosts verwenden, geben Sie mit diesem Parameter eine IP-Adresse oder einen Hostnamen an. Der Server verwendet die Anweisung nur für Anforderungen an diese IP-Adresse oder für diesen Host. Bei einer IP-Adresse ist dies die Adresse der Netzverbindung des Servers und nicht die Adresse des Clients, von dem die Anforderung stammt.

Sie können eine IP-Adresse (z. B. `240.146.167.72`) oder einen Hostnamen (z. B. `hostA.raleigh.ibm.com`) angeben.

Dieser Parameter ist optional. Ohne diesen Parameter verwendet der Server die Anweisung für alle Anforderungen ungeachtet der IP-Adresse, unter der die Anforderungen eingehen, oder des Hostnamens im URL.

Für die IP-Adresse eines Servers kann kein Platzhalterzeichen verwendet werden.

Beispiele

- Im folgenden Beispiel ändert der Server in jeder Anforderung, die mit `/stuff/` beginnt, die Zeichenfolge `/stuff/` der Anforderung in `/good/stuff/`. Alle Angaben nach `/stuff/` in der ursprünglichen Anforderung werden in die neue Anforderungszeichenfolge übernommen. Daher wird z. B. `/stuff/whatsup/` in `/good/stuff/whatsup/` geändert. Der Server verwendet die neue Anforderungszeichenfolge und gleicht sie mit den Anforderungsschablonen der nachfolgenden Anweisungen ab.

  ```
  Map   /stuff/*   /good/stuff/*
  ```

- Die folgenden Beispiele verwenden den optionalen Parameter IP-Adresse. Wenn der Server Anforderungen empfängt, die mit `/stuff/` beginnen, ändert er die Anforderung auf der Basis der IP-Adresse der Netzverbindung, in der die Anforderung empfangen wurde, in eine andere Anforderungszeichenfolge. Bei Anforderungen, die an der Adresse 240.146.167.72 empfangen werden, ändert der Server die Zeichenfolge `/stuff/` der Anforderung in `/customerA/good/stuff/`. Bei Anforderungen, die über eine Verbindung mit der Adresse 0.83.100.45 empfangen werden, ändert der Server die Zeichenfolge `/stuff/` der Anforderung in `/customerB/good/stuff/`.

  ```
  Map   /stuff/*   /customerA/good/stuff/*   240.146.167.72
  Map   /stuff/*   /customerB/good/stuff/*   0.83.104.45
  ```

- In den folgenden Beispielen wird der optionale Parameter Hostname verwendet. Wenn der Server Anforderungen empfängt, die mit der Zeichenfolge `/stuff/` beginnen, ändert er die Anforderung basierend auf dem Hostnamen im URL in eine andere Anforderungszeichenfolge. Bei Anforderungen, die für hostA empfangen werden, ändert der Server die Zeichenfolge `/stuff/` in der Anforderung in `/customerA/good/stuff//`. Bei Anforderungen, die für hostB empfangen werden, ändert der Server die Zeichenfolge `/stuff/` der Anforderung in `/customerB/good/stuff/`.

  ```
  Map   /stuff/*   /customerA/good/stuff/*   hostA.bcd.com
  Map   /stuff/*   /customerB/good/stuff/*   hostB.bcd.com
  ```

Standardwert

Keiner.

MaxActiveThreads - Die maximale Anzahl aktiver Threads angeben

Mit dieser Anweisung können Sie die maximale Anzahl Threads festlegen, die gleichzeitig aktiv sein sollen. Wird diese Zahl erreicht, hält der Server neue Anforderungen in einer Warteschleife, bis eine andere Anforderung beendet ist und Threads verfügbar werden. Im Allgemeinen gilt: Je höher die Leistungsfähigkeit einer Maschine ist, desto größer kann der für diese Anweisung angegebene Wert sein. Wenn eine Maschine für System-Tasks, wie zum Beispiel das Auslagern von Speicher, zu viel Zeit benötigt, sollten Sie diesen Wert verkleinern.

Format

`MaxActiveThreads` *`Anzahl_Threads`*

Standardwert

```
MaxActiveThreads 100
```

MaxContentLengthBuffer - Die Größe des Puffers für dynamische Daten festlegen

Mit dieser Anweisung können Sie die Größe des Puffers für die durch den Server generierten dynamischen Daten festlegen. Dynamische Daten werden als Ausgabe von CGI-Programmen, Server-Side Includes und API-Programmen erzeugt.

Der Wert kann in Byte (B), Kilobyte (K), Megabyte (M) oder Gigabyte (G) angegeben werden. Zwischen Zahl und Maßeinheit (B, K, M, G) kann ein Leerzeichen stehen oder nicht.

Format

```
MaxContentLengthBuffer Größe
```

Standardwert

```
MaxContentLengthBuffer 100 K
```

MaxLogFileSize - Maximale Größe jeder Protokolldatei festlegen

Legen Sie mit dieser Anweisung die maximale Größe jeder Protokolldatei fest. Eine Protokolldatei darf die mit dieser Anweisung definierte Größe nicht überschreiten. Sobald eine Protokolldatei die definierte maximale Größe erreicht, wird sie geschlossen, und eine neue Protokolldatei mit demselben Namen wird erstellt. An den Namen wird die nächsthöhere ganze Zahl angehängt.

Anmerkungen:

1. Caching Proxy ist eine 32-Bit-Anwendung und öffnet seine Protokolldateien mit einer 32-Bit-Funktion. Aufgrund dieser Einschränkung darf MaxLogFileSize *nicht* größer als 2 GB sein. Caching Proxy kann blockiert werden, wenn die Protokolldatei eine Größe von 2 GB überschreitet und Caching Proxy versucht, bei gleichzeitiger aktiver Verarbeitung von Anforderungen in die Protokolldatei zu schreiben.
2. Auf Linux- und UNIX-Plattformen werden die Protokolldateien nicht erstellt, falls für die Verzeichnisse, in denen die Protokolldateien gespeichert sind, nicht zumindest für die Gruppe, unter der der Dämon ibmproxy ausgeführt wird, die Schreibberechtigung definiert ist. Mit anderen Worten, die Verzeichnisse für die Protokolldateien müssen für die Protokollierungsanweisungen in der Datei ibmproxy.conf mindestens Schreibberechtigungen für die Gruppe festlegen, die in der Datei ibmproxy.conf durch die Anweisung GroupId definiert ist. Dies kann nur dann problematisch werden, wenn das Standardverzeichnis der Protokolldateien geändert wurde, oder wenn die UserId- oder GroupId-Standardanweisung in der Datei ibmproxy.conf geändert wurde.

Die maximale Größe kann in Byte (B), Kilobyte (K), Megabyte (M) und Gigabyte (G) angegeben werden.

Format

```
MaxLogFileSize maximale_Größe {B | K | M | G}
```

Standardwert

```
MaxLogfileSize 128 M
```

MaxPersistRequest - Die maximale Anzahl Anforderungen angeben, die über eine persistente Verbindung empfangen werden können

Geben Sie mit dieser Anweisung die maximale Anzahl der Anforderungen an, die der Server über eine persistente Verbindung empfangen kann. Berücksichtigen Sie beim Festlegen dieses Wertes die Anzahl der Grafiken, die auf Ihren Seiten verwendet werden. Für jede Grafik ist eine eigene Anforderung erforderlich.

Format

```
MaxPersistRequest Anzahl
```

Standardwert

```
MaxPersistRequest 5
```

MaxQueueDepth - Die maximale Anzahl URLs angeben, die in der Warteschlange gespeichert werden sollen

Mit dieser Anweisung können Sie die maximale Länge für die Warteschlange des Cache-Agenten angeben, in der die anstehende Anforderungen für Seitenabfragen gespeichert werden. Bei einem großen System mit einer hohen Speicherkapazität können Sie eine größere Warteschlange für Anforderungen von Seitenabfragen definieren, ohne den gesamten verfügbaren Speicher zu belegen.

Die Warteschlange für URLs, die im Cache gespeichert werden soll, wird bei jedem Start des Cache-Agenten definiert. Wird der Cache-Agent angewiesen, den Hypertext-Links zu anderen URLs zu folgen, werden diese anderen URLs bei der Cache-Warteschlangenlänge nicht berücksichtigt. Sobald der in der Anweisung MaxURLs angegebene Wert erreicht wird, wird der Cache-Agent gestoppt, auch wenn sich in der Warteschlange noch weitere URLs befinden.

Format

```
MaxQueueDepth maximale_Warteschlangenlänge
```

Standardwert

```
MaxQueueDepth 250
```

MaxRuntime - Die maximale Laufzeit für den Cache-Agenten angeben

Mit dieser Anweisung können Sie festlegen, wieviel Zeit dem Cache-Agenten maximal für den Abruf von URLs zugestanden wird. Der `0` bedeutet, dass der Cache-Agent so lange ausführt wird, bis alle URLs abgerufen sind.

Format

```
MaxRuntime {0 | maximale_Zeit}
```

Beispiel

```
MaxRuntime 2 hours 10 minutes
```

Standardwert

```
MaxRuntime 2 hours
```

MaxSocketPerServer - Die maximale Anzahl offener Sockets für den Server angeben

Mit dieser Anweisung können Sie die maximale Anzahl offener Sockets angeben, die für einen Ursprungsserver verwaltet werden sollen. Diese Anweisung sollte nur verwendet werden, wenn die Anweisung ServerConnPool auf `on` gesetzt ist.

Format

```
MaxSocketPerServer Anzahl
```

Beispiel

```
MaxSocketPerServer 10
```

Standardwert

```
MaxSocketPerServer 5
```

MaxUrls - Die maximale Anzahl URLs angeben, die aktualisiert werden sollen

Mit dieser Anweisung können Sie die maximale Anzahl der URLs angeben, die der Cache-Agent in einem Verarbeitungsdurchlauf abrufen darf. Wenn keine Begrenzung vorgesehen ist, muss der Wert `0` eingegeben werden. Wird der Cache-Agent im Automatikmodus ausgeführt, haben die Anweisungen LoadURL und LoadTopCached Vorrang vor der Anweisung MaxURLs.

Format

```
MaxURLs maximale_Anzahl
```

Standardwert

```
MaxURLs 2000
```

Member - Ein Member eines Bereichs angeben

Mit dieser Anweisung können Sie die Member für die Bereiche angeben, die von den Servern, die den Fernzugriff auf den Cache (RCA = Remote Cache Access) verwenden, gemeinsam genutzt werden.

Anmerkung: Beim Einrichten eines Bereichs muss die Anweisung Hostname für alle Member des Bereichs gleich konfiguriert werden.

Format

```
Member Name {
untergeordnete_Anweisung
untergeordnete_Anweisung
.
.
}
```

Folgende untergeordnete Anweisungen sind verfügbar:

RCAAddr
Diese untergeordnete Anweisung, deren Angabe erforderlich ist, gibt die IP-Adresse oder den Hostnamen für die RCA-Kommunikation an.

RCAPort
Diese untergeordnete Anweisung, deren Angabe erforderlich ist, gibt den Port für die RCA-Kommunikation an. Der Port muss größer als 1024 und kleiner als 65535 sein.

CacheSize {*n* bytes | *n* Kbytes | *n* Mbytes | *n* Gbytes}
Mit dieser untergeordneten Anweisung, deren Angabe erforderlich ist, wird für dieses Member die Größe seines Cache festgelegt. Es muss ein positiver Wert angegeben werden.

[Timeout *n* milliseconds | *n* seconds | *n* hours | *n* days | *n* months | *n* years | forever]
Gibt an, wie lange auf dieses Member gewartet werden soll. *n* muss eine positive ganze Zahl sein. Die Angabe von `Timeout` ist optional. Der Standardwert beträgt `1000 milliseconds`. Die Werte für Timeout werden normalerweise in Sekunden oder Millisekunden angegeben.

[BindSpecific {On | Off}]
Ermöglicht die Kommunikation in einem privaten Teilnetz mit einem gewissen Grad an Sicherheit. Die Angabe von `BindSpecific` ist optional; der Standardwert ist `On`.

[ReuseAddr {On | Off}]
Erlaubt eine schnellere Wiederverwendung des Bereichs. Die Einstellung `On` erlaubt es anderen Prozessen, den Port zu verwenden, was zu einem undefinierten Verhalten führen kann. Die Angabe von `ReuseAddr` ist optional. Der Standardwert lautet `Off`.

Beispiel

```
Member bittersweet.chocolate.ibm.com {
  RCAAddr 127.0.0.1
  RCAPort 6294
  CacheSize 25G
  Timeout 500 milliseconds
  BindSpecific On
  ReuseAddr Off
  }
```

Standardwert

Keiner.

Midnight - API-Plug-in für die Archivierung von Protokollen angeben

Mit dieser Anweisung können Sie das Anwendungs-Plug-in angeben, das um 00:00 Uhr zur Archivierung der Protokolle ausgeführt wird. Diese Anweisung wird während der Installation initialisiert. Falls Sie diese Anweisung in der Konfigurationsdatei nicht angeben, findet keine Archivierung statt.

Format

`Midnight` */Pfad/Datei:Funktionsname*

/Pfad/Datei
Der vollständig qualifizierte Dateiname des kompilierten Programms einschließlich der Erweiterung.

Funktionsname
Der Name der Anwendungsfunktion in Ihrem Programm.

Standardwerte

- **Linux und UNIX:** `Midnight /usr/lib/archive.so:begin`
- **Windows:** `Midnight C:\Programme\IBM\edge\cp\\bin\archive.dll:begin`

NameTrans - Schritt ″Name Translation″ anpassen

Mit dieser Anweisung können Sie eine angepasste Anwendungsfunktion angeben, die der Server im Schritt ″Name Translation″ aufruft. Dieser Code stellt eine Methode bereit, mit der der virtuelle Pfad in der Anforderung in den physischen Pfad im Server übersetzt wird, wobei die URLs spezifischen Objekten zugeordnet werden.

Anmerkung: Dies ist keine Regel für Terminalzuordnung. Der übersetzte URL muss trotzdem mit einer der Regelanweisungen für die Terminalzuordnung übereinstimmen, wie zum Beispiel Exec, Fail, Map, Pass, Redirect und Service.

Format

```
NameTrans Anforderungsschablone /Pfad/Datei:Funktionsname
  [Server-IP-Adresse | Hostname]
```

Anforderungsschablone
Eine Schablone für Anforderungen, die festlegt, ob eine Anwendungsfunktion aufgerufen wird. Die Schablone kann als Angaben das Protokoll, die Domäne und den Host enthalten, sie darf als vorangestelltes Zeichen einen Schrägstrich (/) verwenden sowie als Platzhalterzeichen einen Stern (*). Beispielsweise sind folgende Schablonen gültig: `/front_page.html`, `http://www.ics.raleigh.ibm.com`, `/pub*`, `/*` und `*`.

/Pfad/Datei
Der vollständig qualifizierte Dateiname des kompilierten Programms einschließlich der Erweiterung.

Funktionsname
Der Name der Anwendungsfunktion in Ihrem Programm.

[*Server-IP-Adresse* **|** *Hostname***]**
Falls Sie mehrere IP-Adressen oder virtuelle Hosts verwenden, geben Sie mit diesem Parameter an, ob die Anwendungsfunktion nur für Anforderungen aufgerufen wird, die an einer bestimmten IP-Adresse oder für einen bestimmten Host empfangen werden.

Für die IP-Adresse eines Servers kann kein Platzhalterzeichen verwendet werden.

Anmerkung: Die Anweisung muss in einer Zeile eingegeben werden, auch wenn sie hier zur besseren Lesbarkeit auf zwei Zeilen verteilt ist.

Beispiel

```
NameTrans /index.html /api/bin/icsextpgm.so:trans_url
```

Standardwert

Keiner.

NoBG - Den Caching-Proxy-Prozess im Vordergrund ausführen

Mit dieser Anweisung können Sie auf Linux- und UNIX-Plattformen verhindern, dass der Caching-Proxy-Serverprozess automatisch im Hintergrund ausgeführt wird. Diese Anweisung, die standardmäßig auf off gesetzt ist, hat folgendes Format:

```
NoBG [on | off]
```

Anmerkung: Die Option **-nobg** für den Befehl **ibmproxy** ist für Windows-Systeme ungültig.

Beispiel

```
NoBG on
```

Standardwert

```
NoBG off
```

NoCaching - Dateien, deren URLs mit einer Schablone übereinstimmen, nicht im Cache speichern

Legen Sie mit dieser Anweisung fest, dass der Server Dateien, deren URLs mit der angegebenen Schablone übereinstimmen, nicht im Cache speichern soll. Sie können mehrere dieser Anweisungen in der Konfigurationsdatei definieren. Fügen Sie für jede Schablone eine eigene Anweisung ein. Die URL-Schablone muss das Protokoll enthalten.

Wird weder die Anweisung CacheOnly noch die Anweisung NoCaching festgelegt, wird jeder URL im Cache gespeichert.

Format

```
NoCaching URL-Muster
```

Beispiel

```
NoCaching http://joke/*
```

Standardwert

Keiner.

NoLog - Protokolleinträge für bestimmte Hosts oder Domänen, die mit einer Schablone übereinstimmen, unterdrücken

Legen Sie mit dieser Anweisung fest, dass Zugriffsanforderungen von bestimmten Hosts oder Domänen, die mit einer bestimmten Schablone übereinstimmen, nicht protokolliert werden. Auf diese Weise kann zum Beispiel das Protokollieren von Zugriffsanforderungen von lokalen Hosts unterdrückt werden.

Sie können mehrere dieser Anweisungen in der Konfigurationsdatei angeben. Außerdem können mit derselben Anweisung mehrere Schablonen erfasst werden. Dazu sind diese Schablonen durch ein oder mehrere Leerzeichen voneinander zu trennen. In den Schablonen können Hostnamen oder numerische IP-Adressen verwendet werden.

Anmerkung: Um Schablonen für Hostnamen verwenden zu können, muss die Anweisung DNS-Lookup auf `On` festgelegt sein. Ist die Anweisung DNS-Lookup auf `Off` festgelegt (Standardwert), können ausschließlich Schablonen für IP-Adressen verwendet werden.

Format

```
NoLog  {Hostname | IP-Adresse}  [...]
```

Beispiel

```
NoLog 128.0.*  *.edu  localhost.*
```

Standardwert

Keiner.

no_proxy - Schablonen für Direktverbindungen zu Domänen angeben

Falls Sie eine der Anweisungen http_proxy, ftp_proxy oder gopher_proxy für die Kettung von Proxy-Servern verwenden, können Sie mit dieser Anweisung die Domänen angeben, zu denen der Server keine Proxy-Server-Verbindung, sondern eine Direktverbindung herstellt.

Geben Sie den Wert als Zeichenfolge von Domänennamen oder Domänennamenschablonen ein. Trennen Sie die einzelnen Einträge in der Zeichenfolge mit einem Komma (,). Fügen Sie *keine* Leerzeichen in die Zeichenfolge ein.

Bei dieser Anweisung werden die Schablonen auf andere Weise eingegeben als bei anderen Anweisungen. Der wichtigste Unterschied besteht darin, dass *kein* Platzhalterzeichen (*) verwendet werden darf. Eine Schablone *kann* angegeben werden, indem nur der letzte Teil eines Domänennamens eingegeben wird. Der Server stellt zu jeder Domäne, deren letzter Namensteil mit der angegebenen Schablone übereinstimmt, eine Direktverbindung her. Diese Anweisung gilt nur für die Kettung von Proxy-Servern und entspricht der Zeile `@/=` in der SOCKS-Konfigurationsdatei.

Format

`no_proxy` *`Domänenname_oder_Schablone`*`[,...]`

Beispiel

```
no_proxy www.someco.com,.raleigh.ibm.com,.some.host.org:8080
```

In diesem Beispiel verwendet der Server für die folgenden Anforderungen keine Proxy-Verbindung:

- alle Anforderungen von Domänen, die mit www.someco.com enden,
- alle Anforderungen von Domänen, die mit .raleigh.ibm.com enden, wie zum Beispiel blugrass.raleigh.ibm.com oder keystone.raleigh.ibm.com
- alle Anforderungen für Port 8080 von Domänen, die mit .some.host.org enden, wie zum Beispiel meinname.some.host.org:8080. (Dies gilt nicht für Anforderungen für alle anderen Ports derselben Domäne, wie z. B. meinname.some.host.org, die den Standard-Port 80 verwendet.)

Standardwert

Keiner.

NoProxyHeader - Die Client-Header angeben, die blockiert werden sollen

Mit dieser Anweisung können Sie die Client-URL-Header angeben, die blockiert werden sollen. Jeder durch einen Client gesendete HTTP-Header kann blockiert werden, auch erforderliche Header. Gehen Sie beim Blockieren von Headern sehr sorgfältig vor. Zu den allgemeinen Headern gehören:

- `Pragma:` wird normalerweise verwendet, um die Browser und Server mit Caches anzuweisen, bei jeder Anforderung einer Datei, die Datei vom Ursprungsserver abzurufen.
- `Referer:` ist der URL der Datei, aus der der Anforderungs-URI abgerufen wurde.

Einzelheiten zu diesen und anderen Headern finden Sie in der Spezifikation des HTTP-Protokolls. Sie können mehrere dieser Anweisungen angeben.

Format
NoProxyHeader *Header*

Beispiel
NoProxyHeader Referer:

Standardwert
Keiner.

NumClients - Die Anzahl der zu verwendenden Threads des Cache-Agenten angeben

Mit dieser Anweisung können Sie die Anzahl der Threads angeben, die der Cache-Agent zum Abrufen von Seiten in die Warteschlange verwendet. Richten Sie sich bei der Angabe der Anzahl Threads nach der Übertragungsgeschwindigkeit Ihres internen Netzes und Ihrer Internet-Verbindung. Der gültige Bereich liegt zwischen 1 und 100.

Anmerkung: Die Verwendung von mehr als sechs Threads kann möglicherweise dazu führen, dass eine zu hohe Anzahl schnell aufeinanderfolgender Anforderungen an den Inhaltsserver gesendet werden.

Format
NumClients *Zahl*

Standardwert
NumClients 4

ObjectType - Schritt ″Object Type″ anpassen

Mit dieser Anweisung können Sie eine angepasste Anwendungsfunktion angeben, die der Server im Schritt ″Object Type″ aufrufen soll. Mit diesem Code wird das angeforderte Objekt im Dateisystem gesucht und sein MIME-Typ angegeben.

Format
ObjectType *Anforderungsschablone /Pfad/Datei*:*Funktionsname*

Anforderungsschablone
Eine Schablone für Anforderungen, die festlegt, ob eine Anwendungsfunktion aufgerufen wird. Die Schablone kann als Angaben das Protokoll, die Domäne und den Host enthalten, sie darf als vorangestelltes Zeichen einen Schrägstrich (/) verwenden sowie als Platzhalterzeichen einen Stern (*). Beispielsweise sind folgende Schablonen gültig: /front_page.html, http://www.ics.raleigh.ibm.com, /pub*, /* und *.

/Pfad/Datei
Der vollständig qualifizierte Dateiname des kompilierten Programms einschließlich der Erweiterung.

Funktionsname
Der Name der Anwendungsfunktion in Ihrem Programm.

Beispiel
ObjectType /index.html /api/bin/icsextpgm.so:obj_type

Standardwert
Keiner.

OutputTimeout - Zeitlimit für die Ausgabe angeben

Mit dieser Anweisung können Sie festlegen, wieviel Zeit dem Server für das Senden der Ausgabe an einen Client zugestanden wird. Das Zeitlimit gilt für Anforderungen von lokalen Dateien und für Anforderungen, bei denen der Server als Proxy-Server agiert. Das Zeitlimit gilt nicht für Anforderungen, mit denen ein lokales CGI-Programm gestartet wird.

Sendet der Server innerhalb der in dieser Anweisung angegebenen Zeitperiode nicht die vollständige Antwort, beendet der Server die Verbindung. Geben Sie die Zeitperiode in einer beliebigen Kombination aus Stunden, Minuten und Sekunden an.

Format

```
OutputTimeout Zeit
```

Standardwert

```
OutputTimeout 30 minutes
```

PacFilePath - Das Verzeichnis mit den PAC-Dateien angeben

Mit dieser Anweisung können Sie das Verzeichnis für die PAC-Dateien (PAC = Proxy Autoconfiguration Files) angeben, die mit dem Formular für die Fernkonfiguration der PAC-Datei generiert wurden.

Format

```
PacFilePath Verzeichnispfad
```

Standardwerte

- **Windows:** `PacFilePath c:\Programme\IBM\edge\cp\\HTML\pacfiles`
- **Linux und UNIX:** `PacFilePath /opt/ibm/edge/cp/server_root/pub/pacfiles`

Pass - Die Schablone zum Akzeptieren von Anforderungen angeben

Legen Sie mit dieser Anweisung eine Schablone für Anforderungen fest, die akzeptiert und mit einer Datei von Ihrem Server beantwortet werden sollen. Wenn eine Anforderung mit einer Schablone einer Pass-Anweisung übereinstimmt, wird diese Anforderung nicht mehr mit den Anforderungsschablonen nachfolgender Anweisungen verglichen.

Format

```
Pass  Anforderungsschablone [Dateipfad [IP-Adresse_des_Servers | Hostname]]
```

Anforderungsschablone
: Eine Schablone für Anforderungen, die der Server akzeptieren und mit der Rückgabe einer Datei beantworten soll.

  In der Schablone kann ein Stern (*) als Platzhalterzeichen verwendet werden. Für das Tilde-Zeichen (~) direkt nach dem Schrägstrich (/) muss eine exakte Übereinstimmung bestehen. Ein Platzhalterzeichen wird nicht als Übereinstimmung gewertet.

[*Dateipfad***]**
: Der Pfad zur Datei, die der Server zurückgeben soll. Der *Dateipfad* darf ein Platzhalterzeichen verwenden, wenn die *Anforderungsschablone* ebenfalls ein Platzhalterzeichen enthält. Der Teil der Anforderung, der mit dem Platzhalterzeichen in der *Anforderungsschablone* übereinstimmt, wird an Stelle des Platzhalterzeichens im *Dateipfad* eingesetzt.

Dieser Parameter ist optional. Falls kein Pfad angegeben wird, wird die Anforderung selbst als Pfad verwendet.

[*Server-IP-Adresse* | *Hostname*]
Falls Sie mehrere IP-Adressen oder virtuelle Hosts verwenden, geben Sie mit diesem Parameter eine IP-Adresse oder einen Hostnamen an. Der Server verwendet die Anweisung nur für Anforderungen an diese IP-Adresse oder für diesen Host. Bei einer IP-Adresse ist dies die Adresse der Netzverbindung des Servers und nicht die Adresse des Clients, von dem die Anforderung stammt.

Sie können eine IP-Adresse (z. B. `240.146.167.72`) oder einen Hostnamen (z. B. `hostA.raleigh.ibm.com`) angeben.

Dieser Parameter ist optional. Ohne diesen Parameter verwendet der Server die Anweisung für alle Anforderungen ungeachtet der IP-Adresse, unter der die Anforderungen eingehen, oder des Hostnamens im URL.

Für die IP-Adresse eines Servers kann kein Platzhalterzeichen verwendet werden.

Beispiele

- In den folgenden Beispielen beantwortet der Server eine Anforderung, die mit `/updates/parts/` beginnt, mit einer Datei aus dem aufgelisteten Pfad des entsprechenden Betriebssystems. Alle Angaben nach `/updates/parts/` werden ebenfalls zur Angabe der Datei verwendet.

  Linux- und UNIX-Systeme: `Pass /updates/parts/* /opt/ibm/edge/cp/server_root/pub/*`

  Windows-Systeme: `Pass /updates/parts/* c:\Programme\IBM\edge\cp\pub\*`

- Im folgenden Beispiel beantwortet der Server eine Anforderung, die mit `/gooddoc/` beginnt, mit einer Datei aus dem Verzeichnis /gooddoc. Folglich antwortet der Server auf die Anforderung `/gooddoc/volume1/issue2/newsletter4.html` mit dem Dokument in der Datei /gooddoc/volume1/issue2/newsletter4.html.

  ```
  Pass
  /gooddoc/*
  ```

- Die folgenden Beispiele verwenden den optionalen Parameter IP-Adresse. Wenn der Server Anforderungen empfängt, die mit `/parts/` beginnen, gibt er je nach IP-Adresse der Netzverbindung, über die die Anforderung empfangen wird, eine Datei aus einem anderen Verzeichnis zurück. Für Anforderungen, die unter der Adresse 240.146.167.72 empfangen werden, gibt der Server eine Datei aus dem Verzeichnis /customerA/catalog/ zurück. Für Anforderungen, die über eine Verbindung zur Adresse 0.83.100.45 empfangen werden, gibt der Server eine Datei aus dem Verzeichnis /customerB/catalog/ zurück.

  ```
  Pass   /parts/*   /customerA/catalog/*   240.146.167.72
  Pass   /parts/*   /customerB/catalog/*   0.83.100.45
  ```

- In den folgenden Beispielen wird der optionale Parameter Hostname verwendet. Wenn der Server Anforderungen empfängt, die mit `/parts/` beginnen, gibt er auf der Basis des URL verwendeten Hostnamen eine Datei aus einem anderen Verzeichnis zurück. Gehen Anforderungen für hostA ein, gibt der Server eine Datei aus dem Verzeichnis /customerA/catalog/ zurück. Gehen Anforderungen für hostB ein, gibt der Server eine Datei aus dem Verzeichnis /customerB/catalog/ zurück.

AIX-Systeme

```
Pass    /Admin/*    /usr/lpp/internet/server_root/Admin/*
Pass    /Docs/*     /usr/lpp/internet/server_root/Docs/*
Pass /errorpages/*  /usr/lpp/internet/server_root/pub/errorpages/*
Pass /*          /usr/lpp/internet/server_root/pub/*
```

Solaris-, HP-UX- und Linux-Systeme

```
Pass    /Admin/*    /opt/ibm/edge/cp/server_root/Admin/*
Pass    /Docs/*     /opt/ibm/edge/cp/server_root/Docs/*
Pass /errorpages/*  /opt/ibm/edge/cp/server_root/pub/errorpages/*
Pass /*          /opt/ibm/edge/cp/server_root/pub/*
```

Standardwerte

AIX-Systeme

```
Pass    /Admin/*    /usr/lpp/internet/server_root/Admin/*
Pass    /Docs/*     /usr/lpp/internet/server_root/Docs/*
Pass    /errorpages/*  /usr/lpp/internet/server_root/pub/errporpages/*
Pass /*          /usr/lpp/internet/server_root/pub/*
```

HP-UX-, Linux- und Solaris-Systeme

```
Pass    /Admin/*    /opt/ibm/edge/cp/server_root/Admin/*
Pass    /Docs/*     /opt/ibm/edge/cp/server_root/Docs/*
Pass    /errorpages/*  /opt/ibm/edge/cp/server_root/pub/errorpages/*
Pass    /*          /opt/ibm/edge/cp/server_root/pub/*
```

Windows-Systeme

```
Pass    /icons/*    C:\Programme\IBM\edge\cp\icons\*
Pass    /Admin/*    C:\Programme\IBM\edge\cp\Admin\*
Pass    /Docs/*     C:\Programme\IBM\edge\cp\Docs\*
Pass    /erropages/*  C:\Programme\IBM\edge\cp\pub\errorpages\*
Pass    /*          C:\Programme\IBM\edge\cp\pub\*
```

PersistTimeout - Wartezeit zwischen Clientanforderungen angeben

Mit dieser Anweisung können Sie angeben, wie lange der Server zwischen Clientanforderungen wartet, bevor er eine persistente Verbindung abbricht. Als Zeit kann jede gültige Zeitangabe verwendet werden, normalerweise wird die Zeit jedoch in Sekunden und Minuten angegeben.

Mit Hilfe einer anderen Zeitlimitanweisung, der Anweisung InputTimeout, bestimmt der Server, wie lange er auf die erste Clientanforderung wartet, nachdem die Verbindung hergestellt wurde. Nähere Informationen über das Zeitlimit für die Eingabe finden Sie im Abschnitt „InputTimeout - Zeitlimit für Eingabe festlegen" auf Seite 229.

Nachdem der Server sein erste Antwort gesendet hat, bestimmt er anhand des mit der Anweisung PersistTimeout festgelegten Werts, wie lange er auf die nachfolgende Anforderung warten soll, bevor er die persistente Verbindung abbricht.

Format

```
PersistTimeout Zeit
```

Standardwert

```
PersistTimeout 4 seconds
```

PICSDBLookup - Schritt für Abfrage des PICS-Kennsatzes anpassen

Mit dieser Anweisung können Sie eine angepasste Anwendungsfunktion angeben, die der Server aufruft, um PICS-Kennsätze für einen angegebenen URL abzurufen. Diese Funktion kann einen PICS-Kennsatz entweder dynamisch für die angeforderte Datei erstellen oder in einer alternativen Datei oder Datenbank nach einem PICS-Kennsatz suchen.

Format

```
PICSDBLookup  /Pfad/Datei:Funktionsname
```

/Pfad/Datei
Der vollständig qualifizierte Dateiname des kompilierten Programms einschließlich der Erweiterung.

Funktionsname
Der Name der Anwendungsfunktion in Ihrem Programm.

Beispiel

```
PICSDBLookup /api/bin/icsext05.so:get_pics
```

Standardwert

Keiner.

PidFile (nur Linux und UNIX) - Die Datei angeben, in der die Prozess-ID von Caching Proxy gespeichert werden soll

Diese Anweisung ist nur für **Linux- und UNIX-Systeme gültig**. Mit dieser Anweisung können Sie die Adresse der Datei angeben, die die Prozess-ID von Caching Proxy enthält. Wenn der Serverprozess startet, zeichnet er seine Prozess-ID (PID) in einer Datei auf. Werden auf einem System mehrere Instanzen des Servers ausgeführt, muss jede Instanz eine eigene Anweisung PidFile besitzen.

Format

```
PidFile Pfad_zur_PID-Dateiinformation
```

Beispiel

```
PidFile /usr/pidinfo
```

Standardwerte

- Ist eine ServerRoot-Anweisung angegeben: `PidFile server_root /ibmproxy-pid`
- Ist keine ServerRoot-Anweisung angegeben: `PidFile /tmp/ibmproxy-pid`

Anweisungen für Plug-in-Module

Die nachfolgend aufgeführten Anweisungen wurden zur Datei ibmproxy.conf von Caching Proxy hinzugefügt, um neue Funktionen und Plug-ins zu aktivieren. Zum Editieren der meisten Anweisungen stehen keine Konfigurations- und Verwaltungsformulare zur Verfügung. Die Anweisungen müssen in einem Standardtexteditor wie vi oder emacs editiert werden. Nähere Informationen zu jeder der folgenden neuen Anweisungen sind in diesem Kapitel in alphabetischer Reihenfolge enthalten.

- „ExternalCacheManager - Caching Proxy für dynamisches Caching vom IBM WebSphere Application Server konfigurieren" auf Seite 219
- „ICP_Address — IP-Adresse für ICP-Abfragen angeben" auf Seite 225
- „ICP_Port — Port-Nummer für ICP-Abfragen angeben" auf Seite 227

- „ICP_Timeout — Maximale Wartezeit für ICP-Abfragen angeben" auf Seite 227
- „Occupier - Member eines ICP-Cluster angeben" auf Seite 226
- „ICP_MaxThreads - Maximale Anzahl Threads für ICP-Abfragen angeben" auf Seite 226
- „SignificantURLTerminator - Abschlusscode für URL-Anforderungen festlegen" auf Seite 278
- „SSLCertificate - Schlüsselkennsätze für Zertifikate angeben" auf Seite 279
- „SSLOnly - Empfangs-Threads für HTTP-Anforderungen inaktivieren" auf Seite 281

In der Datei ibmproxy.conf müssen Anweisungen zum Konfigurieren von Plug-in-Modulen für Caching Proxy im folgenden Format eingegeben werden:

```
<MODULEBEGIN> Name des Plug-in
untergeordnete_Anweisung1
untergeordnete_Anweisung2

<MODULEEND>
```

Jedes Plug-in-Programm führt eine Syntaxanalyse der Datei ibmproxy.conf durch und liest nur den eigenen Block mit untergeordneten Anweisungen. Der Parser von Caching Proxy ignoriert alle Einträge zwischen `<MODULEBEGIN>` und `<MODULEEND>`.

Für die Plug-in-Module von Caching Proxy und für einige neuen Funktionen müssen API-Anweisungen zur Datei ibmproxy.conf hinzugefügt werden. Weil der Proxy-Server die Plug-in-Module in der Reihenfolge verarbeitet, in der sie aufgelistet sind, müssen Sie beim Anordnen der Anweisungen in der Konfigurationsdatei des Proxy-Servers sorgfältig vorgehen. Beachten Sie, dass die prototype-Anweisungen (in Form von Kommentaren) zum API-Abschnitt der Datei ibmproxy.conf hinzugefügt wurden. Diese API-Anweisungen sind in einer zweckmäßigen Reihenfolge angeordnet. Wenn Sie API-Anweisungen hinzufügen, um neue Funktionen und Plug-in-Module zu aktivieren, sollten Sie die Anweisungen wie im Prototypabschnitt der Konfigurationsdatei gezeigt anordnen. Alternativ dazu können Sie, falls erforderlich, die Kommentarzeichen für API-Anweisungen entfernen und API-Anweisungen editieren, um die Unterstützung für jede gewünschte Funktion oder jedes gewünschte Plug-in hinzuzufügen. Plug-in-Module, die vom Benutzer erstellt wurden, müssen hinter den Plug-in-Modulen des Produkts hinzugefügt werden.

Port - Den Port angeben, an dem der Server Anforderungen empfängt

Mit dieser Anweisung können Sie die Nummer des Port angeben, an dem der Server auf Anforderungen wartet. Die Standard-Port-Nummer für HTTP ist 80. Andere Port-Nummern unter 1024 sind für andere TCP/IP-Anwendungen reserviert und dürfen nicht verwendet werden. Die für Proxy-Webserver verwendeten allgemeinen Ports sind 8080 und 8008.

Wird eine andere Port-Nummer als 80 verwendet, sind Clients erforderlich, um für Anforderungen an den Server eine spezielle Port-Nummer anzusprechen. Vor der Port-Nummer steht ein Doppelpunkt (:), und die Port-Nummer wird im URL an den Hostnamen angehängt. Beispielsweise wird bei Angabe des URL `http://www.turfco.com:8008/` im Browser die Standardbegrüßungsseite des Host mit dem Namen www.turfco.com angefordert, der an Port 8008 Anforderungen empfängt.

Sie können die Option **-p** im Befehl **ibmproxy** angeben, um diese Einstellung beim Starten des Servers außer Kraft zu setzen.

Format

```
Port Nummer
```

Wenn Sie diese Anweisung ändern, müssen Sie den Server manuell stoppen und anschließend erneut starten, damit die Änderung wirksam wird. Der Server erkennt die Änderung erst, wenn Sie ihn erneut starten. (Nähere Informationen hierzu finden Sie in Kapitel 5, „Caching Proxy starten und stoppen”, auf Seite 15.)

Standardwert

```
Port 80
```

PostAuth — Schritt ″PostAuth″ anpassen

Mit dieser Anweisung können Sie eine angepasste Anwendungsfunktion angeben, die der Server während des Schritts ″PostAuth″ aufrufen soll. Dieser Code wird ohne Berücksichtigung der Rückkehrcodes früherer Schritte oder anderer Post-Auth-Steuerroutinen ausgeführt. Er ermöglicht die Bereinigung aller Ressourcen, die zur Verarbeitung der Anforderung zugewiesen wurden.

Format

```
PostAuth  /Pfad/Datei:Funktionsname
```

/Pfad/Datei
: Der vollständig qualifizierte Dateiname des kompilierten Programms einschließlich der Erweiterung.

Funktionsname
: Der Name der Anwendungsfunktion in Ihrem Programm.

Beispiel

```
AuthExit  /ics/api/bin/icsext05.so:post_exit
```

Standardwert

Keiner.

PostExit - Schritt ″PostExit″ anpassen

Mit dieser Anweisung können Sie eine angepasste Anwendungsfunktion angeben, die der Server während des Schritts ″PostExit″ aufrufen soll. Dieser Code wird ohne Berücksichtigung der Rückkehrcodes früherer Schritte oder anderer PostExit-Steuerroutinen ausgeführt. Er ermöglicht die Bereinigung aller Ressourcen, die zur Verarbeitung der Anforderung zugewiesen wurden.

Format

```
PostExit  /Pfad/Datei:Funktionsname
```

/Pfad/Datei
: Der vollständig qualifizierte Dateiname des kompilierten Programms einschließlich der Erweiterung.

Funktionsname
: Der Name der Anwendungsfunktion in Ihrem Programm.

Beispiel

```
PostExit /ics/api/bin/icsext05.so:post_exit
```

Standardwert
Keiner.

PreExit - Schritt ″PreExit″ anpassen

Mit dieser Anweisung können Sie eine angepasste Anwendungsfunktion angeben, die der Server während des Schritts ″PreExit″ aufrufen soll. Dieser Code wird nach dem Lesen einer Clientanforderung, aber vor allen weiteren Verarbeitungsschritten ausgeführt. In diesem Schritt kann das Modul GoServe aufgerufen werden.

Format
```
PreExit /Pfad/Datei:Funktionsname
```

/Pfad/Datei
Der vollständig qualifizierte Dateiname der kompilierten DLL einschließlich der Erweiterung.

Funktionsname
Der Name der Anwendungsfunktion in Ihrem Programm.

Beispiel
```
PreExit /ics/api/bin/icsext05.so:pre_exit
```

Standardwert
Keiner.

Protect - Eine Zugriffsschutzkonfiguration für Anforderungen aktivieren, die mit einer Schablone übereinstimmen

Mit dieser Anweisung können Sie für Anforderungen, die mit einer Schablone übereinstimmen, die Zugriffsschutzkonfiguration aktivieren.

Anmerkung: Damit der Zugriffsschutz ordnungsgemäß funktioniert, müssen die Anweisungen DefProt und Protect in der Konfigurationsdatei vor allen Pass- Exec-, oder Proxy-Anweisungen stehen.

Eine Zugriffsschutzkonfiguration wird mit untergeordneten Anweisungen für den Zugriffsschutz definiert. Das Format der Anweisung Protect ist davon abhängig, ob auf einen Kennsatz oder eine Datei verwiesen werden soll, die die untergeordneten Anweisungen für Zugriffsschutz enthält, oder ob die einzelnen untergeordneten Anweisungen für Zugriffsschutz als Bestandteil der Anweisung Protect eingegeben werden sollen.

Format
Dieser Parameter kann folgende Formate verwenden:

- Die Anweisung Protect kann als vollständiger Pfad und Dateiname einer separaten Datei angegeben werden, die die untergeordneten Anweisungen für den Zugriffsschutz enthält. Sie kann auch mit einem Kennsatznamen für eine Zugriffsschutzkonfiguration angegeben werden, der mit einem Namen übereinstimmt, der zuvor in einer Protection-Anweisung definiert wurde. Die Anweisung Protection enthält die untergeordneten Anweisungen für den Zugriffsschutz. Verwenden Sie folgendes Format:

  ```
  Protect Anforderungsschablone [Konfigurationsdatei | Kennsatz]
      [FOR Server-IP-Adresse | Hostname]
  ```

Anmerkung: Die Anweisung muss in einer Zeile eingegeben werden, auch wenn sie hier zur besseren Lesbarkeit in zwei Zeilen dargestellt ist.

- Sie können die untergeordneten Anweisungen für den Zugriffsschutz als Teil der Anweisung Protect angeben. Die untergeordneten Anweisungen müssen in geschweifte Klammern ({}) eingeschlossen werden. Die linke geschweifte Klammer muss in derselben Zeile wie die Anweisung Protect das letzte Zeichen stehen. Jede untergeordnete Anweisung steht in einer separaten Zeile. Die rechte geschweifte Klammer muss im Anschluss an die letzte untergeordnete Anweisung in einer separaten Zeile stehen. Zwischen den geschweiften Klammern dürfen keine Kommentarzeilen enthalten sein. Sollen die untergeordneten Anweisungen für Zugriffsschutz als Teil der Anweisung Protect eingegeben werden, gilt das folgende Format:

```
Protect Anforderungsschablone [FOR Server-IP-Adresse | Hostname]
    untergeordnete Anweisung  Wert
    untergeordnete Anweisung  Wert
    .
    .
    .
  }
```

Folgende Parameter werden verwendet:

Anforderungsschablone
Eine Schablone für Anforderungen, für die der Zugriffsschutz aktiviert werden soll. Der Server vergleicht die eingehenden Clientanforderungen mit der Schablone und aktiviert bei Übereinstimmung den Zugriffsschutz.

[*Konfigurationsdatei* **|** *Kennsatz***]**
Falls auf einen Kennsatz oder auf eine Datei verwiesen wird, die die untergeordneten Anweisungen für Zugriffsschutz enthält, legt dieser Parameter die Zugriffsschutzkonfiguration fest, die für Anforderungen aktiviert werden soll, die mit der *Anforderungsschablone* übereinstimmen.

Dieser Parameter ist optional. Wird dieser Parameter weggelassen, wird die Zugriffsschutzkonfiguration aktiviert, die durch die zuletzt bearbeitete DefProt-Anweisung mit einer übereinstimmenden Schablone definiert wurde.

[FOR *Server-IP-Adresse* **|** *Hostname***]**
Falls Sie mehrere IP-Adressen oder virtuelle Hosts verwenden, geben Sie mit diesem Parameter eine IP-Adresse oder einen Hostnamen an. Der Server verwendet die Anweisung nur für Anforderungen an diese IP-Adresse oder für diesen Host. Bei einer IP-Adresse ist dies die Adresse der Netzverbindung des Servers und nicht die Adresse des Clients, von dem die Anforderung stammt. Wenn Sie eine IP-Adresse schützen, sind sowohl die IP-Adresse als auch der vollständig qualifizierte Hostname geschützt. Falls der Server jedoch in seinem Netzwerk mit einem anderen Namen als dem vollständig qualifizierten Namen aufgerufen wird, z. B. mit einem Eintrag aus einer Datei mit Hostnamen, ist er nicht geschützt.

Beispiel:

```
Protect http://x.x.x.x  PROT-ADMIN
```

In einem Webbrowser:

- http://x.x.x.x ist geschützt
- http://hostname.Beispiel.com ist geschützt
- http://hostname ist nicht geschützt

Beispiel:

```
Protect http://hostname.Beispiel.com  PROT-ADMIN
```

In einem Webbrowser:
- http://x.x.x.x ist nicht geschützt
- http://hostname.Beispiel.com ist geschützt
- http://hostname ist nicht geschützt

Sie können eine IP-Adresse (z. B. `FOR 240.146.167.72`) oder einen Hostnamen (z. B. `FOR hostA.bcd.com`) angeben.

Platzhalterzeichen dürfen zur Angabe von Server-IP-Adressen nicht verwendet werden.

Dieser Parameter ist optional. Ohne diesen Parameter verwendet der Server die Anweisung für alle Anforderungen ungeachtet der IP-Adresse, unter der die Anforderungen eingehen, oder des Hostnamens im URL.

Anmerkung: Der Parameter [*Server-IP-Adresse* | *Hostname*] wird entweder mit dem Parameter [*Konfigurationsdatei* | *Kennsatz*] oder mit dem Parameter *untergeordnete_Anweisung Wert* verwendet.
- Damit der Parameter [*Server-IP-Adresse* | *Hostname*] mit dem Parameter [*Konfigurationsdatei* | *Kennsatz*] verwendet werden kann, muss die Zeichenfolge `FOR` oder eine andere Zeichenfolge (ohne Leerzeichen) zwischen den Parameter [*Konfigurationsdatei* | *Kennsatz*] und den Parameter [*Server-IP-Adresse* | *Hostname*] gesetzt werden.
- Damit der Parameter [*Server-IP-Adresse* | *Hostname*] mit dem Parameter *untergeordnete_Anweisung Wert* verwendet werden kann, darf die Zeichenfolge `FOR` *nicht* vor der *IP-Adresse* oder vor dem *Hostnamen* angegeben werden.

untergeordnete_Anweisung Wert

Verwenden Sie diesen Parameter, um die untergeordneten Anweisungen für Zugriffsschutz als Teil der Anweisung Protect einzugeben. Beschreibungen der untergeordneten Anweisungen für den Zugriffsschutz finden Sie in folgenden Abschnitten:

- „AuthType - Authentifizierungsart angeben” auf Seite 259
- „DeleteMask - Die Benutzernamen, Gruppen und Adressen angeben, denen das Löschen von Dateien erlaubt ist” auf Seite 259
- „GetMask - Die Benutzernamen, Gruppen und Adressen angeben, denen das Abrufen von Dateien erlaubt ist” auf Seite 259
- „GroupFile - Die Position der zugeordneten Gruppendatei angeben” auf Seite 260
- „Mask - Die Benutzernamen, Gruppen und Adressen angeben, die HTTP-Anforderungen stellen dürfen” auf Seite 260
- „PasswdFile - Die Position der zugeordneten Kennwortdatei angeben” auf Seite 260
- „PostMask - Die Benutzernamen, Gruppen und Adressen angeben, denen das Versenden von Dateien erlaubt ist” auf Seite 261
- „PutMask - Die Benutzernamen, Gruppen und Adressen angeben, denen das Ablegen von Dateien erlaubt ist” auf Seite 261
- „ServerID - Einen Namen angeben, der der Kennwortdatei zugeordnet werden soll” auf Seite 261

Beispiele

- Im folgenden Beispiel wird der Zugriffsschutz wie folgt durch den Server aktiviert:
  - Anforderungen, die mit /secret/scoop/ beginnen, aktivieren den Zugriffsschutz. Die Zugriffsschutzkonfiguration ist in der Konfigurationsdatei für Zugriffsschutz, /server/protect/setup1.acc definiert. Weil die Anweisung Protect keine Zugriffsschutzkonfiguration festlegt, wird die Zugriffsschutzkonfiguration aus der letzten übereinstimmenden DefProt-Anweisung verwendet.
  - Anforderungen, die mit /secret/business/ beginnen, aktivieren den Zugriffsschutz. Die Zugriffsschutzkonfiguration ist in einer Protection-Anweisung mit dem Kennsatz BUS-PROT definiert.
  - Anforderungen, die mit /topsecret/ beginnen, aktivieren den Zugriffsschutz. Diese Zugriffsschutzkonfiguration wurde direkt in der Anweisung Protect definiert.

  Diese Beispiele verwenden IP-Adressen. Wenn der Server Anforderungen empfängt, die mit /secret/ oder /topsecret/ beginnen, aktiviert er auf der Basis der IP-Adresse der Netzverbindung, in der eine Anforderung empfangen wird, die unterschiedlichen Zugriffsschutzkonfigurationen für die Anforderungen.
  - Werden an der Adresse 0.67.106.79 Anforderungen empfangen, die mit /secret/ beginnen, aktiviert der Server die Zugriffsschutzkonfiguration, die in einer Protection-Anweisung mit dem Kennsatz CustomerA-PROT definiert ist. Für Anforderungen, die mit /topsecret/ requests beginnen und an der Adresse 0.67.106.79 empfangen werden, aktiviert der Server die Zugriffsschutzkonfiguration, die als Teil der Protect-Anweisung für /topsecret/ definiert ist.
  - Für Anforderungen, die mit /secret/ beginnen und an der Adresse 0.83.100.45 empfangen werden, aktiviert der Server die Zugriffsschutzkonfiguration, die in einer Protection-Anweisung mit dem Kennsatz CustomerB-PROT definiert ist. Für Anforderungen, die mit /topsecret/ beginnen und an der Adresse 0.83.100.45 empfangen werden, aktiviert der Server die Zugriffsschutzkonfiguration, die als Teil der Protect-Anweisung für /topsecret/ definiert ist.

```
Protection BUS-PROT {
   UserID    busybody
   GroupID   webgroup
  AuthType Basic
   ServerID restricted
   PasswdFile /docs/WWW/restrict.pwd
   GroupFile  /docs/WWW/restrict.grp
   GetMask authors
   PutMask authors
}
DefProt  /secret/*  /server/protect/setup1.acc
Protect  /secret/scoop/*
Protect  /secret/business/*   BUS-PROT
Protect  /topsecret/*  {
  AuthType Basic
   ServerID restricted
   PasswdFile /docs/WWW/restrict.pwd
   GroupFile  /docs/WWW/restrict.grp
   GetMask topbrass
   PutMask topbrass
}
Pass   /secret/scoop/*       /WWW/restricted/*
Pass   /secret/business/*    /WWW/confidential/*
Pass   /topsecret/*          /WWW/topsecret/*
```

```
Protect   /secret/*   CustomerA-PROT   FOR  0.67.106.79
Protect   /secret/*   CustomerB-PROT   FOR  0.83.100.45
Protect  /topsecret/*   0.67.106.79  {
  AuthType Basic
   ServerID restricted
   PasswdFile /docs/WWW/customer-A.pwd
   GroupFile  /docs/WWW/customer-A.grp
   GetMask A-brass
   PutMask A-brass
}
Protect  /topsecret/*   0.83.100.45   {
  AuthType Basic
   ServerID restricted
   PasswdFile /docs/WWW/customer-B.pwd
   GroupFile  /docs/WWW/customer-B.grp
   GetMask B-brass
   PutMask B-brass
}
```

- In den folgenden Beispielen werden virtuelle Hosts verwendet. Wenn der Server Anforderungen empfängt, die mit `/secret/` oder `/topsecret/` beginnen, werden auf der Basis des Hostnamens im URL unterschiedliche Zugriffsschutzkonfigurationen für die Anforderungen aktiviert.
  - Für Anforderungen, die mit `/secret/` beginnen und für hostA.bcd.com empfangen werden, aktiviert der Server die Zugriffsschutzkonfiguration, die in einer Protection-Anweisung mit dem Kennsatz `CustomerA-PROT` definiert ist. Für Anforderungen, die mit `/topsecret/` beginnen und für hostA.bcd.com empfangen werden, aktiviert der Server die Zugriffsschutzkonfiguration, die als Teil der Protect-Anweisung für `/topsecret/` definiert ist.
  - Für Anforderungen, die mit `/secret/` beginnen und für hostB.bcd.com empfangen werden, aktiviert der Server die Zugriffsschutzkonfiguration, die in einer Protection-Anweisung mit dem Kennsatz `CustomerB-PROT` definiert ist. Für Anforderungen, die mit `/topsecret/` beginnen und für hostB.bcd.com empfangen werden, aktiviert der Server die Zugriffsschutzkonfiguration, die als Teil der Protect-Anweisung für `/topsecret/` definiert ist.
  - Bei weitergeleiteten Anforderungen aktiviert der Server die Zugriffsschutzkonfiguration, die in einer Protection-Anweisung mit dem Kennsatz `proxy-prot` definiert ist. Beispiel:

    ```
    Protect http://host1/* proxy-prot
    ```

```
Protect /secret/* CustomerA-PROT FOR hostA.bcd.com
Protect /secret/* CustomerB-PROT FOR hostB.bcd.com
Protect  /topsecret/*   hostA.bcd.com  {
  AuthType Basic
   ServerID restricted
   PasswdFile /docs/WWW/customer-A.pwd
   GroupFile  /docs/WWW/customer-A.grp
   GetMask A-brass
   PutMask A-brass
}
Protect  /topsecret/*   hostB.bcd.com {
  AuthType Basic
   ServerID restricted
   PasswdFile /docs/WWW/customer-B.pwd
   GroupFile  /docs/WWW/customer-B.grp
   GetMask B-brass
   PutMask B-brass
   }
```

Standardwert

Standardmäßig wird der Zugriffsschutz für die Konfigurations- und Verwaltungsformulare durch eine Protect-Anweisung mit der Anforderungsschablone `/admin-bin/*` bereitgestellt.

Protection - In der Konfigurationsdatei eine benannte Zugriffsschutzkonfiguration definieren

Mit dieser Anweisung können Sie in der Konfigurationsdatei eine Zugriffsschutzkonfiguration definieren. Weisen Sie der Zugriffsschutzkonfiguration einen Namen zu und definieren Sie mit Hilfe der untergeordneten Anweisungen für den Zugriffsschutz die Art des Zugriffsschutzes.

Anmerkungen:

1. In der Konfigurationsdatei müssen die Protection-Anweisungen vor allen DefProt- und Protect-Anweisung stehen, die auf die Protection-Anweisungen verweisen.
2. Zur Verwendung von Domänennamen in den Zugriffsschutzregeln muss die Anweisung DNS-Lookup auf `on` gesetzt sein.

Format

```
Protection Kennsatzname  {
    untergeordnete Anweisung  Wert
    untergeordnete Anweisung  Wert
    .
    .
    .
  }
```

Kennsatzname
: Der Name, der dieser Zugriffsschutzkonfiguration zugeordnet werden soll. Der Name kann dann durch nachfolgende DefProt- und Protect-Anweisungen als Zeiger auf diese Zugriffsschutzkonfiguration verwendet werden.

untergeordnete_Anweisung Wert
: Die untergeordneten Anweisungen sind in geschweifte Klammern ({ }) eingeschlossen. Die linke geschweifte Klammer muss in der Zeile, in der der *Kennsatzname* angegeben ist, das letzte Zeichen sein. Jede untergeordnete Anweisung steht in einer separaten Zeile. Die rechte geschweifte Klammer muss nach der Zeile mit der letzten untergeordneten Anweisung in einer separaten Zeile stehen. Zwischen den geschweiften Klammern dürfen keine Kommentarzeilen enthalten sein.

  Beschreibungen der untergeordneten Anweisungen für den Zugriffsschutz finden Sie im Abschnitt „Untergeordnete Anweisungen für den Zugriffsschutz - Angeben, wie eine Gruppe von Ressourcen geschützt wird" auf Seite 259.

Beispiel

```
Protection NAME-ME   {
  AuthType Basic
   ServerID restricted
   PasswdFile  /WWW/password.pwd
   GroupFile   /WWW/group.grp
   GetMask groupname
   PutMask groupname
}
```

Standardwert

```
Protect /admin-bin/* {
  ServerId Private_Authorization
  AuthType Basic
  GetMask All@(*)
  PutMask All@(*)
  PostMask All@(*)
  Mask All@(*)
  PasswdFile   /opt/ibm/edge/cp/server_root/protect/webadmin.passwd
}
```

Untergeordnete Anweisungen für den Zugriffsschutz - Angeben, wie eine Gruppe von Ressourcen geschützt wird

Nachstehend finden Sie die Beschreibungen der untergeordneten Anweisungen für den Zugriffsschutz, die in einer Zugriffsschutzkonfiguration verwendet werden können. Die untergeordneten Anweisungen werden in alphabetischer Reihenfolge aufgeführt.

Zugriffsschutzkonfigurationen können entweder in separaten Dateien definiert sein oder in der Konfigurationsdatei als Teil von DefProt-, Protect- oder Protection-Anweisungen.

AuthType - Authentifizierungsart angeben

Verwenden Sie diese untergeordnete Anweisung für den Zugriffsschutz, wenn der Zugriff über Benutzernamen und Kennwörter eingeschränkt werden soll. Geben Sie die Authentifizierungsart an, die zu verwenden ist, wenn der Client an den Server ein Kennwort sendet. Bei der Authentifizierungsart Grundeinstellungen (`AuthType Basic`) werden die Kennwörter an den Server als Textdatei gesendet. Sie werden zwar codiert, aber nicht verschlüsselt.

Standardwert:

```
AuthType Basic
```

DeleteMask - Die Benutzernamen, Gruppen und Adressen angeben, denen das Löschen von Dateien erlaubt ist

Verwenden Sie diese untergeordnete Anweisung für den Zugriffsschutz zur Angabe von Schablonen für die Benutzernamen, Gruppen und Adressen, die berechtigt sind, DELETE-Anforderungen für ein geschütztes Verzeichnis auszuführen.

Beispiel:

```
DeleteMask authors,(niceguy,goodie)@45.96.3.1,128.0.*.*
```

GetMask - Die Benutzernamen, Gruppen und Adressen angeben, denen das Abrufen von Dateien erlaubt ist

Verwenden Sie diese untergeordnete Anweisung für den Zugriffsschutz zur Angabe von Schablonen für die Benutzernamen, Gruppen und Adressen, die zu GET-Anforderungen für ein geschütztes Verzeichnis berechtigt sind.

Beispiel:

```
GetMask authors,(niceguy,goodie)@45.96.3.1,128.0.*.*
```

Standardwert:

```
GetMask All@(*)
```

GroupFile - Die Position der zugeordneten Gruppendatei angeben

Verwenden Sie diese untergeordnete Anweisung für den Zugriffsschutz zur Angabe des Pfades und des Dateinamens der Servergruppendatei, die von dieser Zugriffsschutzkonfiguration verwendet wird. Die Gruppen, die in der Servergruppendatei definiert sind, können anschließend von folgenden Komponenten verwendet werden:

- Von jeder untergeordneten Anweisung für Maskierung, die Teil der Zugriffsschutzkonfiguration ist. (Die untergeordneten Anweisungen für Maskierung sind DeleteMask, GetMask, Mask, PostMask und PutMask.)
- Von jeder ACL-Datei in einem Verzeichnis, das durch die Zugriffsschutzkonfiguration geschützt wird.

Beispiel:

```
GroupFile /docs/etc/WWW/restrict.group
```

Mask - Die Benutzernamen, Gruppen und Adressen angeben, die HTTP-Anforderungen stellen dürfen

Verwenden Sie diese untergeordnete Anweisung zur Angabe von Schablonen für die Benutzernamen, Gruppen und Adressen, die zur Ausführung von HTTP-Anforderungen berechtigt sind, die von keiner anderen untergeordneten Anweisungen für Maskierung abgedeckt werden.

Beispiele:

```
Mask authors,(niceguy,goodie)@45.96.3.1,128.0.*.*
```

Anmerkung: Beachten Sie bei der Verwendung der Anweisung Mask, dass zwischen Groß- und Kleinschreibung unterschieden wird. Nachfolgend wird ein Beispiel für eine Maskierung gezeigt, die für eine Benutzer-ID angegeben wird:

```
MASK WEBADM,webadm
```

PasswdFile - Die Position der zugeordneten Kennwortdatei angeben

Verwenden Sie diese untergeordnete Anweisung für den Zugriffsschutz, wenn der Zugriff über Benutzernamen und Kennwörter eingeschränkt werden soll. Geben Sie den Pfad und den Dateinamen der Kennwortdatei an, die von dieser Zugriffsschutzkonfiguration verwendet werden soll.

Weil einige Browser Benutzer-IDs und Kennwörter im Host nach Sicherheitsbereichen (ServerID) im Cache zwischenspeichern, sollten Sie bei Angabe der Server-ID und Kennwortdateien die folgenden Richtlinien beachten:

- Für Zugriffsschutzkonfigurationen, die dieselbe Kennwortdatei verwenden, sollte dieselbe Server-ID verwendet werden.
- Für Zugriffsschutzkonfigurationen, die verschiedene Kennwortdateien verwenden, sollten auch verschiedene Server-IDs verwendet werden.

Beispiel:

```
PasswdFile /docs/etc/WWW/restrict.password
```

Anmerkung: Wenn der Pfad oder der Dateiname Leerzeichen enthält, müssen Sie den vollständigen Pfad und Dateinamen in Anführungszeichen ("") einschließen.

```
PasswdFile "c:\test this\admin.pwd"
```

PostMask - Die Benutzernamen, Gruppen und Adressen angeben, denen das Versenden von Dateien erlaubt ist

Verwenden Sie in einem sicheren Server diese untergeordnete Anweisung für den Zugriffsschutz zur Angabe von Schablonen für die Benutzernamen, Gruppen und Adressen, die zu POST-Anforderungen für ein geschütztes Verzeichnis berechtigt sind.

Beispiel:

```
PostMask authors,(niceguy,goodie)@45.96.3.1,128.0.*.*
```

PutMask - Die Benutzernamen, Gruppen und Adressen angeben, denen das Ablegen von Dateien erlaubt ist

Verwenden Sie diese untergeordnete Anweisung für den Zugriffsschutz zur Angabe von Schablonen für die Benutzernamen, Gruppen und Adressen, die zu PUT-Anforderungen für ein geschütztes Verzeichnis berechtigt sind.

Beispiel:

```
PutMask authors,(niceguy,goodie)@45.96.3.1,128.0.*.*
```

ServerID - Einen Namen angeben, der der Kennwortdatei zugeordnet werden soll

Verwenden Sie diese untergeordnete Anweisung für den Zugriffsschutz, wenn der Zugriff über Benutzernamen und Kennwörter eingeschränkt werden soll. Geben Sie einen Namen an, der der Kennwortdatei, die verwendet wird, zugeordnet werden soll. Der Name muss nicht der Name einer realen Maschine sein.

Der Name wird als Kennung für den Requester verwendet. Da unterschiedliche Zugriffsschutzkonfigurationen unterschiedliche Kennwortdateien verwenden können, wird dem Client durch die Zuordnung eines Namens für die Zugriffsschutzkonfiguration die Entscheidung erleichtert, welches Kennwort er zu senden hat. Die meisten Clients zeigen diesen Namen an, während Sie die Eingabe eines Benutzernamens und eines Kennwortes anfordern.

Weil einige Browser Benutzer-IDs und Kennwörter im Host nach Sicherheitsbereichen (ServerID) im Cache zwischenspeichern, sollten Sie bei Angabe der Server-ID und Kennwortdateien die folgenden Richtlinien beachten:

- Für Zugriffsschutzkonfigurationen, die dieselbe Kennwortdatei verwenden, sollte dieselbe Server-ID verwendet werden.
- Für Zugriffsschutzkonfigurationen, die verschiedene Kennwortdateien verwenden, sollten auch verschiedene Server-IDs verwendet werden.

Beispiel:

```
ServerID restricted
```

Proxy - Proxy-Protokolle oder einen Reverse Proxy angeben

Mit dieser Anweisung können Sie die Protokolle angeben, die Caching Proxy verarbeiten soll, einem Server Anforderungen zuordnen. Gültige Protokolle sind `http`, `ftp` und `gopher`.

Mit der Parameteroption UseSession der Anweisung Proxy kann ein persistenter Channel auf Serverseite eingerichtet werden, wenn die Anforderungen von Clients an denselben Server und von demselben clientseitigen TCP-Channel weitergeleitet werden. UseSession überschreibt die Anweisung ServerConnPool, die ermöglicht, dass Anforderungen von unterschiedlichen Clients dieselbe persistente Verbindung zum Back-End-Server verwenden können.

Die Parameteroption JunctionPrefix der Anweisung Proxy wird zusammen mit dem Plug-in JunctionRewrite verwendet. Nähere Informationen hierzu finden Sie in den Abschnitten „Umschreiben von Junctions aktivieren (optional)" auf Seite 48 und „Junction mit der Option JunctionPrefix definieren (empfohlene Methode)" auf Seite 48.

Format

```
Proxy Anforderungsschablone Pfad_des_Zielservers [[IP]:Port]
      [UseSession] [JunctionPrefix:URL-Präfix]
```

Diese Anweisung übergibt die Anforderung an einen fernen Server. Beispielsweise bewirkt die folgende Anweisung, dass alle Anforderungen an den angegebenen URL weitergeleitet werden:

```
Proxy /*   http://Name.des.Proxy-Servers/*
```

Verwenden Sie für einen geschützten Reverse Proxy die folgende Anweisung:

```
Proxy /*  https://Name.des.Proxy-Servers/*
```

Wenn der Proxy-Server weniger restriktiv funktionieren soll, entfernen Sie die Kommentarzeichen vor den folgenden Anweisungen in der Konfigurationsdatei. (Diese Anweisungen können allerdings ein Sicherheitsproblem verursachen, wenn der Proxy als Reverse Proxy konfiguriert ist.)

```
Proxy http:*
Proxy ftp:*
Proxy gopher:*
```

Das folgende Beispiel zeigt die Verwendung der Option UseSession in der Anweisung Proxy:

```
Proxy  /abc/*   http://server1/default/abc/*  :80  UseSession
```

Wenn die Clientanforderung an Port 80 eingeht und der URL in der Clientanforderung dem Muster /abc/* entspricht, wird der URL http://server1/default/abc/* zugeordnet.

Das folgende Beispiel zeigt das Format der Option JunctionPrefix in der Anweisung Proxy:

```
Proxy URL-Muster1 URL_Muster2 JunctionPrefix:URL-Präfix
```

Nähere Informationen zur Verwendung der Parameteroption JunctionPrefix finden Sie in den Abschnitten „Umschreiben von Junctions aktivieren (optional)" auf Seite 48 und „Junction mit der Option JunctionPrefix definieren (empfohlene Methode)" auf Seite 48.

Standardwerte

Keiner.

ProxyAccessLog - Name und Pfad für die Proxy-Zugriffsprotokolldatei angeben

Mit dieser Anweisung können Sie den Pfad und Namen der Datei angeben, in der der Server die Zugriffsstatistik für Proxy-Anforderungen protokollieren soll. Standardmäßig schreibt der Server jedes Mal einen Eintrag in dieses Protokoll geschrieben, wenn der Server für eine Clientanforderung als Proxy-Server auftritt. Sie können die Anweisung NoLog verwenden, falls Anforderungen von bestimmten Clients nicht protokolliert werden sollen.

Wenn der Server aktiv ist, startet er täglich um 00:00 Uhr eine neue Protokolldatei. Andernfalls startet der Server eine neue Protokolldatei, wenn er zum ersten Mal am Tag gestartet wird. Beim Erstellen der Datei verwendet der Server den von Ihnen angegebenen Dateinamen und hängt an diesen ein Suffix für das Datum oder eine Erweiterung an. Für das Datumssuffix oder die Erweiterung wird das Format *TTMmmJJJJ* verwendet, wobei *Mmm* die ersten drei Buchstaben des Monatsnamens, *TT* der Tag des Monats und *JJJJ* das Jahr sind.

Es ist sinnvoll, alte Protokolldateien zu löschen, weil sie viel Speicher auf dem Festplattenlaufwerk belegen können.

Format

```
ProxyAccessLog Pfad/Datei
```

Standardwerte

- **Linux- und UNIX-Systeme:** `ProxyAccessLog /opt/ibm/edge/cp/server_root/logs/proxy`
- **Windows-Systeme:** `ProxyAccessLog Laufwerk:\Programme\IBM\edge\cp\logs\proxy`

ProxyAdvisor - Bereitstellung von Proxy-Anforderungen anpassen

Mit dieser Anweisung können Sie eine angepasste Anwendung angeben, die der Server während des Schritts "Proxy Advisor" aufrufen soll. Dieser Code verarbeitet die Anforderung.

Format

```
ProxyAdvisor /Pfad/Datei:Funktionsname
```

/Pfad/Datei
Der vollständig qualifizierte Dateiname des kompilierten Programms.

Funktionsname
Der Name der Anwendungsfunktion in Ihrem Programm.

Beispiel:

```
ProxyAdvisor /api/bin/customadvise.so:proxyadv
```

Standardwert

Keiner.

ProxyForwardLabels - PICS-Filterung definieren

Verwenden Sie die Anweisung ProxyForwardLabels, um für den Proxy-Server und für den Client oder für zwei Proxy-Server in einer Proxy-Hierarchie eine PICS-Filterung zu definieren.

Wird ProxyForwardLabels auf on gesetzt, generiert der Proxy-Server HTTP-Header des Typs PICS-Label: für alle gefundenen PICS-Kennsätze, einschließlich der Kennsätze vom Ursprungsserver, Kennsätze von Kennsatzbüros, Kennsätze aus dem Kennsatz-Cache von Caching Proxy sowie Kennsätze von Plug-ins, die Kennsätze bereitstellen.

Wird ProxyForwardLabels auf Off gesetzt, werden keine HTTP-Header des Typs PICS-Label: generiert.

Format

```
ProxyForwardLabels {On | Off}
```

Standardwert

```
ProxyForwardLabels Off
```

ProxyFrom - Einen Client mit einem Header des Typs From: angeben

Mit dieser Anweisung können Sie einen Header des Typs From: generieren. Dieser Header wird normalerweise zur Angabe der E-Mail-Adresse des Proxy-Administrators verwendet.

Format

ProxyFrom *E-Mail-Adresse*

Beispiel

Die Einstellung ProxyFrom webmaster@proxy.ibm.com bewirkt folgende Änderungen im Header:

Original-Header

```
Location: http://www.ibm.com/
Last Modified: Tue 5 Nov 1997 10:05:39
GMT
Pragma: no-cache
```

Geänderter Header

```
Location: http://www.ibm.com/
Last Modified: Tue 5 Nov 1997 10:05:39
GMT
From: webmaster@proxy.ibm.com
Pragma: no-cache
```

Standardwert

Keiner.

ProxyIgnoreNoCache - Anforderung zum erneuten Laden ignorieren

Mit dieser Anweisung können Sie festlegen, wie der Server reagiert, wenn Benutzer im Browser die Schaltfläche **Aktualisieren** anklicken. Ist die Anweisung ProxyIgnoreNoCache auf on gesetzt, dann wird die Seite bei hoher Serverauslastung nicht vom Ursprungsserver angefordert, sondern eine Kopie der Datei wird aus dem Cache bereitgestellt, falls vorhanden. Der Server ignoriert im Wesentlichen den vom Browser gesendeten Header Pragma: no-cache.

Format

```
ProxyIgnoreNoCache  {on | off}
```

Standardwert

```
ProxyIgnoreNoCache off
```

ProxyPersistence - Persistente Verbindungen zulassen

Geben Sie mit dieser Anweisung an, ob eine persistente Verbindung zum Client bestehen soll. Eine persistente Verbindung reduziert die Wartezeit für Benutzer und verringert die CPU-Auslastung auf dem Proxy-Server. Sie erfordert jedoch eine größere Anzahl Ressourcen. Für eine persistente Verbindung sind zusätzliche Threads und folglich zusätzlicher Speicher im Proxy-Server erforderlich.

Persistente Verbindungen dürfen in einer Konfiguration mit Proxy-Servern auf mehreren Ebenen nicht verwendet werden, wenn ein Proxy-Server HTTP/1.1 nicht unterstützt.

Format

```
ProxyPersistence {on | off}
```

Standardwert

```
ProxyPersistence on
```

ProxySendClientAddress - Den Header "Client IP:" generieren

Legen Sie mit dieser Anweisung fest, ob der Proxy die IP-Adresse des Client an den Zielserver weiterleitet.

Format

```
ProxySendClientAddress  {Client-IP: | OFF}
```

Beispiel

Die Anweisung `ProxySendClientAddress Client-IP:` bewirkt folgende Änderungen der Header:

Original-Header

```
Location: http://www.ibm.com/
Last Modified: Tue 5 Nov 1997 10:05:39
GMT
Pragma: no-cache
```

Geänderter Header

```
Location: http://www.ibm.com
Last Modified: Tue 5 Nov 1997 10:05:39
GMT
Client-IP: 0.67.199.5
Pragma: no-cache
```

Standardwert

Keiner.

ProxyUserAgent - Die Zeichenfolge "User Agent" ändern

Legen Sie mit dieser Anweisung eine Zeichenfolge für den `User Agent` (Benutzeragent) fest, die die vom Client gesendete Zeichenfolge ersetzt. Dadurch wird beim Besuch von Websites eine größere Anonymität erreicht. Allerdings besitzen einige Websites angepasste Seiten, die auf der Zeichenfolge `User Agent` basieren. Bei Verwendung der Anweisung ProxyUserAgent werden diese angepassten Seiten nicht angezeigt.

Format

```
ProxyUserAgent Produktname/Version
```

Beispiel

Die Anweisung `ProxyUserAgent Caching Proxy/6.0` bewirkt folgende Änderungen im Header:

Original-Header	Geänderter Header
`Location: http://www.ibm.com/`	`Location: http://www.ibm.com`
`Last Modified: Tue 5 Nov 1997 10:05:39 GMT`	`Last Modified: Tue 5 Nov 1997 10:05:39 GMT`
`User Agent: Mozilla/ 2.02 OS2`	**`User Agent: Caching Proxy/6.0`**
`Pragma: no-cache`	`Pragma: no-cache`

Standardwert

Keiner.

ProxyVia - Format des HTTP-Headers angeben

Mit dieser Anweisung können Sie das Format des HTTP-Header festlegen. Für diese Anweisung gibt es vier mögliche Werte. Ist ProxyVia auf `Full` gesetzt, fügt Caching Proxy in die Anforderung oder in die Antwort einen Via-Header ein. Ist bereits ein Via-Header im Datenstrom vorhanden, fügt Caching Proxy die Hostdaten an das Ende des Datenstroms an. Ist diese Anweisung auf `Set` gesetzt, setzt Caching Proxy den Via-Header auf die Hostdaten. Ist der Via-Header bereits im Datenstrom enthalten, wird er von Caching Proxy entfernt. Wird diese Anweisung auf `Pass` gesetzt, werden von Caching Proxy alle Header-Daten unverändert weitergeleitet. Bei Angabe von `Block` für die Anweisung werden von Caching Proxy keine Via-Header weitergeleitet.

Format

```
ProxyVia {Full | Set | Pass | Block}
```

Beispiel

```
ProxyVia Pass
```

Standardwert

```
ProxyVia Full
```

ProxyWAS - Festlegen, dass Anforderungen an WebSphere Application Server gesendet werden

Die Zuordnungsanweisung ProxyWAS funktioniert genauso wie die Proxy-Anweisung, legt für Caching Proxy jedoch zusätzlich fest, dass übereinstimmende Anweisungen an einen WebSphere Application Server gesendet werden. Beispiele zur Verwendung dieser Anweisung finden Sie im Abschnitt „Proxy - Proxy-Protokolle oder einen Reverse Proxy angeben” auf Seite 261.

Format

```
ProxyWAS Anforderungsschablone Pfad_des_Zielservers [UseSession]
        [JunctionPrefix:URL-Präfix]
```

Standardwert

Keiner.

PureProxy - Dedizierten Proxy inaktivieren

Geben Sie mit dieser Anweisung an, ob der Server als Proxy-Server oder als Proxy- und Inhaltsserver auftritt. Es wird empfohlen, Caching Proxy ausschließlich als Proxy-Server zu verwenden.

Format

```
PureProxy {on | off}
```

Standardwert

```
PureProxy on
```

PurgeAge - Altersgrenze für ein Protokoll angeben

Geben Sie mit dieser Anweisung das Alter eines Protokolls in Tagen an, bei dessen Erreichen das Protokoll gelöscht wird. Ist PurgeAge auf `0` gesetzt, wird das Protokoll niemals gelöscht.

Anmerkung: Das Plug-in löscht niemals das Protokoll für den laufenden oder den vergangenen Tag.

Format

```
PurgeAge Zahl
```

Standardwert

```
PurgeAge 7
```

Zugehörige Anweisungen

- „CompressAge - Zeitpunkt für die Komprimierung von Protokollen angeben" auf Seite 201
- „CompressDeleteAge - Zeitpunkt zum Löschen von Protokollen angeben" auf Seite 202
- „CompressCommand - Komprimierungsbefehl und zugehörige Parameter angeben" auf Seite 201
- „Midnight - API-Plug-in für die Archivierung von Protokollen angeben" auf Seite 242
- „LogArchive - Verhalten der Protokollarchivierung angeben" auf Seite 235
- „PurgeSize - Begrenzung für die Größe des Protokollarchivs angeben"

PurgeSize - Begrenzung für die Größe des Protokollarchivs angeben

Geben Sie mit dieser Anweisung die zulässige Größe der Protokolldateien in Megabyte fest, bei deren Überschreiten das Protokollarchiv gelöscht wird. Wird die Anweisung PurgeSize auf `0` gesetzt, besteht keine Größenbeschränkung, und es werden keine Dateien gelöscht.

Die Einstellung für PurgeSize bezieht sich auf *alle* Protokolle einer Protokollart. Werden z. B. Fehler protokolliert (d. h., wurde in die Konfigurationsdatei ein Error-Log-Eintrag geschrieben) und ist PurgeSize auf 10 MB gesetzt, berechnet Caching Proxy die Größen aller Fehlerprotokolle, addiert diese und löscht anschließend Protokolle, bis die Gesamtgröße aller Protokolle unter 10 MB liegt.

Anmerkung: Das Plug-in löscht niemals das Protokoll für den laufenden oder den vergangenen Tag. Beim Löschen von Protokolldateien werden die ältesten Protokollen gelöscht. Dieser Vorgang wird so lange fortgesetzt, bis die Größe der Protokolldateien jeder Protokollart kleiner oder gleich dem Wert ist, der mit der Anweisung PurgeSize (in Megabyte) definiert wurde.

Format

```
PurgeSize Anzahl_MB
```

Standardwert

```
PurgeSize 0
```

Zugehörige Anweisungen

- „CompressAge - Zeitpunkt für die Komprimierung von Protokollen angeben" auf Seite 201
- „CompressDeleteAge - Zeitpunkt zum Löschen von Protokollen angeben" auf Seite 202
- „CompressCommand - Komprimierungsbefehl und zugehörige Parameter angeben" auf Seite 201
- „LogArchive - Verhalten der Protokollarchivierung angeben" auf Seite 235
- „Midnight - API-Plug-in für die Archivierung von Protokollen angeben" auf Seite 242
- „PurgeAge - Altersgrenze für ein Protokoll angeben" auf Seite 267

RCAConfigFile - Aliasnamen für ConfigFile angeben

Mit dieser Anweisung können Sie den Namen und die Position der RCA-Konfigurationsdatei (RCA = Remote Cache Access) angeben.

Anmerkung: Die RCA-Konfigurationsdatei wurde in die Datei ibmproxy.conf eingefügt. Aus Gründen der Rückwärtskompatibilität wird RCAConfigFile als Aliasname für ConfigFile unterstützt.

Format

```
RCAConfigFile /etc/Dateiname
```

Beispiel

```
RCAConfigFile /etc/user2rca.conf
```

Standardwert

```
RCAConfigFile /etc/rca.conf
```

RCAThreads - Anzahl Threads pro Port angeben

Geben Sie mit dieser Anweisung die Anzahl Threads an, die an einem RCA-Port aktiv sind.

Format

```
RCAThreads Anzahl_Threads
```

Beispiel

```
RCAThreads 50
```

Standardwert

```
MaxActiveThreads x [(ArraySize -1) / (2 x ArraySize -1)]
```

ReadTimeout - Zeitlimit für eine Verbindung angeben

Geben Sie mit dieser Anweisung das zulässige Zeitlimit an, in dem keine Netzaktivitäten stattfinden können, ohne dass die Netzverbindung abgebrochen wird.

Format

```
ReadTimeout Zeit
```

Standardwert

```
ReadTimeout 5 minutes
```

Redirect - Schablone für Anforderungen angeben, die an einen anderen Server gesendet werden

Legen Sie mit dieser Anweisung eine Schablone für Anforderungen fest, die akzeptiert und an einen anderen Server gesendet werden sollen. Wenn eine Anforderung mit einer Schablone einer Redirect-Anweisung übereinstimmt, wird diese Anforderung nicht mehr mit Schablonen anderer Anweisungen in der Konfigurationsdatei verglichen.

Format

```
Redirect Anforderungsschablone URL  [Server-IP-Adresse | Hostname]
```

Anforderungsschablone
Eine Schablone für Anforderungen, die der Server an einen anderen Server senden soll.

In der Schablone kann ein Stern (*) als Platzhalterzeichen verwendet werden. Für das Tilde-Zeichen (~) direkt nach dem Schrägstrich (/) muss eine exakte Übereinstimmung bestehen. Ein Platzhalterzeichen wird nicht als Übereinstimmung gewertet.

URL
Die URL-Anforderung, die der Server an einen anderen Server sendet. Die Antwort auf diese Anforderung geht ohne die Meldung, dass sie nicht von Ihrem Server stammt, an den ursprünglichen Requester.

URL muss die Angaben eines Protokolls enthalten sowie den Namen des Servers, an den die Anforderung gesendet werden soll. Er kann auch einen Pfad oder einen Dateinamen enthalten. Wird in der *Anforderungsschablone* ein Platzhalterzeichen verwendet, kann im Pfad oder Dateinamen im *URL* ebenfalls ein Platzhalterzeichen verwendet werden. Der Teil der Anforderung, der mit dem Platzhalter in der *Anforderungsschablone* übereinstimmt, wird an die Stelle des Platzhalters im *URL* eingesetzt.

[*Server-IP-Adresse* | *Hostname*]
Falls Sie mehrere IP-Adressen oder virtuelle Hosts verwenden, geben Sie mit diesem Parameter eine IP-Adresse oder einen Hostnamen an. Der Server verwendet die Anweisung nur für Anforderungen an diese IP-Adresse oder für diesen Host. Bei einer IP-Adresse ist dies die Adresse der Netzverbindung des Servers und nicht die Adresse des Clients, von dem die Anforderung stammt.

Sie können eine IP-Adresse (z. B. `240.146.167.72`) oder einen Hostnamen (z. B. `hostA.bcd.com`) angeben.

Dieser Parameter ist optional. Ohne diesen Parameter verwendet der Server die Anweisung für alle Anforderungen ungeachtet der IP-Adresse, unter der die Anforderungen eingehen, oder des Hostnamens im URL.

Für die IP-Adresse eines Servers kann kein Platzhalterzeichen verwendet werden.

Beispiele

- Im folgenden Beispiel sendet der Server alle Anforderungen, die mit `/chief/stuff/` beginnen, an das Verzeichnis wahoo des Servers www.other.org server.

  ```
  Redirect /chief/stuff/* http://www.other.org/wahoo/*
  ```

- Die folgenden Beispiele verwenden den optionalen Parameter IP-Adresse. Wenn der Server Anforderungen empfängt, die mit `/stuff/` beginnen, leitet er die Anforderung an andere Server um, wobei er die IP-Adresse der Netzverbindung verwendet, an der die Anforderung empfangen wurde. Anforderungen, die an der Adresse 240.146.167.72 empfangen werden, sendet der Server an das Verzeichnis wahoo des Servers www.chief.org. Anforderungen, die über eine Verbindung zur Adresse 0.83.100.45 empfangen werden, sendet der Server an das Verzeichnis pound des Servers the www.dawg.com.

  ```
  Redirect  /stuff/*   http://www.chief.org/wahoo/*   240.146.167.72
  Redirect  /stuff/*   http://www.dawg.com/pound/*    0.83.100.45
  ```

- Die folgenden Beispiele verwenden den optionalen Parameter IP-Adresse. Wenn der Server Anforderungen empfängt, die mit `/stuff/` beginnen, leitet er die Anforderung auf der Basis des Hostnamens im URL an andere Server um. Gehen Anforderungen für hostA ein, sendet der Server die Anforderung an das Verzeichnis wahoo des Servers www.chief.org. Gehen Anforderungen für hostB ein, sendet der Server die Anforderung an das Verzeichnis pound des Servers www.dawg.com.

  ```
  Redirect /stuff/* http://www.chief.org/wahoo/* hostA.bcd.com
  Redirect /stuff/* http://www.dawg.com/pound/* hostB.bcd.com
  ```

Standardwert

Keiner.

ReversePass - Automatisch umadressierte Anforderungen abfangen

Die Zuordnungsanweisung ReversePass untersucht den Datenstrom der Serverantwort auf direkte Anforderungen, die im Rahmen einer automatischen Umadressierung umgeschrieben wurden. Wenn ein Server einen HTTP-Code der Klasse 3*xx* sendet (z. B. 301, d. h. permanent verschoben, oder 303, d. h. siehe andere Adresse), dann sendet der Server in der Regel eine Nachricht mit der Antwort, in der der anfordernde Client angewiesen wird, künftige Anforderungen an den richtigen URL und die richtige IP-Adresse zu senden. In einer Konfiguration mit einem Reverse Proxy kann eine Umadressierungsnachricht vom Ursprungsserver dazu führen, dass Client-Browser den Proxy-Server bei nachfolgenden Anforderungen übergehen. Um zu verhindern, dass Clients direkt mit dem Ursprungsserver kommunizieren, sollte die Anweisung ReversePass verwendet werden, mit der Anforderungen, die speziell an den Ursprungsserver gerichtet sind, abgefangen werden.

Anders als andere Zuordnungsanweisungen, die den Anforderungsdatenstrom verarbeiten, vergleicht die Anweisung ReversePass ihre Schablone mit dem Antwortdatenstrom. Der Antwortdatenstrom ist die Antwort, die der Proxy-Server vom Ursprungsserver abruft und an den Client sendet.

Format

```
ReversePass umgeschriebener_URL Proxy-URL [Host:Port]
```

Mit der Option *Host*:*Port* kann der Proxy-Server basierend auf dem Hostnamen und Port des Back-End-Servers eine jeweils unterschiedliche ReversePass-Regel anwenden.

Beispiele

- Die folgende Beispielanweisung verhindert direkte Anforderungen an den Ursprungsserver:

  ```
  ReversePass  http://backend.company.com:9080/*  http://edge.company.com/*
  ```

  Port 9080 ist der Standard-Port für Application Service at the Edge. Diese Art Anforderung könnte generiert werden, wenn der ursprüngliche Anwendungsserver einen Code vom Typ 3*xx* an den Client zurückgegeben hat.
- Mit der folgenden Beispielanweisung werden Anforderungen abgefangen, die über den Code 301 vom Edge-Anwendungsserver umadressiert wurden.

  ```
  ReversePass  http://edge.company.com:9080/*  http://edge.company.com/*
  ```

Anmerkung: Der Inhalt des Musters *Proxy-URL* muss bis zum Platzhalterzeichen (*) exakt mit den Informationen übereinstimmen, die der Back-End-Server im Header location sendet. Andernfalls schlägt die Anweisung fehl.

Standardwert

Keiner.

RewriteSetCookieDomain — Umzuschreibendes Domänenmuster angeben

Mit dieser Anweisung können Sie das Domänenmuster angeben, das umgeschrieben werden muss. Die Anweisung setzt die Domäne von *Domänenmuster1* in *Domänenmuster2* um.

Format

```
RewriteSetCookieDomain Domänenmuster1 Domänenmuster2
```

Beispiel

```
RewriteSetCookieDomain .internal.com .external.com
```

Standardwert

Keiner.

Zugehörige Anweisungen

- „JunctionRewriteSetCookiePath — Bei Verwendung des Plug-in JunctionRewrite die Pfadoption im Header Set-Cookie umschreiben” auf Seite 230

RTSPEnable - RTSP-Umleitung aktivieren

Mit dieser Anweisung wird die RTSP-Umleitung aktiviert bzw. inaktiviert. Mögliche Optionen sind `on` und `off`.

Format

```
RTSPEnable {on | off}
```

Beispiel

```
RTSPEnable  on
```

Standardwert

Keiner.

rtsp_proxy_server - Server für die Umleitung angeben

Mit dieser Anweisung werden die RTSP-Proxy-Server angegeben, die umgeleitete Anforderungen erhalten sollen. Für verschiedene Datenstromtypen können verschiedene Server angegeben werden. Das Format der Anweisung lautet wie folgt:

```
rtsp_proxy_server DNS-Adresse_des_Servers[:Port] Standardrang [Liste_der_MIME-Typen]
```

Beispiel

```
rtsp_proxy_server    rproxy.meinefirma.com:554     1
rtsp_proxy_server    fw1.meinefirma.com:554        2
rtsp_proxy_server    fw1.meinefirma.com:555        3
rtsp_proxy_server    fw2.meinefirma.com:557        4
```

Standardwert

Keiner.

rtsp_proxy_threshold - Anzahl Anforderungen vor der Umleitung in einen Cache angeben

Mit dieser Anweisung wird festgelegt, wie viele Anforderungen empfangen werden sollen, bevor eine RTSP-Anfrage an einen Proxy-Server anstatt an den Ursprungsserver umgeleitet wird. RealNetworks leiten Cache-Datenströme bei der ersten Anforderung weiter, und das Caching erfordert anfangs die doppelte Bandbreite, die beim Empfang eines Datenstroms belegt wird. Wird ein Schwellenwert größer als 1 angegeben, wird verhindert, dass Anforderungen, die nur einmal gestellt werden, im Cache gespeichert werden. Das Format der Anweisung lautet wie folgt:

```
rtsp_proxy_threshold Anzahl_Treffer
```

Beispiel

```
rtsp_proxy_threshold 5
```

Standardwert

Keiner.

rtsp_url_list_size - Anzahl URLs im Proxy-Speicher angeben

Mit dieser Anweisung wird die Anzahl der eindeutigen URLs angegeben, die zur Umleitung im Speicher verbleiben. Der Proxy-Server ermittelt anhand dieser Liste, ob ein bestimmter URL bereits vorgekommen ist. Umfangreiche Listen verbessern die Fähigkeit des Proxy-Servers, eine nachfolgende Anforderung an denselben Proxy-Server zu senden, der die vorherige Anforderung erhalten hat. Jeder Listeneintrag belegt jedoch etwa 16 Bytes im Speicher.

Format

`rtsp_url_list_size` *`Größe_der_Liste`*

Beispiel

```
rtsp_url_list_size 8192
```

Standardwert

Keiner.

ScriptTimeout - Zeitlimiteinstellung für Scripts angeben

Mit dieser Anweisung können Sie die maximale Ausführungsdauer für ein CGI-Programm festlegen, das vom Server gestartet wird. Nach Ablauf dieser Zeit beendet der Server das Programm. Auf Linux- und UNIX-Plattformen wird hierfür das Signal KILL verwendet.

Geben Sie die Zeitspanne in einer beliebigen Kombination aus Stunden, Minuten und Sekunden an.

Format

`ScriptTimeout` *`Zeitlimit`*

Standardwert

```
ScriptTimeout 5 minutes
```

SendHTTP10Outbound - Protokollversion für weitergeleitete Anforderungen angeben

Legen Sie mit dieser Anweisung fest, dass Anforderungen, die von Caching Proxy an einen Downstream-Server gesendet werden, das Protokoll HTTP Version 1.0 verwenden müssen. (Ein *Downstream*-Server ist ein anderer Proxy-Server in einer Kette von Proxy-Servern oder ein Ursprungsserver, der die Anforderung verarbeiten wird.)

Wird diese Anweisung verwendet, legt Caching Proxy das Protokoll HTTP 1.0 als Protokoll für die Anforderungen fest. In diesem Fall werden an den Downstream-Server nur Funktionen aus dem Funktionsumfang von HTTP 1.0 sowie bestimmte Funktionen von HTTP 1.1 gesendet. Zu den Letzteren gehören zum Beispiel Header zur Steuerung der Cache-Funktion (cache-control), die durch die meisten HTTP-1.0-Server unterstützt werden. Verwenden Sie diese Anweisung, falls Sie einen Downstream-Server einsetzen, der Anforderungen nach dem Protokoll HTTP 1.1 nicht ordnungsgemäß verarbeiten kann.

Wird die Anweisung SendHTTP10Outbound *nicht* angegeben, legt Caching Proxy für die Anforderungen das Protokoll HTTP 1.1 fest. Die Funktionalität von HTTP 1.1, z. B. persistente Verbindungen, darf in der Anforderung ebenfalls verwendet werden.

Format

```
SendHTTP10Outbound URL-Muster
```

Beispiele

Diese Anweisung kann auch mehrfach angegeben werden, zum Beispiel:

```
SendHTTP10Outbound http://www.hosta.com/*
SendHTTP10Outbound http://www.hostb.com/*
```

Aus Gründen der Abwärtskompatibilität wird die frühere Syntax von SendHTTP10Outbound wie folgt behandelt:

- `SendHTTP10Outbound on` wird behandelt wie `SendHTTP10Outbound *` .
- `SendHTTP10Outbound off` wird ignoriert.

Anmerkung: Ist sowohl `SendHTTP10Outbound off` als auch `SendHTTP10Outbound URL-Muster` angegeben, wird `SendHTTP10Outbound off` ignoriert, aber eine Warnung wird zurückgegeben.

Standardwert

Keiner.

SendRevProxyName - Hostnamen des Caching Proxy im Header HOST angeben

Bei Verwendung als Reverse Proxy empfängt Caching Proxy HTTP-Anforderungen von einem Client und sendet die Anforderungen an den Ursprungsserver. Standardmäßig schreibt Caching Proxy den Hostnamen des Ursprungsservers in den Header HOST der Anforderung, die er an den Ursprungsserver sendet. Ist die Anweisung SendRevProxyName auf ″yes″ gesetzt, schreibt Caching Proxy stattdessen seinen eigenen Hostnamen in den Header HOST. Diese Anweisung kann dazu verwendet werden, eine spezielle Konfiguration für Back-End-Server zu aktivieren, weil damit eine Anforderung an den Ursprungsserver immer so angezeigt wird, als würde sie vom Proxy-Server stammen, selbst dann, wenn sie von einem Back-End-Server an einen anderen umgeleitet wurde.

Diese Anweisung weicht in folgendem Punkt von der Zuordnungsanweisung ReversePass ab: Mit der Anweisung ReversePass werden Anforderungen, die eine angegebene Syntax enthalten, abgefangen und durch einen anderen Anforderungsinhalt, den Sie angeben, ersetzt. Durch die Anweisung SendRevProxyName wird lediglich der Hostname des Ursprungsservers durch den Hostnamen von Caching Proxy ersetzt. Diese Anweisung ist nicht sinnvoll bei der Konfiguration von Application Service at the Edge.

Format

```
SendRevProxyName  {yes | no}
```

ServerConnGCRun - Intervall für die Ausführung des Garbage-Collection-Thread festlegen

Mit dieser Anweisung wird das Intervall festgelegt, in dem der Garbage-Collection-Thread ausgeführt wird und nach Serververbindungen sucht, deren Zeitlimit (das mit der Anweisung ServerConnTimeout festgelegt wird) abgelaufen ist. Diese Anweisung sollte nur verwendet werden, wenn die Anweisung ServerConnPool auf `on` gesetzt ist.

Format

ServerConnGCRun *Zeitintervall*

Beispiel

```
ServerConnGCRun 2 minutes
```

Standardwert

```
ServerConnGCRun 2 minutes
```

ServerConnPool - Das Pooling von Verbindungen zum Ursprungsserver festlegen

Mit dieser Anweisung kann der Proxy-Server seine abgehenden Verbindungen an Ursprungsserver in einem Pool zusammenfassen. Wird diese Anweisung auf `on` gesetzt, verbessert sich die Leistung und die Ursprungsserver, die persistente Verbindungen zulassen, können besser genutzt werden. Sie können außerdem mit der Anweisung ServerConnTimeout festlegen, wie lange eine nicht verwendete Verbindung beibehalten werden soll.

Anmerkung: Diese Anweisung sollte am besten in einer kontrollierten Umgebung aktiviert werden. Andernfalls kann sich die Leistung verschlechtern, wenn ein Forward Proxy verwendet wird oder wenn die Ursprungsserver nicht kompatibel sind mit HTTP 1.1.

Format

```
ServerConnPool {on | off}
```

Standardwert

```
ServerConnPool off
```

ServerConnTimeout - Maximale Zeit der Inaktivität festlegen

Mit dieser Anweisung können Sie das Zeitlimit für Netzinaktivität festlegen, nach dessen Ablauf die Netzverbindung abgebrochen wird. Diese Anweisung sollte nur verwendet werden, wenn die Anweisung ServerConnPool auf `on` gesetzt ist.

Format

ServerConnTimeout *Zeitangabe*

Beispiel

```
ServerConnTimeout 30 seconds
```

Standardwert

```
ServerConnTimeout 10 seconds
```

ServerInit - Schritt ″Server Initialization″ anpassen

Mit dieser Anweisung können Sie eine angepasste Anwendungsfunktion angeben, die der Server während der Ausführung der Initialisierungsroutinen aufrufen soll. Dieser Code wird vor dem Lesen einer Clientanforderung und nach jedem Neustart des Servers ausgeführt.

Werden die GoServe-Module in den Schritten ″PreExit″ oder ″Service″ verwendet, muss das Modul `gosclone` an dieser Stelle aufgerufen werden.

Format

ServerInit */Pfad/Datei*:*Funktionsname* [*Initialisierungszeichenfolge*]

/Pfad/Datei
Der vollständig qualifizierte Dateiname des kompilierten Programms einschließlich der Erweiterung.

Funktionsname
Der Name der Anwendungsfunktion in Ihrem Programm.

Initialisierungszeichenfolge
Optional. Legt eine Textzeichenfolge fest, die an die Anwendungsfunktion übergeben wird.

Beispiel

```
ServerInit /ics/api/bin/icsext05.so:svr_init
```

Standardwert

Keiner.

ServerRoot - Das Verzeichnis angeben, in dem das Serverprogramm installiert ist

Mit dieser Anweisung können Sie das Verzeichnis angeben, in dem das Serverprogramm installiert ist (das aktuelle Arbeitsverzeichnis des Servers). Die Protokollierungsanweisungen verwenden das aktuelle Arbeitsverzeichnis als Standardstammverzeichnis, wenn relative Pfadnamen verwendet werden.

Auf Windows-Systemen wird das Verzeichnis während der Installation angegeben.

Format

ServerRoot *Verzeichnispfad*

Standardwerte

- **Linux- und UNIX-Systeme:** `ServerRoot /opt/ibm/edge/cp/server_root/`
- **Windows-Systeme:** `C:\Programme\IBM\edge\cp\bin\`

Anmerkung: Sie können den Standardwert zwar ändern, aber dies hat auf die Art und Weise, in der der Server Anforderungen bearbeitet, keinen Einfluss.

Anmerkung: Die Regeln PASS und EXEC können von diesem Verzeichnis unabhängig sein.

ServerTerm - Schritt ″Server Termination″ anpassen

Mit dieser Anweisung können Sie eine angepasste Anwendungsfunktion angeben, die der Server im Schritt ″Server Termination″ aufrufen soll. Dieser Code wird beim ordnungsgemäßen Beenden und bei jedem Neustart des Servers ausgeführt. Er ermöglicht die Freigabe der Ressourcen, die durch eine PreExit-Anwendungsfunktion zugewiesen wurden.

Format

ServerTerm */Pfad/Datei*:*Funktionsname*

/Pfad/Datei
Der vollständig qualifizierte Dateiname des kompilierten Programms einschließlich der Erweiterung.

Funktionsname
Der Name der Anwendungsfunktion in Ihrem Programm.

Beispiel
```
ServerTerm /ics/api/bin/icsext05.so:shut_down
```

Standardwert
Keiner.

Service - Schritt "Service" anpassen
Mit dieser Anweisung können Sie eine angepasste Anwendungsfunktion angeben, die der Server im Schritt "Service" aufrufen soll. Dieser Code bearbeitet die Clientanforderung. Er sendet beispielsweise eine Datei oder führt das CGI-Programm aus.

Es gibt für diese Anweisung keinen Standardwert. Falls die Anforderung mit einer Service-Regel übereinstimmt (d. h., falls eine in einer Service-Anweisung angegebene Anwendungsfunktion ausgeführt wird), aber von der Funktion HTTP_NOACTION zurückgegeben wird, generiert der Server einen Fehler, und die Anforderung wird zurückgewiesen.

Format
```
Service Anforderungsschablone/Pfad/Datei:Funktionsname
[Server-IP-Adresse | Hostname]
```

Anforderungsschablone
Eine Schablone für Anforderungen, die festlegt, ob eine Anwendungsfunktion aufgerufen wird. Die Schablone kann als Angaben das Protokoll, die Domäne und den Host enthalten, sie darf als vorangestelltes Zeichen einen Schrägstrich (/) verwenden sowie als Platzhalterzeichen einen Stern (*). Beispielsweise sind folgende Schablonen gültig: `/front_page.html`, `http://www.ics.raleigh.ibm.com`, `/pub*`, `/*` und `*`.

/Pfad/Datei
Der vollständig qualifizierte Dateiname des kompilierten Programms einschließlich der Erweiterung.

Funktionsname
Der Name der Anwendungsfunktion in Ihrem Programm.

[*Server-IP-Adresse* **|** *Hostname***]**
Falls Sie mehrere IP-Adressen oder virtuelle Hosts verwenden, legt dieser Parameter fest, ob die Anwendungsfunktion nur für Anforderungen aufgerufen wird, die an einer bestimmten IP-Adresse oder für einen bestimmten Host eingehen.

Für die IP-Adresse eines Servers kann kein Platzhalterzeichen verwendet werden.

Beispiele
```
Service  /index.html /ics/api/bin/icsext05.so:serve_req
Service  /cgi-bin/hexcalc* /ics/api/calculator:HEXcalc*
```

Anmerkung: Soll eine vollständige Pfadübersetzung einschließlich der *Abfragezeichenfolge* erfolgen, muss sowohl im Parameter *Anforderungsschablone* als auch im Parameter *Pfad/Datei:Funktionsname*, wie im zweiten Beispiel dargestellt, ein Stern (*) angegeben werden.

Standardwert
Keiner.

SignificantURLTerminator - Abschlusscode für URL-Anforderungen festlegen

Legen Sie mit dieser Anweisung einen Abschlusscode für URL-Anforderungen fest. Wird in einer Anforderung ein Abschlusscode verwendet, werden nur die Zeichen vor dem Abschlusscode von Caching Proxy ausgewertet, wenn die Anforderung verarbeitet wird oder wenn überprüft wird, ob das Ergebnis bereits im Cache gespeichert ist. Sind mehrere Abschlusscodes definiert, vergleicht Caching Proxy die ankommenden URLs in der Reihenfolge mit den Abschlusscodes, in der diese in der Datei ibmproxy.conf definiert sind.

Format
```
SignificantURLTerminator Abschlusszeichenfolge
```

Beispiel
```
SignificantURLTerminator  &.
```

In diesem Beispiel werden die folgenden beiden Anforderungen als identische Anforderungen behandelt:

```
http://www.exampleURL.com/tx.asp?id=0200&.;x=004;y=001
http://www.exampleURL.com/tx.asp?id=0200&.;x=127;y=034
```

Standardwert
Keiner.

SMTPServer (nur Windows) - Einen SMTP-Server für die Sendmail-Routine festlegen

Legen Sie mit dieser Anweisung den SMTP-Server fest, der von der internen Sendmail-Routine innerhalb des Caching Proxy für Windows verwendet wird. Die folgenden zwei Anweisungen müssen für diese Routine ebenfalls festgelegt werden: „WebMasterEMail - Eine E-Mail-Adresse für den Empfang von ausgewählten Serverberichten festlegen" auf Seite 287 und „WebMasterSocksServer (nur Windows) - Einen Socks-Server für die Sendmail-Routine festlegen" auf Seite 287.

Format
```
SMTPServer IP-Adresse oder Hostname des SMTP-Servers
```

Beispiel
```
SMTPServer meinebox.com
```

Standardwert
```
Keiner.
```

SNMP - SNMP-Unterstützung aktivieren und inaktivieren

Mit dieser Anweisung können Sie die SNMP-Unterstützung aktivieren und inaktivieren.

Format
```
SNMP   {on | off}
```

Standardwert
```
SNMP off
```

SNMPCommunity - Ein Sicherheitskennwort für SNMP angeben

Mit dieser Anweisung können Sie das Kennwort für die Kommunikation zwischen dem DPI-Subagenten des Webservers (DPI = Distributed Protocol Interface) und dem SNMP-Agenten definieren. Der Name der SNMP-Community berechtigt einen Benutzer, die Leistungsvariablen anzuzeigen, die SNMP für eine bestimmte Community von Servern überwacht. Der Systemadministrator definiert, von welchen Servern welche Variablen nach der Eingabe eines Kennwortes angezeigt werden können. Wenn Sie den Namen der SNMP-Community ändern, müssen Sie auch den in der Datei /etc/snmpd.conf angegebenen Community-Namen ändern.

Format

```
SNMPCommunity Name
```

Standardwert

```
SNMPCommunity public
```

SSLCaching - Das Caching für eine sichere Anforderung aktivieren

Mit dieser Anweisung können Sie den Inhalt einer gesicherten Anforderung in den Cache stellen, wenn ein Reverse Proxy verwendet wird. Diese Anweisung konfiguriert das Caching für alle Verbindungen zum Proxy-Server, d. h. Clientverbindungen und Verbindungen zu einem Back-End-Inhaltsserver.

Anmerkung: Die SSL-Anweisungen werden unter SuSE Linux nicht unterstützt.

Format

```
SSLCaching {on | off}
```

Standardwert

```
SSLCaching off
```

SSLCertificate - Schlüsselkennsätze für Zertifikate angeben

Geben Sie mit dieser Anweisung die Schlüsselkennsätze an, mit denen der Proxy-Server festlegt, welches Zertifikat er an den Client senden soll, wenn Caching Proxy als Reverse Proxy für mehrere Domänen auftritt, die eigene SSL-Zertifikate bereitstellen. Außerdem wird damit der Proxy-Server angewiesen, ein PKI-Zertifikat auf der Clientseite zwecks Clientauthentifizierung abzurufen oder nicht abzurufen.

Format

```
SSLCertificate Server-IP/Hostname Zertifikatkennsatz
[NoClientAuth | ClientAuthRequired]
```

Server-IP/Hostname
Sie können eine IP-Adresse (z. B. 204.146.167.72) oder einen Hostnamen (z. B. hostA.raleigh.ibm.com) für den Server angeben, an den die SSL-Anforderung gerichtet ist.

Zertifikatkennsatz
Der Name des Zertifikats, das verwendet werden soll, falls für SSL-Anforderungen, die an die angegebene IP-Adresse oder den angegebenen Hostnamen gerichtet sind, eine Clientauthentifizierung erforderlich ist.

[NoClientAuth | ClientAuthRequired]
Dieses Argument weist den Proxy-Server an, ein PKI-Zertifikat auf der Clientseite abzurufen oder nicht abzurufen.

Beispiele

```
SSLCertificate  www.abc.com      ABCCert
SSLCertificate  204.146.167.72   intABCCert
SSLCertificate  www.xyz.com      XYZCert
SSLCertificate  www.xyz.com      XYZCert      ClientAuthRequired
```

Standardwert

Keiner.

SSLCryptoCard - Die installierte Verschlüsselungskarte angeben

Mit dieser Anweisung können Sie dem Proxy-Server mitteilen, dass eine Verschlüsselungskarte installiert ist und um welche Karte es sich handelt.

Format

```
SSLCryptoCard {rainbowcs | nciphernfast} {on | off}
```

Beispiel

```
SSLCryptoCard rainbowcs on
```

Standardwert

Keiner.

SSLEnable - Angeben, ob an Port 443 gesicherte Anforderungen empfangen werden sollen

Mit dieser Anweisung können Sie festlegen, dass Caching Proxy an Port 443 sichere Anforderungen empfangen soll.

Anmerkung: Die SSL-Anweisungen werden unter SuSE Linux nicht unterstützt.

Format

```
SSLEnable {on | off}
```

Standardwert

```
SSLEnable off
```

SSLForwardPort - Den Port angeben, der für HTTP-SSL-Upgrades verwendet werden soll

Legen Sie mit dieser Anweisung den Port fest, der für HTTP-Anforderungen verwendet werden soll, die Caching Proxy durch Implementierung von SSL in HTTPS-Anforderungen umwandelt. Geben Sie einen anderen Port als den HTTP-Haupt-Port 80 oder den SSL-Haupt-Port 443 an.

Format

SSLForwardPort *Port-Nummer*

Beispiel

```
SSLForwardPort 8888
```

Standardwert

Keiner.

SSLOnly - Empfangs-Threads für HTTP-Anforderungen inaktivieren

Mit dieser Anweisung können Sie Empfangs-Threads für Standard-HTTP-Anforderungen (normalerweise an Port 80 und 8080) inaktivieren, wenn SSL (normalerweise an Port 443) aktiviert ist.

Format

```
SSLOnly    {on | off}
```

Standardwert

```
SSLOnly   off
```

SSLPort — Als HTTPS-Empfangs-Port einen anderen als den Standard-Port angeben

Mit dieser Anweisung können Sie einen anderen als den Standard-HTTP-Port 442 von ibmproxy als HTTPS-Empfangs-Port angeben.

Anmerkung: ibmproxy unterstützt einen HTTPS-Port für jede Instanz, deshalb dürfen mit der Anweisung nicht mehrere HTTPS-Ports angegeben werden. Für die Unterstützung mehrerer HTTPS-Ports müssen Sie mehrere ibmproxy-Instanzen mit unterschiedlichen Dateien `ibmproxy.conf` starten.

Format

```
SSLPort Port-Wert
```

Port-Wert steht für einen ganzzahligen Wert größer als 0. Außerdem muss der *Port-Wert* vom Betriebssystem unterstützt und darf von keine anderen Anwendung verwendet werden.

Beispiel

```
SSLPort 8443
```

Standardwert

443

SSLTunneling - SSL-Tunnelung aktivieren

Mit dieser Anweisung können Sie die Tunnelung von SSL-Anforderungen für jeden Port auf dem Zielhost aktivieren. Setzen Sie diese Anweisung auf "on", um für jeden Port des Zielservers die SSL-Tunnelung zu aktivieren. Wird diese Anweisung auf "off" festgelegt, wird die SSL-Tunnelung nur für die in Proxy-Regeln angegebenen Ports aktiviert. Sind keine Proxy-Regeln für die SSL-Tunnelung angegeben und ist die Anweisung SSLTunneling auf `off` gesetzt, ist die SSL-Tunnelung nicht aktiviert. Ist die Anweisung SSLTunneling auf `on` gesetzt, müssen Sie auch die Methode "CONNECT" mit der Anweisung Enable aktivieren.

Wenn Sie Caching Proxy als Reverse Proxy verwenden, können Sie sich durch Inaktivieren dieser Anweisung (Standardeinstellung) gegen Angriffe auf Sicherheitslücken bei der SSL-Tunnelung schützen.

Anmerkung: Verwenden Sie zur Aktivierung der SSL-Tunnelung für einen bestimmten Port des Zielhost die Anweisung Proxy.

Format
```
SSLTunneling {on | off}
```

Standardwert
```
SSLTunneling off
```

SSLVersion - Die Version von SSL angeben

Geben Sie mit dieser Anweisung an, welche Version von SSL verwendet werden soll: V2, V3 oder alle Versionen. Setzen Sie diese Anweisung auf `V2`, wenn Sie Server verwenden, die SSL Version 3 nicht unterstützen.

Anmerkung: Die SSL-Anweisungen werden unter SuSE Linux nicht unterstützt.

Format
```
SSLVersion  {SSLV2 | SSLV3 | all}
```

Standardwert
```
SSLVersion SSLV3
```

SSLV2Timeout - Die Wartezeit vor dem Ablaufen einer SSLV2-Sitzung angeben

Mit dieser Anweisung geben Sie an, wie lange (in Sekunden) eine SSL-Sitzung (Version 2) inaktiv bleiben kann, bevor sie verfällt.

Anmerkung: Die SSL-Anweisungen werden unter SuSE Linux nicht unterstützt.

Format
`SSLV2Timeout` *Sekunden*

Dabei steht *Sekunden* für einen Wert zwischen `0` und `100`.

Standardwert
```
SSLV2Timeout 100
```

SSLV3Timeout - Die Wartezeit vor dem Ablaufen einer SSLV3-Sitzung angeben

Geben Sie mit dieser Anweisung die Zeit in Sekunden an, während der eine Sitzung mit SSL Version 3 bei Inaktivität warten soll, bevor die Sitzung abläuft.

Anmerkung: Die SSL-Anweisungen werden unter SuSE Linux nicht unterstützt.

Format
`SSLV3Timeout` *Sekunden*

Dabei steht *Sekunden* für einen Wert zwischen 1 und 86400 Sekunden (dies entspricht 1 Tag in Sekunden).

Standardwert
```
SSLV3Timeout 100
```

SuffixCaseSense - Angeben, ob bei Suffixdefinitionen zwischen Groß- und Kleinschreibung unterschieden wird

Mit dieser Anweisung können Sie festlegen, ob der Server zwischen Großbuchstaben und Kleinbuchstaben unterscheiden soll, wenn in den Anweisungen AddClient, AddCharSet, AddType, AddEncoding und AddLanguage Dateisuffixe mit Suffixmustern verglichen werden. Standardmäßig unterscheidet der Server nicht zwischen Großbuchstaben und Kleinbuchstaben.

Format

```
SuffixCaseSense  {on | Off}
```

Standardwert

```
SuffixCaseSense Off
```

TLSV1Enable - Das Protokoll ″Transport Layer Secure″ aktivieren

Mit dieser Anweisung können Sie das Protokoll TLS Version 1 in SSL-Verbindungen aktivieren. Wenn diese Anweisung auf `on` gesetzt ist, sucht die SSL-Verbindung zunächst nach dem Protokoll TLS, dann nach dem Protokoll SSLv3 und zuletzt nach dem Protokoll SSLv2.

Anmerkung: Diese Anweisung funktioniert für Internet Explorer und andere Browser, jedoch nicht für Netscape.(Die Verwendung des Browser Netscape wird für Caching Proxy nicht empfohlen.)

Format

```
TLSV1Enable {on | off}
```

Beispiel

```
TLSV1Enable on
```

Anfangseinstellung in der Konfigurationsdatei

Keiner.

Transmogrifier - Schritt ″Data Manipulation″ anpassen

Mit dieser Anweisung können Sie eine angepasste Anwendungsfunktion abgeben, die der Server im Schritt ″Data Manipulation″ aufrufen soll. Dieser Code stellt drei Anwendungsfunktionen bereit:

- eine Funktion *open* (Öffnen), um vor der Verarbeitung der Daten eine Initialisierung durchzuführen
- eine Funktion *write* (Schreiben) für die Verarbeitung der Daten
- eine Funktion *close* (Schließen), um nach der Verarbeitung der Daten eine Bereinigung durchzuführen
- eine Funktion *error* (Fehler), um aufgetretene Fehler zu melden

Es können mehrere Transmogrifier für jede Instanz des Servers aktiv sein.

Format

```
Transmogrifier /Pfad/Datei:Funktionsname:Funktionsname:Funktionsname
```

/Pfad/Datei
: Der vollständig qualifizierte Dateiname des kompilierten Programms einschließlich der Erweiterung.

Funktionsname
Der Name der Anwendungsfunktion in Ihrem Programm. Sie müssen die Namen der Funktionen für das Öffnen, Schreiben und Schließen angeben.

Beispiel
```
Transmogrifier /ics/bin/icsext05.so:open_data:write_data:close_data
```

Standardwert
Keiner.

TransmogrifiedWarning - Warnung an Client senden
Mit dieser Anweisung können Sie eine Nachricht an den Client senden.

Format
```
transmogrifiedwaning {yes|no}
```

Standardwert
Yes

TransparentProxy - Für Linux oder AIX den transparenten Proxy-Server aktivieren
Nur für Linux und AIX: Legen Sie mit dieser Anweisung fest, ob der Server als transparenter Proxy-Server arbeiten kann.

Anmerkung: Ist die Anweisung TransparentProxy auf `On` gesetzt, wird die Anweisung BindSpecific ignoriert und auf den Standardwert `Off` gesetzt. Da der meiste HTTP-Verkehr über den Port 80 abgewickelt wird, wird dringend empfohlen, dass dieser Port ebenfalls konfiguriert wird.

Format
```
TransparentProxy  {On | Off}
Port 80
```

Standardwert
```
TransparentProxy Off
```

Anmerkung: Nachdem der transparente Proxy-Server gestartet wurde, müssen Sie außerdem die folgenden Befehle als root ausführen, wenn Caching Proxy gestoppt werden soll:

```
ibmproxy -unload
```

Auf Linux-Systemen entfernt dieser Befehl die Firewall-Umleitungsregeln, und auf AIX-Systemen wird damit die Kernel-Erweiterung für transparenten Proxy-Server (Transparent Proxy Kernel Extension) entfernt. Wird dieser Befehl nach dem Stoppen des Servers nicht ausgeführt, wird die Maschine Anforderungen akzeptieren, die nicht für sie bestimmt sind.

UpdateProxy - Die Zieladresse des Cache angeben
Legen Sie mit dieser Anweisung fest, welchen Proxy-Server der Cache-Agent aktualisieren soll. Diese Angabe ist erforderlich, wenn der Cache-Agent einen anderen als den lokalen Proxy-Server, auf dem er läuft, aktualisieren soll. Sie können wahlweise den Port angeben.

Anmerkung: Auf Linux- und UNIX-Plattformen ist diese Anweisung erforderlich, damit der Cache-Agent verwendet werden kann. Wenn Sie für den Proxy-Server nur eine Maschine verwenden, geben Sie den Hostnamen an.

Obwohl der Cache-Agent den Cache eines anderen Servers aktualisieren kann, ist er nicht in der Lage, das Cache-Zugriffsprotokoll von dieser Maschine abzurufen. Aus diesem Grund wird, falls in der Anweisung UpdateProxy ein anderer Host als der lokale Host angegeben ist, die Anweisung LoadTopCached ignoriert.

Format

```
UpdateProxy vollständig_qualifizierter_Hostname_des_Proxy-Servers
```

Beispiel

```
UpdateProxy proxy15.ibm.com:1080
```

Standardwert

Keiner.

UserId - Standard-Benutzer-ID angeben

Mit dieser Anweisung können Sie den Benutzernamen oder die ID angeben, zu dem bzw. der der Server wechseln soll, bevor er auf Dateien zugreift.

Wenn Sie diese Anweisung ändern, müssen Sie den Server manuell stoppen und anschließend erneut starten, damit die Änderung wirksam wird. Der Server erkennt die Änderung erst, wenn Sie ihn erneut starten. (Nähere Informationen hierzu finden Sie in Kapitel 5, „Caching Proxy starten und stoppen", auf Seite 15.)

Anmerkung: Wenn Sie die Standardeinstellungen des Servers für Benutzer-ID, Gruppen-ID oder Protokollverzeichnispfade ändern, müssen Sie die neuen Verzeichnisse erstellen und die für diese Verzeichnisse erforderlichen Berechtigungen und Eigentumsrechte aktualisieren. Damit der Server Daten in ein benutzerdefiniertes Protokollverzeichnis schreiben kann, muss die Berechtigung für dieses Verzeichnis auf 755 gesetzt werden und als Eigner die benutzerdefinierte Benutzer-ID des Servers festgelegt werden. Wenn Sie beispielsweise die Benutzer-ID des Servers vom Standardwert in jdoe und das Standardprotokollverzeichnis in server_root/account ändern, muss das Verzeichnis server_root/account die Berechtigung 755 besitzen, und als Eigner muss jdoe festgelegt sein.

Format

```
UserId  {ID-Name | Nummer}
```

Standardwert

AIX, Linux, Solaris: `UserId nobody`

HP-UX: `UserId www`

V2CipherSpecs - Unterstützte Verschlüsselungsspezifikationen für SSL Version 2 auflisten

Diese Anweisung listet die verfügbare Verschlüsselungsspezifikation für SSL Version 2 auf.

Anmerkung: Die SSL-Anweisungen werden unter SuSE Linux nicht unterstützt.

Format

`V2CipherSpecs` *`Spezifikation`*

Jede Kombination der folgenden Werte ist zulässig. Kein Wert darf zweimal verwendet werden.

- `1` - RC4 US
- `2` - RC4 Export
- `3` - RC2 US
- `4` - RC2 Export
- `6` - DES 56-bit
- `7` - Triple DES US
- `NULL` - die Standardspezifikationen zur Verschlüsselung werden verwendet

Beispiele

- Für USA: `V2CipherSpecs '137624'`
- Für Export: `V2 Cipherspecs '246'`

Standardwert

Keiner (SSL ist standardmäßig inaktiviert).

V3CipherSpecs - Unterstützte Verschlüsselungsspezifikationen für SSL Version 3 auflisten

In dieser Anweisung werden die verfügbaren Spezifikationen zur Verschlüsselung für SSL Version 3 aufgelistet.

Anmerkung: Die SSL-Anweisungen werden unter SuSE Linux nicht unterstützt.

Format

`V3CipherSpecs` *`Spezifikation`*

Es sind die folgenden Werte zulässig:

- `00` - NULL NULL
- `01` - NULL MD5
- `02` - NULL SHA
- `03` - RC4 MD5 Export
- `04` - RC4 MD5 US
- `05` - RC4 SHA US
- `06` - RC2 MD5 Export
- `09` - DES SHA Export
- `0A` - Triple DS SHA US
- `62` - 56-bit DES CBC SHA
- `64` - 56-bit RC4 SHA
- `NULL` - die Standardspezifikationen zur Verschlüsselung werden verwendet

Beispiele

- Für USA: `V3CipherSpecs '0A09060564620403020100'`
- Für Export: `V3Cipherspecs '0906646203020100'`

Standardwert

Keiner (SSL ist standardmäßig inaktiviert).

WebMasterEMail - Eine E-Mail-Adresse für den Empfang von ausgewählten Serverberichten festlegen

Mit dieser Anweisung wird eine E-Mail-Adresse festgelegt, an der ausgewählte Caching-Proxy-Berichte empfangen werden, beispielsweise die Benachrichtigung 30 Tage vor Ablauf eines SSL-Zertifikats. Auf Linux- und UNIX-Systemen muss ein Sendmail-Prozess aktiv sein. Bei Windows-Systemen ist der Sendmail-Prozess im Caching Proxy integriert, so dass kein externer Mail-Server erforderlich ist. Außerdem müssen zwei weitere Anweisungen festgelegt werden: „WebMasterSocksServer (nur Windows) - Einen Socks-Server für die Sendmail-Routine festlegen" und „SMTPServer (nur Windows) - Einen SMTP-Server für die Sendmail-Routine festlegen" auf Seite 278.

Anmerkung: Diese E-Mail-Adresse wird außerdem als anonymes FTP-Kennwort verwendet.

Format

```
WebMasterEMail E-Mail-Adresse_des_Webmaster
```

Beispiel

```
WebMasterEmail webmaster@computer.com
```

Standardwert

```
WebMasterEmail webmaster
```

WebMasterSocksServer (nur Windows) - Einen Socks-Server für die Sendmail-Routine festlegen

Legen Sie mit dieser Anweisung den Socks-Server fest, der von der internen Sendmail-Routine innerhalb des Caching Proxy für Windows verwendet wird. Die folgenden zwei Anweisungen müssen für diese Routine ebenfalls festgelegt werden: „WebMasterEMail - Eine E-Mail-Adresse für den Empfang von ausgewählten Serverberichten festlegen" und „SMTPServer (nur Windows) - Einen SMTP-Server für die Sendmail-Routine festlegen" auf Seite 278.

Format

```
WebMasterSocksServer IP-Adresse oder Hostname des Socks-Servers
```

Beispiel

```
WebMasterSocksServer socks.meinebox.com
```

Standardwert

```
Keiner.
```

Welcome - Die Namen von Begrüßungsdateien angeben

Mit dieser Anweisung können Sie den Namen einer Begrüßungsdatei abgeben, die der Server als Antwort auf Anforderungen suchen soll, die keinen speziellen Dateinamen enthalten. Sie können eine Liste von Begrüßungsdateien erstellen, indem Sie diese Anweisung in Ihrer Konfigurationsdatei mehrfach angeben.

Bei Anforderungen, die keinen speziellen Datei- oder Verzeichnisnamen enthalten, sucht der Server immer im Dateistammverzeichnis nach einer Datei, deren Name mit dem in einer Welcome-Anweisung angegebenen Namen übereinstimmt. Wird eine solche Datei gefunden, wird sie an den Requester zurückgegeben.

Bei Anforderungen, die einen Verzeichnisnamen, aber keinen Dateinamen enthalten, steuert die Anweisung AlwaysWelcome, ob der Server in diesem Verzeichnis nach einer Begrüßungsdatei sucht. Standardmäßig ist AlwaysWelcome auf `On` festgelegt. Damit sucht der Server immer im angeforderten Verzeichnis nach einer Datei, deren Name mit dem in einer Welcome-Anweisung angegebenen Namen übereinstimmt. Wird eine solche Datei gefunden, wird sie an den Requester zurückgegeben.

Findet der Server in einem Verzeichnis mehrere Dateien, die mit Welcome-Anweisungen übereinstimmen, bestimmt die Reihenfolge dieser Welcome-Anweisungen, welche Datei zurückgegeben wird. Dabei wird die Welcome-Anweisung verwendet, die in der Konfigurationsdatei am weitesten oben steht.

Format

`Welcome` *`Dateiname`* [*`Server-IP-Adresse`* | *`Hostname`*]

`Dateiname`
Der Name einer Datei, die als Begrüßungsdatei dienen soll.

[*`Server-IP-Adresse`* | *`Hostname`*]
Falls Sie mehrere IP-Adressen oder virtuelle Hosts verwenden, geben Sie mit diesem Parameter eine IP-Adresse oder einen Hostnamen an. Der Server verwendet die Anweisung nur für Anforderungen an diese IP-Adresse oder für diesen Host. Bei einer IP-Adresse ist dies die Adresse der Netzverbindung des Servers und nicht die Adresse des Clients, von dem die Anforderung stammt.

Sie können eine IP-Adresse (z. B. `240.146.167.72`) oder einen Hostnamen (z. B. `hostA.bcd.com`) angeben.

Dieser Parameter ist optional. Ohne Angabe dieses Parameters verwendet der Server die Anweisung für alle Anforderungen ungeachtet der IP-Adresse, unter der die Anforderungen ankommen, oder des Hostnamens in den URLs.

Für die IP-Adresse eines Servers kann kein Platzhalterzeichen verwendet werden.

Beispiele

- Im folgenden Beispiel werden zwei Begrüßungsseiten definiert. Dabei wird vorausgesetzt, dass die Anweisung AlwaysWelcome auf ihren Standardwert `On` gesetzt ist. Bei Anforderungen, die keinen Dateinamen enthalten, sucht der Server in dem Verzeichnis, das in der Anforderung angegeben ist (oder im Dateistammverzeichnis, falls in der Anforderung kein Datei- oder Verzeichnisname angegeben ist), nach einer Begrüßungsdatei. Der Server sucht zuerst nach einer Datei mit dem Namen letsgo.html. Wenn er keine Datei mit diesem Namen im Verzeichnis findet, sucht der Server nach einer Datei mit dem Namen Welcome.html.

```
Welcome letsgo.html
Welcome Welcome.html
```

- Im folgenden Beispiel sucht der Server auf der Basis der IP-Adresse der Netzverbindung, in der die Anforderung empfangen wurde, nach unterschiedlichen Begrüßungsdateien. Für Anforderungen, die an der Adresse 0.67.106.79 empfangen werden, sucht der Server Begrüßungsdateien mit dem Namen CustomerA.html. Für Anforderungen, die an der Adresse 0.83.100.45 empfangen werden, sucht der Server nach Begrüßungsdateien mit dem Namen CustomerB.html. Geht die Anforderung unter einer anderen IP-Adresse ein, sucht der Server nach der Standardadresse.

  ```
  Welcome CustomerA.html  0.67.106.79
  Welcome CustomerB.html  0.83.100.45
  ```

- Im folgenden Beispiel sucht der Server auf der Basis des Hostnamens im URL nach unterschiedlichen Begrüßungsdateien. Für Anforderungen, die für hostA empfangen werden, sucht der Server nach Begrüßungsdateien mit dem Namen CustomerA.html. Für Anforderungen, die für hostB empfangen werden, sucht der Server nach Begrüßungsdateien mit dem Namen CustomerB.html. Geht die Anforderung für einen anderen Host ein, sucht der Server nach dem Standardhostnamen.

  ```
  Welcome CustomerA.html hostA.bcd.com
  Welcome CustomerB.html hostB.bcd.com
  ```

Standardwerte

Diese Standardwerte sind in der Reihenfolge aufgeführt, die in der Standardkonfiguration verwendet wird:

```
Welcome Welcome.html
Welcome welcome.html
Welcome index.html
Welcome Frntpage.html
```

Bemerkungen

Die vorliegenden Informationen wurden für Produkte und Services entwickelt, die auf dem deutschen Markt angeboten werden.

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

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

IBM Europe
Director of Licensing
92066 Paris la Defense Cedex
France

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

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

Werden an IBM Informationen eingesandt, können diese beliebig verwendet werden, ohne dass eine Verpflichtung gegenüber dem Einsender entsteht.

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

IBM Corporation
ATTN: Software Licensing
11 Stanwix Street
Pittsburgh, PA 15222-9183
U.S.A.

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

Die Lieferung des im Handbuch aufgeführten Lizenzprogramms sowie des zugehörigen Lizenzmaterials erfolgt im Rahmen der Internationalen Nutzungsbedingungen der IBM für Programmpakete oder einer äquivalenten Vereinbarung.

Alle in diesem Dokument enthaltenen Leistungsdaten stammen aus einer gesteuerten Umgebung. Die Ergebnisse, die in anderen Betriebsumgebungen erzielt werden, können daher erheblich von den hier erzielten Ergebnissen abweichen. Einige Daten stammen möglicherweise von Systemen, deren Entwicklung noch nicht abgeschlossen ist. Eine Garantie, dass diese Daten auch in allgemein verfügbaren Systemen erzielt werden, kann nicht gegeben werden. Darüber hinaus wurden einige Daten unter Umständen durch Extrapolation berechnet. Die tatsächlichen Ergebnisse können abweichen. Benutzer dieses Dokuments sollten die entsprechenden Daten in ihrer spezifischen Umgebung prüfen. Diese Daten stellen deshalb keine Leistungsgarantie dar.

Informationen über Produkte anderer Hersteller als IBM wurden von den Herstellern dieser Produkte zur Verfügung gestellt, bzw. aus von ihnen veröffentlichten Ankündigungen oder anderen öffentlich zugänglichen Quellen entnommen. IBM hat diese Produkte nicht getestet und übernimmt im Hinblick auf Produkte anderer Hersteller keine Verantwortung für einwandfreie Funktion, Kompatibilität oder andere Ansprüche. Bei Fragen zur Leistungsfähigkeit von Produkten anderer Hersteller als IBM wenden Sie sich an die Hersteller dieser Produkte.

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

Diese Veröffentlichung enthält Beispiele für Daten und Berichte des alltäglichen Geschäftsablaufes. Diese Beispiele enthalten Namen von Personen, Firmen, Marken oder Produkten. Alle diese Namen sind frei erfunden; Ähnlichkeiten mit tatsächlichen Namen und Adressen sind rein zufällig.

Marken

Die folgenden Namen sind in gewissen Ländern Marken der IBM Corporation:

- AIX
- IBM
- Netfinity
- RS/6000
- SecureWay
- Tivoli
- ViaVoice
- WebSphere

Java und alle Java-basierten Marken sind in gewissen Ländern Marken von Sun Microsystems, Inc.

Microsoft, Windows, Windows NT und das Windows-Logo sind in gewissen Ländern Marken der Microsoft Corporation.

Intel, Intel Inside (Logos), MMX und Pentium sind in gewissen Ländern Marken der Intel Corporation.

UNIX ist in gewissen Ländern eine eingetragene Marke von The Open Group.

Linux ist in gewissen Ländern eine Marke von Linus Torvalds.

Andere Namen von Unternehmen, Produkten und Dienstleistungen können Marken anderer Unternehmen sein.

GC12-3421-01