IBM Web Traffic Express for Multiplatforms User's Guide

Version 1.0

IBM Web Traffic Express for Multiplatforms User's Guide

Version 1.0

Contents

Welcome

The Internet is rapidly reaching almost every business and home in the world. Millions of people have the ability to create their own Web site, and put anything they want to on it. As more and more of these custom sites are published, the traffic on the Internet increases. Server response time decreases and clients end up waiting longer for information than reading it. Organizations using firewall servers or proxy servers have the opportunity to provide safe and efficient access to the Internet.

The IBM Web Traffic Express is a caching proxy server that allows administrators to filter content using Platform for Internet Content Selection (PICS) filters, and to manipulate caching details for faster Web document retrieval.

This book is a guide through the installation, configuration, and tuning of the proxy portion of Web Traffic Express. The most current version of this document, as well as the Webmaster's Guides are available at

http://www.ics.raleigh.ibm.com/WebTrafficExpress/

Web Traffic Express is supported on many competitive platforms including AIX, Windows NT, and OS/2 Warp.

Benefits of IBM Web Traffic Express

IBM Web Traffic Express is the next generation of IBM's Internet Connection Servers, ICS. It acts as a PICS content filtering server, a caching server, and a proxy gateway while providing the reliability you've come to expect from IBM.

A key feature of Web Traffic Express is content filtering based on Platform for Internet Content Selection (PICS) labels. Language, nudity, and violence are examples of criteria you can use for filtering Web site content. For example, administrators can set the grades of sensitive material presented at the proxy server level.

Other features allow administrators to increase client anonymity by configuring the server to strip or modify HTTP headers, ″FROM:″ fields, and client IP addresses automatically.

Using Web Traffic Express helps you get your Internet information quicker and increases network bandwidth by enhancing ICS's proxy function. We've rewritten the caching algorithm to accommodate Web objects with differing arrival characteristics. Also included in Web Traffic Express is a Cache Agent that eases setup and maintenance of small to very large caches (over 10 GB). Web Traffic Express's caching is different from others in that it does not have to wait for a page to be requested before it is cached. Caching can be automatic, specified by an administrator, or both. A Proxy Activity Monitor gives summary information on the activity of the cache.

The flexible SOCKSification feature with Web Traffic Express allows you to control routes for requested URLs.

All these features provide scalable solutions for a small business, an expanding Intranet, or even a large Internet Service Provider (ISP). ISPs will be interested in Web Traffic Express because of the proxy functions it provides. For an ISP, bandwidth is money. Therefore, any opportunity to save bandwidth should be explored. The advanced caching of Web Traffic Express allows you to customize which sites are cached, and specify how large to make the cache, so you can spend less network traffic repeatedly retrieving the same pages. The server administrator can specify when the information in that URL will expire, and when it is time to update it.

Also important in this time of Internet decency, Web Traffic Express's PICS filtering gives many options to control the content of the information you are providing. Using PICS labels, you can provide an appropriate level of content.

Related information

IBM Web Traffic Express Webmaster's Guide
: explains basic settings for your server. Available at the Web Traffic Express Web site.

http://www.ics.raleigh.ibm.com/WebTrafficExpress
: Web Traffic Express's Web site including documentation, and online forums for questions

Chapter 1. Installing Web Traffic Express

This chapter will help you install your Web Traffic Express proxy server.

Before you begin

Before you install Web Traffic Express, there are some issues to consider.

- Decide whether to install the Web Traffic Express on the firewall, behind the firewall, or on an exposed or unprotected machine.
  - Normally, you would install the proxy server behind the firewall, where you can take advantage of the flexible SOCKsification feature of Web Traffic Express. Flexible SOCKS allows you to relieve the load on the firewall server, which protects the internal network and the proxy from outsiders.
  - If there are limitations such as hardware, you may choose to install on the same machine as the firewall. Your proxy will still be protected from outsiders, but all requests will be sent to the firewall machine. The load on the firewall/proxy is increased, decreasing performance.
  - If you want your users to be able to access your proxy from anywhere on the Web, or you realize that there is no critical information on your proxy to protect, you can get rid of the firewall software altogether. However, installing Web Traffic Express on an exposed machine creates the problem of user authentication. You must protect your proxy from being used by more people than intended.
- Make sure to test your network connections to and from the proxy machines by using the **ping** command.
  1. At the client's command prompt, enter

     ping *hostname*

     where *hostname* is the machine hostname where you are installing Web Traffic Express.
  2. At the command prompt of the machine to install the proxy, enter

     ping *client*

     where *client* is the machine hostname of your client.
- If you are using multiple servers, decide which cache will be updated by the cache agent.

After installing Web Traffic Express, see "Chapter 2. Configuration quick start" on page 23 to start using your proxy.

On AIX

System requirements

Hardware

- A RISC System/6000 or IBM Power Series Family with AIX Version 4.2 or later
- A mouse, trackball, TrackPoint, or Pen. Although all functions can be performed with the keyboard, a pointing device is recommended.
- Any communication hardware adapter that uses the TCP/IP protocol stack to make network connections.
- A minimum of 64 MB of RAM if you are not using caching.

  If you are using caching, an additional 2 MB is recommended per 100 MB of cache.
- A minimum of 128 MB of virtual paging space. To increase the amount of paging space, see “Chapter 9. Answers to common questions” on page 65.
- Up to 24 MB of free disk space:
  - Approximately 10 MB to install the server, which includes the base files, the message catalog, and the security filesets
  - Approximately 14 MB to install the included Java Developer’s Kit. (Necessary only if you want to develop Java servlets.)
- Additional disk space for your cache.

Software

- AIX Version 4.2.1 or later
  - **Attention:** If you are running AIX version 4.3, you must request PTF number U452478.

    If you are running AIX version 4.2.1 or earlier, install APAR number IX73200.

Before you begin installing on AIX

- Ensure that you have root authority to install the server software.
- Choose which method you will use to install the software on AIX.
- Choose the type of installation.

Choose the AIX installation method

- System Management Interface Tool (SMIT), a graphical interface that uses a mouse or other pointing device to select options.
- SMITTY, a variation of SMIT. Instead of using a pointing device to select options, you use arrow and PF keys.
- Visual System Management (VSM) graphical interface. With VSM, you can install by directly manipulating objects (icons).

This book describes how to install your server using SMIT. For more information on using SMITTY or the VSM graphical interface, refer to your *AIX Installation Guide*. The *Installation Guide* is also part of the InfoExplorer hypertext library.

Choose the type of installation to perform

Choose which type of installation you want to perform:

- First time installation

  If this is the first time you have installed the server on your machine, follow the steps under “Installing for the first time” on page 14.

- Reinstallation of the server

  If you want to reinstall the server, follow the instructions under“Reinstalling” on page 4 .

The SMIT install method described is smit install_selectable_all. With AIX V4.1.3 or later, you can use this method to install the base and secure filesets.

Installing for the first time using SMIT

To install your server using SMIT:

1. **Prepare to install**.
   - Logon as root.
   - Ensure that the hostname command returns your system’s correct hostname.
   - Insert the CD-ROM that contains the server software into the appropriate drive.
2. **Start the System Management Interface Tool**.

   At the prompt, enter **smit install_selectable_all**.
3. **Select the input device**.

   From the Install/Update From All Available Software window:
   - Click **List** for the INPUT device / directory for software option. The Single Select List window appears with a list of available input devices.
   - From the Single Select List window, click **/dev/cd0 (Multimedia CD-ROM Drive)**.

     The INPUT device / directory for software option in the Install/Update From All Available Software window is updated to show you selected the CD-ROM drive.
   - Click **OK**.

     An updated Install/Update From All Available Software window appears with additional installation options.
4. **Select installation filesets and packages**.
   - From the Install/Update From All Available Software window, click **List** for the SOFTWARE to Install option.

     The Multi-select List window appears with a list of filesets and packages you can selectively install.
   - Click the items you want to install.
     - **internet_server.base_all_filesets** to install the base server package
     - **1.0.0.0 internet_server.java** to install the JDK 1.1.1 for Java servlet support.
     - **1.0.0.0 internet_server.msg.en_US** packageto install message catalogs
   - Click **OK**. Your selections will appear as the SOFTWARE to install.

5. **Complete the installation**.

   From the Install/Update From All Available Software window:

   - Review the other installation options. You may want to consider some of the other options, such as whether to preview the install before accepting it.
     - To get an explanation of an option, click **?** and move your mouse pointer over the option of interest.
     - To change an option, enter the new choice or use the buttons to the right of the option.
   - After making all your selections, click **OK**. You'll get a message asking if you are sure. Click **OK**.
   - Verify the server is successfully installed.

     When the server is successfully installed, you will get an OK message in the top right corner of the Install/Update From All Available Software window.
   - After installing your server, look at the server README file for any late changes. The server README files are:
     - **/usr/lpp/internet_server.base/README**
6. **Define an administration user name and password**

   To use the Administration and Configuration forms from a Web browser you must define an administration user name and password. Use the **htadm** command as follows:

   htadm -adduser *password_file username password*

   For *password_file*, specify **/usr/lpp/internet/server_root/protect/webadmin.passwd**. The *username* and *password* that you specify will be added to that file.
7. **Connect to your server**.

   You can use an http browser to connect to your server's Front Page by going to the URL **http://***your.server.name*, where *your.server.name* is the fully qualified name of your host.

   The Front Page contains links that let you:

   - Access the Configuration and Administration forms
   - Create sample home pages
   - Access the Web Traffic Express Web site
   - Read online Web Traffic Express documentation

   See later chapters of this book and the *IBM Web Traffic Express Webmaster's Guide* to learn how to configure your server to your exact specifications.

Reinstalling

To reinstall your server using SMIT:

1. **Prepare to reinstall**.
   - Logon as root.

- Ensure that the hostname command returns your system's correct hostname.
- Insert the CD-ROM that contains the server software into the appropriate drive.

2. **Start the System Management Interface Tool**.

   At the prompt, enter **smit install_selectable_all**.

3. **Select the input device**.

   From the Install/Update From All Available Software window:

   - Click **List** for the INPUT device / directory for software option. The Single Select List window appears with a list of available input devices.
   - From the Single Select List window, click **/dev/cd0 (Multimedia CD-ROM Drive)**.

     The INPUT device / directory for software option in the Install/Update From All Available Software window is updated to show you selected the CD-ROM drive.
   - Click **OK**.

     An updated Install/Update From All Available Software window appears with additional installation options.

4. **Select installation filesets and packages**.

   - From the Install/Update From All Available Software window, click **List** for the SOFTWARE to Install option.

     The Multi-select List window appears with a list of filesets and packages you can selectively install.
   - Click the items you want to install.
     - **internet_server.base_all_filesets** to install the base server package
     - **1.0.0.0 internet_server.java** to install the JDK 1.1.1 for Java servlet support.
     - **1.0.0.0 internet_server.msg.en_US** packageto install message catalogs
     - **internet_server.proxy** to install the proxy package
   - Click **OK**. Your selections will appear as the SOFTWARE to install.

5. **Complete the installation**.

   From the Install/Update From All Available Software window:

   - Review the other installation options. You may want to consider some of the other options, such as whether to preview the install before accepting it.
     - To get an explanation of an option, click **?** and move your mouse pointer over the option of interest.
     - To change an option, enter the new choice or use the buttons to the right of the option.
   - To reinstall, you also need to specify the following options:
     - **Yes** for SAVE replaced files?

       Also, you may want to specify a directory for ALTERNATE save directory.
     - **No** for AUTOMATICALLY install requisite software?
     - **Yes** for OVERWRITE same or newer versions?

     Notes:

     a. Always back up your configuration file before overwriting.

b. Do not specify **Yes** for both AUTOMATICALLY install requisite software and OVERWRITE same or newer versions.

- After making all your selections, click **OK**. You'll get a message asking if you are sure. Click **OK**.
- Verify the server is successfully installed.

  When the server is successfully installed, you will get an OK message in the top right corner of the Install/Update From All Available Software window.
- After reinstalling your server, look at the server README file for any late changes. The server README files are:
  - **/usr/lpp/internet_server.base/README**

6. **Define an administration user name and password**

   To use the Administration and Configuration forms from a Web browser you must define an administration user name and password. Use the **htadm** command as follows:

   htadm -adduser *password_file username password*

   For *password_file*, specify **/usr/lpp/internet/server_root/protect/webadmin.passwd**. The *username* and *password* that you specify will be added to that file.

7. **Connect to your server**.

   Use an http browser to connect to your server's Front Page by going to the URL **http://***your.server.name*, where *your.server.name* is the fully qualified name of your host.

   The Front Page contains links that let you:

   - Access the Configuration and Administration forms
   - Access the Web Traffic Express Web site
   - Read online Web Traffic Express documentation

   See later chapters in this book and the *IBM Web Traffic Express Webmaster's Guide* to learn how to configure your server to your exact specifications.

Removing filesets

To remove filesets using SMIT follow the directions below.

1. **Prepare to remove filesets**.
   - Logon as root.
   - Ensure that the hostname command returns your system's correct hostname.
2. **Start the System Management Interface Tool**.

   At the prompt, enter **smit install_remove**.
3. **Select the fileset(s) to be removed**.
   - From the Remove Software Products window, click **List** for *SOFTWARE name. The Multi-select List window appears with a list of filesets installed on your system.

- From the Multi-select List window, click the name of the fileset(s) you want to remove.
- Click **OK**. The Remove Software Products window appears.

4. **Complete the removal of the fileset(s)**.
   - To remove the fileset(s) you've selected, you must specify **No** for PREVIEW only. There are several other options you may also want to consider.
   - Click **OK**.

     When the fileset is removed, you will get an OK message in the top right corner of the Remove Software Products window.

On Windows NT

System requirements

Hardware

- Any personal computer that can support Windows NT** 3.51 or 4.0
- A mouse or compatible pointing device
- Any communication hardware adapter supported by Windows NT
- 32 MB of memory (RAM) and a 32 MB swap file on a hard disk, if caching will not be used.

  If caching is used, the amount of memory depends on the size of your cache. An additional 2 MB is recommended per 100 MB of cache
- Approximately 18 MB of free disk space for installation:
  - Approximately 5.5 MB to install the server
  - Approximately 0.1 MB to install the NT service files
  - Approximately 12 MB for the Java Developer's Kit (JDK)
- Additional disk space is required for your cache
- A CD-ROM drive, if you are installing your server from a CD

Software

- Microsoft** Windows NT Client (Workstation) 3.51 or 4.0, with TCP/IP configured
- A partition formatted using the NT File System (NTFS) or the High Performance File System (HPFS). We recommend that you use NTFS for best performance.
- For SNMP support, install the SystemView Agent Developers toolkit which you can be download from URL `http://www.software.ibm.com/download/`.

Installing for the first time or reinstalling

1. **Prepare to install or reinstall**.
   - If you are reinstalling Web Traffic Express, it's recommended that you uninstall the version currently on your system. To uninstall:
     a. Stop the server and click the Uninstall icon in the program folder for Web Traffic Express.

b. When prompted to select which components to uninstall, select all the components and click OK.

c. When progress screens are displayed, click OK to continue.

- To install or reinstall Web Traffic Express, put the server CD-ROM in your CD-ROM drive. Usually the installation program starts automatically. If you need to start the installation program manually, follow the instructions below. Otherwise skip to Step 2.

  You can start the installation program manually just like any other Windows program:

  a. On Windows NT 4.0, click **Start** on the taskbar followed by **Run**. On Windows NT 3.51, click **File** on the Program Manager window, followed by **Run**.

  b. On the Run dialog, click **Browse** to find the appropriate installation program on the CD-ROM.

     Look in the \NT directory for the program named setup.exe.

  c. When you have found the setup.exe file, select it and click **Open** or **OK** to close the Browse dialog. Then click **OK** to start the installation. The installation program is named setup.exe for Windows NT.

2. **Deselect the components you do not want to install**.

   - From the Select Components window, deselect any components that you do not want to install:
     – IBM Web Traffic Express

       This is the base Web proxy server product.

       Note: The product documentation is automatically installed when you install this component.

     – NT Service

       For Windows NT, the NT Service component allows you to run the server as an NT service.

       Once your server is installed as an NT service, the Web Traffic Express program group will contain only the Uninstall icon because you can start a true Windows NT service only from the Services panel. You can reinstall the base program to get the program icon.

   - Click **Next**.

3. **Choose a target directory**.

   Note: The directory you specify for the installation must be on a drive in an NTFS, HPFS, or FAT partition. We recommend that you install onto an NTFS drive for better file protection.

   From the Choose Target Directory window:

   - Click **Browse** if you want to change the drive or directory destination. The default destination directory for the server is C:\WWW\.
   - Click **Next**.

4. **Choose component directories**.

- From the Choose Component Directories window, review the destination directories for the server components. These directories define where you want to install your server components and where you want to store the resources you will be making available through your server.

  Note: You can change the path for a directory using either of the following methods:
  - Type over the default path.
  - Click the button next to the directory name and select the new path in the **Choose directory** window. Click **OK** to continue.

  The Choose Component Directories window initially displays the first five directories. Click **Next** to display the remaining directories in a separate window.

  - **Administration directory**

    The files used by the server Configuration and Administration forms are installed in this directory.
  - **Executables directory**

    The server executable program files and other related files are installed in this directory.
  - **CGI Bin scripts directory**

    Your script programs that use the Common Gateway Interface (CGI) and the server htimage program are installed in this directory.
  - **Documentation directory**

    The product documentation is installed in this directory.
  - **HTML directory**

    Your HTML documents, the server sample HTML pages, and the server Front Page are installed in this directory.
  - **Icons and graphics directory**

    The default directory list icons are installed in this directory. You may also choose to put your own icon and graphics files in this directory.
  - **Labels directory**

    Example labels for Platform for Internet Content Selection (PICS) support are installed in this directory.
  - **Logs directory**

    The server puts log files in this directory.
  - **Servlets directory**

    This is the directory for all Java servlets.
- Click **Next**.

  If you selected any of the component directories, you will be prompted on additional windows for those directories.

5. **Select configuration file**

   If you are reinstalling and already have a server configuration file, you will be asked if you want to keep the existing configuration files. Like the admin.pwd file, the

httpd.cnf file is located in the Windows directory (\WINNT or \WINNT35). Web Traffic Express installs a javelin.cnf file in the same directory. This file includes many directives specific to the proxy portion of the server.

- Click **No** if you want to install with the default configuration settings.

  Note: Your current configuration files will be copied to a file named httpd.$nf and javelin.$nf.

  It is recommended that you choose **No**, which is the default, because many new directives have been added to the configuration file.
- Click **Yes** if you are reinstalling and want to save your current configuration settings.

6. **Select PICS configuration file**.
   - If an existing PICS configuration file is found on your system, you will be asked if you want to keep the existing file.
   - Click **Yes** or **No** to continue.

     If you select **No**, the existing file will be backed up with the file name ICS_PICS.$NF.
7. **Select administration password file**

   If you are reinstalling and already have an administration password file, you will be asked if you want to keep the existing admin.pwd file. The file is usually located in the directory identified in the ETC environment variable (some TCP/IP implementations set the ETC variable). If the ETC environment variable is not defined, the file is located in the Windows system directory (\WINNT or \WINNT35).

   Click **No** if you do not want to use the existing file. In this case, you must specify an administrator ID and password when prompted in Step 8.

   Note: Your current admin.pwd file will be copied to a file named admin.$wd.

   Click **Yes** if you want to keep your existing administrator IDs and passwords. In this case, do not enter an Administrator ID and password in Step 8. Leave the fields blank.
8. **Choose configuration parameters**.
   - From the Configuration Parameters window, review the default values for host name, HTTP port.
     - **Host Name**

       The default value is the host name defined as part of your TCP/IP configuration. If you want to use an alias, you can change this field to a fully qualified host name that is defined in your domain name server.
     - **HTTP Port (80)**

       The default value of 80 is the well known port number for Hypertext Transfer Protocol (HTTP). Other port numbers less than 1024 are reserved for other TCP/IP applications. Port numbers 8080 and 8008 are commonly used for testing servers.

- **Administrator ID**

  This is the ID of your server administrator. Anyone attempting to use the server Configuration and Administration Forms will be prompted to enter this ID. There is no default value. Unless you elected to use an existing admin.pwd file, you must specify an administrator ID. If you are using an existing admin.pwd file, you do not need to specify an administrator ID.

- **Administrator Password**

  This is the password that protects access to the Configuration and Administration forms. Anyone attempting to use the server Configuration and Administration forms will be prompted to enter this password. There is no default value. Unless you elected to use an existing admin.pwd file, you must specify an administrator password. If you are using an existing admin.pwd file, you do not need to specify an administrator password.

- After reviewing the default values, make your changes by editing the parameter fields.
- Click **Next**.

9. **Enable Java support**
   - You will be asked if you want to enable Java servlet support.
   - Click **Yes** or **No** to begin the installation.
10. **Complete the installation**.
    - During the installation, the **Setup** window is displayed and shows you the status of the installation. If you changed any of the default values, you may be prompted on additional windows for configuration values. You can stop the installation by clicking **Cancel**.
    - The server files will be installed and the server icons added.

      If you installed your server as an NT service, you will get a message stating that you can start the program from the Services panel.
    - After installation is completed, a message will be issued if you need to reboot your system.
    - Click **OK**.

Uninstalling

You can uninstall the Web Traffic Express by either clicking the Uninstall icon or issuing the command to uninstall.

1. Click the Uninstall icon or type **ibmwsuin.exe uninst.exe** from a command prompt.
2. Select the components you want to uninstall. From the Select components window, select any component that you want to uninstall:
   - Service component
   - Base component
   - Proxy component
3. Click **Next**.

The uninstall program automatically loads the log files for each component you selected and uninstalls the components. The following log files are in the directory *server_root*\uninst\:

- DeIsL1.isu: Registry component log file
- DeIsL2.isu: Service component log file
- DeIsL4.isu: Base component log file

A status window will be displayed for each component showing the status of the uninstall. Click **OK** to continue.

On OS/2 Warp

System requirements

Hardware

- Any personal computer or IBM Personal System/55 that can support OS/2 Warp V3.0 or later
- A mouse or compatible pointing device
- Any communication hardware adapter that is supported by the TCP/IP protocol stack. See “Q. How can I do a stand-alone test of my server?” on page 67.
- 32 MB of memory in RAM and a 32 MB swap file on a hard disk, if caching will not be used.

  If caching is used, the amount of RAM depends on the size of your cache. An additional 2 MB is recommended per 100 MB of cache
- Up to 20 MB of free disk space:
  - Approximately 8 MB to install the server, which includes the base files and the server messages.
  - Approximately 12 MB to install the Java Developer’s Kit (JDK)
- A CD-ROM drive, if you are installing your server from a CD

Software

- OS/2 Warp V3.0 or later or OS/2 Warp Server

  Note: To use Java servlet support, OS/2 V4.0 or later is required.
- A partition formatted using the High Performance File System (HPFS). Only the server needs to be on the HPFS partition, not the entire operating system.
- One of the following:
  - TCP/IP V3.0, which is a part of Warp Connect and OS/2 Warp Server
  - Internet Connection for OS/2, which is included in all versions of Warp including Warp Connect.
- For using Simple Network Management Protocol (SNMP), the following applies:
  - For OS/2 Warp V3.0, SNMP support was shipped with the product.

– For OS/2 V4.0, install one of the following:
  - The SystemView Agent Developer's toolkit can be downloaded from URL `http://www.software.ibm.com/download/`.
  - The TME 10 NetFinity Server can be downloaded from URL `http://www.software.ibm.com/download/`.

- Corrective service

  You can determine the service level for TCP/IP, MPTS, and other OS/2 components by entering the **syslevel** command at an OS/2 command prompt.

  Table 1 summarizes the corrective service available for TCP/IP and MPTS.

  Notes:

  1. You must apply corrective service updates for the server to work correctly. Always apply updates in this order:
     a. CSD
     b. RSU
     c. APAR

Table 1. Summary of Corrective Service

OS/2 Level	TCP/IP Level	MPTS CSD	TCP/IP CSD	TCP/IP APAR
Warp	V2.0 (Not supported)			
Warp Connect	V3.0	WR08415 (1)	UN00959 (2)	IC17248 (3)
Warp Server	V3.1	WR08415 (1)	UN00959 (2)	IC17639 (4)
Warp V4 (Merlin)	V4.0	WR08415 (1)	None	None
Warp Server SMP	V3.5	WR08503 (5)	None	None

Notes:

1. To obtain CSD WR8415, go to URL:

   `ftp://wr08415:by4taxis@testcase.software.ibm.com/`

2. To obtain CSD UN00959, go to URL:

   `ftp://service.boulder.ibm.com/ps/products/tcpip/fixes/v3.1os2/un00959/`

3. To obtain APAR IC17248, go to URL:

   `ftp://service.boulder.ibm.com/ps/products/tcpip/fixes/v3.1os2/ic17639/`

4. To obtain APAR IC17639, go to URL:

   `ftp://service.boulder.ibm.com/ps/products/tcpip/fixes/v3.1os2/ic17639/`

5. To obtain CSD WR08503, go to URL:

   `ftp://wr08503:laws5yet@testcase.software.ibm.com/`

You must apply RSU updates for all levels. For RSU updates, go to URL:

http://service.software.ibm.com/pbin-usa-ps/getobj.pl?/pdocs-usa/tcpip_stack_rsu.html/

Before you begin

Choose the type of installation to perform

First time installation

If this is the first time you have installed the server on your machine, follow the steps under “Installing for the first time”.

Reinstallation of the server

If you want to reinstall the server, follow the instructions under “Reinstalling the server” on page 18.

Installing for the first time

To install your server:

1. **Prepare to install**.
   - Put the server CD-ROM in your CD-ROM drive, and enter the install command.
     - Change to the OS2 subdirectory on the CD-ROM.
     - enter the following command at an OS/2 prompt:

       d:\OS2**install**

       For *d:*, enter the drive where you put the server CD-ROM.
   - From the Instructions for Installation window, click **Continue**.
   - From the Install window, click **OK**.
2. **Select the installation packages**.

   From the Install - directories window, select the product components that you want to install.
   - **Web Traffic Express** to install the base server files.
   - **Installation and maintenance** to install the server installation utility.

   Note: The server documentation is automatically installed.
3. **Change the default installation directories**

   Optionally, use the fields in the bottom half of the Install - directories window to change the default installation directories. These directories define where you want to install the server components and where you want to store the resources you will be making available through the server.

   You can change these paths by clicking **Disk space** and selecting the drive where you want the directories installed.

   Attention: You must use the scroll bar to see the complete list of directories. Each directory you specify must be on a drive in a High Performance File System (HPFS) partition.
   - **Administration directory**

The files used by the server Configuration and Administration forms are installed in this directory.

- **Executables directory**

  The server executable program files and other related files are installed in this directory.
- **CGI Bin scripts directory**

  Your script programs that use the Common Gateway Interface (CGI) and the server htimage program are installed in this directory.
- **Documentation directory**

  The product documentation is installed in this directory.
- **HTML directory**

  Your HTML documents, the server sample HTML pages, and the server Front Page are installed in this directory.
- **Icons and graphics directory**

  The default directory list icons are installed in this directory. You may also choose to put your own icon and graphics files in this directory.
- **Labels directory**

  Example labels for Platform for Internet Content Selection (PICS) support are installed in this directory.
- **Logs directory**

  The server puts log files in this directory.
- **Servlets directory**

  This is the directory for all Java servlets.
- Click **Next**.

  If you selected any of the component directories, you will be prompted on additional windows for those directories.

4. **Start the installation**.

   Click **Install**.
5. **Select administration password file**

   If the installation process finds that you already have an administration password file named admin.pwd, you will be asked if you want to keep the existing password file.

   - Click **No** if you want to create a new password file.

     Note: Your current admin.pwd file will be copied to a file named admin.$wd.

     If you choose **No**, you must specify an Administrator ID and password on the server Installation Configuration window (which is the next installation step).
   - Click **Yes** if you want to use your current admin.pwd file. In this case, you do not need to specify an Administrator ID and password in the next step. Leave the fields blank.
6. **Choose configuration values**.

From the server Install Configuration window, optionally change the default values for host name, HTTP port. You must specify a value for Administrator ID and Password unless you are using your current password file.

- **Host Name**

  The default value is the host name defined in your CONFIG.SYS file. If you want to use an alias, you can change this field to a fully qualified host name that is defined in your domain name server.

- **HTTP Port**

  The default value of 80 is the well known port number for Hypertext Transfer Protocol (HTTP). Other port numbers less than 1024 are reserved for other TCP/IP applications. Port numbers 8080 and 8008 are commonly used for testing servers.

- **Administrator ID**

  This is the ID of your server administrator. Anyone attempting to use the server Configuration and Administration forms will be prompted to enter this ID. There is no default value. Unless you are using an admin.pwd file from a previous installation, you must specify an Administrator ID.

- **Password**

  This is the password you use to protect access to the Configuration and Administration forms. Anyone attempting to use the Configuration and Administration formswill be prompted to enter this password. There is no default value. Unless you are using an admin.pwd file from a previous installation, you must specify a password.

7. **Select automatic server startup**.

   Use the **Auto Start Server at Bootup** check box to indicate if you want the server to start automatically when you start your host machine. If you check the box, the server will be added to your OS/2 Startup folder.

8. **Note target directory assignments**.

   Target directories cannot be changed. You may want to make a note of the information in this window because it shows where the server will look for certain files when it is running.

   - **Configuration files**

     This is the file that contains the server configuration settings. The files are named httpd.cnf and javelin.cnf, and and is put in the path specified on the SET ETC statement in your CONFIG.SYS file.

   - **CGI scripts**

     This is the same directory you specified for your CGI script programs on the Install - directories window.

   - **HTML documents**

     This is the same directory you specified for your HTML documents on the Install - directories window.

   - **Configuration password file**

This is the server password file that will contain the values you entered in the **Administrator ID** and **Password** fields. The file is named admin.pwd and is put in the path specified on the SET ETC statement in your CONFIG.SYS file.

Click **OK** to continue.

9. **Select configuration file settings**

   If the installation process finds that you already have a server configuration file named httpd.cnf, and javelin.cnf you will be asked if you want to keep the existing configuration files.

   - Click **No** if you want to install with the default configuration settings.

     Note: Your current httpd.cnf configuration file will be copied to a file named httpd.$nf, and javelin.cnf.

   - Click **Yes** if you are reinstalling and want to save your current configuration settings.

   Click **OK** to continue.

10. **Select PICS configuration file**.

    - If an existing PICS configuration file is found on your system, you will be asked if you want to keep the existing file.
    - Click **Yes** or **No** to continue.

      If you select **No**, the existing file will be backed up in your ETC directory with the file name ICS_PICS.$NF.

11. **Enable Java support**.

    - You will be asked if you want to enable Java servlet support.
    - Click **Yes** or **No** to continue.

12. **Complete the installation**.

    The installation program transfers files to your computer. When installation is complete, an informational message is displayed. Click **OK**.

    If the server Installation window is still open, click **Exit**.

    If the installation procedure updated your CONFIG.SYS file, you will be prompted to reboot your system before starting the server.

    If you checked the **Auto Start Server at Bootup** check box in the server Install Configuration window, the server will be placed in your OS/2 Startup folder and will start each time you start or reboot your system.

13. **Connect to your server**.

    Use your favorite browser to connect to your server's Front Page by going to the URL **http://***your.server.name*, where *your.server.name* is the fully qualified name of your host. The Front Page contains additional links that let you:

    - Access the Configuration and Administration forms
    - Access the Web Traffic Express Web site
    - Read online Web Traffic Express documentation

    See later chapters and the *Webmaster's Guide* to learn how to configure your server to your exact specifications.

Reinstalling the server

Notes about reinstalling:

1. All of your current configuration settings are saved except for the protection setup that protects the Configuration and Administration forms. In the configuration file, this protection setup has a name of PROT-ADMIN. Your current PROT-ADMIN protection is overwritten with the default administration protection setup. If you changed this protection setup in your previous configuration file, it goes back to the default protection setup. You can either use the default or change it again after reinstalling the server.
2. By default, the admin.pwd password file on the path specified on the SET ETC statement in your CONFIG.SYS file controls access to the Configuration and Administration forms. During reinstallation, you will be asked whether to use the admin.pwd file. If you choose not to use the admin.pwd file, you must specify an Administrator ID and password when prompted.

To reinstall the server:

1. **Prepare to install**.
   - Put the server CD-ROM in your CD-ROM drive, and enter the install command:
     – Enter the following command at an OS/2 prompt:

       ```
       d:\OS2\install
       ```

       where *d:* is your CD-ROM drive
   - From the Instructions for Installation window, click **Continue**.
2. **Delete server components**

   From the Installation Options window:
   - Click **Delete the installed product and re-install** to delete and reinstall the current version of your server.
   - Click **Continue**.
   - From the Delete window, select the components you want to delete, either by clicking them individually or by clicking **Select all**.
   - Click **Delete**.

     Only the components you select are deleted. This procedure does not delete other related files such as your configuration file, log files, Access Control List files, or protection setup files.
   - From the Installation and Maintenance window, click **OK**.
   - From the Installation window, click **OK**.
   - Reboot your system.
3. **Reinstall the server**
   - Put the server CD-ROM in your CD-ROM drive, and enter the install command.
     – enter the following command at an OS/2 prompt:

       ```
       d:\OS2\install
       ```

       where *d:* is your CD-ROM drive

- From the Instructions for Installation window, click **Continue**.
- From the Install window, click **OK**.

4. **Select the installation packages**.

   From the Install - directories window, select the product components that you want to install.

   Note: The documents are automatically installed.

   - **Web Traffic Express** to install the base proxy server files.
   - **Installation and maintenance** to install the server installation utility.

5. **Change the default installation directories**.

   Optionally, use the fields in the bottom half of the Install - directories window to change the default installation directories. These directories define where you want to install the server components and where you want to store the resources you will be making available through the server.

   You can change these paths by clicking **Disk space** and selecting the drive where you want the directories installed.

   Attention: You must use the scroll bar to see the complete list of directories. Each directory you specify must be on a drive in a High Performance File System (HPFS) partition.

   - **Administration directory**

     The files used by the server Configuration and Administration forms are installed in this directory.
   - **Executables directory**

     The server executable program files and other related files are installed in this directory.
   - **CGI Bin scripts directory**

     Your script programs that use the Common Gateway Interface (CGI) and the server htimage program are installed in this directory.
   - **Documentation directory**

     The product documentation is installed in this directory.
   - **HTML directory**

     Your HTML documents, the server sample HTML pages, and the server Front Page are installed in this directory.
   - **Icons and graphics directory**

     The default directory list icons are installed in this directory. You may also choose to put your own icon and graphics files in this directory.
   - **Labels directory**

     Example labels for Platform for Internet Content Selection (PICS) support are installed in this directory.
   - **Logs directory**

     The server puts log files in this directory.
   - **Servlets directory**

This is the directory for all Java servlets.

- Click **Next**.

  If you selected any of the component directories, you will be prompted on additional windows for those directories.

6. **Start the installation**.

   Click **Install**.

7. **Select administration password file**

   If the installation process finds that you already have an administration password file named admin.pwd, you will be asked if you want to keep the existing password file.

   - Click **No** if you want to create a new password file.

     Note: Your current admin.pwd file will be copied to a file named admin.$wd.

     If you choose **No**, you must specify an Administrator ID and password on the server Installation Configuration window (which is the next installation step).
   - Click **Yes** if you want to use your current admin.pwd file. In this case, you do not need to specify an Administrator ID and password in the next step. Leave the fields blank.

8. **Choose configuration values**.

   From the server Install Configuration window, optionally change the default values for host name, HTTP port. You must specify a value for Administrator ID and Password.

   - **Host Name**

     The default value is the host name defined in your CONFIG.SYS file. If you want to use an alias, you can change this field to a fully qualified host name that is defined in your domain name server.
   - **HTTP Port**

     The default value of 80 is the well known port number for Hypertext Transfer Protocol (HTTP). Other port numbers less than 1024 are reserved for other TCP/IP applications. Port numbers 8080 and 8008 are commonly used for testing servers.
   - **Administrator ID**

     This is the ID of your server administrator. Anyone attempting to use the server Configuration and Administration forms will be prompted to enter this ID. There is no default value. Unless you are using an admin.pwd file from a previous installation, you must specify an Administrator ID.
   - **Password**

     This is the password you use to protect access to the Configuration and Administration forms. Anyone attempting to use the Web Traffic Express Configuration and Administration forms will be prompted to enter this password. There is no default value. Unless you are using an admin.pwd file from a previous installation, you must specify a password.

9. **Select automatic server startup**.

Use the **Auto Start Server at Bootup** check box to indicate if you want the server to start automatically when you start your host machine. If you check the box, the server will be added to your OS/2 Startup folder.

10. **Note target directory assignments**.

    Target directories cannot be changed. You may want to make a note of the information in this window because it shows where the server will look for certain files when it is running.

    - **Configuration file**

      This is the file that contains the server configuration settings. The files are named httpd.cnf, javelin.cnf, and socks.cnf, and are put in the path specified on the SET ETC statement in your CONFIG.SYS file.

    - **CGI scripts**

      This is the same directory you specified for your CGI script programs on the Install - directories window.

    - **HTML documents**

      This is the same directory you specified for your HTML documents on the Install - directories window.

    - **Configuration password file**

      This is the server password file that will contain the values you entered in the **Administrator ID** and **Password** fields. The file is named admin.pwd and is put in the path specified on the SET ETC statement in your CONFIG.SYS file.

    Click **OK** to continue.

11. **Select configuration file settings**

    If the installation process finds that you already have server configuration files named httpd.cnf and javelin.cnf you will be asked if you want to keep the existing configuration file.

    - Click **No** if you want to install with the default configuration settings.

      Note: Your current configuration files will be copied to a files with an extension of .$nf.

    - Click **Yes** if you are reinstalling and want to save your current configuration settings.

    Click **OK** to continue.

12. **Select PICS configuration file**.

    - If an existing PICS configuration file is found on your system, you will be asked if you want to keep the existing file.
    - Click **Yes** or **No** to continue.

      If you select **No**, the existing file will be backed up in your ETC directory with the file name ICS_PICS.$NF.

13. **Enable Java support**.

    - You will be asked if you want to enable Java servlet support.
    - Click **Yes** or **No** to continue.

14. **Complete the installation**.

    The installation program copies files to your computer. When installation is complete, an informational message is displayed. Click OK.

    If the server Installation window is still open, click **Exit**.

    If the installation procedure updated your CONFIG.SYS file, you will be prompted to reboot your system before starting the server.

    If you checked the **Auto Start Server at Bootup** check box in the server Install Configuration window, the server will be placed in your OS/2 Startup folder and will start each time you start or reboot your system.

    When you first start the server, it uses the information you entered during installation for Host name, HTTP Port, Administrator ID, Password. It uses default values for the other configuration settings.

15. **Connect to your server**.

    Use your favorite browser to connect to your server's Front Page by going to the URL **http://***your.server.name*, where *your.server.name* is the fully qualified name of your host. The Front Page contains additional links that let you:

    - Access the Configuration and Administration forms
    - Access the Web Traffic Express Web site
    - Read online Web Traffic Express documentation

    See later chapters and the *Webmaster's Guide* to learn how to configure your server to your exact specifications.

Chapter 2. Configuration quick start

This chapter outlines the minimum steps to get the features of your proxy working. Configuration files installed with Web Traffic Express are:

On AIX in the /ETC/ directory:	httpd.conf javelin.conf socks.conf ics_pics.conf
On Windows NT in the \WINNT\ or \WIN35\ directory on the boot drive:	httpd.cnf javelin.cnf socks.cnf ics_pics.cnf
On OS/2 in the directory specified by the SET ETC= statement in the **config.sys** (usually \MPTN\ETC\ on the boot drive):	httpd.cnf javelin.cnf socks.cnf ics_pics.cnf

You can edit these files with any ASCII editor.

Refer to later chapters for more information on these features and tuning them for maximum performance.

Basic configuration

Once Web Traffic Express is installed, make sure your proxy can resolve domain names by setting `DNS-Lookup On` in the **httpd** configuration file.

These directives were added to the **httpd** configuration file to start the server with the proxy enabled. A proxy is a gateway that assumes the responsibility of retrieving URLs for multiple clients, decreasing the load on their own machines. No further action is required to act as a proxy server.

- `ProxyAccessLog` *server_root*`/logs/httpd_proxy`
- `Proxy http:*`
- `Proxy ftp:*`
- `Proxy gopher:*`

where *server_root* is the root directory of the proxy server. If you want to save, or cache URLs as they are retrieved, please continue.

Caching proxy configuration

To begin using the caching capabilities of Web Traffic Express, you must add the following directives to the **httpd** configuration file. These changes were not automatically made because of the options available for caching.

- `CacheRoot` *cache_directory*
- `Caching On`
- `CacheAccessLog` *cache_log_path*

where *cache_directory* is the root directory to store cached files, and *cache_log_path* is the directory to store the cache log.

`Caching` instructs the server to start saving files when they are loaded. The `CacheRoot` directive assigns your cache to a directory. All files in the cache will be under this directory. For the server activity monitor, the `CacheAccessLog` records all files retrieved from the cache. `CacheAccessLog` is not required if you do not want to use the Cache Log under the server activity monitor or automatic cache refresh.

After Web Traffic Express has started caching files, you must maintain the cache by removing stale files to keep the cache up to date.

Garbage collection

Garbage collection removes old files from the cache to provide space for newer files. To enable garbage collection, the following directives were added to the **httpd** configuration file. No action is necessary unless you wish to change the time of garbage collection.

- `GC On`
- `GCDailyGC 03:00`

The garbage collection time is on a 24 hour clock and by default, begins at 3:00 AM.

Note: Garbage collection takes a significant amount of CPU resources, and should be scheduled to take place when the load on the proxy server is low.

Automatic cache refresh

You can instruct your server to refresh the cache with currently loaded URLs or even URLs that have not been cached before. See “Chapter 4. Configuring and starting automatic cache refreshing” on page 33 for information. To enable the automatic cache refresh agent, you must add the following:

1. To the **httpd** configuration file:
   - `CacheAccessLog` *cache_log_path*
2. To the **javelin** configuration file:
   - `AutoCacheRefresh On`
   - `Proxy` *proxy_server*

- `MaxRunTime` *time* [minutes, seconds, hours, days]
- `LoadTopCached` *xx*

where:

cache_log_path
: is the directory to store the cache log. This directive is required for automatic cache refreshing.

proxy_server
: is the name of the proxy server whose cache is to be updated

time
: is the maximum time to allow the cache agent to run

xx
: is the number of URLs from the previous night's cache access log you wish to refresh

See "Chapter 4. Configuring and starting automatic cache refreshing" on page 33 for more information.

SOCKS configuration

If your proxy server is located behind a firewall, you can use SOCKS to allow it to retrieve sites on the Internet. Using the `SocksServer` directive in the **httpd** configuration file allows you direct **all** requests through the firewall or SOCKS server. Every time a request is made using this method of SOCKS causes an interruption in the firewall while it routes the request to the correct destination.

However, flexible SOCKification allows you to configure direct connections for specific IP addresses or hosts, and configure requests that pass through the SOCKS sever for other addresses. Now, only requests that must go to the Internet must pass through the firewall, increasing efficiency of and reducing strain on the firewall or SOCKS server.

Remove the `SocksServer` directive from the **httpd** configuration file and add `FlexibleSocks On` to the **javelin** configuration file. You must still configure the **socks** configuration.

See "Chapter 6. Configuring flexible SOCKSification" on page 53 for more information.

Secure connection through a proxy server

Secure Secure connections are supported with Web Traffic Express using SSL tunneling. Add these directives to your **httpd** configuration file:

- `Proxy *:`*443*
- `ENABLE CONNECT`

where *443* is the port number on the destination server the proxy connects to for a secure connection.

Secure connections that involve encryption and decryption are established between the client browser and destination content server. Secure requests are not cached because the proxy server makes no attempt to decrypt the client request. Instead, the proxy establishes a connection to the content server and passes the request through the destination server without looking at the data.

This pass through protocol is known as Secure Socket Layer (SSL) Tunneling. The Web client requests a secure connection through a proxy server by configuring the ″security proxy″ setting on the browser.

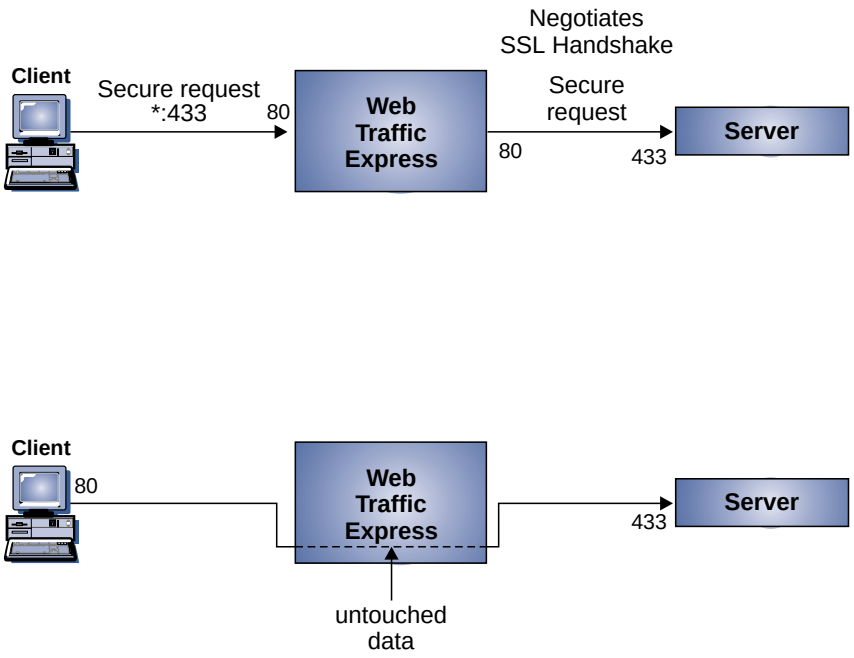

Figure 1. SSL Tunneling

1. Client makes a tunneling request using the syntax **https://***host:port*. The port number is optional and is usually 443
2. Proxy accepts connection on port 80, receives the request, and connects to the destination on the requested port
3. Proxy replies that a connection is established
4. The destination server returns a handshake request, since the request is on a secure port, to the proxy
5. Proxy forwards the server handshake to the client on port 80
6. Once the secure handshake has been completed, the proxy sends and receives encrypted data to be decrypted at the client

7. Once the client or the destination server requests a close on either port, the proxy resumes its normal activity and closes port 433 and 80

Chapter 3. Proxy and caching concepts

Overview of proxy servers

A proxy server acts as a type of gateway for Web requests and performs basic Web server duties, such as receiving requests and serving URLs. It receives a request for a URL from a client, and sends the request to the destination server. Once it receives the information, the proxy forwards the information to the client. This transfers the responsibility and the machine load associated with making the URL request to a dedicated machine.

A proxy compared to a firewall

A physical machine that performs many similar functions to a proxy server is a firewall server. The two are commonly mistaken for the same thing — probably because many computers currently function as a firewall and proxy. This is not required. With Web Traffic Express's flexible SOCKSification feature, the proxy can be on a totally separate machine from the firewall. This allows a firewall to secure the internal network while another machine searches for previously cached documents.

A caching proxy server:

- Fetches Internet content
- Speeds up retrieval time by using caching
- Allows content filtering at the server level

A firewall:

- Does not retrieve Internet information
- Filters packets
- May **contain** a proxy server
- Is designed to secure an internal network

Overview of caching

Caching proxy servers perform the same way a normal proxy server does. They both forward requests from clients and other Web servers and return responses from servers to clients. The difference is a caching proxy server can save (or cache) Web documents it retrieves from other sites, and serve subsequent requests locally. The user gets the information faster, while saving network bandwidth.

A caching proxy:

- Receives requests from Web clients
- Serves requests for documents from the cache, if possible
- Fetches documents from destination servers if they are not available in the cache

- Manages the cache of documents

The best way to redirect Web requests to a caching proxy server is to configure the client browser to do so automatically. The browser sends the request to the proxy server. The proxy first attempts to serve the request from the cache. If the document is available in the cache, it is returned to the Web client. If the document is not in the cache or is old, the proxy forwards the request to the appropriate content server and relays the response to the Web client. If the proxy determines the response is cacheable, it is stored in the cache and is then available for the next request.

A common concern when using a cache is disk space and file maintenance. Web Traffic Express allows you to limit the amount of disk space used for the entire cache; see “CacheSize - Specify cache size” on page 72. In addition, Web Traffic Express has an automatic process, known as garbage collection, which it uses to remove files from the cache. Old or unused files are removed to make room for more current files.

What documents are cached

The server administrator defines rules using configuration directives to determine which documents should be cached, how long they should be cached, and which files are never to be cached.

However, some files are never cached:

- Requests that use methods other than GET. For example, posting a form and PUT requests are not cached.
- Any documents requiring authentication or payment.
- The dynamic output of any CGI script or Java servlet, because this is unique each time it is requested.
- Any information passed on an SSL connection, because the proxy cannot decrypt the data passing through it.
- Any URL containing a ″?″.

Cache freshness

Keeping cached documents consistent with the original document located at the content server is known as maintaining cache freshness. Web Traffic Express computes an expiry time for each document it caches. The HTTP header of the document, generated by the content server, contains the expiration information. Expiration can be specified by a content server in one of several ways, in order of preference:

1. The content server specifies a header saying ″cache-control: max-age=*xxx*″. This tells the proxy that the document is good for *xxx* seconds after it is received.
2. The content server specifies a header saying: ″Expires: *xxx*″. This tells the proxy that the document is good until the time specified by *xxx*.
3. The content server indicates when the document was last modified, using a ″Last-Modified: *xxx*″ header. The proxy server computes how long it’s been since the document was last modified, multiplies this by the `CacheLastModifiedFactor`, and assumes the document will be good for that long.

For example, if the content server said the document was last modified 1 week (7 days) ago, and `CacheLastModifiedFactor` is 0.14, then the proxy server will assume the document is good for about 1 day.

4. If none of the above information is specified by the content server, Web Traffic Express will look for the `CacheDefaultExpiry` directive that matches the current URL, and use that for the expiry time.

Notes:

1. **Attention:** Administrators should be very careful when setting `CacheDefaultExpiry` to anything other than 0 minutes for http: URLs. Many dynamically-generated pages include none of the headers mentioned above, so the `CacheDefaultExpiry` mechanism is used for those pages. Setting the `CacheDefaultExpiry` for that URL to more than 0 minutes will allow caching those pages, but this may mean that users get out-of-date content.
2. Very few documents come with a ″Cache-Control: max-age″ or ″Expires:″ header.
3. Almost all static Web documents (as opposed to dynamically-generated documents) include a ″Last-Modified:″ header. This is the most common way that proxies compute expiry times for documents.
4. Dynamically-generated pages, which frequently are not cacheable, may include a header saying ″Expires: 0″ or ″Cache-Control: no-cache″, which means the document expires immediately.

Once the expiry time has been computed, using the details above, Web Traffic Express checks to see if there is a `CacheMinHold` directive that applies for this URL. If there is, and the time it specifies is **longer** than the expiry time computed above, then the time in the `CacheMinHold` directive is used. This is true even if Web Traffic Express computes an expiry time of 0 minutes for a document.

Once the final expiry time is computed, this is checked against the time specified in the `CacheTimeMargin` directive. If the expiry time is greater than the `CacheTimeMargin` value, the document is cached; otherwise, it will not be added to the cache.

If the document is found in the cache, but expired, Web Traffic Express issues a special request known as an ″if-modified-since″ request to the content server. This request causes the content server to send the document only if it has been modified since it was last received by the proxy. If the document has not been modified, the content server sends a message indicating so, and does not send the entire page. Then, the proxy serves the cached document.

Cache indexing

Web Traffic Express's cache directory structure and its lookup methods are different from many other proxy servers. The proxy creates an index of the files in the cache to keep in memory as each page is added. Using RAM instead of other media, such as a hard drive, results in faster lookup and retrieval times. The index separates the cached file into the number of subcaches specified in the `ProxyNumTables` directive. The URL, expiry information, and its corresponding filename are stored in the index in memory. For this reason, memory (RAM) required is directly proportional to the number of files in the cache.

When a request is received from a client, the proxy checks the index in memory for that URL:

- If the file is not in the index, the request is made to the destination server
  - The retrieved URL is then checked to make sure the document is cacheable, and caches it if allowed
  - The index is then updated with new URL, subcache and expiry information
- If the file is in the index
  - The expiry information is checked to determine if the URL is stale
    - If the URL has expired, the destination server is contacted, and the URL is replaced by the newly retrieved document with expiry and subcache information updated in the index.
    - If the URL is consistent, the document is served.

The cache contains shadow files called **.cache_info** that mirror the index information for the proxy to use only when the server is started. After starting, that index is loaded into memory and, in the case of a server restart, does not need to be refreshed. The information in the shadow file includes the document expiry time, how large the file is, and the time when the server last checked the file. The garbage collection process updates the cached document index files.

Garbage collection

As part of the effort to keep cached documents fresh and minimize usage of system resources, Web Traffic Express performs a nightly cleanup process known as garbage collection. This process examines the files in the cache directory and attempts to eliminate expired files to reduce the size of the cache and make room for new files. The amount of pruning of expired files is governed by the system administrator via Web Traffic Express configuration directives.

The `GCMaxInUse` directive is used to set a maximum threshold for cache utilization. It specifies the maximum cache size to keep after garbage collection. The default is 75 percent of total cache utilization, meaning if a cache is 100 percent full, garbage collection must remove at least 25 percent of the files. The `CacheSize` and `CacheFiles` directives are used to limit the size of the cache in bytes or by number of files.

Chapter 4. Configuring and starting automatic cache refreshing

Most proxy servers have some kind of caching available. But Web Traffic Express has a ″cache agent″ that gives more control to the administrator. Usually, the only time a page is cached is *after* a user requests it. Web Traffic Express's cache agent can retrieve specified URLs before a user requests them and refresh them automatically.

Advantages

- Caching is applied to specified URLs without user requests for the pages.
- Updated URLs are supplied to users more quickly.

Disdvantages

- The server is busy even during hours of low activity as a result of pages being cached.

Web Traffic Express includes a cache refresh agent, which automatically handles reloading the proxy's cache. The cache agent, by default, is started every night at midnight local time. To change the time the cache agent is started, see "Starting the cache agent" on page 37. The automatic refreshing has two modes of operation: administrator-controlled and automatic.

Administrator-controlled operation allows the proxy administrator to give the server a certain set of pages to load. The proxy will retrieve those pages, and optionally inline images, when the cache agent starts.

The administrator could optionally instruct the server follow down the path of HTML links and request all of those pages. For example, the administrator could load http://www.netscape.com/, and the hierarchy 2 levels below. The cache would retrieve and replicate the first two levels of pages under Netscape.

With the automatic mode of operation, the proxy finds the most popular cacheable pages and automatically reloads them. The proxy accomplishes this by checking the cache-access log, sorts it by frequency of requests, and then picks the most popular pages. It can refresh the top *N* pages, where *N* is set by the administrator.

Note: You may use the administrator-controlled and automatic modes together. For example, specify a destination URL and check **Load linked pages off cached URLs** and then enter **2** for the number of link levels to follow. The server will cache the specified URL and also cache URL links two levels below on the same host. If **Follow links across hosts** is marked, all links will be cached to two levels.

Configuring the cache agent

You can use both automatic and administrator-controlled modes when running the cache agent.

Automatic mode

Automatic caching accesses finds the most popular cachable pages and automatically reloads them. This is accomplished by checking the cache-access log, sorting it by frequency of requests, and then picking the most popular pages. It can refresh the top *N* pages where *N* is a configurable value.

Following links and delving

When using automatic cache refreshing, you are prompted for the level you wish to ″delve into″. Delving is when the cache agent follows hypertext links from other cached pages. The majority of people do not stop when they have read information from one URL. That URL will usually have links going to sites with related information, and people follow the path linking from one page to another. Delving is a way to cache these logical information paths and keep them for others.

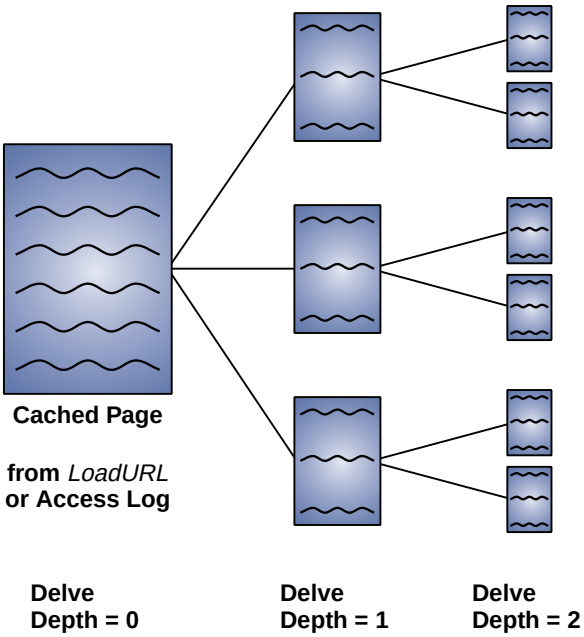

Figure 2. Following hypertext links on URLs is known as delving

When using the automatic mode of the cache agent, you can specify the maximum number of URL using the `MaxURLs` directive. The MaxURLs value is not checked until the cache agent starts delving into pages. This value should be larger than combining the `LoadURL` and `LoadTopCached` directives. If it is smaller, Web Traffic Express assumes the LoadURL and LoadTopCache directives are correct.

For example:

Directives in the javelin configuration file	Result

`LoadURL http://www.getthis.com/main.html`
`LoadURL http://www.getmetoo.com/welcome.htm`
`LoadTopCached 30`
`MaxURLs 50`
	`MaxUrls` has a valid value in the configuration file. If the cache agent has more than 30 unique URLs, it will retrieve **main.html**, **welcome.htm**, the **top 30 requested URLs** from the cache access log, and **up to 18 linked URLs** from earlier pages.

`LoadURL http://www.joesmith.edu/favorites.html`
`LoadURL http://www.janesmith.edu/`
`dislikes.htm`
`LoadTopCached 30`
`MaxURLs 25`
	`MaxUrls` has an invalid value in the configuration file. If the cache agent has more than 30 unique URLs, it will retrieve **favorites.html**,**dislikes.htm**, the **top 30 requested URLs**from the cache access log. No other files are retrieved because the value in `MaxUrls` has been exceeded.

`LoadURL http://www.hello.com/hi.html`
`LoadURL http://www.ballyhoo.com/index.html`
`LoadTopCached 30`
`MaxURLs 25`
	`MaxUrls` has an invalid value in the configuration file. If the cache agent has 20 unique URLs, it will retrieve **hi.html**, **index.html**, the **top 20 requested URLs** from the cache access log, and **up to 3 linked URLs** from the earlier pages.

To configure the automatic cache agent, use the Configuration and Administration forms:

1. Click **Proxy Configuration** from the Configuration and Administration form.
2. Under the **Web Traffic Express Settings** section, click **Cache Refresh**
3. Fill in or check the following fields
   - **Number of most popular URLs to cache from last night's cache log -** provides the cache agent with the number of pages from the previous night's access log to refresh.
   - **Load linked pages off cached URLs -** specify whether the cache agent will load linked pages from the URLs to be cached. Values are:

     always
     cache agent caches all linked pages off previously cached URLs

     never
     cache agent ignores all links on URL

     admin
     cache agent only follows links on URLs specified in the `LoadURL` directives

log
cache agent only follow links from URLs in the server access log

- **Number of link levels to follow -** If you specified always load linked pages, the cache agent will follow the links until it reaches the number specified in this field.
- **Follow links across hosts -** check this button if you would like the proxy to follow the links to different sites. For example, many sites have hypertext links to Web documents on other sites as well as their own. Use this instruction if you want the cache agent to follow the links to the other sites.
- **Insert delay between requests -** check this button if you want the server to insert delays between sending concurrent requests for documents. Inserting a delay decreases the load on the destination server, but it increases the time it takes for the proxy to return the document. Unless you have a slow connection, we recommend that you DO insert a delay between requests.
- **Attempt to retrieve inline images -** check this box if you want the server to cache the images available on the sites. Caching inline images speeds up retrieval because graphics take longer to load than text.
- **Refresh cache at midnight -** check this box if you want the cache agent to automatically start refreshing pages at midnight. If you do not want Web Traffic Express to start the cache agent, see “Starting the cache agent” on page 37.
- **Number of threads for requests -** provides the cache agent with the number of threads to use to make the requests. Each cache agent thread uses one server thread to make its own request so retrieving pages can occur simultaneously. However, the number of threads is limited to the amount of memory you have.
- **Maximum work queue depth -** specifies the maximum depth the proxy will maintain in the queue of URLs to request.
- **Maximum URLs to request -** sets the maximum number of URLs that are allowed in the proxy’s queue of URLs to request during a particular run.
- **Maximum time to request URLs -** sets the maximum time the cache agent has to complete a refresh run.

4. Click **Apply** to make the changes to the configuration file.

The changes will take effect for the next scheduled run.

Administrator-controlled mode

To configure the administrator controlled cache agent, use the Configuration and Administration forms:

1. Click **Proxy Configuration Server Settings**from the Configuration and Administration form.
2. Under the **Web Traffic Express Settings** section, click **Cache Refresh**.
3. Fill in or check the following fields:
   - **Insert before**, **Insert after**, **Replace**, or **Remove -** Specify the position you want the current entry to be placed in the index, or which entry you wish to remove from the index

- **Index -** Enter the index position
- **URL or IP address -** When removing an entry, leave this field empty. Enter the IP address or URL of the destination server you wish to cache or ignore for all actions except Remove.
- **Cache** or **Ignore -** To save the documents for later use, select cache. If you do not wish to save the URL, select ignore. When removing an entry, this field is not used.

4. Click **Apply** to make the changes to the configuration file.

The changes will take effect for the next scheduled run.

Starting the cache agent

The cache agent can be started automatically by Web Traffic Express or manually. The executable is:

On AIX *server_root*/**bin/cacheagt**

where *server_root* is the directory where you installed Web Traffic Express. The default path is:

```
usr/lpp/internet/server_root/bin/cacheagt
```

On OS/2 or Windows NT
*server_root***bin\cacheagt.exe**

where *server_root* is the drive and directory where you installed Web Traffic Express. For example, if you installed the proxy in C:\WWW\ the executable is:
`C:\WWW\BIN\cacheagt.exe`

By default, the cache agent is started at 12:00 midnight by Web Traffic Express. On AIX, you can automatically run the cache agent at a different time using the ″cron″ daemon. Cron jobs are specified by adding a line to the system **crontab** file. An example entry of the command file is:

```
45 16 * * * /usr/lpp/internet/server_root/bin/cacheagt
```

This would start the cache agent every day at 4:45 PM local time. For more information, see the man page on cron.

Related directives

Related directives are:

- `DelayPeriod`
- `DelveAcrossHosts`
- `DelveDepth`
- `DelveInto`
- `IgnoreURL`

- `LoadTopCached`
- `LoadURL`
- `LoadInlineImages`
- `MaxQueueDepth`
- `MaxUrls`
- `Proxy`

For more information and syntax on directives, see “Appendix A. Configuration directives” on page 69.

Chapter 5. Configuring PICS-based filtering

Some browsers allow you to determine what sites are viewable and which are not. Web Traffic Express provides the same content filtering control at the proxy server level.

Advantages

- Allows content filtering at the proxy server level based on PICS labels
- Selectively pass or fail certain URLs

Disadvantages

- Filtering decreases performance

Platform for Internet Content Selection (PICS) is an evolving set of specifications used to govern the creation and use of ratings for Web information. PICS development began in mid-1995 when the computing and online industries became sensitized to the possibility of online content censorship by the U.S. government and other governments around the world.

Before the PICS system was implemented, unique filtering solutions such as CyberPatrol and NetNanny combined ratings and filtering software. Now, because of its flexibility, the PICS standard may bring about many new labeling systems to further categorize content on the Web.

New PICS-compliant ratings systems are being introduced by companies such as the Recreational Software Advisory Council (RSAC), a non-profit organization. RSAC implements PICS labels using a number rating scale, allowing easy conversion of the label to different languages. The numbers act as a scale of varying degrees of content sensitivity. For example, if you want to block sites with violence, but think some foul language is acceptable, you may have a filter to accept sites with a violence maximum of 2, and a language maximum of 3.

Web Traffic Express's PICS filtering supports version 1.0 of the PICS Rules specification located at **http://www1.raleigh.ibm.com/pics/PICSRules_1.0.html**. It covers labeling documents for offensive content or simply for describing information in a machine-readable way.

Web Traffic Express allows an administrator to specify filtering rules based on PICS labels. When a URL is accessed, the proxy uses these rules to determine if it is passed or failed.

Some definitions

content filtering
: the action of allowing or stopping Web pages from loading based on a label that represents the ″appropriateness″

failed URL
: a URL that the proxy does not allow a client to view

label bureau
a server that issues PICS labels to different sites

passed URL
a URL that the proxy does allow a client to view

.RAT file
the individual file on a bureau's site that contain categories of filtering criteria and their corresponding values

rating service
an unaffiliated organization that supplies PICS labels and ratings on Internet sites

PICS label
the machine readable format the server uses to filter content

Proxy filtering

A distinction should be made between maintaining PICS filters at a proxy level and at a browser level. Some browsers have the ability to filter content using PICS as content is received from the destination. However, settings are also adjusted at the browser, and may be easily compromised.

Using a proxy server to determine if a URL is viewed takes the responsibility away from the client, and places it on the provider. This method ensures the client will only get the level of content specified at the proxy. Interaction between the client and the provider would be required to change the sensitivity of the filter.

When PICS filters are applied at the proxy level, the processing will be transparent to most users. They will either see the requested URL or they will see an ″403 — Blocked by Filtering Rule″ error.

The labels themselves are supplied to the server in differing ways. They can be stored locally on the proxy's hard disk, supplied by a ratings bureau, or even provided by the content server. Some URLs have labels embedded within their HTML under the <META> tag or in the response header.

If the proxy is set up to receive label from a service, information must be provided about the service. Usually, a service will have a separate URL for advertising the service, and one for where the PICS labels are actually stored. The advertising and information on the service is referred to as the ″Service URL″ and the site where the labels are served is referred to as the ″Bureau URL″.

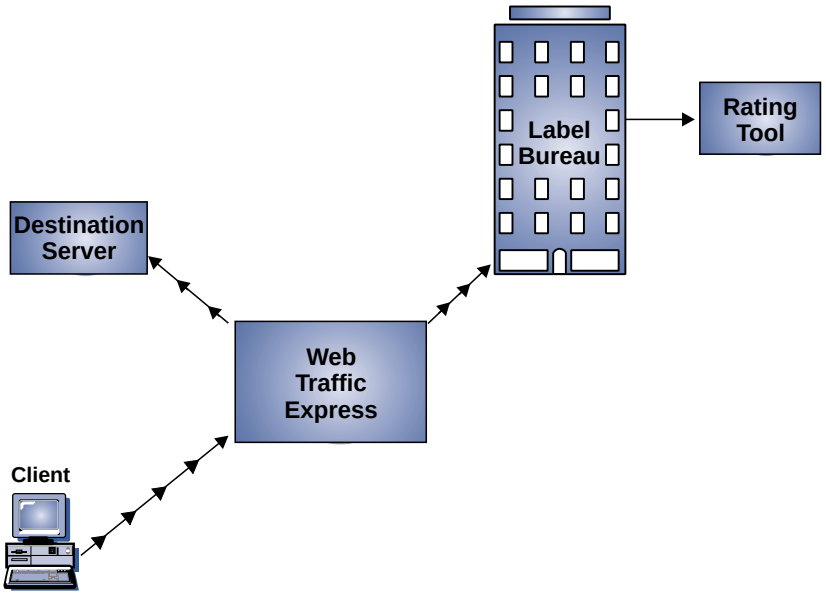

Figure 3. PICS filtering flow diagram

Label supplier API

Instead of using a label bureau, you can write your own label supplier API. This API will allow you to create and use your own PICS labels. When a request with a matching rule arrives, use the API to supply Web Traffic Express with it. This example API C code and other files necessary to compile it are available at www.ics.raleigh.ibm.com/WebTrafficExpress/.

Example

```
/*                    Labeler Exit                    Labeler.c
 *
 * This module provides a simple ICAPI routine which shows how to supply
 * labels to ICS. It looks at the name of the service requested, and
 * can return "labels" for the RSACi system and for the Coolness system.
 * Labels are in quotes because we aren't actually generating real labels.
 * They're syntactically valid labels, but we assign the same rating to
 * each and every page.
 *
 *
 * To build on OS/2: (using VisualAge C++)
 *     icc /Ge- /Gm+ /c labeler.c
 *     ILINK /NOFREE Labeler.obj, Labeler.dll, Labeler.map, httpdapi.lib,
 *     Labeler.def;
 * [httpdapi.lib is supplied with ICS]
 *
 * To build on Windows NT, compile Labeler.c, link with HTTPDAPI.lib, and use
```

```
 * module-definition file LabelerW.def.
 *
 * To build on AIX:
 *  cc_r -c Labeler.c
 *  cc_r -bM:SRE -bnoentry -o libLabeler.o Labeler.o -bI:libhttpdapi.exp -bE:Labeler.exp
 */
#include >
#include >
#include >
#include "HTAPI.h" /* Included with ICS. */

/* Rating service names and templates for labels.
 * These will be used to assemble outgoing labels.
 *
 * Note: none of these strings should ever be translated.
 */

    /* RSACi */
char *RSAC_Label_partA =
    "(PICS-1.1 \"http://www.rsac.org/ratingsv01.html\" l for \"";
char *RSAC_Label_partB =
    "\" r (n 0 s 0 v 0 l 0))";
char *RSAC_Name =
    "http://www.rsac.org/ratingsv01.html";

    /* Coolness */
char *Cool_Label =
    "(PICS-1.1 \"http://coolness.raleigh.ibm.com/ratings/V1.html\" l r (Coolness 1 Age-
range 1 Graphics 1))";
char *Cool_Name =
    "http://coolness.raleigh.ibm.com/ratings/V1.html";

/* Names of variables we'll extract from ICS using ICAPI.
 * PICS_SERVICENAME = the name of the rating service that a label is
 *     wanted from.
 * PICS_SITENAME    = the protocol & hostname part of URL that a label is
 *     needed for.
 * PICS_PATHNAME    = the path/filename part of the URL that a label is
 *     needed for.
 */
char *SVCNAME  = "PICS_SERVICENAME";
char *SITENAME = "PICS_SITENAME";
char *PATHNAME = "PICS_PATHNAME";

/* Helper functions to manage strings in the heap.
*/
static char *mystrdup(char *s) /* Make a copy of a string, in the heap. */
{
    char *ret = malloc(strlen(s) + 1);
    if (ret) {
        strcpy(ret, s);
    }
    return(ret);
}
```

```
static char *mystrcat(char *a, char *b) /* Glue together two strings; free the first. */
{
    char * ret = malloc(strlen(a) + strlen(b) + 1);
    if (ret) {
        strcpy(ret, a);
        strcat(ret, b);
    }
    free(a);
    return(ret);
}

/* This is the label-supplier exit. ICS will call this function when the
 * proxy server needs a label to make a filtering decision. If you compile
 * this module to "Labeler.dll", then you'd put the following into the ICS
 * config file (httpd.cnf/httpd.conf):
 *
 * OS/2, NT:
 *    PicsDBLookup                    Labeler.dll:BogusLabelSupplier
 * AIX:
 *    PicsDBLookup                    libLabeler.o:BogusLabelSupplier
 *
 * On OS/2 and NT, if Labeler.dll isn't in the directory you start ICS from,
 * then put a full pathname on it. On AIX, put libLabeler.o in /usr/lib, or
 * give a full path to the file.
 */
void HTTPD_LINKAGE BogusLabelSupplier(unsigned char *handle, long *return_code)
{
    char ReturnBuffer[1024];/* A str buffer to hold results from API functions. */
    char *URL = NULL;       /* The URL ICS is examining. */
    char *Service = NULL;   /* The URL of the rating service ICS is considering. */
    unsigned long l1, l2;   /* Lengths of various things. */
    long retc;              /* Stores return code from ICAPI functions. */

    /* Determine the URL in question. This is in two steps: the protocol/host
     * part, and the path/file part.
     */
    l1 = strlen(SITENAME);
    l2 = 1023;
    HTTPD_extract(handle,
        SITENAME, /* Fetch protocol/site part of the URL. */
        &l1,
        ReturnBuffer,
        &l2,
        &retc);
    if (retc != HTTPD_SUCCESS) {
        *return_code = 500;
        goto DropOut;
    }
    ReturnBuffer[l2] = '\0';
    URL = mystrdup(ReturnBuffer);
    if (!URL) {
        *return_code = 500;
        goto DropOut;
```

```
}

l1 = strlen(PATHNAME);
l2 = 1023;
HTTPD_extract(handle,
    PATHNAME, /* Fetch path/file part of the URL. */
    &l1,
    ReturnBuffer,
    &l2,
    &retc);
if (retc != HTTPD_SUCCESS) {
    *return_code = HTTP_NOACTION; /*Let others (if any) try to satisfy request. */
    goto DropOut;
}
ReturnBuffer[l2] = '\0';
/* And now assemble the full URL. */
URL = mystrcat(URL, ReturnBuffer);
if (!URL) {
    *return_code = HTTP_NOACTION; /* Let others (if any) have a shot. */
    goto DropOut;
}

/* Determine what service this label is for. */
l1 = strlen(SVCNAME);
l2 = 1023;
HTTPD_extract(handle,
    SVCNAME,
    &l1,
    ReturnBuffer,
    &l2,
    &retc);
if (retc != HTTPD_SUCCESS) {
    *return_code = HTTP_NOACTION;
    goto DropOut;
}
/* NUL-terminate the string... */
ReturnBuffer[l2] = '\0';
/* ...and make a copy of it. */
Service = mystrdup(ReturnBuffer);
if (!Service) {
    *return_code = HTTP_NOACTION;
    goto DropOut;
}

/* At this point, we should actually generate a real label. Since this
 * is an example program, we just return a hard-coded (fake) label
 * above. But we'll at least insert the URL into the label before we
 * send it...if it's an RSACi label.
 */

if (!strcmp(ReturnBuffer, RSAC_Name)) {
    char *RSAC_Label = mystrdup(RSAC_Label_partA);
    if (RSAC_Label) RSAC_Label = mystrcat(RSAC_Label, URL);
    if (RSAC_Label) RSAC_Label = mystrcat(RSAC_Label, RSAC_Label_partB);
```

```
        if (!RSAC_Label) {
            *return_code = HTTP_SERVER_ERROR;
/* At this point, we know we should've handled the request - but we couldn't. */
            goto DropOut;
        }
        l1 = strlen(RSAC_Label);
        HTTPD_supply_label(handle,
            RSAC_Label,
            &l1,
            &retc);
        if (retc != HTTPD_SUCCESS) {
            *return_code = HTTP_SERVER_ERROR;
            return;
        }
        *return_code = HTTP_OK;
    } else if (!strcmp(ReturnBuffer, Cool_Name)) {
        /* The Cool_Label doesn't need to be formatted as it doesn't
         * contain the URL...so just transmit it as-is.
         */
        l1 = strlen(Cool_Label);
        HTTPD_supply_label(handle,
            Cool_Label,
            &l1,
            &retc);
        if (retc != HTTPD_SUCCESS) {
            *return_code = HTTP_SERVER_ERROR;
            goto DropOut;
        }
        *return_code = HTTP_OK;

    } else {
        /* Label requested from an unknown service...we'll just indicate
         * that we have no knowledge.
         */
        *return_code = HTTP_NOACTION;
    }

DropOut:
    /* Free anything that we've got left lying around on the heap. */
    if (URL) free(URL);
    if (Service) free(Service);

    /* And return. ICS will examine the value of "return_code" to tell
     * if we were able to provide a label.
     */
    return;
}
```

For more information on writing API's, see the *Web Programming Guide Version 4.2* at www.ics.raleigh.ibm.com/pub/icswpg42.htm#HDRICAPI

Configuring and maintaining PICS filters

PICS rules are processed in the order they appear in the configuration file. Web Traffic Express picks the first applicable rule for each request and applies it to the URL. For this reason, you should order your rules from specific to general.

There are several ways to manage PICS labels. In general, the procedure is easiest through the Configuration and Administrative forms.

To create and maintain local filters

1. Click **Proxy Configuration** from the Configuration and Administration forms.
2. Under the **Web Traffic Express Settings** section, click **PICS Filter Control**.
3. Click **Manage PICS filters**
4. Select one of the following actions:
   - **Create a new filter** Enter where to list the new filter, **before** or **after** the value entered in the **Index** field
   - **Copy an existing filter** Select the value for the index entry to copy in the table.

     Enter where to list the new filter, **before** or **after** the value entered in the **Index** field
   - **Edit an existing filter** Select the value for the location in the table.
   - **Delete an existing filter** Select the value for the location in the table.
5. Click **Apply** to go to the next form.
6. Choose your next step
   - If you deleted or copied a filter, restart your server
   - If you created a new filter or edited an existing one:
     a. Enter or edit the following fields:
        – **Filter Name -** name that is easily remembered. For example, *NoJunk*
        – **Filter Description -** short description for the current filter. For example, *filter to cut non-business related Internet activity*
        – Optionally, check the **days the filter is active**.
        – Optionally, enter the **start and end time** for the filter.
        – The table displays the users' Internet addresses to which this filter applies. These users can be changed on the next form.
        – Continue by pressing the **Apply** button
     b. Ensure the fields are correctly entered. If they are incorrect, follow the **settings** link and repeat the step.
     c. To update the list of users, click **user list**
        1) Select one of the following actions: **Insert Before, Insert After, Replace, or Remove**
        2) Enter the location in the **Index** field
        3) Enter the **User ID and hostname**

4) Click **Apply**

d. To list the URLs to allow using this filter, click **Pass URL**

1) Select one of the following actions: **Insert Before, Insert After, Replace, or Remove**
2) Enter the location in the **Index** field
3) Enter the User ID and hostname
4) Enter the **URL to always Pass**. For example: *http://www.passthis.com*
5) Click **Apply**

e. To list the URLs to deny using this filter, click **Fail URL**

1) Select one of the following actions: **Insert Before, Insert After, Replace, or Remove**
2) Enter the location in the **Index** field
3) Enter the User ID and hostname
4) Enter the **URL to always Fail**. For example: *http://www.failthis.com*
5) Click **Apply**

f. To list the URLs to conditionally filter using this filter, click **Conditionally filter**

1) Select one of the following actions: **Insert Before, Insert After, Replace, or Remove**
2) Enter the location in the **Index** field
3) Select the **Service Information definition** from the list
4) Click **Apply**

To maintain service information definitions

1. Click **Proxy Configuration** from the Configuration and Administration forms.
2. Under the **Web Traffic Express Settings** section, click **PICS Filter Control**.
3. Click **Service Information Definition**.
4. Select one of the following actions:
   - **Create a new definition** Enter where to list the new definition, **before** or **after** the value entered in the **Index** field
   - **Copy existing definition -** Select the value for the index entry to copy in the table.

     Enter where to list the new definition, **before** or **after** the value entered in the **Index** field
   - **Edit existing definition -** Select the value for the location in the table.
   - **Delete existing definition -** Select the value for the location in the table.
5. Click **Apply** to go to the next form
6. Choose your next step
   - If you deleted or copied a filter, restart your server
   - Enter or check the following fields:
     – **Short name -** common name for the service. For example: `RatingService`

- **Service name URL -** the URL that gives information on the label service being provided. This URL may not be used for actual label retrieval, but may provide information of the labels themselves. For example: `http://www.rating.com/pics_labels`
- **Bureau URL -** the URL where the service stores the labels
- **Ratings file -** the file used to categorize criteria for filtering. For example: `rating.rat`
- Click **Apply** to make the change to the configuration file

Writing a PICS rule

PICS rules can be created by the Configuration and Administration forms (recommended) or by editing the javelin configuration file `DefinePICSRule` directives.

This section describes basic syntax for PICS rules step by step. More detailed information on syntax for PICS rules can be found at **http://www1.raleigh.ibm.com/pics/PICSRules_1.0.html**.

- **Step 1:** The filter name is provided

```
DefinePICSRule filtername {
}
```

- **Step 2:** Any pass or fail rules are added using passURL and failURL. A passURL statement will always pass the specified URL and the FailURL will always fail it. Wildcards can be used and multiple pass and fail instructions are allowed.

```
DefinePICSRule filtername {
           passURL ("http://www.passthisURL1.com")
           passURL ("http://*.passthis.com/*")
           failURL ("http://www.failthisURLa.com")
}
```

- **Step 3:** When you want to conditionally filter using PICS labels from a service, service information must be inserted into the rule. The available-with-content variable informs the rule if the label is provided within the HTML <META> tag of the requested page.

  Note: You must also insert a passURL statement to ensure the service URL is passed

```
DefinePICSRule filtername {
     passURL ("http://www.passthisURL1.com")
     passURL ("http://*.passthis.com/*")
     failURL ("http://www.failthisURLa.com")
     passURL("http://otherserver.com/Ratings")

    serviceinfo (
```

```
            name "http://www.ratings.com/pics/pics_service.html"
            shortname "RatingService" available-with-content "YES"
            bureauURL "http://otherserver.com/Ratings"
            ratfile "filename.rat"

                )
}
```

- **Step 4:** The next addition to the rule is the filter name information. The *rulename* is what the user sees when the page is failed or blocked. The *description* is a text description for the rule writer.

```
DefinePICSRule filtername {
     passURL ("http://www.passthisURL1.com")
     passURL ("http://*.passthis.com/*")
     failURL ("http://www.failthisURLa.com")
     passURL ("http://otherserver.com/Ratings")

     serviceinfo (
            name "http://www.ratings.com:80/pics/pics_service.html"
            shortname "Rating Service" available-with-content "YES"
            bureauURL "http://otherserver.com/Ratings"
            ratfile "filename.rat"
                 )

     name (
           rulename "Language is greater than 3"
           description "fail the url when extreme language is used"
          )

}
```

- **Step 5:** The filter describes the operation done on the category to be filtered in order to be passed to the requester. The operator can be: >, <, >=, <=, !=, ==, ||, &&, AND, or OR.

  The *RatingService.language* is a language category held in the RAT file (on the label bureau) for the service with a short name of *RatingService*. The rat file contains information on the criteria, or categories, of content being filtered. These categories are then given allowable ranges of values.

  The label bureau uses these categories and rates the sites in respect to those categories. The result is a PICS label that has information on different categories.

  The filter reads the label category and compares the value in the label in an operation in the *filter* statement. The URL is then passed or failed.

  Two operations can be performed in the filter statement: *pass* and *block*. This example uses the pass statement.

```
DefinePICSRule filtername {
     passURL ("http://www.passthisURL1.com")
     passURL ("http://*.passthis.com/*")
     failURL ("http://www.failthisURLa.com")
```

```
     passURL ("http://otherserver.com/Ratings")

     serviceinfo (
            name "http://www.ratings.com:80/pics/pics_service.html"
            shortname "RatingService" available-with-content "YES"
            bureauURL "http://otherserver.com/Ratings"
            ratfile "filename.rat"
                 )

     name (
           rulename "Language is greater than 3"
           description "fail the url when extreme language is used"
          )

     filter (
            pass ("RatingService.language <= 3")
           )
}
```

- **Step 6:** This filter uses the block statement.

```
DefinePICSRule filtername {
     passURL ("http://www.passthisURL1.com")
     passURL ("http://*.passthis.com/*")
     failURL ("http://www.failthisURLa.com")
     passURL ("http://otherserver.com/Ratings")

     serviceinfo (
            name "http://www.ratings.com:80/pics/pics_service.html"
            shortname "RatingService" available-with-content "YES"
            bureauURL "http://otherserver.com/Ratings"
            ratfile "filename.rat"
                 )

     name (
           rulename "Language is greater than 3"
           description "fail the url when extreme language is used"
          )

     filter (
           block("RatingService.language >3)"
           )
}
```

Passed URLs	Failed URLs
http://www.passthisURL1.com	Any URL with a language rating above 3
http://otherserver.com/Ratings	the URL http://www.failthisURLa.com
any URL on the passthis.com domain	
any URL that has a language rating of 3 or less	

Both of the filters in Step 5 and 6 will have the same results when a rating is provided. The difference is their behavior when a rating for the page is unavailable.

- In the pass statement, the comparison must be true for the URL to be passed
- In the block statement, the comparison must be true for the URL to be blocked

Result: Because the comparison is false without a *RatingService.language*, the pass statement (Step 5) would FAIL the URL while the block statement (Step 6) would PASS the URL.

Chapter 6. Configuring flexible SOCKSification

Flexible-client SOCKSification of a proxy allows the proxy server to reside behind a firewall or SOCKS server without sharing the same physical machine. Requests going to the proxy can then be routed directly to the destination content server, instead of interrupting the SOCKS server.

Advantages:

- Helps in security by allowing a firewall server to be isolated from the proxy server
- Reduces load on the firewall machine by having the proxy run behind the firewall
- Allows the administrator to specify which requests go through the SOCKS server and which are redirected back to the local domain

Disadvantages

- Requires additional hardware
- Higher latency for requests

On AIX , the SOCKS configuration file is **socks.conf**. On OS/2 and Windows NT, the SOCKS configuration file is **socks.cnf**

Web Traffic Express's SOCKSification lets you specify which IP addresses or domains should be contacted directly without the help of the SOCKS server. It is located the ETC directory or in the same directory as the **httpd** and **javelin** configuration files. If your system does not have a **socks** configuration file upon installation, a default SOCKS configuration file will be installed. The default file contains nothing but comments, and syntax. See "Using Configuration and Administration forms" on page 54 and "Manually editing the **socks** configuration file" on page 55 for information on changing the **socks** configuration file.

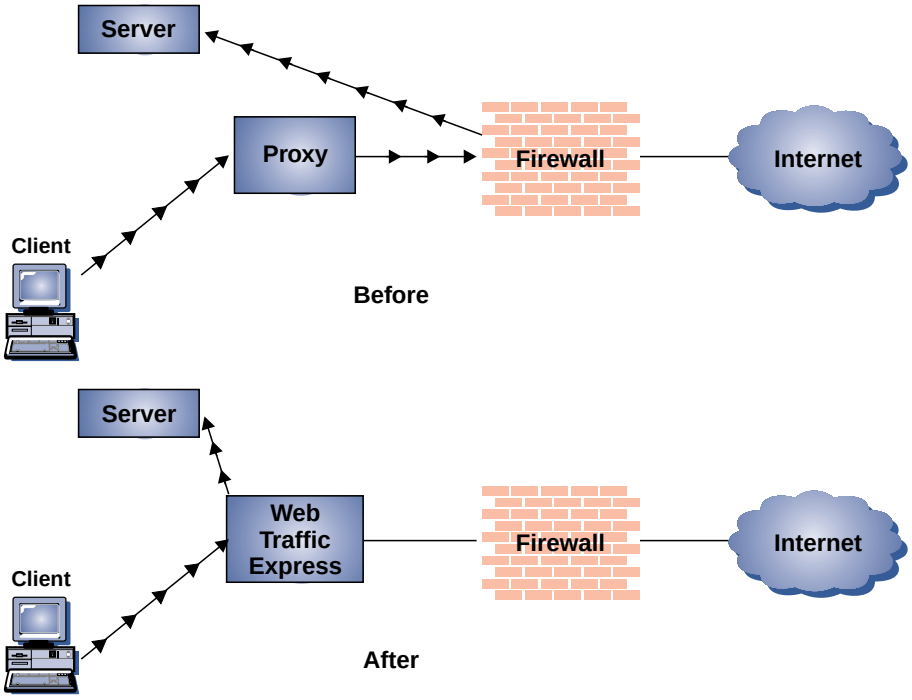

Figure 4. Flexible SOCKSification

Version 4.0 of OS/2 has SOCKSification built into the operating system. We recommend disabling this SOCKS configuration and using the configuration file installed with Web Traffic Express.

Using Configuration and Administration forms

Note: For help in configuring socksification, you may need to contact your network administrator.

To configure flexible socksification, you can use the Configuration and Administration forms:

1. Click **Proxy Configuration** from the Configuration and Administration forms.
2. Under **Web Traffic Express Settings**, click **SOCKS Servers**.
3. There are two tables on this form:
   - **Direct Connections** for requests NOT passing through the SOCKS server.
   - **SOCKS Connections** for requests to pass through the SOCKS server.
4. Specify where in the index you want the entry inserted or deleted

5. If you are adding or replacing an entry, enter the following fields:
   a. **Host Name -** enter the hostname of the destination server in the field
   b. **Subnet Mask -** enter the destination mask
   c. **Port and Port Number -** these are optional fields to specify the port to connect
   d. **SOCKS servers to check -** when specifying URLs to pass to the SOCKS servers, enter the name of the SOCKS server. If you specify more than one SOCKS server, URLs are passed to them in the order they appear until one responds. Multiple SOCKS server entries must be separated by a comma with no spaces between.
6. If you are deleting an entry:
   a. Select the **Remove** radio button
   b. Select the **Index** of the entry to remove
   c. Click **Apply** to make the change in the configuration file

Manually editing the socks configuration file

You can also change the SOCKS configuration manually. Web Traffic Express installs a default **socks** configuration file if you do not have one. You must change the default file to implement SOCKS. This file can be edited using a text editor.

Each line in the file may be up to 1024 characters long. Lines starting with a '#' are comments. Non-comment lines must be one of the three forms:

```
deny dstaddr dstmask [op dstport]
direct dstaddr dstmask [op dstport]
sockd @=serverlist dstaddr dstmask [op dstport]
```

where:

dstaddr
is the destination IP address to be masked

dstmask
is the mask used to filter the requested address. A 255 in a bit field requires an exact match for the *dstaddr*, and a 0 in a bit field means that field should be ignored. Specifying 255.0.255.0 means match the first and third fields in the IP address before performing the action

serverlist
is a list of the SOCKS server IP addresses, separated by commas. These servers are processed in the order they are listed

op is an optional condition used with *dstport*. *op* is the operation applied to the destination port number:

- eq - equal to
- neq - not equal to
- lt - less than

- gt - greater than
- le - less than or equal to
- ge - greater than or equal to

dstport
is optional specifying either the port number or the service name specified in the system SERVICES file.

Examples

```
direct 9.0.0.0 255.0.0.0
```

will attempt direct connections for all IP addresses beginning with a 9.

```
deny 5.67.132.4 255.255.255.255
```

will deny any connections to the IP address 5.67.132.4 and an error code will return an error code to the requester

```
sockd @=socks.yourdomain.com 9.67.193.2 255.255.255.0 gt 24
sockd @=9.12.133.2,9.25.11.133 9.120.130.2 0.0.0.0
```

The first **sockd** statement will only pass the request to the SOCKS server if the requesting IP address begins with 9.67.193.*, and if the port is greater than 24. The last bit setting of the IP address is ignored.

The second statement will pass the requests to the SOCKS server 9.12.133.2 first. If a connection cannot be established, 9.25.11.133 will be contacted. The request will be sent through the server only if the requesting address is 9.120.130.2

Attention: no spaces are allowed between the @= and the first server in the list.

See the default SOCKS configuration file for more examples. On AIX , the SOCKS configuration file is **socks.conf**. On OS/2 and Windows NT, the SOCKS configuration file is **socks.cnf**

Chapter 7. Advanced Settings

Web Traffic Express allows several options for information that actually passes through the proxy, and information that is kept hidden from the destination.

Advantages

- Increase client anonymity.

Disadvantages

- Some pages use header information to customize material on their site; these customizations cannot be loaded.

Header generation occurs automatically when a request is sent. An example header:

```
User-Agent: Mozilla 2.02/OS2
Client-IP: 9.37.192.3
Referer: http://www.ics.raleigh.ibm.com/WebTrafficExpress/main.html
Server: Web Traffic Express/1.0
```

Descriptions of the fields in this header are

- `User-Agent:` provides browser and operating system information
- `Client-IP:` provides the IP address of the client requesting the URL.
- `Referer:` provides the destination server with the URL of the referring link to this page
- `Server:` provides information on the client's server used to access the Internet

All headers can be blocked using advanced settings. However, some header fields are required.

Configuring advanced settings

1. Click **Proxy Configuration** from the Configuration and Administration forms.
2. From the **Web Traffic Express Settings** section, click **Advanced Setting**.
3. Enter or check the following fields relating to specific header information:
   - **Override client reload requests -** check this box if you wish the server to ignore the client's request to reload a URL. Under periods of high load, it may be more efficient to allow the proxy to ignore such a request and concentrate on serving other documents.
   - **Forward client IP address -** check this box if you want the requesting client's IP address to be forwarded to the destination server. If you do not check this box, the destination server will receive the IP address of the proxy server. This helps increase client's anonymity while surfing the Web.
4. **Client HTTP header to block-** Add to or remove from the list of HTTP headers to block in the table by:
   a. Selecting the **add** or **remove** radio button

b. If you are adding an header, enter the **HTTP header name** to add and the **Index** position

c. If you are removing a header, enter the **Index** position

d. Select **Apply** to continue

5. **User-agent string -** Enter the string to send in the header to the destination server to replace the type of browser and operating system a client is using. For example: specifying ″Web Traffic Express/1.0″ would replace Mozilla 2.02/OS2 in the following header:

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

6. **From:** Enter the email address the destination server reads when it parses the From: header. You may wish to specify the email address of the proxy administrator in this field, because any problems should be reported to them.
7. Click **Apply** to make the changes to the configuration file.
8. Restart your server.

Related directives

Directives for the advanced functions are:

- NoProxyHeader
- ProxyFrom
- ProxyIgnoreNoCache
- ProxySendClientAddress
- ProxyUserAgent

Chapter 8. Monitoring and tuning your proxy

The Server Activity Monitor consists of multiple pages that contain information about the activity of the server. Proxy-specific pages are also included in this monitor.

Click **Server Activity Monitor** under the **System Management** section of the server Configuration and Administration form.

Several pages make up the server activity monitor. You may click **Refresh now** on each page at any time to update the information. Below the description of each page are directives you can manipulate to further enhance performance. See “Appendix A. Configuration directives” on page 69.

- **Basic Status** provides basic information on thread and connection activity, request statistics, and server response time. More detailed inspection of some of these items can be found on other monitor pages described below. By changing the following directives, you can customize your server for your needs:
  1. In the **httpd** configuration file:
     - `MaxActiveThreads` specifies how many threads are in the server pool. You increase or decrease the number of threads available depending on how much memory you have. This directive will have an effect on many of the statistics on this page.
  2. In the **javelin** configuration file:
     - `ProxyPersistence` specifies whether the proxy will allow persistent connections from a client. This directive may have an effect on network throughput, depending on usage.
- **Network Status** provides information about the network the proxy is running on such as data rate, number of packets, and bytes sent and received. Using a socks configuration file increases network performance:
  1. In the **javelin** configuration file:
     - `FlexibleSocks` specifies whether the proxy will use the standard SOCKS configuration file when retrieving requests. Using flexible SOCKSification reduces request latency and interruptions to the SOCKS server.
  2. In the **SOCKS** configuration file:
     - Specify the hosts to connect to directly, which to pass to the SOCKS server, and which to deny connections to altogether. See “Manually editing the **socks** configuration file” on page 55.
- **Access Log** provides information on users accessing your proxy; IP addresses, userid (if protected), data and time of access, and the method (GET, POST) performed. By changing some of the following directives, you can customize what is displayed:
  1. In the **httpd** configuration file:
     - `AccessLogExcludeURL` specifies URLs you wish to suppress from the Access Log report

- `AccessLogExcludeMethod` specifies methods you wish to suppress from the Access Log report. For example, you can suppress logging of all POST requests
- `AccessLogExcludeMimeType` specifies mime types you wish to suppress from the Access Log report. For example, you can suppress logging of all GIF or JPEG files
- `AccessLogExcludeReturnCode` specifies return codes you wish to suppress from the Access Log report
- `AccessLogSizeLimit` specifies maximum size for the Access Log

You can see more directives for the access logs in ″Chapter 3. Using the configuration file″ of the *Webmaster's Guide*.

- **Proxy Log** provides information on the proxy activity such as which URLs were requested and if they were retrieved from the cache. Following the URLs are the return codes given to the client, and file size in bytes. Error codes are located in the *Webmaster's Guide*. By changing some of the following directives, you can increase the number of cache hits:

  1. In the **javelin** configuration file
     - `AutoCacheRefresh` turns the automatic mode of the cache agent on or off. The automatic mode will refresh a number of URLs from the previous night's access log. If clients are requesting many of the same documents everyday, this will increase cache hits.
     - `CacheMinHold` allows you to override the expiry information in the header of URLs. Some sites routinely force documents to immediately expire when they actually have a longer lifetime. You can specify the URL mask for the URL to override and the time to keep the file. When you override the expiry information, clients could receive stale data.
     - `PureProxy` determines whether the proxy is just a proxy, or a proxy and content server. We do not recommend using Web Traffic Express as a content server, and the `PureProxy` directive is defaulted to `On`.

  Although it is possible to run your proxy as a content server proxy combination, we recommend that you use Web Traffic Express as a proxy only. Performance will increase if the machine is dedicated in function.

  Web Traffic Express supports persistent connections, where a separate thread is used to keep each client connection open to the proxy. In most cases, loading documents on your intranet is faster than documents on the Internet; however, sites on your domain may be slow, so you may wish to cache them. If your organization has an internal network running Web servers, they are considered to be on the same domain. You can instruct the cache agent to ignore sites on your domain when refreshing the cache.

- **Cache Status** provides information on the cache and index.
  - Whether the cache is currently operational, still being reindexed from a server start, or if garbage collection is running. For example: `Cache operational`
  - How many subcaches are currently being utilized.

- A table showing the separate subcaches, their current size, the number of files in the subcache, and the size of the index for the subcache.
- Whether any of the subcaches are full. For example: `None of the subcaches are full`

By changing some of the following directives, you can customize your server for your needs:

1. In the **httpd** configuration file,
   - `Caching` turns caching on or off
   - `CacheSize` sets the maximum cache size in drive space
2. In the **javelin** configuration file,
   - `ProxyNumTables` sets the number of subcaches your cache is divided into. If you specify a `0` for this directive, Web Traffic Express will determine the number automatically based on the number of files in the cache.
   - `CacheFiles` sets the maximum cache size in number of files

- **Garbage Collection Summary** provides:
  - Starting time of last garbage collection
  - Ending time of last garbage collection
  - Number of files, directories, and bytes in the cache after garbage collection last ran
  - The size of the cache as a percentage of the maximum cache size
  - Number of files, directories, and bytes removed during garbage collection
  - Memory used during garbage collection

Garbage collection must have been run for this page to display any information. By changing some of the following directives, you can customize your server for your needs:

1. In the **httpd** configuration file:
   - `GCDailyGC` sets the time garbage collection begins. This is set in local time using a 24 hour clock. For example, 16:30 is 4:30 PM
   - `GCMemUsage` specifies the amount of memory (Kilobytes) the garbage collection process uses. Smaller values are less efficient, but larger values require more resources.
2. In the **javelin** configuration file:
   - `CacheMinHold` allows you to override the ″expires″ information in the header of URLs. Some sites routinely force documents to immediately expire when they actually have a longer lifetime. You can specify the URL mask for the URL to override and the time to keep the file. When you override the expiry information, clients could receive stale data.
   - `GCMaxInUse` determines the maximum cache utilization **after** garbage collection has completed.

- **Cache Refresh Summary** provides information on the cache agent's last run. The cache agent must have run at least once to display any information. By changing some of the following directives, you can customize your server for your needs:
  1. In the **javelin** configuration file:
     - `CacheLocalDomain` instructs the cache agent whether to cache pages on the local domain or not. If most of the traffic on your Intranet is to local sites, you should specify `CacheLocalDomain On`.
     - `LoadURL` specifies a URL the cache agent will refresh on the next refresh run. If many clients are requesting the same page, but for some reason are not included in the access log, you can manually set the URL to be refreshed. Multiple `LoadURL` directives are allowed.
     - `LoadTopCached` specifies the number of most popular URLs to load using automatic cache refreshing
     - `MaxRunTime` specifies the maximum time the cache agent is allowed to run. A value of `0` would allow the cache agent any amount of time to complete. Cache refreshing is memory intensive so if you are refreshing a very large cache, you may wish to limit the time the cache agent runs so normal operating hours are not interrupted.

Configuring performance using Configuration and Administration forms

1. Click **Proxy configuration** from the Configuration and Administration form.
2. From the **Web Traffic Express Settings**, click **Proxy Performance**.
3. Fill in or check the following fields:
   - **Run as a pure proxy -** mark this box if you want to increase proxy performance by strictly running a proxy server. We recommend you run as a pure proxy.
   - **Allow persistent connections -** use this setting to allow clients to force the server to keep an open connection with them. This decreases client's lag time associated with requesting documents from the proxy. However, persistent connections require network bandwidth as well as a dedicated server thread to maintain. Do not allow persistent connections if your setup limits the number of available threads.
   - **Cache Local domain -** check this box if you want the proxy to cache URLs from your local domain. Web documents on your Intranet, for example, usually load faster and do not need to be cached.
   - **Use a SOCKS configuration file -** check this box if you want the proxy to look at the SOCKS configuration file to decide whether or not to connect through the SOCKS server.
   - **Specify maximum number of files to cache -** specify the maximum number of files the proxy will cache. Files will still be swapped in and out of the cache if the garbage collection is configured correctly
   - **Specify number subcaches**-specify the number of sub-caches you want the proxy to use.

- **Specify the amount of cache in use after garbage collection -** specify the percentage of cache space you want utilized following the removal of expired documents.
- **How long documents reside in the cache -** some pages have a ″no-cache″ or ″expiry=0″ command in the header information to require the server to reload the document every time it is requested. You can override these document's commands by specifying a minimum number of time to keep documents in the cache.

4. Click **Apply** to make the changes to the configuration file

Server and proxy threads

A process is started every time you start a program, such as Web Traffic Express. The operating system can then keep track of information on any application running. Some processes can create ″threads″ to perform multiple tasks simultaneously. Web Traffic Express is a multi-threaded server, making it more reliable by controlling behavior of threads which time-out.

Web Traffic Express creates additional threads to complete smaller tasks. One of these threads is dedicated to document retrieval. Worker threads are created and are responsible for handling individual URL requests.

You can specify the number of worker threads using the `MaxActiveThreads` directive in the **httpd** configuration file. Please note memory consumption will increase as the number of threads increases.

Chapter 9. Answers to common questions

This section answers common questions you may have about Web Traffic Express. There are additional forums for Web Traffic Express at www.ics.raleigh.ibm.com/WebTrafficExpress/ where you can post any questions you have, and they will be answered by an IBM product expert.

Q. Where are my configuration files?

A. The configuration files for Web Traffic Express are located in the system ETC directory by default. The names are

On AIX in the /ETC/ directory:	httpd.conf javelin.conf socks.conf ics_pics.conf
On Windows NT in the \WINNT\ or \WIN35\ directory on the boot drive:	httpd.cnf javelin.cnf socks.cnf ics_pics.cnf
On OS/2 in the directory specified by the SET ETC= statement in the **config.sys** (usually \MPTN\ETC\ on the boot drive):	httpd.cnf javelin.cnf socks.cnf ics_pics.cnf

Q. I would like to run multiple instances of the server (each on a different port) within the same machine using unique configuration files. How do I move my configuration files to another directory?

A. You can point to a different location when you start the server using the -r option. For example, on OS/2:

```
start httpd -r d:\www\httpd.cnf
```

The Javelin and PICS configuration files must be located in the same -r directory as the server configuration file. If the httpd configuration file is located in d:\www, then the javelin ics_pics configuration file must be located there also.

Q. My proxy is not caching any files

A1. Make sure `Caching` is on and follow the steps in “Chapter 2. Configuration quick start” on page 23.

A2. Make sure the userid the proxy is running on can write to the directory specified in the `CacheRoot` directive in the **httpd** configuration file. For example on AIX, the default userid is ″nobody″, so you must give all users read write access to the `CacheRoot` directory.

Q. I've put a new PICS rule in the configuration file, but it had no effect on the content being passed

A. When directives that are used more than once conflict, unexpected or unwanted results may occur. Directives are processed in the order they appear in the configuration. For example:

```
DefinePICSRule passthis{
      passURL("http://www.passthis.com/*")
}
DefinePICSRule failthis{
      failURL("http://www.passthis.com/failme.html")
}
```

The failURL statement will never be read because the passURL statement above it satisfies the rule for any matching request.

Also:

```
CacheDefaultExpiry http:* 14 days
CacheDefaultExpiry http://some.host.net/*  6 days
```

The first directive applies to ALL HTTP requests. The proxy will never see the second directive. To avoid this problem, enter the directives in order from most specific to most general.

Q. I keep getting ″Error 403 - File not found″ when trying to load a page

A. Replace the `Pass http:*` directive with `Proxy http:*` in your httpd configuration file. Unless you proxy the request, Web Traffic Express will not retrieve it.

Q. Why can't I retrieve pages outside of the firewall?

A1. You must have a SOCKS configuration file with the name of a SOCKS server specified. See "Chapter 6. Configuring flexible SOCKSification" on page 53.

A2. Make sure your proxy can resolve external hostnames.

Q. How can I do a stand-alone test of my server?

A. A hardware adapter is not required for a stand-alone test or demo of your server on OS/2. Use the TCP/IP loopback interface IP address and the following steps:

1. Enter **ifconfig lo 127.0.0.1** at the OS/2 command line.
2. When your server comes up, use your favorite browser and go to URL **http://127.0.0.1** to view your server's Front Page.

Q. How do I increase my paging space?

A. **On AIX,** using the System Management Interface Tool (SMIT) or the command line interface, SMITTY:

1. Select the System Storage Management (Physical and Logical Storage) option on the main menu.
2. Under Logical Volume Manager, select the Paging Space option:
   - To change an existing paging space, select Change/Show Characteristics of a Paging Space.
   - To add a paging space, select Add Another Paging Space

On Windows NT

1. Bring up the control panel and open the **System** object
2. Click on the **Performance** tab
3. Under **Virtual Memory**, the current paging space will be displayed
4. Click **Change...**
5. Select the drive to use for the paging file, enter the initial and maximum settings, and click **Set**. Windows NT allows you to use multiple paging files on different drives. Create paging files as needed.
6. Click **OK**
7. Click **OK** to close the **System** object
8. Restart Windows NT

On OS/2, paging space is managed dynamically. Ensure the SWAPPATH= statement in the **config.sys** points to a drive with the required amount of disk space. If possible, paging space should be on a disk partition that contains no other files.

Q. I can connect to sites using IP addresses, but not names

A. Ensure that you have configures DNS correctly on the proxy.

Q. I can't connect to secure pages

A. Web Traffic Express does allows SSL Tunneling. Make sure

```
Proxy *.443
Enable CONNECT
```

directives are set in the **httpd** configuration file.

Q. The cache agent is not refreshing cached pages

A. You must have the `CacheAccessLog` directive specified in your httpd configuration file for the cache agent's automatic mode to work.

Q. Requests to the label bureau seem to hang when the server is being used as a label bureau and a proxy.

A. To allow Web Traffic Express a chance to filter with the above scenario, you must pass the label bureau URL using the `PassURL` statement in the rule. See "Writing a PICS rule" on page 48 for more information.

Appendix A. Configuration directives

This chapter describes each proxy server directive. The proxy configuration directive are split into 2 configuration files: **httpd** and **javelin**. SOCKS configuration is covered in “Chapter 6. Configuring flexible SOCKSification” on page 53. The directives are grouped according to the configuration file they reside in. Within each group, the directives are listed alphabetically.

Notes:

1. On AIX, the files are httpd.conf, and javelin.conf
2. On Windows NT, and OS/2 Warp, the files are httpd.cnf and javelin.cnf

HTTPD server proxy directives

These directives are in the **httpd** configuration file.

On AIX	**httpd.conf**
On Windows NT and OS/2:	**httpd.cnf**

CacheDefaultExpiry - Specify default expiration time for files

Use this directive to set a default expiration time for files that the server did not give either an Expires or a Last-Modified header to. You specify a URL template and the expiration time for files with URLs that match the template. You can have multiple occurrences of this directive in the configuration file. Include a separate directive for each template. The URL template must include the protocol. Specify the time value in any combination of months, weeks, days, and hours.

Examples

```
CacheDefaultExpiry ftp:* 1 month
CacheDefaultExpiry gopher:* 10 days
```

Initial configuration file setting

```
CacheDefaultExpiry ftp:*  1 day
CacheDefaultExpiry gopher:*  1 day
CacheDefaultExpiry http:*  0 days
```

Notice in the above defaults that the default expiration for HTTP is 0. HTTP should be kept at 0 because many script programs don’t give an expiration date, yet their output expires immediately (a value other than 0 may cause clients to see out of date content).

CacheExpiryCheck - Turn cache expirations off

Use this directive to specify whether you want the server to return cached files that have expired. Specify `Off` for the value if you want the server to be able to return

expired files. Use the default value of `On` if you do not want the server to return expired files. Generally, you will not want the server to return expired files. An exception might be if you were demonstrating the server and do not particularly care about the content being returned.

Example

```
CacheExpiryCheck Off
```

Initial configuration file setting

```
CacheExpiryCheck On
```

CacheLastModifiedFactor - Specify fraction of Last-Modified time to be used for determining expiration date

HTTP servers usually give the Last-Modified time for a file, but not the Expires date. Use this directive to have your server approximate the expiration date of these files based on the Last-Modified time. The server uses the Last-Modified date to determine how long it has been since the file was modified. The server multiplies that length of time by the value on the CacheLastModifiedFactor directive. The result of this calculation is the lifetime of the file, or how long before the file becomes stale.

Examples

```
CacheLastModifiedFactor 0.2
```

The above example would cause files modified five months ago to expire after one month.

```
CacheLastModifiedFactor Off
```

The above example would turn this function off.

Initial configuration file setting

```
CacheLastModifiedFactor 0.14
```

The default of 0.14 causes files modified in the past week to be updated in one day.

CacheLimit_2 - Specify upper limit for cached file size

Use this directive to specify the maximum size of files to be cached. Files larger than this size will not be cached. The value can be specified in bytes (B), kilobytes (K), megabytes (M), or gigabytes (G). It does not matter if you have a space between the number and the value (B, K, M, G).

Example

```
CacheLimit_2 2000 K
```

Initial configuration file setting

```
CacheLimit_2 400 K
```

CacheNoConnect - Specify stand alone cache mode

Use this directive to specify whether you want the proxy server to retrieve files from remote servers. Use the default value of `Off` if you want the server to be able to retrieve files from remote servers.

Specify `On` if you want the server to run in stand alone cache mode. This means that the server can return only files already stored in its cache. Typically, you would also set the CacheExpiryCheck directive to `Off` when running the server in this mode.

Running the server in stand alone cache mode can be useful if you are using the server for demonstrations. If you know all the files you want to use for a demonstration are stored in the cache, then you do not need a network connection.

Example

```
CacheNoConnect On
```

In the above example, the server returns only files stored in its cache.

Initial configuration file setting

```
CacheNoConnect Off
```

CacheOnly - Cache only files with URLs that match a template

Use this directive to specify that only files with URLs that match the given template should be cached. You can have multiple occurrences of this directive in the configuration file. Include a separate directive for each template. The URL template must include the protocol. If this directive is not given, any URLs not matching a `NoCache` directive is a candidate to be cached.

Example

```
CacheOnly http://realstuff/*
```

Initial configuration file setting

None.

CacheRoot - Specify cache root directory

Use this directive to specify the top directory in the cache hierarchy. The server will create subdirectories within this directory for each cached protocol. It will also create subdirectories under each protocol subdirectory for each remote server.

Example

```
CacheRoot /webcache
```

Initial configuration file setting

None.

CacheSize - Specify cache size

Use this directive to set the maximum amount of disk space you want the proxy cache to use. If you have plenty of disk space, you may want to substantially increase the 500 M default size. The size of the cache will usually stay below the maximum, but may occasionally grow slightly larger. When the maximum size is reached, no additional files will be added to the cache until the next garbage collection process begins. The value can be specified in bytes (B), kilobytes (K), megabytes (M), or gigabytes (G). It does not matter if you have a space between the number and the value (B, K, M, G).

Example

```
CacheSize 5 G
```

Initial configuration file setting

```
CacheSize 500 M
```

CacheUnused - Specify how long to keep unused cached files

Use this directive to set the maximum amount of time for the server to keep unused cached files with URLs matching a given template. The server deletes unused files with URLs matching the template after they have been cached for the specified time, regardless of their expiration date. You can have multiple occurrences of this directive in the configuration file. Include a separate directive for each template. The URL template must include the protocol. Specify the time value in any combination of months, weeks, days, and hours.

Examples

```
CacheUnused ftp:* 3 weeks
CacheUnused gopher:* 3 days 12 hours
CacheUnused * 4 weeks
```

Initial configuration file setting

None.

Caching - Turn proxy caching on/off

Use this directive to enable the caching of files. With caching turned on, the proxy server can store the files it retrieves from other servers in a local cache. The server can then respond to subsequent requests for the same files without having to retrieve them from another servers. This can improve response time.

Example

```
Caching On
```

Initial configuration file setting

```
Caching Off
```

ftp_proxy - Specify a proxy server to connect to for FTP requests

If your proxy server is part of a chain of proxies, use this directive to specify the name of another proxy that this server should contact for FTP requests. You must specify a full URL including the trailing slash.

Example

```
ftp_proxy http://outer.proxy.server/
```

Initial configuration file setting

None.

DiskBlockSize - Specify allocation unit size

Use this directive to inform Web Traffic Express of the allocation unit size of the drive specified in the `CacheRoot` directive. Web Traffic Express uses this value to round up file sizes when computing proxy utilization.

Example

```
DiskBlockSize 512
```

Initial configuration file setting

```
On OS/2 and Windows NT:  DiskBlockSize 512
On AIX:             DiskBlockSize 4096
```

Gc - Turn garbage collection on or off

If you have enabled caching, the server uses the garbage collection process to delete files that should no longer be cached. Files are deleted based on their expiration date and other proxy directive values. Use this directive to turn garbage collection on or off. Generally, you would not turn off garbage collection if you have enabled caching. If you do, your cache file could grow beyond the maximum size you set for it.

Assuming garbage collection is turned on, the garbage collection process runs when the cache reaches its maximum size (as specified on the CacheSize directive). The garbage collection process will also run at the time of day specified on the GcDailyGc directive.

Example

```
Gc Off
```

Initial configuration file setting

```
Gc On
```

GcDailyGc - Specify a daily time for garbage collection

Use this directive to specify a particular time of day to run the garbage collection process. Garbage collection occurs automatically when the cache size limit is reached. By specifying a daily time for garbage collection you can also remove cached files before the cache reaches its maximum. Specify the time value in 24:00 hour format. Generally, you would want the garbage collection process to run when your server is not being used much for other things. This is why the default is 03:00.

Example

```
GcDailyGc 22:00
```

The above example would start the garbage collection process at 10 P.M.

```
GcDailyGc Off
```

The above example would disable daily garbage collection.

Initial configuration file setting

```
GcDailyGc 03:00
```

GcMemUsage - Specify how much memory to use for garbage collection

Garbage collection works best if it can read all cache information into memory at one time. It may not be able to read in the entire cache if your system does not have enough main memory.

Use this directive to specify how much memory garbage collection can use. The value you specify should be approximately the amount of virtual memory that the server may use while performing garbage collection. The amount of memory needed will vary based on dynamic changes such as the directory structure of cached files. Specify the value as a number that represents kilobytes, but do not put a K next to the number.

If garbage collection fails because there is not enough memory on your system, set this directive to a smaller value. If you have plenty of memory to spare, you may want to set this value above the default of 500.

Example

```
GcMemUsage 100
```

The example above might be used for a machine with a small amount of memory.

Initial configuration file setting

```
GcMemUsage 1000
```

gopher_proxy - Specify a proxy server to connect to for Gopher requests

If your proxy server is part of a chain of proxies, use this directive to specify the name of another proxy that this server should contact for Gopher requests. You must specify a full URL including the trailing slash.

Example

```
gopher_proxy gopher://outer.proxy.server/
```

Initial configuration file setting

None.

http_proxy - Specify a proxy server to connect to for HTTP requests

If your proxy server is part of a chain of proxies, use this directive to specify the name of another proxy that this server should contact for HTTP requests. You must specify a full URL including the trailing slash.

Example

```
http_proxy http://outer.proxy.server/
```

Initial configuration file setting

None.

MaxContentLengthBuffer - Set the size of the buffer for dynamic data

Use this directive to set the size of the buffer for dynamic data generated by the server. Dynamic data is output from CGI programs, server-side includes, and API programs. This buffering is not done for proxy requests.

The value can be specified in bytes (B), kilobytes (K), megabytes (M), or gigabytes (G). It does not matter if you have a space between the number and the value (B, K, M, G).

Example

```
MaxContentLengthBuffer 2 M
```

Initial configuration file setting

On AIX:

```
MaxContentLengthBuffer 100 K
```

On Windows NT and OS/2:

```
MaxContentLengthBuffer 50 K
```

no_proxy - Connect directly to domains matching templates

If you are using proxy chaining by using `http_proxy`, `ftp_proxy`, or `gopher_proxy` directives, you can specify the domains that you want the server to directly connect to rather than going through a proxy.

Specify the value as a string of domain names or domain name templates. Separate each entry in the string with a comma. Do **not** put any spaces in the string.

You specify templates on this directive a bit differently than the way you specify templates on other directives. Most importantly, you **cannot** use the wildcard character (*). What you **can** do is specify a template by including only the last part of a domain name. The server connects directly to any domains that end with a string matching the templates you specify. This directive only applies to proxy chaining and is equivalent to a direct @= line in the SOCKS configuration file. The following example shows how this works:

Example

```
no_proxy   www.someco.com,.raleigh.ibm.com,.some.host.org:8080
```

In the above example the server would not go through a proxy for the following requests:

- Any requests to domains ending with www.someco.com,
- Any requests to domains ending with .raleigh.ibm.com, such as blugrass.raleigh.ibm.com or keystone.raleigh.ibm.com
- Any requests to port 8080 of domains ending with .some.host.org, such as myname.some.host.org:8080. (This would not include requests to any other ports of the same domain, such as myname.some.host.org, which assumes the default port 80.)

Initial configuration file setting

None.

NoCaching - Do not cache files with URLs that match a template

Use this directive to specify that the server should not cache files with URLs matching the given template. You can have multiple occurrences of this directive in the configuration file. Include a separate directive for each template. The URL template must include the protocol.

Example

```
NoCaching http://joke/*
```

Initial configuration file setting

None.

ProxyAccessLog — Name the path for the proxy access log file

Use this directive to specify the path and file name where you want the server to log access statistics that pertain to proxy requests. By default, the server writes an entry to this log each time it acts as a proxy for a client request. You can use the NoLog directive if you do not want to log requests from certain clients.

The server starts a new log file each day at midnight if it is running. Otherwise, the server starts a new log file the first time you start it on a given day. When creating the file, the server uses the file name you specify and appends a date suffix or extension. The date suffix or extension is in the format *Mmmddyyyy*, where *Mmm* is the first three letters of the month; *dd* is the day of the month; and *yyyy* is the year.

It is a good idea to remove old log files, because they can take up a significant amount of space on your hard drive.

Example

```
ProxyAccessLog  c:\server\logs\proxylog

ProxyAccessLog  /logs/proxylog
```

Initial configuration file setting

```
AIX:                      ProxyAccessLog /usr/lpp/internet/server_root/logs/proxy-log

Windows NT and OS/2:      ProxyAccessLog  c:\path\proxy-log
```

*c:**path*\ is the value you entered for Logs directory at installation. The installation default on Windows NT, and OS/2 is c:\WWW\LOGS

On AIX, the example gives you *ServerRoot*/logs/proxy-log *datesuffix.file_extension*. In other words, if *ServerRoot* is /usr/lpp/internet/server_root/ and the server is started on January 15, 1996 the result is:

```
/usr/lpp/internet/server_root/logs/proxy-log.Jan151996
```

Web Traffic Express directives

These directives are located in the javelin configuration file in the ETC directory.

On AIX	**javelin.conf**
On Windows NT and OS/2:	**javelin.cnf**

AutoCacheRefresh - Turn cache refreshing on and off

Use this directive to turn cache refreshing on or off. By default, the cache agent runs at 12 midnight

Example

```
AutoCacheRefresh Off
```

Initial configuration file setting

```
AutoCacheRefresh On
```

CacheFiles - Specify number of files to store in cache

Use this directive to specify the maximum number of files stored in the cache. See the `CacheSize` directive to limit the size of the cache in bytes. A value of `0` means the number of files are not limited.

Example

```
CacheFiles 10000
```

Initial configuration file setting

```
CacheFiles 0
```

CacheLocalDomain - Specify whether to cache local domain

Use this directive to specify whether or not to cache local domain sites. When running an intranet, local sites usually don't have to be cached because the internal bandwidth is sufficient to quickly load URLs. Not caching local sites saves cache space for heavier requests.

Example

```
CacheLocalDomain off
```

Initial configuration file setting

```
CacheLocalDomain on
```

CacheMinHold - Specify how long to keep files available

Use this directive to specify URLs to overrride the expires tag. Some sites set documents to expire even when they have a longer lifetime, requring the server to request the document more frequently. This directive holds this expired document for

the specified amount of time before requesting it again. You can specify this directive multiple times. **Attention:** If expiration dates are overridden, the files in the cache may become obsolete or out of date.

Example

```
CacheMinHold http://www.cachebusters.com/* 1 hour
```

Initial configuration file setting

There is no default value

DefinePicsRule - Supply a content filtering rule

Supply the proxy with the necessary information to filter URLs for content including rating service information. You can specify this directive multiple times. See “Chapter 5. Configuring PICS-based filtering” on page 39 for information on PICS filters

DelayPeriod - Turn off pausing between requests

Use this directive to specify whether the cache agent should wait between sending requests to destination servers. Turning on a delay between requests will reduce the load on the proxy machine and your network link, as well as being kinder to the destination servers. Turning the delay off will let the cache agent run at maximum speed.

Attention: if you have a fast connection to the Internet (>56K), set `DelayPeriod On` to avoid rapid-fire attacks on sites being refreshed.

Example

```
DelayPeriod On
```

Initial configuration file setting

```
DelayPeriod Off
```

DelveAcrossHosts - Turn on caching across domains

Use this directive to specify whether the cache agent will follow hypertext links across hosts. If a cached URL contains links to other servers, the server can ignore the link or follow it. This directive is only applicable when `DelveInto always` is specified.

Example

```
DelveAcrossHosts On
```

Initial configuration file setting

```
DelveAcrossHosts Off
```

DelveDepth - Specify how far to follow links while caching

Use this directive to specify the number of link levels to follow when searching for pages to load into cache. This directive is applicable when ″DelveInto always″ is specified.

Example

```
DelveDepth 4
```

Initial configuration file setting

```
DelveDepth 2
```

DelveInto - Tell the cache agent to follow links

Use this directive to specify if the cache agent should load pages linked off of cached URLs, from the server access log, from the configuration file, or not at all. Possible values for this directive are:

always
cache agent caches all linked pages off previously cached URLs

never
cache agent ignores all links on URL

admin
cache agent only follows links on URLs specified in the `LoadURL` directives

log
cache agent only follow links from URLs in the server access log

Example

```
DelveInto admin
```

Initial configuration file setting

```
DelveInto always
```

FlexibleSocks - Enable flexible SOCKSification

Use this directive to instruct the proxy to use the SOCKS configuration file to determine the type of connection to make.

Example

```
FlexibleSocks off
```

Initial configuration file setting

```
FlexibleSocks on
```

GCMaxInUse - Specify limit of used cache

Use this directive to specify the maximum percentage of the total cache size to be used after garbage collection removes expired files.

Example

```
GCMaxInUse 50
```

Initial configuration file setting

```
GCMaxInUse 75
```

IgnoreURL - Specify URLs not to cache

Use this directive to specify URLs that are not to be loaded. You can have multiple occurences of `IgnoreURL` specifying different URLs or URL masks. This directive is useful when the cache agent is loading linked pages off of cached URLs. The value for this directive may contain wildcards (*). This directive applies to the cache agent.

Examples

```
IgnoreURL http://www.yahoo.com/
IgnoreURL http://*.ibm.com/*
```

Initial configuration file setting

```
IgnoreURL */cgi-bin/*
```

LoadInlineImages - Turn off caching of images

Use this directive to specify whether inline images should be retrieved by the cache agent.

Example

```
LoadInlineImages never
```

Initial configuration file setting

```
LoadInlineImages always
```

LoadTopCached - Specify number of popular pages to refresh

Use this directive to instruct the cache agent to access the previous night's cache access log and load the most requested URLs.

The `Caching` directive must be `On` and the `CacheAccessLog` directive must have a valid value when using `LoadTopCached`.

Example

```
LoadTopCached 30
```

Initial configuration file setting

no default setting

LoadURL - Specify URLs to cache

Use this directive to specify URLs to be loaded into cache. Note that these URLs are placed at the top of the cache agent's queue, followed by most URLs from the previous night's cache access log from the `LoadTopCached` directive. `LoadURL` can have several occurences and may contain wildcards.

Example

```
LoadURL http://www.ibm.com/
LoadURL http://*.netscape.com/
```

Initial configuration file setting

no default setting

MaxQueueDepth - Specify maximum number URLs to queue

Use this directive to specify the maximum depth of the cache agent's queue of outstanding page retrieval requests. If you have a large system with lots of memory, you may define a larger queue of page retrieval requests without running out of memory.

The queue of URLs to cache is determined at the beginning of each cache agent run. If you instruct the cache agent to follow the hypertext links to other URLs, these other URLs are not in the cache queue. Once the value specified in the `MaxURLs` directive is reached, the cache agent stops, even if there are more URLs in the queue.

Example

```
MaxQueueDepth 500
```

Initial configuration file setting

```
MaxQueueDepth 150
```

MaxRunTime - Specify maximum time for a cache agent run

Use this directive to specify the maximum time the cache agent will retrieve URLs during a particular run. A value of `0` means the cache agent runs until completion.

Example

```
MaxRunTIme 2 hours 10 minutes
```

Initial configuration file setting

```
MaxRunTime 2 hours
```

MaxUrls - Specify maximum number of URLs

Use this directive to specify the maximum number of URLs the cache agent will retrieve during a particular run. A value of `0` means there is no limit. When using the automatic mode of the cache agent, the `LoadURL` and `LoadTopCached` directives take precedence. See “Following links and delving” on page 34 for more information.

Example

```
MaxURLs 1000
```

Initial configuration file setting

```
MaxURLs 200
```

NoProxyHeader - Specify client headers to block

Use this directive to specify client URL headers to block. Headers contain fields such as:

- **Last-Modified:** provides information on when the document was last modified
- **Expires:** provides expiry information on the document
- **Pragma:** usually used to instruct browsers and servers with caches to fetch the document from the original server everytime the file is requested.
- **Referer:** URL of the document that the Request-URI was obtained.
- **Server:** information about the software used by the server handling the request.

See the HTTP protocol specification for details of these and other headers. You can specify this directive multiple times.

Note: Any HTTP header can be blocked, including required headers. Use extreme care when blocking headers.

Example

```
NoProxyHeader Referer:
```

Initial configuration file setting

There is no default value

NumClients - Specify number of threads to use

Use this directive to specify the number of threads the cache agent uses to retrieve pages in the queue.

Tune `NumClients` based on the speed of your internal network and your connection to the Internet. The allowable range is 1 through 100.

Example

```
NumClients 50
```

Initial configuration file setting

```
NumClients 4
```

Proxy - Specify cache destination

Use this directive to specify which proxy server the cache agent should update. This is required when the cache agent needs to update a proxy server other than the local proxy server it is running on. Optionally, you can specify the port.

Attention: On AIX, this directive is required for using the cache agent. If you are only using one machine for the proxy, specify the hostname.

Example

```
Proxy proxy15.ibm.com:1080
```

Initial configuration file setting

defaults to the host on which the cache agent is running

ProxyFrom - Specify client ″From:″ header

Use this directive to generate a ″**From:**″ header. This is typically used to give an email address of the proxy administrator

Example

```
ProxyFrom webmaster@proxy.ibm.com
```

would result in the following header change:

Original header	Modified 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
Pragma: no-cache	**From: webmaster@proxy.ibm.com**
	Pragma: no-cache

Initial configuration file setting

There is no default value

ProxyIgnoreNoCache - Ignore reload request

Use this directive to specify how the server reacts when a user clicks the Reload button on their browser. If `ProxyIgnoreNoCache` is `On`, during periods of high load, the server will not request the page from the destination server, and will supply the cached copy of the document, if available. The server essentially disregards the ″Pragma: no-cache″ header sent from the browser.

Example

```
ProxyIgnoreNoCache on
```

Initial configuration file setting

```
ProxyIgnoreNoCache off
```

ProxyNumTables - Specify number of subcaches

Use this directive to specify the number of sub-caches the server utilizes in the automatic cache function. The value range is between 1 and 150; prime numbers are recommended.

Example

```
ProxyNumTables 10
```

Initial configuration file setting

```
ProxyNumTables 20
```

ProxyPersistence - Allow persistent connections

Specifies whether or not a persistent connection is maintained with the destination server. A persistent connection reduces latency for users and reduces the load on the proxy server while requiring more resources. More threads, and therefore more memory on the proxy server, are required for a persistent connection.

Persistent connections must not be used on a multi-level proxy server setup if any of the proxies are not HTTP/1.1 compliant.

Example

```
ProxyPersistence off
```

Initial configuration file setting

```
ProxyPersistence on
```

ProxySendClientAddress - Specify the ″Client IP Address:″ header

Use this directive to allow the proxy to forward the IP address of the client to the destination server. Values for this directive are OFF, or Client-IP:.

Example

```
ProxySendClientAddress Client-IP:
```

would result in the following header change:

Original header	Modified 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
Pragma: no-cache	**Client-IP: 9.67.199.5**
	Pragma: no-cache

Initial configuration file setting

There is no default value

ProxyUserAgent - Specify ″User Agent″ string

Use this directive to specify a different user agent string to replace the one the client sends. This allows greater anonymity while visiting Web sites. However, some sites have customized pages based on the user agent string. Using ProxyUserAgent will not allow these custom pages to be displayed.

Example

```
ProxyUserAgent WebTrafficExpress/1.0
```

would result in the following header change:

Original header	Modified 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: WebTrafficExpress/1.0**
Pragma: no-cache	Pragma: no-cache

Initial configuration file setting

There is no default value

PureProxy - Disable a dedicated proxy

Use this directive to specify whether the server is acting as a proxy, or as a proxy and content server. We recommend that you use Web Traffic Express as a proxy only.

Example

```
PureProxy off
```

Initial configuration file setting

```
PureProxy on
```

Appendix B. Notices

References in this publication to IBM products, programs, or services do not imply that IBM intends to make them available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program or service may be used. Subject to IBM's valid intellectual property or other legally protectable rights, any functionally equivalent product, program, or service may be used instead of the IBM product, program, or service. The evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, are the responsibility of the user.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, 500 Columbus Avenue, Thornwood, NY 10594, U.S.A.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

Site Counsel
IBM Corporation
P.O. Box 12195
3039 Cornwallis
Research Triangle Park, NC 27709-2195
USA

Site Counsel
IBM Corporation
P.O. Box 12195
3039 Cornwallis Road
Research Triangle Park, NC 27709-2195
USA

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement.

This document is not intended for production use and is furnished as is without any warranty of any kind, and all warranties are hereby disclaimed including the warranties of merchantability and fitness for a particular purpose.

This product includes computer software created and made available by CERN. This acknowledgement shall be mentioned in full in any product which includes the CERN computer software included herein or parts thereof.

Trademarks

The following terms are trademarks of IBM Corporation in the United States or other countries or both.

AIX

IBM

InfoExplorer

OS/2

Power Series

PS/2

RISC System/6000

SystemView

TrackPoint

Microsoft, Windows, Windows NT and the Windows 95 logo are trademarks or registered trademarks of Microsoft Corporation.

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

Other company, product, and service names, which may be denoted by a double asterisk (**), may be trademarks or service marks of others.

Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.