websense

Online Help

Websense Content Gateway

v7.0.x

Websense Content Gateway Online Help

2009

Contents

1 | Overview

Websense® Content Gateway is a high-performance Web proxy cache that improves network efficiency and performance by caching frequently accessed information at the edge of the network. This brings content physically closer to end users for faster delivery and reduces bandwidth usage.

Websense Content Gateway can be deployed:

◆ *As a Web proxy cache*

◆ *In a cache hierarchy*

◆ *In a cluster*

◆ *As an SSL server*

Related topics:

Release Notes for this version of the product
Deployment options, page 1
Components, page 3
Traffic analysis options, page 5
Security options, page 6
Online Help, page 7
Technical Support, page 7

Deployment options

As a Web proxy cache

When Websense Content Gateway is deployed as a Web proxy, user requests for Web content pass through Websense Content Gateway on the way to the destined Web server (origin server). If the cache contains the requested content, it serves the content directly. If the cache does not have the requested content, Websense Content Gateway acts as a proxy, fetching the content from the origin server on the user's behalf, while keeping a copy to satisfy future requests.

Websense Content Gateway provides the following proxy caching options:

◆ *Explicit proxy caching,* where the user's client software must be configured to send requests directly to Websense Content Gateway. See *Explicit Proxy Caching*, page 35.

◆ *Transparent proxy caching*, where user requests are automatically injected into a Websense Content Gateway cache on their way to the eventual destination. Users request Internet content as usual, without any browser configuration, and the proxy serves their requests. The user's client software (typically a browser) is unaware that it is communicating with a proxy. See *Transparent Proxy Caching and ARM*, page 41.

In a cache hierarchy

Websense Content Gateway can participate in flexible cache hierarchies, where Internet requests not fulfilled in one cache can be routed to other regional caches, taking advantage of their contents and proximity. In a hierarchy of proxy servers, Websense Content Gateway can act either as a parent or child cache, either to other Websense Content Gateway systems or to other caching products.

Websense Content Gateway can be a member of an HTTP cache hierarchy and an ICP (Internet Cache Protocol) cache hierarchy. See *Hierarchical Caching*, page 69.

In a cluster

Websense Content Gateway scales from a single node into multiple nodes that form a cluster, improving system performance and reliability. The proxy detects the addition or removal of nodes. If Websense Content Gateway's virtual IP failover option is enabled, Websense Content Gateway maintains a pool of virtual IP addresses that it assigns to the nodes of the cluster. Websense Content Gateway can detect hard node failures (such as power supply or CPU failures) and reassign IP addresses of the failed node to the operational nodes.

Websense Content Gateway has the following clustering modes:

◆ *Management-only mode*, where you can administer all the nodes in a cluster at the same time. Nodes automatically share configuration information.

◆ *Full-clustering mode*, where the node caches act as a single aggregate cache. A Websense Content Gateway cluster distributes its cache across its nodes into a single, virtual object store, rather than replicating the cache node by node.

A fully clustered Websense Content Gateway system provides a single system image to both users and administrators, appearing as a single virtual server. Full-clustering mode includes management-only mode.

See *Clusters*, page 61.

As an SSL server

If your subscription includes SSL Manager, data can be decrypted and then re-encrypted as it travels from the client to the origin server. SSL Manager includes a

complete set of certificate-handling capabilities. See *Working With Encrypted Data*, page 121.

Components

Cache

The *cache* consists of a high-speed object database called the object store. The object store indexes objects according to URLs and associated headers. The object store can cache alternate versions of the same object, varying on spoken language or encoding type, and can store small and large documents, minimizing wasted space. When the cache is full, the proxy removes stale data, ensuring that frequently requested objects are fresh.

Websense Content Gateway tolerates disk failure on any cache disk. If the disk fails completely, Websense Content Gateway marks the disk as corrupt and continues using the remaining disks. If all cache disks fail, Websense Content Gateway goes into proxy-only mode.

You can partition the cache to reserve disk space for storing data for specific protocols and origin servers. See *Configuring the Cache*, page 75.

RAM cache

Websense Content Gateway maintains a small RAM memory cache of extremely popular objects. This *RAM cache* serves the most popular objects quickly and reduces load on disks, especially during traffic peaks. You can configure the RAM cache size. See *Changing the size of the RAM cache*, page 78.

Adaptive Redirection Module

The Adaptive Redirection Module (ARM) is used in transparent proxy caching to redirect user requests destined for an origin server to the proxy. Before the traffic is redirected by the ARM, it is intercepted by a Layer 4 switch or router.

To redirect user requests to the proxy, the ARM changes an incoming packet's address. The packet's destination IP address is changed to the IP address of the proxy, and the packet's destination port is changed according to the protocol used. For example, for HTTP, the packet's destination port is changed to the proxy's HTTP port (usually 8080).

The ARM supports automatic bypass of sites that do not function properly with proxy caches.

The ARM also prevents client request overloads. When there are more client connections than the specified limit, the ARM forwards incoming requests directly to the origin server. See *Connection load shedding*, page 58.

Host database

The host database stores the Domain Name Server (DNS) entries of origin servers to which the proxy connects. Among other information, the host database tracks:

◆ DNS information (for fast conversion of host names to IP addresses)

◆ The HTTP version of each host (so advanced protocol features can be used with hosts running modern servers)

◆ Host reliability and availability information (to avoid waits for non-functional servers)

DNS resolver

The proxy includes an asynchronous DNS resolver to streamline conversion of host names to IP addresses. Websense Content Gateway implements the DNS resolver natively, directly issuing DNS command packets, rather than relying on resolver libraries. Many DNS queries can be issued in parallel and a fast DNS cache maintains popular bindings in memory, reducing DNS traffic.

Processes

The `content_gateway` process accepts connections, processes protocol requests, and serves documents from the cache or origin server.

The `content_manager` process launches, monitors, and reconfigures the `content_gateway` process. The `content_manager` process is also responsible for the Websense Content Manager user interface, the proxy auto-configuration port, the statistics interface, cluster administration, and virtual IP failover. If the `content_manager` process detects a `content_gateway` process failure, it restarts the process and also maintains a connection queue of all incoming requests. Incoming connections that arrive in the several seconds before server restart are saved in the connection queue and processed in sequence. This connection queueing shields users from server restart downtime.

The `content_cop` process monitors the health of `content_gateway` and `content_manager`. The `content_cop` process periodically (several times each minute) queries `content_gateway` and `content_manager` by issuing heartbeat requests to fetch synthetic Web pages. If no response is received within a timeout interval or if an incorrect response is received, `content_cop` restarts `content_manager` and `content_gateway`.

Administration tools

Websense Content Gateway offers administration alternatives:

◆ *Websense Content Manager* is a Web-based interface accessible through a browser. Websense Content Manager provides graphs and statistical displays for monitoring Websense Content Gateway performance and network traffic, and options for configuring and fine-tuning the proxy. Websense Content Manager

offers password-protected, SSL-encrypted, single-point administration for an entire Websense Content Gateway cluster.

◆ A *command-line interface* enables you to monitor Websense Content Gateway performance and network traffic, and configure the proxy. You can execute individual commands or script a series of commands in a shell.

◆ *Configuration files* allow complete administration through a file-editing and signal-handling interface. You can change configuration options by editing configuration files instead of using Websense Content Manager or the command-line interface. Any changes you make through Websense Content Manager or the command-line interface are automatically made to the configuration files.

Related topics:

Websense Content Manager, page 91
Command-line interface, page 94
Configuration files, page 95

Traffic analysis options

Websense Content Gateway provides options for network traffic analysis and monitoring:

◆ *Manager statistics and graphs* show network traffic information. View graphs and statistics from Websense Content Manager, or collect and process statistics using the command-line interface.

◆ *MRTG* (Multi Router Traffic Grapher) provides a variety of graphs showing historical information about virtual memory usage, client connections, document hit rates, and so on. Access MRTG from Websense Content Manager.

◆ *SNMP Network Management* support lets you monitor and manage Websense Content Gateway through SNMP network management facilities. Websense Content Gateway supports two management information bases (MIBs): MIB-2, a standard MIB, and the proprietary Websense Content Gateway MIB that provides more specific node and cluster information.

◆ *Manager alarms* are presented in Websense Content Manager. Websense Content Gateway signals an alarm for any detected failure condition. You can configure Websense Content Gateway to send email or page support personnel when an alarm occurs.

◆ *Transaction logging* lets you record information in a log file about every request the proxy cache receives and every error it detects. Use the logs to determine how many people use the proxy cache, how much information each person requested, and which pages are most popular. You can see why a transaction was in error and see the state of the proxy cache at a particular time. For example, you can see that Websense Content Gateway was restarted or that cluster communication timed out.

Websense Content Gateway supports several standard log file formats, such as Squid and Netscape, and its own custom format. You can analyze the standard

format log files with off-the-shelf analysis packages. To help with log file analysis, separate log files so that they contain information specific to protocol or hosts.

For traffic analysis options, see *Monitoring Traffic*, page 79. For logging options, see *Working With Log Files*, page 149.

Security options

Websense Content Gateway combined with Websense Enterprise or Websense Web Security Suite provides protection and filtering along with Web proxy technology. Not only can you provide a proxy and caching to improve responsiveness, you can also provide an increased level of security for Internet usage.

In addition, Websense Content Gateway enables you to establish secure communication between the Websense Content Gateway system and other computers on the network. You can:

◆ Control client access to the proxy cache.

◆ Control which hosts are allowed to access the proxy cache machine (ARM security).

◆ Configure Websense Content Gateway integration into your firewall and control traffic through a SOCKS server.

◆ Configure Websense Content Gateway to use different DNS servers, depending on whether it needs to resolve host names located inside or outside a firewall. This enables you to keep your internal network configuration secure while providing transparent access to external sites on the Internet.

◆ Configure the proxy to ensure that clients are authenticated before they can access content from the cache. Websense Content Gateway supports LDAP, RADIUS, and NTLM proxy authentication in explicit mode only.

◆ Control access to Websense Content Manager using:

   ▪ SSL (Secure Sockets Layer) protection for encrypted, authenticated access

   ▪ An access control list (ACL) that defines which hosts are allowed to access Websense Content Manager

   ▪ User accounts that define which users can access Websense Content Manager and which activities they can perform (for example, view statistics only or view statistics and configure Websense Content Gateway).

See *Security*, page 101.

In addition, Websense Content Gateway supports features of ICAP that enable sites using Websense Data Security Suite to examine outbound material such as Web postings, and either block or allow the posting based on company policy. See *Working With Websense Data Security Suite*, page 99.

Online Help

Select **Get Help** within the program to display detailed information about using the product.

> ⚠ **Important**
>
> Default Microsoft Internet Explorer settings may block operation of the Help system. If a security alert appears, select **Allow Blocked Content** to display Help.
>
> If your organization's security standards permit, you can permanently disable the warning message on the Advanced tab of the **Tools** > **Internet Options** interface. (Check **Allow active content to run in files on My Computer** under Security options.)

To see a PDF version of the online help, click the icon in the upper right corner of the Help screen.

Technical Support

Technical information about Websense products is available online 24 hours a day, including:

- ◆ latest release information
- ◆ searchable Websense Knowledge Base
- ◆ show-me tutorials
- ◆ product documents
- ◆ tips
- ◆ in-depth technical papers

Access support on the Web site at:

www.websense.com/SupportPortal/

For additional questions, fill out the online support form at:

www.websense.com/SupportPortal/Contact.aspx

If your issue is urgent, please call one of the offices listed below. You will be routed to the first available technician, who will gladly assist you.

Location	Contact information
North America	+1 858-458-2940
France	Contact your Websense Reseller. If you cannot locate your Reseller: +33 (0) 1 57 32 32 27
Germany	Contact your Websense Reseller. If you cannot locate your Reseller: +49 (0) 69 51 70 93 47
UK	Contact your Websense Reseller. If you cannot locate your Reseller: +44 (0) 20 3024 4401
Rest of Europe	Contact your Websense Reseller. If you cannot locate your Reseller: +44 (0) 20 3024 4401
Middle East	Contact your Websense Reseller. If you cannot locate your Reseller: +44 (0) 20 3024 4401
Africa	Contact your Websense Reseller. If you cannot locate your Reseller: +44 (0) 20 3024 4401
Australia/NZ	Contact your Websense Reseller. If you cannot locate your Reseller: +61 (0) 2 9414 0033
Asia	Contact your Websense Reseller. If you cannot locate your Reseller: +86 (10) 5884-4200
Latin America and Caribbean	Contact your Websense Reseller.

For telephone requests, please have ready:

◆ Websense subscription key

◆ Access to Websense Manager

◆ Access to the machine running Filtering Service, the machine running reporting tools, and the database server (Microsoft SQL Server or MSDE)

◆ Access to the machine running Websense Content Gateway

◆ Permission to access the Websense Log Database

◆ Familiarity with your network's architecture, or access to a specialist

◆ Specifications of machines running Filtering Service and Websense Manager

◆ A list of other applications running on the Filtering Service machine

2 | Getting Started

After you have installed Websense Content Gateway on your system or on the nodes of your Websense Content Gateway cluster, the proxy cache is ready for use.

Refer to the following procedures to get started:

Starting and stopping Websense Content Gateway

Websense Content Gateway starts automatically when installation is completed. To stop or start Websense Content Gateway from the command line:

1. Become root:

       su

2. Change to the Websense Content Gateway installation directory. The default location is **/opt/WCG**.

3. To start the proxy:

   `./WCGAdmin start`

   To stop the proxy

   `./WCGAdmin stop`

   To restart the proxy

   `./WCGAdmin restart`

   To see what services you have running:

   `./WCGAdmin status`

   > ✔ **Note**
   > Always use this procedure to stop Websense Content
   > Gateway.

After you've installed Websense Content Gateway, open Websense Content Manager (the Websense Content Gateway management interface) to ensure that the proxy is running. See *Starting Websense Content Manager*, page 10 and *Verifying that the proxy is running*, page 12.

Starting Websense Content Manager

Websense Content Manager is the management interface to Websense Content Gateway. Websense Content Manager is supported on Microsoft Internet Explorer v7.0 and later and on Firefox v2.0 and later.

Java and JavaScript must be enabled in your browser. See your browser documentation for information on enabling Java and JavaScript.

To access Websense Content Manager:

1. Open your Web browser.

2. Enter the following location in your browser:

   `https://nodename:adminport`

   where `nodename` is the name of the proxy node and `adminport` is the number assigned to the Websense Content Manager port (the default value for `adminport` is 8081). For more information on using HTTPS to start Websense Content Manager, see *Using SSL for secure administration*, page 105.

3. Log on to Websense Content Manager with the administrator ID (the default is admin) and password, or your user account.

   The Websense Content Gateway password is set during installation. You can change the ID and password, as well as create and modify user accounts. See *Controlling access to Websense Content Manager*, page 103.

   Figure 1 shows the Websense Content Manager interface.

Figure 1 Websense Content Manager

Websense Content Manager opens to the **Monitor > My Proxy > Summary** page. This page provides information on the features of your subscription and details of your Websense Content Gateway system. See *Viewing statistics*, page 79, for additional information on the Monitor tab and *Configuring the System*, page 91 for information on the configuration options in Websense Content Manager.

If you receive a message indicating that Websense Content Manager is finding an unrecognized or invalid certificate, install the **content_gateway_ca.cer** certificate provided by Websense. It is located in the **/home/Websense** directory.

Use the **Configure > SSL > Certificates > Add Root CA** page to import this certificate.

1. Browse to **/home/Websense**, and select **content_gateway_ca.cer**.
2. Click **Open**.
3. Click **Add Certificate Authority**.

Entering your subscription key

4. On the **Configure > My Proxy > Subscription > Subscription Management** tab, enter the subscription key that Websense provided to you.
5. Click **Apply**.
6. Click **Restart** on **Configure > My Proxy > Basic > General** page.

Related topic:

Providing system information, page 11

Providing system information

If your subscription includes the real-time scanning features, you must provide information to facilitate communicating with Websense filtering software.

1. Select the **Configure > My Proxy > Subscription > Real-Time Scanning** tab. Notice the IP address and port for the filtering service. This is the information that you entered when you installed the filtering product.

    > ✔ **Note**
    > The Real-Time Scanning tab appears only if you have subscribed to Websense Security Gateway. It can take several minutes after you enter your subscription key before this tab appears in the Websense Web Filter user interface.
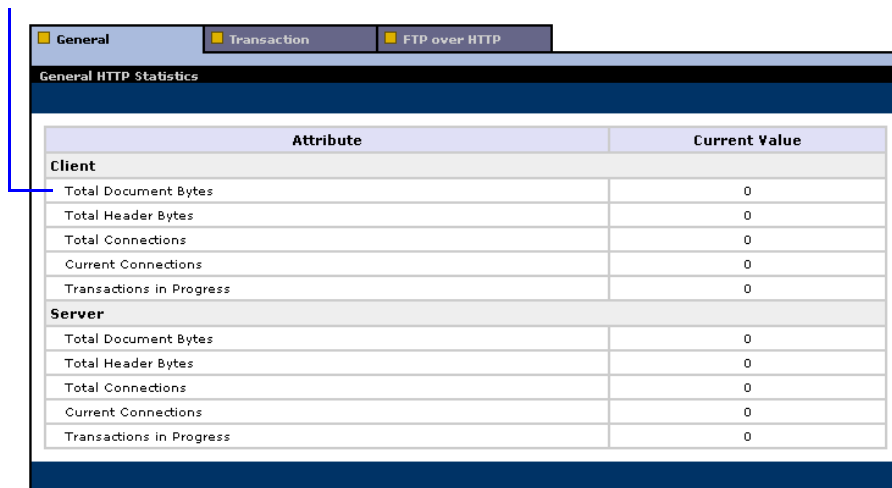
2. Select the appropriate check box to permit or block traffic if communication with Policy Server or Filtering Service fails.

3. Click **Apply**.

Verifying that the proxy is running

After you have installed the proxy, verify that it is processing requests for Web content.

1. Navigate to **Monitor > Protocol > HTTP > General** to display the General HTTP Statistics table (Figure 2).

2. Note the current **Total Document Bytes** statistic in the **Client** section of the table.

Check the value of this statistic.

Figure 2 General HTTP Statistics Table

3. Set your browser to the proxy port.

4. Browse the Internet.

5. Recheck the **Total Document Bytes** statistic.

   This value increases as the proxy processes HTTP requests.

Using the command-line interface

The command-line interface provides a quick way to view proxy statistics and configure the Websense Content Gateway system if you do not have access to a browser or if you prefer to use a UNIX shell-like command interface.

You can execute individual commands or script multiple commands in a shell. See *Websense Content Gateway commands*, page 191.

1. Become the Websense Content Gateway user called "websense":

```
su websense
```

2. Change to the Websense Content Gateway **bin** directory (default location is in /**opt/WCG**). Run Websense Content Gateway commands from this directory.

   Commands take the following form:

   ```
   content_line -command argument
   ```

3. For a list of content_line commands, enter:

   ```
   content_line -h
   ```

   > ✔ **Note**
   >
   > If the Websense Content Gateway **bin** directory is not in your path, prepend the command with ./ (for example,
   >
   > ```
   > ./content_line -h).
   > ```

3 | Web Proxy Caching

Web proxy caching stores copies of frequently accessed Web objects (such as documents, images, and articles) close to users and serves this information to them. Internet users get their information faster, and Internet bandwidth is freed for other tasks.

Internet users direct their requests to Web servers all over the Internet. For a caching server to serve these requests, it must act as a Web proxy server. A Web proxy server receives user requests for Web objects and either serves the requests or forwards them to the *origin server* (the Web server that contains the original copy of the requested information).

Websense Content Gateway proxy supports both *transparent proxy caching*, where the user's client software (typically a browser) is unaware that it is communicating with a proxy, and *explicit proxy caching*, where the user's client software must be configured to send requests directly to the proxy.

Cache requests

The following overview illustrates how Websense Content Gateway serves a user request.

1. Websense Content Gateway receives a user request for a document, image, news article, or other Web object.
2. Using the object address, the proxy tries to locate the requested object in its object database (cache).
3. If the object is in the cache, the proxy checks to see if the object is fresh enough to serve (see *Ensuring cached object freshness*, page 16). If the object is fresh, the proxy serves it to the user as a *cache hit*.
4. If the data in the cache is stale, the proxy connects to the origin server and asks if the object is still fresh (a revalidation). If the object is still fresh, the proxy sends the cached copy to the user immediately.

5. If the object is not in the cache (a cache miss) or the server indicates that the cached copy is no longer valid, the proxy obtains the object from the origin server, simultaneously streaming it to the user and the cache. Subsequent requests for the object will be served faster because the object will come directly from the cache.

Related topics:

Ensuring cached object freshness, page 16
Scheduling updates to local cache content, page 21
Pinning content in the cache, page 23
To cache or not to cache?, page 24
Caching HTTP objects, page 24
Forcing object caching, page 29
Caching HTTP alternates, page 30
Caching FTP objects, page 31

Ensuring cached object freshness

When Websense Content Gateway receives a request for a Web object, it tries to locate the requested object in its cache. If the object is in the cache, the proxy checks to see if the object is fresh enough to serve.

The protocol determines how the proxy handles object freshness in the cache:

◆ HTTP objects support author-specified expiration dates. The proxy adheres to these expiration dates; otherwise, it picks an expiration date based on how frequently the object is changing and on administrator-chosen freshness guidelines. In addition, objects can be revalidated, checking with the origin server if an object is still fresh. See *HTTP object freshness*, page 16.

◆ FTP objects stay in the cache for a specified time period. See *FTP object freshness*, page 20.

HTTP object freshness

Websense Content Gateway determines whether an HTTP object in the cache is fresh by:

◆ Checking the **Expires** or **max-age** header

Some HTTP objects contain **Expires** headers or **max-age** headers that define how long the object can be cached. Comparing the current time with the expiration time tells the proxy whether or not the object is fresh.

◆ Checking the **Last-Modified / Date** headers

If an HTTP object has no **Expires** header or **max-age** header, the proxy can calculate a freshness limit using the following formula:

```
freshness_limit =(date - last_modified) * 0.10
```

where *date* is the date in the object's server response header, and *last_ modified* is the date in the **Last-Modified** header. If there is no **Last-Modified** header, the proxy uses the date that the object was written to cache. You can increase or reduce the value 0.10 (10 percent). See *Modifying the aging factor for freshness computations*, page 17.

The computed freshness limit is bound by minimum and maximum boundaries. See *Setting an absolute freshness limit*, page 17.

◆ Checking the absolute freshness limit

For HTTP objects that do not have **Expires** headers or do not have both **Last-Modified** and **Date** headers, the proxy uses a maximum and minimum freshness limit. See *Setting an absolute freshness limit*, page 17.

◆ Checking revalidate rules in the **cache.config** file

Revalidate rules apply freshness limits to specific HTTP objects. You can set freshness limits for objects originating from particular domains or IP addresses, objects with URLs that contain specified regular expressions, and objects requested by particular clients, for example. See *cache.config*, page 261.

Modifying the aging factor for freshness computations

If an object does not contain any expiration information, Websense Content Gateway can estimate its freshness from the **Last-Modified** and **Date** headers. By default, the proxy stores an object for 10% of the time that elapsed since it last changed. You can increase or reduce the percentage.

1. Open the **records.config** file located in the Websense Content Gateway **config** directory.

2. Edit the following variable:

Variable	Description
`proxy.config.http.cache. heuristic_lm_factor`	Specify the aging factor for freshness computations. The default value is 0.10 (10 percent).

3. Save and close the file.

4. From the Websense Content Gateway **bin** directory, run content_line -x to apply the changes.

Setting an absolute freshness limit

Some objects do not have **Expires** headers or do not have both **Last-Modified** and **Date** headers. You can control how long these objects are considered fresh in the cache by specifying an absolute freshness limit. A longer lifetime means objects are kept in the cache longer. Performance can improve if pages are taken from the cache rather than going out to the network.

1. Navigate to the **Configure > Protocols > HTTP > Cacheability** tab.
2. In the **Minimum Heuristic Lifetime** area of the **Freshness** section, specify the minimum amount of time that HTTP objects without an expiration date can remain fresh in the cache before being considered stale. The default value is 3600 seconds (1 hour).
3. In the **Maximum Heuristic Lifetime** field, specify the maximum amount of time that HTTP objects without an expiration date can remain fresh in the cache before being considered stale. The default value is 86400 seconds (1 day).
4. Click **Apply**.

Specifying header requirements

To ensure freshness of the objects in the cache, configure Websense Content Gateway to cache only objects with specific headers.

> **Warning**
>
> By default, the proxy caches all objects (including objects with no headers). Websense recommends that you change the default setting only for specialized proxy situations. If you configure the proxy to cache only HTTP objects with **Expires** or **max-age** headers, the cache hit rate will be seriously reduced (very few objects have explicit expiration information).

1. Navigate to the **Configure > Protocols > HTTP > Cacheability** tab.
2. In the **Required Headers** area of the **Behavior** section, select one of the following:
   - **An Explicit Lifetime Header** to cache only HTTP objects with **Expires** or **Cache-Control** headers.
   - **A Last-Modified Header** to cache only HTTP objects with **Expires** or **Last-Modified** headers.
   - **No Required Headers** to cache all HTTP objects (no specific headers are required). This is the default.
3. Click **Apply**.

Cache-Control headers

Even though an object might be fresh in the cache, clients or servers might have constraints that prevent them from retrieving the object from the cache. For example, a client might request that a object not come from a cache, or if it does, it cannot have been cached for more than 10 minutes.

Websense Content Gateway bases the servability of a cached object on **Cache-Control** headers. **Cache-Control** headers can appear in both client requests and server responses.

The following **Cache-Control** headers affect whether objects are served from the cache:

◆ The **no-cache** header, sent by clients, tells the proxy to serve *no* objects directly from the cache; always obtain the object from the origin server. You can configure the proxy to ignore client **no-cache** headers (see *Configuring the proxy to ignore client no-cache headers*, page 25).

◆ The **max-age** header, sent by servers, is compared to the object age; if the age is less than **max-age**, the object is fresh and can be served.

◆ The **min-fresh** header, sent by clients, is an *acceptable freshness tolerance*. The client wants the object to be at least this fresh. If a cached object does not remain fresh at least this long in the future, it is revalidated.

◆ The **max-stale** header, sent by clients, permits the proxy to serve stale objects provided they are not too old. Some browsers might be willing to take slightly old objects in exchange for improved performance, especially during periods of poor Internet availability.

The proxy applies Cache-Control servability criteria *after* HTTP freshness criteria. For example, an object might be considered fresh, but if its age is greater than its `max-age`, it is not served.

Revalidating HTTP objects

When a client requests an HTTP object that is stale in the cache, Websense Content Gateway revalidates the object, querying the origin server to check if the object is unchanged. Revalidation results in one of the following:

◆ If the object is still fresh, the proxy resets its freshness limit and serves the object.

◆ If a new copy of the object is available, the proxy caches the new object, replacing the stale copy, and serves the object to the user simultaneously.

◆ If the object no longer exists on the origin server, the proxy does not serve the cached copy.

◆ If the origin server does not respond to the revalidation query, the proxy does not perform any validation; it serves the stale object from the cache.

By default, the proxy revalidates a requested HTTP object in the cache if it considers the object to be stale. The proxy evaluates object freshness as described in *HTTP object freshness*, page 16. You can configure how often you want the proxy to revalidate an HTTP object.

1. Navigate to the **Configure > Protocols > HTTP > Cacheability** tab.

2. In the **When to Revalidate** area of the **Behavior** section, select:

   ■ **Never Revalidate** to never verify the freshness of a requested HTTP object with the origin server.

   ■ **Always Revalidate** to always verify the freshness of a requested HTTP object with the origin server.

- **Revalidate if Heuristic Expiration** to verify the freshness of a requested HTTP object with the origin server if the object contains no **Expires** or **Cache-Control** headers. Websense Content Gateway considers all HTTP objects without **Expires** or **Cache-Control** headers to be stale.

- **Use Cache Directive or Heuristic** to verify the freshness of a requested HTTP object with the origin server when Websense Content Gateway considers the object in the cache to be stale. This is the default.

3. Click **Apply**.

> **Note**
>
> You can also set specific revalidation rules in the **cache.config** file. See *cache.config*, page 261.

FTP object freshness

FTP objects carry no time stamp or date information and remain fresh in the cache for the period of time you specify (from 15 minutes to 2 weeks), after which they are considered stale.

FTP objects can be requested from either an HTTP client (such as a browser) or an FTP client (such as WS_FTP). Websense Content Gateway caches the FTP objects requested from HTTP clients in HTTP format and the FTP objects requested from FTP clients in a proprietary format.

The procedure for specifying the freshness limit for objects requested by HTTP clients is different from the procedure for objects requested by FTP clients. Follow the procedure appropriate for your needs.

FTP objects requested by HTTP clients

You can set an absolute freshness limit for FTP objects requested by HTTP clients (FTP over HTTP objects).

> **Note**
>
> In addition to setting an absolute freshness limit for all FTP objects requested by HTTP clients, you can set freshness rules for specific FTP objects in the **cache.config** file (see *cache.config*, page 261).

1. Navigate to **Configure > Protocols > HTTP > Cacheability**.
2. In the **FTP Document Lifetime** area of the **Freshness** section, enter the amount of time that FTP objects requested by HTTP clients can remain fresh in the cache before being considered stale. The default value is 259200 seconds (3 days).
3. Click **Apply**.

FTP objects requested by FTP clients

You can set freshness limits for different types of FTP objects that are requested from FTP clients: directory listings, login messages, and FTP files.

1. Navigate to **Configure > My Proxy > Basic >General**.

2. Select **On** for FTP, if it is not already enabled. Then click **Apply** and click **Restart**.

3. Navigate to **Configure > Protocols > FTP > Cacheability**.

4. In the **Login Information** area of the **Freshness** section, enter the amount of time that FTP login messages can remain fresh in the cache before the proxy considers them stale. The default value is 2592000 seconds (30 days).

5. In the **Directory Listings** field, enter the amount of time that FTP directory listings can remain in the cache before the proxy considers them stale. The default value is 604800 seconds (7 days).

6. In the **Files** field, enter the amount of time that FTP files can remain fresh in the cache before the proxy considers them stale. The default value is 259200 seconds (3 days).

7. Click **Apply**.

Scheduling updates to local cache content

To further increase performance and to ensure that HTTP and FTP objects (requested from HTTP clients) are fresh in the cache, you can use the Scheduled Update option to configure the proxy to load specific objects into the cache at scheduled times.

> **Note**
>
> The Scheduled Update option does not update FTP objects requested from FTP clients.

To use the Scheduled Update option:

◆ Specify the list of URLs that contain the objects you want to schedule for update, the time the update should take place, and the recursion depth for the URL.

◆ Enable the Scheduled Update option and configure optional retry settings.

See *Configuring the Scheduled Update option*, page 22 for more information.

Websense Content Gateway uses the information you specify to determine the URLs for which it is responsible and, for each URL, derives all recursive URLs if applicable. It then generates a unique URL list. Using this list, the proxy initiates an HTTP GET

for each unaccessed URL, ensuring that it remains within the user-defined limits for HTTP concurrency at any given time.

> ✓ **Note**
> The system logs the completion of all HTTP GET operations, enabling you to monitor the performance of this feature.

The Force Immediate Update option that enables you to update URLs without waiting for the specified update time to occur. You can use this option to test your scheduled update configuration. See *Forcing an immediate update*, page 23.

Configuring the Scheduled Update option

1. Navigate to **Configure > Protocols > HTTP Scheduled Update > Update URLs**.
2. In the **Scheduled Object Update** area, click **Edit File** to open the configuration file editor for the **update.config** file.
3. Enter the following information:
   - In the **URL** field, enter the URL you want to schedule for update.
   - *Optional*. In the **Request Headers** field, enter the semicolon-separated list of headers passed in each GET request. You can define any request header that conforms to the HTTP specification.
   - In the **Offset Hour** field, enter the base hour used to derive the update periods. You can specify a value in the range 00 to 23.
   - In the **Interval** field, enter the interval (in seconds) at which updates occur, starting at the offset hour.
   - In the **Recursion Depth** field, enter the depth to which referenced URLs are recursively updated, starting at the given URL. For example, a recursion depth of 1 updates the given URL, as well as all URLs immediately referenced by links from the original URL.
4. Click **Add**, and then click **Apply**.
5. Click **Close**.
6. Click the **General** tab.
7. Enable **Scheduled Update**.
8. In the **Maximum Concurrent Updates** field, enter the maximum number of simultaneous update requests allowed at any time to prevent the scheduled update process from overburdening the host. The default is 100.
9. In the **Count** field of the **Retry on Update Error** section, enter the number of times you want to retry the scheduled update of a URL in the event of failure. The default value is 10.

10. In the **Interval** field of the **Retry on Update Error** section, enter the delay in seconds between each scheduled update retry for a URL in the event of failure. The default value is 2.

11. Click **Apply**.

Forcing an immediate update

The Force Immediate Update option lets you verify the URLs listed in the **update.config** file immediately. This option disregards the offset hour and interval set in the **update.config** file and updates the URLs listed.

> **Important**
>
> When you enable the Force Immediate Update option, the proxy continually updates the URLs specified in the **update.config** file until you disable the option.

1. Navigate to **Configure > Protocols > HTTP Scheduled Update > General**.

2. Ensure that **Scheduled Update** is enabled.

3. Click the **Update URLs** tab.

4. Enable **Force Immediate Update**.

5. Click **Apply**.

Pinning content in the cache

The cache pinning option configures Websense Content Gateway to keep certain HTTP objects (and FTP objects requested from HTTP clients) in the cache for a specified time. Use this option to ensure that the most popular objects are in the cache when needed and that the proxy does not delete important objects from the cache.

> **Note**
>
> The proxy observes Cache-Control headers and pins an object in the cache only if it is cacheable.

To use cache pinning, perform the following tasks:

- Set cache pinning rules in the **cache.config** file. See *Setting cache pinning rules*, page 23.

- Enable the cache pinning option. See *Enabling cache pinning*, page 24.

Setting cache pinning rules

1. Navigate to **Configure > Protocols > HTTP > Cacheability**.

2. Click **Edit File** at the end of the page to display the configuration file editor for the **cache.config** file.

3. In the fields provided, supply the following information:

   - From the **Rule Type** drop-down box, select **pin-in-cache**.
   - From the **Primary Destination Type** drop-down box, select **url_regex**.
   - In the **Primary Destination Value** field, specify the URL you want to pin in the cache.
   - In the **Time Period** field, specify the amount of time that the proxy pins the object in the cache.

     In addition, you can add secondary specifiers (such as **Prefix** and **Suffix**) to the rule. All the fields are described under *HTTP*, page 207.

4. Click **Add** to add the rule to the list, and then click **Apply**.

5. Click **Close**.

Enabling cache pinning

1. On **Configure > Subsystems > Cache > General**, enable **Allow Pinning**.

2. Click **Apply**.

To cache or not to cache?

When Websense Content Gateway receives a request for a Web object that is not in the cache, it retrieves the object from the origin server and serves it to the client. At the same time, the proxy checks if the object is cacheable before storing it in its cache to serve future requests.

Websense Content Gateway determines if an object is cacheable based on protocol:

- For HTTP objects, the proxy responds to caching directives from clients and origin servers. In addition, you can configure the proxy not to cache certain objects. See *Caching HTTP objects*, page 24.
- For FTP objects, the proxy responds to caching directives you specify through configuration options and files. See *Caching FTP objects*, page 31.

Caching HTTP objects

Websense Content Gateway responds to caching directives from clients and origin servers, as well as directives you specify through configuration options and files.

This section discusses the following topics:

- *Client directives*, page 25
- *Origin server directives*, page 26

◆ *Configuration directives*, page 28

Client directives

By default, Websense Content Gateway does *not* cache objects with the following request headers:

◆ `Cache-Control: no-store`

◆ `Cache-Control: no-cache`

> ✔ **Note**
>
> You can configure the proxy to ignore the `Cache-Control: no-cache` header. See *Configuring the proxy to ignore client no-cache headers*, page 25.

◆ `Cookie:` (for text objects)

By default, the proxy caches objects served in response to requests that contain cookies unless the object is text. You can configure the proxy to *not* cache cookied content of any type, cache all cookied content, or cache cookied content that is of image type only. See *Caching cookied objects*, page 29.

◆ `Authorization:`

> ✔ **Note**
>
> FTP objects requested from HTTP clients can also contain `Cache-Control: no-store`, `Cache-Control: no-cache`, or `Authorization` headers. If an FTP object requested from an HTTP client contains such a header, the proxy does not cache it unless explicitly configured to do so.

Configuring the proxy to ignore client no-cache headers

By default, Websense Content Gateway observes client `Cache Control:no-cache` directives. If a requested object contains a `no-cache` header, the proxy forwards the request to the origin server even if it has a fresh copy in the cache.

You can configure the proxy to ignore client `no-cache` directives. In this case, the proxy ignores `no-cache` headers from client requests and serves the object from its cache.

> ❗ **Important**
>
> The default behavior of observing `no-cache` directives is appropriate in most cases. Configure Websense Content Gateway to ignore client `no-cache` directives only if you are knowledgeable about HTTP 1.1.

1. Navigate to **Configure > Protocols > HTTP > Cacheability**.
2. In the **Behavior** section, enable the **Ignore "no-cache" in Client Requests** option.
3. Click **Apply**.

> ✔ **Note**
>
> Certain versions of Microsoft Internet Explorer do not request cache reloads from transparent caches when the user presses the browser **Refresh** button. This can prevent content from being loaded directly from the origin server. You can configure Websense Content Gateway to treat Microsoft Internet Explorer requests more conservatively, providing fresher content at the cost of serving fewer documents from the cache. You can configure the proxy to add `no-cache` headers to requests from Microsoft Internet Explorer in Websense Content Manager (in the **Behavior** section **Configure > Protocols > HTTP > Cacheability** tab).

Origin server directives

By default, Websense Content Gateway does not cache objects with the following response headers:

◆ `Cache-Control: no-store`

◆ `Cache-Control: private`

◆ `WWW-Authenticate:`

> ✔ **Note**
>
> You can configure the proxy to ignore `WWW-Authenticate` headers. See *Configuring the proxy to ignore WWW-Authenticate headers*, page 27.

◆ `Set-Cookie:`

◆ `Cache-Control: no-cache`

> ✔ **Note**
>
> You can configure the proxy to ignore `no-cache` headers. See *Configuring the proxy to ignore server no-cache headers*, page 27.

◆ `Expires:` header with value of 0 (zero) or a past date

Configuring the proxy to ignore server no-cache headers

By default, Websense Content Gateway observes `Cache-Control:no-cache` directives. A response from an origin server with a `no-cache` header is not stored in the cache, and any previous copy of the object in the cache is removed.

> **Important**
> If you configure the proxy to ignore `no-cache` headers, it also ignores `no-store` headers.

> **Important**
> The default behavior of observing `no-cache` directives is appropriate in most cases. Configure the proxy to ignore origin server `no-cache` headers only if you are knowledgeable about HTTP 1.1.

You can configure the proxy to ignore origin server `no-cache` headers.

1. Open the **records.config** file located in the Websense Content Gateway **config** directory.
2. Edit the following variable:

Variable	Description
`proxy.config.http.cache.ignore_server_no_cache`	Set to 1 to ignore server directives to bypass the cache.

3. Save and close the file.
4. From the Websense Content Gateway **bin** directory, run `content_line -x` to apply the changes.

Configuring the proxy to ignore WWW-Authenticate headers

By default, Websense Content Gateway does not cache objects that contain `WWW-Authenticate` response headers. The `WWW-Authenticate` header contains authentication parameters that the client uses when preparing the authentication challenge response to an origin server.

> **Important**
> The default behavior of not caching objects with `WWW-Authenticate` headers is appropriate in most cases. Configure the proxy to ignore server `WWW-Authenticate` headers only if you are knowledgeable about HTTP 1.1.

You can configure the proxy to ignore origin server `WWW-Authenticate` headers, in which case, objects with `WWW-Authenticate` headers are stored in the cache for future requests.

1. Open the **records.config** file located in the Websense Content Gateway **config** directory.
2. Edit the following variable:

Variable	Description
`proxy.config.http.cache.ignore_` `authentication`	Set to 1 to cache objects with `WWW-Authenticate` headers.

3. Save and close the file.
4. From the Websense Content Gateway **bin** directory, run `content_line -x` to apply the changes.

Configuration directives

In addition to client and origin server directives, Websense Content Gateway responds to directives you specify through configuration options and files.

You can configure the proxy to:

◆   *Not* cache any HTTP objects. See *Disabling HTTP object caching*, page 28.

◆   Cache dynamic content (objects with URLs that contain a question mark (?), a semicolon (;), or cgi, or that end in .asp). See *Caching dynamic content*, page 29.

◆   Cache objects served in response to the `Cookie:` header. See *Caching cookied objects*, page 29.

◆   Observe never-cache rules in the **cache.config** file. See *cache.config*, page 261.

Disabling HTTP object caching

By default, Websense Content Gateway caches all HTTP objects except those for which you have set never cache rules in the **cache.config** file. You can disable HTTP object caching so that all HTTP objects are served from the origin server and never cached.

1. Navigate to **Configure** > **Protocols** > **HTTP > Cacheability**.
2. Disable **HTTP Caching**.
3. Click **Apply**.

Caching dynamic content

A URL is considered dynamic if it contains a question mark (?), a semicolon (;), or cgi, or if it ends in .asp. By default, Websense Content Gateway does *not* cache dynamic content. However, you can configure the proxy to cache this content.

> ⚠️ **Warning**
> It is recommended that you configure the proxy to cache dynamic content for specialized proxy situations only.

1. Navigate to **Configure > Protocols > HTTP > Cacheability**.
2. In the **Dynamic Caching** section, enable **Caching Documents with Dynamic URLs**.
3. Click **Apply**.

Caching cookied objects

By default, Websense Content Gateway caches objects served in response to requests that contain cookies *unless* the object is text. The proxy does not cache cookied text content, because object headers are stored as well as the object, and personalized cookie header values could be saved with the object.

With non-text objects, personalized headers are unlikely to be delivered or used.

1. Navigate to **Configure > Protocols > HTTP > Cacheability**.
2. In the **Caching Response to Cookies** area of the **Dynamic Caching** section, select a caching option:
   - Select **Cache All but Text** to cache all cookied content except content that is text (this is the default setting).
   - Select **Cache Only Image Types** to cache cookied content that is an image.
   - Select **Cache Any Content Type** to cache cookied content of all types.
   - Select **No Cache on Cookies** to *not* cache cookied content of any type.
3. Click **Apply**.

Forcing object caching

You can force Websense Content Gateway to cache specific URLs (including dynamic URLs) for a specified duration regardless of `Cache-Control` response headers.

1. Navigate to **Configure > Protocols > HTTP > Cacheability**.
2. Click **Edit File** at the end of the page to display the configuration file editor for the **cache.config** file.
3. In the fields provided, supply the following information:
   - From the **Rule Type** drop-down box, select **ttl-in-cache**.

- From the **Primary Destination Type** drop-down box, select **url_regex**.

- In the **Primary Destination Value** field, specify the URL you want to force cache.

- In the **Time Period** field, specify the amount of time that the proxy can serve the URL from the cache.

  In addition, you can add secondary specifiers (such as **Prefix** and **Suffix**) to the rule. All the fields are described in *HTTP*, page 207.

4. Click **Add**, and then click **Apply**.
5. Click **Close**.

Caching HTTP alternates

Some origin servers answer requests to the same URL with a variety of objects. The content of these objects can vary, according to whether a server delivers content for different languages, targets different browsers with different presentation styles, or provides different document formats (HTML, PDF). Different versions of the same object are termed *alternates* and are cached by Websense Content Gateway based on `Vary` response headers.

Configuring how Websense Content Gateway caches alternates

You can specify additional request and response headers for specific content types that the proxy will identify as alternates for caching.

1. Navigate to **Configure > Protocols > HTTP > Cacheability**.
2. In the **Vary Based on Content Type** section, click **Enabled** to cache alternate versions of HTTP documents that do not contain the **Vary** header.
3. Specify additional request and response headers for the proxy server to identify:

   - In the **Vary by Default on Text** field, enter the HTTP header field on which you want to vary if the request is for text (for example, an HTML document).

   - In the **Vary by Default on Images** field, enter the HTTP header field on which you want to vary if the request is for images (for example, a `.gif` file).

- ■ In the **Vary by Default on Other Document Types** field, enter the HTTP header field on which you want to vary if the request is for anything other than text or images.

> **Note**
>
> If you specify `Cookie` as the header field on which to vary in the above fields, make sure that the appropriate option is enabled in the **Caching Response to Cookies** area of the **Dynamic Caching** section. For example, if you enable the **Cache Only Image Types** option in the **Caching Response to Cookies** area and you enable the **Vary by Default on Text** option in the **Vary Based on Content Type** section, alternates by cookie will not apply to text.

4. Click **Apply**.

Limiting the number of alternates for an object

You can limit the number of alternates Websense Content Gateway can cache per object. The default number of alternates is 3.

> **Note**
>
> Large numbers of alternates can affect proxy performance because all alternates have the same URL. Although Websense Content Gateway can look up the URL in the index very quickly, it must scan sequentially through available alternates in the object store.

1. Navigate to **Configure > Protocols > HTTP > Cacheability**.
2. In the **Maximum Alternates** field, enter the maximum number of alternate versions of an object you want the proxy to cache. The default value is 3.
3. Click **Apply**.

Caching FTP objects

FTP objects can be requested from either an HTTP client (such as a browser) or an FTP client (such as WS_FTP).

For FTP objects requested from HTTP clients (FTP over HTTP), perform the following configuration to determine what the proxy stores in the cache:

- ◆ Disable FTP over HTTP caching so that the proxy does not cache any FTP objects requested from HTTP clients (see *Disabling FTP over HTTP caching*, page 32).
- ◆ Set never cache rules in the **cache.config** file (see *cache.config*, page 261).

◆ Configure the proxy to ignore client `Cache-Control: no-store` or `Cache-Control: no-cache` headers (see *Configuring the proxy to ignore client no-cache headers*, page 25).

For FTP objects requested from FTP clients (FTP proxy objects), perform the following configuration to determine what Websense Content Gateway stores in the cache:

◆ Disable FTP proxy object caching so that Websense Content Gateway does not cache any FTP objects requested from FTP clients (see *Disabling FTP proxy object caching*, page 32).

◆ Disable caching of simple and/or full directory listings (see *Disabling caching of full or simple directory listings*, page 32).

Disabling FTP over HTTP caching

You can configure Websense Content Gateway not to cache any FTP objects that are requested from HTTP clients by disabling the FTP over HTTP option. The proxy processes the requests by forwarding them to the FTP server but does not cache any requested objects.

1. Navigate to **Configure > Protocols > HTTP > Cacheability**.
2. In the **Caching** section, disable **FTP over HTTP Caching**.
3. Click **Apply**.

Disabling FTP proxy object caching

You can configure Websense Content Gateway not to cache any FTP objects that are requested from FTP clients.

The proxy processes FTP requests by forwarding them to the FTP server (as long as the **FTP** processing option is enabled in the **Features** table on the **Configure/My Proxy/Basic** tab) but does not cache any requested objects.

1. Navigate to **Configure > Protocols > FTP > Cacheability**.
2. Disable **FTP Caching**.
3. Click **Apply**.

Disabling caching of full or simple directory listings

You can configure Websense Content Gateway not to cache simple and/or full directory listings. A simple directory listing contains no arguments: for example, `dr/ls`. A full directory listing does contain arguments: for example, `ls *.txt`.

1. Navigate to **Configure > Protocols > FTP > Cacheability**.
2. In the **Directory Caching** section:

- ■ Disable the **Simple** option if you do not want to cache simple directory listings.
- ■ Disable the **Full** option if you do not want to cache full directory listings.

3. Click **Apply**.

4 | Explicit Proxy Caching

If you do not configure Websense Content Gateway to use the transparency option (where client requests are intercepted on the way to origin servers by a switch or router and rerouted to the proxy), clients must configure their Web browsers to send requests to the proxy cache. Clients can configure their Web browsers in one of three ways:

◆ By configuring their browsers to send requests directly to the proxy. See *Manual browser configuration*, page 35.

◆ By configuring their browsers to download proxy configuration instructions from a PAC (Proxy Auto-Configuration) file. See *Using a PAC file*, page 36.

◆ By using WPAD (Web Proxy Auto-Discovery) to download proxy configuration instructions from a WPAD server (Microsoft Internet Explorer only). See *Using WPAD*, page 37.

Manual browser configuration

To configure a browser to send requests to Websense Content Gateway, clients must provide the following information for each protocol they want the proxy to serve to their browsers:

◆ The proxy's host name or IP address

◆ The proxy server port. The Websense Content Gateway default proxy server port is 8080.

> **Important**
>
> Do not set up the IP address of the Websense Content Gateway proxy to be a virtual IP address.
>
> Although Websense Content Manager does not prohibit the entry of a virtual IP address, the proxy does not function properly if a VIP is used.

In addition, clients can specify not to use the proxy for certain sites. Requests to the listed sites go directly to the origin server.

For Microsoft Internet Explorer version 7.0 and later, proxy configuration settings are in **Tools > Internet Options > Connections > LAN Settings**. By default, Microsoft Internet Explorer sets all protocols to the same proxy server. To configure each protocol separately, click **Advanced** in the **LAN Settings** section. See the browser documentation for complete proxy configuration instructions.

For Mozilla Firefox 2.0 and later, proxy configuration settings are in **Tools > Options > Advanced > Network > Settings > Connection Settings > Manual Proxy Configuration**. By default, you must configure each protocol separately. However, you can set all protocols to the same proxy server by selecting **Use this proxy server for all protocols.**

You do not have to set configuration options on the proxy to accept requests from manually configured browsers.

Using a PAC file

A PAC file is a JavaScript function definition that a browser calls to determine how requests are handled. Clients must specify in their browser settings the URL from which the PAC file is loaded.

You can store a PAC file on the proxy and provide the URL for this file to your clients. If you have a **proxy.pac** file, copy it into the Websense Content Gateway **config** directory.

> ✓ **Note**
>
> The PAC file can reside on any server in your network.
>
> If your subscription includes SSL Manager, refer to *Running in explicit proxy mode*, page 122, for information on a PAC file to use with HTTPS traffic.

1. If you have an existing **wpad.dat** file, replace the **wpad.dat** file located in the Websense Content Gateway **config** directory with your existing file.
2. Navigate to the **Configure** > **Content Routing** > **Browser Auto-Config > PAC** tab.
3. In the **Auto-Configuration Port** field, specify the port that Websense Content Gateway uses to serve the PAC file. The default port is 8083.
4. The PAC Settings area displays the **proxy.pac** file:
   - If you copied an existing PAC file into the Websense Content Gateway **config** directory, the **proxy.pac** file contains your proxy configuration settings. Check the settings and make changes if necessary.
   - If you did not copy an existing PAC file into the Websense Content Gateway **config** directory, the PAC Settings area is empty. Enter the script that provides the proxy server configuration settings. A sample script is provided in *Sample PAC file*, page 37.

5. Click **Apply**.

6. Click **Restart** on **Configure > My Proxy > Basic > General**.

7. Inform your users to set their browsers to point to this PAC file.

   For example, if the PAC file is located on the proxy server with the host name `proxy1` and Websense Content Gateway uses the default port 8083 to serve the file, users must specify the following URL in the proxy configuration settings:

   ```
   http://proxy1.company.com:8083/proxy.pac
   ```

The procedures for specifying the PAC file location vary among browsers. For example, for Microsoft Internet Explorer version 6.0, you set the location of the PAC file in the **Use automatic configuration script** field under **Tools > Internet Options > Connections > LAN Settings**. For Mozilla Firefox 2.0.0 and later, proxy configuration settings are in **Tools > Options > Advanced > Network > Settings > Connection Settings > Automatic proxy configuration URL**. See the documentation for your browser for details.

Sample PAC file

The following sample PAC file instructs browsers to connect directly to all hosts without a fully qualified domain name and to all hosts in the local domain. All other requests go to the proxy server called `myproxy.company.com`.

```
function FindProxyForURL(url, host)
{
if (isPlainHostName(host) || dnsDomainIs(host,
".company.com"))
return "DIRECT";
else
return "PROXY myproxy.company.com:8080; DIRECT";
}
```

Using WPAD

WPAD allows Internet Explorer version 7 and later to automatically detect a server that will supply it with proxy server configuration settings. Clients do not have to configure their browsers to send requests to a proxy server: a single server provides the settings to all clients on the network.

When an Internet Explorer version 7 or later browser starts up, it searches for a WPAD server that will supply it with proxy server configuration settings. It prepends the host name WPAD to the current fully qualified domain name. For example, a client in `x.y.company.com` searches for a WPAD server at `wpad.x.y.company.com`. If unsuccessful, the browser removes the bottommost domain and tries again; for example, it tries `wpad.y.company.com`. The browser stops searching when it detects a WPAD server or reaches the third-level domain, `wpad.company.com`. The

algorithm stops at the third level so that the browser does not search outside the current network.

> **✓ Note**
>
> By default, Microsoft Internet Explorer version 7 and later are set to automatically detect WPAD servers. However, browser users can disable this setting.

You can configure Websense Content Gateway to be a WPAD server:

1. If you have an existing **wpad.dat** file, replace the **wpad.dat** file located in the Websense Content Gateway **config** directory with your existing file.

2. Navigate to **Configure** > **My Proxy > Basic > General**.

3. In the **Features** table, ensure that **ARM** is **On** (the default) in the **Networking** section.

4. Navigate to **Configure** > **Content Routing** > **Browser Auto-Config > WPAD** to display the **wpad.dat** file**.**

5. The WPAD Settings area displays the **wpad.dat** file:

   ■ If you copied an existing **wpad.dat** file into the Websense Content Gateway **config** directory, the file contains your proxy configuration settings. Check the settings and make changes if necessary.

   ■ If you did not copy an existing **wpad.dat** file into the Websense Content Gateway **config** directory (default location is in **/opt/WCG**), the WPAD Settings area is empty. Enter a script that will provide the proxy server configuration settings. A sample script is provided in *Sample PAC file*, page 37 (a **wpad.dat** file can contain the same script as a **proxy.pac** file).

6. Click **Apply**.

7. Navigate to **Configure** > **Networking > ARM**.

8. In the **Network Address Translation (NAT)** section, click **Edit File** to add a special remap rule to the **ipnat.conf** file.

9. Enter information in the fields provided, and then click **Add**:

   ■ In the **Ethernet Interface** field, enter the network interface that receives browser WPAD requests (for example hme0 or eth0).

   ■ From the **Connection Type** drop-down list, select **tcp**.

   ■ In the **Source IP** field, enter the IP address of the Websense Content Gateway server that will be resolved to the WPAD server name by the local name servers followed by
   /32; for example: 123.456.7.8/32.

   ■ In the **Source Port** field, enter **80**.

   ■ In the **Destination IP** field enter the same IP address you entered in the **Source IP** field but omit /32.

   ■ In the **Destination Port** field, enter **8083**.

10. Use the arrow keys on the left side to move the new rule to the first line in the file.

11. Click **Apply**, and then click **Close**.

12. Click **Restart** on the **Configure > My Proxy > Basic > General**.

5 | Transparent Proxy Caching and ARM

The transparent proxy caching option enables Websense Content Gateway to respond to Internet requests without requiring users to reconfigure their browser settings. It does this by redirecting the traffic flow into the proxy cache after the traffic has been intercepted by a Layer 4 (L4) switch or router.

In transparent interception:

1. The proxy intercepts client requests to origin servers by a switch or router. See *Interception strategies*, page 42.

2. The Adaptive Redirection Module (ARM) changes the destination IP address of an incoming packet to the proxy's IP address and the destination port to the proxy port (if different). See *Enabling the ARM option*, page 42.

3. The proxy receives and begins processing the intercepted client requests. If a request is a cache hit, the proxy serves the requested object. If a request is a miss, the proxy retrieves the object from the origin server and serves it to the client.

4. On the way back to the client, the ARM changes the source IP address to the origin server IP address and the source port to the origin server port.

For HTTP, the proxy can identify problem clients and servers, and the ARM can disable interception for these clients and servers, passing their traffic to the origin server. You can create static rules to exempt clients and servers from caching. See *Interception bypass*, page 54.

To prevent unauthorized access to machines running Websense Content Gateway, you can configure the ARM to use an access control list employing administrator-specified rules to allow or deny other computers from communicating with the machine. This enables you to effectively create a firewall in front of Websense Content Gateway, denying potentially malicious packets from reaching the TCP/IP stack on the machine. See *Controlling host access to the proxy server*, page 102.

Related topics:

Connection load shedding, page 58
Reducing DNS lookups, page 58
IP spoofing, page 59

Enabling the ARM option

The Websense Content Gateway ARM inspects incoming packets before the IP layer sees them and readdresses the packets to the proxy server so that they can be served from the cache.

The ARM can make two changes to an incoming packet's address: its destination IP address and its destination port. For example, the destination IP address of an HTTP packet is readdressed to the IP address of the proxy and the destination HTTP port is readdressed to the Websense Content Gateway HTTP proxy port (usually port 8080).

On the way back to the client, the ARM changes the source IP address to the origin server IP address and the source port to the origin server port.

The ARM component consists of several files and a kernel module, which are installed during product installation. The installation program also creates redirection rules to readdress packets using the IP address of the proxy machine and default port assignments.

For the proxy to serve HTTP or FTP requests transparently, you must check the redirection rules in the **ipnat.conf** file and edit them if necessary.

1. Check that the ARM has been enabled by looking at the **Configure > My Proxy > Basic > General** tab. Click ARM **On** if it is not already selected. (ARM is enabled by default during installation.)

2. Navigate to the **Configure > Networking > ARM > General** tab.

   The **Network Address Translation (NAT)** section displays the redirection rules in the **ipnat.conf** file. Check the redirection rules and make changes if necessary.

   a. To change a redirection rule, click **Edit File** to open the configuration file editor for the **ipnat.conf** file.

   b. Select the rule you want to edit and then modify the appropriate fields. Click **Set** and then click **Apply** to apply your changes. Click **Close** to exit the configuration file editor.

   All fields are described in *Configuration Options*.

3. Click **Restart** on **Configure > My Proxy > Basic > General**.

Interception strategies

Websense Content Gateway supports the following transparency routing solutions:

◆ A Layer 4 switch. See *Using a Layer 4 switch*, page 43.

◆ A Cisco IOS-based router using the Web Cache Control Protocol (WCCP). See *Using a WCCP-enabled router*, page 44.

◆ Policy-based routing. See *Using multicast mode*, page 52.

◆ Software routing. See *Using software-based routing*, page 53.

How client requests reach Websense Content Gateway depends on network topology. In a complex network, you must decide which clients are to be served transparently and make sure that the proxy is positioned to intercept their requests. Websense Content Gateway, or routers or switches feeding Websense Content Gateway, are often deployed at a major artery or aggregation pipe to the Internet.

Using a Layer 4 switch

Layer 4 switches can redirect supported protocols to the proxy, while passing all other Internet traffic through directly to its destination, as shown below for HTTP.
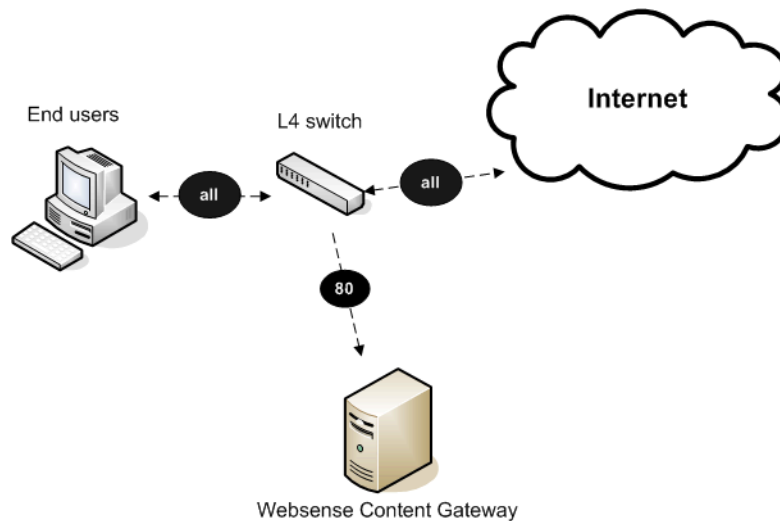
Using a Layer 4 switch to filter HTTP requests

Layer 4 switches offer the following features, depending on the particular switch:

◆ A Layer 4 switch that can sense downed hosts on the network and redirect traffic adds reliability.

◆ If a single Layer 4 switch feeds several proxy servers, the switch handles load balancing among the Websense Content Gateway nodes. Different switches might use different load-balancing methods, such as round-robin or hashing. If a node becomes unavailable, the switch redistributes the load. When the node returns to service, some switches return the node to its previous workload, so that the node cache need not be repopulated; this feature is called *cache affinity*.

> **Note**
>
> It is recommended that you do *not* enable Websense Content Gateway virtual IP failover when a switch is providing load balancing in a cluster configuration. Also, if the switch provides a load-balancing option that has cache affinity, it is recommended that you use management-only clustering mode.

Using a WCCP-enabled router

Websense Content Gateway supports WCCP 1.0 and WCCP 2.0.

A WCCP 1.0-enabled router can send all port 80 (HTTP) traffic to the proxy server, as shown below. The ARM readdresses port 80 to the Websense Content Gateway proxy port (by default, port 8080). The proxy processes the request as usual, retrieving the requested document from the cache if it is a hit and sending the response back to the client. Along the way, the ARM readdresses the proxy port in the response header to port 80 (undoing the readdressing it did on the way to the proxy). The user then sees the response as if it had been sent directly from the origin server.

A WCCP 2.0-enabled router works in the same way. In addition to port 80 (HTTP) traffic, WCCP 2.0 supports additional protocols, including FTP and HTTPS.
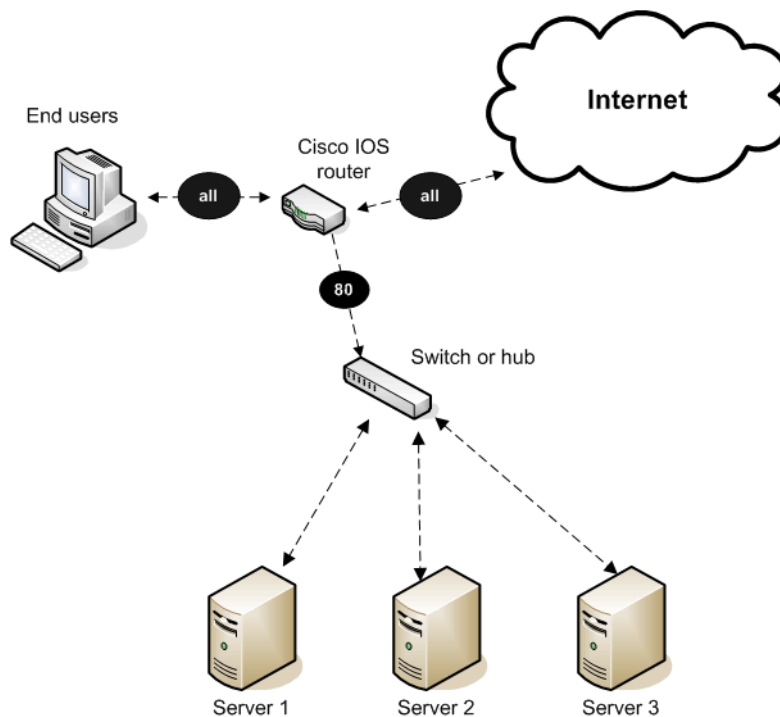
Using a Cisco IOS router to send port 80 traffic to several Websense Content Gateway servers

WCCP provides the following routing benefits:

◆ The WCCP-enabled router and the proxy exchange heartbeat messages, letting each other know they are running. The WCCP router reroutes traffic if the proxy server becomes unavailable.

◆ If several Websense Content Gateway servers receive traffic from a WCCP router, WCCP balances the load among the servers. The group of servers is called a *WCCP cache farm*. See *WCCP load balancing*, page 45.

◆ Websense Content Gateway handles node failure in WCCP cache farms. If one node becomes unavailable, its load is redistributed among the remaining nodes.

◆ In WCCP 2.0, you can use multiple routers. Traffic flowing through multiple routers can share the same pool of caches.

ARM bypass and WCCP

If Websense Content Gateway has an ARM bypass rule (discussed in *Interception bypass*, page 54), Websense Content Gateway forwards particular client requests directly to the origin server, bypassing the cache.

Bypassed requests are unchanged by the ARM; they retain their client source IP addresses. In WCCP 1.0, ARM bypass rules cannot work if the WCCP router is also the Websense Content Gateway default gateway router. The WCCP router sends port 80 traffic to the proxy server *and* it serves as the servers' default gateway or *next hop* to the Internet. Bypassed requests go to the WCCP router, which sends them back to Websense Content Gateway.

In WCCP 2.0, you can exclude certain router interfaces from redirection. Websense Content Gateway bypass rules can work if you exclude the router interface on which Websense Content Gateway is connected from using WCCP. You can do this by setting the router configuration command `ip wccp redirect exclude in` (see Websense Content Gateway *Installation Guide* for information about router configuration).

WCCP load balancing

If a WCCP router serves several nodes, the router balances load among the proxy caching servers. The router sends each node requests aimed at a particular range of IP addresses, so that each node is responsible for caching content residing at particular IP addresses. If a node becomes unavailable, its traffic is redistributed.

Websense Content Gateway also supports cache affinity. If a node becomes unavailable and then recovers, Websense Content Gateway returns the node to its former load distribution. The node's cache does not need to be repopulated.

The WCCP cache farm acts as a simple form of distributed cache, which is sufficient for many applications. A WCCP-enabled network device distributes traffic to individual proxy servers based on the IP address of the origin server. Each node caches objects requested from a set of origin servers that belong to that node's assigned range of destination IP addresses.

Busy origin servers are often mapped to several IP addresses (using a DNS round-robin mechanism). Using WCCP-based load balancing alone, each of these different IP addresses could be allotted to different Websense Content Gateway nodes. This can

result in a slightly lower hit rate and wasted cache space, since the same content is being replicated across nodes. The Websense Content Gateway full-clustering mode ensures that all requests to a specific page on that origin server (no matter which IP address is used) are cached on the same node.

With full clustering, objects are distributed among nodes according to their URLs; WCCP distributes objects according to destination IP address. If a particular IP address is receiving many requests, WCCP load balancing might lead to a hot spot, where all of that site's traffic is cached on one node, instead of being distributed among the nodes. Full-clustering mode distributes different pages from the busy site to different Websense Content Gateway nodes.

In general, if load-handling capacity and latency are most important, management-only clustering is recommended in WCCP environments.

If you are running clustered Websense Content Gateway proxy caching servers, it is recommended that you do not enable virtual IP failover in WCCP environments. The Websense Content Gateway WCCP failover mechanism handles node failures and restarts. See *Virtual IP failover*, page 65, for details.

Configuring WCCP routers

You must deploy with a WCCPv2-compliant router if the proxy must support HTTPS (your subscription includes SSL Manager) or FTP. Otherwise, you can deploy either a WCCPv1- or WCCPv2-compliant router.

To use a WCCPv1 router with the proxy, you need only to enable WCCP processing in Websense Content Manager (see *Enabling WCCP 1.0*, page 50). There is just one protocol (HTTP) and port (80), and it is already configured in the router.

To prepare a WCCPv2 router for use with the proxy, do the following:

1. Configure a service group for each protocol you intend to use (see *Configuring service groups*, page 47).

2. Configure the router to enable WCCP processing for these service groups (see *Enabling WCCP processing for a service group*, page 48).

3. Optionally, enable router security (see *Enabling WCCP 2.0 security on the router*, page 49).

> ✔ **Note**
>
> For instructions on configuring your specific router, please refer to the documentation provided by your hardware vendor. For Cisco routers, see http://www.cisco.com/univercd/cc/td/doc/product/core/ and search for your IOS and device version, for example, IOS 12.2(4) 3750.

4. When you are done configuring the router, you must also enable WCCP processing in Websense Content Manager. (See *Enabling WCCP 2.0*, page 51.)

Configuring service groups

When configuring a WCCPv2-compliant router, you must specify *service group ID numbers* when executing certain commands. (This is not required for WCCPv1.)

A service group ID number represents a group of Websense Content Gateway servers and routers that support a specific protocol. The service group ID also represents the router port and Websense Content Gateway port(s) associated with that particular protocol. For example, service group ID 0 is used for port 80 HTTP traffic, and service group ID 1 is used for port 119.

The default service group IDs for WCCPv2 are:

Service ID	Port	Traffic Type
0	80	HTTP
5	21	FTP
70	443	HTTPS (requires subscription to SSL Manager)

You must configure the service group IDs on your router so it can send traffic to Websense Content Gateway servers. Follow the instructions in your router documentation for specifics, but in general:

1. Specify the WCCP version you want to use, for example:

    ```
    ip wccp version 2
    ```

2. If you used another proxy cache with your router before, disable the service ID that was previously used. For example, if you have a Cisco router, disable the service ID `webcache` by issuing this command:

    ```
    no ip wccp webcache
    ```

    To see what has been configured on the router for WCCP, type:

    ```
    sh run | incl wccp
    ```

3. Specify the service group IDs you will be using with the Websense Content Gateway. For the specific commands to use, see your router documentation.

    You must configure each service group supported by the router individually. You cannot configure a router globally.

All Websense Content Gateway servers (and routers) assigned to a particular service group must support the protocol associated with that service group. If you want to use a different service ID for HTTP traffic than the default, you can assign a different service ID. To do so, follow these steps:

1. Add the variable `proxy.config.wccp2.HTTP_svc_id` to the **records.config** file (default location **/opt/WCG/config**).

2. Assign the appropriate service ID as the value for this variable. From the Websense Content Gateway **bin** directory, run:

    ```
    content_line -x
    ```

    to apply the changes.

Note that on a given router, service group IDs are unique; there is only one service ID 0, and only one service ID 1, and so on.

Enabling WCCP processing for a service group

For each service group that you configure (WCCPv2 only), you must enable WCCP processing. WCCPv2 routers contain multiple network interfaces, including:

◆ an interface dedicated to outbound traffic, that is, aimed at the Internet

◆ one or more interfaces connected to Websense Content Gateway

◆ one or more interfaces that receive inbound client traffic

Following are some guidelines for enabling WCCP processing for a service group on a router. Consult the procedures in your router documentation for specific details.

1. Turn on the WCCP feature:

   ```
   ip wccp <service group ID> password [0-7] <passwd>
   ```

2. On the router/switch interface, enable redirection for incoming packets or outgoing packets.

   For *incoming* packets:

   ```
   ip wccp <service group ID> redirect in
   ```
   For example, to turn on redirection for HTTP, enter:
   ```
   ip wccp 0 redirect in
   ```
   If your subscription includes SSL Manager, enter the following to turn on redirection for HTTPS:
   ```
   ip wccp 70 redirect in
   ```
   Run the command above for each protocol you want to support, but *only for the interface dedicated to inbound traffic*.

   For *outgoing* packets:

   ```
   ip wccp <service group ID> redirect out
   ```
   For example, to turn on redirection for HTTP, enter:
   ```
   ip wccp 0 redirect out
   ```
   If your subscription includes SSL Manager, enter the following to turn on redirection for HTTPS:
   ```
   ip wccp 70 redirect out
   ```
   Run the command above for each protocol you want to support, but *only for the interface dedicated to outbound traffic*.

3. Turn off redirection for packets received on the interface:

   ```
   ip wccp redirect exclude in
   ```

   Run the command above when dynamic or static bypass are enabled. With bypass enabled, the proxy redirects bypassed traffic to the Internet. This command prevents looping back of packets by stopping the router from redirecting this same bypassed traffic back to Websense Content Gateway. Run the command above for each interface connected to the proxy.

Disabling WCCP processing for a service group

If you need to disable WCCP processing for any reason, issue this command to turn off the WCCP feature:

```
no ip wccp <service group ID> password [0-7] <passwd>
```

Enabling WCCP 2.0 security on the router

If you are running WCCP 2.0 on your routers, you can enable security on the Websense Content Gateway node so that the proxy caching server and your routers can authenticate each other. You must enable security for each service group that the router supports individually. You cannot configure a router globally as you would Websense Content Gateway.

You enable the security option and provide the authentication password in Websense Content Manager.

The authentication password you specify must match the authentication password configured on the router for each service group being intercepted (HTTP, FTP, SOCKS). The following procedure provides an example of how to set an authentication password for different service groups on a WCCP 2.0-enabled router.

1. Telnet to the router and switch to Enable mode.
2. At the prompt, enter the following command to configure the router from the terminal:

    ```
    config t
    ```
3. If you defined a password when you enabled WCCP processing on the router, skip to step 4. Otherwise, enter the following command for each service group that the router intercepts:

    ```
    hostname(config)# ip wccp service_group password password
    ```

    where *hostname* is the host name of the router you are configuring, *service_group* is the service group ID (for example, 0 for HTTP), and *password* is the password you want to use to authenticate Websense Content Gateway. This password must match the password you specify in Websense Content Gateway configuration.

    > **Important**
    > Websense Content Gateway supports only one password. Therefore, you must use the same password for all service groups. Passwords longer than 8 characters are truncated.

4. Exit and save the router configuration.

Enabling WCCP in Websense Content Manager

Once you have configured your WCCP router to send traffic to the proxy server, you must also enable WCCP on Websense Content Gateway. You do this through the Websense Content Manager.

> **Important**
>
> Before you enable WCCP, make sure that your configuration meets the following requirements:
>
> ◆ The WCCP router is running the correct Cisco IOS release.
>
> ◆ If you are using several Websense Content Gateway nodes, determine whether you want the nodes to have management-only clustering or full clustering. See *WCCP load balancing*, page 45 and *Clusters*, page 61.
>
> ◆ Each proxy server must have the ARM enabled. Ensure that ARM is **On** on **Configure** > **My Proxy > Basic > General**.

Related topics:

Enabling WCCP 1.0, page 50
Enabling WCCP 2.0, page 51
Configuring WCCP routers, page 46
Enabling WCCP 2.0 security on the router, page 49

Enabling WCCP 1.0

1. Navigate to **Configure** > **My Proxy > Basic > General**.
2. In the **Features** table, click WCCP **On** in the **Networking** section.
3. Click **Apply**.
4. Navigate to **Configure** > **Networking** > **WCCP > General**.
5. Enable the **WCCP v1.0** option, and then click **Apply**.
6. Select the **WCCP v1.0** tab.
7. In the **WCCP Network Interface** field, enter the interface on the Websense Content Gateway system that receives traffic from the router.
8. In the **WCCP Router IP Address** field, enter the IP address of the router that sends traffic to Websense Content Gateway.
9. Click **Apply**.
10. Click **Restart** on **Configure** > **My Proxy > Basic > General**.

Enabling WCCP 2.0

1. Navigate to **Configure** > **My Proxy > Basic > General**.

2. In the **Features** table, click **WCCP On** in the **Networking** section.

3. Click **Apply**.

4. Navigate to the **Configure** > **Networking** > **WCCP > General** tab.

5. Enable the **WCCP v2.0** option, and then click **Apply**.

6. Select the **WCCP v2.0** tab.

7. In the **WCCP Network Interface** field, enter the interface on the Websense Content Gateway system that receives traffic from the router.

8. (*Optional*.) In the **Security** area, enable the **Security** option if you want Websense Content Gateway and your routers to authenticate each other.

   In the **Password** field, enter the password used for service group authentication with the router. The password must match the authentication password for the service group specified on the router and must be a maximum of eight characters long. See *Enabling WCCP processing for a service group*, page 48, step 1. See also *Enabling WCCP 2.0 security on the router*, page 49.

9. To run in multicast mode, enable the **Multicast** option in the **Multicast** area, and then enter the multicast IP address for WCCP in the **Multicast IP Address** field. The multicast IP address must match the multicast IP address specified on the router. See *Using multicast mode*, page 52.

10. To run in unicast mode, specify the number of WCCP routers that direct traffic to Websense Content Gateway in the **Number of WCCP Routers** field of the **Routers** area. Then specify the IP addresses of each router in the fields provided. For more information on unicast mode, see the Websense Content Gateway *Installation Guide*.

11. In the **Services Enable/Disable** section, make sure that the appropriate services are enabled so that the router is allowed to send HTTP, FTP, SOCKS, or DNS traffic to the proxy.

12. In the **Miscellaneous** section, enable the **Encapsulation** option to configure the proxy to send encapsulated returned (bypassed) packets to the router.

13. Click **Apply**.

14. Click **Restart** on **Configure** > **My Proxy > Basic > General**.

> ✔ **Note**
> To check that the router is sending traffic to the proxy,
> examine the statistics in the Websense Content Manager
> **Monitor** pane. For example, check that the **Objects
> Served** statistic in the **My Proxy > Summary** section
> increases.

> ✔ **Note**
> It may take more than 30 seconds for the router to report
> that a new proxy caching server has joined a service group.

Using multicast mode

To configure Websense Content Gateway to run in multicast mode, you must enable
multicast mode and specify the multicast IP address in Websense Content Manager.

In addition, you must set the multicast address on your routers for each service group
being intercepted (HTTP, FTP, DNS, and SOCKS). The following procedure provides
an example of how to set the multicast address for different service groups on a
WCCP 2.0-enabled router.

1. Telnet to the router and switch to Enable mode.

2. At the prompt, enter the following command to configure the router from the
   terminal:

   ```
   config t
   ```

3. At the prompt, enter the following command for each service group that the router
   intercepts:

   ```
   hostname(config)# ip wccp service_group group-address
   multicast_address
   ```

   where *hostname* is the host name of the router you are configuring,
   service_group is the service group ID (for example, 0 for HTTP), and
   multicast_address is the IP multicast address.

4. At the prompt, enter the following command to configure the network interface:

   ```
   interface interface_name
   ```

   where *interface_name* is the network interface on the router that is being
   intercepted and redirected.

5. At the prompt, enter the following command for each service group that the router
   intercepts:

   ```
   hostname(config-if)# ip wccp service_group group-listen
   ```

6. Exit and save the router configuration.

Using policy-based routing

Instead of the WCCP protocol, you can use the policy routing capabilities of a router to send traffic to Websense Content Gateway. WCCP or a Layer 4 switch are generally preferable to this configuration because policy-based routing has a performance impact on the router, and policy-based routing does not support load balancing or heartbeat messaging.

- ◆ All client Internet traffic is sent to a router that feeds Websense Content Gateway.
- ◆ The router sends port 80 (HTTP) traffic to the proxy and sends the remaining traffic to the next hop router.
- ◆ The ARM translates intercepted requests into Websense Content Gateway requests.
- ◆ Translated requests are sent to the proxy.
- ◆ Web objects to be served transparently are readdressed by the ARM on the return path to the client, so that the documents appear to have come from the origin server.

A Websense Content Gateway cluster with virtual IP failover adds reliability; if one node fails, another node can take up its transparency requests. See *Virtual IP failover*, page 65.
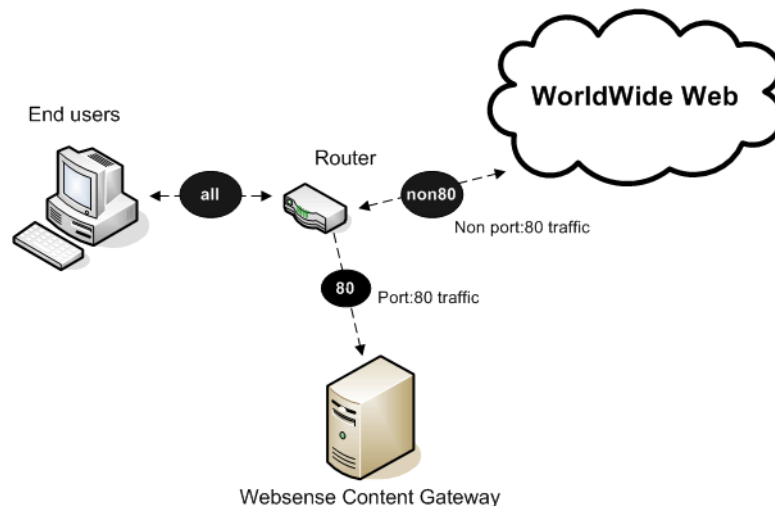
Using a router to filter HTTP requests

Using software-based routing

You can deploy Websense Content Gateway without adding routers or switches by using routing software on the Websense Content Gateway node. In this case, Websense Content Gateway is a software router and directs all traffic through the proxy machine. This solution can be useful in low-traffic situations, where the performance cost of using the proxy machine as a router is not high.

On UNIX systems, you can use the `routed` and `gated` daemons as a software-based routing solution. The `routed` daemon is a bundled part of all normal UNIX distributions. The `gated` daemon is an extensible commercial software package from the Merit GateD Consortium.

When you use routing software with Websense Content Gateway:

◆ All Internet traffic goes through Websense Content Gateway from machines behind it in the network.

◆ The routing software routes all non-transparent requests to the Internet; it routes port 80 HTTP requests to the proxy cache.

◆ The ARM translates intercepted requests into proxy requests.

◆ Translated requests are sent to the proxy.

◆ Web objects to be served transparently are readdressed by the ARM on the return path to the client, so that the objects appear to have come from the origin server.

> **Note**
>
> Although Websense Content Gateway machines can function as routers, they are not expressly designed to be routers. For reliability, you can use a Websense Content Gateway cluster with the virtual IP failover option. If one node fails, another cluster node takes over. See *Virtual IP failover*, page 65.) The Websense Content Gateway cluster failover mechanism is similar to the Hot Standby Router Protocol (HSRP).

Interception bypass

A small number of clients and servers do not work correctly with Web proxies. Some reasons include:

◆ Client software bugs (customized, non-commercial browsers).

◆ Server software bugs.

◆ Applications that send non-HTTP traffic over HTTP ports as a way of defeating security restrictions.

◆ Server IP authentication (the origin server limits access to a few client IP addresses, but the Websense Content Gateway IP address is different, so it cannot get access). This is not in frequent use because many ISPs dynamically allocate client IP dial-up addresses, and more secure cryptographic protocols are now more often used.

Web proxies are very common in corporate and Internet use, so interoperability problems are rare. However, Websense Content Gateway contains an adaptive learning module that recognizes interoperability problems caused by transparent proxy caching and automatically bypasses the traffic around the proxy server without operator intervention.

Websense Content Gateway follows 2 types of bypass rules:

◆ *Dynamic* (also called adaptive) bypass rules are generated dynamically if you configure Websense Content Gateway to bypass the cache when it detects non-HTTP traffic on port 80 or when it encounters certain HTTP errors. See *Dynamic bypass rules*, page 55.

◆ *Static* bypass rules must be manually configured in the **bypass.config** file. See *Static bypass rules*, page 56.

> ✔ **Note**
> Do not confuse bypass rules with client access control lists. Bypass rules are generated in response to interoperability problems. Client access control is simply restriction of the client IP addresses that can access the proxy cache, as described in *Controlling client access to the proxy cache*, page 101.

Dynamic bypass rules

When configured to do so, the proxy watches for protocol interoperability errors. As it detects errors, it configures the ARM to bypass the proxy for those clients and/or servers causing the errors.

In this way, the small number of clients or servers that do not operate correctly through proxies are auto-detected and routed around the proxy caching server so that they can continue to function (but without caching).

You can configure the proxy to dynamically bypass the cache for any of the following errors:

Error code	Description
N/A	Non-HTTP traffic on port 80
400	Bad Request
401	Unauthorized
403	Forbidden (authentication failed)
405	Method Not Allowed
406	Not Acceptable (access)
408	Request Timeout
500	Internal Server Error

For example, when Websense Content Gateway is configured to bypass on authentication failure (403 Forbidden), if any request to an origin server returns a 403 error, Websense Content Gateway generates a destination bypass rule for the origin server's IP address. All requests to that origin server are bypassed until you restart the proxy.

In another example, if the ARM detects that a client is sending a non-HTTP request on port 80 to a particular origin server, Websense Content Gateway generates a source/destination rule. All requests from that particular client to the origin server are bypassed; requests from other clients are not bypassed.

Bypass rules that are generated dynamically are purged after a Websense Content Gateway restart. If you want to preserve dynamically generated rules, you can save a snapshot of the current set of bypass rules. See *Viewing the current set of bypass rules*, page 57.

To prevent Websense Content Gateway from bypassing certain IP addresses dynamically, you can set dynamic deny bypass rules in the **bypass.config** file. Deny bypass rules can prevent the proxy from bypassing itself. For information about setting dynamic deny bypass rules, see *bypass.config*, page 258.

Related topics:

Setting dynamic bypass rules, page 56
Viewing dynamic bypass statistics, page 56

Setting dynamic bypass rules

By default, Websense Content Gateway is not configured to bypass the cache when it encounters HTTP errors or non-HTTP traffic on port 80. You must enable dynamic bypass rules by setting the appropriate options.

1. Navigate to **Configure** > **My Proxy > Basic > General**.
2. In the **Features** table, make sure that **ARM** is **On** in the **Networking** section.
3. Navigate to **Configure** > **Networking** > **ARM > Dynamic Bypass**.
4. Enable the **Dynamic Bypass** option.
5. In the **Behavior** section, select the dynamic bypass rules you want to use.
6. Click **Apply**.
7. Click **Restart** on the **Configure** > **My Proxy > Basic > General** tab.

Viewing dynamic bypass statistics

Websense Content Gateway tallies bypassed requests for each type of dynamic bypass trigger. For example, Websense Content Gateway counts all requests that are bypassed in response to a 401 error.

▶ Navigate to **Monitor** > **Networking** > **ARM**.

The statistics are displayed in the **HTTP Bypass Statistics** section of the table.

Static bypass rules

You can configure bypass rules to direct requests from certain clients or to particular origin servers around the proxy. Unlike dynamic bypass rules that are purged when you restart the proxy, these static bypass rules are saved in a configuration file.

For example, you might want client IP addresses that did not pay for a caching service to be steered around the cache, while paying clients can obtain the benefits of caching. Or you might want to remove some servers from caching lists because they do not want to have their pages cached.

You can configure 3 types of static bypass rules:

◆ Source bypass, in which Websense Content Gateway bypasses a particular source IP address or range of IP addresses. For example, you can use this solution to bypass clients who want to opt out of a caching solution.

◆ Destination bypass, in which Websense Content Gateway bypasses a particular destination IP address or range of IP addresses. For example, these could be origin servers that use IP authentication based on the client's real IP address. Destination bypass rules prevent Websense Content Gateway from caching an entire site. You will experience hit rate impacts if the site you bypass is popular.

◆ Source/destination pair bypass, in which Websense Content Gateway bypasses requests that originate from the specified source to the specified destination. For example, you could route around specific client-server pairs that experience broken IP authentication or out of band HTTP traffic problems when cached.

Source/destination bypass rules might be preferable to destination rules because they block a destination server only for those particular users that experience problems.

To configure static bypass rules, edit the **bypass.config** file (See *bypass.config*, page 258).

Viewing the current set of bypass rules

The ARM has a supporting utility called `print_bypass` that allows you to view the current dynamic and static bypass rules.

To view all current dynamic and static bypass rules:

1. Log on to a Websense Content Gateway node as the Websense Content Gateway administrator, and then change to the Websense Content Gateway **bin** directory (default location is in **/opt/WCG**).

2. Enter the following command at the prompt and press **Return**:

    `./print_bypass`

    All current static and dynamic bypass rules are displayed on screen. The rules are sorted by IP address. You can direct the output of `print_bypass` to a file and save it.

Configuring ARM security

To prevent unauthorized access to machines running Websense Content Gateway, you can configure the ARM to use an access control list employing administrator-specified rules to allow or deny other computers from communicating with the machine. This

enables you to effectively create a firewall in front of Websense Content Gateway, denying potentially malicious packets from reaching the TCP/IP stack on the machine. See *Controlling host access to the proxy server*, page 102.

Connection load shedding

The load shedding feature prevents client request overloads. When there are more client connections than the specified limit, the ARM forwards incoming requests directly to the origin server. The default client connection limit is 1 million connections.

1. Navigate to **Configure** > **Networking** > **Connection Management > Load Shedding**.
2. In the **Maximum Connections** field, specify the maximum number of client connections allowed before the ARM starts forwarding requests directly to the origin server.
3. Click **Apply**.
4. Click **Restart** on **Configure > My Proxy > Basic > General**.

Reducing DNS lookups

If you are running Websense Content Gateway in transparent proxy mode, you can enable the *Always Query Destination* option to reduce the number of DNS lookups and improve response time. When enabled, the Always Query Destination option configures the proxy to always obtain the original destination IP address of incoming requests from the ARM driver. Websense Content Gateway then uses that IP address to determine the origin server instead of doing a DNS lookup on the host name of the

request. Because the client already performed a DNS lookup, Websense Content Gateway does not have to.

✓ **Note**

It is recommended that you do not enable the Always Query Destination option if Websense Content Gateway is running in both explicit proxy caching mode and transparent proxy caching mode. See *How do you configure Websense Content Gateway to serve only transparent requests?*, page 358, for information about running Websense Content Gateway in transparent proxy caching mode only. In explicit proxy caching mode, the client does not perform a DNS lookup on the host name of the origin server, so the proxy must perform a DNS lookup.

In addition, do not enable the Always Query Destination option if you want domain names, rather than IP addresses, to be captured in the log server.

To enable the Always Query Destination option:

1. Open the **records.config** file located in the Websense Content Gateway **config** directory (default location is in **/opt/WCG**).

2. Edit the following variable:

Variable	Description
`proxy.config.arm.always_query_dest`	Set to 0 to disable the Always Query Destination option. Domain names are captured.
	Set to 1 to enable the Always Query Destination option. IP addresses are captured; domain names are not.

3. Save and close the file.

4. From the Websense Content Gateway **bin** directory, run `content_line -x` to apply the changes.

IP spoofing

The IP spoofing option configures Websense Content Gateway to use the IP address of the client when communicating with origin servers instead of its own IP address. As a result, HTTP requests appear to come from the client rather than from the proxy.

When you enable the IP spoofing option, the proxy uses the client IP address instead of its own in *all* origin server communications. This option affects non-transparent requests as well as transparent requests.

> **Important**
> Symmetric routing is necessary for the IP spoofing option to work. Make sure that all traffic from the origin server to the client goes through Websense Content Gateway.

> **Important**
> Do not use the IP spoofing option in transparent proxy mode if you are using a WCCP-enabled router.

1. Navigate to **Configure** > **Networking** > **ARM > General**. (If ARM is not a choice, ensure that ARM is **On** on **Configure > My Proxy > Basic > General**.)
2. Enable **IP Spoofing**.
3. Click **Apply**.
4. Click **Restart** on **Configure > My Proxy > Basic > General**.

6 | Clusters

Websense Content Gateway scales from a single node to multiple nodes that form a cluster, allowing you to improve system performance and reliability. The nodes in a cluster share configuration information and can form a single logical cache or Websense Content Gateway node.

Websense Content Gateway detects the addition and deletion of nodes in the cluster and can detect when a node is down. You can add or delete a node from a cluster at any time. When you remove a node from the cluster, Websense Content Gateway removes all references to the missing node.

When the *virtual IP failover* feature (see *Virtual IP failover*, page 65) is enabled, the live nodes in a cluster can assume a failed node's responsibilities.

Websense Content Gateway uses a proprietary protocol for clustering, which is multicast for node discovery and heartbeat, but unicast for all data exchange within the cluster.

Websense Content Gateway has two clustering modes:

◆ Management-only mode
◆ Full-clustering mode

Either mode of clustering requires a dedicated network interface for cluster communication.

> **Important**
> In a proxy hierarchy, the nodes in the cluster cannot be ICP siblings or be a mixture of HTTP parents and children; you must configure each node in a Websense Content Gateway cluster as a single node in the hierarchy, because they share a common configuration.

> Related topics:
>
> *Changing clustering mode*, page 62
> *Adding nodes to a cluster*, page 63
> *Deleting nodes from a cluster* , page 65
> *Virtual IP failover*, page 65

Management-only clustering

In management-only clustering mode, Websense Content Gateway cluster nodes share configuration information. You can administer all the nodes at the same time.

Websense Content Gateway uses a multicast management protocol to provide a single system image of your Websense Content Gateway cluster. Information about cluster membership, configuration, and exceptions is shared across all nodes, and the content_manager process automatically propagates configuration changes to all the nodes.

Full clustering

In full-clustering mode, in addition to sharing configuration information, a Websense Content Gateway cluster distributes its cache across its nodes into a single, virtual object store, rather than replicating the cache node by node. Websense Content Gateway can provide an aggregate cache size and can maximize the cache hit rate by storing objects only once across the entire cluster.

A fully clustered Websense Content Gateway maps objects to specific nodes in the cluster. When a node receives a request, it checks to see if the request is a hit somewhere in the cluster. If the request is a hit on a different node, the node handling the request fetches the object from the hit node and serves it to the client. Websense Content Gateway uses a proprietary communication protocol to fetch an object from other cluster nodes.

If a node fails or is shut down and removed, Websense Content Gateway removes references to the missing node on all nodes in the cluster. If virtual IP failover is enabled (see *Virtual IP failover*, page 65), requests destined for the missing node are handled by another node.

> **✓ Note**
>
> A second network interface is required for cluster communication.

Changing clustering mode

Clustering is usually configured when you install the proxy. You can, however, configure clustering through Websense Content Manager as well.

1. At the command line, enter the following:

   route add *multicast.group address*/**32 dev** *interface_name*
   where *interface_name* is the name of the interface used for cluster communication. For example:

   route add 224.0.1.37/32 dev eth1

   You need the multicast group address and interface name to configure through Websense Content Manager.

2. In Websense Content Manager, navigate to the **Configure > My Proxy > Basic > Clustering** page.

3. In the **Cluster Type** area, select the clustering mode:

   - Select **Single Node** if this node is not part of a cluster.

   - Select **Management Clustering** if cluster nodes share configuration information only.

   - Select **Full Cache Clustering** to share configuration information among cluster nodes and to distribute the cache across all the cluster nodes into a single, virtual object store.

4. For management or full cache clustering, enter the communication interface and the multicast group address, if necessary. These are the same interface and address you entered in Step 1.

5. Click **Apply**.

6. Click **Restart** on **Configure > My Proxy > Basic > General**.

✓ **Note**

Websense Content Gateway does not apply the clustering mode change to all the nodes in the cluster. You must change the clustering mode on each node individually.

Adding nodes to a cluster

Websense Content Gateway detects new Websense Content Gateway nodes on your network and adds them to the cluster, propagating the latest configuration information to the newcomer. This provides a convenient way to bootstrap new machines.

To connect a node to a Websense Content Gateway cluster, you need only install Websense Content Gateway software on the new node, making sure that the cluster

name and port assignments are the same as those of the existing cluster. Websense Content Gateway recognizes the new node.

> **Important**
>
> The nodes in a cluster must be homogeneous; each node must be on the same hardware platform, each must run the same version of the same operating system, and Websense Content Gateway must be installed in the same directory (including path) on each node. For example, the default installation directory for Websense Content Gateway is **/opt/WCG**. If the proxy is installed elsewhere on one node in the cluster, either that node should have the proxy reinstalled at the default location, or all nodes must have the proxy installed in the alternate location.

1. Install the appropriate hardware and connect it to your network. (Consult the hardware documentation for hardware installation instructions.)

2. Install the Websense Content Gateway software using the appropriate procedure for installing a cluster node. See the Websense Content Gateway *Installation Guide*. During the installation procedure, make sure that the following is true:

   ■ The cluster name that you assign to the new node is the same as the cluster name for the existing cluster.

   ■ The port assignments for the new node are the same as the port assignments used by the other nodes in the cluster.

   ■ You have added multicast addresses and multicast route settings.

3. Restart Websense Content Gateway. See *Starting and stopping Websense Content Gateway*, page 9.

If you have an existing Websense Content Gateway installation and you want to add that server to the cluster, you do *not* have to reinstall the Websense Content Gateway on the node. Instead, you can edit configuration variables on the existing Websense Content Gateway node.

1. Open the **records.config** file located in the **config** directory (default location is /**opt/WCG/config**) on the node you want to add to the cluster.

2. Edit the following variables:

Variable	Description
proxy.config.cluster.type	Specify the clustering mode used by the other nodes in the cluster: 1 for full-clustering mode 2 for management-only mode
proxy.config.proxy_name	Specify the name of the Websense Content Gateway cluster. All nodes in a cluster must use the same name.

Variable	Description
proxy.config.cluster.mc_group_addr	Specify the multicast address for cluster communications. All nodes in a cluster must use the same multicast address.
proxy.config.cluster.rsport	Specify the reliable service port. The reliable service port is used to send data between the nodes in the cluster. All nodes in a cluster must use the same reliable service port. The default value is 8087.
proxy.config.cluster.mcport	Specify the multicast port. The multicast port is used for node identification. All nodes in a cluster must use the same multicast port. The default port number is 8088.
proxy.config.cluster.ethernet_interface	Specify the network interface for cluster traffic. All nodes in a cluster must use the same network interface.

3. Save and close the file.
4. Restart Websense Content Gateway (**/opt/WCG/WCGAdmin restart**).
5. To change from Full mode to Management mode or vice versa:

    a. Access Websense Content Gateway Manager.

    b. Navigate to **Configure > My Proxy > Basic > Clustering**.

    c. In the **Cluster Type** area, select the appropriate type (**Full** or **Management**).

    d. Click **Apply**.

    e. Click **Restart** on **Configure > My Proxy > Basic > General**.

Deleting nodes from a cluster

On the node you want to remove from the cluster:

1. Navigate to **Configure > My Proxy > Basic > Clustering**.
2. In the **Cluster Type** area, select **Single Node**.
3. Click **Apply**.
4. Click **Restart** on **Configure > My Proxy > Basic > General**.

Virtual IP failover

Through the virtual IP failover feature, Websense Content Gateway maintains a pool of virtual IP addresses that it assigns to the nodes in the cluster as necessary. These

addresses are virtual only in the sense that they are not tied to a specific machine; Websense Content Gateway can assign them to any of its nodes. To the outside world, these virtual IP addresses are the addresses of Websense Content Gateway servers.

Virtual IP failover assures that if a node in the cluster fails, other nodes can assume the failed node's responsibilities. Websense Content Gateway handles virtual IP failover in the following ways:

◆ The content_manager process maintains cluster communication. Nodes automatically exchange statistics and configuration information through multicast communication. If multicast heartbeats are not received from one of the cluster nodes, the other nodes recognize it as unavailable.

◆ The content_manager process reassigns the IP addresses of the failed node to the remaining operational nodes within approximately 30 seconds, so that service can continue without interruption.

◆ The IP addresses are assigned to new network interfaces, and the new assignment is broadcast to the local network. The IP reassignment is done through a process called *ARP rebinding*.

What are virtual IP addresses?

Virtual IP addresses are IP addresses that are not tethered to particular machines. Thus, they can rotate among nodes in a Websense Content Gateway cluster.

It is common for a single machine to represent multiple IP addresses on the same subnet. This machine would have a primary or real IP address bound to its interface card and also serve many more virtual addresses.

You can set up your user base to use a DNS round-robin pointing at virtual IP addresses, as opposed to using the real IP addresses of the Websense Content Gateway machines.

Because virtual IP addresses are not bound to machines, a Websense Content Gateway cluster can take addresses from inactive nodes and distribute those addresses among the remaining live nodes.

Using a proprietary management protocol, Websense Content Gateway nodes communicate their status with their peers. If a node fails, its peers notice the failure and negotiate which of the remaining nodes will mask the fault by taking over the failed node's virtual interface.

Related topics:

Enabling and disabling virtual IP addressing

1. Navigate to **Configure > My Proxy > Basic > General**.

2. Under the Networking section in the Features table, select **On** or **Off** for **Virtual IP** to enable or disable Virtual IP addressing.

3. Click **Apply**.

4. Click **Restart** on **Configure > My Proxy > Basic > General** to restart Websense Content Gateway on all the nodes in the cluster.

Adding and editing virtual IP addresses

Virtual IP addresses must be pre-reserved like all IP addresses before they can be assigned to Websense Content Gateway.

> ⚠ **Warning**
>
> Incorrect IP addressing can disable your system. Make sure you understand how virtual IP addresses work before changing them.

1. Navigate to **Configure** > **Networking** > **Virtual IP**.

   The **Virtual IP Addresses** area displays the virtual IP addresses managed by Websense Content Gateway.

   > ✓ **Note**
   >
   > The Virtual IP button is displayed only if you have enabled the Virtual IP option in the Features table on **Configure > My Proxy > Basic > General**.

2. Click **Edit File** to add new or edit existing virtual IP addresses.

3. To edit a virtual IP address, select it from the table at the top of the page, edit the fields provided, and then click **Set**.

   To delete the selected IP address, click **Clear Fields**.

   To add a virtual IP address, specify the virtual IP address, the Ethernet interface, and the Subinterface in the fields provided, and then click **Add**.

4. Click **Apply**, and then click **Close**.

5. Click **Restart** on **Configure > My Proxy > Basic > General**.

7 | Hierarchical Caching

Websense Content Gateway can participate in cache hierarchies, where requests not fulfilled in one cache can be routed to other regional caches, taking advantage of the contents and proximity of nearby caches.

A cache hierarchy consists of levels of caches that communicate with each other. Websense Content Gateway supports several types of cache hierarchies. All cache hierarchies recognize the concept of *parent* and *child*. A parent cache is a cache higher up in the hierarchy, to which the proxy can forward requests. A child cache is a cache for which the proxy is a parent.

Websense Content Gateway can be a member of the following cache hierarchies:

◆ An HTTP cache hierarchy (described in *HTTP cache hierarchies*, page 69)
◆ An ICP (Internet Cache Protocol) cache hierarchy (described in *ICP cache hierarchies*, page 71)

HTTP cache hierarchies

In an HTTP cache hierarchy, if a Websense Content Gateway node cannot find a requested object in its cache, it can search a parent cache—which itself can search other caches—before resorting to retrieving the object from the origin server.

You can configure a Websense Content Gateway node to use one or more HTTP parent caches, so that if one parent is unavailable, another parent can service requests. This is called parent failover and is described in *Parent failover*, page 70.

> **Note**
>
> If you do not want all requests to go to the parent cache, you can configure the proxy to route certain requests directly to the origin server (for example, requests that contain specific URLs) by setting parent proxy rules in the **parent.config** configuration file (described in *parent.config*, page 280).

> **Note**
>
> If the request is a cache miss on the parent, the parent retrieves the content from the origin server (or from another cache, depending on the parent's configuration). The parent caches the content and then sends a copy to the proxy (its child), where it is cached and served to the client.

Parent failover

When you configure the proxy to use more than one parent cache, the proxy detects when a parent is not available and sends missed requests to another parent cache. If you specify more than two parent caches, the order in which the parent caches are queried depends upon the parent proxy rules configured in the parent configuration file described in *parent.config*, page 280. By default, the parent caches are queried in the order in which they are listed in the configuration file.

Configuring Websense Content Gateway to use an HTTP parent cache

1. On the **Configure > Content Routing > Hierarchies > Parenting** page, enable **Parent Proxy**.
2. Click **Edit File** to open the configuration file editor for the *parent.config* file.
3. Enter information in the fields provided, and then click **Add**. All the fields are described in Appendix B: *Configuration Options*.
4. Click **Apply**, and then click **Close**.

5. On the **Parenting** tab, click **Apply** to save your configuration.

> **Important**
>
> Perform this procedure on the *child* proxy. Do not make any changes on the parent.

ICP cache hierarchies

The Internet Cache Protocol (ICP) is a protocol used by proxy caches to exchange information about their content. ICP query messages ask other caches if they are storing a particular URL. ICP response messages reply with a hit (the URL is stored) or miss (the URL is not stored) answer.

A cache exchanges ICP messages only with specific ICP peers, which are neighboring caches that can receive ICP messages. An ICP peer can be a sibling cache, which is at the same level in the hierarchy, or a parent cache, which is one level up in the hierarchy.

If Websense Content Gateway has ICP caching enabled, it sends out ICP queries to its ICP peers in the event of a cache miss on an HTTP request. If there are no hits and parents exist, a parent is selected using a round-robin policy. If no ICP parents exist, the proxy forwards the request to its HTTP parents. If there are no HTTP parent caches, the proxy forwards the request to the origin server.

> **Note**
>
> If Websense Content Gateway receives a hit message from an ICP peer, the proxy sends the HTTP request to that peer.
>
> Under some circumstances, what appears to be a cache hit is really a miss; this happens when the original HTTP request contains header information that is not communicated by the ICP query. For example, the hit might not be the requested alternate.
>
> If an ICP hit turns out to be a miss, the proxy forwards the request to either its HTTP parent caches or to the origin server.

Configuring Websense Content Gateway to use an ICP cache hierarchy

1. Navigate to the **Configure > Content Routing > Hierarchies > ICP Peering** tab.
2. In the ICP mode area, select:
   - **Only Receive Queries** to configure Websense Content Gateway to only receive ICP queries from other ICP peers.

- **Send/Receive Queries** to configure Websense Content Gateway to both send and receive ICP queries.

- **Disabled** to configure Websense Content Gateway to neither send nor receive ICP queries.

3. Enter the name of the interface to use for ICP messages. This field is populated automatically, but if the machine is running with more than one interface, you can specify a different one here.

4. Enter the port to use for ICP messages. The default is 3130.

5. Enable the **ICP Multicast** option to send ICP messages through multicast if your proxy server has a multicast channel connection to its ICP peers.

6. Enter the timeout for ICP queries. The default is 2 seconds.

7. Click **Edit File** to open the configuration file editor for the **icp.config** file.

8. Enter information in the fields provided, and then click **Add**. All the fields are described in Appendix B: *Configuration Options*.

9. Click **Apply**, and then click **Close**.

10. On the **ICP Peering** tab, click **Apply** to save your configuration.

8 | Configuring the Cache

The cache consists of a high-speed object database called the *object store*. The object store indexes objects according to URLs and associated headers, enabling Websense Content Gateway to store, retrieve, and serve Web pages, and also parts of Web pages, providing optimum bandwidth savings. Using object management, the object store can cache alternate versions of the same object, varying on spoken language or encoding type, and can store small and large documents, minimizing wasted space. When the cache is full, Websense Content Gateway removes stale data.

Websense Content Gateway can tolerate disk failures on any cache disk. If the disk fails, Websense Content Gateway marks the disk as corrupt and continues using the remaining disks. An alarm is sent to Websense Content Manager, indicating which disk failed. If all cache disks fail, Websense Content Gateway goes into proxy-only mode.

You can perform the following cache configuration tasks:

◆ Change the total amount of disk space allocated to the cache. See *Changing cache capacity*, page 76.

◆ Specify a size limit for objects allows in the cache. See *Configuring cache object size limit*, page 77

◆ Delete all data in the cache. See *Clearing the cache*, page 77.

◆ Change the size of the RAM cache. See *Changing the size of the RAM cache*, page 78.

RAM cache

Websense Content Gateway maintains a small RAM cache of popular objects. This RAM cache serves the most popular objects as fast as possible and reduces load on disks, especially during temporary traffic peaks. You can configure the RAM cache size. See *Changing the size of the RAM cache*, page 78.

Changing cache capacity

You can increase or reduce the total amount of disk space allocated to the cache *without* clearing the content.

> ✔ **Note**
> You can check the size of the cache in bytes by specifying the command `content_line -r proxy.process.cache.bytes_total`.

Increasing cache capacity

To increase the total disk space allocated to the cache on existing disks or add new disks to a Websense Content Gateway node:

1. Stop Websense Content Gateway. See *Starting and stopping Websense Content Gateway*, page 9.
2. Add hardware, if necessary.
3. Edit the **storage.config** file located in the Websense Content Gateway **config** directory (default location is **/opt/WCG/config**) to increase the amount of disk space allocated to the cache on existing disks or to describe the new hardware you are adding. See *storage.config*, page 336.

   If you add a new disk, you must edit the **/etc/rc.d/init.d/content_gateway** file to add a raw disk binding. The raw disk binding must have Websense user permissions. Instructions on adding a raw disk binding are located in the **storage.config** file.
4. Restart Websense Content Gateway.

Reducing cache capacity

You can reduce the total amount of disk space allocated to the cache on an existing disk or remove disks from a Websense Content Gateway node.

1. Stop Websense Content Gateway.
2. Remove hardware, if necessary.
3. Edit the **storage.config** file to reduce the amount of disk space allocated to the cache on existing disks or to delete the reference to the hardware you are removing. See *storage.config*, page 336.
4. If you remove a disk, you must edit the **/etc/rc.d/init.d/content_gateway** file to remove the raw disk binding for the disk.

5. Restart Websense Content Gateway.

> **Important**
>
> In the **storage.config** file, a formatted or raw disk must be at least 128 MB.

Configuring cache object size limit

By default, Websense Content Gateway allows objects of any size in the cache. You can change the default behavior and specify a size limit for objects in the cache.

1. Select **Configure** > **Subsystems** > **Cache > General**.
2. In the **Maximum Object Size** field, enter the maximum size allowed (in bytes) for objects in the cache. Enter 0 (zero) if you do not want to have a size limit.
3. Click **Apply**.

Clearing the cache

When you clear the cache, you remove all data from the entire cache, which includes the data in the host database. Clear the cache before performing certain cache configuration tasks, such as partitioning.

> **Note**
>
> You cannot clear the cache when Websense Content Gateway is running.

1. Stop Websense Content Gateway.See *Starting and stopping Websense Content Gateway*, page 9.
2. Enter the following command to clear the cache:

```
content_gateway -Cclear
```

> **Warning**
>
> The clear command deletes all data in the object store and the host database. Websense Content Gateway does *not* prompt you to confirm the deletion.

3. Restart Websense Content Gateway.

Changing the size of the RAM cache

Websense Content Gateway provides a dedicated RAM cache for fast retrieval of popular small objects. The default RAM cache size is calculated based on the number and size of the cache partitions you have configured. You can increase the RAM cache size for better cache hit performance.

> **Warning**
>
> If you increase the size of the RAM cache and observe a decrease in Websense Content Gateway performance (such as increased latencies), the operating system might require more memory for network resources. Return the RAM cache size to its previous value.

> **Note**
>
> If you have partitioned your cache according to protocol or hosts, the size of the RAM cache for each partition is proportional to the size of that partition.

1. Select **Configure** > **Subsystems** > **Cache > General**.
2. In the **Ram Cache Size** field, enter the amount of space (in megabytes) you want to allocate to the RAM cache.

> **Note**
>
> The default value of -1 means that the RAM cache is automatically sized at approximately one MB per GB of disk.

3. Click **Apply**.
4. Click **Restart** on **Configure > My Proxy > Basic > General**.

9 | Monitoring Traffic

Websense Content Gateway provides the following tools to monitor system performance and analyze network traffic:

◆ Websense Content Manager provides statistics that show Websense Content Gateway performance and network traffic information. See *Viewing statistics*, page 79. The command-line interface provides an alternative method of viewing this information. See *Viewing statistics from the command line*, page 83.

◆ Websense Content Manager presents alarms that signal any detected failure conditions. See *Working with alarms*, page 84.

◆ MRTG (Multi Router Traffic Grapher) provides a variety of graphs that show historical Websense Content Gateway performance and network traffic information. See *Using MRTG*, page 85.

◆ SNMP (Simple Network Management Protocol) support lets you monitor and manage Websense Content Gateway through SNMP network management facilities. See *Using SNMP*, page 86.

◆ Create reports through SSL Manager to see the status of certificate authorities and incidents. See *Creating reports with SSL Manager*, page 87.

Viewing statistics

Use Websense Content Manager to collect and interpret statistics about Websense Content Gateway performance and Web traffic. View statistics using Monitor mode.

Starting Monitor mode

1. Open your Web browser.

   Websense Content Manager requires Java and JavaScript; be sure to enable Java and JavaScript in your browser.

2. Enter the following location in your browser:

   `https://nodename:adminport`

   where *nodename* is the name of the Websense Content Gateway node and *adminport* is the number assigned to the Websense Content Manager port.

3. If necessary, log on to Websense Content Manager with the administrator ID and password, or use your user account. The administrator ID and password are set during installation. You can change the ID and password, as well as create and modify user accounts. For more information, see *Controlling access to Websense Content Manager*, page 103.

Websense Content Gateway displays the **Monitor** tab (shown in Figure 3).

Figure 3   Monitor tab

Using Monitor mode

In Monitor mode, Websense Content Manager displays a series of buttons on the left of the display. Click a button to view its statistics.

All the statistics displayed in Monitor mode are described in detail in *Statistics*, page 173.

My Proxy

Click **My Proxy** to see statistics about your proxy.

◆ Click **Summary** to see a concise view of your Websense Content Gateway system. The top half of the page contains information about the features in your Websense Content Gateway or Websense Security Gateway subscription, including the expiration date. The bottom part of the page contains statistics on proxy nodes, displaying all cluster nodes by name and tracking essential statistics for each node. If you want to display detailed information about a particular node in a cluster, click the node's name in the Summary table, and then click one of the other buttons on the **Monitor** tab.

◆ Click **Node** to see information about the selected node. You can see if the node is active or inactive, the date and time that the content_gateway process was started, cache performance information (document hit rate, bandwidth savings, and what percentage of the cache is currently free), the number of client and server connections currently open, and the number of transfers currently in

progress. You can also see name resolution information, such as the host database hit rate and the number of DNS lookups per second.

> ✔ **Note**
> If the node is part of a cluster, two sets of statistics are shown: information about the single node and information showing an average value for all nodes in the cluster. Click the name of a statistic to display the information in graphical format.

◆ Click **Graphs** to view the same statistics displayed on the **Node** page (cache performance, current connections and transfers, network, and name resolution) in graphical format. You can display multiple statistics in one graph.

To display a particular statistic in graphical format, click the box next to the name of the graph, and then click **Graph**. To display multiple statistics in one graph, click the box next to the name of each graph you want to display, and then click **Graph**.

◆ Click **Alarms** to view the alarms that Websense Content Gateway has signaled. See *Working with alarms*, page 84.

Protocols

The Protocols button provides information about HTTP and FTP transactions.

◆ Click **HTTP** to see information about HTTP transactions and speeds (such as cache misses, cache hits, connection errors, aborted transactions) and client and server connection information. Also see information about FTP requests from HTTP clients, such as the number of open FTP server connections, the number of successful and unsuccessful PASV and PORT connections, and the number of cache lookups, hits, and misses.

◆ Click **FTP** to see information about FTP requests from FTP clients. For example, see the number of open FTP client and FTP server connections, the number of file hits and misses, change directory hits and misses, and list directory hits and misses.

> ✔ **Note**
> The **FTP** button appears only if you have enabled FTP processing in the **Features** table under the **Configure > My Proxy > Basic** tab.

Content Routing

Click **Content Routing** to see ICP statistics that include information about queries originating from the Websense Content Gateway node and from ICP peers (parents and siblings).

Security button

The Security button provides information about ARM, proxy authentication, and SOCKS server connections:

◆ Click **ARM Security** to see the number of dropped TCP and UDP connections.

◆ Click **LDAP** to see the number of LDAP cache hits and misses, and the number of LDAP authentication server errors and unsuccessful authentication attempts. The LDAP button appears only if you have enabled the LDAP option in the Features table on the **Configure > My Proxy > Basic > General** tab.

◆ Click **NTLM** to see the number of NTLM cache hits and misses, and the number of NTLM authentication server errors and unsuccessful authentication attempts. The NTLM button appears only if you have enabled the NTLM option in the Features table on the **Configure > My Proxy > Basic > General** tab.

◆ Click **SOCKS** to see the number of successful and unsuccessful connections to the SOCKS server and the number of connections currently in progress. The SOCKS button appears only if you have enabled the SOCKS option in the Features table on the **Configure > My Proxy > Basic > General** tab.

Subsystems

The Subsystems button provides information about the proxy cache, clusters, and event logging:

◆ Click **Cache** to see information about the proxy cache. See how much space in the cache is currently being used, the total size of the cache in gigabytes, the total size of the RAM cache in bytes, the number of RAM cache hits and misses, and the number of cache lookups, object reads, writes, updates, and removes.

◆ Click **Clustering** to see the number of nodes in the cluster, the total number of cluster operations, the number of bytes read and written to all the nodes in the cluster, and the current number of open connections in the cluster.

◆ Click **Logging** to see the number of log files currently open, the amount of space currently being used for log files, the number of access events and error events logged, and the number of access events skipped.

Networking

The Networking button provides information about system network configuration, the ARM, WCCP routers, domain name resolution, and virtual IP addressing.

◆ Click **System** to see system network configuration, including the host name assigned to the proxy machine and the default gateway, search domain, and DNS servers that the proxy machine uses.

◆ Click **ARM** to see information about Network Address Translation and dynamic bypass. The ARM button appears only if you have enabled ARM in the Features table under the **Configure > My Proxy > Basic** tab.

◆ Click **WCCP** to see WCCP version 1.0 or 2.0 statistics that include information about the routers being used, the number of active nodes, the leader's IP address, and whether WCCP is currently enabled on the Websense Content Gateway node.

The WCCP button appears only if you have enabled WCCP in the Features table on the **Configure > My Proxy > Basic > General** tab.

◆ Click **DNS Resolver** to see the total number of lookups and hits in the host database, and the average lookup time, the total number of lookups, and the number of successful lookups in the DNS server.

◆ Click **Virtual IP Address** to see the current virtual IP address mappings. The Virtual IP Address button appears only if you have enabled the Virtual IP option in the Features table on the **Configure > My Proxy > Basic > General** tab.

MRTG

Displays MRTG graphs. See *Using MRTG*, page 85.

Viewing statistics from the command line

You can use the command-line interface to view statistics about Websense Content Manager performance and Web traffic.

You can also configure, stop, and restart the Websense Content Gateway system from the command line. See *Command-line interface*, page 94, and *Websense Content Gateway commands*, page 191.

To view specific information about a Websense Content Gateway node or cluster, specify the variable that corresponds to the desired statistic.

1. Become root:

   su

2. Log on to a node as the Websense Content Gateway administrator.

3. From the Websense Content Gateway **bin** directory (default location is **/opt/WCG/bin**), enter the following command:

   content_line -r *variable*

where *variable* is the variable that represents the information you want. For a list of the variables you can specify, see *Websense Content Gateway variables*, page 192.

For example, the following command displays the document hit rate for the node:

   content_line -r proxy.node.http.cache_hit_ratio

> ✓ **Note**
>
> If the Websense Content Gateway **bin** directory is not in your path, prepend the command with ./ (for example, ./content_line -r *variable*).

Working with alarms

Websense Content Gateway signals an alarm when it detects a problem (for example, if the space allocated to event logs is full or if it cannot write to a configuration file).

Navigate to **Monitor > My Proxy > Alarms** to see a listing of current alarms, as shown below.

The Alarm! (pending) bar appears at the top of the display when alarms exist.
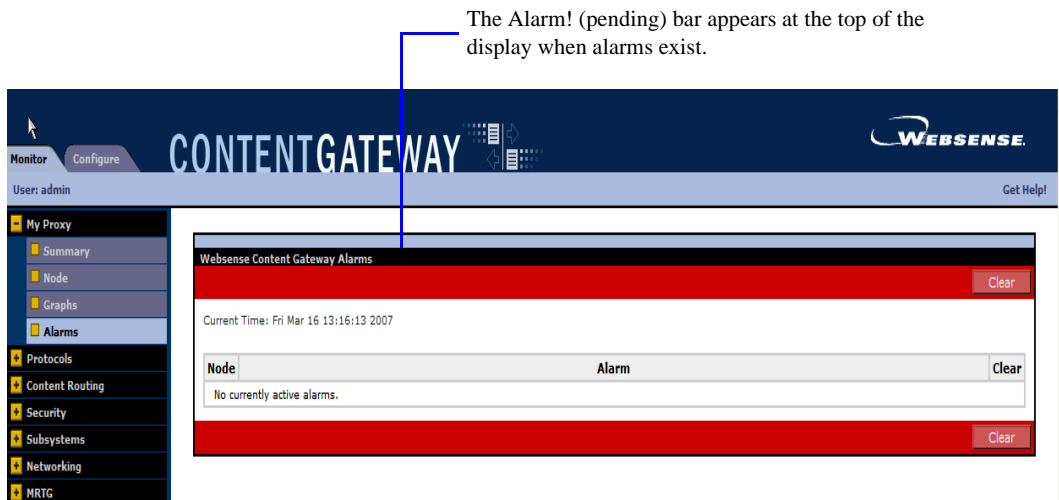
Alarms in Websense Content Manager

Clearing alarms

After you have read an alarm message, you can click **Clear** in the alarm message window to dismiss the alarm. *Alarm messages*, page 347, provides a description of some of the alarm messages that Websense Content Gateway provides.

> **Important**
>
> Clicking **Clear** only dismisses alarm messages; it does not actually resolve the cause of the alarms.

Configuring Websense Content Gateway to email alarms

1. Navigate to the **Configure > My Proxy > Basic > General** tab.
2. In the **Alarm eMail** field, enter the email address to which you want to send alarms. Be sure to use the full mail address including @ notation, for example:

   `receivername@company.com`
3. Click **Apply**.

Using a script file for alarms

Alarm messages are built into Websense Content Gateway; you cannot change them. However, you can write a script file to execute certain actions when an alarm is signaled.

A sample script file named **example_alarm_bin.sh** is provided in the **bin** directory (default location is **/opt/WCG/bin**). You can modify this file.

Using MRTG

The MRTG (Multi Router Traffic Grapher) tool lets you monitor Websense Content Gateway performance and analyze network traffic. MRTG provides graphs that show information about virtual memory usage, client connections, cache hit and miss rates, and so on. The information provided is recorded from the time that Websense Content Gateway was started. MRTG uses 5-minute intervals to formulate the statistics and provides historical information.

Access MRTG from the Monitor tab in Websense Content Manager.

> **Important**
> To run MRTG in UNIX, you must have Perl version 5.005 or later installed on your Websense Content Gateway system.

1. If your Websense Content Gateway node is in a cluster, select the node whose statistics you want to view from the **Monitor > My Proxy > Summary** display.
2. On the **Monitor** tab, click **MRTG**.
3. Click **Overview** to see a subset of the graphs available in MRTG.

   Click **Daily** to see statistics for the current day.

   Click **Weekly** to see statistics for the current week.

   Click **Monthly** to see statistics for the current month.

   Click **Yearly** to see statistics for the current year.
4. Wait at least 15 minutes before looking at the graphs. It takes several 5-minute sample intervals for MRTG to initialize statistics.

If MRTG has not been configured, the system displays a message indicating that MRTG is not available. To configure MRTG:

1. Make sure Perl 5.005 is installed on your system.
2. At the command prompt, type

   `perl ./pathfix.pl 'which perl'`

   to ensure that the perl binary is in your PATH.
3. Change to the Websense Content Gateway **bin** directory.

4. Modify the MRTG update interval by typing the following at the command prompt:

    ```
    ./update_mrtg;sleep 5;./update_mrtg;sleep 5;
    ```

    By default, an MRTG update interval is set to 15 minutes. This command sets the update to 5 minutes.

5. Start the MRTG cron updates:

    ```
    ./mrtgcron start
    ```

6. Wait about 15 minutes before accessing MRTG from the Websense Content Manager.

> ✔ **Note**
> To stop MRTG cron updates, type the command
> `./mrtgcron stop`.

Using SNMP

The Simple Network Management Protocol (SNMP) is a standard protocol used for network management. SNMP agents collect and store management information in Management Information Bases (MIBs), and SNMP managers can probe the agents for this information. In addition, SNMP agents can send alarms and alerts called *SNMP traps* to the SNMP manager to warn of any problems.

The Websense Content Gateway SNMP agent supports access to MIB-2, a well-known standard MIB. The Websense Content Gateway SNMP agent supports access to two MIBs: MIB-2 (a standard MIB) and the Websense Content Gateway MIB. Descriptions of the Websense Content Gateway MIB variables are provided in the **websense-wcg-mib.my** file in the **/opt/WCG/config/mibs** directory. The WCG MIB contains both node-specific and cluster-wide information.

To use SNMP on your Websense Content Gateway system, you must:

◆ Control MIB access to specific hosts (see *Controlling MIB access*, page 86).

◆ Configure Websense Content Gateway to send SNMP traps (see *Configuring SNMP trap destinations*, page 87).

◆ Enable the Websense Content Gateway SNMP agent (see *Enabling SNMP*, page 87).

Controlling MIB access

By default, read-only access to the Websense Content Gateway MIBs is granted to any host that makes SNMP requests using the community string `public`. Configure your Websense Content Gateway system to control MIB access so that only certain hosts can access SNMP information.

To configure Websense Content Gateway to control MIB access, edit the **snmpd.cnf** file located in the Websense Content Gateway **config** directory (default location is **/opt/WCG/config**). See *snmpd.cnf*, page 331.

Configuring SNMP trap destinations

To configure SNMP trap destinations, edit the **snmpd.cnf** file located in the Websense Content Gateway **config** directory (default location is **/opt/WCG/config**). See *snmpd.cnf*, page 331.

Enabling SNMP

You must enable the Websense Content Gateway SNMP agent so that SNMP managers can access the MIBs and gather information.

1. Navigate to the **Configure > My Proxy > Basic > General** page.
2. Click SNMP **On** in the Features table.
3. Click **Apply**.

Creating reports with SSL Manager

If your subscription includes SSL Manager, you can request a report detailing the status of certificate authorities (see *Certificate Authorities*, page 87) or listing incidents (see *Incidents*, page 88). Reports can be either in HTML or comma-separated format. The comma-separate reports appears as Excel spreadsheets in SSL Manager.

Certificate Authorities

1. Navigate to the **Monitor > SSL > Reports > Certificate Authorities** tab.
2. Select whether the format of the report should be in HTML or comma-separated (CSV) format. If you select comma-separated, the report is created in an Excel spreadsheet.
3. Specify the time period the report should cover. You can specify
   a. a number of days
   b. a date range
   c. the period since SSL Manager was deployed
4. Indicate the sort order for the report.
   a. Listing authorities by date
   b. Listing OCSP good responses first
   c. Listing OCSP bad responses first

   See *Keeping revocation information up to date*, page 136.

HTML output looks like this:

The same report in comma-separated format appears as follows:

Incidents

1. Navigate to the **Monitor > SSL > Reports > Incidents** tab.
2. Select HTML or comma-separated (CSV) format. If you select comma-separated, the report is created in an Excel spreadsheet.
3. Specify the time period the report should cover. You can specify
   a. a number of days
   b. a date range
   c. the period since SSL Manager was deployed
4. Indicate the sort order for the report.
   a. Listing incidents by date
   b. Listing incidents by URL
   c. Listing the number of times each incident occurred

   See *Managing Web HTTPS site access*, page 138.

HTML output looks like this:

The same report in comma-separated format appears as follows:

10 | Configuring the System

Websense Content Gateway provides several options for configuring the system:

◆ *Websense Content Manager*, page 91
◆ *Command-line interface*, page 94
◆ *Configuration files*, page 95
◆ *Saving and restoring configurations*, page 96

We recommend that you restart Websense Content Gateway any time you make configuration changes.

Websense Content Manager

Use Configure mode to view and change your Websense Content Gateway configuration.

> ✔ **Note**
> Certain options can be changed only by editing configuration variables either in the **records.config** file or from the command-line interface. See *Command-line interface*, page 94 and *Configuration files*, page 95.

Starting Configure mode

1. Open your Web browser.

   Websense Content Manager requires Java and JavaScript; be sure to enable Java and JavaScript in your browser.

2. Enter the following location in your browser:

   `https://nodename:adminport`

   where `nodename` is the name of the node and `adminport` is the number assigned to the Websense Content Manager port.

3. If necessary, log on to Websense Content Manager with the administrator ID and password, or use your user account. The administrator ID and password are set during product installation. You can change the ID and password, as well as create and modify user accounts. For more information, refer to *Controlling access to Websense Content Manager*, page 103.

   Websense Content Manager starts by default in Monitor mode.

4. Click the **Configure** tab to display the Configure mode buttons (shown in Figure 4).

Shows the current user logged on to Websense Content Gateway.

Click to display the Configure buttons.

Click a tab to display more options.

Click here to display the online Help system.

Click a button to display its configuration options.

Click the Apply button to save the configuration changes on the current tab.
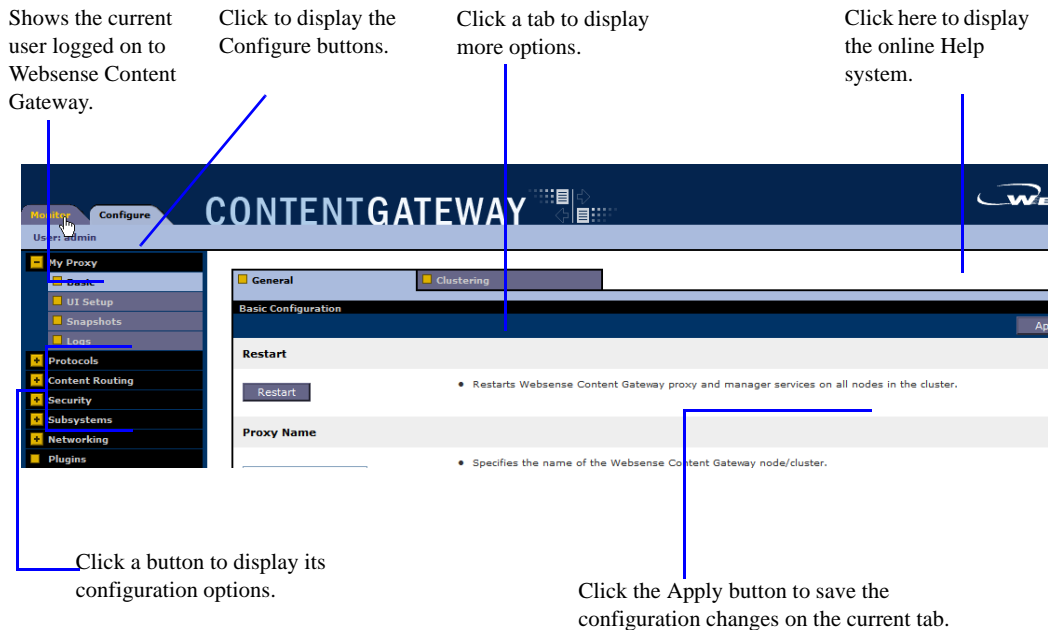
Figure 4 Configure tab

Using Configure mode

In Configure mode, Websense Content Manager displays a series of buttons. Each button represents a group of configuration options.

All the configuration options available in Configure mode are described in *Configuration Options*.

My Proxy

◆ Click **Basic** to restart the proxy and manager services (you need to restart after changing certain configuration options), identify the name of the Websense Content Gateway node, set alarm email, and enable or disable features (such as FTP processing, proxy authentication, ARM, WCCP, cluster options, and so on.

◆ Click **Subscription** to enter your subscription key and provide Websense Web Filter information if your subscription includes the real-time options. See the Websense Manager Help system for more information on the real-time options.

◆ Click **UI Setup** to identify and change the port on which browsers can connect to Websense Content Manager, enable SSL connections to Websense Content

Manager, specify how often Websense Content Manager refreshes the statistics on the Monitor tab, and configure access control lists, administrator accounts, and user accounts to secure Websense Content Manager access.

◆ Click **Snapshots** to take and restore configuration snapshots.

◆ Click **Logs** to display, delete, or copy a selected log file to the local filesystem.

Protocols

◆ Click **HTTP** to configure HTTP caching and tune HTTP timeouts.

◆ Click **HTTP Responses** to specify which HTTP responses are sent to clients when the proxy detects an HTTP problem with a client transaction (such as unavailable origin servers, authentication requirements, and protocol errors).

◆ Click **HTTP Scheduled Update** to configure the proxy to load specific objects into the cache at scheduled times.

◆ Click **FTP** to configure FTP caching and to tune FTP timeouts.

The FTP options affect requests that originate from FTP clients only. You can configure options that affect FTP requests originating from HTTP clients from the HTTP group. The FTP button appears only if you have enabled FTP processing in the **Features** table on **Configure > My Proxy > Basic > General**.

◆ Click **HTTPS** to specify port information for inbound and outbound HTTPS traffic.

Content Routing

◆ Click **Hierarchies** to configure parent caching and ICP peer options.

◆ Click **Mapping and Redirection** to set URL remapping rules and FTP remapping rules.

◆ Click **Browser Auto-Config** to identify the port used to download browser auto-configuration files, and to set PAC and WPAD options.

Security

◆ Click **Connection Control** to specify which hosts are allowed to communicate with the Websense Content Gateway server (ARM security) and which clients are allowed to access the proxy cache.

◆ Click **Access Control** to set filtering rules and set proxy authentication options (LDAP, RADIUS, and NTLM).

◆ Click **SOCKS** to configure Websense Content Gateway to use a SOCKS firewall. The SOCKS button appears only if you have enabled SOCKS in the Features table on **Configure > My Proxy > Basic > General**.

Subsystems

◆ Click **Cache** to enable or disable cache pinning, configure the RAM cache size, specify the maximum size of objects allowed in the cache, and partition your cache according to protocol and origin servers.

◆ Click **Logging** to enable or disable event logging and set logging configuration options.

Networking

◆ Click **Connection Management** to specify the maximum number of connections the proxy can accept. For transparent proxy caching, you can specify the maximum number of client connections allowed before the proxy starts forwarding incoming requests directly to the origin server.

Ensure that the ARM is **On** on **Configure > My Proxy > Basic > General** before setting redirection rules.

◆ Click **ARM** to set redirection rules that specify how incoming packets are readdressed in transparent mode. You can also set dynamic and static bypass rules. The ARM button appears only if ARM is enabled in the Features table on **Configure > My Proxy > Basic > General**.

◆ Click **WCCP** to set WCCP configuration settings. The WCCP button appears only if you have enabled WCCP in the Features table under the My Proxy > Basic > Configure tab.

◆ Click **DNS Resolver** to enable or disable local domain expansion and tune host database timeouts.

◆ Click **Virtual IP** to enable or disable virtual IP failover and specify the virtual IP addresses managed by the Websense Content Gateway node. The Virtual IP button appears only if you have enabled Virtual IP in the Features table on **Configure > My Proxy > Basic > General**.

Command-line interface

As an alternative to Websense Content Manager, you can use the command-line interface to view and change your Websense Content Gateway configuration.

1. Log on to a Websense Content Gateway node as the Websense Content Gateway administrator, and then make the Websense Content Gateway **bin** directory (default location is **/opt/WCG/bin**) your working directory.

2. To view a configuration setting, enter the following command:

```
content_line -r var
```

where *var* is the variable associated with the configuration option (for a list of the variables, refer to *Configuration variables*, page 284).

3. To change the value of a configuration setting, enter the following command:

```
content_line -s var -v value
```

where *var* is the variable associated with the configuration option and *value* is the value you want to use.

For example, to change the FTP inactivity timeout option to 200 seconds, enter the following command at the prompt and press Return:

```
content_line -s
proxy.config.ftp.control_connection_timeout -v 200
```

> ✔ **Note**
>
> If the Websense Content Gateway **bin** directory is not in
> your path, prepend the command with `./` (for example, `./
> content_line -r *variable*`).

Configuration files

You can change Websense Content Gateway configuration options by editing specific
variables in the **records.config** file, located in the Websense Content Gateway **config**
directory (default location is **/opt/WCG/config**). Open the file in a text editor (such as
`vi` or `emacs`) and change the value of the variable.

> ✔ **Note**
>
> After you modify the **records.config** file, Websense
> Content Gateway must reread the configuration files; from
> the Websense Content Gateway **bin** directory (default
> location is **/opt/WCG/bin**), enter the command
> `content_line -x`. In some cases, you have to restart the
> proxy to apply the changes.

The figure below shows a sample portion of the **records.config** file:

Sample records.config file

Websense Content Gateway provides other configuration files that are used to configure specific features. All the configuration files are described in *Configuration Files*, page 255.

Saving and restoring configurations

The configuration snapshot feature lets you save all current configuration settings and restore them if needed. Websense Content Gateway can store configuration snapshots on the node where they are taken, on an FTP server, and on portable media. Websense Content Gateway restores a configuration snapshot on all the nodes in the cluster.

> ✓ **Note**
> It is recommended that you take a configuration snapshot before performing system maintenance or attempting to tune system performance. Taking a configuration snapshot takes only a few seconds and it can save you hours of correcting configuration mistakes.

This section describes how to perform the following tasks:

◆ Take a snapshot of the current configuration. See *Taking configuration snapshots*, page 97.

◆ Restore previously taken configuration snapshots. See *Restoring configuration snapshots*, page 97.

◆ Delete configuration snapshots stored on the Websense Content Gateway node. See *Deleting configuration snapshots*, page 98.

Taking configuration snapshots

You can save all the current configuration settings on your Websense Content Gateway system through Websense Content Manager.

To take a configuration snapshot and save it on the local system

1. Navigate to **Configure > Snapshots > File System**.
2. The **Change Snapshot Directory** field displays the name of the directory where Websense Content Gateway saves configuration snapshots. The default location is the Websense Content Gateway **config/snapshots** directory. To change the directory, enter the full path in the **Change Snapshot Directory** field. If you enter a relative path, Websense Content Gateway assumes that the directory is located in its **config** directory (for example, **/opt/WCG/config**).
3. In the **Save Snapshot** field, type the name you want to use for the current configuration.
4. Click **Apply**.

To take a configuration snapshot and save it on an FTP server

1. Navigate to **Configure > Snapshots > FTP Server**.
2. In the fields provided, enter the FTP server name, the login and password, and the remote directory where the FTP server stores configuration snapshots.
3. Click **Apply**.

   After you have successfully logged on to the FTP server, the **FTP Server** page displays additional fields.
4. In the **Save Snapshot to FTP Server** field, enter the name of the configuration snapshot you want to take.
5. Click **Apply**.

Restoring configuration snapshots

If you are running a cluster of Websense Content Gateway servers, the configuration is restored to all the nodes in the cluster.

To restore a configuration snapshot stored on the local node

1. Navigate to the **Configure > Snapshots > File System** tab.

2. From the **Restore > Delete Snapshot** drop-down list, select the configuration snapshot that you want to restore.

3. Click the **Restore Snapshot from "*directory_name*" Directory** box.

4. Click **Apply**.

   The Websense Content Gateway system or cluster uses the restored configuration.

To restore a configuration snapshot from an FTP server

1. Navigate to **Configure > Snapshots > FTP Server**.

2. In the fields provided, enter the FTP server name, the login and password, and the remote directory in which the FTP server stores configuration snapshots.

3. Click **Apply**.

   After you have successfully logged on to the FTP server, the **FTP Server** tab displays additional fields.

4. In the **Restore Snapshot** drop-down list, select the configuration snapshot that you want to restore.

5. Click **Apply**.

   The Websense Content Gateway system or cluster uses the restored configuration.

Deleting configuration snapshots

1. Navigate to **Configure > Snapshots > File System**.

2. From the **Restore > Delete a Snapshot** drop-down list, select the configuration snapshot you want to delete.

3. Click the **Delete Snapshot from "*directory_name*" directory** box.

4. Click **Apply**.

   The configuration snapshot is deleted.

11 | Working With Websense Data Security Suite

Websense Content Gateway support for ICAP (Internet Content Adaptation Protocol) with Websense Data Security Suite enables users of Websense Data Security Suite to control information leakage that can occur through postings to the World Wide Web.

ICAP facilitates off-loading of content for analysis to designated servers. Outgoing content, such as an upload or posting, is examined, and then either blocked or forwarded to its destination. The proxy acts as an ICAP client communicating with Websense Data Security Suite, which is acting as an ICAP server.

1. The proxy intercepts outbound content and provides that content to the DSS server.

2. Websense Data Security Suite, acting as the ICAP server, examines the content to determine if the FTP upload or Web posting should be allowed or blocked. This determination is based on the Websense Data Security Suite policy, and is communicated to the proxy.

3. The proxy then acts on the Websense Data Security Suite determination.

   a. If the content is blocked, it is not transmitted to the remote host, and Websense Data Security Suite returns a block page to the sender. Configure this page in Websense Data Security Suite; see your Websense Data Security Suite documentation for additional details.

   b. If the content is allowed, it is forwarded to its destination.

Websense Content Gateway support for ICAP with Websense Data Security Suite includes the following protocols:

◆ HTTP

◆ HTTPS (if your subscription includes SSL Manager)

Websense Content Gateway and Websense Data Security Suite support ICAP v1.0. ICAP support is for interoperability with Websense Data Security Suite.

> Related topic:
>
> *Configuring the ICAP client*, page 100

Configuring the ICAP client

Use the **Configure > My Proxy > Basic > General** page to configure the ICAP client.

1. In the **Networking** section of the Features table, select ICAP **On**.

2. Click **Apply**, and then click **Restart**.

3. Navigate to **Configure > Networking > ICAP > General**.

4. In the **ICAP Service URI** field, enter the Uniform Resource Identifier (URI) for the ICAP service. A URI is similar to a URL, but the URI ends with a directory, rather than a page. Obtain the identifier from your Websense Data Security Suite administrator. Enter the URI in the following format:

   `icap://hostname:port/path`

   For `hostname`, enter the IP address or hostname of the Websense Data Security Suite Protector appliance.

   The default ICAP port is 1344.

   `Path` is the path of the ICAP service on the host machine.

   For example:

   `icap://ICAP_machine:1344/REQMOD`

   You do not need to specify the port if you are using the default ICAP port 1344. The above URI can also be entered without the default port:

   `icap://ICAP_machine/REQMOD`

5. Under **Action for Communication Errors**, select whether to permit traffic or send a block page if Websense Content Gateway encounters an error while communicating with Websense Data Security Suite.

6. Under **Action for Large Files**, select whether to permit traffic or send a block page if a file larger than the size limit specified in Websense Data Security Suite is sent. The default size limit in DSS is 12 MB.

7. **Under Analyze Secure Content,** indicate if decrypted traffic should be sent to Websense Data Security Suite for analysis or sent directly to the destination. You must be running SSL Manager to send traffic to Websense Data Security Suite. See *Working With Encrypted Data*, page 121.

8. Click **Apply**.

> **Note**
> If you change the URI, you must restart Websense Content Gateway. Other changes do not require a restart.

12 | Security

Websense Content Gateway enables you to establish secure communication between the Websense Content Gateway system and other computers on the network. You can:

◆ Control which clients are allowed to access the proxy cache. See *Controlling client access to the proxy cache*, page 101.

◆ Control which hosts are allowed to access the proxy server. See *Controlling host access to the proxy server*, page 102.

◆ Control and secure access to Websense Content Manager using:

  ■ Administrator accounts (see *Setting the administrator ID and password*, page 103 and *Creating a list of user accounts*, page 104).

  ■ An access control list that defines which hosts are allowed to access Websense Content Manager (see *Controlling host access to Websense Content Manager*, page 105).

  ■ SSL (Secure Sockets Layer) protection for encrypted, authenticated access (see *Using SSL for secure administration*, page 105).

◆ Configure Websense Content Gateway integration into your firewall and control traffic through the SOCKS server. See *Configuring SOCKS firewall integration*, page 106.

◆ Configure Websense Content Gateway to use proxy authentication. The proxy supports LDAP, RADIUS, and NTLM proxy authentication. See *Configuring proxy authentication*, page 109.

Controlling client access to the proxy cache

1. Navigate to the **Configure > Security > Connection Control > Proxy Access** page.
2. Click **Edit File** to open the configuration file editor for the **ip_allow.config** file.
3. Enter information in the fields provided, and then click **Add**. All the fields are described in *Configuration Options*.

4. Click **Apply** to save the information, and then click **Close**.

> ✔ **Note**
> If an unauthorized client tries to access Websense Content Gateway, a message is displayed in their browser, indicating that the requested content cannot be obtained.

Controlling host access to the proxy server

You might want to restrict the type of communication possible with machines running Websense Content Gateway. Using the ARM security option, you can create an access control list that is used to either allow or deny other hosts from communicating with the Websense Content Gateway machine on specific ports. This *firewall* prevents potentially malicious packets from disrupting the operation of the machine.

When the ARM security option is enabled, the ARM examines UDP and TCP packets as they arrive at the Websense Content Gateway machine and matches them against the access control list that you specify in a configuration file. The ARM checks all UDP packets (since UDP communication is connectionless) and looks at the first TCP packet initiating the session against the configuration file access control list. Acceptable packets using either protocol are then passed up the network stack. Only incoming UDP and TCP packets are affected. This means that it is always possible to initiate TCP and UDP connections from the proxy cache, regardless of the access control list configured.

To use the ARM security feature, you must perform the following procedures in the order listed:

1. Edit the **arm_security.config** file (default location is in **/opt/WCG/config**) to open specific ports and define the hosts that are allowed to communicate with the Websense Content Gateway machine.
2. Enable the ARM security option.

> ❗ **Important**
> By default, when you enable the ARM security option, all ports on the Websense Content Gateway machine are closed, except for the Websense Content Gateway ports, DNS service ports, and ssh port 22. Before you enable the ARM security option, ensure that you have either console access to the Websense Content Gateway machine or that you have added the appropriate rules to the **arm_security.config** file to allow `telnet` or `ssh` access for yourself.

When you enable the ARM security option, the ports you specify in the access control list remain closed even when Websense Content Gateway is not running.

> **Important**
>
> You must define ports and hosts in the **arm_security.config** file before you enable the ARM Security option so that you do not lock yourself out of the Websense Content Gateway machine.

1. Navigate to **Configure > Security > Connection Control > ARM Security**.
2. Click **Edit File** to open the configuration file editor for the **arm_security.config** file.
3. Enter information in the fields provided, and then click **Add**. All fields are described in *Configuration Options*, page 197.
4. Click **Apply**, and then click **Close**.
5. On the **ARM Security** tab, enable the **ARM Security** option.
6. Click **Apply**.
7. Click **Restart** on **Configure > My Proxy > Basic > General**.

Controlling access to Websense Content Manager

You can restrict access to Websense Content Manager to ensure that only authenticated users can change configuration options and view performance and network traffic statistics.

You can:

◆ Set the master administrator ID and password. A user who logs on to Websense Content Manager with the administrator ID has access to all Websense Content Manager activities. See *Setting the administrator ID and password*, page 103.

◆ Create and maintain a list of user accounts that determines who can log on to Websense Content Manager and which activities they can perform. See *Creating a list of user accounts*, page 104.

◆ Create an access control list of IP addresses that defines which machines can access Websense Content Manager. See *Controlling host access to Websense Content Manager*, page 105.

◆ Use SSL for secure administration. See *Using SSL for secure administration*, page 105.

Setting the administrator ID and password

During installation, you assign a password that controls administrative access to Websense Content Manager. A user who logs on to Websense Content Manager using

the correct ID and password can view all the statistics on the Monitor tab and change any configuration options on the Configure tab.

You can change the administrator ID and password at any time.

1. Navigate to the **Configure > My Proxy > UI Setup > Login tab**.

2. Make sure that **Basic Authentication** is enabled.

   When Basic Authentication is disabled, any user can access Websense Content Manager unless you have set up a list of IP addresses that are denied access (see *Controlling host access to Websense Content Manager*, page 105).

3. To change the current administrator ID, type a new ID in the **Login** field of the **Administrator** section.

4. To change the current password, type the current password in the Old Password field. Type the new password in the New Password field, and then retype the new password in the New Password (Retype) field.

   If you have forgotten the current administrator password, see *How do you access Websense Content Manager if you forget the master administrator password?*, page 354.

5. Click **Apply**.

Creating a list of user accounts

If a single administrator ID and password for Websense Content Manager is not sufficient security for your needs, you can create a list of user accounts that define who has access to the Websense Content Manager and which activities they can perform.

1. Navigate to **Configure > My Proxy > UI Setup > Login**.

2. Enter the name of the user allowed to access Websense Content Manager.

3. Enter the password for the user, and then enter the password again in the New Password (Retype) field.

4. Click **Apply**.

5. In the **Access** drop-down list of the user table, select which Websense Content Manager activities the user can perform:

   - Select **No Access** to disable Websense Content Manager access for the user.

   - Select **Monitor Only** to allow the user to view statistics from the Monitor tab only.

   - Select **Monitor and View Configuration** to allow the user to view statistics from the Monitor tab and to view configuration options from the Configure tab.

   - Select **Monitor and Modify Configuration** to allow the user to view statistics from the **Monitor** tab and to change configuration options from the Configure tab.

6. Click **Apply**.

7. Repeat Step 2 through Step 6 for each user allowed to access Websense Content Manager.

8. Make sure that **Basic Authentication** is enabled.

   Websense Content Gateway checks user names and passwords only if this option is enabled.

Controlling host access to Websense Content Manager

In addition to using an administrator ID and user accounts, you can control which hosts have access to Websense Content Manager.

1. Navigate to **Configure > My Proxy > UI Setup Access**.

2. In the Access Control area, click **Edit File** to open the configuration file editor for the **mgmt_allow.config** file.

3. Enter information in the fields provided, and then click **Add**. All the fields are described in *UI Setup*, page 201.

4. Click **Apply**, and then click **Close**.

Using SSL for secure administration

Websense supports the Secure Sockets Layer protocol (SSL) to provide protection for remote administrative monitoring and configuration using Websense Content Manager. SSL security provides authentication of both ends of a network connection using certificates and provides privacy using encryption.

To use SSL, you must:

◆ Obtain an SSL certificate

◆ Enable the Websense Content Manager SSL option

Obtaining an SSL certificate

You can obtain an SSL certificate from a recognized certificate authority (for example, VeriSign). Install the certificate in the Websense Content Gateway **config** directory (default location is **/opt/WCG/bin**). You must either rename the certificate to the default filename **private_key.pem**, or specify the name of the certificate using Websense Content Manager (follow the procedure in *Enabling SSL*, page 105).

Enabling SSL

After you have obtained an SSL certificate, you can enable SSL.

1. Navigate to the **Configure > My Proxy > UI Setup > General** tab.

2. Enable the **HTTPS** option.

3. In the Certificate File field, specify the filename of the SSL certificate.

   You have to change the filename only if the certificate file does not use the default name **private_key.pem**.

4. Click **Apply**.

Controlling access to Web sites

You can permit or block particular URL requests, keep or strip header information from client requests, and specify authentication rules. For information on authentication, see *Configuring proxy authentication*, page 109.

1. On the **Configure** > **Security** > **Access Control > Filtering** tab, click **Edit File** to open the configuration file editor for the *filter.config* file.

2. Add a rule for each IP address or IP address range allowed to deliver requests, as follows:

   - From the **Rule Type** drop-down box, select **allow**.

   - From the **Primary Destination Type** drop-down box, select **dest_domain**.

   - In the **Primary Destination Value** field enter **.** (a period).

   - Select one of the following for Header Type:

   - **LDAP Server Name** can contain the name of one LDAP server or a list of servers.

   - LDAP Base Distinguished Name

3. For information on optional fields and secondary specifiers, see *Access Control*, page 227.

Configuring SOCKS firewall integration

SOCKS is commonly used as a network firewall that allows hosts behind a SOCKS server to gain full access to the Internet and prevents unauthorized access from the Internet to hosts inside the firewall.

When the proxy receives a request for content that is not in the cache or is stale, it must request the content from the origin server. In a SOCKS configuration, instead of accessing the origin server directly, the proxy goes through a SOCKS server. The SOCKS server authorizes communication between the proxy and the origin server and relays the data to the origin server. The origin server then sends the content back to the proxy through the SOCKS server.

The proxy caches the content and sends it to the client.

Websense Content Gateway can act as a *SOCKS client*, where it receives and serves HTTP or FTP requests as usual, and as a *SOCKS proxy*, where it receives SOCKS traffic (usually on port 1080).

As a SOCKS proxy, Websense Content Gateway relays requests to and from the SOCKS server.

> ✓ **Note**
> Websense Content Gateway does not perform authentication with the client. However Websense Content Gateway can perform user name and password authentication with a SOCKS server running SOCKS version 5.

Configuring the proxy to use a SOCKS firewall

1. Navigate to **Configure > My Proxy > Basic > General**.
2. In the Features table, click **SOCKS On** in the Security section.
3. Click **Apply**.
4. Navigate to **Configure > Security > SOCKS > General**.
5. Specify the SOCKS version running on your SOCKS servers.
6. Click **Apply**.
7. Click the **Server** tab.
8. In the Default Servers field of the SOCKS Server section, enter the host names of your default SOCKS servers and the ports through which the proxy communicates with the SOCKS servers. Separate the host name and the port with a colon (`:`) and separate each entry with a semicolon (`;`): for example, `socks1:1080;socks2:4080`.
9. Click **Apply**.
10. In the SOCKS Server Rules area, click **Edit File** to perform additional SOCKS server configuration, such as SOCKS server bypass and authentication.

    The configuration file editor for the **socks.config** file opens.
11. Enter information in the fields provided, and then click **Add**. All the fields are described in *Configuration Options*.
12. Click **Apply**, and then click **Close**.
13. Click **Restart** on **Configure > My Proxy > Basic > General**.

Setting SOCKS proxy options

To configure Websense Content Gateway as a SOCKS proxy, you must enable the SOCKS proxy option and specify the port on which WCG accepts SOCKS traffic from SOCKS clients.

As a SOCKS proxy, Websense Content Gateway can receive SOCKS packets (usually on port 1080) from the client and forwards all requests directly to the SOCKS server.

> **Note**
> ✔ You must set SOCKS proxy options in addition to enabling the SOCKS option and specifying SOCKS server information described in *Configuring the proxy to use a SOCKS firewall*, page 107.

1. Navigate to **Configure** > **Security** > **SOCKS > Proxy**.
2. Enable **SOCKS Proxy**.
3. Specify the port on which Websense Content Gateway accepts SOCKS traffic. The default is port 1080.
4. Click **Apply**.
5. Click **Restart** on **Configure > My Proxy > Basic > General**.

Setting authentication

For SOCKS version 5, you can configure the proxy to use username and password authentication with the SOCKS server. You specify the username and password the proxy must use in the **socks.config** file.

1. Navigate to **Configure > Security > SOCKS > Server** to display the **socks.config** file.
2. At the end of the file, add a line using the following format:

   ```
   auth u username password
   ```

   username is the username and *password* is the password Websense Content Gateway must use for authentication with the SOCKS version 5 server.
3. Click **Apply**.
4. Click **Restart** on **Configure > My Proxy > Basic > General**.

Setting SOCKS server bypass

You can configure Websense Content Gateway to bypass the SOCKS server and access certain origin servers directly (SOCKS Server bypass).

1. Navigate to **Configure > Security > SOCKS > Server** to display the **socks.config** file.
2. Enter a line in the file specifying the IP addresses or IP address range of the origin servers that you want Websense Content Gateway to access directly. Use the following format:

   ```
   no_socks IPaddresses or IPaddress range
   ```

IPaddresses or *IPaddress range* is a comma-separated list of the IP addresses or IP address ranges associated with the origin servers you want Websense Content Gateway to access directly.

3. Click **Apply**

4. Click **Restart** on **Configure > My Proxy > Basic > General**.

Example

To configure Websense Content Gateway to access the origin servers associated with the range of IP addresses 123.14.15.1 - 123.14.17.4 and the IP address 113.14.18.2 directly without going through the SOCKS server.

```
no_socks 123.14.15.1 - 123.14.17.4, 113.14.18.2
```

Configuring proxy authentication

Websense Content Gateway supports LDAP, RADIUS, and NTLM proxy authentication in both explicit and transparent proxy modes.

> **Important**
>
> Enable only one authentication option.

See the following sections for information:

◆ *Authentication in transparent proxy mode*, page 110

◆ *Using LDAP proxy authentication*, page 112

◆ *Using RADIUS proxy authentication*, page 114

◆ *Using NTLM proxy authentication*, page 117

Websense Content Gateway does not support proxy authenication in clustered environments using WCCP load balancing.

When sending traffic through a Web Cache Control Protocol (WCCP)-enabled router, WCCP decides which proxy to use based on the destination host. This maximizes the potential that the cache will have the content it requires. One user can leverage many proxies.

Authentication on the proxy requires users to identify themselves. When WCCP is combined with load balancing, end users are prompted multiple times (once for each proxy used.)

To support authentication, the load balancer needs to ensure flow affinity by the client (source) as opposed to the destination. This requires special hardware and is not supported on basic WCCP routers and switches.

For more info see the Mask-Assignment section on the following Web site:
http://www.cisco.com/en/US/docs/ios/ipapp/configuration/guide/ipapp_wccp.html

Authentication in transparent proxy mode

Websense Content Gateway must be in one of two modes to support authentication in transparent proxy mode:

◆ In *IP mode* (the default), a client IP address is associated with a username after the user session is authenticated. Requests made from this client IP are not required to authenticate for the duration specified in the *proxy.config.http.transparent_auth_session_time* variable. (The default is 15 minutes.) Requests made from this IP address within the timeout window are considered to have been made by the user associated with this IP address. When the timeout period expires, the client is required to re-authenticate.

◆ *Cookie mode* is required to uniquely identify users who share a single IP address, such as, for example, in environments where proxy-chaining is used or where network address translation (NAT) occurs.

In either mode, client authentication is valid for a default of 15 minutes.

You can change modes or the default timeout period for authentication by editing the value of variables in the **records.config** file.

1. Open the **records.config** file located in the Websense Content Gateway **config** directory (default location is **/opt/WCG/config**).

2. Edit the following variables:

Variable	Description
`proxy.config.http.transparent_auth_type`	Enter: • 0 (cookie mode) to associate a session ID with the username after the user session is authenticated. This setting is required to uniquely identify users who share a single IP address, such as, for example, in environments where proxy-chaining is used or where network address translation (NAT) occurs. • 1 (IP mode) to associate a client IP address with a username after the user session is authenticated. In either mode, the length of time before a client must re-authenticate is determined by the value of the `proxy.config.http.transparent_auth_session_time` variable.
`proxy.config.http.transparent_auth_session_time`	Specify the length of time (in minutes) before the browser must re-authenticate. This is required in both IP and cookie modes.

3. Save and close the file.
4. From the Websense Content Gateway **bin** directory (default location is **/opt/WCG/bin**), run `content_line -L` to restart Websense Content Gateway on the local node or `content_line -M` to restart Websense Content Gateway on all the nodes in a cluster.

By default, clients are redirected to the Proxy Name of WCG to authenticate. If the workstations on the network are not able to resolve the Proxy Name via DNS or an alternate DNS name for the proxy is defined, use the `proxy.config.http.transparent_auth_hostname` configuration variable to specify an alternate DNS name for the proxy

> **Note**
> This variable is not included in the **records.config** file.
> You must enter both the variable and its value in the file.

Variable	Description
`proxy.config.http.transparent_auth_hostname`	Specify an alternate name for the proxy. This may be needed if workstations on the network are unable to resolve the proxy name.

Using LDAP proxy authentication

Websense Content Gateway provides the LDAP option to ensure that users are authenticated with an LDAP server before accessing content through the proxy cache. In the proxy, LDAP authentication supports both simple and anonymous bind.

When you enable the LDAP option, the proxy acts as an LDAP client and directly challenges users who request content for a username and password. After receiving the username and password from a client, the proxy contacts the LDAP server to check that they are correct. If the LDAP server *accepts* the username and password, the proxy serves the client with the requested content and stores the username and password entry in the Websense Content Gateway LDAP cache; all future authentication requests for that user are served from the LDAP cache until the entry expires. If the LDAP server *rejects* the username and password, the user's browser displays a message indicating that authorization failed and prompts again for a username and password.

Configuring Websense Content Gateway to be an LDAP client

1. Navigate to **Configure > My Proxy > Basic > General**.
2. In the Features table, click LDAP **On** in the Authentication section.
3. Click **Apply**.
4. Navigate to **Configure > Security > Access Control > LDAP**.
5. Enable **Purge Cache on Authentication Failure** to configure Websense Content Gateway to delete the authorization entry for the client in the LDAP cache if authorization fails.
6. Enter the host name of the LDAP server.
7. Enter the port on which Websense Content Gateway communicates with the LDAP server. The default is port 389.
8. Select the type of your directory service to set the filter for searching. The default is `sAMAccountName` for Active Directory. Select **uid** for eDirectory or other directory services.
9. Enter the Full Distinguished Name (fully qualified name) of a user in the LDAP-based directory service. For example:

    `CN=John Smith,CN=USERS,DC=MYCOMPANY,DC=COM`

    Enter a maximum of 128 characters in this field.

    If you do not specify a value for this field, the proxy attempts to bind anonymously.

10. Enter a password for the user specified in the previous step.

11. Enter the Base Distinguished Name (DN). Obtain this value from your LDAP administrator.

12. Click **Apply**.

13. Click **Restart** on **Configure > My Proxy > Basic > General**.

As optional steps, you can change LDAP cache options (see *Setting LDAP cache options*, page 113) and configure Websense Content Gateway to allow certain clients to access specific sites on the Internet without being authenticated by the LDAP server (see *Access Control*, page 227).

Setting LDAP cache options

By default, the LDAP cache is configured to store 5000 entries and each entry is considered fresh for 3000 minutes. Change these options by editing the **records.config** file.

1. Open the **records.config** file located in the Websense Content Gateway **config** directory (default location is in **/opt/WCG/config**).

2. Edit the following variables:

Variable	Description
`proxy.config.ldap.cache.size`	Specify the number of entries allowed in the LDAP cache. The minimum value is 256 entries.
`proxy.config.ldap.auth.ttl_value`	Specify the number of minutes that Websense Content Gateway can store username and password entries in the LDAP cache.
`proxy.config.ldap.cache.storage_size`	Specify the maximum amount of space (in bytes) that the LDAP cache can occupy on disk. When modifying this value, you must update the value of `proxy.config.ldap.cache.size` proportionally. For example, if you double the storage size, also double the cache size. Modifying this variable without modifying `proxy.config.ldap.cache.size` causes the LDAP subsystem to stop functioning.

3. Save and close the file.

4. From the Websense Content Gateway **bin** directory (default location is in **/opt/ WCG/bin**), run `content_line -L` to restart the proxy on the local node or `content_line -M` to restart the proxy on all the nodes in a cluster.

Configuring secure LDAP

By default, LDAP traffic is transmitted unsecured. You can make LDAP traffic confidential and secure by using Secure Sockets Layer (SSL) / Transport Layer Security (TLS) technology. You can enable LDAP over SSL (LDAPS) by installing a properly formatted certificate from either a Microsoft certification authority (CA) or a non-Microsoft CA.

To use LDAPS with Websense Content Gateway:

1. Open the **records.config** file located in the Websense Content Gateway **config** directory (default location is in **/opt/WCG/config**).

2. Add following entry to **records.config**:

   `CONFIG proxy.config.ldap.secure.bind.enabled INT 1`

3. Navigate to **Configure > Security > Access Control > LDAP** and change the port to 3269.

> ✓ **Note**
> The Directory Service must be configured to support LDAPS authentication. Please refer to the documentation provided by your directory provider for instructions.

Using RADIUS proxy authentication

Websense Content Gateway provides the RADIUS option to ensure that users are authenticated with a RADIUS server before accessing content the proxy cache.

When you enable the RADIUS option, Websense Content Gateway acts as a RADIUS client and directly challenges users who request content for a username and password. After receiving the username and password from a client, Websense Content Gateway contacts the RADIUS server to check that they are correct. If the RADIUS server *accepts* the username and password, the proxy serves the client with the requested content and stores the username and password entry in the RADIUS cache; all future authentication requests for that user are served from the RADIUS cache until the entry expires. If the RADIUS server *rejects* the username and password, the user's browser displays a message indicating that authorization failed and prompts again for a username and password.

Websense Content Gateway supports a primary RADIUS server and a secondary RADIUS server for failover. If the primary server does not respond to the proxy request within the specified timeout (60 seconds by default), Websense Content Gateway tries to check the username and password again. If a response from the primary RADIUS server is not received after the maximum number of retries (10 by default), the proxy contacts the secondary RADIUS server. If Websense Content

Gateway cannot contact the secondary RADIUS server, the user is prompted again for a username and password.

The RADIUS cache is held in memory and stored on disk. Websense Content Gateway updates the data on disk every 60 seconds. In addition, Websense Content Gateway stores username and password entries in the RADIUS cache for 60 minutes. If a password and username entry is expired in the RADIUS cache, Websense Content Gateway contacts the RADIUS server to accept or reject the username and password.

To configure Websense Content Gateway to be a RADIUS client:

◆   Enable the RADIUS option.

◆   Specify the host name or IP address of the primary and secondary (optional) RADIUS servers, and the port and shared key that Websense Content Gateway uses to communicate with the RADIUS servers.

See *Configuring Websense Content Gateway to be a RADIUS client*, page 115.

As optional steps, you can change RADIUS cache and server timeout options (see *Setting RADIUS cache and server timeout options*, page 116) and configure Websense Content Gateway to allow certain clients to access specific sites on the Internet without being authenticated by the RADIUS server (see *Access Control*, page 227).

Configuring Websense Content Gateway to be a RADIUS client

1.   Navigate to **Configure > My Proxy > Basic > General**.

2.   In the Features table, click Radius **On** in the Authentication section.

3.   Click **Apply**.

4.   Navigate to **Configure > Security > Access Control > Radius**.

5.   Enter the host name of your primary RADIUS server.

6.   Enter the port number through which Websense Content Gateway communicates with the primary RADIUS server.

7.   Enter the key used for encoding.

8.   If you are using a secondary RADIUS server, enter the host name, port, and shared key in the appropriate fields of the **Secondary Radius Server (Optional)** area.

9.   Click **Apply**.

10. Click **Restart** on **Configure > My Proxy > Basic > General**.

> **✓ Note**
>
> In addition to performing these procedures, you must add the Websense Content Gateway machine as a trusted client on the primary and secondary RADIUS servers and provide the shared key you want to use for the Websense Content Gateway machine (the shared key must be the same one you specify in the procedure below). See your RADIUS server documentation.

Setting RADIUS cache and server timeout options

By default, the RADIUS cache and RADIUS server timeout options are configured as follows:

◆ The RADIUS cache is configured to store 1,000 entries and each entry is considered fresh for 60 minutes.

◆ Websense Content Gateway can try to re-establish a connection to the RADIUS server if the connection remains idle for 10 seconds and can retry the connection a maximum of 10 times.

Change these default values by editing the **records.config** file.

1. Open the **records.config** file located in the Websense Content Gateway **config** directory (default location is in **/opt/WCG/config**).

2. Edit the following variables:

Variable	Description
`proxy.config.radius.auth.min_timeout`	Specify the amount of time in seconds that the Websense Content Gateway connection to the RADIUS server remains idle before Websense Content Gateway closes the connection.
`proxy.config.radius.auth.max_retries`	Specify the maximum number of times Websense Content Gateway tries to connect to the RADIUS server.
`proxy.config.radius.cache.size`	Specify the number of entries allowed in the RADIUS cache. The minimum value is 256 entries. If you enter a value lower than 256, Websense Content Gateway signals a SEGV.

Variable	Description
`proxy.config.radius.auth.ttl_value`	Specify the number of minutes that Websense Content Gateway can store username and password entries in the RADIUS cache.
`proxy.config.radius.cache.storage_size`	Specify the maximum amount of space that the RADIUS cache can occupy on disk. This value must be at least 100 times the number of entries. It is recommended that you provide the maximum amount of disk space possible.

3. Save and close the file.

4. From the Websense Content Gateway **bin** directory (default location is in **/opt/WCG/bin**), run `content_line -L` to restart Websense Content Gateway on the local node or `content_line -M` to restart WCG on all the nodes in a cluster.

Using NTLM proxy authentication

Websense Content Gateway provides the NTLM (NT LAN Manager) option to ensure that users in a Windows network are authenticated before they access protected content on the Internet.

When you enable the NTLM option, the proxy challenges users who request content for proof of their credentials. The proxy then sends the proof of the user's credentials directly to the Windows domain controller to be validated. If the credentials are valid, the proxy serves the requested content and stores the credentials in the NTLM cache for future use. If the credentials are not valid, the proxy sends an *authentication failed* message to the user

Websense Content Gateway supports both *Single Sign-On* for Microsoft Internet Explorer (MSIE) and Basic authentication for Netscape and other browsers. Single Sign-On allows users to sign on only once, so that they can seamlessly access all authorized network resources. Therefore, if a user has already logged on to the Windows network successfully, the credentials specified during Windows logon are used for authentication and the user is not prompted again for a username and password. With Basic authentication, users *are* prompted for a username and password before they can access the protected content.

Whether Websense Content Gateway will be running in explicit or transparent mode, if authentication is through NTLM with Active Directory 2003, SMB signing must be configured. See http://technet.microsoft.com/en-us/library/cc775652.aspx for an overview of the steps needed to configure SMB signing on your Windows Server

2003-based domain controllers. For more in-depth information on SMB, including how to disable it, see http://technet.microsoft.com/en-us/library/cc781273.aspx.

Websense Content Gateway supports the use of backup domain controllers for failover. If the primary domain controller does not respond to the proxy request, Websense Content Gateway contacts the next domain controller in the list (the backup domain controller). For the next request, the proxy tries to contact the primary domain controller again and then contacts the backup domain controller if the connection fails. Websense Content Gateway does this five times before considering the server unavailable. After considering the primary domain controller unavailable, Websense Content Gateway waits 5 minutes before trying to contact it again.

> **Important**
> Websense Content Gateway does not support WINS resolution. Domain controllers must have host names that can be resolved by a DNS server.

> **Important**
> NTLM does not work if Extended Security is enabled on the Domain Controller.

Websense Content Gateway must be on the same domain as the directory service/ authentication server (for example, Active Directory). Websense Content Gateway supports access to Windows NT domain controllers and to the Windows 2000 and Windows 2003 Active Directory.

You must disable SMB signing when using NTLM authentication.

Configuring NTLM proxy authentication

1. Navigate to **Configure > My Proxy > Basic > General**.
2. In the Features table, click NTLM **On** in the Authentication section.
3. Click **Apply**.
4. Navigate to **Configure > Security > Access Control > NTLM**.
5. Enter the host names of your domain controllers in IP address format. Separate each entry with a comma.
6. Enter the domain name of the domain controller you want Websense Content Gateway to authenticate against. Enter only the name of the domain; do not enter an extension (such as ".com" or ".org").
7. Enable **Load Balancing** if you want the proxy to balance the load when sending authentication requests to the domain controllers.

The **Configure > Security > Access Control > NTLM** page looks similar to this screen.
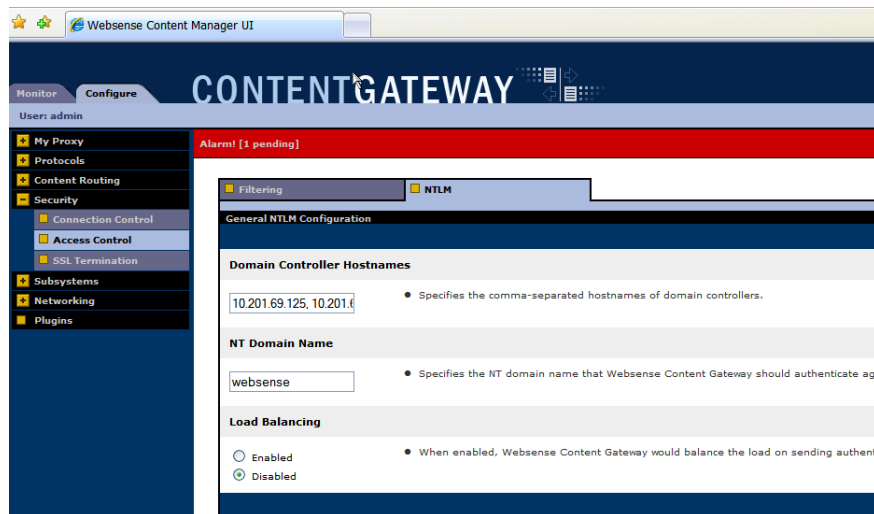
8. Click **Apply**.

9. Click **Restart** on **Configure > My Proxy > Basic > General**.

As optional steps, you can change NTLM cache storage options (see *Setting NTLM cache options*, page 119) and configure Websense Content Gateway to allow certain clients access to specific sites on the Internet without being authenticated by a domain controller (see *Access Control*, page 227).

✔ **Note**

In addition to performing the procedure described above, you must add the Websense Content Gateway machine to the domain Windows NT controllers or to the Windows 2000 or Windows 2003 Active Directory. See your Windows documentation for details.

Setting NTLM cache options

By default, the NTLM cache is configured to store 15728640 entries and each entry is considered fresh for 60 minutes. You can change these values by editing the **records.config** file.

1. Open the **records.config** file located in the Websense Content Gateway **config** directory (default location is in **/opt/WCG/config**).

2. Edit the following variables:

Variable	Description
`proxy.config.ntlm.cache.enabled`	Set to 0 to disable the NTLM cache. When disabled, Websense Content Gateway does not store any credentials in the NTLM cache for future use.
`proxy.config.ntlm.cache.ttl_value`	Specify the amount of time (in seconds) that Websense Content Gateway can store entries in the NTLM cache.
`proxy.config.ntlm.cache.size`	Specify the number of entries allowed in the NTLM cache.
`proxy.config.ntlm.cache.storage_size`	Specify the maximum amount of space that the NTLM cache can occupy on disk. This value should be proportionate to number of entries in the NTLM cache. For example, if each entry in the NTLM cache is approximately 128 bytes and the number of entries allowed in the NTLM cache is 5000, the cache storage size should be at least 64000.

3. Save and close the file.

4. From the Websense Content Gateway **bin** directory (default location is in **/opt/ WCG/bin**), run `content_line -L` to restart Websense Content Gateway on the local node or `content_line -M` to restart Websense Content Gateway on all the nodes in a cluster.

13 | Working With Encrypted Data

SSL (Secure Sockets Layer) is the industry standard for transmitting secure data over the Internet. It is based on a system of trusted certificates issued by certificate authorities and recognized by servers.

If your subscription includes SSL Manager, SSL-encrypted traffic is decrypted, inspected, and then re-encrypted before it is sent to its destination. To preserve performance, only HTTPS traffic is diverted in this manner; HTTP traffic goes through the proxy to the requested site.

Each SSL-based request consists of two separate sessions:

◆ From the client browser to SSL Manager. This is considered *inbound* SSL traffic.
◆ From SSL Manager to the Web server that will receive the secure data. This is considered *outbound* SSL traffic.

Different certificates are required for these sessions.
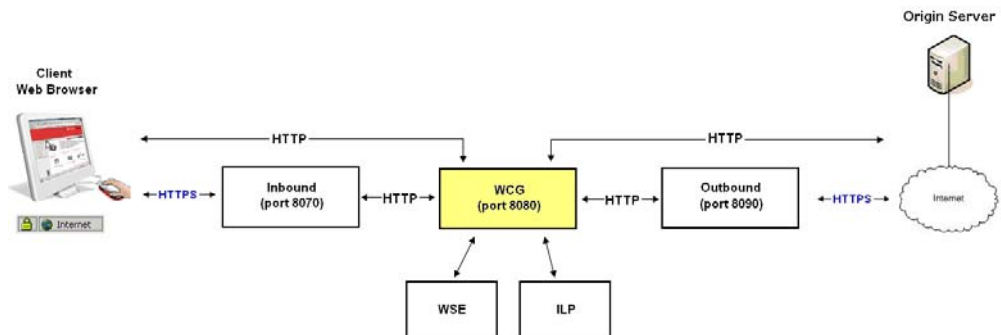
Websense Content Gateway does not cache HTTPS data.

For additional information on certificates and SSL, consult any of the commercially available books on SSL for a complete description of certificates. For information on preparing your system, see the Websense Content Gateway *Installation Guide*.

Running in explicit proxy mode

If you have an existing PAC file, replace the **proxy.pac** file located in the Websense Content Gateway **config** directory (default location is **/opt/WCG/config**) with the existing file. If you do not have a PAC file already, see Step 4 below for a script you can copy.

1. On the **Configure > My Proxy > Basic > General** tab, ensure that both ARM and HTTPS are enabled.

   For ARM, check the Networking section; for HTTPS check the Protocols section. If they are disabled, set them to **On**. Then click **Apply**, followed by **Restart**.

2. Navigate to **Configure > Content Routing > Browser Auto-Config > PAC**.

3. In the **Auto-Configuration Port** field, specify the port that the proxy uses to serve the PAC file. The default port is 8083.

4. The PAC Settings area displays the **proxy.pac** file:

   - If you copied an existing PAC file into the Websense Content Gateway **config** directory, the **proxy.pac** file contains your proxy configuration settings. Check the settings and make changes if necessary.

   - If you did not copy an existing PAC file into the Websense Content Gateway **config** directory, the **proxy.pac** file is empty. Copy and paste the following script for your PAC settings. You must provide the proxy domain name or IP address.

     ```
     function FindProxyForURL(url, host)
     {
       url = url.toLowerCase();
       host = host.toLowerCase();
     ```

```
        if(url.substring(0, 5) == "http:"){
          return "PROXY WCG_DOMAIN_NAME_or_IP_Address:8080";
        }
        else if(url.substring(0, 4) == "ftp:"){
          return "PROXY WCG_DOMAIN_NAME_or_IP_Address:2121";
        }
        else if(url.substring(0, 6) == "https:"){
          return "PROXY WCG_DOMAIN_NAME_or_IP_Address:8070";
        }
        else{
          return "DIRECT";
        }
      }
```

5. Click **Apply**.

6. Click **Restart** on **Configure > My Proxy > Basic > General**.

Once the new PAC information is in place, you must inform your users to set their browsers to point to the PAC file. For example, if the PAC file is located on the proxy server with the host name `proxy1` and Websense Content Gateway uses the default port 8083 to serve the file, users must specify the following URL in the proxy configuration settings:

```
        http://proxy1.company.com:8083/proxy.pac
```

The procedures for specifying the PAC file location vary among browsers.

For Microsoft Internet Explorer version 7.0 and later:

1. Navigate to **Tools > Internet Options > Connections > LAN Settings**.

2. Select **Use automatic configuration script** field, and enter

    `http://WCG_Domain_Name_or_IP_Address:8083/proxy.pac`

    in the **Address** field.

3. Click **OK**.

For Mozilla Firefox 2.0.0 and later:

1. Navigate to **Tools > Options > Advanced > Network > Settings > Connection > Settings.**

2. Select **Automatic proxy configuration URL** field, and enter

    `http://WCG_Domain_Name_or_IP_Address:8083/proxy.pac`

3. Click **Reload**, and then click **OK**.

See the documentation for your browser for details.

Enabling SSL Manager

If your subscription includes SSL Manager, follow these steps:

1. Ensure that you have entered your subscription key on the **Configure > My Proxy > Subscription > Subscription Management** tab. See *Entering your subscription key*, page 11.

2. On **Configure > My Proxy > Basic > General**, click HTTPS **On**.

> ✓ **Note**
>
> If you are running with other Websense products, such as Websense Data Security Suite, that inspect HTTPS traffic, you must select enable HTTPS here.

3. Click **Apply** and then click **Restart**.

4. On **Configure > My Proxy > UI Setup > General**, specify the port for the SSL Manager user interface. The default is 8071. This must be a different port than the Websense Content Manager interface (default 8081).

5. Enter the name of the SSL certificate file. See *Creating a sub-certificate authority*, page 127.

Use the **Configure > Protocols > HTTPS** page to provide port information.

1. In the **HTTPS Proxy Server Port** field, enter the port for inbound (client to SSL Manager) HTTPS traffic.The default is 8070.

2. In the **SSL Outbound Port** field, enter the port SSL Manager users for HTTPS traffic from SSL Manager to the destination server. The default is 8090.

Tasks

For inbound (client to SSL Manager) traffic, perform these steps in preparation to deploying SSL Manager:

1. Create an internal root CA (certificate authority). In order to sign SSL traffic, SSL Manager requires an internal SSL Certificate Authority that has the capability to sign SSL certificates. This is for traffic between the browser and SSL Manager. See *Internal Root CAs*, page 125.

2. Add this CA to the certificate tree. Servers, such as destination servers, check this tree to ensure that they can trust users because they have certificates from an authority listed here. The certificates listed on the certificate tree are certificate authorities you empower (trust) to verify the validity of individual Web sites. Any Web site signed by a certificate authority in the certificate tree with the "allow" status is allowed through SSL Manager. See *Managing certificates*, page 129

3. Customize pages that browser users will see. See *Customizing messages*, page 145. Among the pages that can be customized are a connect failure and certificate verification failure page.

Certificates

Security revolves around certificates. One role SSL Manager plays is to ensure that certificates are valid. A certificate must meet three criteria:

◆ It must be current (has not expired or been revoked). See *Validating certificates*, page 133.

◆ It must be issued by a trusted CA (certificate authority). See *Managing certificates*, page 129

◆ The URL and the certificate owner match. See *Configuring validation*, page 133.

Traffic from the client browser to SSL Manager requires a certificate issued by an internal root certificate authority. See *Internal Root CAs*, page 125.

Traffic from SSL Manager to the destination server requires a certificate issued by one of the authorities listed on the Certificate Authority Tree on the **Configure > SSL > Certificates > Certificate Authorities** tab. See *Managing certificates*, page 129.

Internal Root CAs

The internal Root CA dynamically generates all certificates used between the client browser and SSL Manager. You must have an internal Root CA to pass inbound traffic to SSL Manager. You can either import or create this CA. Internal Root CAs are stored in **/opt/wcg/sxsuite/db/server/default.pem.**

> **Important**
> Be sure to back up any existing Internal Root CAs before importing or creating new ones. This enables you to return to an earlier version of the certificate, if necessary. See *Backing up your internal Root CA*, page 128 for details.
>
> Only one internal Root CA can be active at any time.

There are three options for creating an internal Root CA:

◆ Leverage an existing corporate CA and import it into SSL Manager. See *Importing your Root CA*, page 126.

◆ Create a new CA for proxies and make that CA available to browsers. See *Creating your new Root CA*, page 126.

◆ Create a sub-certificate CA. This leverages a corporate CA, but can also be revoked by the corporate CA. See *Creating a sub-certificate authority*, page 127.

Importing your Root CA

If your organization already has a root certificate authority, you can import it. This certificate must be trusted by all browsers in your organization. Be sure to back up any new internal Root CAs that you import. See *Backing up your internal Root CA*, page 128 for details.

1. Navigate to **Configure > SSL > Internal Root CA** > **Import Root CA**.

2. Browse to select the certificate. The certificate must be in X.509 format and base64-encoded.

3. Browse to select the private key. It must correspond to the certificate you selected in Step 2.

4. Enter, and then confirm, the passphrase.

5. Click **Import Root CA**. The imported CA is stored in **/opt/WCG/sxsuite/db/**.

Creating your new Root CA

If you do not already have a Root CA, fill in the fields on this tab to create one. Be sure to back up any new internal Root CAs that you create. See *Backing up your internal Root CA*, page 128 for details.

An asterisk (*) on this page indicates a required field.

1. Select **Configure > SSL > Internal Root CA**, and then select **Create Root CA**.

2. Provide requested information in the fields, particularly noting the following:

   ■ The fields **Organization**, **Organizational Unit**, (this field is optional) and **Common Name** comprise a *distinguished name*.

   • For **Organization**, enter the name of your company.

   • For **Common Name**, enter the name of your company certificate authority.

   ■ The comment becomes part of the certificate. The first line you enter can be seen by end users.

   ■ Enter, and then confirm, the passphrase. (A passphrase is similar to a password. Usually, however, it is longer to provide greater security. It is recommended that you use a strong passphrase, with a combination of numbers, characters, and upper- and lower-case letters.

3. Click **Generate and Deploy Certificate** to deploy the certificate to the Content Gateway server.

> Related topic:
>
> *Creating a sub-certificate authority*, page 127

Creating a sub-certificate authority

Creating a sub-certificate authority (sub-CA) enables you to take advantage of all the information already existing for your Root CA. However, the Root CA can revoke the sub-CA at any time.

Follow these steps to generate a sub CA using the certificate services in Microsoft Windows 2003.

Preparation

▶ Install the OpenSSL toolkit from www.openssl.org.

Creating a request

1. Create a request for the sub-CA with OpenSSL.

   a. Create a folder to store the certificate files.

   b. Create a **req_ca.cnf** file and put it in the folder you created in step a. You can copy the file in *The req_ca.cnf File*, page 343.

   c. Answer all the questions, or put the defaults of your CA into the **req_ca.cnf** file. The password you enter here is needed later when you import the sub-CA into SSL Manager. This password protects the private key of your new sub-CA.

2. Create the request with the private key:

   ```
   openssl req -newkey rsa:1024 -keyout subca.key -new -days
   3650 -config ./req_ca.cnf -out subca.req
   ```

You now have a subca.key and a **subca.req** file. The subca.key is the private key of the new sub-CA.

Signing the request

You must sign the request with Microsoft Certificate Services.

1. Select **Start > Administrative Tools > Certification Authority**.
2. Right-click on the name of the server and select **All Tasks > Submit new request**.
3. From the Browse dialog box, select the **subca.req** file.
4. Select **Pending Requests** to see your request.
5. Right-click your request and select **All Tasks > Issue.**
6. Select **Issued Certificates** to see that your new certificate is listed there.

Exporting the public key

1. Add a column to the Issued Certificates window.

   a. Right-click **Issued Certificates** and select **View > Add/Remove Columns**.

   b. Click **Binary Public Key** and then click **Add** to add a heading to the new column.

2. Export the binary data.

      a. In the Issued Certificates window, select your new certificate and then select **All Tasks > Export Binary Data**.

      b. Select **Binary Certificate** from the drop-down list.

      c. Mark **Save binary data to a file**.

      d. Click **OK**.

3. Name your certificate file **subca.cer**. This file is the public key of the sub-CA in binary format.

4. Move the **subca.cer** file to the directory where the subca.key resides. See *Creating a request*, page 127.

5. Convert the binary file into pem format using this OpenSSL command:

```
openssl x509 -in subca.cer -inform der -out subca.pem -
outform pem
```

6. Through Internet Explorer, export the **rootca.pem** file. This is the public key of the root CA in base64 format.

Import the sub-CA into SSL Manager

1. Create a file called **PCAcertchain.pem**. This file contains the **rootca.pem** and the **subca.pem** in one file.

```
cat rootca.pem > PCAcertchain.pem
cat subca.pem >> PCAcertchain.pem
```

2. Import the sub-CA.

      a. Navigate to **Configure > SSL > Internal Root CA > Import Root CA**.

      b. Browse to select the certificate. The certificate must be in X.509 format and base64-encoded.

      c. Browse to select the private key. It must correspond to the certificate you selected in step 2.

      d. Enter, and then confirm, the passphrase.

      e. Click **Import Root CA**.

3. Place the previously created **PCAcertchain.pem** into the **/opt/WCG/sxsuite/conf/CA_default/PCA** folder. See step 1 in this section.

4. Restart Websense Content Gateway.

Backing up your internal Root CA

Always back up the public and private keys of your internal Root CAs before importing or creating new ones. This enables you to return to an earlier version of the certificate, if necessary. In addition, back up any new Root CAs that you import or create.

1. Navigate to **Configure > SSL > Internal Root CA** > **Backup Root CA**.

2. Click **Save Public CA Key** to view or save the public CA key. This public key must be trusted by the users' Web browsers. Consult your network administrator if you do not have the key.

3. Click **Save Private CA Key** to view or save the private CA key. Consult your network administrator if you do not have the key.

Managing certificates

All certificate authorities trusted by Internet Explorer 7 are listed on the tab **Configure > SSL > Certificates > Certificate Authorities**; destination servers (traffic outbound from SSL Manager) can trust the Web servers with these certificates. Note that a small "i" appears before the name of some certificates validated via CRL (certificate revocation lists) or OCSP (online certification status protocol). These certificates provide URLs where you can verify their revocation status. See *Keeping revocation information up to date*, page 136 for information on checking the revocation status of a certificate. SSL Manager checks the revocation status of a certificate for both inbound and outbound traffic.

Click on the name of a certificate authority to:

◆ *View a certificate*, page 129
◆ *Delete a certificate*, page 129
◆ *Change the allow/deny status of a certificate*, page 130

Related topics:

Adding new certificate authorities, page 130
Backing up certificates, page 131
Restoring certificates, page 131

View a certificate

1. Navigate to **Configure > SSL > Certificates** > **Certificate Authorities**.
2. Select the name of the authority whose status you want view.
3. In the pop-up window, select **Click to view certificate**.
4. Follow the directions in the Opening window to open or save the file.

Delete a certificate

1. Navigate to **Configure > SSL > Certificates** > **Certificate Authorities**.
2. Select the name of the certificate authority you want to delete.
3. In the pop-up window, select **Click to delete certificate**.
4. Confirm or deny that you want to delete the certificate.

5. If you confirm that you want to delete the certificate, check that the certificate is no longer listed on **Configure > SSL > Certificates** > **Certificate Authorities**.

Change the allow/deny status of a certificate

1. Navigate to **Configure > SSL > Certificates** > **Certificate Authorities**.

2. Select the name of the authority whose status you want to change.

3. In the pop-up window, select **Click to change status to**. Depending on the status of the certificate, your choice is **allow** or **deny**. If you change the status to deny, a red X appears next to the name of the certificate authority in the certificate authority tree. If you change the status to allow, a green circle appears next to the name of the certificate authority.

Adding new certificate authorities

Use the page **Configure > SSL > Certificates** > **Add Root CA** to manually import additional certificate authorities. Certificates that you import manually have a default status of allow.

> **Important**
>
> It is recommended that you back up your current certificates before making any changes, such as adding or deleting certificates. See *Backing up certificates*, page 131. If you wish to back up your entire Websense Content Gateway configuration, see *Saving and restoring configurations*, page 96.

1. Click **Browse** to navigate through the directory structure to find certificates. Look for files that have a ".cer" extension. The certificate must be in X.509 format and base64-encoded.

2. Click **Add Certificate Authority**.

3. If the import was successful, check that the new certificate is listed on **Configure > SSL > Certificates** > **Certificate Authorities**.

New CAs are also added when users visit a site signed by that authority. These certificates may be allowed or denied. See *Change the allow/deny status of a certificate*, page 130 for additional information.

Related topics:

Backing up certificates, page 131
Restoring certificates, page 131

Backing up certificates

As a precaution, it is recommended that you back up the database containing the CA certificates whenever you make changes, such as adding or deleting a certificate. They can then be restored at a later date.

Backing up certificates also backs up your SSL Manager settings.

Use the page **Configure > SSL > Certificates** > **Backup Certificates** to back up certificates and your SSL Manager settings.

▶ Click **Back Up Configuration to Database**.

To back up not only certificates, but your entire Websense Content Gateway configuration, see *Saving and restoring configurations*, page 96.

Restoring certificates

Restoring certificates also restores the configuration database. However, because revocation lists are updated on a regular basis, they are not restored as part of this process. See *Keeping revocation information up to date*, page 136 for information on updating certificate revocation lists.

Use the page **Configure > SSL > Certificates** > **Restore Certificates** to restore the configuration database, which includes certificates and your SSL Manager settings.

1. Click **Browse** to navigate to the location of the backup certificate database.
2. Click **Restore**. You receive a message telling you that the restore was successful and indicating where the previous certificate database was backed up.

If you are running multiple proxies, use this restore feature to ensure that all the proxies have the same configuration.

Decryption specifics

Configuring SSL Manager for inbound traffic

> Related topics:
>
> *Configuring SSL Manager for outbound traffic*, page 132

Use the page **Configure > SSL > Decryption / Encryption** > **Inbound** to configure how SSL Manager handles inbound traffic. Inbound traffic travels from the browser to SSL Manager, where the content is decrypted and inspected.

1. Select **IP Address** to forward authentication credentials to the next proxy.

2. Select **Send VIA-Header** to add a special header to the HTTP header to describe the proxy chain traffic passed through. This can be helpful in troubleshooting. If you do not want to include a VIA-Header, do not select this box.

3. Under **Protocol Settings**, indicate which protocols you want SSL Manager to support. Supported protocols are SSLv2 and v3, and TLS v1. Select the protocol that your enterprise browser supports; you must select at least one protocol. The default is SSLv2. These settings override the settings for these protocols in the users' browsers.

   You can select different protocols for outbound traffic.

4. Select **Session Cache** if you want to cache keys until the time specified in Session Cache Timeout elapses. This can improve performance. If keys are not cached, each request is negotiated again.

5. Indicate, in seconds, how long keys should be kept in the cache. The default is 300 seconds (5 minutes).

6. The cipher list describes available algorithms and level of encryption between the client and SSL Manager. The default settings indicate to use all available ciphers except the eNULL and the ADH Suite. The strongest cipher (providing the highest level of encryption) is applied first. This can be set to a different level of encryption than for outbound traffic. Setting encryption to a high level for inbound traffic can help ensure the integrity and security of your system.

   Additional cipher settings are:

   - **High** encryption cipher suites: those with key lengths larger than 128 bits, and some cipher suites with 128-bit keys.
   - **Medium** encryption cipher suites: those using 128 bit encryption.
   - **Low** encryption cipher suites: those using 64- or 56-bit encryption algorithms but excluding export cipher suites.

   For inbound requests (requests from a client browser in your organization to SSL Manager), consider using Low encryption to improve performance.

   For more information on ciphers, refer to www.openssl.org/docs.

7. Click **Apply**.

Configuring SSL Manager for outbound traffic

Use the page **Configure > SSL > Decryption / Encryption** > **Outbound** to configure how SSL Manager handles outbound traffic. Outbound traffic travels from SSL Manager to the destination Web server. SSL Manager checks the revocation status of this site's certificate before forwarding re-encrypted data to it.

1. Select **IP Address** to forward authentication credentials from one proxy to the next if there are multiple proxies between SSL Manager and the destination host.

2. Select **Send VIA-Header** to add a special header to the HTTP header to describe the proxy chain traffic passed through. This can be helpful in troubleshooting. If you do not want to include a VIA-Header, do not select this box.

3. Under **Protocol Settings**, indicate which protocols you want SSL Manager to support. Supported protocols are SSLv2 and v3, and TLS v1. Select the protocol that your enterprise browser supports; you must select at least one protocol. The default is SSLv2. These settings override the settings for these protocols in the users' browsers.

   You can select different protocols for inbound traffic.

4. Select **Session Cache** if you want to cache keys until the time specified in Session Cache Timeout elapses. This can improve performance. If keys are not cached, each request is negotiated again.

5. Indicate, in seconds, how long keys should be kept in the cache. The default is 300 seconds (5 minutes).

6. The cipher list describes available algorithms and level of encryption between the client and SSL Manager. The default settings indicate to use all available ciphers except the eNULL and the ADH Suite. The strongest cipher (providing the highest level of encryption) is applied first. This can be set to a different level of encryption than for outbound traffic. Setting encryption to a high level for inbound traffic can help ensure the integrity and security of your system.

   Additional cipher settings are:

   - **High** encryption cipher suites: those with key lengths larger than 128 bits, and some cipher suites with 128-bit keys.

   - **Medium** encryption cipher suites: those using 128 bit encryption.

   - **Low** encryption cipher suites: those using 64- or 56-bit encryption algorithms but excluding export cipher suites.

   For outbound requests (requests from SSL Manager to the destination server that is receiving the encrypted data), consider using one of the higher encryption levels to improve security.

   For more information on ciphers, refer to www.openssl.org/docs.

7. Click **Apply**.

Validating certificates

As part of forwarding outbound traffic, SSL Manager checks the certificates of the destination server. Use these tabs to configure certificate validation and to specify what to do in the case of invalid certificates.

Related topics:

Bypassing verification, page 136
Keeping revocation information up to date, page 136

Configuring validation

1. Navigate to the page **Configure > SSL > Validation > General**.

2. Select or clear **Enable the Certificate Verification Engine** to enable or disable verifying certificates and checking for certificate revocation. If this option is not selected, checking does not occur. It is recommended that, at initial installation, you accept the default and keep the certificate verification engine disabled, and monitor traffic through your network. After a period, you can then enable this feature and specify how SSL Manager should process frequently access sites by creating incidents. See *Managing Web HTTPS site access*, page 138, for additional information.

> **Important**
>
> If you disable the certificate verification engine, you need to provide information only on the following pages:
>
> - Configure > SSL > Decryption / Encryption > Inbound
> - Configure > SSL > Decryption / Encryption > Outbound
> - Configure > SSL > Logging pages
> - Configure > SSL > Customization > Connection Error
>
> If you enable the certificate verification engine, you must restart the proxy for the change to take effect.

3. Select **Deny Certificates where the common name does not match the URL** to define how the proxy handles those cases where the common name of the certificate does not match the URL of the Web server. For example, a certificate from www.company.com does not match the URL www.company.de.

4. Select **Allow wildcard certificates** if you want to accept a single certificate for an entire domain. This means that individual servers within that domain are not verified; they are all included because of the wildcard.

5. Select **No expired or not yet valid certificates** to deny access to sites whose certificates fit that criteria. If this box is not selected, access to those sites is permitted.

> **Note**
>
> Self-signed certificates (certificates without an official certificate authority) are considered invalid and belong in this category.

6. Select **Verify entire certificate chain** to verify all certificates between a certificate and its root certificate authority.

7. Select **Check certification revocation by CRL** to use certificate revocation lists (CRLs) to check a certificate's revocation state.

8. Select **Check certification revocation by OCSP** to use the Online Certificate Status Protocol to check a certificate's revocation state. Currently, OCSP is not used as widely as CRLs, so it is recommended that you indicate CRL in this field and use OCSP as a backup to CRLs.

> ✔ **Note**
>
> Certification revocation lists are used far more widely. It is recommended that you use OCSP in addition to, rather than instead of, CRLs. See *Keeping revocation information up to date*, page 136 for more information on CRLs and OCSP.

9. If you enable checking by both CRLs and OCSP, indicate which method SSL Manager should use first for revocation checking.

10. Indicate whether access should be permitted or denied to sites whose certificate revocation status cannot be determined. If this option is selected, access is also denied to sites whose certificates do not contain CRL or OCSP information. You can see this information when you select a certificate authority and choose view certificate. See *View a certificate*, page 129 for details. This can result in a highly restrictive security policy, with many access denials. You can allow for exceptions by using the incident list to manage access to Web sites. See *Managing Web HTTPS site access*, page 138.

11. For troubleshooting purposes, you can run an external program on incidents. An incident is logged whenever a client receives an access denied message. See *Managing Web HTTPS site access*, page 138 for more information on incidents. Enter the path to the script in this field.

The minimum permissions for running this script should be as follows:

```
chmod 700 /opt/WCG/sxsuite/bin/script.sh
chown Websense /opt/WCG/sxsuite/bin/script.sh
chgrp Websense /opt/WCG/sxsuite/bin/script.sh
```

It is recommended that you copy and paste for following script for help in troubleshooting. It captures the following pieces of information and writes them to a file.

- The account that created the incident
- The client IP or the IP address of the previous proxy if the client IP address is not forwarded
- The ID of the incident as shown in the incident list
- A detailed message on what caused the incident
- The profile within the account that caused the incident
- The host section of the URL that precipitated the incident

```
#!/bin/sh
OUTFILE=/root/incidents.log

date >> $OUTFILE
echo "Account: $SCIP_INCIDENT_ACCOUNT" >> $OUTFILE
echo "Client-IP: $SCIP_INCIDENT_CLIENTIP" >> $OUTFILE
```

```
echo "Incident-ID: $SCIP_INCIDENT_ID" >> $OUTFILE
echo "Detailed Message: $SCIP_INCIDENT_MESSAGE" >> $OUTFILE
echo "Profile: $SCIP_INCIDENT_PROFILE" >> $OUTFILE
echo "Destination Host URL: $SCIP_INCIDENT_REMOTEHOST" >>
$OUTFILE
echo "User: $SCIP_INCIDENT_USER" >> $OUTFILE
echo >> $OUTFILE
```

> **Important**
>
> It is recommended that you do not enter any of the other commands in the **/opt/WCG/sxsuite/bin/** directory in this field, and that you exercise caution if you enter a script other than the one provided above.

Bypassing verification

Use the **Configure > SSL > Validation** > **Verification Bypass** page to enable users to visit a site even if the certificate is invalid.

1. Select **Permit users to visit sites with certificate failure after confirmation** to enable users to proceed to a site after they have been informed that the site has an invalid certificate. If this check box is not selected, users do not have the option to browse to the site.

2. Select **Enable the SSL session cache for bypassed certificates** to store information about bypassed certificates in cache and reuse the connections.

   - If this option is selected, performance is better, but not all users are notified that they are trying to access a site where verification has failed.

   - If this option is not selected, all users are notified about sites that do not have valid certificates, but performance is not as fast.

3. For **Timeout**, specify the period of inactivity that elapses between notifications to users who bypassed this site that the site has an invalid certificate. The default is 6 minutes (360 seconds).

It is recommended that you deploy initially with bypass verification enabled. Then, as the incident rate changes, you can use the incident list to enforce policy. See *Managing Web HTTPS site access*, page 138.

Keeping revocation information up to date

It is recommended that before your site accepts certificates, it checks the status of the certificate to ensure that it has not been revoked. There are two methods of doing this: through CRLs (see *Certificate revocation lists*, page 137) and through OCSP (see *Online certification status protocol (OCSP)*, page 137).

Certificate revocation lists

Use the **Configure > SSL > Validation > Revocation Settings** page to configure how SSL Manager keeps revocation information current. By default, SSL Manager downloads CRLs on a daily basis.

1. For daily downloads of the CRLs, select **Download the CRL at**, and select the time when the CRL download occurs.

2. Click **Apply**.

Use this page as well if you need an immediate CRL update.

1. Click **Update CRL Now** to download the CRLs at a time other than that specified. For example, if your subscription includes SSL Manager, download the CRLs after you install the program.

> ✓ **Note**
> The CRL files can contain thousands of certifications, so downloading CRLs can take some time and consume CPU resources. It is recommended that you download CRLs at a time when Internet traffic on your system is light.

2. Click **View CRL Update Progress** to see the status of the update.

For more information on certificate revocation lists, see RFC 3280.

Online certification status protocol (OCSP)

OCSP is a protocol that operates on a request/response basis. That is, when a site wants to verify the revocation status of a certificate, it sends a request to the CA about the status of the certificate. The CA then responds, confirming the validity (or revocation) of the certificate.

OCSP, because it is dealing with requests, rather than downloading CRLs, can provide improved performance. However, not all CAs provide responses, so CRLs can provide information about the status of more certificates.

SSL Manager enables you to cache OCSP responses about the revocation state of a certificate. Caching responses may be useful in environments with high amounts of SSL traffic and where saving bandwidth is important.

Use the **Configure > SSL > Validation > Revocation Settings** page to configure how SSL Manager keeps revocation information current.

1. Specify, in days, how long OCSP data should be cached. If you do not want to cache OCSP data, enter **0**. The maximum is 1000 days

2. Click **Apply**.

For more information on OCSP, see RFC 2560.

Managing Web HTTPS site access

These tabs can help you manage access to Web sites and can aid the Help Desk in troubleshooting access issues. Entries and changes made to this page are saved in the SSL Manager database.

When a client receives an access denial message because the Web site does not comply with security policies, SSL Manager generates an incident. See *Viewing incidents*, page 138.

If you want to specify how SSL Manager treats a particular site, you can add that to the incident list as well. See *Adding Web sites to the incident list*, page 141.

> Related topics:
>
> *Changing the status of an incident*, page 140
> *Deleting an incident*, page 140
> *Changing the text of a message*, page 140
> *Viewing incident details*, page 140

Viewing incidents

Use the **Configure > SSL > Incidents > Incident List** page to see a report of those times when clients received an access denial message. You can use the fields in this report to determine how SSL Manager treats requested access to a site in the future.

▶ To view incidents:

- To view a specific incident, enter the ID number and click **Search**.

- Click **Show All** to see the complete list of incidents.

The incident report

You can sort on any column by clicking on the small triangle next to the column heading.

The incident report contains these fields:

Field	Description
ID	Assigned by the system, this is the incident ID number, also called the Ticket ID. The Help Desk can ask the user for the Ticket ID in the error message and quickly retrieve it from the URL Incident List. The end user sees the Ticket ID and a denial message.

Status	Determines how SSL Manager will treat this Web site in the future. Four conditions are possible:
	
	• Allow
	
	  Users can access the site even if the certificate is not valid. Traffic is decrypted, and certificate checking is disabled.
	
	• Blacklisted
	
	  The site is completely blocked. Users cannot access this site even if the verification bypass has been selected. See *Bypassing verification*, page 136 for details on this option.
	
	• Block
	
	  Access to the specific Web site is blocked. The user will always get this message when trying to access this site.
	
	• Tunnel
	
	  A site that is white-listed (tunneled). Traffic is not decrypted and SSL Manager does not check the certificate. Tunneling can be used to bypass inspection of trusted sites and improve performance.
	
	You can change the status of a Web site via the drop-down box in the Action column.

Type	Indicates whether the site was added based on its URL or its certificate. It is recommended that you add sites to the incident list by certificate. See *Adding Web sites to the incident list*, page 141.
URL	The URL of a site whose certificate could not be validated.
Message	Enables you to edit the error message. See *Changing the text of a message*, page 140 for information on customizing error messages. The pencil and the magnifying glass each represent links. See *Viewing incident details*, page 140 for details on these links.
Action	Enables you to change the status of an incident. Also allows you to delete the incident. See *Deleting an incident*, page 140.

Changing the status of an incident

When you change the status of an incident, you are changing how SSL Manager will treat the listed URL in the future.

1. Navigate to **Configure > SSL > Incidents** > **Incident List**.
2. Select one of the following from the drop-down list in the Actions column. See *The incident report*, page 138 for an explanation of these options.
   - Tunnel
   - Block
   - Blacklist
   - Allow
3. Click **Go**. The icon in the Status column changes to reflect the new status.

Deleting an incident

1. Navigate to **Configure > SSL > Incidents** > **Incident List**.
2. Select the incident to delete. If the incident is not visible, you can search by ID. See *Viewing incidents*, page 138.
3. In the Action column, select **Delete** from the Action drop-down list, and then click **Go**.

Changing the text of a message

1. Navigate to **Configure > SSL > Incidents** > **Incident List**.
2. Locate the incident you want to examine more closely. See *Viewing incidents*, page 138.
3. Click the pencil to open a window where you can change the text of this error message. For example, the Help Desk can add more detail to an error message.
4. Click **Submit** when the new text is complete, or click **Close Window** if you are not making any changes.

Viewing incident details

1. Navigate to **Configure > SSL > Incidents** > **Incident List**.
2. Locate the incident you want to examine more closely. See *Viewing incidents*, page 138.
3. Click the magnifying glass to see additional details about the incident, such as the:
   - Description (this is the message that appears in the incident listing)
   - Time the incident was created
   - Time the incident was modified
   - Incident count (how many times users have tried to access this site)

Adding Web sites to the incident list

Use **Configure > SSL > Incidents** > **Add Website** page to specify sites that you want to allow, blacklist, or tunnel. Sites that are added manually are assigned chronological Ticket IDs. These appear on the incident list. See *Viewing incidents*, page 138.

1. Enter the URL of the site you are adding to the Incident List.
2. Select either **By Certificate** or **By URL**.
   - **By Certificate** provides greater security. If you add a Web site by certificate, clients cannot bypass the policy by using the IP address rather than the URL. If you select "By Certificate," SSL Manager retrieves the server certificate and adds the site to the incident list. Every new site is whitelisted by default. To blacklist the site, find its entry in the URL Incident List and select the desired action. See *Viewing incidents*, page 138.

     If sites are blocked by certificates, wildcard certificates are not accepted, even if the common name is recognized.
   - Select **By URL** only if you want to allow the site.
3. In the Action drop-down list, specify if the site should be added with Tunnel, Allow, or Blacklist status. See *The incident report*, page 138 for details.
4. Click **Add URL**.

It is recommended that you manually add sites to the incident list after you have monitored your network traffic for a period of time, with the certificate verification engine disabled. (See *Configuring validation*, page 133.) This enables you to improve performance by tunneling trusted sites and blocking those you know should not be accessed. See *The incident report*, page 138 for information about assigning a status, such as tunneling, to a site and incident.

Client certificates

For security, the destination server may request a client certificate.

> Related topics:
>
> *Importing client certificates*, page 142
> *When a client certificate is always required: the hostlist*, page 142
> *Deleting client certificates*, page 142

When a client certificate is requested

1. Navigate to **Configure > SSL > Client Certificates** > **General**.

2. Select **Tunnel** or **Create incident** to specify how SSL Manager should handle that certificate and site. You must choose Create incident if you want any disposition other than tunnel (white listing). White listing will always provide the certificate to the server. See *The incident report*, page 138 for a listing of possible dispositions.

3. Click **Apply**.

Importing client certificates

Use the **Configure > SSL > Client Certificates** > **Import** page to import certificates from the organization represented by the client.

> **Important**
> Remember to use only X.509-formatted, base64-encoded certificates.

1. Enter the name of the client certificate.

2. Enter the public key for the certificate. You may need to check with your network administrator for the key.

3. Enter the private key for the certificate. You may need to check with your network administrator for the key.

4. Enter, and then confirm, the passphrase. It is recommended that you use a strong passphrase, with a combination of numbers, characters, and upper- and lower-case letters. You may need to check with your network administrator for the passphrase.

5. Click **Import**.

When a client certificate is always required: the hostlist

Use the **Configure > SSL > Client Certificates** > **Hostlist** page to list those destination servers that always require a client certificate. Be sure to import the certificate before adding it to the hostlist. See *Importing client certificates*, page 142.

1. Enter the URL of the destination server that requires the client certificate.

2. In the **Client Certificate** drop-down list, select the name of the client certificate. Only certificates you have already imported appear in this list.

3. Click **Add**.

Deleting client certificates

Use the **Configure > SSL > Client Certificates > Manage Certificates** page to delete imported client certificates.

1. Select the certificate you want to delete.

2. Click **Delete**.

Configuring logging

SSL Manager creates 2 types of log files.

◆ Activity logs. These logs monitor SSL Manager activity and include messages at a level specified in the user interface

◆ Access logs

You can log activity for both inbound (client to SSL Manager) and outbound (SSL Manager to server) traffic. You have the option of logging data to the system log (syslog) or to a file.

> Related topics:
>
> *How long should log files be kept?*, page 144
> *How big can log files grow?*, page 144
> *What fields should appear in the access log files?*, page 144

Use the **Configure > SSL > Logging > General** page to specify the name and location of log files.

1. For *inbound* traffic, select the type of log files you want to keep. For activity logs, you are specifying the level of detail in the log.

2. Enter a number from 1 to 7 to indicate the level of detail you want logged. Note that each level provides more information; level 7 is the most verbose The levels of logging and granularity are:

0 (emergency)	Log panic situations
1 (alert)	Log conditions that should be corrected immediately, such as a corrupted system file
2 (critical)	Log conditions such as device failures
3 (normal)	Log errors
4 (warning)	Log warnings
5 (notice)	Log conditions that are not error conditions, but may still require attention
6 (information)	Log informational messages
7 (debugging)	Log debugging information. Level 7 includes the most log output.

3. Indicate if log data should go to the syslog or to a file.

4. Repeat Step 2 and Step 3 for the access log file.

5. For *outbound* traffic, repeat Step 2 through Step 4.

6. Click **Apply**.

Logs are written to /**opt/WCG/sxsuite/log**.

How long should log files be kept?

A new set of log files is created every 24 hours. By default, this occurs at midnight. This rotation happens regardless of the size of the log file. In addition, the log file is rotated if it reaches its maximum size before the scheduled rotation. In that case, the scheduled rotation still takes place at midnight. See *How big can log files grow?*, page 144 for information on specifying the maximum log size.

Use the **Configure > SSL > Logging > Options** page to specify how long to keep log files.

1. Specify, in days, how long log files should be kept. The default is 3.
2. Set any additional options on this page and then click **Apply**.

How big can log files grow?

Log files are rotated every night at midnight. However, a new log file is started when the file reaches its specified maximum size, even if this is before the scheduled daily rotation. Because the size of log files is checked every minute, it is possible that a log file may be larger than its maximum size for a brief period.

When a log file reaches its maximum size, it is saved with an extension of ".*x*" (where *x* is 1, 2, or 3, etc.), and a new file is started. If this should happen multiple times in a 24-hour period, you must indicate how many files (generations) should be kept. See *How long should log files be kept?*, page 144 for information on log rotation.

Use the **Configure > SSL > Logging > Options** page to specify how large log files can grow.

1. Indicate, in KB, the maximum size for log files. The default is 50,000 KB.
2. For generations, indicate how many log files should be kept if the file reaches its maximum size multiple times before daily rotation. Once this number is reached and new log files are created, the oldest log file is deleted. The default is 3 generations.
3. Set any additional options on this page and then click **Apply**.

What fields should appear in the access log files?

Use the **Configure > SSL > Logging > Options** page to add or delete fields to the log file.

1. Delete or add fields in the Access log file customization box. The fields are:

time_stamp	The timestamp in the following format:[YYYY.MM.DD HH:MM:SS]
time_of_day	The timestamp in raw format: Sec.mSec starting from 1st Jan 1970 UTC
src_ip	The client's IP Address
auth_user	The user who has been authenticated
account	The account the user belongs to
profile	The user's profile
req_line	The Request in the following format: "method path protocol/version.subversion". For example: GET / HTTP/1.1
status_code	The HTTP status response code sent by the Web server
user_agent	The name of the client browser
referer	The host section of the URL
content_type	Content, such as HTML, text, image, etc.
content_length	In bytes
server_host	IP address of the Web server
bytes_from_client	Bytes transferred from the client to SSL Manager
bytes_to_client	Bytes transferred from SSL Manager to the client
bytes_from_server	Bytes transferred from the Web server to SSL Manager
bytes_to_server	Bytes transferred from SSL Manager to the Webserver

2. Click **Apply**.

Customizing messages

You can customize the message users receive when:

◆ They are trying to connect to a site that has an invalid certificate. See *Certificate validation failed*, page 146.

◆ There is a connection failure. See *Connection failure*, page 146.

The following variables are available within the message templates.

%P	Protocol (HTTP or HTTPS)
%h	The IP address and port of the host of the proxy that generated the message
%o	The IP address of the host of the proxy that generated the message
%H	Remote hostname of the request
%t	Time
%s	Name of the SSL Manager server
%u	Complete URL
$$DETAILS	Detailed error message
$$TICKET_ID	The ID number of the incident.

Certificate validation failed

Use the **Configure > SSL > Customization > Certificate Failure** page to customize the message users receive when certificate validation fails.

✔ **Note**
You may find it helpful to click **Preview** to see how the default message appears.

1. Edit the HTML code in the window to reflect your message. See *Customizing messages*, page 145 for a listing of variables you can use in the message.
2. Click **Preview** to see your changes.
3. Repeat steps 1 and 2 until the message appears appropriately.
4. Click **Apply** to confirm your edits or **Cancel** to return to the original message.

Connection failure

Use the **Configure > SSL > Customization > Connect Error** page to customize the message users receive when SSL Manager is unable to connect to the destination Web server.

✔ **Note**
You may find it helpful to click **Preview** to see how the default message appears.

1. Edit the text in the window to reflect your message. See *Customizing messages*, page 145 for a listing of variables you can use in the message.

2. Click **Preview** to see your changes.
3. Repeat steps 1 and 2 until the message appears appropriately.
4. Click **Apply** to confirm your edits or **Cancel** to return to the original message.

14 | Working With Log Files

Websense Content Gateway keeps 3 types of log files:

◆ *System log files* record system information, which includes messages about the state of Websense Content Gateway and any errors or warnings that it produces. This information might include a note that event log files were rolled, a warning that cluster communication timed out, or an error indicating that Websense Content Gateway was restarted. (Websense Content Gateway posts alarms signifying error conditions on Websense Content Manager; see *Working with alarms*, page 84, for details.)

In *UNIX*, all system information messages are logged with the system-wide logging facility **syslog** under the daemon facility. The **syslog.conf** configuration file (stored in the **/etc** directory) specifies where these messages are logged. A typical location is **/var/log/messages**.

The syslog process works on a system-wide basis, so it is the single repository for messages from all Websense Content Gateway processes, including content_gateway, content_manager, and content_cop.

Each log entry in the log contains information about the date and time the error was logged, the host name of the proxy server that reported the error, and a description of the error or warning.

See *Websense Content Gateway error messages*, page 345, for a list of the system information messages that Websense Content Gateway logs.

◆ *Error log files* record information about why a transaction was in error.

◆ *Event log files* (also called *access log files*) record information about the state of each transaction that Websense Content Gateway processes.

Websense Content Gateway creates both error and event log files and records system information in system log files. You can disable event logging and/or error logging. It is recommended that you log errors only or disable logging during peak usage hours.

▶ On the **Configure > Subsystems > Logging** tab, select one of the following options: **Log Transactions and Errors**, **Log Transactions Only**, **Log Errors Only**, or **Disabled**.

Event log files

Event log files record information about every request that Websense Content Gateway processes. By analyzing the log files, you can determine how many people use the proxy cache, how much information each person requested, what pages are most popular, and so on.

Websense Content Gateway supports several standard log file formats, such as Squid and Netscape, and user-defined custom formats. You can analyze the standard format log files with off-the-shelf analysis packages. To help with log file analysis, you can separate log files so that they contain information specific to protocol or hosts. You can also configure Websense Content Gateway to roll log files automatically at specific intervals during the day.

The following sections describe how to:

◆ Manage your event log files

   You can choose a central location for storing log files, set how much disk space to use for log files, and set how and when to roll log files. See *Managing event log files*, page 151.

◆ Choose different event log file formats

   You can choose which standard log file formats you want to use for traffic analysis (for example, Squid or Netscape). Alternatively, you can use the Websense Content Gateway custom format, which is XML-based and enables you to institute more control over the type of information recorded in log files. See *Event log file formats*, page 153.

◆ Roll event log files automatically

   You can configure Websense Content Gateway to roll event log files at specific intervals during the day so that you can identify and manipulate log files that are no longer active. See *Rolling event log files*, page 158.

◆ Separate log files according to protocols and hosts

You can configure the proxy to create separate log files for ICP and HTTP/FTP transactions. You can also configure the proxy to generate separate log files for different protocols based on the host. See *Splitting event log files*, page 161.

◆ Collate log files from different nodes

You can designate one or more nodes on the network to serve as log collation servers. These servers, which might either be stand-alone or part of Websense Content Gateway, enable you to keep all logged information in well-defined locations. See *Collating event log files*, page 164.

◆ View statistics about the logging system

Websense Content Gateway provides statistics about the logging system. You can access the statistics through Websense Content Manager or through the command line interface. See *Viewing logging statistics*, page 167.

◆ View log files

You can view the system, event, and error log files that Websense Content Gateway creates. You can view an entire log file, a specified last number of lines in the log file, or all lines that contain a specified string.

◆ Interpret log file entries for the standard log file formats. See *Example event log file entries*, page 169.

Managing event log files

You can manage your event log files and control where they are located, how much space they can consume, and how low disk space in the logging directory is handled.

Choosing the logging directory

By default, Websense Content Gateway writes all event log files in the `logs` directory, which is located in the directory where you installed Websense Content Gateway. To use a different directory, see *Setting log file management options*, page 152.

Controlling logging space

You can control the amount of disk space that the logging directory can consume. This allows the system to operate smoothly within a specified space window for a long period of time.

After you establish a space limit, Websense Content Gateway continues to monitor the space in the logging directory. When the free space dwindles to the headroom limit (see *Setting log file management options*, page 152), Websense Content Gateway enters a low space state and takes the following actions:

◆ If the autodelete option (discussed in *Rolling event log files*, page 158) is *enabled*, Websense Content Gateway identifies previously rolled log files (log files with a `.old` extension) and starts deleting files one by one—beginning with the oldest

file—until it emerges from the low state. Websense Content Gateway logs a record of all files it deletes in the system error log.

◆ If the autodelete option is *disabled* or there are not enough old log files to delete for the system to emerge from its low space state, Websense Content Gateway issues a warning and continues logging until space is exhausted. Websense Content Gateway resumes event logging when enough space becomes available for it to exit its low space state. You can make space available by removing files from the logging directory or by increasing the logging space limit.

You can run a `cron` script in conjunction with Websense Content Gateway to automatically remove old log files from the logging directory (before Websense Content Gateway enters the low space state) and relocate them to a temporary partition. Once the files are relocated, you can run log analysis scripts on them, and then you can compress the logs and move them to an archive location or delete them.

Setting log file management options

1. Navigate to **Configure > Subsystems > Logging**.

2. In the **Log Directory** field, enter the path to the directory in which you want to store event log files. This can be an absolute path or a path relative to the directory in which Websense Content Gateway is installed. The default directory is **logs**, located in the Websense Content Gateway installation directory.

> ✔ **Note**
> The directory you specify must already exist.
>
> The Websense user must have read/write permissions for the directory storing the log files.

3. In the **Limit** field of the **Log Space** area, enter the maximum amount of space you want to allocate to the logging directory. The default value is 20480 MB.

> ✔ **Note**
> All files in the logging directory contribute to the space used, even if they are not log files.

4. In the **Headroom** field, enter the tolerance for the log space limit. The default value is 100 MB.

   If the **Auto-Delete Rolled Files** option is enabled in the **Log Rolling** section, autodeletion is triggered when the amount of free space available in the logging directory is less than the headroom. For information about log file rolling, see *Rolling event log files*, page 158.

5. Click **Apply**.

Event log file formats

Websense Content Gateway supports the following log file formats:

◆ *Standard formats*, such as Squid or Netscape (see *Using standard formats*, page 153)

◆ the Websense Content Gateway *custom format* (see *Custom format*, page 153)

In addition to the standard and custom log file format, you must choose whether to save log files in *binary* or *ASCII*. See *Choosing binary or ASCII*, page 156.

> ❗ **Important**
> Event log files consume a large amount of disk space. Creating log entries in multiple formats at the same time can consume disk resources very quickly and affect proxy performance.

Using standard formats

The standard log formats include Squid, Netscape Common, Netscape Extended, and Netscape Extended-2.

The standard log file formats can be analyzed with a wide variety of off-the-shelf log-analysis packages. You should use one of the standard event log formats unless you need information that these formats do not provide. See *Custom format*, page 153.

By default, Websense Content Gateway is configured to use the Netscape Extended log file format only.

Setting standard log file format options

1. Navigate to **Configure > Subsystems > Logging > Formats**.
2. Enable the format you want to use.
3. Select the log file type (**ASCII** or **binary**).
4. In the **Filename** field, enter the name you want to use for your event log files.
5. In the **Header** field, enter a text header that appears at the top of the event log files. Leave this field blank if you do not want to use a text header.
6. Click **Apply**.
7. Click **Restart** on **Configure > My Proxy > Basic > General**.

Custom format

The XML-based custom log format is more flexible than the standard log file formats, giving you more control over the type of information in your log files. Create a custom log format if you need data for analysis that is not available in the standard formats.

You can decide what information to record for each Websense Content Gateway transaction and create filters to define which transactions to log.

The heart of the custom logging feature is an XML-based logging configuration file (**logs_xml.config**) that enables you to create modular descriptions of logging objects. The `logs_xml.config` file uses three types of objects to create custom log files:

◆ The `LogFormat` defines the content of the log file using `printf`-style format strings.

◆ The `LogFilter` defines a filter so that you include or exclude certain information from the log file.

◆ The `LogObject` specifies all the information needed to produce a log file. For example:

■ The name of the log file (required).

■ The format to be used (required). This can be a standard format (Squid or Netscape) or a previously defined custom format (a previously defined `LogFormat` object).

■ The file mode (ASCII, Binary, or ASCII_PIPE). The default is ASCII.

The ASCII_PIPE mode writes log entries to a UNIX named pipe (a buffer in memory). Other processes can then read the data using standard I/O functions. The advantage of using this option is that Websense Content Gateway does not have to write to disk, freeing disk space and bandwidth for other tasks.

> ✔ **Note**
>
> When the buffer is full, Websense Content Gateway drops log entries and issues an error message indicating how many entries were dropped. Websense Content Gateway writes only complete log entries to the pipe; therefore, only full records are dropped.

■ Any filters you want to use (previously defined `LogFilter` objects).

■ The collation servers that are to receive the log files.

■ The protocols you want to log (if the protocols tag is used, Websense Content Gateway logs only transactions from the protocols listed; otherwise, all transactions for all protocols are logged).

■ The origin servers you want to log (if the servers tag is used, Websense Content Gateway logs only transactions for the origin servers listed; otherwise, transactions for all origin servers are logged).

■ The header text you want the log files to contain. The header text appears at the beginning of the log file, just before the first record.

- The log file rolling options.

> ✓ **Note**
>
> To generate a custom log format, you must specify at least
> one `LogObject` definition. One log file is produced for
> each `LogObject` definition. You can create a custom log
> format by through Websense Content Manager or by
> editing a configuration file.

1. On **Configure** > **Subsystems** > **Logging** > **Custom**, enable the **Custom Logging**
    option.
2. The **Custom Log File Definitions** area displays the `logs_xml.config` file. Add
    `LogFormat`, `LogFilter`, and `LogObject` specifications to the configuration
    file.

    For detailed information about the **logs_xml.config** file and associated object
    specifications, see *logs_xml.config*, page 272.
3. Click **Apply**.

Creating summary log files

Websense Content Gateway performs several hundred operations per second;
therefore, event log files can grow quite large. Using SQL-like aggregate operators,
you can configure Websense Content Gateway to create summary log files that
summarize a set of log entries over a specified period of time. This can reduce the size
of the log files generated.

You generate a summary log file by creating a `LogFormat` object in the XML-based
logging configuration file (**logs_xml.config**) using the following SQL-like aggregate
operators:

◆ COUNT

◆ SUM

◆ AVERAGE

◆ FIRST

◆ LAST

You can apply each of these operators to specific fields, requesting it to operate over a
specified interval.

Summary log files represent a trade-off between convenience and information
granularity. Since you must specify a time interval during which only a single record
is generated, you can lose information. If you want the convenience of summary logs
and need the detail of a conventional log file, consider creating and enabling two
custom log formats—one using aggregate operators and the other not using aggregate
operators.

To create a summary log file format:

1. Navigate to **Configure** > **Subsystems** > **Logging** > **Custom** to display the **logs_xml.config** file.

2. Define the format of the log file as follows:

```
<LogFormat>
    <Name = "summary"/>
    <Format = "%<operator(field)> : %<operator(field)>"/>
    <Interval = "n"/>
</Format>
```

where:

operator is one of the five aggregate operators (COUNT, SUM, AVERAGE, FIRST, LAST). You can specify more than one operator in the format line.

field is the logging field that you want to aggregate.

n is the interval in seconds between summary log entries.

For more information, see .

For example, the following format generates one entry every 10 seconds, with each entry summarizing the time stamp of the last entry of the interval, a count of the number of entries seen within that 10-second interval, and the sum of all bytes sent to the client:

```
<LogFormat>
    <Name = "summary"/>
    <Format = "%<LAST(cqts)> : %<COUNT(*)> :
    %<SUM(psql)>"/>
    <Interval = "10"/>
</Format>
```

> **❗ Important**
>
> You cannot create a format specification that contains both aggregate operators and regular fields. For example, the following specification would be invalid:
>
> ```
> <Format = "%<LAST(cqts)> : %<COUNT(*)> :
> %<SUM(psql)> : %<cqu>"/>
> ```

3. Define a `LogObject` that uses this format.

4. Click **Apply**.

Choosing binary or ASCII

You can configure Websense Content Gateway to create event log files in either of the following:

◆ **ASCII**: these files can be processed using standard, off-the-shelf log-analysis tools. However, Websense Content Gateway must perform additional processing to create the files in ASCII, resulting in an increase in overhead. Also, ASCII files tend to be larger than the equivalent binary files. ASCII log files have a `.log` filename extension by default.

◆ **Binary**: these files generate lower system overhead, as well as generally occupying less space on the disk, depending on the type of information being logged. You must, however, use a converter application before you can read or analyze these files using standard tools. Binary log files use a .blog filename extension by default.

While binary log files typically require less disk space, this is not always the case. For example, the value 0 (zero) requires only one byte to store in ASCII but requires four bytes when stored as a binary integer. If you define a custom format that logs IP addresses, a binary log file would require only four bytes of storage per 32-bit address. However, the same IP address stored in dot notation would require around 15 characters (bytes) in an ASCII log file.

For standard log formats, you select **Binary** or **ASCII** on the **Configure > Subsystems > Logging > Formats** tab in Websense Content Manager. See *Setting standard log file format options*, page 153. For the custom log format, you specify ASCII or Binary mode in the LogObject. Refer to *Custom format*, page 153.

> **Note**
>
> ✓ For custom log files, in addition to the ASCII and Binary options, you can also write log entries to a UNIX named pipe (a buffer in memory). Other processes can then read the data using standard I/O functions. The advantage of using this option is that Websense Content Gateway does not have to write to disk, freeing disk space and bandwidth for other tasks. In addition, writing to a pipe does not stop when logging space is exhausted because the pipe does not use disk space. See *logs_xml.config*, page 272, for more information about the ASCII_PIPE option.

Before selecting ASCII versus binary for your log files, consider the type of data that will be logged. Try logging for one day using ASCII and then one day using binary. Assuming that the number of requests is roughly the same for both days, you can calculate a rough metric comparing the two formats.

Using logcat to convert binary logs to ASCII

You must convert a binary log file to ASCII before you can analyze it using standard tools.

1. Change to the directory containing the binary log file.
2. Make sure that the logcat utility is in your path.
3. Enter the following command:

```
logcat options input_filename...
```

The following table describes the command-line options.

Option	Description
-o output_file	Specifies where the command output is directed.

-a	Automatically generates the output filename based on the input filename. If the input is from stdin, this option is ignored.
	For example:
	`logcat -a squid-1.blog squid-2.blog squid-3.blog`
	generates
	`squid-1.log, squid-2.log, squid-3.log`

-S	Attempts to transform the input to Squid format, if possible.
-C	Attempts to transform the input to Netscape Common format, if possible.
-E	Attempts to transform the input to Netscape Extended format, if possible.
-2	Attempt to transform the input to Netscape Extended-2 format, if possible.

> ✔ **Note**
> Use only one of the following options at any given time:
> -S, -C, -E, or -2.

If no input files are specified, logcat reads from the standard input (stdin). If you do not specify an output file, logcat writes to the standard output (stdout).

For example, to convert a binary log file to an ASCII file, you can use the logcat command with either of the following options:

```
logcat binary_file > ascii_file
logcat -o ascii_file binary_file
```

The binary log file is not modified by this command.

Rolling event log files

Websense Content Gateway provides automatic log file rolling. This means that at specific intervals during the day, Websense Content Gateway closes its current set of log files and opens new log files.

Log file rolling offers the following benefits:

◆ It defines an interval over which log analysis can be performed.

◆ It keeps any single log file from becoming too large and assists in keeping the logging system within the specified space limits.

◆ It provides an easy way to identify files that are no longer being used so that an automated script can clean the logging directory and run log analysis programs.

You should roll log files several times a day. Rolling every six hours is a good guideline to follow.

Rolled log filename format

Websense Content Gateway provides a consistent name format for rolled log files that allows you to identify log files.

When Websense Content Gateway rolls a log file, it saves and closes the old file and starts a new file. Websense Content Gateway renames the old file to include the following information:

◆ The format of the file (for example, **squid.log**).

◆ The host name of the Websense Content Gateway server that generated the log file.

◆ Two timestamps separated by a hyphen (-). The first time stamp is a lower bound for the time stamp of the first record in the log file. The lower bound is the time when the new buffer for log records is created. Under low load, the first time stamp in the filename can be different from the timestamp of the first entry. Under normal load, the first time stamp in the filename and the time stamp of the first entry are similar.

The second time stamp is an upper bound for the time stamp of the last record in the log file (this is normally the rolling time).

◆ The suffix .old, which makes it easy for automated scripts to find rolled log files.

The timestamps have the following format:

```
%Y%M%D.%Hh%Mm%Ss-%Y%M%D.%Hh%Mm%Ss
```

The following table describes the format:

Code	Definition	Example
%Y	The year in four-digit format	2000
%M	The month in two-digit format, from 01-12	07
%D	The day in two-digit format, from 01-31	19
%H	The hour in two-digit format, from 00-23	21
%M	The minute in two-digit format, from 00-59	52
%S	The second in two-digit format, from 00-59	36

The following is an example of a rolled log filename:

```
squid.log.mymachine.20000912.12h00m00s-
20000913.12h00m00s.old
```

In this example, the file is squid log format and the host machine is mymachine. The first time stamp indicates a date and time of year 2000, month September, and day 12 at 12:00 noon. The second time stamp indicates a date and time of year 2000, month September, and day 13 at 12:00 noon. At the end, the file has a .old suffix.

The logging system buffers log records before writing them to disk. When a log file is rolled, the log buffer might be partially full. If so, the first entry in the new log file will have a time stamp earlier than the time of rolling. When the new log file is rolled, its first time stamp will be a lower bound for the time stamp of the first entry. For example, suppose logs are rolled every three hours, and the first rolled log file is:

```
squid.log.mymachine.19980912.12h00m00s-
19980912.03h00m00s.old
```

If the lower bound for the first entry in the log buffer at 3:00:00 is 2:59:47, the next log file, when rolled, will have the following time stamp:

```
squid.log.mymachine.19980912.02h59m47s-
19980912.06h00m00s.old
```

The contents of a log file are always between the two timestamps. Log files do not contain overlapping entries, even if successive timestamps appear to overlap.

Rolling intervals

Log files are rolled at specific intervals relative to a given hour of the day. Two options control when log files are rolled:

◆ The offset hour, which is an hour between 0 (midnight) and 23

◆ The rolling interval

Both the offset hour and the rolling interval determine when log file rolling starts. Rolling occurs every rolling interval *and* at the offset hour.

For example, if the rolling interval is six hours and the offset hour is 0 (midnight), the logs roll at midnight (00:00), 06:00, 12:00, and 18:00 each day. If the rolling interval is 12 hours and the offset hour is 3, logs roll at 03:00 and 15:00 each day.

Setting log file rolling options

1. Navigate to **Configure** > **Subsystems** > **Logging > General**.

2. In the **Log Rolling** section, ensure the **Log Rolling** option is enabled (the default).

3. In the **Offset Hour** field, enter a specific time each day you want log file rolling to take place. Websense Content Gateway forces the log file to be rolled at the offset hour each day.

   You can enter any hour in the range 0 (midnight) to 23.

4. In the **Interval** field, enter the amount of time Websense Content Gateway enters data in the log files before rotation takes place.

The minimum value is 300 seconds (five minutes). The maximum value is 86400 seconds (one day).

> **Note**
> If you start Websense Content Gateway within a few minutes of the next rolling time, rolling may not occur until the following rolling time.

5. Ensure the **Auto-Delete Rolled Files** option is enabled (the default). This enables auto deletion of rolled log files when available space in the log directory is low.

   Auto deletion is triggered when the amount of free space available in the log directory is less than the headroom.

6. Click **Apply**.

> **Note**
> You can fine tune log file rolling settings for a custom log file in the `LogObject` specification in the **logs_xml.config** file. The custom log file uses the rolling settings in its `LogObject`, which override the default settings you specify in Websense Content Manager or the **records.config** file described above.

Splitting event log files

By default, Websense Content Gateway uses standard log formats and generates log files that contain HTTP, FTP, NNTP, and ICP transactions in the same file. However, you can enable log splitting if you prefer to log transactions for different protocols in separate log files.

> **Note**
> If you are using a standard log file format (such as Squid or Netscape), Websense Content Gateway always records HTTP and FTP transactions in the same log file. You cannot generate separate log files for transactions using these two protocols.

ICP log splitting

When ICP log splitting is enabled, Websense Content Gateway records ICP transactions in a separate log file with a name that contains `icp`. For example, if you enable the Squid format, all ICP transactions are recorded in the **squid-icp.log** file.

When you disable ICP log splitting, Websense Content Gateway records all ICP transactions in the same log file as HTTP and FTP transactions.

NNTP log splitting

When NNTP log splitting is enabled, Websense Content Gateway records all commands it sends to the Network News Transfer Protocol (NNTP) server in a separate log file. This allows you to log transactions from incoming news feeds. The log filename contains "nntp". For example, if you enable the Squid format, all NNTP transactions are recorded in the squid-nntp.log file.

When you disable NNTP log splitting, Websense Content Gateway records all NNTP transactions in the same log file as HTTP and FTP transactions.

HTTP host log splitting

HTTP host log splitting enables you to record HTTP and FTP transactions for different origin servers in separate log files. When HTTP host log splitting is enabled, Websense Content Gateway creates a separate log file for each origin server listed in the **log_hosts.config** file (see *Editing the log_hosts.config file*, page 163).

When ICP, NNTP, and HTTP host log splitting are all enabled, Websense Content Gateway generates separate log files for HTTP/FTP transactions, based on the origin server, and places all ICP and NNTP transactions in their own respective log files.

For example, if the **log_hosts.config** file contains the two origin servers `uni.edu` and `company.com`, and the Squid format is enabled, Websense Content Gateway generates the following log files:

Log Filename	Description
`squid-uni.edu.log`	All HTTP and FTP transactions for `uni.edu`
`squid-company.com.log`	All HTTP and FTP transactions for `company.com`
`squid-icp.log`	All ICP transactions for all hosts
`squid.log`	All HTTP and FTP transactions for other hosts

If you disable ICP and NNTP log splitting, these transactions are placed in the same log file as HTTP and FTP transactions. Using the previous example hosts and assuming the Squid log format is used, Websense Content Gateway generates these log files:

Log Filename	Description
`squid-uni.edu.log`	All entries for `uni.edu`

Log Filename	Description
`squid-company.com.log`	All entries for `company.com`
`squid.log`	All other entries

Websense Content Gateway also enables you to create XML-based custom log formats that offer even greater control over log file generation based on protocol and host name. See *Custom format*, page 153.

Setting log splitting options

1. Navigate to **Configure** > **Subsystems** > **Logging > Splitting**.

2. Enable the **Split ICP Logs** option to record all ICP transactions in a separate log file. Disable the **Split ICP Logs** option to record all ICP transactions in the same log file as HTTP/FTP transactions.

3. Enable the **Split NNTP Logs** option to record all NNTP transactions in a separate log file. Disable the **Split NNTP Logs** option to record all NNTP transactions in the same log file as HTTP/FTP transactions.

4. Enable the **Split Host Logs** option to record all HTTP and FTP transactions for each origin server listed in the **log_hosts.config** file in a separate log file. Disable the **Split Host Logs** option to record all HTTP and FTP transactions for each origin server listed in the **log_hosts.config** file in the same log file.

5. Click **Apply**.

Editing the log_hosts.config file

The default **log_hosts.config** file is located in the Websense Content Gateway **config** directory (default location is **/opt/WCG/config**). To record HTTP and FTP

transactions for different origin servers in separate log files, you must specify each origin server's host name on a separate line in the file.

> **Note**
>
> You can specify keywords in the **log_hosts.config** file to record in a separate log file all transactions from origin servers that contain the specified keyword in their names. For example, if you specify the keyword `sports`, Websense Content Gateway records all HTTP and FTP transactions from `sports.yahoo.com` and `www.foxsports.com` in a log file called `squid-sport.log` (if the Squid format is enabled).

> **Note**
>
> If Websense Content Gateway is clustered and if you enable log file collation, it is recommended that you use the same **log_hosts.config** file on every node in the cluster.

1. Open the **log_hosts.config** file located in the Websense Content Gateway **config** directory (default location is **/opt/WCG/config**).

2. Enter the host name of each origin server on a separate line in the file. For example:

   ```
   webserver1
   webserver2
   webserver3
   ```

3. Save and close the file.

4. From the Websense Content Gateway **bin** directory (default location is **/opt/WCG/bin**), run su `./content_line -x` to apply the changes.

Collating event log files

You can use the log file collation feature to keep all logged information in one place. This allows you to analyze Websense Content Gateway as a whole rather than as individual nodes and to use a large disk that might only be located on one of the nodes in a cluster.

Websense Content Gateway collates log files by using one or more nodes as log collation servers and all remaining nodes as log collation clients. When a node generates a buffer of event log entries, it determines whether it is the collation server or a collation client. The collation server node simply writes all log buffers to its local disk, just as it would if log collation were not enabled.

The collation client nodes prepare their log buffers for transfer across the network and send the buffers to the log collation server. When the log collation server receives a log buffer from a client, it writes it to its own log file as if it were generated locally. If log clients cannot contact their log collation server, they write their log buffers to their local disks, into *orphan* log files. Orphan log files require manual collation. Log collation servers can be stand-alone or they can be part of a node running Websense Content Gateway.

✔ **Note**
Log collation can have an impact on network performance. Because all nodes are forwarding their log data buffers to the single collation server, a bottleneck might occur in the network, where the amount of data being sent to a single node in the network exceeds the node's ability to process it quickly.

✔ **Note**
Collated log files contain time-stamp information for each entry, but entries do not appear in the files in strict chronological order. You can sort collated log files before doing analysis.

Configuring Websense Content Gateway to be a collation server

1. Navigate to **Configure > Subsystems > Logging > Collation**.
2. In the **Collation Mode** section, enable the **Be A Collation Server** option.
3. In the **Log Collation Port** field, enter the port number used for communication with collation clients. The default port number is 8085.
4. In the **Log Collation Secret** field, enter the password used to validate logging data and prevent the exchange of arbitrary information.

✔ **Note**
All collation clients must use this same secret.

5. Click **Apply**.

! **Important**
If you modify the collation port or secret after connections between the collation server and collation clients have been established, you must restart Websense Content Gateway.

Configuring Websense Content Gateway to be a collation client

1. Navigate to **Configure** > **Subsystems** > **Logging > Collation**.

2. In the **Collation Mode** section, enable the **Be a Collation Client** option to set the Websense Content Gateway node as a collation client and send the active standard formatted log entries (such as Squid and Netscape) to the log collation server.

> ✔ **Note**
>
> To send custom XML-based formatted log entries to the collation server, you must add a log object specification to the **logs_xml.config** file. See *Custom format*, page 153.

3. In the **To Collation Server** field, enter the host name of the collation server. This could be the Websense Content Gateway collation server or a stand-alone collation server.

4. In the **Log Collation Port** field, enter the port number used for communication with the collation server. The default port number is 8085.

5. In the **Log Collation Secret** field, enter the password used to validate logging data and prevent the exchange of arbitrary information. This must be the same secret you set on the collation server.

6. Enable the **Log Collation Host Tagged** option if you want to preserve the origin of log entries in the collated log files.

7. In the **Log Collation Orphan Space** field, enter the maximum amount of space (in megabytes) you want to allocate to the logging directory on the collation client for storing orphan log files. (Orphan log files are created when the log collation server cannot be contacted). The default value is 25 MB.

8. Click **Apply**.

> ❗ **Important**
>
> If you modify the collation port or secret after connections between the collation clients and collation server have been established, you must restart Websense Content Gateway.

Using a stand-alone collator

If you do not want the log collation server to be a Websense Content Gateway node, you can install and configure a stand-alone collator (SAC) which can dedicate more of its power to collecting, processing, and writing log files.

> ✔ **Note**
>
> The stand-alone collator is currently available for the UNIX platform only.

1. Configure your Websense Content Gateway nodes as log collation clients. See *Configuring Websense Content Gateway to be a collation client*, page 166.

2. Copy the **sac** binary from the Websense Content Gateway **bin** directory (default location is **/opt/WCG/bin**) to the machine serving as the stand-alone collator.

3. Create a directory called **config** in the directory that contains the **sac** binary.

4. Create a directory called **internal** in the **config** directory you created in Step 3. This directory will be used internally by the stand-alone collator to store lock files.

5. Copy the **records.config** file (default location is in **/opt/WCG/config**) from a Websense Content Gateway node configured to be a log collation client to the **config** directory you created in Step 3 on the stand-alone collator.

   The **records.config** file contains the log collation secret and port you specified when configuring nodes to be collation clients. The collation port and secret must be the same for all collation clients and servers.

6. Open the **records.config** file on the stand-alone collator and edit the following variable:

Variable	Description
`proxy.config.log2.logfile_dir`	Specify the directory where you want to store the log files. You can specify an absolute path to the directory or a path relative to the directory from which the `sac` binary is executed.
	Note: The directory must already exist on the machine serving as the stand-alone collator.

7. Save and close the file.

8. Enter the following command:

   ```
   sac -c config
   ```

Viewing logging statistics

Websense Content Gateway generates statistics about the logging system that help you see the following information:

◆ How many log files (formats) are currently being written.

◆ The current amount of space being used by the logging directory, which contains all of the event and error logs.

◆ The number of access events that have been written to log files since Websense Content Gateway installation. This counter represents one entry in one file. If multiple formats are being written, a single event will create multiple event log entries.

◆ The number of access events skipped (because they were filtered out) since Websense Content Gateway installation.

◆ The number of access events that have been written to the event error log since Websense Content Gateway installation.

You can view the statistics from the Monitor tab in Websense Content Manager or retrieve them through the command-line interface. See *Monitoring Traffic*, page 79.

Viewing log files

You can view the system, event, and error log files that Websense Content Gateway creates from Websense Content Manager. You can view an entire log file, a specified last number of lines in the log file, or all lines that contain a specified string.

You can also delete a log file or copy it to your local system.

> ✓ **Note**
> You must have the correct user permissions to copy and delete log files.

> ✓ **Note**
> Websense Content Gateway displays only 1 MB of information in the log file. If the log file you select to view is bigger than 1 MB, Websense Content Gateway truncates the file and displays a warning message indicating that the file is too big.

You can now access log files through Websense Content Manager.

1. Navigate to **Configure** > **My Proxy > Logs > System**.

2. To view, copy, or delete a system log file, go to Step 3.

   To view, copy, or delete an event or error log file, select the **Access** tab.

3. In the **Log File** drop-down list, select the log file you want to view, copy, or delete.

   Websense Content Gateway lists the system log files logged with the system-wide logging facility **syslog** under the daemon facility.

   Websense Content Gateway lists the event log files located in the directory specified in the **Logging Directory** field in the **Configure > Subsystems > Logging > General** tab or by the configuration variable **proxy.config.log2.logfile_dir** in the **records.config** file. The default directory is **logs** in the Websense Content Gateway installation directory.

4. In the **Action** area, select one of the following options:

   ▪ **Display the selected log file** to view the entire log file.

- **Display last lines of the selected file** to view the last lines of the log file. Enter the number of lines you want to view in the field provided.

- **Display lines that match in the selected log file** to view all the lines in the log file that match a particular string. Enter the string in the field provided.

- **Remove the selected log file** to delete the selected log file from the Websense Content Gateway system.

- **Save the selected log file in local filesystem** to save a copy of the selected log file on your local system.

5. Click **Apply**.

   If you selected to view the log file, Websense Content Gateway displays the file at the end of the page.

   If you selected to delete the log file, Websense Content Gateway deletes the file. You are not prompted to confirm the deletion.

   If you selected to save the log file, you are prompted for the location where you want to save the file on your local system.

Related topics:

Squid format, page 170
Netscape examples, page 171

Example event log file entries

This section shows examples of a log file entry in each of the standard log formats supported by Websense Content Gateway:

- *Squid format*, page 170
- *Netscape examples*, page 171
- *Netscape Extended format*, page 171
- *Netscape Extended-2 format*, page 171

Squid format

The following figure shows a sample log entry in a **squid.log** file. The table below describes each field.

Field	Description
1	The client request time stamp in Squid format; the time of the client request in seconds since January 1, 1970 UTC (with millisecond resolution).
2	The time the proxy spent processing the client request; the number of milliseconds between the time that the client established the connection with the proxy and the time that the proxy sent the last byte of the response back to the client.
3	The IP address of the client's host machine.
4	The cache result code; how the cache responded to the request: HIT, MISS, and so on. Cache result codes are described in *In Squid- and Netscape-format log files, what do the cache result codes mean?*, page 356. The proxy response status code (the HTTP response status code from Websense Content Gateway to client).
5	The length of the Websense Content Gateway response to the client in bytes, including headers and content.
6	The client request method: GET, POST, and so on.
7	The client request canonical URL; blanks and other characters that might not be parsed by log analysis tools are replaced by escape sequences. The escape sequence is a percentage sign followed by the ASCII code number of the replaced character in hex.
8	The authenticated client's user name. A hyphen (-) means that no authentication was required.
9	The proxy hierarchy route; the route Websense Content Gateway used to retrieve the object. The proxy request server name; the name of the server that fulfilled the request. If the request was a cache hit, this field contains a hyphen (-).
10	The proxy response content type; the object content type taken from the Websense Content Gateway response header.

Netscape examples

Netscape Common format

The following figure shows a sample log entry in a **common.log** file. The table below describes each field.

Netscape Extended format

The following figure shows a sample log entry in an **extended.log** file. The table below describes each field.

Netscape Extended-2 format

The following figure shows a sample log entry in an **extended2.log** file. The table below describes each field.

Field	Description
	Netscape Common
1	The IP address of the client's host machine.
2	This hyphen (-) is always present in Netscape log entries.
3	The authenticated client user name. A hyphen (-) means no authentication was required.
4	The date and time of the client's request, enclosed in brackets.
5	The request line, enclosed in quotes.

Field	Description
6	The proxy response status code (HTTP reply code).
7	The length of the Websense Content Gateway response to the client in bytes.
	Netscape Extended
8	The origin server's response status code.
9	The server response transfer length; the body length in the origin server's response to the proxy, in bytes.
10	The client request transfer length; the body length in the client's request to the proxy, in bytes.
11	The proxy request transfer length; the body length in the proxy request to the origin server.
12	The client request header length; the header length in the client's request to the proxy.
13	The proxy response header length; the header length in the proxy response to the client.
14	The proxy request header length; the header length in the proxy request to the origin server.
15	The server response header length; the header length in the origin server's response to the proxy.
16	The time Websense Content Gateway spent processing the client request; the number of seconds between the time that the client established the connection with the proxy and the time that the proxy sent the last byte of the response back to the client.
	Netscape Extended-2
17	The proxy hierarchy route; the route Websense Content Gateway used to retrieve the object.
18	The client finish status code: FIN if the client request completed successfully or INTR if the client request was interrupted.
19	The proxy finish status code: FIN if the Websense Content Gateway request to the origin server completed successfully or INTR if the request was interrupted.
20	The cache result code; how the Websense Content Gateway cache responded to the request: HIT, MISS, and so on. Cache result codes are described in *In Squid- and Netscape-format log files, what do the cache result codes mean?*, page 356.

A | Statistics

This appendix describes the following statistics on the Websense Content Manager Monitor tab:

My Proxy

My Proxy statistics are divided into the following categories:

Summary

Statistic/Field	Description
	Subscription Details
Feature	Lists features purchased, such as SSL Manager, and real-time options. See *Working With Encrypted Data*, page 121, for information on SSL Manager, and also *Analyze Content with the Real-Tine Options* in the Websense Web Security online help.
Purchased Status	Indicates if a feature has been purchased or not.

Statistic/Field	Description
Expiration Date	If a feature has been purchased, displays the expiration date of the subscription key.
More Details	
Subscription key	Display the subscription key entered on the Real-Time Scanning tab. See *Entering your subscription key*, page 11.
User count	Indicates how many users are included in the subscription. If you site includes more users than this number, contact Websense Technical Support. Go to www.websense.com/SupportPortal/default.aspx for Technical Support contact information
Last successful download time	Indicates the time of the most recent successful download of the database used by the real-time options. See *Analyze Content with the Real-Tine Options* in the Websense Web Security online help for additional information.
Node Details	
Node	Name of the Websense Content Gateway node or cluster.
On/Off	Indicates if the proxy is running (the proxy and manager services are running).
Objects Served	Total number of objects served by the Websense Content Gateway node.
Ops/Sec	Number of operations per second processed by the Websense Content Gateway node.
Hit Rate	Percentage of HTTP requests served from the cache, averaged over the past 10 seconds.
Throughput (Mbit/sec)	Number of megabits per second passing through the Websense Content Gateway node (and cluster).
HTTP Hit (ms)	Amount of time it takes for an HTTP object that is fresh in the cache to be served to the client.
HTTP Miss (ms)	Amount of time it takes for an HTTP object that is not in the cache or is stale to be served to the client.
More Detail	
cache hit rate	Percentage of HTTP requests served from the cache, averaged over the past 10 seconds. This value is refreshed every 10 seconds.
errors	Percentage of requests that end in early hangups.
aborts	Percentage of aborted requests.
active clients	Current number of open client connections.
active servers	Current number of open origin server connections.
node IP address	IP address assigned to the node. If virtual IP addressing is enabled, several virtual IP addresses could be assigned to this node.

Statistic/Field	Description
cache free space	Amount of free space in the cache.
HostDB hit rate	Ratio of host database hits to total host database lookups, averaged over a 10-second period.

Node

Statistic	Description
	Node Summary
Status	Indicates if Websense Content Gateway is running on this node (active or inactive).
Up Since	Date and time Websense Content Gateway was started.
Clustering	Indicates if clustering is on or off on this node.
	Cache
Document Hit Rate	Ratio of cache hits to total cache requests, averaged over 10 seconds. This value is refreshed every 10 seconds.
Bandwidth Savings	Ratio of bytes served from the cache to total requested bytes, averaged over 10 seconds. This value is refreshed every 10 seconds.
Cache Percent Free	Ratio of cache free space to total cache space.
	In Progress
Open Server Connections	Number of currently open origin server connections.
Open Client Connections	Number of currently open client connections.
Cache Transfers in Progress	Number of cache transfers (cache reads and writes) in progress.
	Network
Client Throughput (Mbit/Sec)	Number of megabits per second passing through the node (and cluster).
Transactions per Second	Number of HTTP transactions per second.
	Name Resolution
Host Database Hit Rate	Ratio of host database hits to total host database lookups, averaged over 10 seconds. This value is refreshed every 10 seconds.
DNS Lookups per Second	Number of DNS lookups per second.

Graphs

The Graphs page displays the same statistics listed on the *Node* page (cache performance, current connections and transfers, network, and name resolution) but in graphical format. You can choose the statistics you want to present in a graph. See *Viewing statistics*, page 79.

Alarms

Websense Content Gateway signals an alarm when it detects a problem (for example, if the space allocated to event logs is full or if Websense Content Gateway cannot write to a configuration file) and displays a description of the alarm in the alarm message window. In addition, the Alarm pending! bar at the top of the Websense Content Manager display indicates when alarms are detected and how many alarms exist.

After you have read an alarm message, click **Clear** in the alarm message window to dismiss the alarm. Clicking **Clear** only dismisses alarm messages; it does not actually resolve the cause of the alarms.

For information about working with alarms, see *Working with alarms*, page 84.

Protocols

Protocol statistics are divided into the following categories:

- *HTTP*, page 176
- *FTP*, page 178

HTTP

Statistic	Description
	General
Client	
Total Document Bytes	Total amount of HTTP data served to clients since installation.
Total Header Bytes	Total amount of HTTP header data served to clients since installation.
Total Connections	Total number of HTTP client connections since installation.
Current Connections	Current number of HTTP client connections
Transactions in Progress	Total number of HTTP client transactions in progress.

Statistic	Description
Server	
Total Document Bytes	Total amount of HTTP data received from origin servers since installation.
Total Header Bytes	Total amount of HTTP header data received from origin servers since installation.
Total Connections	Total number of HTTP server connections since installation.
Current Connections	Current number of HTTP server connections
Transactions in Progress	Total number of HTTP server connections currently in progress.
	Transaction
Hits	
Fresh	Percentage of hits that are fresh and their average transaction times.
Stale Revalidated	Percentage of hits that are stale and revalidated and turn out to be still fresh and served, and their average transaction times.
Misses	
Now Cached	Percentage of requests for documents that were not in the cache (but are now) and their average transaction times.
Server No Cache	Percentage of requests for HTTP objects that were not in the cache, but have server no-cache headers (cannot be cached); and their average transaction times.
Stale Reloaded	Percentage of misses that are revalidated and turn out to be changed, reloaded, and served, and their average transaction times.
Client No Cache	Percentage of misses with client no-cache headers and their average transaction times.
Errors	
Connection Failures	Percentage of connect errors and their average transaction times.
Other Errors	Percentage of other errors and their average transaction times.
Aborted Transactions	
Client Aborts	Percentage of client-aborted transactions and their average transaction times.
Questionable Client Aborts	Percentage of transactions that could possibly be client aborted and their average transaction times.
Partial Request Hangups	Percentage of early hangups (after partial requests) and their average transaction times.

Statistic	Description
Pre-Request Hangups	Percentage of pre-request hangups and their average transaction times.
Pre-Connect Hangups	Percentage of pre-connect hangups and their average transaction times.
Other Transactions	
Unclassified	Percentage of unclassified transactions and their average transaction times.
	FTP over HTTP
Connections	
Open Server Connections	Number of open connections to the FTP server.
Successful PASV Connections	Number of successful PASV connections since installation.
Failed PASV Connections	Number of failed PASV connections since installation.
Successful PORT Connections	Number of successful PORT connections since installation.
Failed PORT Connections	Number of failed PORT connections since installation.
Cache Statistics	
Hits	Number of HTTP requests for FTP objects served from the cache.
Misses	Number of HTTP requests for FTP objects forwarded directly to the origin server because the object is not in the cache or is stale.
Lookups	Number of times Websense Content Gateway looked up an HTTP request for an FTP object in the cache.

FTP

Statistic	Description
	Client
Open Connections	Number of client connections currently open.
Bytes Read	Number of client request bytes read since installation.
Bytes Written	Number of client request bytes written since installation.
	Server
Open Connections	Number of FTP server connections currently open.
Bytes Read	The number of bytes read from FTP servers since installation.
Bytes Written	Number of bytes written to the cache since installation.

Statistic	Description
	Operations
File Hit	Number of FTP files served from the cache since installation.
File Misses	Number of FTP file misses since installation.
Change Directory Hits	Number of change directory hits since installation.
Change Directory Misses	Number of change directory misses since installation.
List Directory Hits	The number of list directory hits since installation.
List Directory Misses	The number of list directory misses since installation.

Content Routing (ICP)

Statistic	Description
	Queries Originating from this Node
Query Requests	Number of HTTP requests that generate ICP query messages.
Query Messages	Total number of ICP query messages sent to ICP peers. This number is larger than the number of Query Requests if there are multiple ICP peers.
Peer Hit Messages Received	Number of ICP peer hit messages received in response to ICP queries from this node.
Peer Miss Messages Received	Number of ICP peer miss messages received in response to ICP queries from this node.
Total Responses Received	Number of response messages received from ICP peers (siblings and parents).
Average ICP Message Response Time (ms)	Average time for an ICP peer to respond to an ICP query message from this node. This is a cumulative average value.
Average ICP Request Time	Average time for an HTTP request (that is sent to ICP) to receive an ICP response. This is a cumulative average value.
	Queries Originating from ICP Peers
Query Messages Received	Number of ICP query messages received from remote ICP peers (siblings and parents).
Remote Query Hits	Number of successful cache lookups in response to queries from ICP peers.
Remote Query Misses	Number of unsuccessful cache lookups in response to queries from ICP peers.
Successful Response Messages Sent to Peers	Number of successful ICP messages written in response to ICP queries from remote ICP peers.

Security

Security statistics are divided into the following categories:

- *ARM security*, page 180
- *LDAP* , page 180
- *NTLM* , page 180
- *SOCKS*, page 181

ARM security

Statistic	Description
	Security Statistics
TCP Dropped	Number of dropped TCP connections.
UDP Dropped	Number of dropped UDP connections.

LDAP

Statistic	Description
	Cache
Hits	Number of hits in the LDAP cache.
Misses	The number of misses in the LDAP cache.
	Errors
Server	Number of LDAP server errors.
	Unsuccessful Authentications
Authorization Denied	Number of times the LDAP Server denied authorization.
Authorization Timeouts	Number of times authorization timed out.
Authentication Cancelled	Number of times authentication was cancelled.

NTLM

Statistic	Description
	Cache
Hits	Number of hits in the NTLM cache.

Statistic	Description
Misses	Number of misses in the NTLM cache.
	Errors
Server	Number of NTLM server errors.
	Unsuccessful Authentications
Authorization Denied	Number of times the LDAP server denied authorization.
Authorization Cancelled	Number of times authentication was cancelled.

SOCKS

Statistic	Description
Unsuccessful Connections	Number of unsuccessful connections to the SOCKS server since Websense Content Gateway was started.
Successful Connections	Number of successful connections to the SOCKS server since Websense Content Gateway was started.
Connections in Progress	Number of connections to the SOCKS server currently in progress.

Subsystems

Subsystems statistics are divided into the following categories:

- *Cache* , page 181
- *Clustering* , page 182
- *Logging*, page 183

Cache

Statistic	Description
	General
Bytes Used	Number of bytes currently used by the cache.
Cache Size	Number of bytes allocated to the cache.
	Ram Cache
Bytes	Total size of the RAM cache, in bytes.
Hits	Number of document hits from the RAM cache.

Statistic	Description
Misses	Number of document misses from the RAM cache. The documents may be hits from the cache disk.
Reads	
In Progress	Number of cache reads in progress (HTTP and FTP).
Hits	Number of cache reads completed since Websense Content Gateway was started (HTTP and FTP).
Misses	Number of cache read misses since Websense Content Gateway was started (HTTP and FTP).
Writes	
In Progress	Number of cache writes in progress (HTTP and FTP).
Successes	Number of successful cache writes since Websense Content Gateway was started (HTTP and FTP).
Failures	Number of failed cache writes since Websense Content Gateway was started (HTTP and FTP).
Updates	
In Progress	Number of HTTP document updates in progress. An update occurs when the Websense Content Gateway revalidates an object, finds it to be fresh, and updates the object header.
Successes	Number of successful cache HTTP updates completed since Websense Content Gateway was started.
Failures	Number of cache HTTP update failures since Websense Content Gateway was started.
Removes	
In Progress	Number of document removes in progress. A remove occurs when the Websense Content Gateway revalidates a document, finds it to be deleted on the origin server, and deletes it from the cache (includes HTTP and FTP removes).
Successes	Number of successful cache removes completed since Websense Content Gateway was started (includes HTTP and FTP removes).
Failures	Number of cache remove failures since Websense Content Gateway was started (includes HTTP and FTP removes).

Clustering

Statistic	Description
Bytes Read	Number of bytes read by this node from other nodes in the cluster since Websense Content Gateway was started.
Bytes Written	Number of bytes this node has written to other cluster nodes since Websense Content Gateway was started.
Connections Open	Total number of intra-cluster connections opened since Websense Content Gateway was started.

Statistic	Description
Total Operations	Total number of cluster transactions since Websense Content Gateway was started.
Network Backups	Number of times this node encountered intra-cluster network congestion and reverted to proxy-only mode since Websense Content Gateway was started.
Clustering Nodes	Number of clustering nodes.

Logging

Statistic	Description
Currently Open Log Files	Number of event log files (formats) that are currently being written.
Space Used for Log Files	Current amount of space being used by the logging directory, which contains all of the event and error logs.
Number of Access Events Logged	Number of access events that have been written to log files since Websense Content Gateway installation. This counter represents one entry in one file. If multiple formats are being written, a single access creates multiple event log entries.
Number of Access Events Skipped	Number of access events skipped (because they were filtered out) since Websense Content Gateway installation.
Number of Error Events Logged	Number of access events that have been written to the event error log since Websense Content Gateway installation.

Networking

Networking statistics are divided into the following categories:

- *System*, page 184
- *ARM*, page 184
- *WCCP*, page 185
- *DNS Resolver*, page 186
- *Virtual IP*, page 186

System

Statistic/Field	Description
	General
Hostname	The *hostname* assigned to this Websense Content Gateway machine.
Default Gateway	IP address of the default gateway used to forward packets from this Websense Content Gateway machine to other networks or subnets.
Search Domain	Search domain that this Websense Content Gateway machine uses.
Primary DNS	IP address of the primary DNS server that this Websense Content Gateway machine uses to resolve host names.
Secondary DNS	Secondary DNS server that this Websense Content Gateway machine uses to resolve host names.
Tertiary DNS	Third DNS server that this Websense Content Gateway machine uses to resolve host names.

ARM

Statistic	Description
	Network Address Translation (NAT) Statistics
Client Connections Natted	Number of client connections redirected transparently by the ARM.
Client Connections in Progress	Number of client connections currently in progress with the ARM.
Total Packets Natted	Number of packets translated by the ARM.
DNS Packets Natted	Number of DNS packets translated by the ARM.
	Bypass Statistics
Total Connections Bypassed	Total number of connections bypassed by the ARM.
DNS Packets Bypassed	Number of DNS packets bypassed by the ARM.
	HTTP Bypass Statistics
Bypass on Bad Client Request	Number of requests forwarded directly to the origin server because Websense Content Gateway encountered non-HTTP traffic on port 80.
Bypass on 400	Number of requests forwarded directly to the origin server because an origin server returned a 400 error.

Statistic	Description
Bypass on 401	Number of requests forwarded directly to the origin server because an origin server returned a 401 error.
Bypass on 403	Number of requests forwarded directly to the origin server because an origin server returned a 403 error.
Bypass on 405	Number of requests forwarded directly to the origin server because an origin server returned a 405 error.
Bypass on 406	Number of requests forwarded directly to the origin server because an origin server returned a 406 error.
Bypass on 408	Number of requests forwarded directly to the origin server because an origin server returned a 408 error.
Bypass on 500	Number of requests forwarded directly to the origin server because an origin server returned a 500 error.

WCCP

WCCP 1.0 statistics are displayed only if WCCP version 1.0 is enabled. WCCP 2.0 statistics are displayed only if WCCP version 2.0 is enabled.

Statistic/Field	Description
	WCCP v1.0 Statistics
Router Information	
Router IP address	IP address of the router sending traffic to Websense Content Gateway.
Router Status	Status of the router: *up* (if the proxy is able to communicate with the router) or *down* (if the proxy is unable to communicate with the router).
Node Information	
My IP address	IP address of this Websense Content Gateway node.
Percentage of Traffic Directed to This Node	Percentage of traffic directed to this Websense Content Gateway node.
Number of Heartbeats Received	Number of heartbeats received by this Websense Content Gateway node.
Protocol Information	
Leader's IP Address	IP address of the leader in the WCCP cache farm.
Number of Active Nodes	Number of active Websense Content Gateway nodes in the WCCP cache farm.
	WCCP v2.0 Statistics
WCCP v2.0 Fragmentation	
Total Fragments	Total number of WCCP fragments.

Statistic/Field	Description
Fragmentation Table Entries	Number of entries in the fragmentation table.
Out of Order Fragments	Number of fragments out of order.
Matches	Number of fragments that match a fragment in the fragmentation table.
HTTP/FTP/SOCKS/DNS Service	
Service ID	Service ID for the protocol being serviced.
My IP address	IP address to which the router is sending traffic.
Leader's IP address	IP address of the leader in the WCCP cache farm.
Number of buckets assigned to me	Number of buckets assigned to this Websense Content Gateway node.
Number of caches	The number of caches in the WCCP cache farm.
Number of routers	The number of routers sending traffic to this Websense Content Gateway node.

DNS Resolver

Statistic	Description
	DNS Resolver
Total Lookups	Total number of DNS lookups (queries to name servers) since installation.
Successes	Total number of successful DNS lookups since installation.
Average Lookup Time (ms)	Average DNS lookup time.
	Host Database
Total Lookups	Total number of lookups in the Websense Content Gateway host database since installation.
Total Hits	Total number of host database lookup hits since installation.
Average TTL (min)	Average time to live in minutes.

Virtual IP

The Virtual IP table displays the virtual IP addresses that are managed by the proxies in the cluster.

MRTG

MRTG (Multi Router Traffic Grapher) enables you to monitor Websense Content Gateway performance and analyze network traffic. MRTG provides graphs that show information about virtual memory usage, client connections, document hit rates, hit and miss rates, and so on. MRTG uses 5-minute intervals to formulate the statistics and provides historical information. MRTG provides the following information.

Statistic	Description
Overview	Displays a subset of the graphs available in MRTG.
Daily	Displays graphs that provide historical information for the current day.
Weekly	Displays graphs that provide historical information for the current week.
Monthly	Displays graphs that provide historical information for the current month.
Yearly	Displays graphs that provide historical information for the current year.

> **Important**
>
> To run MRTG in UNIX, you must have Perl version 5.005 or later installed on your Websense Content Gateway system.

SSL

These tabs are visible only if your subscription includes SSL Manager.

SSL Key Data

These fields provide information about the status of the SSL connection and activity between the client and SSL Manager and SSL Manager and the destination server.

Statistic/Field	Description
	SSL Inbound Key Data
Is Alive	Online indicates that SSL Manager is enabled
Current SSL Connections	Number of active inbound (browser to SSL Manager) SSL requests

Statistic/Field	Description
Total SSL Server Connects	Number of browser requests
Total SSL Server Connects that finished	Number of browser requests where data went to SSL Manager for decryption
Total SSL Server Renegotiation Requests	Number of browser requests renegotiated due to handshake failures or invalid certificates between the browser and SSL Manager
Total SSL Session Cache Hits	Number of times that a request was validated by a key in the session cache
Total SSL Session Cache Misses	Number of times that a request could not be validated by a key in the session cache
Total SSL Session Cache Timeouts	Number of time keys were removed from the session cache because the timeout period expired
	SSL Outbound Key Data
Is Alive	Online indicates that SSL Manager is enabled
Current SSL Connections	Number of active outbound (SSL Manager to designating server) SSL requests
Total SSL Client Connects	Number of browser requests
Total SSL Client Connects that finished	Number of requests where data went from SSL Manager to the destination server
Total SSL Client Renegotiation Requests	Number of requests were renegotiated due to handshake failures or invalid certificates between SSL Manager and the destination server
Total SSL Session Cache Hits	Number of times that a request was validated by a key in the session cache
Total SSL Session Cache Misses	Number of times that a request could not be validated by a key in the session cache
Total SSL Session Cache Timeouts	Number of time keys were removed from the session cache because the timeout period expired

CRL Statistics

These fields provide information about certificate status.

Statistic/Field	Description
	CRL Statistics
CRL List Count	The number of certificates on the Certificate Revocation List. This list is downloaded every night. See *Keeping revocation information up to date*, page 136.
	OCSP Statistics

Statistic/Field	Description
OCSP Good Count	The number of responses that certificates are valid.
OCSP Unknown Count	The number of OCSP responses where the certificate cannot be verified

Reports

See *Creating reports with SSL Manager*, page 87, for information on creating reports on certificate authorities or incidents.

B | Commands and Variables

Websense Content Gateway commands

Use the command line to execute individual commands and script multiple commands in a shell.

Execute Websense Content Gateway commands from the Websense Content Gateway **bin** directory.

> ✔ **Note**
>
> If the Websense Content Gateway **bin** directory is not in your path, prepend the command with `./` (for example, `./content_line -p`).

Command	Description
`WCGAdmin start`	Starts the Websense Content Gateway service
`WCGAdmin stop`	Stops the Websense Content Gateway service
`WCGAdmin restart`	Stops the Websense Content Gateway service and then starts it again
`WCGAdmin status`	Displays the status (running or not running) of the Websense Content Gateway services: Websense Content Gateway, Websense Content Manager, and `content_cop`.
`WCGAdmin help`	Displays a list of the WCGAdmin commands
`content_line -p socket_path`	Specifies the location (directory and path) of the file used for Websense Content Gateway command line and Websense Content Manager communication. The default path is **install_dir/config/cli**
`content_line -r variable`	Displays specific performance statistics or a current configuration setting. For a list of the variables you can specify, see *Websense Content Gateway variables*, page 192.

Command	Description
`content_line -s variable -v value`	Sets configuration variables. `variable` is the configuration variable you want to change and `value` is the value you want to set. See *records.config*, *page 282*, for a list of the configuration variables you can specify.
`content_line -h`	Displays the list of Websense Content Gateway commands.
`content_line -x`	Initiates a Websense Content Gateway configuration file reread. Executing this command is similar to clicking **Apply** in Websense Content Manager.
`content_line -M`	Restarts the `content_manager` process and the `content_gateway` process on all the nodes in a cluster.
`content_line -L`	Restarts the `content_manager` process and the `content_gateway` process on the local node.
`content_line -S`	Shuts down Websense Content Gateway on the local node.
`content_line -U`	Starts Websense Content Gateway on the local node.
`content_line -B`	Bounces Websense Content Gateway wide. Bouncing Websense Content Gateway shuts down and immediately restarts the proxy cache node by node.
`content_line -b`	Bounces Websense Content Gateway on the local node. Bouncing Websense Content Gateway shuts down and immediately restarts the proxy cache on the node.

Websense Content Gateway variables

You can view statistics and change configuration options via the command line by using specific variables, which are described in *records.config*, *page 282*. See *Viewing statistics from the command line*, *page 83*, and *Command-line interface*, *page 94*.

Statistics

The following table lists the variables you can specify on the command line to view individual statistics. See *Statistics*, *page 173* for additional information.

To view a statistic, enter `content_line -r variable` at the prompt.

Statistic	Variable
	Summary
Node name	`proxy.node.hostname`
Objects served	`proxy.node.user_agents_total_documents_served`

Statistic	Variable
Transactions per second	`proxy.node.user_agent_xacts_per_second`
	Node
Document hit rate	`proxy.node.cache_hit_ratio_avg_10s` `proxy.cluster.cache_hit_ratio_avg_10s`
Bandwidth savings	`proxy.node.bandwidth_hit_ratio_avg_10s` `proxy.cluster.bandwidth_hit_ratio_avg_10s`
Cache percent free	`proxy.node.cache.percent_free` `proxy.cluster.cache.percent_free`
Open origin server connections	`proxy.node.current_server_connections` `proxy.cluster.current_server_connections`
Open client connections	`proxy.node.current_client_connections` `proxy.cluster.current_client_connections`
Cache transfers in progress	`proxy.node.current_cache_connections` `proxy.cluster.current_cache_connections`
Client throughput (Mbits/sec)	`proxy.node.client_throughput_out` `proxy.cluster.client_throughput_out`
Transactions per second	`proxy.node.http.user_agent_xacts_per_second` `proxy.cluster.http.user_agent_xacts_per_second`
DNS lookups per second	`proxy.node.dns.lookups_per_second` `proxy.cluster.dns.lookups_per_second`
Host database hit rate	`proxy.node.hostdb.hit_ratio_avg_10s` `proxy.cluster.hostdb.hit_ratio_avg_10s`
	HTTP
Total document bytes from client	`proxy.process.http.user_agent_response_document_total_size`
Total header bytes from client	`proxy.process.http.user_agent_response_header_total_size`
Total connections to client	`proxy.process.http.current_client_connections`
Client transactions in progress	`proxy.process.http.current_client_transactions`
Total document bytes from origin server	`proxy.process.http.origin_server_response_document_total_size`
Total header bytes from origin server	`proxy.process.http.origin_server_response_header_total_size`
Total connections to origin server	`proxy.process.http.current_server_connections`

Statistic	Variable
Origin server transactions in progress	`proxy.process.http.current_server_transactions`
	FTP
Currently open FTP connections	`proxy.process.ftp.connections_currently_open`
Successful PASV connections	`proxy.process.ftp.connections_successful_pasv`
Unsuccessful PASV connections	`proxy.process.ftp.connections_failed_pasv`
Successful PORT connections	`proxy.process.ftp.connections_successful_port`
Unsuccessful PORT connections	`proxy.process.ftp.connections_failed_port`
	ICP
ICP query requests originating from this node	`proxy.process.icp.icp_query_requests`
ICP query messages sent from this node	`proxy.process.icp.total_udp_send_queries`
ICP peer hit messages received from this node	`proxy.process.icp.icp_query_hits`
ICP peer miss messages received from this node	`proxy.process.icp.icp_query_misses`
Total ICP responses received from this node	`proxy.process.icp.icp_remote_responses`
Average ICP message response time (ms) from this node	`proxy.process.icp.total_icp_response_time`
Average ICP request time (ms) from this node	`proxy.process.icp.total_icp_request_time`
Query messages received from ICP peers	`proxy.process.icp.icp_remote_query_requests`
Remote query hits from ICP peers	`proxy.process.icp.cache_lookup_success`
Remote query misses from ICP peers	`proxy.process.icp.cache_lookup_fail`

Statistic	Variable
Successful response messages sent to peers	`proxy.process.icp.query_response_write`
WCCP	
WCCP router's IP address	`proxy.node.wccp.router_ip`
WCCP router status	`proxy.node.wccp.router_status`
WCCP node IP address	`proxy.node.wccp.my_ip`
Percentage of WCCP traffic received	`proxy.node.wccp.my_share`
Number of WCCP heartbeats	`proxy.node.wccp.hbeats_received`
Enabled	`proxy.node.wccp.enabled`
WCCP leader's IP address	`proxy.node.wccp.leader_ip`
Number of active WCCP nodes	`proxy.node.wccp.number_of_caches_up`
Cache	
Bytes used	`proxy.process.cache.bytes_used`
Cache size	`proxy.process.cache.bytes_total`
Lookups in progress	`proxy.process.cache.lookup.active`
Lookups completed	`proxy.process.cache.lookup.success`
Lookup misses	`proxy.process.cache.lookup.failure`
Reads in progress	`proxy.process.cache.read.active`
Reads completed	`proxy.process.cache.read.success`
Read misses	`proxy.process.cache.read.failure`
Writes in progress	`proxy.process.cache.write.active`
Writes completed	`proxy.process.cache.write.success`
Write failures	`proxy.process.cache.write.failure`
Updates in progress	`proxy.process.cache.update.active`
Updates completed	`proxy.process.cache.update.success`
Update failures	`proxy.process.cache.update.failure`
Removes in progress	`proxy.process.cache.remove.active`
Remove successes	`proxy.process.cache.remove.success`
Remove failures	`proxy.process.cache.remove.failure`

Statistic	Variable
Host DB	
Total lookups	*proxy.process.hostdb.total_lookups*
Total hits	*proxy.process.hostdb.total_hits*
Time TTL (min)	*proxy.process.hostdb.ttl*
DNS	
DNS total lookups	*proxy.process.dns.total_dns_lookups*
Average lookup time (ms)	*proxy.process.dns.lookup_avg_time*
DNS successes	*proxy.process.dns.lookup_successes*
Cluster	
Bytes read	proxy.process.cluster.read_bytes
Bytes written	proxy.process.cluster.write_bytes
Connections open	proxy.process.cluster.connections_open
Total operations	proxy.process.cluster.connections_opened
Network backups	proxy.process.cluster.net_backup
Clustering nodes	proxy.process.cluster.nodes
SOCKS	
Unsuccessful connections	*proxy.process.socks.connections_unsuccessful*
Successful connections	*proxy.process.socks.connections_successful*
Connections in progress	*proxy.process.socks.connections_currently_open*
Logging	
Currently open log files	*proxy.process.log2.log_files_open*
Space used for log files	*proxy.process.log2.log_files_space_used*
Number of access events logged	*proxy.process.log2.event_log_access*
Number of access events skipped	*proxy.process.log2.event_log_access_skip*
Number of error events logged	*proxy.process.log2.event_log_error*

C | Configuration Options

The following configuration options are available on the Websense Content Manager Configure pane:

My Proxy

The My Proxy configuration options are divided into the following categories:

Basic

When you set these options, ensure that each option is running on a unique port.

Option	Description
	General
Restart	Restarts the proxy and manager services (the `content_gateway` and `content_manager` processes). You must restart the proxy and manager services after modifying certain configuration options. In a cluster configuration, the **Restart** button restarts the proxy and manager services on all the nodes in the cluster.

Option	Description
Proxy Name	Specifies the name of your Websense Content Gateway node (by default, this is the hostname of the machine running Websense Content Gateway). If this node is part of a cluster, this option specifies the name of the Websense Content Gateway cluster (in a Websense Content Gateway cluster, all nodes must share the same name).
Alarm E-Mail	Specifies the email address to which Websense Content Gateway sends alarm notifications.
Features	
General: SNMP	Enables or disables the Websense Content Gateway SNMP agent. The SNMP agent supports access to 2 management information bases (MIBs): MIB-2 (a standard MIB) and the Websense Content Gateway MIB. Descriptions of the Websense Content Gateway MIB variables are provided in the **websense-Websense Content Gateway-mib.my** file in the Websense Content Gateway **config/mibs directory**.The Websense Content Gateway MIB contains both node-specific and cluster-wide information Configure your system so that only certain hosts can access this MIB. You configure access control and SNMP trap destinations in the **snmpd.cnf** file in the Websense Content Gateway **config** directory. See *snmpd.cnf*, page 331.
Protocols: FTP	Enables or disables processing of FTP requests from FTP clients. When this option is enabled, Websense Content Gateway accepts FTP requests from FTP clients. When this option is disabled, Websense Content Gateway does not accept FTP requests from FTP clients. If you change this option, you must restart Websense Content Gateway.
Protocols: HTTPS	Activates SSL Manager to enable or disable processing of HTTPS requests (encrypted data). After selecting HTTPS **On**, you must provide additional information on the **Configure > Protocols > HTTPS** page and on the **Configure > SSL** pages. See *Working With Encrypted Data*, page 121.
Networking: ARM	Enables or disables the ARM. If you change this option, you must restart Websense Content Gateway.
Networking: WCCP	Enables or disables WCCP. Enable this option if you are using a WCCP-enabled router for transparent proxy caching. See *Using a WCCP-enabled router*, page 44. If you change this option, you must restart Websense Content Gateway.

Option	Description
Networking: Virtual IP	Enables or disables the virtual IP failover option. When this option is enabled, Websense Content Gateway maintains a pool of virtual IP addresses that it assigns to the nodes in a cluster as necessary. See *Virtual IP failover*, page 65.
Networking: ICAP	Enables or disables ICAP support for use with Data Security Suite. See *Working With Websense Data Security Suite*, page 99.
Security: SOCKS	Enables or disables the SOCKS option. When SOCKS is enabled, Websense Content Gateway can talk to your SOCKS servers. See *Configuring SOCKS firewall integration*, page 106. If you change this option, you must restart Websense Content Gateway.
Authentication: None	The proxy will not perform authentication. This is the default setting.
Authentication: LDAP	Enables or disables LDAP proxy authentication. When LDAP is enabled, you can ensure that users are authenticated by an LDAP server before accessing content from the cache. See *Using LDAP proxy authentication*, page 112. If you change this option, you must restart Websense Content Gateway.
Authentication: Radius	Enables or disables RADIUS proxy authentication. When RADIUS is enabled, you can ensure that users are authenticated by a RADIUS server before accessing content from the cache. See *Using RADIUS proxy authentication*, page 114. If you change this option, you must restart Websense Content Gateway.
Authentication: NTLM	Enables or disables NTLM proxy authentication. When NTLM is enabled, you can ensure that users in a Windows network are authenticated by a Domain Controller before accessing content from cache. See *Using NTLM proxy authentication*, page 117. If you change this option, you must restart Websense Content Gateway.

Option	Description
	Clustering
Cluster: Type	Specifies the clustering mode:
	Select **Single Node** to run this Websense Content Gateway (Websense Content Gateway) server as a single node. This node will not be part of a cluster.
	Select **Management Clustering** to run management-only clustering mode. The nodes in the cluster share configuration information and you can administer all the nodes at the same time.
	Select **Full Cache Clustering** to run full-clustering mode. In full-clustering mode, as well as sharing configuration information, a Websense Content Gateway cluster distributes its cache across its nodes into a single, virtual object store, rather than replicating the cache node by node.
	For more information about clustering mode, see *Clusters*, page 61.
	If you change this option, you must restart Websense Content Gateway.
Cluster: Interface	Specifies the interface on which Websense Content Gateway communicates with other nodes in the cluster: for example, eth0 on Linux.
	It is recommended that you use a dedicated secondary interface for cluster communication.
	If you change this option, you must restart Websense Content Gateway.
Cluster: Multicast Group Address	Specifies the multicast group address on which Websense Content Gateway communicates with its cluster peers.

Subscription

Option	Description
	Subscription Management
Subscription Key	Enter the subscription key you received from Websense. This key reflects the products you have purchased.
	Real-Time Scanning
Policy Server	
IP address	Specify the IP address of the Websense Web Filter (check new name on this) Policy Server
Port	Specify the port used by the Websense Web Filter (check new name on this) Policy Server
Filtering Service	

Option	Description
IP address	Specify the IP address for Filtering Service.
Port	Specify the port used by Filtering Service.
Action for Communication Errors	
Permit traffic	Permits all pages if communication with Policy Server or Filtering Service fails.
Block traffic	Bocks all pages if communication with Filtering Service fails.

UI Setup

Option	Description
	General
UI Port	Specifies the port on which browsers can connect to Websense Content Manager. The port must be on Websense Content Gateway and it must be dedicated to Websense Content Gateway use. The default port is 8081.
	If you change this setting, you must restart Websense Content Gateway.
SSL UI Port	Specifies for port for the SSL Manager user interface. Through this interface you can specify data decryption and certificate management. The default port is 8071. See *Working With Encrypted Data*, page 121.
	The Websense Content Manager interface and the SSL Manager interface must be on different ports.
	If you change this setting, you must restart Websense Content Gateway.
HTTPS: Enable/Disable	Enables/disables support for SSL connections to Websense Content Manager. SSL provides protection for remote administrative monitoring and configuration. To use SSL for Content Manager connections, you must install an SSL certificate on the Websense Content Gateway server machine. For more information, see *Using SSL for secure administration*, page 105.
HTTPS: Certificate File	Specifies the name of the SSL certificate file used to authenticate users who want to access Websense Content Manager.
Monitor Refresh Rate	Specifies how often Websense Content Manager refreshes the statistics on the **Monitor** pane. The default value is 30 seconds.
	Login

Option	Description
Basic Authentication	Enables or disables basic authentication. When this option is enabled, Websense Content Gateway checks the administrator login and password or the user name and password (if user accounts have been configured) whenever a user tries to access Websense Content Manager.
Administrator: Login	Specifies the administrator login. The administrator login is the master login that has access to both Configure and Monitor mode in Websense Content Manager.
	Note: Websense Content Gateway checks the administrator login only if the Basic Authentication option is enabled.
Administrator: Password	Lets you change the administrator password that controls access to Websense Content Manager.
	To change the password, enter the current password in the **Old Password** field, and then enter the new password in the **New Password** field. Retype the new password in the **New Password (Retype)** field, and then click **Apply**.
	Note: Websense Content Gateway checks the administrator login and password only if the Basic Authentication option is enabled.
	During installation, you select the administrator password. The installer automatically encrypts the password and stores the encryptions in the **records.config** file so that no one can read them. Each time you change the password in Websense Content Manager, Websense Content Gateway updates the **records.config** file. If you forget the administrator password and cannot access the Websense Content Manager, see *How do you access Websense Content Manager if you forget the master administrator password?*, page 354.

Option	Description
Additional Users	Lists the current user accounts and lets you add new user accounts. User accounts determine who has access Websense Content Manager and which activities they can perform. You can create a list of user accounts if a single administrator login and password is not sufficient security for your needs.
	To create a new account, enter the user login in the **New User** field, and then enter the user password in the **New Password** field. Retype the user password in the **New Password (Retype)** field, and then click **Apply**. Information for the new user is displayed in the table. From the **Access** drop-down list in the table, select the activities that the new user can perform (Monitor, Monitor and View Configuration, or Monitor and Modify Configuration). For more information about user accounts, see *Creating a list of user accounts*, page 104.
	Note: Websense Content Gateway checks the user login and password only if the Basic Authentication option is enabled.
	Access
Access Control	Displays a table listing the rules in the *mgmt_allow.config* file that specify the remote hosts allowed to access Websense Content Manager. The entries in this file ensure that only authenticated users can change configuration options and view performance and network traffic statistics.
	Note: By default, all remote hosts are allowed to access the Websense Content Manager.
Refresh	Updates the table to display the most up-to-date rules in the **mgmt_allow.config** file.
Edit File	Opens the configuration file editor so that you can edit and add rules to the **mgmt_allow.config** file.
	mgmt_allow.config Configuration File Editor
rule display box	Lists the **mgmt_allow.config** file rules. Select a rule to edit it. The buttons on the left of the box allow you to delete or move the selected rule up or down in the list. Websense Content Gateway applies the rules in the order listed, starting from the top.
Add	Adds a new rule to the rule display box at the top of the configuration file editor page.
Set	Updates the rule display box at the top of the configuration file editor page.

Option	Description
IP Action	Lists the type of rules you can add.
	An `ip_allow` rule allows the remote hosts specified in the **Source IP** field to access Websense Content Manager.
	An `ip_deny` rule denies the remote hosts specified in the **Source IP** field access to Websense Content Manager.
Source IP	Specifies the IP addresses that are allowed or denied access to Websense Content Manager. You can enter a single IP address (111.111.11.1) or a range of IP addresses (0.0.0.0-255.255.255.255).
Apply	Applies the configuration changes.
Close	Exits the configuration file editor.
	Click **Apply** before you click **Close**; otherwise, all configuration changes will be lost.

Snapshots

Option	Description
	File System
Change Snapshot Directory	Specifies the directory in which snapshots are stored on this Websense Content Gateway node.
Snapshots: Save Snapshot	Specifies the name of the configuration snapshot you want to take. Click **Apply** to save the configuration on the local node. Websense Content Gateway saves the configuration snapshot in the directory specified in the **Change Snapshot Directory** field.
	It is recommended that you take a snapshot before performing system maintenance or attempting to tune system performance. Taking a snapshot takes only a few seconds and can save you hours of correcting configuration mistakes.
Snapshots: Restore/Delete Snapshot	Lists the snapshots that are stored on this node. Select the snapshot that you want to restore or delete from the drop-down list.
Snapshots: Restore Snapshot from "*directory_name*" Directory	Restores the snapshot selected in the **Restore/ Delete Snapshot** drop-down box.
	In a cluster configuration, snapshots are restored on all nodes in the cluster.
Snapshots: Delete Snapshot from "*directory_name*" Directory	Deletes the snapshot selected in the **Restore/ Delete Snapshot** drop-down box.
	FTP Server

Option	Description
FTP Server	Specifies the name of the FTP server from which you want to restore a configuration snapshot or to which you want to save a configuration snapshot.
Login	Specifies the login needed to access the FTP server.
Password	Specifies the password needed to access the FTP server.
Remote Directory	Specifies the directory on the FTP server from which you want restore, or in which you want to save a configuration snapshot.
Restore Snapshot	Lists the configuration snapshots on the FTP server that you can restore. This field appears after you have logged on to the FTP server successfully.
Save Snapshot to FTP Server	Specifies the name of the configuration snapshot you want to take and save on the FTP server. This field appears after you have logged on to the FTP server successfully.

Logs

Option	Description
	System
Log File	Lists the system log files you can view, delete or copy to your local system. Websense Content Gateway lists the system log files logged with the system-wide logging facility **syslog** under the daemon facility.
Action: Display the selected log file	When this option is enabled, Websense Content Gateway displays the entire system log file selected in the **Log File** drop-down list.
Action: Display last lines of the selected file	When this option is enabled, Websense Content Gateway displays the last specified number of lines in the selected system log file.
Action: Display lines that match in the selected log file	When this option is enabled, Websense Content Gateway displays all the lines in the selected system log file that match the specified string.
Action: Remove the selected log file	When this option is enabled, Websense Content Gateway deletes the selected log file.
Action: Save the selected log file in local filesystem	When this option is enabled, Websense Content Gateway saves the selected log file on the local system in a location you specify.
	Access

Option	Description
Log File	Lists the event or error log files you can view, delete or copy to your local system. Websense Content Gateway lists the event log files located in the directory specified in the **Logging Directory** field under **Subsystems/Logging** and by the configuration variable **proxy.config.log2.logfile_dir** in the **records.config** file. The default directory is `logs` in the Websense Content Gateway installation directory.
Action: Display the selected log file	When this option is enabled, Websense Content Gateway displays the entire event or error log file selected in the **Log File** drop-down list.
Action: Display last lines of the selected file	When this option is enabled, Websense Content Gateway displays the last specified number of lines in the event or error log file selected from the **Log File** drop-down list.
Action: Display lines that match in the selected log file	When this option is enabled, Websense Content Gateway displays all the lines in the selected event or error log file that match the specified string.
Remove the selected log file	When this option is enabled, Websense Content Gateway deletes the selected log file.
Action: Save the selected log file in local filesystem	When this option is enabled, Websense Content Gateway saves the selected log file on the local system in a location you specify.

Protocols

The Protocol configuration options are divided into the following categories:

HTTP, page 207

HTTP Responses, page 215

HTTP Scheduled Update, page 216

HTTPS, page 217

FTP, page 217

HTTP

Option	Description
	General
HTTP Proxy Server Port	Specifies the port that Websense Content Gateway uses when acting as a Web proxy server for HTTP traffic or when serving HTTP requests transparently. The default port is 8080.
	If you change this option, you must restart Websense Content Gateway.
URL Expandomatic	Enables or disables **.com** domain expansion. When this option is enabled, Websense Content Gateway attempts to resolve unqualified hostnames by redirecting them to the expanded address, prepended with www. and appended with **.com**. For example, if a client makes a request to *company*, Websense Content Gateway redirects the request to **www.company.com**.
	If local domain expansion is enabled (see *DNS Resolver*, page 246), Websense Content Gateway attempts local domain expansion before **.com** domain expansion; Websense Content Gateway tries **.com** domain expansion only if local domain expansion fails.
HTTPS Redirect	Specifies the range of ports used for tunneling. Websense Content Gateway allows tunnels only to the specified ports. For example, to retrieve an object using HTTPS via Websense Content Gateway, you must establish a tunnel via Websense Content Gateway to an origin server.
FTP over HTTP: Anonymous Password	Specifies the anonymous password Websense Content Gateway must use for FTP server connections that require a password. This option affects FTP requests from HTTP clients.

Option	Description
FTP over HTTP: Data Connection Mode	An FTP transfer requires two connections: a control connection to inform the FTP server of a request for data and a data connection to send the data. Websense Content Gateway always initiates the control connection. FTP mode determines whether Websense Content Gateway or the FTP server initiates the data connection.
	Select **PASV then PORT** for Websense Content Gateway to attempt PASV connection mode first. If PASV mode fails, Websense Content Gateway tries PORT mode and initiates the data connection. If successful, the FTP server accepts the data connection.
	Select **PASV only** for Websense Content Gateway to initiate the data connection to the FTP server. This mode is firewall friendly, but some FTP servers do not support it.
	Select **PORT only** for the FTP server to initiate the data connection and for Websense Content Gateway to accept the connection.
	The default value is **PASV then PORT**.
	Cacheability
Caching: HTTP Caching	Enables or disables HTTP caching. When this option is enabled, Websense Content Gateway serves HTTP requests from the cache. When this option is disabled, Websense Content Gateway acts as a proxy server and forwards all HTTP requests directly to the origin server.
Caching: FTP over HTTP Caching	Enables or disables FTP over HTTP caching. When this option is enabled, Websense Content Gateway serves FTP requests from HTTP clients from the cache. When this option is disabled, Websense Content Gateway acts as a proxy server and forwards all FTP requests from HTTP clients directly to the FTP server.
Behavior: Required Headers	Specifies the minimum header information required for an HTTP object to be cacheable.
	Select **An Explicit Lifetime Header** to cache only HTTP objects with **Expires or max-age** headers.
	Select **A Last-Modified Header** to cache only HTTP objects with **lastmodified** headers.
	Select **No Required Headers** to cache HTTP objects that do not have `Expires`, `max-age`, or `last-modified` headers. This is the default option.
	Caution: By default, Websense Content Gateway caches all objects (including objects with no headers). It is recommended that you change the default setting only for specialized proxy situations. If you configure Websense Content Gateway to cache only HTTP objects with **Expires** or `max-age` headers, the cache hit rate is reduced (very few objects have explicit expiration information).

Option	Description
Behavior: When to Revalidate	Specifies how Websense Content Gateway evaluates HTTP object freshness in the cache:
	Select **Never Revalidate** to never revalidate HTTP objects in the cache with the origin server (Websense Content Gateway considers all HTTP objects in the cache to be fresh).
	Select **Always Revalidate** to always revalidate HTTP objects in the cache with the origin server (Websense Content Gateway considers all HTTP objects in the cache to be stale).
	Select **Revalidate if Heuristic Expiration** to verify the freshness of an HTTP object with the origin server if the object contains no `Expires` or `Cache-Control` headers; Websense Content Gateway considers all HTTP objects without `Expires` or `Cache-Control` headers to be stale.
	Select **Use Cache Directive or Heuristic** to verify the freshness of an HTTP object with the origin server when Websense Content Gateway considers the object in the cache to be stale according to object headers, absolute freshness limit, and/or rules in the **cache.config** file. This is the default option.
	For more information about revalidation, see *Revalidating HTTP objects*, page 19.
Behavior: Add "no-cache" to MSIE Requests	Specifies when Websense Content Gateway adds `no-cache` headers to requests from Microsoft Internet Explorer.
	Certain versions of Microsoft Internet Explorer do not request cache reloads from transparent caches when the user presses the browser **Refresh** button. This can prevent content from being loaded directly from the origin servers. You can configure Websense Content Gateway to treat Microsoft Internet Explorer requests more conservatively, providing fresher content at the cost of serving fewer documents from cache.
	Select **To All MSIE Requests** to always add `no-cache` headers to all requests from Microsoft Internet Explorer.
	Select **To IMS MSIE Requests** to add `no-cache` headers to IMS (If Modified Since) Microsoft Internet Explorer requests.
	Select Not to Any MSIE Requests to never add `no-cache` headers to requests from Microsoft Internet Explorer.
Behavior: Ignore "no-cache" in Client Requests	When this option is enabled, Websense Content Gateway ignores `no-cache` headers in client requests and serves the requests from the cache.
	When this option is disabled, Websense Content Gateway does not serve requests with `no-cache` headers from the cache but forwards them to the origin server.

Option	Description
Freshness: Minimum Heuristic Lifetime	Specifies the minimum amount of time that an HTTP object can be considered fresh in the cache.
Freshness: Maximum Heuristic Lifetime	Specifies the maximum amount of time that an HTTP object can be considered fresh in the cache.
Freshness: FTP Document Lifetime	Specifies the maximum amount of time that an FTP file can stay in the cache. This option affects FTP requests from HTTP clients only.
Maximum Alternates	Specifies the maximum number of alternate versions of HTTP objects Websense Content Gateway can cache.
	Caution: If you enter 0 (zero), there is no limit to the number of alternates cached. If a popular URL has thousands of alternates, you might observe increased cache hit latencies (transaction times) as Websense Content Gateway searches over the thousands of alternates for each request. In particular, some URLs can have large numbers of alternates due to cookies. If Websense Content Gateway is set to vary on cookies, you might encounter this problem.
Vary Based on Content Type: Enable/ Disable	Enables or disables caching of alternate versions of HTTP documents that do not contain the **Vary** header. If no `Vary` header is present, Websense Content Gateway varies on the headers specified below, depending on the document's content type.
Vary by Default on Text	Specifies the header field on which Websense Content Gateway varies for text documents.
Vary by Default on Images	Specifies the header field on which Websense Content Gateway varies for images.
Vary by Default on Other Document Types	Specifies the header field on which Websense Content Gateway varies for anything other than text and images.
Dynamic Caching: Caching Documents with Dynamic URLs	When this option is enabled, Websense Content Gateway attempts to cache dynamic content. Content is considered dynamic if it contains a question mark (?), a semicolon (;), **cgi**, or if it ends in **.asp**.
	Caution: It is recommended that you configure Websense Content Gateway to cache dynamic content for specialized proxy situations only.
Dynamic Caching: Caching Response to Cookies	Specifies how responses to requests that contain cookies are cached:
	Select **Cache All but Text** to cache cookies that contain any type of content except text. This is the default.
	Select **Cache Only Image Types** to cache cookies that contain images only.
	Select **Cache Any Content-Type** to cache cookies that contain any type of content.
	Select **No Cache on Cookies** to not cache cookies at all.

Option	Description
Caching Policy/Forcing Document Caching	Displays a table listing the rules in the **cache.config** file that specify how a particular group of URLs should be cached. This file also lets you force caching of certain URLs for a specific amount of time.
Refresh	Updates the table to display the most up-to-date rules in the **cache.config** file. Click **Refresh** after you have added or modified rules with the configuration file editor.
Edit File	Opens the configuration file editor so that you can edit and add rules to the **cache.config** file.
	cache.config Configuration File Editor
Rule display box	Lists the **cache.config** file rules. Select a rule to edit it. The buttons on the left of the box allow you to delete or move the selected rule up or down in the list.
Add	Adds a new rule to the rule display box at the top of the configuration file editor page.
Set	Updates the rule display box at the top of the configuration file editor page.
Rule Type	Lists the type of rules you can add to the **cache.config** file:
	A `never-cache` rule configures Websense Content Gateway to never cache specified objects.
	An `ignore-no-cache` rule configures Websense Content Gateway to ignore all `Cache-Control: no-cache` headers.
	An `ignore-client-no-cache` rule configures Websense Content Gateway to ignore `Cache-Control: no-cache` headers from client requests.
	An `ignore-server-no-cache` rule configures Websense Content Gateway to ignore `Cache-Control: no-cache` headers from origin server responses.
	A `pin-in-cache` rule configures Websense Content Gateway to keep objects in the cache for a specified time.
	A `revalidate` rule configures Websense Content Gateway to consider objects fresh in the cache for a specified time.
	A `ttl-in-cache` rule configures Websense Content Gateway to serve certain HTTP objects from the cache for the amount of time specified in the **Time Period** field regardless of certain caching directives in the HTTP request and response headers.

Option	Description
Primary Destination Type	Lists the primary destination types: `dest_domain` is a requested domain name. `dest_host` is a requested hostname. `dest_ip` is a requested IP address. `url_regex` is a regular expression to be found in a URL.
Primary Destination Value	Specifies the value of the primary destination type. For example, if the Primary Destination Type is `dest_ip`, the value for this field can be 123.456.78.9.
Additional Specifier: Time Period	Specifies the amount of time that applies to the `revalidate`, `pinin-cache`, and `ttl-in-cache` rule types. The following time formats are allowed: *d* for days (for example 2d) *h* for hours (for example, 10h) *m* for minutes (for example, 5m) *s* for seconds (for example, 20s) mixed units (for example, 1h15m20s)
Secondary Specifiers: Time	Specifies a time range, such as 08:00-14:00.
Secondary Specifiers: Prefix	Specifies a prefix in the path part of a URL.
Secondary Specifiers: Suffix	Specifies a file suffix in the URL.
Secondary Specifiers: Source IP	Specifies the IP address of the client.
Secondary Specifiers: Port	Specifies the port in a requested URL.
Secondary Specifiers: Method	Specifies a request URL method.
Secondary Specifiers: Scheme	Specifies the protocol of a requested URL.
Secondary Specifiers: MIXT Scheme	Specifies the media protocol type of a requested URL.
Apply	Applies the configuration changes.
Close	Exits the configuration file editor. Click **Apply** before you click **Close**; otherwise, all configuration changes will be lost.
	Privacy
Insert Headers: Client-IP	When this option is enabled, Websense Content Gateway inserts the `Client-IP` header in outgoing requests if the request does not contain an `Client-IP` header.

Option	Description
Remove Headers: Client-IP	When this option is enabled, Websense Content Gateway removes the `Client-IP` header from outgoing requests to protect the privacy of your users.
Remove Headers: Cookie	When this option is enabled, Websense Content Gateway removes the `Cookie` header from outgoing requests to protect the privacy of your users. The `Cookie` header often identifies the user that makes a request.
Remove Headers: From	When this option is enabled, Websense Content Gateway removes the `From` header from outgoing requests to protect the privacy of your users. The `From` header identifies the client's email address.
Remove Headers: Referer	When this option is enabled, Websense Content Gateway removes the `Referer` header from outgoing requests to protect the privacy of your users. The `Referer` header identifies the Web link that the client selects.
Remove Headers: User-Agent	When this option is enabled, Websense Content Gateway removes the `User-Agent` header from outgoing requests to protect the privacy of your users. The `User-Agent` header identifies the agent that is making the request, usually a browser.
Remove Headers: Remove Others	Specifies headers other than `From`, `Referer`, `User-Agent`, and `Cookie`, that you want to remove from outgoing requests to protect the privacy of your users.
	Timeouts
Keep-Alive Timeouts: Client	Specifies (in seconds) how long Websense Content Gateway keeps connections to clients open for a subsequent request after a transaction ends. Each time Websense Content Gateway opens a connection to accept a client request, it handles the request and then keeps the connection alive for the specified timeout period. If the client does not make another request before the timeout expires, Websense Content Gateway closes the connection. If the client does make another request, the timeout period starts again. The client can close the connection at any time.
Keep-Alive Timeouts: Origin Server	Specifies (in seconds) how long Websense Content Gateway keeps connections to origin servers open for a subsequent transfer of data after a transaction ends. Each time Websense Content Gateway opens a connection to download data from an origin server, it downloads the data and then keeps the connection alive for the specified timeout period. If Websense Content Gateway does not need to make a subsequent request for data before the timeout expires, it closes the connection. If it does, the timeout period starts again. The origin server can close the connection at any time.

Option	Description
Inactivity Timeouts: Client	Specifies how long Websense Content Gateway keeps connections to clients open if a transaction stalls. If Websense Content Gateway stops receiving data from a client or the client stops reading the data, Websense Content Gateway closes the connection when this timeout expires.
	The client can close the connection at any time.
Inactivity Timeouts: Origin	Specifies how long Websense Content Gateway keeps connections to origin servers open if the transaction stalls. If Websense Content Gateway stops receiving data from an origin server, it does not close the connection until this timeout has expired.
	The origin server can close the connection at any time.
Active Timeouts: Client	Specifies how long Websense Content Gateway remains connected to a client. If the client does not finish making a request (reading and writing data) before this timeout expires, Websense Content Gateway closes the connection.
	The default value of 0 (zero) specifies that there is no timeout.
	The client can close the connection at any time.
Active Timeouts: Origin Server	Specifies how long Websense Content Gateway waits for fulfillment of a connection request to an origin server. If Websense Content Gateway does not establish connection to an origin server before this timeout expires, Websense Content Gateway terminates the connection request.
	The default value of 0 (zero) specifies that there is no timeout.
	The origin server can close the connection at any time.
FTP Control Connection Timeout	Specifies how long Websense Content Gateway waits for a response from an FTP server. If the FTP server does not respond within the specified time, Websense Content Gateway abandons the client's request for data. This option affects FTP requests from HTTP clients only.

HTTP Responses

Option	Description
	General
Response Suppression Mode	If Websense Content Gateway detects an HTTP problem with a particular client transaction (such as unavailable origin servers, authentication requirements, and protocol errors), it sends an HTML response to the client browser. Websense Content Gateway has a set of hard-coded default response pages that explain each HTTP error in detail to the client.
	Select **Always Suppressed** if you do not want to send HTTP responses to clients.
	Select **Intercepted Traffic Only** if you want to send HTTP responses to nontransparent traffic only. (This option is useful when Websense Content Gateway is running transparently and you do not want to indicate the presence of a cache.)
	Select **Never Suppressed** if you want to send HTTP responses to all clients.
	If you change this option, you must restart Websense Content Gateway.
	Custom
Custom Responses	You can customize the responses Websense Content Gateway sends to clients. By default, the responses you can customize are located in the Websense Content Gateway **config/body_factory/default** directory.
	Select **Enabled Language-Targeted Response** to send your custom responses to clients in the language specified in the `Accept-Language` header.
	Select **Enabled in "default" Directory Only** to send the custom responses located in the default directory to clients.
	Select **Disabled** to disable the custom responses. If **Never Suppressed** or **Intercepted Traffic Only** is selected for the **Response Suppression Mode** option, Websense Content Gateway sends the hard-coded default responses.
	If you change this option, you must restart Websense Content Gateway.
Custom Response Logging	When enabled, Websense Content Gateway sends a message to the error log each time custom responses are used or modified.
	If you change this option, you must restart Websense Content Gateway.
Custom Response Template Directory	Specifies the directory where the custom responses are located. The default location is the Websense Content Gateway **config/body_factory** directory.
	If you change this option, you must restart Websense Content Gateway.

HTTP Scheduled Update

Option	Description
	General
Scheduled Update	Enables or disables the scheduled update option. When this option is enabled, Websense Content Gateway can automatically update certain objects in the local cache at a specified time.
Maximum Concurrent Updates	Specifies the maximum number of simultaneous update requests allowed at any point. This option enables you to prevent the scheduled update process from overburdening the host. The default value is 100.
Retry on Update Error: Count	Specifies the number of times Websense Content Gateway retries the scheduled update of a URL in the event of failure. The default value is 10 times.
Retry on Update Error: Interval	Specifies the delay in seconds between each scheduled update retry for a URL in the event of failure. The default value is 2 seconds.
	Update URLs
Force Immediate Update	When enabled, Websense Content Gateway overrides the scheduling expiration time for all scheduled update entries and initiates updates every 25 seconds.
Scheduled Object Update	Displays a table listing the rules in the *update.config* file that control how Websense Content Gateway performs a scheduled update of specific local cache content.
Refresh	Updates the table to display the most up-to-date rules in the **update.config** file.
Edit File	Opens the configuration file editor so that you can edit and add rules to the **update.config** file.
	update.config Configuration File Editor
rule display box	Lists the **update.config** file rules. Select a rule to edit it. The buttons on the left of the box allow you to delete or move the selected rule up or down in the list.
Add	Adds a new rule to the rule display box at the top of the configuration file editor page.
Set	Updates the rule display box at the top of the configuration file editor page.
URL	Specifies the URL to be updated.
Request Headers (Optional)	Specifies the list of headers (separated by semi-colons) passed in each GET request. You can define any request header that conforms to the HTTP specification. The default is no request header.
Offset Hour	Specifies the base hour used to derive the update periods. The range is 00-23 hours.

Option	Description
Interval	The interval, in seconds, at which updates should occur, starting at Offset Hour.
Recursion Depth	The depth to which referenced URLs are recursively updated, starting at the given URL. For example, a recursion depth of 1 will update the given URL, as well as all URLs immediately referenced by links from the original URL.

HTTPS

Option	Description
	General
HTTPS Proxy Server Port	Specifies the port that Websense Content Gateway uses when acting as a Web proxy server for HTTPS traffic or when serving HTTPS traffic transparently. The default is 8070. This is also known as the SSL Inbound Port.
SSL Outbound Port	Specifies the port HTTPS traffic goes to for re-encryption before being sent to its destination. The default is 8090.

FTP

✓ **Note**
The FTP configuration options appear on the Configure pane only if you have enabled FTP processing in the Features table on the **Configure > My Proxy > Basic > General** tab.

Option	Description
	General
FTP Proxy Server Port	Specifies the port that Websense Content Gateway uses to accept FTP requests. The default port is 21.
Listening Port Configuration	Specifies how FTP opens a listening port for a data transfer.
	Select **Default Settings** to let the operating system choose an available port. Websense Content Gateway sends 0 and retrieves the new port number if the listen succeeds.
	Select **Specify Range** if you want the listening port to be determined by the range of ports specified in the **Listening Port (Max)** and **Listening Port (Min)** fields.

Option	Description
Default Data Connection Method	Specifies the default method used to set up data connections with the FTP server. Select **Proxy Sends PASV** to send a PASV to the FTP server and let the FTP server open a listening port. Select **Proxy Sends PORT** to set up a listening port on the Websense Content Gateway side of the connection first.
Shared Server Connections	When enabled, server control connections can be shared between multiple anonymous FTP clients.
	Cacheability
FTP Caching	Enables or disables FTP caching. When this option is enabled, the proxy serves FTP requests from FTP clients from the cache. When this option is disabled, Websense Content Gateway acts as a proxy server and forwards all FTP requests from FTP clients directly to the FTP server.
Directory Caching: Simple	When this option is enabled, the proxy caches directory listings without arguments (for example, **dir/ls**).
Directory Caching: Full	When this option is enabled, the proxy caches directory listings with arguments (for example, `ls -al`, `ls *.txt`).
Freshness: Login Information	Specifies how long (in seconds) the 220/230 responses (login messages) can stay fresh in the cache. The default value is 2.592 million seconds (30 days).
Freshness: Directory Listings	Specifies how long (in seconds) directory listings can stay fresh in the cache. The default value is 604800 seconds (7 days).
Freshness: Files	Specifies how long (in seconds) FTP files can stay fresh in the cache. The default value is 259200 seconds (3 days).
	Timeouts
Keep-Alive Timeout: Server Control	Specifies the timeout value when the FTP server control connection is not used by any FTP clients. The default value is 90 seconds.
Inactivity Timeouts: Client Control	Specifies how long FTP client control connections can remain idle. The default value is 900 seconds.
Inactivity Timeouts: Server Control	Specifies how long the FTP server control connection can remain idle. The default value is 900 seconds.
Active Timeouts: Client Control	Specifies the how long FTP client control connections can remain open. The default value is 14400 seconds.
Active Timeouts: Server Control	Specifies how long the FTP server control connection can remain open. The default value is 14400 seconds.

Content Routing

The Content Routing configuration options are divided into the following categories:

Hierarchies, page 219

Mapping and Redirection, page 222

Browser Auto-Config, page 224

Hierarchies

Option	Description
	Parenting
Parent Proxy	Enables or disables the HTTP parent caching option. When this option is enabled, Websense Content Gateway can participate in an HTTP cache hierarchy. You can point your Websense Content Gateway server at a parent network cache (either another Websense Content Gateway server or a different caching product) to form a cache hierarchy where a child cache relies upon a parent cache in fulfilling client requests.) See *HTTP cache hierarchies*, page 69.
No DNS and Just Forward to Parent	When this option is enabled, (and if HTTP parent caching is enabled), Websense Content Gateway does no DNS lookups on requested hostnames. If rules in the **parent.config** file are set so that only selected requests are sent to a parent proxy, Websense Content Gateway skips name resolution only for requests that are going to the parent proxy. Name resolution is performed as usual for requests that are not sent to a parent proxy. If the parent proxy is down and the child proxy can go directly to origin servers, the child does a NDNS resolution as required only when the parent is unavailable.
Parent Proxy Cache Rules	Displays a table listing the rules in the *parent.config* file that identify the HTTP parent proxies used in an HTTP cache hierarchy and configure selected URL requests to bypass parent proxies.
Refresh	Updates the table to display the most up-to-date rules in the **parent.config** file.
Add	Opens the configuration file editor so that you can edit and add rules to the **parent.config** file.
	parent.config Configuration File Editor
rule display box	Lists the *parent.config* file rules. Select a rule to edit it. The buttons on the left of the box allow you to delete or move the selected rule up or down in the list.
Add	Adds a new rule to the rule display box at the top of the configuration file editor page.

Option	Description
Set	Updates the rule display box at the top of the configuration file editor page.
Primary Destination Type	Lists the primary destination types: `dest_domain` is a requested domain name. `dest_host` is a requested hostname. `dest_ip` is a requested IP address. `url_regex` is a regular expression to be found in a URL.
Primary Destination Value	Specifies the value of the primary destination type. For example: If the primary destination is `dest_domain`, a value for this field can be yahoo.com If the primary destination type is `dest_ip`, the value for this field can be 123.456.78.9. If the primary destination is `url_regex`, a value for this field can be politics.
Parent Proxies	Specifies the IP addresses or hostnames of the parent proxies and the port numbers used for communication. Parent proxies are queried in the order specified in the list. If the request cannot be handled by the last parent server in the list, it is routed to the origin server. Separate each entry with a semicolon; for example: `parent1:8080; parent2:8080`
Round Robin	Select **true** for the proxy to go through the parent cache list in a round-robin based on client IP address. Select **strict** for the proxy to serve requests strictly in turn. For example, machine proxy1 serves the first request, proxy2 serves the second request, and so on. Select **false** if you do not want round-robin selection to occur.
Go direct	Select **true** for requests to bypass parent hierarchies and go directly to the origin server. Select **false** if you do not want requests to bypass parent hierarchies.
Secondary Specifiers: Time	Specifies a time range, using a 24-hour clock, such as 08:00-14:00. If the range crosses midnight, enter this as two comma-separated ranges. For example, if a range extends from 6:00 in the evening until 8:00 in the morning, enter the following: `18:00 - 23:59, 0:00 - 8:00`
Secondary Specifiers: Prefix	Specifies a prefix in the path part of a URL.
Secondary Specifiers: Suffix	Specifies a file suffix in the URL, such as .htm or .gif.
Secondary Specifiers: Source IP	Specifies the IP address of the client.
Secondary Specifiers: Port	Specifies the port in a requested URL.

Option	Description
Secondary Specifiers: Method	Specifies a request URL method. For example: • `get` • `post` • `put` • `trace`
Secondary Specifiers: Scheme	Specifies the protocol of a requested URL. This must be either HTTP or FTP.
Secondary Specifiers: MIXT Scheme	Specifies the media protocol type of a requested URL.
	ICP Peering
ICP Mode	Specifies the ICP mode for ICP cache hierarchies: Select **Only Receive Queries** to configure the proxy to receive ICP queries only. Select **Send/Receive Queries** to configure the proxy to both send and receive ICP queries. Select **Disabled** to turn off ICP hierarchical caching. See *ICP cache hierarchies*, page 71, for more information.
ICP Interface	Specifies the interface used for ICP messages. This field is populated automatically, but if the machine is running with more than one interface, you can specify a different one here.
ICP Port	Specifies the port Websense Content Gateway uses for ICP messages. The default port is 3130.
ICP Multicast	Enables or disables ICP multicast. Select **Enabled** to send ICP messages through multicast if your proxy server has a multicast channel connection to its peers.
ICP Query Timeout	Specifies the timeout for ICP queries. The default is 2 seconds.
ICP Peers	Displays a table listing the rules in the *icp.config* file that specify the ICP peers (parent and sibling caches).
Refresh	Updates the table to display the most up-to-date rules in the **icp.config** file. Click this button after you have added or modified rules with the configuration file editor.
Edit File	Opens the configuration file editor so that you can edit and add rules to the **icp.config** file.
	icp.config Configuration File Editor
rule display box	Lists the *icp.config* file rules. Select a rule to edit it. The buttons on the left of the box allow you to delete or move the selected rule up or down in the list.
Add	Adds a new rule to the rule display box at the top of the configuration file editor page. Enter information in the fields provided before you click this button.
Peer Hostname	Specifies the hostname of the ICP Peer. This field is optional if the IP address of the ICP peer is specified in the **Peer IP** field below.

Option	Description
Peer IP	Specifies the IP address of the ICP Peer.
	This field is optional if the hostname of the ICP peer is specified in the **Peer Hostname** field above.
Peer Type	Specifies the type of ICP peer: parent or sibling.
Proxy Port	Specifies the port number of the TCP port used by the ICP peer for proxy communication.
ICP Port	Specifies the port number of the UDP port used by the ICP peer for ICP communication.
Multicast	Enables or disables multicast mode.
Multicast IP	Specifies the multicast IP address if multicast is enabled.
Multicast TTL	Specifies the multicast time to live.
	Select **single subnet** if IP multicast datagrams will not be forwarded beyond a single subnetwork.
	Select **multiple subnets** to allow delivery of IP multicast datagrams to more than one subnet (if there are one or more multicast routers attached to the first hop subnet)

Mapping and Redirection

Option	Description
Serve Mapped Hosts Only	Select **Required** if you want the proxy to serve requests only to origin servers listed in the mapping rules of the *remap.config* file. If a request does not match a rule in the **remap.config** file, the browser receives an error. This option provides added security for your Websense Content Gateway system.
Retain Client Host Header	When this option is enabled, Websense Content Gateway retains the client host header in a request (it does not include the client host header in the mapping translation).
Redirect No-Host Header to URL	Specifies the alternate URL to which to direct incoming requests from older clients that do not provide a `Host:` header.
	It is recommended that you set this option to a page that explains the situation to the user and advises a browser upgrade or provides a link directly to the origin server, bypassing the proxy. Alternatively, you can specify a map rule that maps requests without `Host:` headers to a particular server.
URL Remapping Rules	Displays a table listing the mapping rules in the **remap.config** file so that you can redirect HTTP requests permanently or temporarily without the proxy having to contact any origin servers.
Refresh	Updates the table to display the most up-to-date rules in the **remap.config** file.

Option	Description
Edit File	Opens the configuration file editor so that you can edit and add rules to the **remap.config** file.
	remap.config Configuration File Editor
rule display box	Lists the **remap.config** file rules. Select a rule to edit it. The buttons on the left of the box allow you to delete or move the selected rule up or down in the list.
Add	Adds a new rule to the rule display box at the top of the configuration file editor page.
Set	Updates the rule display box at the top of the configuration file editor page.
Rule Type	Lists the type of rules you can add to the remap.config file:
	redirect redirects HTTP requests permanently without having to contact the origin server. Permanent redirects notify the browser of the URL change (by returning an HTTP status code 301) so that the browser can update bookmarks.
	redirect_temporary redirects HTTP requests temporarily without having to contact the origin server. Temporary redirects notify the browser of the URL change for the current request only (by returning an HTTP status code 307).
Scheme	Specifies the protocol of the mapping rule.
From Host	Specifies the hostname of the URL to map from.
From Port (Optional)	Specifies the port number in the URL to map from.
From Path Prefix (Optional)	Specifies the path prefix of the URL to map from.
To Host	Specifies the hostname of the URL to map to.
To Port (Optional)	Specifies the port number of the URL to map to.
To Path Prefix (Optional)	Specifies the path prefix of the URL to map to.
	FTP
FTP Remapping Rules	Displays a table listing the FTP mapping rules in the **ftp_remap.config** file. Websense Content Gateway uses the rules to direct any incoming FTP requests to the FTP server if the requested document is a cache miss or is stale.
Refresh	Updates the table to display the most up-to-date rules in the **ftp_remap.config** file.
Add	Opens the configuration file editor so that you can edit and add rules to the **ftp_remap.config** file.

Option	Description
	ftp_remap.config Configuration File Editor
rule display box	Lists the **ftp_remap.config** file rules. Select a rule to edit it. The buttons on the left of the box allow you to delete or move the selected rule up or down in the list.
Add	Adds a new rule to the rule display box at the top of the configuration file editor page.
Set	Updates the rule display box at the top of the configuration file editor page.
Proxy Hostname	Specifies the IP address or hostname assigned to Websense Content Gateway.
Proxy Port	Specifies the proxy port used by Websense Content Gateway to listen to FTP traffic.
FTP Server	Specifies the IP address or hostname of the FTP server to which you want to redirect requests.
FTP Server Port	Specifies the port number used by the FTP server to handle requests.

Browser Auto-Config

Option	Description
	PAC
Auto-Configuration Port	Specifies the port Websense Content Gateway uses to download the auto-configuration file to browsers. The port cannot be assigned to any other process. The default port is 8083.
	If you change this option, you must restart Websense Content Gateway.
PAC Settings	Lets you edit the PAC file (**proxy.pac**). See *Using a PAC file*, page 36.
	WPAD
WPAD Settings	Lets you edit the **wpad.dat** file. See *Using WPAD*, page 37.

Security

The Security configuration options are divided into the following categories:

Connection Control

Option	Description
	Proxy Access
Access Control	Displays the rules in the *ip_allow.config* file that control which clients can access the proxy cache.
	By default, all remote hosts are allowed to access the proxy cache.
Refresh	Updates the table to display the most up-to-date rules in the **ip_allow.config** file.
Edit File	Opens the configuration file editor for to the **ip_allow.config** file.
	ip_allow.config Configuration File Editor
rule display box	Lists the *ip_allow.config* file rules. Select a rule to edit it. The buttons on the left of the box allow you to delete or move the selected rule up or down in the list.
Add	Adds a new rule to the rule display box at the top of the configuration file editor page.
Set	Updates the rule display box at the top of the configuration file editor page.
IP Action	Lists the type of rules you can add.
	An ip_allow rule allows the clients listed in the **Source IP** field to access the proxy cache.
	An ip_deny rule denies the clients listed in the **Source IP** field access to the proxy cache.
Source IP	Specifies the IP address or range of IP addresses of the clients that are allowed or denied access to the proxy cache.
Apply	Applies the configuration changes.
Close	Exits the configuration file editor.
	Click **Apply** before you click **Close**; otherwise, all configuration changes will be lost.
	ARM Security
ARM Security	Enables or disables the ARM security option that restricts the type of communication possible with machines running Websense Content Gateway. For information about the ARM security option, see *Controlling host access to the proxy server*, page 102.
	If you change this option, you must restart Websense Content Gateway.
Access Control List	Displays a table listing the rules in the *arm_security.config* file that restrict the type of communication possible with machines running Websense Content Gateway. For more details about the ARM security feature, see *Controlling host access to the proxy server*, page 102.

Option	Description
Refresh	Updates the table to display the most up-to-date rules in the **arm_security.config** file.
Edit File	Opens the configuration file editor for the **arm_security.config** file.
	arm_security.config Configuration File Editor
rule display box	Lists the *arm_security.config* file rules. Select a rule to edit it. The buttons on the left of the box allow you to delete or move the selected rule up or down in the list.
Add	Adds a new rule to the rule display box at the top of the configuration file editor page.
Set	Updates the rule display box at the top of the configuration file editor page.
Rule Type	Specifies the rule type:
	An `open` rule specifies the ports that are open by default, for either TCP or UDP. Specify the ports you want to open in the **Open Port** field.
	A `deny` rule specifies the hosts that are denied access to specific destination ports, for either TCP or UDP.
	An `allow` rule specifies the hosts that are allowed access to specific destination ports, for either TCP or UDP.
Connection Type	Specifies the type of connection used: TCP or UDP.
Source IP	Specifies the IP address or range of IP addresses of the source of the communication.
Source Port	Specifies the source port or range of source ports from which TCP traffic is allowed.
Destination IP	Specifies the IP address or range of IP addresses of the destination of the communication.
Destination Port	Specifies the destination port or range of destination ports from which TCP traffic is allowed or denied.
Open Port	Specifies the port or series of ports that are open by default.
Apply	Applies the configuration changes.
Close	Exits the configuration file editor.
	Click **Apply** before you click **Close**; otherwise, all configuration changes will be lost.

Access Control

Option	Description
	Filtering
Filtering	Displays a table listing the rules in the *filter.config* file that deny or allow particular URL requests, keep or strip header information from client requests, and specify LDAP authentication rules.
Refresh	Updates the table to display the most up-to-date rules in the **filter.config** file.
Edit File	Opens the configuration file editor for the **filter.config** file.
	filter.config Configuration File Editor
rule display box	Lists the *filter.config* file rules. Select a rule to edit it. The buttons on the left of the box allow you to delete or move the selected rule up or down in the list.
Add	Adds a new rule to the rule display box at the top of the configuration file editor page.
Set	Updates the rule display box at the top of the configuration file editor page.
Rule Type	Specifies the rule type: Select **allow** to allow particular URL requests; the proxy caches and serves the requested content. Select **deny** to deny requests for objects from specific destinations. When a request is denied, the client receives an access denied message. Select **ldap** to specify authentication rules that determine which users must be authenticated to access particular sites on the Internet and which LDAP servers are used. Select **ntlm** to specify authentication rules that determine which users must be authenticated to access particular sites on the Internet and which domain controllers are used. Select **radius** to specify authentication rules that determine which users must be authenticated to access particular sites on the Internet and which RADIUS servers are used. Select **keep_hdr** to specify which client request header information you want to keep. Select **strip_hdr** to specify which client request header information you want to strip.

Option	Description
Primary Destination Type	Lists the primary destination types: `dest_domain` is a requested domain name. `dest_host` is a requested hostname. `dest_ip` is a requested IP address. `url_regex` is a regular expression to be found in a URL.
Primary Destination Value	Specifies the value of the primary destination type. For example, if the primary destination type is `dest_ip`, the value for this field can be 123.456.78.9.
Additional Specifiers: Header Type	Specifies the client request header information that you want to keep or strip. This option applies to `keep_hdr` or `strip_hdr` rule types only.
Additional Specifiers: LDAP Server Name	Specifies the LDAP server name. This option applies to `ldap` rule types only.
Additional Specifiers: LDAP Base Distinguished Name	Specifies the LDAP Base Distinguished Name. This option applies to `ldap` rule types only.
Additional Specifiers: LDAP UID filter (Optional)	Specifies the LDAP UID filter. This option applies to `ldap` rule types only.
Additional Specifiers: LDAP Attribute Name (Optional)	Specifies the LDAP attribute name. This option applies to `ldap` rule types only.
Additional Specifiers: LDAP Attribute Value (Optional)	Specifies the LDAP attribute pair. This option applies to `ldap` rule types only.
Additional Specifiers: Realm (Optional)	Specifies the realm name for LDAP, NTLM, or Radius. This option applies to `ldap` or `ntlm` rule types only.
Secondary Specifiers: Time	Specifies a time range, such as 08:00-14:00.
Secondary Specifiers: Prefix	Specifies a prefix in the path part of a URL.
Secondary Specifiers: Suffix	Specifies a file suffix in the URL.
Secondary Specifiers: Source IP	Specifies the IP address of the client.
Secondary Specifiers: Port	Specifies the port in a requested URL.
Secondary Specifiers: Method	Specifies a request URL method.
Secondary Specifiers: Scheme	Specifies the protocol of a requested URL.

Option	Description
Apply	Applies the configuration changes.
Close	Exits the configuration file editor.
	Click **Apply** before you click **Close**; otherwise, all configuration changes will be lost.
	LDAP

Note: The LDAP configuration options appear on the Configure pane only if you have enabled LDAP in the Features table on the **Configure > My Proxy > Basic > General** tab.

Option	Description
Purge Cache on Authentication Failure	When this option is enabled, Websense Content Gateway deletes the authorization entry for the client in the LDAP cache if authorization fails.
LDAP Server: Hostname	Specifies the hostname of the LDAP server.
	If you change this option, you must restart Websense Content Gateway.
LDAP Server: Port	Specifies the port used for LDAP communication. The default port number is 389.
	If you change this option, you must restart Websense Content Gateway.
LDAP Server: Server Type	Specifies the search filter. Select either Active Directory or other directory services.
LDAP: Bind Distinguished Name	Specifies the Full Distinguished Name (fully qualified name) of a user in the LDAP-based directory service. For example:
	`CN=John Smith,CN=USERS,DC=MYCOMPANY,DC=COM`
	Enter a maximum of 128 characters in this field.
	If you do not specify a value for this field, the proxy attempts to bind anonymously.
LDAP: Password	Specifies a password for the user identified in the **Bind_DN** field.
LDAP Base Distinguished Name	Specifies the base Distinguished Name (DN). You can obtain this value from your LDAP administrator.
	You must specify a correct base DN; otherwise LDAP authentication will fail to operate.
	If you change this option, you must restart Websense Content Gateway.
	Radius

Note: The Radius configuration options appear on the Configure pane only if you have enabled Radius in the Features table on the **Configure > My Proxy > Basic > General** tab.

Option	Description
Primary Radius Server: Hostname	Specifies the hostname or IP address of the primary RADIUS authentication server.
	If you change this option, you must restart Websense Content Gateway.

Option	Description
Primary Radius Server: Port	Specifies the port that Websense Content Gateway uses to communicate with the primary RADIUS authentication server. The default port is 1812.
	If you change this option, you must restart Websense Content Gateway.
Primary Radius Server: Shared Key	Specifies the key to use for encoding.
	If you change this option, you must restart Websense Content Gateway.
Secondary Radius Server (Optional): Hostname	Specifies the hostname or IP address of the secondary RADIUS authentication server.
	If you change this option, you must restart Websense Content Gateway.
Secondary Radius Server (Optional): Port	Specifies the port that Websense Content Gateway uses to communicate with the secondary RADIUS authentication server. The default port is 1812.
	If you change this option, you must restart Websense Content Gateway.
Secondary Radius Server (Optional): Shared Key	Specifies the key to use for encoding.
	If you change this option, you must restart Websense Content Gateway.
	NTLM

Note: The NTLM configuration options appear on the Configure pane only if you have enabled NTLM in the Features table on the **Configure > My Proxy > Basic > General** tab.

Option	Description
Domain Controller Hostnames	Specifies the hostnames of the domain controllers. You must separate each entry with a comma; for example:
	`host1, host2, host3`
	If you change this option, you must restart Websense Content Gateway.
NT Domain Name	Specifies the NT domain name against which Websense Content Gateway should authenticate.
	If you change this option, you must restart Websense Content Gateway.
Load Balancing	Enables or disables load balancing. When enabled, Websense Content Gateway balances the load when sending authentication requests to the domain controllers.
	If you change this option, you must restart Websense Content Gateway.

SOCKS

✓ **Note**
The SOCKS configuration options appear on the
Configure pane only if you have enabled SOCKS in the
Features table on the **Configure > My Proxy > Basic >
General** tab.

Option	Description
	General
SOCKS Version	Specifies the version of SOCKS used on your SOCKS server. Websense Content Gateway supports SOCKS version 4 and version 5.
	If you change this option, you must restart Websense Content Gateway.
	Proxy
SOCKS Proxy	Enables or disables the SOCKS Proxy option. As a SOCKS proxy, WCG can receive SOCKS packets (usually on port 1080) from the client and forwards all requests directly to the SOCKS server.
	For more information about the SOCKS Proxy option, see *Configuring SOCKS firewall integration*, page 106.
	If you change this option, you must restart Websense Content Gateway.
SOCKS Proxy Port	Specifies the port on which Websense Content Gateway accepts SOCKS traffic. This is usually port 1080.
	If you change this option, you must restart Websense Content Gateway.
	Server
SOCKS Server: Default Servers	Specifies the names and the ports of the default SOCKS servers with which Websense Content Gateway communicates. Each entry must be separated by a semicolon (;); for example: `socks1:1080;socks2:4080`
	If you change this option, you must restart Websense Content Gateway.
	You can perform additional SOCKS server configuration in the *socks.config* file.

Option	Description
Socks Server Rules	Displays a table listing the rules in the `socks.config` file that control the SOCKS servers Websense Content Gateway must go through to access specific origin servers and the order in which Websense Content Gateway goes through the SOCKS server list. You can also specify the origin servers that you want the proxy to access directly without going through the SOCKS server, and the user name and password used by Websense Content Gateway to connect to a SOCKS version 5 server.
Refresh	Updates the table to display the most up-to-date rules in the **socks.config** file.
Edit File	Opens the configuration file editor for the **socks.config** file.
	socks.config Configuration File Editor
rule display box	Lists the *socks.config* file rules. Select a rule to edit it. The buttons on the left of the box allow you to delete or move the selected rule up or down in the list.
Add	Adds a new rule to the rule display box at the top of the configuration file editor page.
Set	Updates the rule display box at the top of the configuration file editor page.
Rule Type	Specifies the rule type:
	Select **no_socks** to specify the origin servers you want the proxy to access directly without going through the SOCKS server. Enter the IP addresses of the origin servers in the **Destination IP** field.
	Select **auth** to specify the user name and password Websense Content Gateway uses for authentication with a SOCKS version 5 server. Enter the username in the **Username** field and the password in the **Password** field.
	Select **multiple_socks** to specify the SOCKS servers Websense Content Gateway must go through to reach specific origin servers. Enter the hostnames or the IP addresses of the SOCKS servers in the **SOCKS Servers** field and the IP addresses of the origin servers in the **Destination IP** field. Select how strictly Websense Content Gateway should follow round robin from the **Round Robin** drop-down list.
Username	Specifies the username Websense Content Gateway must use to authenticate with a SOCKS version 5 server. This field applies to an `auth` rule type only.
Password	Specifies the password Websense Content Gateway must use to authenticate with a SOCKS version 5 server. This field applies to an `auth` rule type only.

Option	Description
Destination IP	For a `multiple_socks` rule, specify either a single IP address *or* a range of IP addresses of the origin servers with which Websense Content Gateway must use the SOCKS servers specified in the **SOCKS Servers** field below.

For a `no_socks` rule, specify the IP addresses of the origin servers that you want the proxy to access directly (without going through the SOCKS server). You can enter a single IP address, a range of IP addresses, or a list of IP addresses. Separate each entry in the list with a comma. |
| SOCKS Servers | Specify the hostnames or the IP addresses of the SOCKS servers you want to go through to service requests for the origin servers listed in the **Destination IP** field above. Separate each entry with a semicolon.

This field applies to a `multiple_socks` rule type only. |
| Round Robin | Specifies how strictly Websense Content Gateway should follow round robin. You can select **strict**, or **false**.

This field applies to a `multiple_socks` rule type only. |
| Apply | Applies the configuration changes. |
| Close | Exits the configuration file editor.

Click **Apply** before you click **Close**; otherwise, all configuration changes will be lost. |

Subsystems

The Subsystems configuration options are divided into the following categories:

Cache, page 233

Logging, page 235

Cache

Option	Description
	General
Allow Pinning	Enables or disables the cache pinning option, which lets you keep objects in the cache for a specified time. Set cache pinning rules in the *cache.config* file.
Ram Cache Size	Specifies the size of the RAM cache, in bytes.

If you change this option, you must restart Websense Content Gateway. |
| Maximum Object Size | Specifies the maximum size allowed for objects in the cache.

A value of 0 (zero) means that there is no size restriction. |

Option	Description
	Hosting
Cache Hosting	Displays a table listing the rules in the `hosting.config` file that controls which cache partitions are assigned to specific origin servers and/or domains.
Refresh	Updates the table to display the most up-to-date rules in the **hosting.config** file.
Edit File	Opens the configuration file editor for the **hosting.config** file. The configuration file editor page is described below.
	hosting.config Configuration File Editor
rule display box	Lists the **hosting.config** file rules. Select a rule to edit it. The buttons on the left of the box allow you to delete or move the selected rule up or down in the list.
Add	Adds a new rule to the rule display box at the top of the configuration file editor page.
Set	Updates the rule display box at the top of the configuration file editor page.
Primary Destination Type	Specifies the primary destination rule type: Select **domain** if you want to partition the cache according to domain. Select **hostname** if you want to partition the cache according to hostname
Primary Destination Value	Specifies the domain or origin server's hostname whose content you want to store on a particular partition.
Apply	Applies the configuration changes.
Close	Exits the configuration file editor. Click **Apply** before you click **Close**; otherwise, all configuration changes will be lost.

Logging

Option	Description
	General
Logging	Enables or disables event logging so that transactions are recorded into event log files and/or error log files.
	Select **Log Transactions and Errors** to log transactions into your selected event log files and errors in the error log files.
	Select **Log Transactions Only** to log transactions into your selected event log files only. Websense Content Gateway does not log errors in the error log files.
	Select **Log Errors Only** to log errors in the error log files only. Websense Content Gateway does not log transactions into your selected event log files.
	Select **Disabled** to turn off logging.
Log Directory	Specifies the path of the directory in which Websense Content Gateway stores event logs. The path of this directory must be the same on every node in the Websense Content Gateway cluster failover group.
Log Space: Limit	Specifies the maximum amount of space (in megabytes) allocated to the logging directory for the log files.
	Note: Transaction logs can consume a lot of space. You should set this limit high enough to accommodate at least a single day's worth of uncompressed transaction logs. Also, make sure that this limit is smaller than the actual space available on the partition that contains the logging directory.
Log Space: Headroom	Specifies the tolerance for the log space limit. If the **Auto-Delete Rolled Files** option is enabled, autodeletion is triggered when the amount of free space available in the logging directory is less than the headroom.
Log Rolling: Enable/ Disable	Enables or disables log file rolling. To keep log files down to manageable sizes, you can roll them at regular intervals. See *Rolling event log files*, page 158.
Log Rolling: Offset Hour	Specifies the hour when log rolling takes place. You can set a time of the day in the range 0 to 23. For example, if the offset hour is 0 (midnight) and the roll interval is 6, the log files are rolled at 00:00, 06:00, noon, and 18:00.
Log Rolling: Interval	Specifies the amount of time Websense Content Gateway enters data in log files before rolling them to **.old** files. The minimum value is 300 seconds (five minutes). The default value is 86400 seconds (1 day).
Log Rolling: Auto-Delete Rolled Files	Enables autodeletion of rolled log files when available space in the log directory is low. Autodeletion is triggered when the amount of free space available in the log directory is less than the **Log Space Headroom**.

Option	Description
Formats	
Squid Format: Enable/ Disable	Enables or disables the Squid log format.
Squid Format: ASCII/ Binary	Select **ASCII** or **Binary** as the type of log files to be created.
Squid Format: Filename	Specifies the name used for Squid log files. The default filename is **squid.log**.
Squid Format: Header	Specifies the text header you want Squid log files to contain.
Netscape Common Format: Enable/ Disable	Enables or disables the Netscape Common log format.
Netscape Common Format: ASCII/ Binary	Select **ASCII** or **Binary** as the type of log file to be created.
Netscape Common Format: Filename	Specifies the name used for Netscape Common log files. The default filename is **common.log**.
Netscape Common Format: Header	Specifies the text header you want Netscape Common log files to contain.
Netscape Extended Format: Enable/ Disable	Enables or disables the Netscape Extended log format.
Netscape Extended Format: ASCII/ Binary	Select **ASCII** or **Binary** as the type of log file to be created.
Netscape Extended Format: Filename	Specifies the name used for Netscape Extended log files. The default filename is **extended.log**.
Netscape Extended Format: Header	Specifies the text header you want Netscape Extended log files to contain.
Netscape Extended 2 Format: Enable/Disable	Enables or disables the Netscape Extended-2 log format.
Netscape Extended 2 Format: ASCII/Binary	Select **ASCII** or **Binary** as the type of log file to be created.
Netscape Extended 2 Format: Filename	Specifies the name used for Netscape Extended-2 log files. The default filename is **extended2.log**.
Netscape Extended 2 Format: Header	Specifies the text header you want Netscape Extended-2 log files to contain.
Splitting	
Split ICP Logs	When this option is enabled, Websense Content Gateway records ICP transactions in a separate log file. When this option is disabled, Websense Content Gateway records ICP transactions in the same log file with HTTP and FTP entries.

Option	Description
Split NNTP Logs	When this option is enabled, Websense Content Gateway records all commands it sends to the Network News Transfer Protocol (NNTP) server in a separate log file. This allows you to log transactions from incoming news feeds. When this option is disabled, Websense Content Gateway records NNTP transactions in the same log file with HTTP and FTP entries.
Split Host Logs	When this option is enabled, Websense Content Gateway creates a separate log file for each of the hosts listed in the **log_hosts.config** file. When this option is disabled, Websense Content Gateway records transactions for all hosts in the same log file.
	Collation
Collation Mode	Specifies the log collation mode for this Websense Content Gateway node. You can use the log file collation feature to keep all logged information in one place. For more information about log file collation, see *Collating event log files*, page 164.
	Select **Collation Disabled** to disable log collation on this Websense Content Gateway node.
	Select **Be a Collation Server** to configure this Websense Content Gateway node to be the collation server.
	Select **Be a Collation Client** to configure this Websense Content Gateway server to be a collation client. A Websense Content Gateway server configured as a collation client sends only the active standard log files, such as Squid, Netscape Common, and so on, to the collation server. If you select this option, enter the hostname of the collation server for your cluster in the **Log Collation Server** field.
	Note: When logs are collated, the source of the log entry—its node of origin—is lost unless you turn on the **Log collation host tagged** option (described below).
	Log collation consumes cluster bandwidth in sending all log entries to a single node. It can therefore affect the performance of the cluster.
	If you want Websense Content Gateway as a collation client to send custom (XML-based) log files, you must specify a LogObject in the **logs_xml.config** file.
Log Collation Server	Specifies the hostname of the log collation server to which you want to send log files.
Log Collation Port	Specifies the port used for communication between the collation server and client. You must specify a port number in all cases, except when log collation is inactive. The default port number is 8085.
	Note: Do not change the port number unless there is a conflict with another service already using the port.

Option	Description
Log Collation Secret	Specifies the password for the log collation server and the other nodes in the cluster. This password is used to validate logging data and prevent the exchange of arbitrary information.
Log Collation Host Tagged	When this option is enabled, Websense Content Gateway adds the hostname of the node that generated the log entry to end of the entry in the collated log file.
Log Collation Orphan Space	Specifies the maximum amount of space (in megabytes) allocated to the logging directory for storing orphan log files on the Websense Content Gateway node. Websense Content Gateway creates orphan log entries when it cannot contact the log collation server.
	Custom
Custom Logging	Enables or disables custom logging.
Custom Log File Definitions	Displays the *logs_xml.config* file so that you can configure custom (XML-based) logging options.

Networking

The Networking configuration options are divided into the following categories:

Connection Management, page 238

ARM, page 240

WCCP, page 244

DNS Resolver, page 246

ICAP, page 247

Virtual IP, page 247

Connection Management

Option	Description
	Throttling
Throttling Net Connections	Specifies the maximum number of network connections that Websense Content Gateway accepts.
	Setting a Websense Content Gateway throttle limit helps to prevent system overload when traffic bottlenecks develop. When network connections reach this limit, Websense Content Gateway queues new connections until existing connections close.
	Do not set this variable below the minimum value of 100.

Option	Description
	Load Shedding
Maximum Connections	Specifies the maximum number of client connections allowed before the ARM (transparent mode) starts forwarding incoming requests directly to the origin server. The default value is 1 million connections.
	If you change this option, you must restart Websense Content Gateway.
	Congestion Control
Enabled/Disabled	The congestion control feature tracks which hosts are congested so Websense Content Gateway will not forward requests to those hosts.
	When this option is enabled, rather than forwarding requests to a congested destination host, the proxy instructs the requestor to try again at a later time.
	When this option is disabled, requests are forwarded to the destination host regardless of host congestion.
	Congestion is determined by rules you set up in the **congestion.config** file on this screen.
Congestion Rules	Displays the **congestion.config** file so that you can define specific congestion configuration rules. Rules are defined by primary and secondary rule pairs.
	By default, the following Congestion Control Rule tags are defined:
	# max_connection_failures=5 # fail_window=120 # proxy_retry_interval=10 # client_wait_interval=300 # wait_interval_alpha=30 # live_os_conn_timeout=60 # live_os_conn_retries=2 # dead_os_conn_timeout=15 # dead_os_conn_retries=1 # max_connection=-1 # error_page="congestion#retryAfter" # congestion_scheme="per_ip" # snmp=on
	Every line must have a primary destination specifier. Secondary specifiers are optional. Each line can have one or more rule pairs. Following is an example of a line in the congestion.config file:
	dest_host=gumby port=80 fail_window=100
	Press **Submit** when you are done editing the file.
	For a description of the Destination Specifiers, see *Destination specifier tags*, page 339. For a description of the Congestion Control Rule tags, see *Congestion control rule variables*, page 340.

ARM

> ✓ **Note**
>
> The ARM configuration options appear on the Configure pane only if you have enabled ARM in the Features table on the **Configure > My Proxy > Basic > General** tab.

Option	Description
	General
IP spoofing	Enables or disables the IP spoofing option, which configures Websense Content Gateway to establish connections to origin servers with the client IP address instead of the Websense Content Gateway IP address.
Network Address Translation (NAT)	Displays the redirection rules in the *ipnat.conf* file that specify how incoming packets are readdressed when the proxy is serving traffic transparently. Websense Content Gateway creates redirection rules during installation. You can modify these rules.
Refresh	Updates the table to display the most up-to-date rules in the **ipnat.conf** file.
Edit File	Opens the configuration file editor for the **ipnat.conf** file.
	ipnat.conf Configuration File Editor
rule display box	Lists the *ipnat.conf* file rules. Select a rule to edit it. The buttons on the left of the box allow you to delete or move the selected rule up or down in the list.
Add	Adds a new rule to the rule display box at the top of the configuration file editor page.
Set	Updates the rule display box at the top of the configuration file editor page.
Ethernet Interface	Specifies the Ethernet interface that traffic will use to access the Websense Content Gateway machine: for example, eth0 on Linux.
Connection Type	Specifies the connection type that applies for the rule: TCP or UDP.
Source IP	Specifies the IP address from which traffic is sent.
Source CIDR	Specifies the IP address in CIDR (Classless Inter-Domain Routing) format, such as 1.1.1.0/24. Entering a value in this field is optional.
Source Port	Specifies the traffic destination port: for example, 80 for HTTP traffic.
Destination IP	Specifies the IP address of your Websense Content Gateway server.

Option	Description
Destination Port	Specifies the proxy port: for example, 8080 for HTTP traffic.
User Protocol (Optional)	When dns is selected, the ARM redirects DNS traffic to Websense Content Gateway: otherwise, DNS traffic is bypassed.
Apply	Applies the configuration changes.
Close	Exits the configuration file editor. Click **Apply** before you click **Close**; otherwise, all configuration changes will be lost.
	Static Bypass
Static Bypass	Displays a table listing the rules in the *bypass.config* file that specify static transparency bypass rules. When transparency is enabled, the proxy uses these rules to determine whether to bypass incoming client requests or attempt to serve them transparently.
Refresh	Updates the table to display the most up-to-date rules in the **bypass.config** file.
Edit File	Opens the configuration file editor for the **bypass.config** file.
	bypass.config Configuration File Editor
rule display box	Lists the *bypass.config* file rules. Select a rule to edit it. The buttons on the left of the box allow you to delete or move the selected rule up or down in the list.
Add	Adds a new rule to the rule display box at the top of the configuration file editor page.
Set	Updates the rule display box at the top of the configuration file editor page.
Rule Type	Specifies the rule type: A **bypass** rule bypasses specified incoming requests. A **deny_dyn_bypass** rule prevents the proxy from bypassing specified incoming client requests dynamically (a deny bypass rule can prevent Websense Content Gateway from bypassing itself).
Source IP	Specifies the source IP address in incoming requests that the proxy must bypass or deny bypass. The IP address can be one of the following: A simple IP address, such as 123.45.67.8 In CIDR (Classless Inter-Domain Routing) format, such as 1.1.1.0/24. A range separated by a dash, such as 1.1.1.1-2.2.2.2 Any combination of the above, separated by commas, such as 1.1.1.0/24, 25.25.25.25, 123.1.23.1-123.1.23.123

Option	Description
Destination IP	Specifies the destination IP address in incoming requests that the proxy must bypass or deny bypass. The IP address can be one of the following:
	A simple IP address, such as 123.45.67.8
	In CIDR (Classless Inter-Domain Routing) format, such as 1.1.1.0/24
	A range separated by a dash, such as 1.1.1.1-2.2.2.2
	Any combination of the above, separated by commas, such as 1.1.1.0/24, 25.25.25.25, 123.1.23.1-123.1.23.123
Apply	Applies the configuration changes.
Close	Exits the configuration file editor.
	Click **Apply** before you click **Close**; otherwise, all configuration changes will be lost.
	Dynamic Bypass
Dynamic Bypass	Enables or disables the dynamic bypass option to bypass the proxy and go directly to the origin server when clients or servers cause problems. Dynamic bypass rules are deleted when you stop Websense Content Gateway.
Behavior: Non-HTTP, Port 80	Select **Enabled** to enable dynamic bypass when Websense Content Gateway encounters non-HTTP traffic on port 80.
	Select **Disabled** to disable dynamic bypass when Websense Content Gateway encounters non-HTTP traffic on port 80.
	Select **Source-Destination** to enable dynamic source/destination bypass when Websense Content Gateway encounters non-HTTP traffic on port 80.
	Select **Destination Only** to enable dynamic destination bypass when Websense Content Gateway encounters non-HTTP traffic on port 80.
Behavior: HTTP 400	Select **Enabled** to enable dynamic bypass when an origin server returns a 400 error.
	Select **Disabled** to disable dynamic bypass when an origin server returns a 400 error.
	Select **Source-Destination** to enable dynamic source/destination bypass when an origin server returns a 400 error.
	Select **Destination Only** to enable dynamic destination bypass when an origin server returns a 400 error.
Behavior: HTTP 401	Select **Enabled** to enable dynamic bypass when an origin server returns a 401 error.
	Select **Disabled** to disable dynamic bypass when an origin server returns a 401 error.
	Select **Source-Destination** to enable dynamic source/destination bypass when an origin server returns a 401 error.
	Select **Destination Only** to enable dynamic destination bypass when an origin server returns a 401 error.

Option	Description
Behavior: HTTP 403	Select **Enabled** to enable dynamic bypass when an origin server returns a 403 error.
	Select **Disabled** to disable dynamic bypass when an origin server returns a 403 error.
	Select **Source-Destination** to enable dynamic source/destination bypass when an origin server returns a 403 error.
	Select **Destination Only** to enable dynamic destination bypass when an origin server returns a 403 error.
Behavior: HTTP 405	Select **Enabled** to enable dynamic bypass when an origin server returns a 405 error.
	Select **Disabled** to disable dynamic bypass when an origin server returns a 405 error.
	Select **Source-Destination** to enable dynamic source/destination bypass when an origin server returns a 405 error.
	Select **Destination Only** to enable dynamic destination bypass when an origin server returns a 405 error.
Behavior: HTTP 406	Select **Enabled** to enable dynamic bypass when an origin server returns a 406 error.
	Select **Disabled** to disable dynamic bypass when an origin server returns a 406 error.
	Select **Source-Destination** to enable dynamic source/destination bypass when an origin server returns a 406 error.
	Select **Destination Only** to enable dynamic destination bypass when an origin server returns a 406 error.
Behavior: HTTP 408	Select **Enabled** to enable dynamic bypass when an origin server returns a 408 error.
	Select **Disabled** to disable dynamic bypass when an origin server returns a 408 error.
	Select **Source-Destination** to enable dynamic source/destination bypass when an origin server returns a 408 error.
	Select **Destination Only** to enable dynamic destination bypass when an origin server returns a 408 error.
Behavior: HTTP 500	Select **Enabled** to enable dynamic bypass when an origin server returns a 500 error.
	Select **Disabled** to disable dynamic bypass when an origin server returns a 500 error.
	Select **Source-Destination** to enable dynamic source/destination bypass when an origin server returns a 500 error.
	Select **Destination Only** to enable dynamic destination bypass when an origin server returns a 500 error.

WCCP

> **✓ Note**
>
> The WCCP configuration options appear on the Configure pane only if you have enabled WCCP in the Features table on the **Configure > My Proxy > Basic > General** tab.

Option	Description
	General
WCCP Version	Specifies the version of WCCP running on the router: WCCP v1.0 or WCCP v2.0.
	If you change this option, you must restart Websense Content Gateway.
	WCCP v1.0
WCCP Network Interface	Specifies the Ethernet interface Websense Content Gateway uses to talk to the WCCP 1.0-enabled router.
	If you change this option, you must restart Websense Content Gateway.
WCCP Router IP Address	Specifies the IP address of the WCCP-enabled router sending traffic to Websense Content Gateway.
	If you change this option, you must restart Websense Content Gateway.
	WCCP v2.0
WCCP Network Interface	Specifies the Ethernet interface Websense Content Gateway uses to talk to the WCCP 2.0-enabled router.
	If you change this option, you must restart Websense Content Gateway.
Security: Enable/Disable	Enables or disables security so that the router and Websense Content Gateway can authenticate each other. (If you enable security in Websense Content Gateway, you must also enable security on the router. See your Cisco router documentation.)
	If you change this option, you must restart Websense Content Gateway.
Security: Password	Specifies the password used for authentication. This password must be the same password configured on the router and can be a maximum of eight characters long.
	If you change this option, you must restart Websense Content Gateway.
Multicast: Enable/Disable	Enables or disables WCCP multicast mode.
	If you change this option, you must restart Websense Content Gateway.

Option	Description
Multicast: IP Address	Specifies the multicast IP address.
	If you change this option, you must restart Websense Content Gateway.
Routers: Number of WCCP Routers	If multicast is *not* enabled, the routers on your network are not automatically discovered. You must specify the number of routers that direct traffic to Websense Content Gateway. WCCP 2.0 supports a maximum of 32 routers.
	If you change this option, you must restart Websense Content Gateway.
Routers: IP Address of 1st Router	Specifies the IP address of the first WCCP 2.0-enabled router.
	If you change this option, you must restart Websense Content Gateway.
Routers: IP Address of 2nd Router	Specifies the IP address of the second WCCP 2.0-enabled router.
	If you change this option, you must restart Websense Content Gateway.
Routers: IP Address of 3rd Router	Specifies the IP address of the third WCCP 2.0-enabled router.
	If you change this option, you must restart Websense Content Gateway.
Services: HTTP	When enabled, a WCCP 2.0-enabled router can send HTTP traffic to Websense Content Gateway.
	If you change this option, you must restart Websense Content Gateway.
Services: HTTPS	When enabled, a WCCP 2.0-enabled router can send secure HTTP traffic to Websense Content Gateway. If you change this option, you must restart Websense Content Gateway.
Services: NNTP	When enabled, a WCCP 2.0-enabled router can send Network News Transfer Protocol (NNTP) traffic to Websense Content Gateway. If you change this option, you must restart Websense Content Gateway.
Services: FTP	When enabled, a WCCP 2.0-enabled router can send FTP traffic to Websense Content Gateway.
	If you change this option, you must restart Websense Content Gateway.
Services: SOCKS	When enabled, a WCCP 2.0-enabled router can send SOCKS traffic to Websense Content Gateway.
	If you change this option, you must restart Websense Content Gateway.

Option	Description
Services: DNS	When enabled, a WCCP 2.0-enabled router can send DNS traffic to Websense Content Gateway.
	If you change this option, you must restart Websense Content Gateway.
Miscellaneous: Encapsulation	Enables or disables packet encapsulation mode, which enables Websense Content Gateway to send encapsulated returned (bypassed) packets to the router.

DNS Resolver

Option	Description
	Resolver
Local Domain Expansion	Enables or disables local domain expansion so that Websense Content Gateway can attempt to resolve unqualified hostnames by expanding to the local domain. For example, if a client makes a request to an unqualified host named hostx, and if the WCG local domain is **y.com**, Websense Content Gateway expands the hostname to **hostx.y.com**.
	Host Database
DNS Lookup Timeout	Specifies the maximum number of seconds the proxy can wait for a lookup response from the Domain Name Server.
Foreground Timeout	Specifies how long DNS entries can remain in the database before they are flagged as stale.
	For example, if this timeout is 24 hours and a client requests an entry that has been in the database for 24 hours or longer, the proxy refreshes the entry before serving it.
	Caution: Setting the foreground timeout too low might slow response time. Setting it too high risks accumulation of incorrect information.

ICAP

> **✔ Note**
> The ICAP configuration options appear on the Configure pane only if you have enabled ICAP in the Features table on the **Configure > My Proxy > Basic > General** tab.

Option	Description
Enable/disable	Specifies whether to send outgoing content to Websense Data Security Suite (DSS) for analysis. The default is disabled.
ICAP Service URI	Specifies the Uniform Resource Identifier for the ICAP service. The format is: `icap://hostname:port/path`. For example: `icap://ICAP_machine:1344/REQMOD`. The default ICAP port is 1344. If you are using the default port, you need not specify it in the URI.
Action for Communication Errors	Select whether to allow traffic or send a block page if Websense Content Gateway receives an error while communication with Websense Data Security Suite.
Action for Large files	Select whether to allow traffic or send a block page if a file larger than the size limit specified in DSS is sent. The default size limit in DSS is 12 MB.
Analyze Secure Content	Select whether decrypted traffic should be sent to Data Security Suite for analysis or sent directly to the destination.

Virtual IP

> **✔ Note**
> The ICAP configuration options appear on the Configure pane only if you have enabled Virtual IP in the Features table on the **Configure > My Proxy > Basic > General** tab.

Option	Description
Virtual IP Addresses	Displays a table listing the virtual IP addresses managed by Websense Content Gateway.
Refresh	Updates the table to display the most up-to-date list of virtual IP addresses. Click this button after you have added to or modified the list of virtual IP addresses with the configuration file editor.

Option	Description
Edit File	Opens the configuration file editor so that you can edit and add to the list of virtual IP addresses.
	vaddrs.config Configuration File Editor
rule display box	Lists the virtual IP addresses. Select a virtual IP address to edit it. The buttons on the left of the box allow you to delete or move the selected virtual IP address up or down in the list.
Add	Adds a new virtual IP address to the rule display box at the top of the configuration file editor page.
Set	Updates the rule display box at the top of the configuration file editor page.
Virtual IP Address	Specifies the virtual IP address managed by Websense Content Gateway.
Ethernet Interface	Specifies the network interface assigned to the virtual IP address.
Sub-Interface	Specifies the subinterface ID. This is a number between 1 and 255 that the interface uses for the address.
Apply	Applies the configuration changes.
Close	Exits the configuration file editor. Click **Apply** before you click **Close**; otherwise, all configuration changes will be lost.

SSL

The SSL configuration options are divided into the following categories:

- Certificates (see *Managing certificates*, page 129)
- Decryption/Encryption (see *Decryption specifics*, page 131)
- Validation (see *Validating certificates*, page 133)
- Incidents (see *Managing Web HTTPS site access*, page 138)
- Client certificates (see *Client certificates*, page 141)
- Logging (see *Configuring logging*, page 143)
- Customization (see *Customizing messages*, page 145)
- Internal Root CA (see *Internal Root CAs*, page 125)

D | Event Logging Formats

Custom logging fields

.

%<field symbol>	Description
{*HTTP header field name*}cqh	Logs the information in the requested field of the client request HTTP header; for example, `%<{Accept-Language}cqh>` logs the **Accept-Language:** field in client request headers.
{*HTTP header field name*}pqh	Logs the information in the requested field of the proxy request HTTP header; for example, `%<{Authorization}pqh>` logs the **Authorization:** field in proxy request headers.
{*HTTP header field name*}psh	Logs the information in the requested field of the proxy response HTTP header; for example, `%<{Retry-After}psh>` logs the **Retry-After:** field in proxy response headers.
{*HTTP header field name*}ssh	Logs the information in the requested field of the server response HTTP header; for example, `%<{Age}ssh>` logs the **Age:** field in server response headers.
caun	The client authenticated user name; result of the `RFC931/ident` lookup of the client user name.
cfsc	The client finish status code; specifies whether the client request to the proxy was successfully completed (FIN) or interrupted (INTR).
chi	The client host IP; the IP address of the client's host machine.
cqbl	The client request transfer length; the body length in the client's request to Websense Content Gateway in bytes.
cqhl	The client request header length; the header length in the client's request to Websense Content Gateway.
cqhm	The HTTP method in the client request to Websense Content Gateway: GET, POST, and so on (subset of `cqtx`).
cqhv	The client request HTTP version.

%<field symbol>	Description
cqtd	The client request time stamp; specifies the date of the client request in the format *yyyy-mm-dd,* where *yyyy* is the 4-digit year, *mm* is the 2-digit month, and *dd* is the 2-digit day.
cqtn	The client request time stamp; date and time of the client's request (in the Netscape time stamp format).
cqtq	The client request time stamp with millisecond resolution.
cqts	The client request time stamp in Squid format; the time of the client request in seconds since January 1, 1970.
cqtt	The client request time stamp; the time of the client request in the format *hh:mm:ss*, where *hh* is the 2-digit hour in 24-hour format, *mm* is the 2-digit minutes, and *ss* is the 2-digit seconds. For example, 16:01:19.
cqtx	The full HTTP client request text, minus headers. For example: GET http://www.company.com HTTP/1.0
cqu	The client request URI; universal resource identifier (URI) of the request from client to Websense Content Gateway (subset of cqtx).
cquc	The client request canonical URL; differs from cqu in that blanks (and other characters that might not be parsed by log analysis tools) are replaced by escape sequences. The escape sequence is a percentage sign followed by the ASCII code number in hex.
cqup	The client request URL path; specifies the argument portion of the URL (everything after the host). For example, if the URL is **http://www.company.com/images/x.gif**, this field displays **/images/x.gif**.
cqus	The client request URL scheme (HTTP, FTP, etc.).
crc	The cache result code; specifies how the cache responded to the request (HIT, MISS, and so on).
pfsc	The proxy finish status code; specifies whether the Websense Content Gateway request to the origin server was successfully completed (FIN) or interrupted (INTR).
phn	The host name of the Websense Content Gateway server that generated the log entry in collated log files.
phr	The proxy hierarchy route; the route that Websense Content Gateway used to retrieve the object.
pqbl	The proxy request transfer length; the body length in the Websense Content Gateway request to the origin server.
pqhl	The proxy request header length; the header length in the Websense Content Gateway request to the origin server.
pqsi	The proxy request server IP address (0 on cache hits and parent-ip for requests to parent proxies).

%<field symbol>	Description
pqsn	The proxy request server name; the name of the server that fulfilled the request.
pscl	The proxy response transfer length; the length of the Websense Content Gateway response to the client in bytes.
psct	The proxy response content type; content type of the document (for example, img/gif) from server response header.
pshl	The proxy response header length; the header length in the Websense Content Gateway response to the client.
psql	The proxy response transfer length in Squid format (includes header and content length).
pssc	The proxy response status code; the HTTP response status code from Websense Content Gateway to the client.
shi	The IP address resolved from the DNS name lookup of the host in the request. For hosts with multiple IP addresses, this field records the IP address resolved from that particular DNS lookup. This can be misleading for cached documents.
	For example, if the first request was a cache miss and came from IP1 for server S and the second request for server S resolved to IP2 but came from the cache, the log entry for the second request will show IP2.
shn	The host name of the origin server.
sscl	The server response transfer length; response length, in bytes, from origin server to Websense Content Gateway.
sshl	The server response transfer length; the body length in the origin server's response to Websense Content Gateway in bytes.
sshv	The server response HTTP version (1.0, 1.1, and so on).
sssc	The server response status code; the HTTP response status code from origin server to Websense Content Gateway.
ttms	The time Websense Content Gateway spends processing the client request; the number of milliseconds between the time that the client establishes the connection with Websense Content Gateway and the time that Websense Content Gateway sends the last byte of the response back to the client.

%<field symbol>	Description
ttmsf	The time Websense Content Gateway spends processing the client request as a fractional number of seconds; specifies the time in millisecond resolution, but instead of formatting the output as an integer (as with *ttms*), the display is formatted as a floating-point number representing a fractional number of seconds. For example, if the time is 1500 milliseconds, this field displays 1.5 while the **ttms** field displays 1500 and the **tts** field displays 1.
tts	The time Websense Content Gateway spends processing the client request; the number of seconds between the time that the client establishes the connection with the proxy and the time that the proxy sends the last byte of the response back to the client.

Related topic:

Logging format cross-reference, page 252

Logging format cross-reference

The following sections illustrate the correspondence between Websense Content Gateway logging fields and standard logging fields for the Squid and Netscape formats.

Squid logging formats

Squid	Websense Content Gateway Field Symbols
time	cqts
elapsed	ttms
client	chi
action/code	crc/pssc
size	psql
method	cqhm
url	cquc
ident	caun
hierarchy/from	phr/pqsn
content	psct

For example, if you want to create a custom format called `short_sq` based on the first three Squid fields, enter a line in the **logs.config** file as follows:

```
format:enabled:1:short_sq:%<cqts> %<ttms>
%<chi>:short_sq:ASCII:none
```

See *Custom format*, page 153, for more information about defining custom log files.

Netscape Common logging formats

Netscape Common	Websense Content Gateway Field Symbols
host	chi
usr	caun
[time]	[cqtn]
"req"	"cqtx"
s1	pssc
c1	pscl

Netscape Extended logging formats

Netscape Extended	Websense Content Gateway Field Symbols
host	chi
usr	caun
[time]	[cqtn]
"req"	"cqtx"
s1	pssc
c1	pscl
s2	sssc
c2	sscl
b1	cqbl
b2	pqbl
h1	cqhl
h2	pshl
h3	pqhl

Netscape Extended	Websense Content Gateway Field Symbols
h4	sshl
xt	tts

Netscape Extended-2 logging formats

Netscape Extended-2	Websense Content Gateway Field Symbols
host	chi
usr	caun
[time]	[cqtn]
"req"	"cqtx"
s1	pssc
c1	pscl
s2	sssc
c2	sscl
b1	cqbl
b2	pqbl
h1	cqhl
h2	pshl
h3	pqhl
h4	sshl
xt	tts
route	phr
pfs	cfsc
ss	pfsc
crc	crc

E | Configuration Files

Websense Content Gateway contains the following configuration files that you can edit to customize the proxy.

Specifying URL regular expressions (url_regex)

Entries of type `url_regex` within the configuration files use regular expressions to perform a match.

The following table offers examples to illustrate how to create a valid `url_regex`.

Value	Description
x	Matches the character x.
.	Match any character.
^	Specifies beginning of line.
$	Specifies end of line.
[xyz]	A *character class*. In this case, the pattern matches either x, y, or z.
[abj-oZ]	A *character class* with a range. This pattern matches a, b, any letter from j through o, or Z.
[^A-Z]	A *negated character class*. For example, this pattern matches any character except those in the class.
r*	Zero or more r's, where r is any regular expression.
r+	One or more r's, where r is any regular expression.
r?	Zero or one r, where r is any regular expression.
r{2,5}	From two to five r's, where r is any regular expression.
r{2,}	Two or more r's, where r is any regular expression.
r{4}	Exactly 4 r's, where r is any regular expression.
"[xyz]\"images"	The literal string [xyz]"images"
\X	If X is a, b, f, n, r, t, or v, then the ANSI-C interpretation of \x; Otherwise, a literal X. This is used to escape operators such as *.
\0	A NULL character.
\123	The character with octal value 123.
\x2a	The character with hexadecimal value 2a.
(r)	Matches an r; where r is any regular expression. You can use parentheses to override precedence.
rs	The regular expression r, followed by the regular expression s.
r|s	Either an r or an s.
#<n>#	Inserts an *end* node causing regular expression matching to stop when reached. The value n is returned.

Examples

You can specify `dest_domain=mydomain.com` to match any host in *mydomain.com*. Likewise, you can specify `dest_domain=.` to match any request.

arm_security.config

The **arm_security.config** file defines which hosts are allowed to communicate with the Websense Content Gateway machine using TCP and UDP through defined ports. Websense Content Gateway uses this configuration file when the ARM security option is enabled. For information about enabling the ARM security option, see *Controlling host access to the proxy server*, page 102.

You can specify the following information:

◆ The ports that are open by default, for either TCP or UDP

◆ The hosts that are denied access to specific destination ports, for either TCP or UDP

◆ The hosts that are allowed access to specific destination ports, for either TCP or UDP

> **Warning**
> Before you enable the ARM security option, ensure that you have either console access to the Websense Content Gateway machine or that you have added the appropriate rules to the configuration file to allow telnet or ssh access for yourself.

The ports you specify in the access control list remain closed even when Websense Content Gateway is not running.

> **Important**
> After you modify this file, you must restart the proxy.

Format

Each line in the **arm_security.config** file uses one of the following formats:

```
open tcp | udp ports o_ports
deny tcp | udp dport d_ports src src_ipaddress
```

```
allow tcp | udp src src_ipaddress dst dst_ipaddress
dport d_ports sport s_ports
```

Field	Allowed Inputs
o_ports	The port or series of ports that are open by default.
d_ports	The destination port, or range of destination ports, through which TCP traffic is either allowed or denied.
s_ports	The source port, or range of source ports, from which TCP traffic is allowed.
src_ipaddress	The IP address, or range of IP addresses, specifying the source of the communication.
dst_ipaddress	The IP address, or range of IP addresses, specifying the destination of the communication.

> **Note**
>
> The **arm_security.config** file does not support spaces before or after the hyphen in an IP address range. For example, you must specify `12.34.56.1-12.34.56.7`.

Examples

The following example defines ports 80, 119, 23, and 554 as open for TCP communication. All other ports are closed.

```
open tcp ports 80 119 23 554
```

In the following example, the first line specifies that all hosts are denied access to destination port 80 using TCP. The second line specifies that host 209.1.2.2 is denied access to destination port 90 using UDP.

```
deny tcp dport 80 src 0.0.0.0-255.255.255.255
deny udp dport 90 src 209.1.2.2
```

In the following example, the first line specifies that host 111.11.11.1 using source port 20 is allowed to communicate with host 123.45.67.8 on destination ports 127-130 using TCP. The second line specifies that all hosts are allowed to communicate with host 123.12.3.4 using UDP.

```
allow tcp src 111.11.11.1 dst 123.45.67.8 dport 127-130
sport 20
allow udp dst 123.12.3.4
```

bypass.config

The **bypass.config** file contains *static* bypass rules that Websense Content Gateway uses in transparent proxy caching mode. Static bypass rules instruct the proxy to bypass certain incoming client requests so that they are served by the origin server.

The **bypass.config** file also accepts *dynamic* deny bypass rules. See *Dynamic deny bypass rules*, page 260.

You can configure three types of static bypass rules:

◆ *Source bypass* rules configure the proxy to bypass a particular source IP address or range of IP addresses. For example, you can bypass clients that do not want to use caching.

◆ *Destination bypass* rules configure the proxy to bypass a particular destination IP address or range of IP addresses. For example, you can bypass origin servers that use IP authentication based on the client's real IP address.

> **Important**
>
> Destination bypass rules prevent the proxy from caching an entire site. You will experience hit rate impacts if the site you bypass is popular.

◆ *Source/destination pair* bypass rules configure the proxy to bypass requests that originate from the specified source to the specified destination. For example, you can route around specific client-server pairs that experience broken IP authentication or out-of-band HTTP traffic problems when cached. Source/destination bypass rules can be preferable to destination rules because they block a destination server only for users that experience problems.

> **Important**
>
> After you modify this file, you must restart the proxy.

Format

Bypass rules have the following format:

```
bypass src ipaddress | dst ipaddress | src ipaddress AND dst
ipaddress
```

Option	Description
src *ipaddress*	Specifies the source (client) IP address in incoming requests that the proxy must bypass. *ipaddress* can be one of the following: A simple IP address, such as `123.45.67.8` • In CIDR (Classless Inter-Domain Routing) format, such as `1.1.1.0/24` • A range separated by a dash, such as `1.1.1.1-2.2.2.2` • Any combination of the above, separated by commas, such as `1.1.1.0/24, 25.25.25.25, 123.1.23.1-123.1.23.123`
dst *ipaddress*	Specifies the destination (origin server) IP address in incoming requests that the proxy must bypass. *ipaddress* can be one of the following: A simple IP address, such as `123.45.67.8` • In CIDR (Classless Inter-Domain Routing) format, such as `1.1.1.0/24` • A range separated by a dash, such as `1.1.1.1-2.2.2.2` • Any combination of the above, separated by commas, such as `1.1.1.0/24, 25.25.25.25, 123.1.23.1-123.1.23.123`
src *ipaddress* AND dst *ipaddress*	Specifies the source and destination IP address pair that the proxy must bypass. *ipaddress* must be a single IP address, such as `123.45.67.8`

Dynamic deny bypass rules

In addition to static bypass rules, the **bypass.config** file also accepts *dynamic deny* bypass rules.

Deny bypass rules prevent the proxy from bypassing certain incoming client requests dynamically (a deny bypass rule can prevent the proxy from bypassing itself). Dynamic deny bypass rules can be source, destination, or source/destination and have the following format:

```
deny_dyn_bypass src ipaddress | dst ipaddress | src
ipaddress AND dst ipaddress
```

For a description of the options, see the table in *Format*, page 260.

✔ **Note**
For the dynamic deny bypass rules to work, you must enable the **Dynamic Bypass** option in Websense Content Manager or set the variable *proxy.config.arm.bypass_dynamic_enabled* to 1 in the **records.config** file.

Important
Static bypass rules overwrite dynamic deny bypass rules. Therefore, if a static bypass rule and a dynamic bypass rule contain the same IP address, the dynamic deny bypass rule is ignored.

Examples

The following example shows source, destination, and source/destination *bypass* rules:

```
bypass src 1.1.1.0/24, 25.25.25.25, 128.252.11.11-
128.252.11.255
bypass dst 24.24.24.0/24
bypass src 25.25.25.25 AND dst 24.24.24.0
```

The following example shows source, destination, and source/destination *dynamic deny bypass* rules:

```
deny_dyn_bypass src 128.252.11.11-128.252.11.255
deny_dyn_bypass dst 111.111.11.1
deny_dyn_bypass src 111.11.11.1 AND dst 111.11.1.1
```

cache.config

The **cache.config** file defines how the proxy caches Web objects. You can add caching rules to specify the following configuration:

- Not to cache objects from specific IP addresses
- How long to pin particular objects in the cache
- How long to consider cached objects as fresh

◆ Whether to ignore no-cache directives from the server

> **Important**
>
> After you modify this file, run `content_line -x` from the Websense Content Gateway **bin** directory (default location is **/opt/WCG/bin**) to apply the changes. When you apply the changes to a node in a cluster, Websense Content Gateway applies the changes to all nodes in the cluster.

Format

Each line in the **cache.config** file contains a caching rule. Websense Content Gateway recognizes three space-delimited tags:

```
primary_destination=value secondary_specifier=value
action=value
```

The following table lists the possible primary destinations and their allowed values.

Primary Destination	Allowed Value
dest_domain	A requested domain name
dest_host	A requested hostname
dest_ip	A requested IP address
url_regex	A regular expression to be found in a URL

Secondary specifiers are optional in the **cache.config** file. The following table lists the possible secondary specifiers and their allowed values.

> **Note**
>
> You can use more than one secondary specifier in a rule. However, you cannot repeat a secondary specifier.

Secondary Specifier	Allowed Value
port	A requested URL port
scheme	A request URL protocol; one of the following: • HTTP • FTP
prefix	A prefix in the path part of a URL
suffix	A file suffix in the URL

Secondary Specifier	Allowed Value
method	A request URL method; one of the following: • `get` • `put` • `trace`
time	A time range, such as 08:00-14:00
src_ip	A client IP address.

The following table lists the possible actions and their allowed values.

Action	Value
action	One of the following values: • `never-cache` configures the proxy to never cache specified objects. • `ignore-no-cache` configures the proxy to ignore all `Cache-Control: no-cache` headers. • `ignore-client-no-cache` configures the proxy to ignore `Cache-Control: no-cache` headers from client requests. • `ignore-server-no-cache` configures the proxy to ignore `Cache-Control: no-cache` headers from origin server responses.
pin-in-cache	The amount of time you want to keep the object(s) in the cache. The following time formats are allowed: • *d* for days (for example 2d) • *h* for hours (for example, 10h) • *m* for minutes (for example, 5m) • *s* for seconds (for example, 20s) • mixed units (for example, 1h15m20s)
revalidate	The amount of time you want to consider the object(s) fresh. Use the same time formats as `pin-in-cache`.
ttl-in-cache	The amount of time you want to keep objects in the cache regardless of Cache-Control response headers. Use the same time formats as `pin-in-cache` and `revalidate`.

Examples

The following example configures the proxy to never cache FTP documents requested from the IP address 112.12.12.12:

```
dest_ip=112.12.12.12 scheme=ftp action=never-cache
```

The following example configures the proxy to keep documents with URLs that contain the regular expression `politics` and the path **prefix/viewpoint** in the cache for 12 hours:

```
url_regex=politics prefix=/viewpoint pin-in-cache=12h
```

The following example configures the proxy to revalidate `gif` and `jpeg` objects in the domain `mydomain.com` every 6 hours and all other objects in `mydomain.com` every hour:

```
dest_domain=mydomain.com suffix=gif revalidate=6h
dest_domain=mydomain.com suffix=jpeg revalidate=6h
dest_domain=mydomain.com revalidate=1h
```

> ✔ **Note**
> The rules are applied in the order listed.

filter.config

The **filter.config** file lets you deny or allow particular URL requests, keep or strip header information from client requests, and specify LDAP and NTLM authentication rules.

> **Important**
> After you modify the file, run `content_line -x` from the Websense Content Gateway **bin** directory (default location is **/opt/WCG/bin**) to apply the changes. When you apply the changes to a node in a cluster, Websense Content Gateway applies the changes to all nodes in the cluster.

Format

Each line in the **filter.config** file contains a filtering rule. Websense Content Gateway applies the rules in the order listed, starting at the top of the file.

Websense Content Gateway recognizes three space-delimited tags:

```
primary_destination=value secondary_specifier=value
action=value
```

The following table lists the possible primary destinations and their allowed values.

Primary Destination	Allowed Value
dest_domain	A requested domain name
dest_host	A requested hostname
dest_ip	A requested IP address
url_regex	A regular expression to be found in a URL

Secondary specifiers are optional in the **filter.config** file. The following table lists the possible secondary specifiers and their allowed values.

✔ **Note**
You can use more than one secondary specifier in a rule. However, you cannot repeat a secondary specifier.

Secondary Specifier	Allowed Value
port	A requested URL port
scheme	A request URL protocol. You can specify one of the following: • HTTP • FTP
prefix	A prefix in the path part of a URL
suffix	A file suffix in the URL
method	A request URL method; one of the following: • get • post • put • trace
time	A time range, such as 08:00-14:00
src_ip	A client IP address.

The following table lists the possible actions and their allowed values.

Action	Allowed Value

action	You can specify the following actions:
	
	• `deny` - to deny requests for HTTP or FTP objects from specific destinations. When a request is denied, the client receives an access denied message.
	
	• `allow` - to allow particular URL requests; the proxy caches and serves the requested content.
	
	• `radius` - to specify authentication rules that determine which users must be authenticated to access particular sites on the Internet and which RADIUS servers are used. You can specify the following optional parameters:
	
	`realm=realm_name`
	
	• `ntlm` - to specify authentication rules that determine which users must be authenticated to access particular sites on the Internet and which domain controllers are used. You can specify the following optional parameters:
	
	`realm=realm_name`
	
	• `ldap` - to specify authentication rules that determine which users must be authenticated to access particular sites on the Internet and which LDAP servers are used. You can specify the following optional parameters:
	
	`realm=realm_name`
	`server=LDAP_server_name:LDAP_server_port`
	`dn=BaseDN uid_filter=UID_filter`
	`attr=additional_LDAP_attribute_name`
	`attr_val=additional_LDAP_attribute_value`
	
	Note: If you specify the `ldap` action alone without any optional parameters, Websense Content Gateway uses the LDAP server and port listed in the **records.config** file. If you use any of the `ldap` optional parameters, you must specify values for the *server* and *dn* parameters.
	
	Note: Websense Content Gateway uses the LDAP rules only if the LDAP option is enabled. If the LDAP option is disabled, Websense Content Gateway treats all authentication rules as `allow` rules - access is granted without authentication. This note applies to RADIUS and NTLM also.

keep_hdr	The client request header information that you want to keep. You can specify the following options: • date • host • cookie • client_ip
strip_hdr	The client request header information that you want to strip. You can specify the same options as `keep_hdr`.

Examples

The following example configures Websense Content Gateway to deny all FTP document requests to the IP address 112.12.12.12:

```
dest_ip=112.12.12.12 scheme=ftp action=deny
```

The following example configures Websense Content Gateway to keep the client IP address header for URL requests that contain the regular expression `politics` and whose path prefix is
/viewpoint:

```
url_regex=politics prefix=/viewpoint keep_hdr=client_ip
```

The following example configures Websense Content Gateway to strip all cookies from client requests destined for the origin server `www.server1.com`:

```
dest_host=www.server1.com strip_hdr=cookie
```

The following example configures Websense Content Gateway to disallow `puts` to the origin server `www.server2.com`:

```
dest_host=www.server2.com method=put action=deny
```

The following example configures Websense Content Gateway to allow only the host associated with the IP address 11.11.1.1 to deliver content directly into the cache (push). A deny rule is also included to prevent unauthorized users from pushing content into the cache.

```
dest_domain=. src_ip=11.11.1.1 method=PUSH action=allow
dest_domain=. method=PUSH action=deny
```

The following example configures Websense Content Gateway to allow access to the origin server `www.server.com` only to users authenticated by the LDAP server running on `ldap.com`:

```
dest_host=www.server.com action=ldap server=ldap.com
dn="o=ldap.com"
```

The following example requires all users to be authenticated by the default LDAP server (the LDAP server listed in the `records.config` file):

```
dest_ip=0.0.0.0-255.255.255.255 action=ldap
```

Websense Content Gateway applies the rules in the order listed in the file. For example, the following sample **filter.config** file configures Websense Content Gateway to do the following:

◆ Allow only users authenticated by the LDAP server on ldap.com to access internal.com

◆ Allow all users (except those trying to access internal.com) to access server1.com

◆ Deny all users access to notthatsite.com

◆ Allow authenticated users (authenticated by the LDAP server on ldap.com) with the attribute `ou="Accounting Department"` in their LDAP profile to access `retirement.com`

◆ Require any user not included in any of the above directives to be authenticated by the default LDAP server

```
dest_host=internal.com action=ldap server=ldap.com
dn="o=ldap.com"
dest_host=server1.com action=allow
dest_host=notthatsite.com action=deny
dest_host=401k.retirement.com action=ldap
server=ldap.com:389
dn="o=ldap.com" attr=ou attr_val="Accounting Department"
dest_ip=0.0.0.0-255.255.255.255 action=ldap
```

icp.config

The **icp.config** file defines ICP peers (parent and sibling caches).

> **Important**
>
> After you modify this file, run `content_line -x` from the Websense Content Gateway **bin** directory (default location is **/opt/WCG/bin**) to apply the changes. When you apply the changes to a node in a cluster, Websense Content Gateway applies the changes to all nodes in the cluster.

Format

Each line in the **icp.config** file contains the name and configuration information for a single ICP peer in the following format:

```
host:host_IP:peer_type:proxy_port:icp_port:MC_on:MC_IP:
MC_TTL:
```

Field	Description
host	The hostname of the ICP peer. This field is optional; if you do not specify the hostname of the ICP peer, you must specify the IP address.
host_IP	The IP address of the ICP peer. This field is optional; if you do not specify the IP address of the ICP peer, you must specify the hostname.
ctype	Use the following options: • 1 to indicate an ICP parent cache • 2 to indicate an ICP sibling cache
proxy_port	The port number of the TCP port used by the ICP peer for proxy communication.
icp_port	The port number of the UDP port used by the ICP peer for ICP communication.

Field	Description
`multicast_member`	Multicast on/off. • 0 if multicast is disabled • 1 if multicast is enabled
`multicast_ip_str`	The multicast IP address.
`multicast_ttl`	The multicast time to live. Use the following options: • 1 if IP multicast datagrams will not be forwarded beyond a single subnetwork • 2 to allow delivery of IP multicast datagrams to more than one subnet (if there are one or more multicast routers attached to the first hop subnet)

Examples

The following example configuration is for three nodes: the local host, one parent, and one sibling.

```
localhost:0.0.0.0:3:8080:3130:0:0.0.0.0:0:
host1:123.12.1.23:1:8080:3131:0:0.0.0.0:0:
host2:123.12.1.24:2:8080:3131:0:0.0.0.0:0:
```

ip_allow.config

The **ip_allow.config** file controls client access to the proxy cache. You can specify ranges of IP addresses that are allowed to use the Websense Content Gateway server as a Web proxy cache.

> **Important**
>
> After you modify the file, run `content_line -x` from the Websense Content Gateway **bin** directory (default location is **/opt/WCG/bin**) to apply the changes. When you apply the changes to a node in a cluster, Websense Content Gateway applies the changes to all nodes in the cluster.

Format

Each line in the **ip_allow.config** file must have the following format:

```
src_ip=ipaddress action=ip_allow | ip_deny
```

where `ipaddress` is the IP address or range of IP addresses of the clients allowed to access the proxy cache.

The action `ip_allow` allows the specified clients to access the proxy cache.

The action `ip_deny` denies the specified clients to access the proxy cache.

By default, the **ip_allow.config** file contains the following line, which allows all clients to access the proxy cache. Comment out or delete this line before adding rules to restrict access.

```
src_ip=0.0.0.0-255.255.255.255 action=ip_allow
```

Examples

The following example allows all clients to access the proxy cache:

```
src_ip=0.0.0.0-255.255.255.255 action=ip_allow
```

The following example allows all clients on a specific subnet to access the proxy cache:

```
src_ip=123.12.3.000-123.12.3.123 action=ip_allow
```

The following example denies all clients on a specific subnet to access the proxy cache:

```
src_ip=123.45.6.0-123.45.6.123 action=ip_deny
```

ipnat.conf

The **ipnat.conf** file contains redirection rules that specify how incoming packets are readdressed when the proxy is serving traffic transparently. Websense Content Gateway creates the redirection rules during installation. You can modify these rules.

> ❗ **Important**
> After you modify this file, you must restart the proxy.

Format

Each line in the **ipnat.conf** file must have the following format:

```
rdr interface 0.0.0.0/0 port dest -> ipaddress port proxy
tcp|udp
```

where:

interface is the Ethernet interface that traffic will use to access the Websense Content Gateway machine (for example, `eth0` on Linux).

dest is the traffic destination port (for example, `80` for HTTP traffic).

ipaddress is the IP address of your Websense Content Gateway server.

proxy is the Websense Content Gateway proxy port (usually `8080` for HTTP traffic).

Examples

The following example configures the ARM to readdress all incoming HTTP traffic to the Websense Content Gateway IP address (111.111.11.1) on the Websense Content Gateway proxy port 8080:

```
rdr hme0 0.0.0.0/0 port 80 -> 111.111.11.1 port 8080 tcp
```

log_hosts.config

To record HTTP/FTP transactions for different origin servers in separate log files, you must list each origin server's hostname in the **log_hosts.config** file. In addition, you must enable the HTTP host splitting option (see *HTTP host log splitting*, page 162).

✔ **Note**

It is recommended that you use the same **log_hosts.config** file on every Websense Content Gateway node in your cluster.

🔵 **Important**

After you modify this file, run content_line -x from the Websense Content Gateway **bin** directory (default location is **/opt/WCG/bin**) to apply the changes. When you apply the changes to a node in a cluster, Websense Content Gateway applies the changes to all nodes in the cluster.

Format

Each line in the **log_hosts.config** file has the following format:

hostname

where *hostname* is the hostname of the origin server.

✔ **Note**

You can specify keywords in the **log_hosts.config** file to record all transactions from origin servers with the specified keyword in their names in a separate log file. See the example below.

Examples

The following example configures Websense Content Gateway to create separate log files containing all HTTP/ FTP transactions for the origin servers webserver1, webserver2, and webserver3.

```
webserver1
webserver2
webserver3
```

The following example records all HTTP and FTP transactions from origin servers that contain sports in their names (for example, sports.yahoo.com and www.foxsports.com) in a log file called **squid-sport.log** (the Squid format is enabled):

```
sports
```

logs_xml.config

The **logs_xml.config** file defines the custom log file formats, filters, and processing options. The format of this file is modeled after XML, the Extensible Markup Language.

Format

The **logs_xml.config** file contains the following specifications:

◆ LogFormat specifies the fields to be gathered from each protocol event access. See *LogFormat*, page 273.

◆ LogFilter specifies the filters that are used to include or exclude certain entries being logged based on the value of a field within that entry. See *LogFilter*, page 274.

◆ LogObject specifies an object that contains a particular format, a local filename, filters, and collation servers. See *LogObject*, page 275.

> ✔ **Note**
> The **logs_xml.config** file ignores extra white space, blank lines, and all comments.

LogFormat

The following table lists the LogFormat specifications.

Field	Allowed Inputs
`<Name = "valid_format_name"/>`	Required. Valid format names include any name except squid, common, extended, or extended2, which are pre-defined formats. There is no default for this tag.
`<Format = "valid_format_specification"/>`	Required. A valid format specification is a `printf`-style string describing each log entry when formatted for ASCII output. Use `'%<field>'` as placeholders for valid field names. For more information, see *Custom logging fields*, page 249. The specified field can be of two types:
	Simple: for example, `%<cqu>`
	A field within a container, such as an HTTP header or a Websense Content Gateway statistic. Fields of this type have the syntax: `'%<{field}container>'`.
`<Interval = "aggregate_interval_secs"/>`	Use this tag when the format contains aggregate operators. The value `"aggregate_interval_secs"` represents the number of seconds between individual aggregate values being produced. The valid set of aggregate operators are: • COUNT • SUM • AVG • FIRST • LAST

LogFilter

The following table lists the `LogFilter` specifications.

Field	Allowed Inputs
`<Name = "valid_filter_name"/>`	Required. All filters must be uniquely named.
`<Condition = "valid_log_field valid_operator valid_comparison_value"/>`	Required. This field contains the following elements: `valid_log_field` - the field that will be compared against the given value. For more information, see *Logging format cross-reference*, page 252. `valid_operator_field` - any one of the following: MATCH, CASE_INSENSITIVE_MATCH, CONTAIN, CASE_INSENSITIVE_CONTAIN. MATCH is true if the field and value are identical (case sensitive). CASE_INSENSITIVE_MATCH is similar to MATCH, only case insensitive. CONTAIN is true if the field contains the value (the value is a substring of the field). CASE_INSENSITIVE_CONTAIN is a case-insensitive version of CONTAIN. `valid_comparison_value` - any string or integer matching the field type. For integer values, all of the operators are equivalent and mean that the field must be equal to the specified value. Note: There are no negative comparison operators. If you want to specify a negative condition, use the **Action** field to REJECT the record.
`<Action = "valid_action_field"/>`	Required. ACCEPT or REJECT. This instructs Websense Content Gateway to either accept or reject records satisfying the condition of the filter.

LogObject

The following table lists the `LogObject` specifications.

Field	Allowed Inputs
`<Format = "valid_format_name"/>`	Required. Valid format names include the predefined logging formats: squid, common, extended, and extended2, as well as any previously-defined custom log formats. There is no default for this tag.
`<Filename = "file_name"/>`	Required. The filename to which the given log file is written on the local file system or on a remote collation server. No local log file will be created if you fail to specify this tag. All filenames are relative to the default logging directory.
	If the name does not contain an extension (for example, `squid`), the extension `.log` is automatically appended to it for ASCII logs and `.blog` for binary logs. (See `<Mode = "valid_logging_mode"/>` below.) If you do not want an extension to be added, end the filename with a single dot (`.`): for example, `squid.`

Field	Allowed Inputs
`<Mode = "valid_logging_mode"/>`	Valid logging modes include `ascii`, `binary`, and `ascii_pipe`. The default is `ascii`.
	Use `ascii` to create event log files in human-readable form (plain ASCII).
	Use `binary` to create event log files in binary format. Binary log files generate lower system overhead and occupy less space on the disk (depending on the information being logged). You must use the `logcat` utility to translate binary log files to ASCII format before you can read them.
	Use `ascii_pipe` to write log entries to a UNIX named pipe (a buffer in memory). Other processes can then read the data using standard I/O functions. Websense Content Gateway does not have to write to disk, freeing disk space and bandwidth for other tasks. In addition, writing to a pipe does not stop when logging space is exhausted because the pipe does not use disk space.
	Note: If you are using a collation server, the log is written to a pipe on the collation server. A local pipe is created even before a transaction is processed so that you can see the pipe right after Websense Content Gateway starts. However, pipes on a collation server *are* created when Websense Content Gateway starts.
`<Filters = "list_of_valid_filter_names"/>`	A comma-separated list of names of any previously defined log filters. If more than one filter is specified, all filters must accept a record for the record to be logged.
`<Protocols = "list_of_valid_protocols"/>`	A comma-separated list of the protocols this object should log. Valid protocol names include `HTTP` and `ICP`.
`<ServerHosts = "list_of_valid_servers"/>`	A comma-separated list of valid hostnames. This tag indicates that only entries from the named servers will be included in the file.
`<CollationHosts = "list_of_valid_hostnames"/>`	A comma-separated list of collation servers to which all log entries (for this object) are forwarded. Collation servers can be specified by name or IP address. Specify the collation port with a colon after the name (for example, `host:port`).

Field	Allowed Inputs
`<Header = "header"/>`	The header text you want the log files to contain. The header text appears at the beginning of the log file, just before the first record.
`<RollingEnabled = "truth value"/>`	Enables or disables log file rolling for the *LogObject*. This setting overrides the value for the configuration setting **Log Rolling: Enabled/Disabled** in Websense Content Manager or *proxy.config.log2.rolling_enabled* in the **records.config** file. Set "truth value" to `1` or `true` to enable rolling; set it to `0` or `false` to disable rolling for this particular *LogObject*.
`<RollingIntervalSec = "seconds"/>`	Specifies the seconds between log file rolling for the *LogObject*. This setting overrides the value for the configuration setting **Log Rolling: Interval** in Websense Content Manager or *proxy.config.log2.rolling_interval_sec* in the **records.config** file. This option allows you to specify different rolling intervals for different *LogObjects*.
`<RollingOffsetHr = "hour"/>`	Specifies an hour (from 0 to 23) at which rolling is guaranteed to "align". Rolling may start before then, but a rolled file will be produced only at that time. The impact of this setting is only noticeable if the rolling interval is larger than one hour. This setting overrides the configuration setting **Log Rolling: Offset Hour** in Websense Content Manager or *proxy.config.log2.rolling_offset_hr* in the **records.config** file.

Examples

The following is an example of a `LogFormat` specification collecting information using three common fields:

```
<LogFormat>
<Name = "minimal"/>
<Format = "%<chi> : %<cqu> : %<pssc>"/>
</LogFormat>
```

The following is an example of a `LogFormat` specification using aggregate operators:

```
<LogFormat>
<Name = "summary"/>
<Format = "%<LAST(cqts)> : %<COUNT(*)> : %<SUM(psql)>"/>
<Interval = "10"/>
</LogFormat>
```

The following is an example of a `LogFilter` that will cause only `REFRESH_HIT` entries to be logged:

```
<LogFilter>
<Name = "only_refresh_hits"/>
<Action = "ACCEPT"/>
<Condition = "%<pssc> MATCH REFRESH_HIT"/>
</LogFilter>
```

> ✔ **Note**
>
> When specifying the field in the filter condition, you can omit the `%<>`. This means that the following filter is equivalent to the example directly above:
>
> ```
> <LogFilter>
>     <Name = "only_refresh_hits"/>
>     <Action = "ACCEPT"/>
>     <Condition = "pssc MATCH REFRESH_HIT"/>
> </LogFilter>
> ```

The following is an example of a `LogObject` specification that creates a local log file for the minimal format defined earlier. The log filename will be **minimal.log** because this is an ASCII log file (the default).

```
<LogObject>
<Format = "minimal"/>
<Filename = "minimal"/>
</LogObject>
```

The following is an example of a `LogObject` specification that includes only HTTP requests served by hosts in the domain `company.com` or by the specific server `server.somewhere.com`. Log entries are sent to port 4000 of the collation host `logs.company.com` and to port 5000 of the collation host `209.131.52.129`.

```
<LogObject>
<Format = "minimal"/>
<Filename = "minimal"/>
<ServerHosts = "company.com,server.somewhere.com"/>
<Protocols = "http"/>
<CollationHosts =
"logs.company.com:4000,209.131.52.129:5000"/>
</LogObject>
```

WELF (WebTrends Enhanced Log Format)

Websense Content Gateway supports WELF, the WebTrends Enhanced Log Format, so that you can analyze Websense Content Gateway log files with WebTrends reporting tools. A predefined `<LogFormat>` that is compatible with WELF is provided at the end of the `logs.config` file (shown below). To create a WELF format log file, create a `<LogObject>` that uses this predefined format.

```
<LogFormat>
<Name = "welf"/>
<Format = "id=firewall time=\"%<cqtd> %<cqtt>\" fw=%<phn>
pri=6 proto=%<cqus> duration=%<ttmsf> sent=%<psql>
rcvd=%<cqhl> src=%<chi> dst=%<shi> dstname=%<shn>
user=%<caun> op=%<cqhm> arg=\"%<cqup>\" result=%<pssc>
ref=\"%<{Referer}cqh>\" agent=\"%<{user-agent}cqh>\"
cache=%<crc>"/>
</LogFormat>
```

mgmt_allow.config

The **mgmt_allow.config** file specifies the IP addresses of remote hosts allowed access or denied access to Websense Content Manager.

> **! Important**
> After you modify this file, run `content_line -x` from the Websense Content Gateway **bin** directory (default location is **/opt/WCG/bin**) to apply the changes. When you apply the changes to a node in a cluster, Websense Content Gateway applies the changes to all nodes in the cluster.

Format

Each line in the **mgmt_allow.config** file has the following format:

```
src_ip=ipaddress action=ip_allow|ip_deny
```

where *ipaddress* is the IP address or range of IP addresses allowed to access Websense Content Manager.

`action` must specify either `ip_allow` to allow access to Websense Content Manager or `ip_deny` to deny access.

By default, the **mgmt_allow.config** file contains the following line, which allows all remote hosts to access Websense Content Manager. Comment out or delete this line before adding rules to restrict access.

```
src_ip=0.0.0.0-255.255.255.255 action=ip_allow
```

Examples

The following example configures Websense Content Gateway to allow only one user to access Websense Content Manager:

```
src_ip=123.12.3.123 action=ip_allow
```

The following example configures Websense Content Gateway to allow a range of IP addresses to access Websense Content Manager:

```
src_ip=123.12.3.000-123.12.3.123 action=ip_allow
```

The following example configures Websense Content Gateway to deny the IP address 123.45.67.8 access to the Websense Content Manager:

```
src_ip=123.45.67.8 action=ip_deny
```

parent.config

The **parent.config** file identifies the HTTP parent proxies used in an HTTP cache hierarchy. Use this file to perform the following configuration:

◆ Set up parent cache hierarchies, with multiple parents and parent failover
◆ Configure selected URL requests to bypass parent proxies

Websense Content Gateway uses the **parent.config** file only when the HTTP parent caching option is enabled. See *Configuring Websense Content Gateway to use an HTTP parent cache*, page 70.

> **Important**
>
> After you modify this file, run `content_line -x` from the Websense Content Gateway **bin** directory (default location is **/opt/WCG/bin**) to apply the changes. When you apply the changes to a node in a cluster, Websense Content Gateway applies the changes to all nodes in the cluster.

Format

Each line in the **parent.config** file must contain a parent caching rule. Websense Content Gateway recognizes three space-delimited tags:

```
primary_destination=value secondary_specifier=value
action=value
```

The following table lists the possible primary destinations and their allowed values.

Primary Destination	Allowed Value
dest_domain	A requested domain name
dest_host	A requested hostname
dest_ip	A requested IP address or range of IP addresses separated by a dash (-).
url_regex	A regular expression to be found in a URL

Secondary specifiers are optional in the `parent.config` file. The following table lists the possible secondary specifiers and their allowed values.

Secondary Specifier	Allowed Value
port	A requested URL port
scheme	A request URL protocol; one of the following: • HTTP • FTP
prefix	A prefix in the path part of a URL
suffix	A file suffix in the URL
method	A request URL method; one of the following: • get • post • put • trace
time	A time range, such as 08:00-14:00, during which the parent cache is used to serve requests
src_ip	A client IP address.

The following table lists the possible actions and their allowed values.

Action	Allowed Value
parent	An ordered list of parent servers. If the request cannot be handled by the last parent server in the list, it will be routed to the origin server. You can specify either a hostname or an IP address. You must specify the port number.
round_robin	One of the following values: • `true` - Websense Content Gateway goes through the parent cache list in a round-robin based on client IP address. • `strict` - Websense Content Gateway machines serve requests strictly in turn. For example, machine `proxy1` serves the first request, `proxy2` serves the second request, and so on. • `false` - round-robin selection does not occur.
go_direct	One of the following values: • `true` - requests bypass parent hierarchies and go directly to the origin server. • `false` - requests do not bypass parent hierarchies.

Examples

The following rule configures a parent cache hierarchy consisting of Websense Content Gateway (which is the child) and two parents, `p1.x.com` and `p2.x.com`. The proxy forwards the requests it cannot serve to the parent servers `p1.x.com` and `p2.x.com` in a round-robin fashion because `round_robin=true`.

```
dest_domain=. method=get parent="p1.x.com:8080;
p2.y.com:8080" round_robin=true
```

The following rule configures Websense Content Gateway to route all requests containing the regular expression `politics` and the path **/viewpoint** directly to the origin server (bypassing any parent hierarchies):

```
url_regex=politics prefix=/viewpoint go_direct=true
```

Every line in the **parent.config** file must contain either a `parent=` or `go_direct=` directive.

records.config

The **records.config** file is a list of configurable variables that Websense Content Gateway software uses.

Many of the variables in the file are set when you set configuration options in Websense Content Manager or through the command-line interface. Some configuration options can be set only by editing variables in the **records.config** file.

> ⚠️ **Warning**
>
> Do not change the **records.config** variables unless you are certain of the effect. Many variables are coupled, meaning that they interact with other variables. Changing a single variable in isolation can cause Websense Content Gateway to fail. Whenever possible, use Websense Content Manager or the command-line interface to configure Websense Content Gateway.

> **Important**
>
> After you modify this file, run content_line -x from the Websense Content Gateway **bin** directory (default location is **/opt/WCG/bin**) to apply the changes. When you apply the changes to a node in a cluster, Websense Content Gateway applies the changes to all nodes in the cluster.

Format

Each variable has the following format:

```
CONFIG variable_name DATATYPE variable_value
```

where *DATATYPE* is INT (an integer), STRING (a string), or FLOAT (a floating point).

Examples

In the following example, the variable proxy.config.proxy_name is of datatype string and its value is my_server. This means that the name of the Websense Content Gateway proxy is my_server.

```
CONFIG proxy.config.proxy_name STRING my_server
```

In the following example, the variable *proxy.config.arm.enabled* is a yes/no flag. A value of 0 (zero) disables the option. A value of 1 enables the option.

```
CONFIG proxy.config.arm.enabled INT 0
```

In the following example, the variable sets the cluster startup timeout to 10 seconds.

```
CONFIG proxy.config.cluster.startup_timeout INT 10
```

Configuration variables

The following tables describe the configuration variables listed in the **records.config** file.

System variables

Local manager

Virtual IP manager

Alarm configuration

ARM (transparency configuration)

Load shedding configuration (ARM)

Authentication basic realm

LDAP

RADIUS authentication

NTLM

Transparent authentication

HTTP engine

Parent proxy configuration

Security

Cache control

Heuristic expiration

Dynamic content and content negotiation

Anonymous FTP password

Cached FTP document lifetime

FTP transfer mode

FTP engine

Customizable user response pages

SOCKS processor

Net subsystem

Cluster subsystem

Cache

DNS

HostDB

System variables

Configuration Variable Data Type	Default Value	Description
`proxy.config.proxy_name` STRING		Specifies the name of the Websense Content Gateway node.
`proxy.config.bin_path` STRING	`/opt/WCG/bin`	Specifies the location of the Websense Content Gateway **bin** directory.
`proxy.config.proxy_binary` STRING	`content_gateway` (UNIX)	Specifies the name of the executable that runs the `content_gateway` process.
`proxy.config.proxy_binary_ opts` STRING	`-M`	Specifies the command-line options for starting Websense Content Gateway.
`proxy.config.manager_ binary` STRING	`content_manager` (UNIX)	Specifies the name of the executable that runs the `content_manager` process.
`proxy.config.cli_binary` STRING	`content_line` (UNIX)	Specifies the name of the executable that runs the command-line interface.
`proxy.config.watch_script` STRING	`content_cop`	Specifies the name of the executable that runs the `content_cop` process.
`proxy.config.env_prep` STRING	`example_prep.sh` (UNIX)	Specifies the script that is executed before the `content_manager` process spawns the `content_gateway` process.

Configuration Variable Data Type	Default Value	Description
`proxy.config.config_dir` `STRING`	`config`	Specifies the directory that contains the Websense Content Gateway configuration files.
`proxy.config.temp_dir` `STRING`	`/tmp`	Specifies the directory used for Websense Content Gateway temporary files
`proxy.config.alarm_email` `STRING`	`websense`	Specifies the email address to which Websense Content Gateway sends alarm messages. During installation, you can specify the email address; otherwise, Websense Content Gateway uses the Websense Content Gateway user account name as the default value for this variable.

`proxy.config.syslog_`
`facility`
`STRING`
	`LOG_DAEMON`	Specifies the facility used to record system log files.
		See *Working With Log Files*, page 149.

`proxy.config.cop.core_`
`signal`
`INT`
	`0`	Specifies the signal that is sent to `content_cop`'s managed processes to stop them. `0` = no signal is sent.

`proxy.config.cop.linux_`
`min_swapfree_kb`
`INT`
	`10240`	Specifies the minimum amount of free swap space allowed before Websense Content Gateway stops the `content_gateway` and `content_manager` processes to prevent the system from hanging.
		
		This configuration variable applies if swap is enabled in Linux 2.2 only.

`proxy.config.cop.linux_`
`min_memfree_kb`
`INT`
	`10240`	Specifies the minimum amount of free memory allowed before Websense Content Gateway stops the `content_gateway` and `content_manager` processes to prevent the system from hanging.
		
		This configuration variable applies if swap is *disabled* in Linux 2.2 only.

Configuration Variable Data Type	Default Value	Description
proxy.config.output. logfile STRING	traffic.out	Specifies the name and location of the file that contains warnings, status messages, and error messages produced by the Websense Content Gateway processes. If no path is specified, Websense Content Gateway creates the file in its logging directory.
proxy.config.snapshot_dir STRING	snapshots	Specifies the directory in which Websense Content Gateway stores configuration snapshots on the local system. Unless you specify an absolute path, this directory is located in the Websense Content Gateway **config** directory.

Local manager

Configuration Variable Data Type	Default Value	Description
proxy.config.lm.sem_id INT	11452	Specifies the semaphore ID for the local manager.
proxy.local.cluster.type INT	3	Sets the clustering mode: • 1 = full-clustering mode • 2 = management-only mode • 3 = no clustering
proxy.config.cluster. rsport INT	8087	Specifies the reliable service port. The reliable service port is used to send configuration information between the nodes in a cluster. All nodes in a cluster must use the same reliable service port.
proxy.config.cluster. mcport INT	8088	Specifies the multicast port. The multicast port is used for node identification. All nodes in a cluster must use the same multicast port.
proxy.config.cluster. mc_group_addr STRING	224.0.1.37	Specifies the multicast address for cluster communications. All nodes in a cluster must use the same multicast address.
proxy.config.lm.sem_id INT	11452	Specifies the semaphore ID for the local manager.

Configuration Variable Data Type	Default Value	Description
`proxy.config.cluster.mc_ttl` `INT`	`1`	Specifies the multicast Time to Live for cluster communications.
`proxy.config.cluster.log_bogus_mc_msgs` `INT`	`1`	Enables (`1`) or disables (`0`) logging of bogus multicast messages.
`proxy.config.admin.html_doc_root` `STRING`	`ui`	Specifies the document root for Websense Content Manager.
`proxy.config.admin.web_interface_port` `INT`	`8081`	Specifies the Websense Content Manager port.
`proxy.config.admin.autoconf_port` `INT`	`8083`	Specifies the autoconfiguration port.
`proxy.config.admin.overseer_port` `INT`	`8082`	Specifies the port used for retrieving and setting statistics and configuration variables.
`proxy.config.admin.admin_user` `STRING`	`admin`	Specifies the administrator ID that controls access to Websense Content Manager.
`proxy.config.admin.admin_password` `STRING`		Specifies the encrypted administrator password that controls access to Websense Content Manager. You cannot edit the password; however, you can specify a value of `NULL` to clear the password. See *How do you access Websense Content Manager if you forget the master administrator password?*, page 354.
`proxy.config.admin.basic_auth` `INT`	`1`	Enables (1) or disables (0) basic user authentication to control access to Websense Content Manager. Note: If basic authentication is *not* enabled, any user can access Websense Content Manager to monitor and configure Websense Content Gateway.
`proxy.config.admin.use_ssl` `INT`	`0`	Enables the Websense Content Manager SSL option for secure communication between a remote host and the Websense Content Manager.

Configuration Variable Data Type	Default Value	Description

`proxy.config.admin.ssl_`
`cert_file`
`STRING`
	`private_key.pem`	Specifies the filename of the SSL certificate installed on the Websense Content Gateway system for secure communication between a remote host and Websense Content Manager.

`proxy.config.admin.number_`
`config_bak`
`INT`
	`3`	Specifies the maximum number of copies of rolled configuration files to keep.

`proxy.config.admin.user_id` `STRING`	`websense`	Specifies the non-privileged user account designated to Websense Content Gateway (UNIX only.)

`proxy.config.admin.ui_`
`refresh_rate`
`INT`
	`30`	Specifies the refresh rate for the display of statistics in the Monitor pages of Websense Content Manager.

`proxy.config.admin.log_`
`mgmt_access`
`INT`
	`0`	Enables (1) or disables (0) logging of all Websense Content Manager transactions to the **lm.log** file.

`proxy.config.admin.log_`
`resolve_hostname`
`INT`
	`1`	When enabled (1), the hostname of the client connecting to Websense Content Manager is recorded in the **lm.log** file.
		
		When disabled (0), the IP address of the client connecting to Websense Content Manager is not recorded in the **lm.log** file.

Process manager

Configuration Variable Data Type	Default Value	Description

`proxy.config.process_`
`manager.mgmt_port`
`INT`
	`8084`	Specifies the port used for internal communication between the `content_manager` process and the `content_gateway` process.

Virtual IP manager

Configuration Variable Data Type	Default Value	Description
`proxy.config.vmap.enabled` `INT`	`0`	Enables (1) or disables (0) the virtual IP option

Alarm configuration

Configuration Variable Data Type	Default Value	Description
`proxy.config.alarm.bin` `STRING`	`example_alarm_ bin.sh` (UNIX)	Specifies the name of the script file that can execute certain actions when an alarm is signaled. The default file is a sample script named **example_alarm_bin.sh** located in the **bin** directory. You must edit the script to suit your needs.
`proxy.config.alarm.abs_ path` `STRING`	`NULL`	Specifies the full path to the script file that sends email to indicate Websense Content Gateway problems.

ARM (transparency configuration)

Configuration Variable Data Type	Default Value	Description
`proxy.config.arm.enabled` `INT`	`0`	Enables (1) or disables (0) the ARM, which is used for transparent proxy caching, IP spoofing, and ARM security. See *Enabling the ARM option*, page 42.
`proxy.config.arm.ignore_ ifp` `INT`	`1`	Configures Websense Content Gateway to ignore the interface when sending packets back to the client if NAT rules are applied.

Configuration Variable Data Type	Default Value	Description
`proxy.config.arm.always_query_dest` `INT`	0	When enabled (1), Websense Content Gateway always asks the ARM driver for the original destination IP address of incoming requests. Therefore, IP addresses are logged, not domain names. When disabled, Websense Content Gateway logs domain names rather than IP addresses. See *Reducing DNS lookups*, page 58 for additional information on logging. It is recommended that you do not enable this variable if Websense Content Gateway is running in *both* explicit proxy caching mode and transparent proxy caching mode. In explicit proxy caching mode, the client does not perform a DNS lookup on the hostname of the origin server, so Websense Content Gateway must do it.
`proxy.config.http.outgoing_ip_ing_enabled` `INT`	0	Enables (1) or disables (0) the IP spoofing option, which allows Websense Content Gateway to establish connections to origin servers with the client IP address instead of the Websense Content Gateway IP address. Note: The variable `proxy.config.arm.enabled` must be enabled for the IP spoofing option to work.
`proxy.config.arm.bypass_dynamic_enabled` `INT`	0	Enables (1) or disables (0) the adaptive bypass option to bypass the proxy and go directly to the origin server when clients or servers cause problems.
`proxy.config.arm.bypass_use_and_rules_bad_client_request` `INT`	0	Enables (1) or disables (0) dynamic source/destination bypass in the event of non-HTTP traffic on port 80. Note: The variable `proxy.config.arm.bypass_on_bad_client_request` must also be enabled for this option to work.

Configuration Variable Data Type	Default Value	Description
`proxy.config.arm.bypass_use_and_rules_400` `INT`	0	Enables (1) or disables (0) dynamic generation of source/destination bypass rules when an origin server returns a 400 error. Note: The variable `proxy.config.arm.bypass_on_400` must also be enabled for this option to work.
`proxy.config.arm.bypass_use_and_rules_401` `INT`	0	Enables (1) or disables (0) dynamic generation of source/destination bypass rules when an origin server returns a 401 error. Note: The variable `proxy.config.arm.bypass_on_401` must also be enabled for this option to work.
`proxy.config.arm.bypass_use_and_rules_403` `INT`	0	Enables (1) or disables (0) dynamic generation of source/destination bypass rules when an origin server returns a 403 error. Note: The variable `proxy.config.arm.bypass_on_403` must also be enabled for this option to work.
`proxy.config.arm.bypass_use_and_rules_405` `INT`	0	Enables (1) or disables (0) dynamic generation of source/destination bypass rules when an origin server returns a 405 error. Note: The variable `proxy.config.arm.bypass_on_405` must also be enabled for this option to work.
`proxy.config.arm.bypass_use_and_rules_406` `INT`	0	Enables (1) or disables (0) dynamic generation of source/destination bypass rules when an origin server returns a 406 error. Note: The variable `proxy.config.arm.bypass_on_406` must also be enabled for this option to work.
`proxy.config.arm.bypass_use_and_rules_408` `INT`	0	Enables (1) or disables (0) dynamic generation of source/destination bypass rules when an origin server returns a 408 error. Note: The variable `proxy.config.arm.bypass_on_408` must also be enabled for this option to work.

Configuration Variable Data Type	Default Value	Description
`proxy.config.arm.bypass_use_and_rules_500` `INT`	0	Enables (1) or disables (0) dynamic generation of source/destination bypass rules when an origin server returns a 500 error. Note: The variable *proxy.config.arm. bypass_on_500* must also be enabled for this option to work.
`proxy.config.arm.bypass_on_bad_client_request` `INT`	0	Enables (1) or disables (0) dynamic destination bypass in the event of non-HTTP traffic on port 80.
`proxy.config.arm.bypass_on_400` `INT`	0	Enables (1) or disables (0) dynamic generation of destination bypass rules when an origin server returns a 400 error.
`proxy.config.arm.bypass_on_401` `INT`	0	Enables (1) or disables (0) dynamic generation of destination bypass rules when an origin server returns a 401 error.
`proxy.config.arm.bypass_on_403` `INT`	0	Enables (1) or disables (0) dynamic generation of destination bypass rules when an origin server returns a 403 error.
`proxy.config.arm.bypass_on_405` `INT`	0	Enables (1) or disables (0) dynamic generation of destination bypass rules when an origin server returns a 405 error.
`proxy.config.arm.bypass_on_406` `INT`	0	Enables (1) or disables (0) dynamic generation of destination bypass rules when an origin server returns a 406 error.
`proxy.config.arm.bypass_on_408` `INT`	0	Enables (1) or disables (0) dynamic generation of destination bypass rules when an origin server returns a 408 error.
`proxy.config.arm.bypass_on_500` `INT`	0	Enables (1) or disables (0) dynamic generation of destination bypass rules when an origin server returns a 500 error.

Load shedding configuration (ARM)

Configuration Variable Data Type	Default Value	Description
`proxy.config.arm. loadshedding.max_ connections` `INT`	1000000	Specifies the maximum number of client connections allowed before the proxy starts forwarding incoming requests directly to the origin server.

Authentication basic realm

Configuration Variable Data Type	Default Value	Description
`proxy.config.proxy. authenticate.basic.realm` `STRING`	NULL	Specifies the authentication realm name. If the default of `NULL` is specified, `Websense Content Gateway` is used.

LDAP

Configuration Variable Data Type	Default Value	Description
`proxy.config.ldap.auth. enabled` `INT`	0	Enables (1) or disables (0) LDAP proxy authentication. See *Using LDAP proxy authentication*, page 112.
`proxy.config.ldap.cache. size` `INT`	5000	Specifies the maximum number of entries allowed in the LDAP cache. If this value is modified, you must update the value of *proxy.config.ldap. cache.storage_size* proportionally. For example, if you double the cache size, also double the cache storage size.

Configuration Variable Data Type	Default Value	Description
`proxy.config.ldap.cache.` `storage_size` `INT`	`24582912`	Specifies the size of the LDAP cache in bytes. This is directly related to the number of entries in the cache. If this value is modified, you must update the value of *`proxy.config.ldap.`* *`cache.size`* proportionally. For example, if you double the storage size, also double the cache size. Modifying this variable without modifying *`proxy.config.ldap.`* *`cache.size`* can cause the LDAP subsystem to stop functioning.
`proxy.config.ldap.auth.` `ttl_value` `INT`	`3000`	Specifies the amount of time (in minutes) that entries in the cache remain valid.
`proxy.config.ldap.auth.` `purge_cache_on_auth_fail` `INT`	`0`	When enabled (1), configures Websense Content Gateway to delete the authorization entry for the client in the LDAP cache if authorization fails.
`proxy.config.ldap.proc.` `ldap.server.name` `STRING`	`NULL`	Specifies the LDAP server name.
`proxy.config.ldap.proc.` `ldap.server.port` `INT`	`389`	Specifies the LDAP port.
`proxy.config.ldap.proc.` `ldap.server.bind_dn`	`NULL`	Specifies the Full Distinguished Name (fully qualified name) of a user in the LDAP-based directory service. For example: `CN=John` `Smith,CN=USERS,` `DC=MYCOMPANY,DC=COM` Enter a maximum of 128 characters in this field. If you do not specify a value for this field, the proxy attempts to bind anonymously.
`proxy.config.ldap.proc.` `ldap.server.bind_pwd`	`NULL`	Specifies a password for the user identified by the *`proxy.config.ldap.`* *`proc.ldap.server.bind_`* *`dn`* variable.

Configuration Variable Data Type	Default Value	Description

`proxy.config.ldap.proc.`
`ldap.base.dn`
`STRING`
	NULL	Specifies the LDAP Base Distinguished Name (DN). Obtain this value from your LDAP administrator.

`proxy.config.ldap.proc.`
`ldap.uid_filter`
`STRING`
	sAMAccountName	Specifies the LDAP login name/ID. Use this as a filter to search the full DN database.
		
		For eDirectory or other directory services, enter **uid** in this field.

RADIUS authentication

Configuration Variable Data Type	Default Value	Description

`proxy.config.radius.auth.`
`enabled`
`INT`
	0	Enables (1) or disables (0) RADIUS proxy authentication.

`proxy.config.radius.proc.`
`radius.primary_server.name`
`STRING`
	NULL	Specifies the hostname or IP address of the primary RADIUS authentication server.

`proxy.config.radius.proc.`
`radius.primary_`
`server.auth_port`
`INT`
	1812	Specifies the port that the proxy uses to communicate with the primary RADIUS authentication server.

`proxy.config.radius.proc.`
`radius.primary_server.`
`shared_key`
`STRING`
	NULL	Specifies the key used for encoding with the first RADIUS authentication server.

`proxy.config.radius.proc.`
`radius.secondary_server.`
`name`
`STRING`
	NULL	Specifies the hostname or IP address of the secondary RADIUS authentication server.

`proxy.config.radius.proc.`
`radius.secondary_server.`
`auth_port`
`INT`
	1812	Specifies the port that the proxy uses to communicate with the secondary RADIUS authentication server.

`proxy.config.radius.proc.`
`radius.secondary_server.`
`shared_key`
`STRING`
	NULL	Specifies the key used for encoding with the secondary RADIUS authentication server.

`proxy.config.radius.auth.`
`min_timeout`
`INT`
	10	Specifies the amount of time the connection to the RADIUS server can remain idle before Websense Content Gateway closes the connection.

Configuration Variable Data Type	Default Value	Description
`proxy.config.radius.auth max_retries` INT	10	Specifies the maximum number of times Websense Content Gateway tries to connect to the RADIUS server.
`proxy.config.radius.cache. size` INT	1000	Specifies the number of entries allowed in the RADIUS cache. The minimum value is 256 entries. If you enter a lower value, Websense Content Gateway signals a SEGV.
`proxy.config.radius.cache. storage_size` INT	15728640	Specifies the maximum amount of space that the RADIUS cache can occupy on disk. This value must be at least one hundred times the number of entries. It is recommended that you provide the maximum amount of disk space possible.
`proxy.config.radius.auth. ttl_value` INT	60	Specifies the number of minutes that Websense Content Gateway stores username and password entries in the RADIUS cache.

NTLM

Configuration Variable Data Type	Default Value	Description
`proxy.config.ntlm.auth. enabled` INT	0	Enables (1) or disables (0) NTLM proxy authentication.
`proxy.config.ntlm.dc.list` STRING	NULL	Specifies the hostnames of the domain controllers. You must separate each entry with a comma; for example: `host1, host2, host3,` Enter the hostname in IP address format.
`proxy.config.ntlm.dc.load_ balance` INT	0	Enables (1) or disables (0) load balancing. When enabled, Websense Content Gateway balances the load when sending authentication requests to the domain controllers.
`proxy.config.ntlm.dc.max_ connections` INT	3	Specifies the maximum number of connections Websense Content Gateway can have open to the domain controller.

Configuration Variable Data Type	Default Value	Description
`proxy.config.ntlm.nt_ domain` `STRING`	`NULL`	Specifies the NT domain name you want the proxy to authenticate against. Enter only the name of the domain; do not enter an extension (such as ".com" or "org").
`proxy.config.ntlm.cache. enabled` `INT`	`1`	Enables (1) or disables (0) the NTLM cache. When disabled, Websense Content Gateway does not store any credentials in the NTLM cache for future use. Websense Content Gateway always sends the credentials to the domain server to be validated.
`proxy.config.ntlm.cache. ttl_value` `INT`	`3600`	Specifies the number of seconds that Websense Content Gateway stores entries in the NTLM cache.
`proxy.config.ntlm.cache. size` `INT`	`5000`	Specifies the number of entries allowed in the NTLM cache.
`proxy.config.ntlm.cache. storage_size` `INT`	`15728640`	Specifies the maximum amount of space that the NTLM cache can occupy on disk. This value should be proportionate to number of entries in the NTLM cache.

Transparent authentication

Configuration Variable Data Type	Default Value	Description
`proxy.config.http.transparent_auth_type` `INT`	1	Enter: • 0 to associate a session ID with the username after the user session is authenticated. This setting is required to uniquely identify users who share a single IP address, such as in proxy-chaining or network address translation. • 1 to associate a client IP address with a username after the user session is authenticated. In either mode, the length of time before a client must re-authenticate is determined by the value of the `proxy.config.http.transparent_auth_sessi on_ time` variable. For additional information, see *Authentication in transparent proxy mode*, page 110.
`proxy.config.http.transparent_auth_session_ time` `INT`	15	Specify the length of time (in minutes) before the browser must re-authenticate. This is required in both IP and cookie modes.

HTTP engine

Configuration Variable Data Type	Default Value	Description
`proxy.config.http.server_ port` `INT`	8080	Specifies the port that Websense Content Gateway uses when acting as a Web proxy server for Web traffic or when serving Web traffic transparently.
`proxy.config.http.server_ port_attr` `STRING`	X	Specifies the server port options. You can specify one of the following: • `C=SERVER_PORT_COMPRESSED` • `X=SERVER_PORT_DEFAULT` • `T=SERVER_PORT_BLIND_TUNN EL`

Configuration Variable Data Type	Default Value	Description
`proxy.config.http.server_other_ports` STRING	NULL	Specifies the ports other than the port specified by the variable *proxy.config.http.server_port* to bind for incoming HTTP requests.
`proxy.config.http.ssl_ports` STRING	443 563	Specifies the range of ports used for tunneling. Websense Content Gateway allows tunnels only to the specified ports.
`proxy.config.http.insert_request_via_str` INT	1	Specify one of the following: • 0 = no extra information is added to the string. • 1 = all extra information is added. • 2 = some extra information is added.
`proxy.config.http.insert_response_via_str` INT	1	Specify one of the following: • 0 = no extra information is added to the string. • 1 = all extra information is added. • 2 = some extra information is added.
`proxy.config.http.enable_url_expandomatic` INT	1	Enables (1) or disables (0) **.com** domain expansion, which configures Websense Content Gateway to attempt to resolve unqualified hostnames by redirecting them to the expanded address, prepended with **www.** and appended with **.com**; for example, if a client makes a request to *host*, Websense Content Gateway redirects the request to www.host.com.
`proxy.config.http.uncacheable_requests_bypass_parent` INT	1	When enabled (1), Websense Content Gateway bypasses the parent proxy for a request that is not cacheable.
`proxy.config.http.keep_alive_enabled` INT	1	Enables (1) or disables (0) the use of keep-alive connections to either origin servers or clients.

Configuration Variable Data Type	Default Value	Description
`proxy.config.http.chunking_enabled` `INT`	0	Specifies whether Websense Content Gateway can generate a chunked response: • 0 = Never • 1 = Always • 2 = Generate a chunked response if the server has returned HTTP 1.1 before • 3 = Generate a chunked response if the client request is HTTP 1.1 and the origin server has returned HTTP 1.1 before
`proxy.config.http.send_http11_requests` `INT`	3	Configures Websense Content Gateway to use HTTP Version 1.1 when communicating with origin servers. You can specify one of the following values: • 1 = Websense Content Gateway always uses HTTP 1.1 when communicating with origin servers. • 2 = Websense Content Gateway uses HTTP 1.1 if the origin server has previously used HTTP 1.1. • 3 = Websense Content Gateway uses HTTP 1.1 if the client request is HTTP 1.1 and the origin server has previously used HTTP 1.1. Note: If HTTP 1.1 is used, Websense Content Gateway can use keep-alive connections with pipelining to origin servers. If HTTP 0.9 is used, Websense Content Gateway does not use keep-alive connections to origin servers. If HTTP 1.0 is used, a Websense Content Gateway can use keep-alive connections without pipelining to origin servers.
`proxy.config.http.share_server_sessions` `INT`	1	Enables (1) or disables (0) the re-use of server sessions. Note: When IP spoofing is enabled, Websense Content Gateway automatically disables this variable.

Configuration Variable Data Type	Default Value	Description
`proxy.config.http.ftp_ enabled` `INT`	1	Enables (1) or disables (0) Websense Content Gateway from serving FTP requests sent via HTTP.
`proxy.config.http.record_ heartbeat` `INT`	0	Enables (1) or disables (0) `content_cop` heartbeat logging.

Parent proxy configuration

Configuration Variable Data Type	Default Value	Description
`proxy.config.http.parent_ proxy_routing_enable` `INT`	0	Enables (1) or disables (0) the HTTP parent caching option. See *Hierarchical Caching*, page 69.
`proxy.config.http.parent_ proxy.retry_time` `INT`	300	Specifies the amount of time allowed between connection retries to a parent cache that is unavailable.
`proxy.config.http.parent_ proxy.fail_threshold` `INT`	10	Specifies the number of times the connection to the parent cache can fail before Websense Content Gateway considers the parent unavailable.
`proxy.config.http.parent_ proxy.total_connect_ attempts` `INT`	4	Specifies the total number of connection attempts allowed to a parent cache before Websense Content Gateway bypasses the parent or fails the request (depending on the `go_direct` option in the `bypass.config` file).
`proxy.config.http.parent_ proxy.per_parent_connect_ attempts` `INT`	2	Specifies the total number of connection attempts allowed per parent if multiple parents are used.
`proxy.config.http.parent_ proxy.connect_attempts_ timeout INT`	30	Specifies the timeout value in seconds for parent cache connection attempts.

HTTP connection timeouts (secs)

Configuration Variable Data Type	Default Value	Description
`proxy.config.http.keep_alive_no_activity_timeout_in` INT	10	Specifies how long Websense Content Gateway keeps connections to clients open for a subsequent request after a transaction ends.
`proxy.config.http.keep_alive_no_activity_timeout_out` INT	10	Specifies how long Websense Content Gateway keeps connections to origin servers open for a subsequent transfer of data after a transaction ends.
`proxy.config.http.transaction_no_activity_timeout_in` INT	120	Specifies how long Websense Content Gateway keeps connections to clients open if a transaction stalls.
`proxy.config.http.transaction_no_activity_timeout_out` INT	120	Specifies how long Websense Content Gateway keeps connections to origin servers open if the transaction stalls.
`proxy.config.http.transaction_active_timeout_in` INT	0	Specifies how long Websense Content Gateway remains connected to a client. If the transfer to the client is not complete before this timeout expires, Websense Content Gateway closes the connection. The default value of 0 specifies that there is no timeout.
`proxy.config.http.transaction_active_timeout_out` INT	0	Specifies how long Websense Content Gateway waits for fulfillment of a connection request to an origin server. If Websense Content Gateway does not complete the transfer to the origin server before this timeout expires, the connection request is terminated. The default value of 0 specifies that there is no timeout.
`proxy.config.http.accept_no_activity_timeout` INT	120	Specifies the timeout interval in seconds before Websense Content Gateway closes a connection that has no activity.

Configuration Variable Data Type	Default Value	Description
`proxy.config.http.` `background_fill_active_` `timeout` `INT`	60	Specifies how long Websense Content Gateway continues a background fill before giving up and dropping the origin server connection.
`proxy.config.http.` `background_fill_completed_` `threshold` `FLOAT`	0.50000	Specifies the proportion of total document size already transferred when a client aborts at which the proxy continues fetching the document from the origin server to get it into the cache (a *background fill*).

Origin server connection attempts

Configuration Variable Data Type	Default Value	Description
`proxy.config.http.connect_` `attempts_max_retries` `INT`	6	Specifies the maximum number of connection retries Websense Content Gateway makes when the origin server is not responding.
`proxy.config.http.connect_` `attempts_max_retries_dead_` `server` `INT`	2	Specifies the maximum number of connection retries Websense Content Gateway makes when the origin server is unavailable.
`proxy.config.http.connect_` `attempts_rr_retries` `INT`	2	Specifies the maximum number of failed connection attempts allowed before a round-robin entry is marked as down if a server has round-robin DNS entries.
`proxy.config.http.connect_` `attempts_timeout` `INT`	30	Specifies the timeout value in seconds for an origin server connection.
`proxy.config.http.down_` `server.cache_time` `INT`	900	Specifies how long in seconds Websense Content Gateway remembers that an origin server was unreachable.
`proxy.config.http.down_` `server.abort_threshold` `INT`	10	Specifies the number of seconds before Websense Content Gateway marks an origin server as unavailable when a client abandons a request because the origin server was too slow in sending the response header.

Negative response caching

Configuration Variable Data Type	Default Value	Description
proxy.config.http. negative_caching_enabled INT	0	When enabled (1), Websense Content Gateway caches negative responses, such as *404 Not Found*, if a requested page does not exist. The next time a client requests the same page, Websense Content Gateway serves the negative response from the cache. Websense Content Gateway caches the following negative responses: 204 No Content 305 Use Proxy 400 Bad Request 403 Forbidden 404 Not Found 405 Method Not Allowed 500 Internal Server Error 501 Not Implemented 502 Bad Gateway 503 Service Unavailable 504 Gateway Timeout
proxy.config.http. negative_caching_lifetime INT	1800	Specifies how long Websense Content Gateway keeps the negative responses as valid in cache.

Proxy users variables

Configuration Variable Data Type	Default Value	Description
proxy.config.http. anonymize_remove_from INT	0	When enabled (1), Websense Content Gateway removes the `From` header that accompanies transactions to protect the privacy of your users.
proxy.config.http. anonymize_remove_referer INT	0	When enabled (1), Websense Content Gateway removes the `Referer` header that accompanies transactions to protect the privacy of your site and users.

Configuration Variable Data Type	Default Value	Description
`proxy.config.http. anonymize_remove_user_ agent` `INT`	0	When enabled (1), Websense Content Gateway removes the `User-agent` header that accompanies transactions to protect the privacy of your site and users.
`proxy.config.http. anonymize_remove_cookie` `INT`	0	When enabled (1), Websense Content Gateway removes the `Cookie` header that accompanies transactions to protect the privacy of your site and users.
`proxy.config.http. anonymize_remove_client_ip` `INT`	0	When enabled (1), Websense Content Gateway removes `Client-IP` headers for more privacy.
`proxy.config.http. anonymize_insert_client_ip` `INT`	1	When enabled (1), Websense Content Gateway inserts `Client-IP` headers to retain the client's IP address.
`proxy.config.http.append_ xforwards_header` `INT`	0	When enabled (1), Websense Content Gateway appends `X-Forwards` headers to outgoing requests.
`proxy.config.http. anonymize_other_header_ list` `STRING`	NULL	Specifies the headers that Websense Content Gateway will remove from outgoing requests.
`proxy.config.http.snarf_ username_from_ authorization` `INT`	0	When enabled (1), Websense Content Gateway takes the username and password from the authorization header for LDAP if the authorization scheme is *Basic*.
`proxy.config.http.insert_ squid_x_forwarded_for` `INT`	0	When enabled (1), Websense Content Gateway adds the client IP address to the `X-Forwarded-For` header.

Security

Cache control

Configuration Variable Data Type	Default Value	Description
`proxy.config.http.cache.http` INT	1	Enables (1) or disables (0) caching of HTTP requests.
`proxy.config.http.cache.ftp` INT	1	Enables (1) or disables (0) caching of FTP requests sent via HTTP.
`proxy.config.http.cache.ignore_client_no_cache` INT	0	When enabled (1), Websense Content Gateway ignores client requests to bypass the cache.
`proxy.config.http.cache.ims_on_client_no_cache` INT	0	When enabled (1), Websense Content Gateway issues a conditional request to the origin server if an incoming request has a `no-cache` header.
`proxy.config.http.cache.ignore_server_no_cache` INT	0	When enabled (1), Websense Content Gateway ignores origin server requests to bypass the cache.
`proxy.config.http.cache.cache_responses_to_cookies` INT	3	Specifies how cookies are cached: • 0 = do not cache any responses to cookies • 1 = cache for any content-type • 2 = cache only for image types • 3 = cache for all but text content-types
`proxy.config.http.cache.ignore_authentication` INT	0	When enabled (1), Websense Content Gateway ignores `WWW-Authentication` headers in responses. `WWW-Authentication` headers are removed and not cached.
`proxy.config.http.cache.cache_urls_that_look_dynamic` INT	0	Enables (1) or disables (0) caching of URLs that look dynamic.
`proxy.config.http.cache.enable_default_vary_headers` INT	0	Enables (1) or disables (0) caching of alternate versions of HTTP objects that do not contain the `Vary` header.

Configuration Variable Data Type	Default Value	Description
`proxy.config.http.cache.when_to_revalidate` `INT`	0	Specifies when to revalidate content: • 0 = Use cache directives or heuristic (the default value). • 1 = Stale if heuristic. • 2 = Always stale (always revalidate). • 3 = Never stale. • 4 = Use cache directives or heuristic (0) unless the request has an `If-Modified-Since` header. If the request has an `If-Modified-Since` header, Websense Content Gateway always revalidates the cached content and uses the client's `If-Modified-Since` header for the proxy request.
`proxy.config.http.cache.when_to_add_no_cache_to_msie_requests` `INT`	0	Specifies when to add `no-cache` directives to Microsoft Internet Explorer requests. You can specify the following: • 0 = `no-cache` not added to MSIE requests. • 1 = `no-cache` added to IMS MSIE requests. • 2 = `no-cache` added to all MSIE requests.
`proxy.config.http.cache.required_headers` `INT`	0	Specifies the type of headers required in a request for the request to be cacheable. • 0 = no required headers to make document cacheable. • 1 = at least `Last-Modified` header required. • 2 = explicit lifetime required, `Expires` or `Cache-Control`.
`proxy.config.http.cache.max_stale_age` `INT`	604800	Specifies the maximum age allowed for a stale response before it cannot be cached.
`proxy.config.http.cache.range.lookup` `INT`	1	When enabled (1), Websense Content Gateway looks up range requests in the cache.

Heuristic expiration

Configuration Variable Data Type	Default Value	Description
`proxy.config.http.cache.heuristic_min_lifetime` `INT`	`3600`	Specifies the minimum amount of time that a document in the cache can be considered fresh.
`proxy.config.http.cache.heuristic_max_lifetime` `INT`	`86400`	Specifies the maximum amount of time that a document in the cache can be considered fresh.
`proxy.config.http.cache.heuristic_lm_factor` `FLOAT`	`0.10000`	Specifies the aging factor for freshness computations.
`proxy.config.http.cache.fuzz.time` `INT`	`240`	Specifies the interval in seconds before the document stale time that the proxy checks for an early refresh.
`proxy.config.http.cache.fuzz.probability` `FLOAT`	`0.00500`	Specifies the probability that a refresh is made on a document during the specified fuzz time.
`proxy.config.http.cache.when_to_revalidate` `INT`	`0`	Specifies when to revalidate content: • 0 = Use cache directives or heuristic (the default value). • 1 = Stale if heuristic. • 2 = Always stale (always revalidate). • 3 = Never stale. • 4 = Use cache directives or heuristic (0) unless the request has an `If-Modified-Since` header. If the request has an `If-Modified-Since` header, Websense Content Gateway always revalidates the cached content and uses the client's `If-Modified-Since` header for the proxy request.
`proxy.config.http.cache.when_to_add_no_cache_to_msie_requests` `INT`	`0`	Specifies when to add `no-cache` directives to Microsoft Internet Explorer requests. You can specify the following: • 0 = `no-cache` not added to MSIE requests. • 1 = `no-cache` added to IMS MSIE requests. • 2 = `no-cache` added to all MSIE requests.

Dynamic content and content negotiation

Configuration Variable Data Type	Default Value	Description
`proxy.config.http.cache. vary_default_text` `STRING`	NULL	Specifies the header on which Websense Content Gateway varies for text documents; for example, if you specify `user-agent`, the proxy caches all the different user-agent versions of documents it encounters.
`proxy.config.http.cache. vary_default_images` `STRING`	NULL	Specifies the header on which Websense Content Gateway varies for images.
`proxy.config.http.cache. vary_default_other` `STRING`	NULL	Specifies the header on which Websense Content Gateway varies for anything other than text and images.

Anonymous FTP password

Configuration Variable Data Type	Default Value	Description
`proxy.config.http.ftp. anonymous_passwd` `STRING`	`the value of the administrator's email as supplied during installation`	Specifies the anonymous password for FTP servers that require a password for access. Websense Content Gateway uses the Websense Content Gateway user account name as the default value for this variable.

Cached FTP document lifetime

Configuration Variable Data Type	Default Value	Description
`proxy.config.http.ftp. cache.document_lifetime` `INT`	259200	Specifies the maximum amount of time that an FTP document can stay in the cache.

FTP transfer mode

Configuration Variable Data Type	Default Value	Description
`proxy.config.http.ftp.binary_transfer_only` `INT`	0	When enabled (1), all FTP documents requested from HTTP clients are transferred in binary mode only. When disabled (0), FTP documents requested from HTTP clients are transferred in ASCII or binary mode, depending on the document type.

Customizable user response pages

Configuration Variable Data Type	Default Value	Description
`proxy.config.body_factory.enable_customizations` `INT`	0	Specifies whether customizable response pages are enabled or disabled and which response pages are used: • 0 = disable customizable user response pages • 1 = enable customizable user response pages in the default directory only • 2 = enable language-targeted user response pages
`proxy.config.body_factory.enable_logging` `INT`	1	Enables (1) or disables (0) logging for customizable response pages. When enabled, Websense Content Gateway records a message in the error log each time a customized response page is used or modified.
`proxy.config.body_factory.template_sets_dir` `STRING`	config/ body_factory	Specifies the customizable response page default directory.
`proxy.config.body_factory.response_suppression_mode` `INT`	0	Specifies when Websense Content Gateway suppresses generated response pages: • 0 = never suppress generated response pages • 1 = always suppress generated response pages • 2 = suppress response pages only for intercepted traffic

FTP engine

Configuration Variable Data Type	Default Value	Description
FTP over HTTP		
`proxy.config.ftp.data_ connection_mode` `INT`	`1`	Specifies the FTP connection mode: • 1 = PASV then PORT • 2 = PORT only • 3 = PASV only
`proxy.config.ftp.control_ connection_timeout` `INT`	`300`	Specifies how long Websense Content Gateway waits for a response from the FTP server.
`proxy.config.ftp.rc_to_ switch_to_PORT` `STRING`	`NULL`	Specifies the response codes for which Websense Content Gateway automatically fails over to the PORT command when PASV fails if the configuration variable *proxy.config.ftp.data_ connection_mode* is set to 1. This variable is used for FTP requests from HTTP clients only.
FTP Proxy		
`proxy.config.ftp.ftp_ enabled` `INT`	`0`	Enables (1) or disables (0) processing of FTP requests from FTP clients.
`proxy.config.ftp.cache_ enabled` `INT`	`1`	Enables (1) or disables (0) FTP documents to be put in the cache. If this option is disabled, Websense Content Gateway always serves FTP documents from the FTP server.
`proxy.config.ftp.logging_ enabled` `INT`	`1`	Enables (1) or disables (0) logging of FTP transactions.
`proxy.config.ftp.proxy_ server_port` `INT`	`21`	Specifies the port used for FTP connections.

Configuration Variable Data Type	Default Value	Description
`proxy.config.ftp.open_ lisn_port_mode` `INT`	1	Specifies how FTP opens a listening port for a data transfer: • 1 = The operating system chooses an available port. Websense Content Gateway sends 0 and retrieves the new port number if the listen succeeds. • 2 = The listening port is determined by the range of ports specified by the Websense Content Gateway variables *proxy.config.ftp. min_lisn_port* and *proxy.config.ftp. max_lisn_port,* described below.
`proxy.config.ftp.min_ lisn_port` `INT`	32768	Specifies the lowest port in the range of listening ports used by Websense Content Gateway for data connections when the FTP client sends a PASV or Websense Content Gateway sends a PORT to the FTP server.
`proxy.config.ftp.max_ lisn_port` `INT`	65535	Specifies the highest port in the range of listening ports used by Websense Content Gateway for data connections when the FTP client sends a PASV or Websense Content Gateway sends a PORT to the FTP server.
`proxy.config.ftp.server_ data_default_pasv` `INT`	1	Specifies the default method used to set up server side data connections: • 1 = Websense Content Gateway sends a PASV to the FTP server and lets the FTP server open a listening port. • 0 = Websense Content Gateway tries PORT first (sets up a listening port on the proxy side of the connection).

Configuration Variable Data Type	Default Value	Description
`proxy.config.ftp. different_client_port_ip_ allowed` INT	0	When enabled (1), Websense Content Gateway can connect to a machine other than the one on which the FTP client is running to establish a data connection. The FTP client uses PORT to set up a listening port on its side and allows Websense Content Gateway to connect to that port to establish the data connection (used to transfer files). When setting up the listening port, an FTP client specifies the IP address and port number for the listening port. If this variable is set to 0 (zero), Websense Content Gateway cannot connect to the FTP client if the IP address sent by the client is different from the IP address of the machine running the FTP client.
`proxy.config.ftp.try_pasv_ times` INT	1024	Specifies the number of times Websense Content Gateway can try to open a listening port when the FTP client sends a PASV.
`proxy.config.ftp.try_port_ times` INT	1024	Specifies the maximum number of times Websense Content Gateway can try to open a listening port when sending a PORT to the FTP server.
`proxy.config.ftp.try_ server_ctrl_connect_times` INT	6	Specifies the maximum number of times Websense Content Gateway can try to connect to the FTP server's control listening port.
`proxy.config.ftp.try_ server_data_connect_times` INT	3	Specifies the maximum number of times Websense Content Gateway can try to connect to the FTP server's data listening port when it sends a PASV to the FTP server and gets the IP/listening port information.
`proxy.config.ftp.try_ client_data_connect_times` INT	3	Specifies the maximum number of times Websense Content Gateway can try to connect to the FTP client's data listening port when the FTP client sends a PORT with the IP/listening port information.
`proxy.config.ftp.client_ ctrl_no_activity_timeout` INT	900	Specifies the inactivity timeout for the FTP client control connection.

Configuration Variable Data Type	Default Value	Description

`proxy.config.ftp.client_`
`ctrl_active_timeout`
`INT`
	14400	Specifies the active timeout for the FTP client control connection.

`proxy.config.ftp.server_`
`ctrl_no_activity_timeout`
`INT`
	900	Specifies the inactivity timeout for the FTP server control connection.

`proxy.config.ftp.server_`
`ctrl_active_timeout`
`INT`
	14400	Specifies the active timeout for the FTP server control connection.

`proxy.config.ftp.pasv_`
`accept_timeout`
`INT`
	120	Specifies the timeout value for a listening data port in Websense Content Gateway (for PASV, the client data connection).

`proxy.config.ftp.port_`
`accept_timeout`
`INT`
	120	Specifies the timeout value for a listening data port in Websense Content Gateway (for PORT, the server data connection).

`proxy.config.ftp.share_`
`ftp_server_ctrl_enabled`
`INT`
	1	Enables (1) or disables (0) sharing the server control connections among multiple anonymous FTP clients.

`proxy.config.ftp.share_`
`only_after_session_end`
`INT`
	1	Specifies how an FTP server control connection is shared between different FTP client sessions:
		
		• 1 = the FTP server control connection can be used by another FTP client session *only* when the FTP client session is complete (typically, when the FTP client sends out a QUIT command).
		
		• 0 = the FTP server control connection can be used by another FTP client session *only* if the FTP client session is not actively using the FTP server connection: for example, if the request is a cache hit or during an idle session.

`proxy.config.ftp.server_`
`ctrl_keep_alive_no_`
`activity_timeout`
`INT`
	90	Specifies the timeout value when the FTP server control connection is not used by any FTP clients.

`proxy.config.ftp.login_`
`info_fresh_in_cache_time`
`INT`
	2592000	Specifies how long the 220/230 responses (login messages) can stay fresh in the cache.

Configuration Variable Data Type	Default Value	Description
`proxy.config.ftp.` `directory_listing_fresh_` `in_cache_time` `INT`	604800	Specifies how long directory listings can stay fresh in the cache.
`proxy.config.ftp.file_` `fresh_in_cache_time` `INT`	259200	Specifies how long FTP files can stay fresh in the cache.
`proxy.config.ftp.simple_` `directory_listing_cache_` `enabled` `INT`	1	Enables (1) or disables (0) caching of directory listings without arguments: for example, **dir/ls.**
`proxy.config.ftp.full_` `directory_listing_cache_` `enabled` `INT`	1	Enables (1) or disables (0) caching of directory listings with arguments: for example, `ls -al, ls *.txt.`

SOCKS processor

Configuration Variable Data Type	Default Value	Description
`proxy.config.socks.socks_` `needed` `INT`	0	Enables (1) or disables (0) the SOCKS option. See *Configuring SOCKS firewall integration*, page 106.
`proxy.config.socks.socks_` `version` `INT`	4	Specifies the SOCKS version.
`proxy.config.socks.` `default_servers` `STRING`		Specifies the names and ports of the SOCKS servers with which Websense Content Gateway communicates.
`proxy.config.socks.accept_` `enabled` `INT`	0	Enables (1) or disables (0) the SOCKS proxy option. As a SOCKS proxy, Websense Content Gateway receives SOCKS traffic (usually on port 1080) and forwards all requests directly to the SOCKS server.
`proxy.config.socks.accept_` `port` `INT`	1080	Specifies the port on which Websense Content Gateway accepts SOCKS traffic.

Net subsystem

Configuration Variable Data Type	Default Value	Description
`proxy.config.net. connections_throttle` `INT`	`10000`	Specifies the maximum number of connections that Websense Content Gateway can handle. If Websense Content Gateway receives additional client requests, they are queued until existing requests are served. Do not set this variable below 100.

Cluster subsystem

Configuration Variable Data Type	Default Value	Description
`proxy.config.cluster. cluster_port`	`8086`	Specifies the port used for cluster communication.
`proxy.config.cluster. ethernet_interface`	*your_interface*	Specifies the network interface used for cluster traffic. All nodes in a cluster must use the same network interface.

Cache

Configuration Variable Data Type	Default Value	Description
`proxy.config.cache.permit. pinning` `INT`	`0`	Enables (1) or disables (0) the cache pinning option, which lets you keep objects in the cache for a specified time. You set cache pinning rules in the `cache.config` file (see *cache.config*, page 261).
`proxy.config.cache.ram_ cache.size` `INT`	`-1`	Specifies the size of the RAM cache, in bytes. -1 means that the RAM cache is automatically sized at approximately one MB per gigabyte of disk.

Configuration Variable Data Type	Default Value	Description

`proxy.config.cache.limits.`
`http.max_alts`
`INT`
	3	Specifies the maximum number of HTTP alternates that Websense Content Gateway can cache.

`proxy.config.cache.max_`
`doc_size`
`INT`
	0	Specifies the maximum size of documents in the cache (in bytes):
		0 = there is no size limit.

DNS

Configuration Variable Data Type	Default Value	Description

`proxy.config.dns.search_`
`default_domains`
`INT`
	1	Enables (1) or disables (0) local domain expansion so that Websense Content Gateway can attempt to resolve unqualified hostnames by expanding to the local domain; for example, if a client makes a request to an unqualified host named `host_x`, and if the Websense Content Gateway local domain is `y.com`, Websense Content Gateway expands the hostname to `host_x.y.com`.

`proxy.config.dns.url_`
`expansions`
`STRING`
	NULL	Specifies a list of hostname extensions that are automatically added to the hostname after a failed lookup; for example, if you want Websense Content Gateway to add the hostname extension `.org`, specify `org` as the value for this variable (Websense Content Gateway automatically adds the dot (`.`).)
		
		Note: If the variable `proxy.config.http.`
		`enable_url_`
		`expandomatic` is set to 1 (the default value), you do not have to add `www.` and `.com` to this list; Websense Content Gateway tries `www.` and `.com` automatically after trying the values you specify.

HostDB

Configuration Variable Data Type	Default Value	Description
proxy.config.hostdb.size INT	200000	Specifies the maximum number of entries allowed in the host database.
proxy.config.hostdb.ttl_ mode INT	0	Specifies the host database time to live mode. You can specify one of the following: • 0 = obey • 1 = ignore • 2 = min(X,ttl) • 3 = max(X,ttl)
proxy.config.hostdb. timeout INT	1440	Specifies the foreground timeout, in minutes.
proxy.config.hostdb. strict_round_robin INT	0	When disabled (0), Websense Content Gateway always uses the same origin server for the same client as long as the origin server is available.

Logging configuration

Configuration Variable Data Type	Default Value	Description
proxy.config.log2.logging_ enabled INT	3	Enables and disables event logging: • 0 = logging disabled • 1 = log errors only • 2 = log transactions only • 3 = full logging (errors + transactions) See *Working With Log Files*, page 149.
proxy.config.log2.max_ secs_per_buffer INT	5	Specifies the maximum amount of time before data in the buffer is flushed to disk.
proxy.config.log2.max_ space_mb_for_logs INT	2000	Specifies the amount of space allocated to the logging directory, in megabytes.

Configuration Variable Data Type	Default Value	Description
`proxy.config.log2.max_ space_mb_for_orphan_logs` `INT`	`25`	Specifies the amount of space allocated to the logging directory, in megabytes, if this node is acting as a collation client.
`proxy.config.log2.max_ space_mb_headroom` `INT`	`10`	Specifies the tolerance for the log space limit in bytes. If the variable *proxy.config.log2. auto_delete_rolled_ file* is set to 1 (enabled), autodeletion of log files is triggered when the amount of free space available in the logging directory is less than the value specified here.
`proxy.config.log2.hostname` `STRING`	`localhost`	Specifies the hostname of the machine running Websense Content Gateway.
`proxy.config.log2.logfile_ dir` `STRING`	*install_dir*/ `logs`	Specifies the full path to the logging directory.
`proxy.config.log2.logfile_ perm` `STRING`	`rw-r--r--`	Specifies the log file permissions. The standard UNIX file permissions are used (owner, group, other). Valid values are: • - = no permission • r = read permission • w = write permission • x = execute permission Permissions are subject to the umask settings for the Websense Content Gateway process. This means that a umask setting of 002 will not allow write permission for others, even if specified in the configuration file. Permissions for existing log files are not changed when the configuration is changed. UNIX only.
`proxy.config.log2.squid_ log_enabled` `INT`	`1`	Enables (1) or disables (0) the squid log file format.
`proxy.config.log2.squid_ log_is_ascii` `INT`	`1`	Specifies the squid log file type: • 1 = ASCII • 0 = binary
`proxy.config.log2.squid_ log_name` `STRING`	`squid`	Specifies the squid log filename.

Configuration Variable Data Type	Default Value	Description
`proxy.config.log2.squid_` `log_header` `STRING`	`NULL`	Specifies the squid log file header text.
`proxy.config.log2.common_` `log_enabled` `INT`	`0`	Enables (1) or disables (0) the Netscape common log file format.
`proxy.config.log2.common_` `log_is_ascii` `INT`	`1`	Specifies the Netscape common log file type: • 1 = ASCII • 0 = binary
`proxy.config.log2.common_` `log_name` `STRING`	`common`	Specifies the Netscape common log filename.
`proxy.config.log2.common_` `log_header` `STRING`	`NULL`	Specifies the Netscape common log file header text.
`proxy.config.log2.` `extended_log_enabled` `INT`	`0`	Enables (1) or disables (0) the Netscape extended log file format.
`proxy.confg.log2.extended_` `log_is_ascii` `INT`	`1`	Specifies the Netscape extended log file type: • 1 = ASCII • 0 = binary
`proxy.config.log2.` `extended_log_name` `STRING`	`extended`	Specifies the Netscape extended log filename.
`proxy.config.log2.` `extended_log_header` `STRING`	`NULL`	Specifies the Netscape extended log file header text.
`proxy.config.log2.` `extended2_log_enabled` `INT`	`0`	Enables (1) or disables (0) the Netscape Extended-2 log file format.
`proxy.config.log2.` `extended2_log_is_ascii` `INT`	`1`	Specifies the Netscape Extended-2 log file type: • 1 = ASCII • 0 = binary
`proxy.config.log2.` `extended2_log_name` `STRING`	`extended2`	Specifies the Netscape Extended-2 log filename.
`proxy.config.log2.` `extended2_log_header` `STRING`	`NULL`	Specifies the Netscape Extended-2 log file header text.
`proxy.config.log2.` `separate_icp_logs` `INT`	`0`	When enabled (1), configures Websense Content Gateway to store ICP transactions in a separate log file.

Configuration Variable Data Type	Default Value	Description
proxy.config.log2. separate_host_logs INT	0	When enabled (1), configures Websense Content Gateway to create a separate log file for HTTP/FTP transactions for each origin server listed in the log_hosts.config file (see *HTTP host log splitting*, page 162).
proxy.local.log2. collation_mode INT	0	Specifies the log collation mode: • 0 = Collation disabled. • 1 = This host is a log collation server. • 2 = This host is a collation client and sends entries using standard formats to the collation server. For information on sending XML-based custom formats to the collation server, see *logs_xml.config*, page 272.
proxy.confg.log2. collation_host STRING	NULL	Specifies the hostname of the log collation server.
proxy.config.log2. collation_port INT	8085	Specifies the port used for communication between the collation server and client.
proxy.config.log2. collation_secret STRING	foobar	Specifies the password used to validate logging data and prevent the exchange of unauthorized information when a collation server is being used.
proxy.config.log2. collation_host_tagged INT	0	When enabled (1), configures Websense Content Gateway to include the hostname of the collation client that generated the log entry in each entry.
proxy.config.log2. collation_retry_sec INT	5	Specifies the number of seconds between collation server connection retries.
proxy.config.log2.rolling_ enabled INT	1	Enables (1) or disables (0) log file rolling. See *Rolling event log files*, page 158.
proxy.config.log2.rolling_ interval_sec INT	86400	Specifies the log file rolling interval, in seconds. The minimum value is 300 (5 minutes). The maximum value is 86400 seconds (one day).

Configuration Variable Data Type	Default Value	Description
`proxy.config.log2.rolling_ offset_hr` `INT`	0	Specifies the file rolling offset hour. The hour of the day that starts the log rolling period.
`proxy.config.log2.auto_ delete_rolled_files` `INT`	1	Enables (1) or disables (0) automatic deletion of rolled files.
`proxy.config.log2. sampling_frequency` `INT`	1	Configures Websense Content Gateway to log only a sample of transactions rather than every transaction. You can specify the following values: • 1 = log every transaction • 2 = log every second transaction • 3 = log every third transaction and so on...

URL remap rules

Configuration Variable Data Type	Default Value	Description
proxy.config.url_remap. default_to_server_pac_port INT	-1	Sets the PAC port so that PAC requests made to the Websense Content Gateway proxy service port are redirected to this port. -1 specifies that the PAC port will be set to the autoconfiguration port (the default autoconfiguration port is 8083). This is the default setting. This variable can be used together with the *proxy.config.url_remap.default_to_server_pac* variable to get a PAC file from a different port. You must create and run a process that serves a PAC file on this port; for example, if you create a Perl script that listens on port 9000 and writes a PAC file in response to any request, you can set this variable to 9000, and browsers that request the PAC file from a proxy server on port 8080 will get the PAC file served by the Perl script.
proxy.config.url_remap. remap_required INT	0	Set this variable to 1 if you want Websense Content Gateway to serve requests only from origin servers listed in the mapping rules of the remap.config file. If a request does not match, the browser will receive an error.
proxy.config.url_remap. pristine_host_hdr INT	0	Set this variable to 1 if you want to retain the client host header in a request during remapping.

ICP configuration

Configuration Variable Data Type	Default Value	Description
proxy.config.icp.enabled INT	0	Sets ICP mode for hierarchical caching: • 0 = disables ICP. • 1 = allows Websense Content Gateway to receive ICP queries only. • 2 = allows Websense Content Gateway to send and receive ICP queries. See *ICP cache hierarchies*, page 71.
proxy.config.icp.icp_ interface STRING	your_interface	Specifies the network interface used for ICP traffic.
proxy.config.icp.icp_port INT	3130	Specifies the UDP port that you want to use for ICP messages.
proxy.config.icp. multicast_enabled INT	0	Enables (1) or disables (0) ICP multicast.
proxy.config.icp.query_ timeout INT	2	Specifies the timeout used for ICP queries.

Scheduled update configuration

Configuration Variable Data Type	Default Value	Description
proxy.config.update. enabled INT	0	Enables (1) or disables (0) the Scheduled Update option.
proxy.config.update.force INT	0	Enables (1) or disables (0) a force immediate update. When enabled, Websense Content Gateway overrides the scheduling expiration time for all scheduled update entries and initiates updates until this option is disabled.
proxy.config.update.retry_ count INT	10	Specifies the number of times Websense Content Gateway retries the scheduled update of a URL in the event of failure.

Configuration Variable Data Type	Default Value	Description
`proxy.config.update.retry_interval` `INT`	2	Specifies the delay in seconds between each scheduled update retry for a URL in the event of failure.
`proxy.config.update.concurrent_updates` `INT`	100	Specifies the maximum simultaneous update requests allowed at any time. This option prevents the scheduled update process from overburdening the host.

SNMP configuration

Configuration Variable Data Type	Default Value	Description
`proxy.config.snmp.master_agent_enabled` `INT`	1	Enables (1) or disables (0) the SNMP agent.
`proxy.config.snmp.snmp_encap_enabled` `INT`	0	When enabled (1), the SNMP agent runs on the encapsulation port instead of the default port 161. When disabled (0), the SNMP agent can run only on the default port 161. Linux only.

WCCP configuration

Configuration Variable Data Type	Default Value	Description
`proxy.config.wccp.enabled` `INT`	0	Enables (1) or disables (0) WCCP.
`proxy.config.wccp.version` `INT`	1	Specifies the version of WCCP being used: • 1 = Version 1.0 • 2 = Version 2.0
WCCP 1.0 variables		
`proxy.config.wccp.router_ip` `STRING`	NULL	Specifies the IP address of the router sending traffic to Websense Content Gateway.

Configuration Variable Data Type	Default Value	Description

`proxy.config.wccp.`
`ethernet_interface`
`STRING`
	your_interface	Specifies the Ethernet interface used to talk to the WCCP 1.0 router.
		
		Note: The Websense Content Gateway installation script detects your Ethernet interface and sets this variable appropriately. If your system has multiple network interfaces, check that this variable specifies the correct interface.

WCCP 2.0 variables		

`proxy.config.wccp2.`
`security_enabled`
`INT`
	0	Enables (1) or disables (0) security so that the router and the the proxy can authenticate each other. (If you enable security in Websense Content Gateway, you must also enable security on the router. See your Cisco router documentation.)

`proxy.config.wccp2.`
`password`
`STRING`
	NULL	Specifies the password used for authentication. This must be the same password configured on the router. It must be at least seven characters long.

`proxy.config.wccp2.`
`multicast_enabled`
`INT`
	0	Enables (1) or disables (0) multicast mode.

`proxy.config.wccp2.`
`multicast_address`
`STRING`
	NULL	Specifies the IP multicast address.

`proxy.config.wccp2.`
`multicast_ttl`
`INT`
	1	Specifies the number of hops multicast traffic must travel to reach its destination. Assign a value to this variable if you anticipate that traffic will travel more than 3 hops.
		
		In addition, assign a value to this variable if your WCCP router is not directly connected to Websense Content Gateway.
		
		The default is 1.

`proxy.config.wccp2.rev_`
`encapsulation`
`INT`
	0	Enables (1) or disables (0) packet encapsulation mode, which enables Websense Content Gateway to send encapsulated returned (bypassed) packets to the router.

Configuration Variable Data Type	Default Value	Description

`proxy.config.wccp2.number_`
`of_routers`
`INT`
	`0`	If multicast is not enabled, the routers on your network are not automatically discovered. You must specify the number of routers that direct traffic to Websense Content Gateway. WCCP 2.0 supports a maximum of 32 routers.

`proxy.config.wccp2.`
`router0_ip`
`STRING`
`...`
`proxy.config.wccp2.`
`router10_ip`
`STRING`
	`NULL`	If multicast is not enabled, the routers on your network are not automatically discovered. You must specify the IP address of each router that directs traffic to the Websense Content Gateway server.

`proxy.config.wccp2.svc_`
`HTTP`
`INT`
	`1`	When enabled (1), a WCCP 2.0 router can route HTTP requests to Websense Content Gateway.

`proxy.config.wccp2.svc_FTP` `INT`	`0`	When enabled (1), a WCCP 2.0 router can route FTP requests from FTP clients to Websense Content Gateway.

`proxy.config.wccp2.svc_`
`SOCKS`
`INT`
	`0`	When enabled (1), a WCCP 2.0 router can route SOCKS traffic to Websense Content Gateway.

`proxy.config.wccp2.svc_DNS` `INT`	`0`	When enabled (1), a WCCP 2.0 router can route DNS requests to Websense Content Gateway.

`proxy.local.wccp2.`
`ethernet_interface`
`STRING`
	your_interface	Specifies the ethernet interface used to talk to the WCCP 2.0 router.
		
		Note: The Websense Content Gateway installation script detects your Ethernet interface and sets this variable appropriately. If your system has multiple network interfaces, check that this variable specifies the correct interface.

ARM (security configuration)

Configuration Variable Data Type	Default Value	Description

`proxy.config.arm.security_`
`enabled`
`INT`
	`0`	Enables (1) or disables (0) ARM security. See *Controlling host access to the proxy server*, page 102.

ICAP

Configuration Variable Data Type	Default Value	Description
`proxy.config.icap.enabled` `INT`	0	Enables (1) or disables (0) ICAP support with Websense Data Security Suite (DSS). See *Working With Websense Data Security Suite*, page 99.

`proxy.config.icap.ICAPUri`
`STRING`
	NULL	Specifies the Uniform Resource Identifier for the ICAP service. Obtain the identifier from your DSS administrator. Enter the URI in the following format:
		`icap://hostname:port/`
		`path`
		For *hostname*, enter the IP address or hostname of the DSS Protector appliance.
		The default ICAP port is 1344.
		Path is the path of the ICAP service on the host machine.
		For example:
		`icap://`
		`ICAP_machine:1344/opt/`
		`icap_services`
		You do not need to specify the port if you are using the default ICAP port 1344.

`proxy.config.icap.FailOpen` `INT`	1	Set to: • 0 to send a block page if Websense Content Gateway encounters an error while communicating with Data Security Suite. • 1 to allow traffic

`proxy.config.icap.`
`BlockHugeContent`
`INT`
	0	Set to:
		• 0 to send a block page if a file larger than the size limit specified in DSS is sent. The default size limit in DSS v6.5 is 12 MB.
		• 1 to allow traffic

`proxy.config.icap.`
`AnalyzeSecureContent`
`INT`
	1	Set to:
		• 0 if decrypted traffic should be sent directly to its destination.
		• 1 if decrypted traffic should be sent to Data Security Suite for analysis.

remap.config

The `remap.config` file contains mapping rules that Websense Content Gateway uses to redirect HTTP requests permanently or temporarily without Websense Content Gateway having to contact any origin server:

> **Important**
>
> After you modify this file, run `content_line -x` from the Websense Content Gateway **bin** directory (default location is **/opt/WCG/bin**) to apply the changes. When you apply the changes to a node in a cluster, Websense Content Gateway applies the changes to all nodes in the cluster.

Format

Each line in the **remap.config** file must contain a mapping rule. Websense Content Gateway recognizes three space-delimited fields: type, target, and replacement. The following table describes the format of each field.

Field	Description
type	Enter one of the following: • `redirect`—redirects HTTP requests permanently without having to contact the origin server. Permanent redirects notify the browser of the URL change (by returning an HTTP status code 301) so that the browser can update bookmarks. • `redirect_temporary`—redirects HTTP requests temporarily without having to contact the origin server. Temporary redirects notify the browser of the URL change for the current request only (by returning an HTTP status code 307).
target	Enter the origin or *from* URL. You can enter up to four components: *scheme://host:port/path_prefix* *scheme* can be `http,` `https,` or `ftp.`
replacement	Enter the destination or *to* URL. You can enter up to four components: *scheme://host:port/path_prefix* *scheme* can be `http,` `https,` or `ftp.`

Examples

The following section shows example mapping rules in the **remap.config** file.

Redirect mapping rules

The following rule *permanently* redirects all HTTP requests for `www.company.com` to `www.company2.com`:

```
redirect http://www.company.com http://www.company2.com
```

The following rule *temporarily* redirects all HTTP requests for `www.company1.com` to `www.company2.com`:

```
redirect_temporary http://www.company1.com http://
www.company2.com
```

snmpd.cnf

The **snmpd.cnf** file configures user access to MIB information and trap destinations. It is beyond the scope of this manual to describe all of the SNMP parameters and formats; only the major parameters affecting access control and trap destination are discussed in this section.

> ✓ **Note**
> Websense Content Gateway supports 64-bit values in SNMP. If you are using an SNMP monitoring tool that does not support SNMPv2c or SNMPv3, you may not be able to view all the Websense Content Gateway values.

> ❗ **Important**
> After you modify this file, you must restart the proxy.

Format

The **snmpd.cnf** file contains a list of configuration parameters. Lines beginning with the # symbol are comments. Each configuration parameter is listed along with formatting variables, as in the following example:

```
#Entry type: snmpNotifyEntry
#Format: snmpNotifyName (text)
# snmpNotifyTag (text) (keyed on snmpTargetAddr table)
# snmpNotifyType (trap(1), inform(2))
# snmpNotifyStorageType (nonVolatile, permanent, readOnly)
#snmpNotifyEntry 31 Console trap nonVolatile
#snmpNotifyEntry 32 TrapSink trap nonVolatile
```

Configuring trap destinations

You must modify the **snmpd.cnf** file to send traps to each of your monitoring stations.

You must configure the `snmpnotifyEntry` and `snmpTargetAddrEntry` entries for trap destinations. `snmpnotifyEntry` sends traps to a particular host or group of hosts. `snmpTargetAddrEntry` defines the IP addresses for a host or group of hosts.

For example, to send traps to a host named *host_a*, you need an `snmpnotifyEntry` line similar to the following:

```
snmpnotifyEntry 31 host_a trap nonVolatile
```

This line defines a trap destination named *host_a*, which can represent a single IP address or a group of IP addresses. In place of *host_a*, enter the name of the host or group of hosts to receive traps on your system. In place of *31*, enter a unique integer.

Then, for each IP address that you want to define for *host_a*, enter a `snmpTargetAddrEntry` line similar to the following. All trap messages destined for `host_a` are sent to the IP addresses defined in the `snmpTargetAddrEntry` lines of the **snmpd.cnf** file.

```
snmpTargetAddrEntry 34 snmpUDPDomain A.B.C.D:0 100 3 host_a
v1ExampleParams nonVolatile 255.255.255.255:0
```

In place of *34*, enter a unique integer. In place of `A.B.C.D`, enter the IP address that you want to define for *host_a*.

Access control

By default, read-only access is granted to any host that makes SNMP requests using the community string `public`. To restrict access, you must remove access-related default entries in the `snmpd.cnf` file and add entries specifying the hosts you want to allow. You must perform the following configuration:

◆ Define the hosts or host groups for your system (use the `snmpTargetAddrEntry` lines to define the IP addresses associated to each host or host group)

◆ Define access communities (a community can consist of a host or group of hosts); you must define hosts before you can define communities

◆ Give access to the communities that you want to have access; you must define communities to give them access

Examples

To restrict access, remove the following default **snmpd.cnf** entries, which allow access to any host:

```
vacmAccessEntry snmpv1 public Anyone nonVolatile
vacmAccessEntry snmpv2c public Anyone nonVolatile
snmpCommunityEntry t0000000 public public localSnmpID -
nonVolatile
```

To allow access to selected hosts, replace the deleted entries with the following. You can allow access to as many hosts as you want. You can configure one host at a time or one subnet at a time.

For example, suppose you want to allow the single host named `OneHost` to have access to MIB information. You would need the following lines in the `snmpd.cnf` file:

```
snmpTargetAddrEntry 33 snmpUDPDomain A.B.C.D:0 100 3 host_a
v1ExampleParams nonVolatile 255.255.255.255:0
snmpCommunityEntry localSnmpID public public localSnmpID
default host_a nonVolatile
vacmAccessEntry OneHost - snmpv1 noAuthNoPriv exact All -
All nonVolatile
vacmSecurityToGroupEntry snmpv1 public OneHost nonVolatile
```

The `snmpTargetAddrEntry` line defines the host, *host_a*, which has the IP address `A.B.C.D`. The `communityEntry` line defines the community `OneHost`, which contains the host `host_a`. The `vacmAccessEntry` and `vacmSecurityToGroupEntry` lines allow access to the community `OneHost`.

To allow MIB access to one subnet named `OneNet`, enter the following lines in the configuration file:

> ✓ **Note**
> Use the netmask `255.255.255.0` for the subnet `A.B.C.xxx` in the `snmpTargetAddrEntry` definition.

```
snmpTargetAddrEntry 34 snmpUDPDomain A.B.C.0:0 100 3 net_a
v1ExampleParams nonVolatile 255.255.255.0:0
```

```
communityEntry localSnmpID public OneNet localSnmpID default
net_a nonVolatile
```

```
vacmAccessEntry OneNet - snmpv1 no AuthNoPriv exact All -
All nonVolatile
```

```
vacmSecurityToGroupEntry snmpv1 public OneNet nonVolatile
```

The `snmpTargetAddrEntry` line defines the subnet, `net_a`, which has the IP address `A.B.C.xxx`. The `communityEntry` line defines the community `OneNet`, which contains the subnet `net_a`. The `vacmAccessEntry` and `vacmSecurityToGroupEntry` lines allow access to the community `OneNet`.

socks.config

The **socks.config** file specifies the following information:

◆ The SOCKS servers Websense Content Gateway passed through to access specific origin servers and the order in which the proxy goes through the SOCKS server list

> ✔ **Note**
> You can specify your *default* SOCKS servers in the Websense Content Manager or by editing the configuration variable *proxy.config.socks.socks.default_servers.* However, the **socks.config** file lets you perform additional SOCKS configuration; you can send requests to specific origin servers through specific SOCKS servers.

◆ The origin servers you want Websense Content Gateway to access directly *without* going through the SOCKS server

◆ The user name and password that Websense Content Gateway uses to connect to a SOCKS server (SOCKS version 5 only)

> ❗ **Important**
> After you modify this file, you must restart the proxy.

Format

To specify the SOCKS servers Websense Content Gateway must go through to reach specific origin servers, you must add a rule to the **socks.config** file with the following format:

```
dest_ip=ipaddress parent=server_name:port
[round_robin=value]
```

where:

ipaddress is the origin server IP address or range of IP addresses separated by – or /.

server_name is the hostname of the SOCKS server.

port is the port number through which the proxy communicates with the SOCKS server.

value is either strict if you want Websense Content Gateway to try the SOCKS servers one by one, or false if you do not want round-robin selection to occur.

To specify the origin servers you want Websense Content Gateway to access directly *without* going through the SOCKS server, enter a rule in the socks.config file in the following format:

```
no_socks ipaddress
```

where *ipaddress* is a comma-separated list of the IP addresses or IP address ranges associated with the origin servers you want Websense Content Gateway to access directly.

To specify the user name and password Websense Content Gateway uses for authentication with the SOCKS version 5 server, enter a rule in the `socks.config` file in the following format:

```
auth u username password
```

where *username* is the user name, and *password* is the password used for authentication.

> ✓ **Note**
> Each rule in the **socks.config** file can consist of a maximum of 400 characters. The order of the rules in the **socks.config** file is not significant.

Examples

The following example configures the proxy to send requests to the origin servers associated with the range of IP addresses 123.15.17.1 - 123.14.17.4 through the SOCKS server `socks1` on port 1080 and `socks2` on port 4080. Because the optional specifier `round_robin` is set to `strict`, the proxy sends the first request to `socks1`, the second request to `socks2`, the third request to `socks1`, and so on.

```
dest_ip=123.14.15.1 - 123.14.17.4
parent=socks1:1080;socks2:4080 round_robin=strict
```

The following example configures the proxy to access the origin server associated with the IP address 11.11.11.1 directly *without* going through the SOCKS server:

```
no_socks 11.11.11.1
```

The following example configures Websense Content Gateway to access the origin servers associated with the range of IP addresses 123.14.15.1 - 123.14.17.4 and the IP address 113.14.18.2 directly *without* going through the SOCKS server:

```
no_socks 123.14.15.1 - 123.14.17.4, 113.14.18.2
```

The following example configures Websense Content Gateway to use the user name `content_gateway` and the password `secret` for authentication with the SOCKS version 5 server:

```
auth u content_gateway secret
```

storage.config

The **storage.config** file lists all the files, directories, or hard disk partitions that make up the cache.

> **Important**
> After you modify this file, you must restart the proxy.

Format

The format of the **storage.config** file is:

    pathname size

where *pathname* is the name of a partition, directory, or file, and *size* is the size of the named partition, directory, or file, in bytes. You must specify a size for directories or files. For raw partitions, size specification is optional.

You can use any partition of any size. For best performance, the following guidelines are recommended:

◆ Use raw disk partitions.
◆ For each disk, make all partitions the same size.
◆ For each node, use the same number of partitions on all disks.

Specify pathnames according to your operating system requirements. See the following examples.

> **Important**
> In the **storage.config** file, a formatted or raw disk must be at least 128 MB.

Examples

The following basic example shows 64 MB of cache storage in the **/big_dir** directory:

    /big_dir 67108864

You can use the . symbol for the current directory. Here is an example of 64 MB of cache storage in the current directory:

    . 67108864

Linux example

The following example is for the Linux operating system:

```
/dev/raw_sdb 9105018880
```

update.config

The **update.config** file controls how Websense Content Gateway performs a scheduled update of specific local cache content. The file contains a list of URLs specifying objects that you want to schedule for update.

A scheduled update performs a local HTTP GET on the objects at the specific time or interval. You can control the following parameters for each specified object:

◆ The URL

◆ URL-specific request headers, which overrides the default

◆ The update time and interval

◆ The recursion depth

> **❗ Important**
>
> After you modify this file, run `content_line -x` from the Websense Content Gateway **bin** directory (default location is **/opt/WCG/bin**) to apply the changes. When you apply the changes to a node in a cluster, Websense Content Gateway applies the changes to all nodes in the cluster.

Scheduled update supports the following tag/attribute pairs when performing recursive URL updates:

◆ <a href=" ">

◆ <img src=" ">

◆ <img href=" ">

◆ <body background=" ">

◆ <frame src=" ">

◆ <iframe src=" ">

◆ <fig src=" ">

◆ <overlay src=" ">

◆ <applet code=" ">

◆ <script src=" ">

◆ <embed src=" ">

◆ <bgsound src=" ">

◆ <area href=" ">

◆ <base href=" ">

◆ <meta content=" ">

Scheduled update is designed to operate on URL sets consisting of hundreds of input URLs (expanded to thousands when recursive URLs are included); it is *not* intended to operate on massively large URL sets, such as those used by Internet crawlers.

Format

Each line in the `update.config` file uses the following format:

```
URL\request_headers\offset_hour\interval\recursion_depth\
```

The following table describes each field.

Field	Allowed Inputs
`URL`	HTTP and FTP-based URLs.
`request_headers`	(*Optional.*) A list of headers (separated by semi-colons) passed in each GET request. You can define any request header that conforms to the HTTP specification. The default is no request header.
`offset_hour`	The base hour used to derive the update periods. The range is 00-23 hours.
`interval`	The interval, in seconds, at which updates should occur, starting at offset hour.
`recursion_depth`	The depth to which referenced URLs are recursively updated, starting at the given URL.

Examples

The following example illustrates an HTTP scheduled update:

```
http://www.company.com\User-Agent: noname user
agent\13\3600\5\
```

This example specifies the URL and request headers, an offset hour of 13 (1 p.m.), an interval of one hour, and a recursion depth of 5. This would result in updates at 13:00, 14:00, 15:00, and so on. To schedule for an update to occur only once a day, use an interval value of 24 hours x 60 minutes x 60 seconds = 86400.

The following example illustrates an FTP scheduled update:

```
ftp://anonymous@ftp.company.com/pub/misc/
test_file.cc\\18\120\0\
```

This example specifies the FTP request, an offset hour of 18 (6 p.m.), and an interval of every two minutes. The user must be *anonymous* and the password must be specified by the variable *proxy.config.http.ftp.anonymous_passwd* in the **records.config** file.

congestion.config

The **congestion.config** file defines what to do when congestion control is on and a server is congested. You edit the file by going to **Configure > Networking > Connection Management > Congestion Control** and modifying the **Congestion Rules** on the screen.

Related topics:

Connection Management, page 238

Format

Each line in the **congestion.config** file consists of one or more sets of tag value pairs, preceded by one or more destination specifiers.

The format for each line is:

```
<dest_specifier_tag>=<value> [secondary_specifier_tag]
<rule1_tag>=<value> … <rule2_tag>=<value>
```

For example:

```
dest_domain=xyz.com max_connection_failures=3 snmp=off
dest_host=gumby port=80 fail_window=100 max_connection=10
```

Destination specifier tags

The following tags specify the destination host to be used for congestion control. You must include exactly one primary destination specifier on each line of the **congestion.config** file. You can include any number of secondary specifiers, but you cannot duplicate them on the same line.

Tag	Type	Description
dest_domain	Primary	The domain name of the destination host. For example: xyz.com
dest_host	Primary	The host name of the destination host. For example: gumby.
dest_ip	Primary	The IP address of the destination host. For example: 123.45.6.7.
dest_regex	Primary	A regular expression describing the destination host. For example: gumby|pokey.
port	Secondary	The port number to use to connect to the destination host. For example: 8877.
Prefix	Secondary	The prefix of the destination host. For example: //NIM1/.

Congestion control rule variables

The following variables or tags specify the rules to use for congestion control. Requests to a server that do not have an applicable rule in **congestion.config** use the values for the "origins server connect attempts" variables in **records.config**.

Variable	Input Parameter	Default Value	Description
max_connection_failures	INT	5	Specifies the maximum number of times that a connection can fail before no further attempts are made.
fail_window	INT	120	The period, in seconds, after which a server is marked congested if more than the specified number connection failures are experienced.
proxy_retry_interval	INT	10	The period, in seconds, after which the proxy should retry a congested server. If a server is marked congested, the proxy does not forward requests to it until Proxy Retry After Time (PRAT), which is current time + the proxy_retry_interval.
client_wait_interval	INT	300	The period, in seconds, the client should wait before retrying the request during the PRAT interval. The Client Retry After Time (CRAT) equals PRAT - current time + the client_wait_interval + a random integer from 0 to the wait_interval_alpha).
wait_interval_alpha	INT	30	The maximum client_wait_interval.
live_os_conn_timeout	INT	60	The number of seconds for each connection attempt before a timeout. If the proxy does not succeed, then one connection failure is counted towards the server.
live_os_conn_retries	INT	2	For each request to a live (non-congested) server, the proxy tries at most this number of times to connect to the server.
dead_os_conn_timeout	INT	15	The timeout, in seconds, before the proxy considers a connection attempt failed.
dead_os_conn_retries	INT	1	The number of times the proxy tries to connect to a congested server after its PRAT time. A congested server will stay dead if the proxy cannot make a successful connection; otherwise, the server becomes live again

Variable	Input Parameter	Default Value	Description
max_connection	INT	-1	The maximum number of connections allowed to the proxy. "-1" means unlimited. Websense Content Gateway temporarily marks a server as congested if a max_connection number to the server is reached. If a new client requests a new connection to the server, the client receives the message, "503 Retry-After". There is no PRAT on the servers that reach their max_connection.
error_page	Page URL	"congestion #retryAfter"	The error that is returned when a connection is found to be congested.
congestion_scheme	per_ip per_host	per_ip	The congestion scheme to use to identify the server. Options include "per_ip" or "per_host". If a domain has 2 IP address and per_ip is used, then each IP address has its own number of connection failures, and each IP address is marked either congested or uncongested. If per_host is used, then one connection failure is counted for the entire host. If the host is marked congested, both IPs are marked congested.
snmp	on off	on	Specifies whether Simple Network Management Protocol (SNMP) traps are on or off. Traps are used to report an alert or other asynchronous event involving the proxy.

F | The req_ca.cnf File

Create a **req_ca.cnf** file and copy the code below into that file. See *Creating a sub-certificate authority*, page 127 for information on the **req_ca.cnf** file.

```
#
# Configuration file for generating a CA Request
#
HOME = .
RANDFILE = $ENV::HOME/.rnd
#
[ policy_match ]
countryName = match
stateOrProvinceName = match
organizationName = match
organizationalUnitName = optional
commonName = supplied
emailAddress = optional
#
[ policy_anything ]
countryName = optional
stateOrProvinceName = optional
localityName = optional
organizationName = optional
organizationalUnitName = optional
commonName = supplied
emailAddress = optional
##################################################################
#
[ req ]
default_bits = 1024
default_keyfile = privkey.pem
distinguished_name = req_distinguished_name
string_mask = nombstr
req_extensions = v3_req # The extensions to add to a certificate
request
[ req_distinguished_name ]
countryName = Country Name (2 letter code)
countryName_default = US
countryName_min = 2
countryName_max = 2
stateOrProvinceName = State or Province Name (full name)
stateOrProvinceName_default = Some-State
localityName = Locality Name (eg, city)
0.organizationName = Organization Name (eg, company)
0.organizationName_default = Internet Widgits Pty Ltd
#organizationalUnitName = Organizational Unit Name (eg, section)
```

```
#organizationalUnitName_default =
commonName = Common Name (Name of Sub-CA)
commonName_max = 64
emailAddress = Email Address
emailAddress_max = 64
[ v3_req ]
# Extensions to add to a certificate request to make it a CA
basicConstraints=CA:TRUE
nsCertType = sslCA
keyUsage = cRLSign, keyCertSign
```

G | Error Messages

Websense Content Gateway error messages

The following table lists messages that can appear in system log files. This list is not exhaustive; it describes warning messages that can occur and might require your attention. For information about warning messages not included in the list below, go to www.websense.com and then navigate to Support and Knowledge Base.

Process fatal

Message	Description
accept port is not between 1 and 65535. Please check configuration	The port specified in the records.config file that accepts incoming HTTP requests is not valid.
self loop is detected in parent proxy configuration	The name and port of the parent proxy are the same as that of Websense Content Gateway. This creates a loop when Websense Content Gateway attempts to send the request to the parent proxy.

Warnings

Message	Description
Logfile error: *error_number*	Generic logging error.
Bad cluster major version range *version1-version2* for node *IP address* connect failed	Incompatible software versions causing a problem.
can't open config file *filename* for reading custom formats	Custom logging is enabled, but Websense Content Gateway cannot find the **logs.config** file.

Message	Description
connect by disallowed client *IP address*, closing	The specified client is not allowed to connect to Websense Content Gateway. The client IP address is not listed in the **ip_allow.config** file.
Could not rename log *filename* to *rolled filename*	System error when renaming log file during roll.
Did *this_amount* of backup still to do *remaining_amount*	Congestion is approaching.
Different clustering minor versions *version 1, version 2* for node *IP address* continuing	Incompatible software versions causing a problem.
log format symbol *symbol_name* not found	Custom log format references a field symbol that does not exist. See *Event Logging Formats*, page 249.
missing field for field marker	Error reading a log buffer.
Unable to accept cluster connections on port: *cluster_port_number*	Contact Websense Technical Support. Go to www.websense.com/SupportPortal/default.aspx for Technical Support contact information
Unable to open log file *filename*, errno=*error_number*	Cannot open the log file.
Error accessing disk *disk_name*	Websense Content Gateway might have a cache read problem. You might have to replace the disk.
Too many errors accessing disk *disk_name*: declaring disk bad	Websense Content Gateway is not using the cache disk because it encountered too many errors. The disk might be corrupt and might have to be replaced.
No cache disks specified in **storage.config** file: cache disabled	The Websense Content Gateway **storage.config** file does not list any cache disks. Websense Content Gateway is running in proxy-only mode. You must add the disks you want to use for the cache to the **storage.config** file (see *storage.config*, page 336).

Alarm messages

The following table describes alarm messages that you may see in Websense Content Manager.

Message	Description/Solution
[Rollback::Rollback] Config file is read-only: *filename*	Go to the Websense Content Gateway **config** directory (default location is **/opt/WCG/config**) and check the indicated file permissions; change them if necessary.
[Rollback::Rollback] Unable to read or write config file *filename*	Go to the Websense Content Gateway **config** directory and make sure the indicated file exists. Check its permissions and change them if necessary.
[Websense Content Manager] Configuration File Update Failed *error_number*	Go to the Websense Content Gateway **config** directory and check the indicated file permissions; change them if necessary.
[Websense Content Manager] Websense Content Gateway process was reset	An informational message informing you that the content_gateway process was down.
Access logging suspended - configured space allocation exhausted.	The space allocated to the event log files is full. You must either increase the space or delete some log files to enable access logging to continue. To prevent this from happening, consider rolling log files more frequently and enabling the autodelete feature. See *Rolling event log files*, page 158.
Access logging suspended - no more space on the logging partition.	The entire partition containing the event logs is full. You must delete or move some log files to enable access logging to continue. To prevent this from happening, consider rolling log files more frequently and enabling the autodelete feature. See *Rolling event log files*, page 158.
Created zero length place holder for config file *filename*	Go to the Websense Content Gateway **config** directory and check the indicated file. If it is indeed zero in length, use a backup copy of the configuration file.
Websense Content Gateway can't open *filename* for reading custom formats	Make sure that the *proxy.config.log2.config_file* variable in the **records.config** file contains the correct path to the custom log configuration file (the default is **logging/logs.config**).
Websense Content Gateway could not open logfile *filename*	Check permissions for the indicated file and the logging directory.

Message	Description/Solution
Websense Content Gateway failed to parse line *line_number* of the logging config file *filename*	Check your custom log configuration file. There may be syntax errors. See *Custom logging fields*, page 249, for correct custom log format fields.
vip_config binary is not setuid root, manager will be unable to enable virtual ip addresses	The content_manager process is not able to set virtual IP addresses. You must setuid root for the **vip_config** file in the Websense Content Gateway **bin** directory.
Websense Content Gateway cannot parse the ICAP URI. Please ensure that the URI is entered correctly in Websense Content Manager or in the *proxy.config.icap. ICAPUri* configuration variable.	The Universal Resource Identifier (URI) is not in the correct format. Enter the URI as follows: icap://hostname:port/path See *Working With Websense Data Security Suite*, page 99 for additional details on the format of the URI.
The specified ICAP server does not have a DNS entry. Please ensure that a valid DSS hostname is entered correctly in Websense Content Manager or in the *proxy.config.icap. ICAPUri* configuration variable.	The hostname in the **records.config** file does not match any entries in the DNS. Ensure that the name of a valid Websense Data Security Suite server is entered correctly in Websense Content Manager. See *Working With Websense Data Security Suite*, page 99 for information on the format of the URI.
Websense Content Gateway is not able to communicate with the DSS server. Please try again.	Ensure that the Websense Data Security Suite server is up and running, and accepting connections on the port specified in the *proxy.config.icap.ICAPUri* variable. Contact your Websense Data Security Suite administrator if this message persists.

HTML messages sent to clients

Websense Content Gateway returns detailed error messages to browser clients when there are problems with the HTTP transactions requested by the browser. These response messages correspond to standard HTTP response codes, but provide more information. A list of the more frequently encountered HTTP response codes is provided in *Standard HTTP response messages*, page 352. You can customize the response messages.

The following table lists the Websense Content Gateway hard-coded HTTP messages, their corresponding HTTP response codes, and their corresponding customizable files.

Title	HTTP Code	Description	Customizable Filename
Access Denied	403	You are not allowed to access the document at location *URL*.	access#denied
Bad HTTP request for FTP Object	400	Bad HTTP request for FTP object.	ftp#bad_request
Cache Read Error	500	Error reading from cache. Please retry request.	cache#read_error
Connection Timed Out	504	Server has not sent any data for too long a time.	timeout#inactivity
Content Length Required	400	Could not process this request because no Content-Length was specified.	request#no_content_length
Cycle Detected	400	Your request is prohibited because it would cause an HTTP proxy cycle.	request#cycle_detected
Forbidden	403	*port_number* is not an allowed port for SSL connections. (You have made a request for a secure SSL connection to a forbidden port number.)	access#ssl_forbidden
FTP Authentication Required	401	You need to specify a correct user name and password to access the requested FTP document *URL*.	ftp#auth_required
FTP Connection Failed	502	Could not connect to the server *server_name*.	connect#failed_connect
FTP Error	502	The FTP server *server_name* returned an error. The request for document *URL* failed.	ftp#error

Title	HTTP Code	Description	Customizable Filename
Host Header Required	400	An attempt was made to transparently proxy your request, but this attempt failed because your browser did not send an HTTP Host header. Manually configure your browser to use https:// *proxy_name:proxy_port* as an HTTP proxy. See your browser's documentation for details. Alternatively, end users can upgrade to a browser that supports the HTTP Host header field.	interception#no_host
Host Header Required	400	Your browser did not send a Host HTTP header field and therefore the virtual host being requested could not be determined. To access this Web site correctly, you will need to upgrade to a browser that supports the HTTP Host header field.	request#no_host
HTTP Version Not Supported	505	The origin server *server_name* is using an unsupported version of the HTTP protocol.	response#bad_version
Invalid HTTP Request	400	Could not process this *client_request* HTTP method request for *URL*.	request#syntax_error
Invalid HTTP Response	502	The host *server_name* did not return the document *URL* correctly.	response#bad_response
Malformed Server Response	502	The host *server_name* did not return the document *URL* correctly.	response#bad_response
Malformed Server Response Status	502	The host *server_name* did not return the document *URL* correctly.	response#bad_response
Maximum Transaction Time exceeded	504	Too much time has passed transmitting document *URL*.	timeout#activity
No Response Header From Server	502	The host *server_name* did not return the document *URL* correctly.	response#bad_response

Title	HTTP Code	Description	Customizable Filename
Not Cached	504	This document was not available in the cache, and you (the client) accept cached copies only.	cache#not_in_cache
Not Found on Accelerator	404	The request for *URL* on host *server_name* was not found. Check the location and try again.	urlrouting#no_mapping
NULL	502	The host *hostname* did not return the document *URL* correctly.	response#bad_response
Proxy Authentication Required	407	Please log in with user name and password.	access#proxy_auth_required
Server Hangup	502	The server *hostname* closed the connection before the transaction was completed.	connect#hangup
Temporarily Moved	302	The document you requested, *URL*, has moved to a new location. The new location is *new_URL*.	redirect#moved_temporarily
Transcoding Not Available	406	Unable to provide the document *URL* in the format requested by your browser.	transcoding#unsupported
Tunnel Connection Failed	502	Could not connect to the server *hostname*.	connect#failed_connect
Unknown Error	502	The host *hostname* did not return the document *URL* correctly.	response#bad_response
Unknown Host	500	Unable to locate the server named *hostname*. The server does not have a DNS entry. Perhaps there is a misspelling in the server name or the server no longer exists. Double-check the name and try again.	connect#dns_failed
Unsupported URL Scheme	400	Cannot perform your request for the document *URL* because the protocol scheme is unknown.	request#scheme_unsupported

Standard HTTP response messages

The following standard HTTP response messages are provided for your information. For a more complete list, see the *Hypertext Transfer Protocol — HTTP/1.1 Specification*.

Message	Description
200	OK
202	Accepted
204	No Content
206	Partial Content
300	Multiple Choices
301	Moved Permanently
302	Found
303	See Other
304	Not Modified
400	Bad Request
401	Unauthorized; retry
403	Forbidden
404	Not Found
405	Method Not Allowed
406	Not acceptable
408	Request Timeout
500	Internal server error
501	Not Implemented
502	Bad Gateway
504	Gateway Timeout

H | FAQs and Troubleshooting Tips

Frequently Asked Questions (FAQs)

- *How do you create a raw disk for the cache if all your disks have mounted file systems?*, page 353
- *How do disk I/O errors affect the cache and what does Websense Content Gateway do when a cache disk fails?*, page 354
- *If a client disconnects during the time that Websense Content Gateway is downloading a large object, is any of the object saved in the cache?*, page 354
- *Can Websense Content Gateway cache Java applets, JavaScript programs, or other application files like VBScript?*, page 354
- *How do you access Websense Content Manager if you forget the master administrator password?*, page 354
- *How do you apply changes to the logs_xml.config file to all nodes in a cluster?*, page 355
- *In Squid- and Netscape-format log files, what do the cache result codes mean?*, page 356
- *What does the cqtx field record in a custom log file?*, page 357
- *Does Websense Content Gateway refresh entries in its host database after a certain period of time if they have not been used?*, page 357
- *Can you improve the look of your custom response pages by using images, animated gifs, and Java applets?*, page 358
- *How do you configure Websense Content Gateway to serve only transparent requests?*, page 358

See *Troubleshooting tips*, page 359 for additional information.

How do you create a raw disk for the cache if all your disks have mounted file systems?

1. Enter the following command at the prompt to examine which file systems are mounted on the disk you want to use for the proxy cache:

   ```
   df -k
   ```

2. Open the **/etc/vfstab** file (**fstab** in Linux) and comment out or delete the file system entries for the disk.

3. Save and close the file.

4. Enter the following command for each file system you want to unmount:

   umount *file_system*

   where *file_system* is a file system you want to unmount.

5. Install Websense Content Gateway. When the installation script prompts you for a cache disk, select the raw disk you just created.

How do disk I/O errors affect the cache and what does Websense Content Gateway do when a cache disk fails?

If a disk drive fails five successive I/O operations, Websense Content Gateway considers the drive inaccessible and removes the whole disk from the cache. Normal cache operation continues on all other Websense Content Gateway disk drives.

If a client disconnects during the time that Websense Content Gateway is downloading a large object, is any of the object saved in the cache?

When a client disconnects during an HTTP or FTP operation, Websense Content Gateway continues to download the object from the origin server for up to 10 seconds. If the transfer from the origin server completes successfully within 10 seconds after the client disconnect, Websense Content Gateway stores the object in the cache. If the origin server download does not complete successfully within 10 seconds, Websense Content Gateway disconnects from the origin server and deletes the object from the cache. Websense Content Gateway does not store partial documents in the cache.

Can Websense Content Gateway cache Java applets, JavaScript programs, or other application files like VBScript?

Websense Content Gateway can store and serve Java applets, JavaScript programs, VBScripts, and other executable objects from its cache according to the freshness and cacheability rules for HTTP objects.

Websense Content Gateway does not execute the applets, scripts, or programs. These objects run only when the client system that sent the request loads them.

How do you access Websense Content Manager if you forget the master administrator password?

During installation, you can specify an administrator password. The installer automatically encrypts the password and stores the encryptions in the

records.config file. Each time you change passwords in Websense Content Manager, Websense Content Gateway updates the **records.config** file.

If you forget the administrator password and cannot access Websense Content Manager, you can clear the current password in the **records.config** file (set the value of the configuration variable to NULL) and then enter a new password in Websense Content Manager. You cannot set passwords in the **records.config** file because the password variables can contain only password encryptions or the value NULL.

1. Open the **records.config** file located in the Websense Content Gateway **config** directory (default location is **/opt/WCG/config)**.

2. Set the variable *proxy.config.admin.admin_password* to NULL to leave the password blank.

> ✓ **Note**
>
> Ensure that there are no trailing spaces after the word NULL.

3. Save and close the file.

4. From the Websense Content Gateway **bin** directory (default location is **/opt/WCG/bin)**, run content_line -x to apply the changes.

5. Log on to Websense Content Manager. When prompted for the user name and password, enter the administrator ID and leave the password entry blank.

    Because you have already cleared the password in the **records.config** file, you do not need a password to log on as the administrator.

6. Navigate to the **Configure > My Proxy > UI Setup > Login** tab.

7. In the **Administrator** section, leave the Old Password field empty. Type the new password in the **New Password** field, and then retype the new password in the **New Password (Retype)** field.

8. Click **Apply**.

    The next time you access Websense Content Manager, you must use the new password.

How do you apply changes to the logs_xml.config file to all nodes in a cluster?

After you modify the logs_xml.config file on one Websense Content Gateway node, enter the following command from the Websense Content Gateway **bin** directory (default location is **/opt/WCG/bin)**:

    content_line -x

Websense Content Gateway applies the changes to all nodes in the cluster. The changes take effect immediately.

In Squid- and Netscape-format log files, what do the cache result codes mean?

The following table describes the cache result codes in the Squid and Netscape log files.

Cache Result Code	Description
TCP_HIT	Indicates that a valid copy of the requested object was in the cache and that the proxy sent the object to the client.
TCP_MISS	Indicates that the requested object was not in the cache and that the proxy retrieved the object from the origin server or from a parent proxy and sent it to the client.
TCP_REFRESH_HIT	Indicates that the object was in the cache but was stale. Websense Content Gateway made an `if-modified-since` request to the origin server and the origin server sent a `304 not-modified` response. The proxy sent the cached object to the client.
TCP_REF_FAIL_HIT	Indicates that the object was in the cache but was stale. Websense Content Gateway made an `if-modified-since` request to the origin server but the server did not respond. The proxy sent the cached object to the client.
TCP_REFRESH_MISS	Indicates that the object was in the cache but was stale. Websense Content Guitar made an `if-modified-since` request to the origin server and the server returned a new object. The proxy served the new object to the client.
TCP_CLIENT_REFRESH	Indicates that the client issued a request with a `no-cache` header. The proxy obtained the requested object from the origin server and sent a copy to the client. Websense Content Gateway deletes any previous copy of the object from the cache.
TCP_IMS_HIT	Indicates that the client issued an `if-modified-since` request and the object was in the cache and fresher than the IMS date, or an `if-modified-since` to the origin server found that the cache object was fresh. The proxy served the cached object to the client.
TCP_IMS_MISS	Indicates that the client issued an `if-modified-since` request and the object was either not in cache or was stale in cache. The proxy sent an `if-modified-since` request to the origin server and received the new object. The proxy sent the updated object to the client.
TCP_SWAPFAIL	Indicates that the object was in the cache but could not be accessed. The client did not receive the object.

Cache Result Code	Description
ERR_CLIENT_ABORT	Indicates that the client disconnected before the complete object was sent.
ERR_CONNECT_FAIL	Indicates that Websense Content Gateway could not reach the origin server.
ERR_DNS_FAIL	Indicates that the Domain Name Server could not resolve the origin server name, or that no Domain Name Server could be reached.
ERR_INVALID_REQ	Indicates that the client HTTP request was invalid. Websense Content Gateway forwards requests with unknown methods to the origin server.
ERR_READ_TIMEOUT	Indicates that the origin server did not respond to the Websense Content Gateway request within the timeout interval.
ERR_PROXY_DENIED	Indicates that client service was denied by access control configuration.
ERR_UNKNOWN	Indicates that the client connected but subsequently disconnected without sending a request.

What does the cqtx field record in a custom log file?

The `cqtx` field records the complete client request text (minus headers) in the log file. For example, `get http://www.company.com HTTP/1.0`.

Does Websense Content Gateway refresh entries in its host database after a certain period of time if they have not been used?

By default, the Websense Content Gateway host database observes the time-to-live (ttl) values set by name servers. You can reconfigure Websense Content Gateway to a different value.

1. Open the **records.config** file located in the Websense Content Gateway **config** directory (default location is **/opt/WCG/config**).

2. Edit the following variable:

Variable	Description
`proxy.config.hostdb.ttl_mode`	Set to `0` to obey the ttl values set by the name servers `1` to ignore the ttl values set by name servers and use the value set by the Websense Content Gateway configuration variable `proxy.config.hostdb.timeout`. Set this variable to a value appropriate for your environment. `2` to use the lower of the two values (the one set by the name server or the one set by Websense Content Gateway) `3` to use the higher of the two values (the one set by the name server or the one set by Websense Content Gateway)

3. Save and close the file.
4. From the Websense Content Gateway **bin** directory (default location is **/opt/WCG/bin**), run `content_line -x` to apply the configuration changes.

Can you improve the look of your custom response pages by using images, animated gifs, and Java applets?

Websense Content Gateway can respond to clients only with a single text or HTML document. However, you can provide references on your custom response pages to images, animated gifs, Java applets, or objects other than text that are located on a Web server.

Add links in the **body_factory** template files in the same way you do for any image in an HTML document, with the full URL in the SRC attribute.

It is recommended that you do not run the Web server and Websense Content Gateway on the same system, to prevent both programs from trying to serve documents on the same port number.

How do you configure Websense Content Gateway to serve *only* transparent requests?

You can configure Websense Content Gateway to serve *only* transparent requests and prevent explicit proxy requests from being served in the following ways:

◆ You can control client access to Websense Content Gateway from the **ip_allow.config** file by specifying ranges of IP addresses that are allowed to use the proxy cache. If Websense Content Gateway receives a request from an IP address not listed in a range specified in the file, it discards the request. See *ip_allow.config*, page 269.

◆ If you do not know the ranges of client IP addresses allowed to access Websense Content Gateway, you can add rules to the **ipnat.conf** file so that only requests that have been redirected by your Layer 4 switch or WCCP router reach the proxy port. To make a transparent-only Websense Content Gateway server, add rules in the **ipnat.conf** file before the normal redirect service rule to redirect explicit proxy traffic to a port on which no service is listening. For example, if you want Websense Content Gateway to ignore explicit HTTP requests, add rules above the normal HTTP redirect rule in the **ipnat.conf** file as shown below (where *ipaddress* is the IP address of your Websense Content Gateway system and *port_number* is a port number on which no service is listening):

> rdr hme0 *ipaddress* port 80 -> *ipaddress* port *port_number* tcp
>
> rdr hme0 *ipaddress* port 8080 -> *ipaddress* port *port_number* tcp
>
> rdr hme0 0.0.0.0/0 port 80 -> *ipaddress* port 8080 tcp

Add equivalent rules to the **ipnat.conf** file for each protocol service port or separate network interface to be served. After you make changes to the **ipnat.conf** file, you must restart the proxy.

◆ If your Websense Content Gateway system has multiple network interfaces or if you configure the Websense Content Gateway operating system to use virtual IP addresses, you can give Websense Content Gateway two IP addresses. One address must be the *real* address that the proxy uses to communicate with origin servers and the other a private IP address (for example 10.0.0.1) for WCCP or switch redirection. After you configure the IP addresses, you must add the following variables to the end of the **records.config** file. Replace *private_ipaddress* with the private IP address used for WCCP or switch redirection and *real_ipaddress* with the IP address the proxy uses to communicate with origin servers.

```
LOCAL proxy.local.incoming_ip_to_bind STRING
private_ipaddress
LOCAL proxy.local.outgoing_ip_to_bind STRING
real_ipaddress
```

Troubleshooting tips

- *Web browsers may display an error document with a data missing message*, page 362
- *Websense Content Gateway does not resolve any Web sites*, page 362
- *Maximum document size exceeded message in the system log file*, page 362
- *DrainIncomingChannel message in the system log file*, page 363
- *No cop file message in the system log file*, page 363
- *Warning in system log file when editing vaddrs.config (Linux)*, page 364
- *Non transparent requests fail after enabling always_query_destination*, page 364
- *Websense Content Gateway is running but no log files are created*, page 364
- *Websense Content Gateway error indicates too many network connections*, page 365
- *Low memory symptoms*, page 366
- *Connection timeouts with the origin server*, page 366
- *IBM Web servers do not work with Websense Content Gateway*, page 366
- *Websense Content Gateway does not start (or stop)*, page 367

When you clear the cache, the statistics in Websense Content Manager do not reset to zero

The clear cache command (`content_gateway -Cclear`) does not reset the statistics to zero in Websense Content Manager. To reset the statistics, follow the procedure below.

1. Access Websense Content Manager.
2. Add **clear_stats.html** to the URL in your browser window and press **Return**. For example:

```
http://proxy1:8081/clear_stats.html
```

Websense Content Manager statistics should all show zero.

The throughput statistic is inaccurate in Websense Content Manager

Websense Content Gateway updates the throughput statistic after it has transferred an entire object. For larger files, the byte count increases sharply at the end of a transfer. The complete number of bytes transferred is attributed to the last 10-second interval, although it can take several minutes to transfer the object.

This inaccuracy is more noticeable with a light load. A heavier load yields a more accurate statistic.

You are unable to execute Websense Content Gateway commands

Commands do not execute under the following conditions:

◆ If the `content_manager` process is not running.

Check if the `content_manager` process is running by entering the following command:

```
ps aux | grep content_manager
```

or

```
./WCGAdmin status
```

If the `content_manager` process is not running, enter the following command from the Websense Content Gateway **bin** directory (default location is **/opt/WCG/bin**) to start it:

```
./content_manager
```

> ❗ **Important**
>
> If you must stop Websense Content Gateway, it is recommended that you restart it using **./WCGAdmin.** Stop it with **./WCGAdmin stop** and start it with **./WCGAdmin start** to ensure that all the processes stop and start correctly. See *Getting Started*, page 9.

◆ If you are not executing the command from **$WCGHome/bin**.

If the Websense Content Gateway **bin** directory is not in your path, prepend the commands with `./` (for example, `./content_line -h`).

◆ If multiple Websense Content Gateway installations are present and you are not executing the command from the active path specified in **/etc/content_gateway**.

Always change to the correct directory by issuing the command:

```
cd `cat /etc/content_gateway`/bin
```

You observe inconsistent behavior when one node obtains an object from another node in the cluster

As part of the system preparation process, you must synchronize the clocks on all the nodes in your cluster. Minor time differences cause no problems, but differences of more than a few minutes can affect Websense Content Gateway operation.

It is recommended that you run a clock synchronization daemon such as `xntpd`. You can obtain the latest version of `xntpd` from the following URL:

http://www.ntp.org

Web browsers may display an error document with a data missing message

A message similar to the following displays in Web browsers:

```
Data Missing
This document resulted from a POST operation and has
expired from the cache. If you wish you can repost the
form data to re-create the document by pressing the
reload button.
```

Web browsers maintain their local cache in memory and/or disk on the client system. Browser messages about documents that have expired from cache see the browser local cache, *not* the Websense Content Gateway cache. There is no Websense Content Gateway message or condition that can cause such messages to appear in a Web browser.

For information about browser cache options and effects, see the browser documentation.

Websense Content Gateway does not resolve any Web sites

The browser indicates that it is contacting the host and then times out with the following message:

```
The document contains no data; Try again later, or contact
the server's Administrator....
```

Make sure that the system is configured correctly and that Websense Content Gateway can read the name resolution file:

◆ Check if the server can resolve DNS lookups by issuing the `nslookup` command. For example:

```
nslookup www.myhost.com
```

◆ Check if the **/etc/resolv.conf** file contains the valid IP address of your DNS server(s).

◆ On some systems, if the **/etc/resolv.conf** file is unreadable or has no name server entry, the operating system will use localhost as a name server. However, Websense Content Gateway does not use this convention. If you want to use localhost as a name server, you must add a name server entry for 127.0.0.1 or 0.0.0.0 in the **/etc/resolv.conf** file.

◆ Check that the Websense Content Gateway user account has permission to read the **/etc/resolv.conf** file. Change the file permissions to `rw-r--r-- (644)`.

Maximum document size exceeded message in the system log file

The following message appears in the system log file.

```
WARNING: Maximum document size exceeded
```

A requested object was larger than the maximum size allowed in the proxy cache. Websense Content Gateway provided proxy service for the oversized object but did not cache it.

You can set the object size limit for the cache by modifying the **Maximum Object Size** field on the **Configure > Subsystems > Cache . General** tab. If you do not want to limit the size of objects in the cache, set the document size to 0 (zero).

DrainIncomingChannel message in the system log file

The following messages appear in the system log file:

```
Feb 20 23:53:40 louis content_manager[4414]: ERROR ==>
[drainIncomingChannel] Unknown message: 'GET http://
www.telechamada.pt/ HTTP/1.0'
Feb 20 23:53:46 louis last message repeated 1 time
Feb 20 23:53:58 louis content_manager[4414]: ERROR ==>
[drainIncomingChannel] Unknown message: 'GET http://
www.ip.pt/ HTTP/1.0'
```

These error messages indicate that a browser is sending HTTP requests to one of the Websense Content Gateway cluster ports, either rsport (default port 8087) or mcport (default port 8088). Websense Content Gateway discards the request. This error does not cause any Websense Content Gateway problems. The browser must be reconfigured to use the correct proxy port.

> ✓ **Note**
> Websense Content Gateway clusters work best when configured to use a separate network interface and cluster on a private subnet so that client machines have no access to the cluster ports.

No cop file message in the system log file

The following message appears repeatedly in the system log file:

```
content_cop[16056]: encountered "config/internal/no_cop"
file...exiting
```

The file **config/internal/no_cop** acts as an administrative control that instructs the content_cop process to exit immediately without starting content_manager or performing any health checks. The **no_cop** file prevents the proxy from starting automatically when it has been stopped with the ./WCGAdmin stop or the stop_content_gateway commands. Without such a static control, Websense Content Gateway would restart automatically upon system reboot. The no_cop control keeps Websense Content Gateway off until it is restarted with the ./WCGAdmin start or the start_content_gateway command.

The Websense Content Gateway installation script creates a **no_cop** file so that Websense Content Gateway does not start automatically. After you have completed

installation and configuration, and have rebooted the operating system, use the `./WCGAdmin start` or the `start_content_gateway` command to start Websense Content Gateway. See *Getting Started*, page 9, for information on starting and stopping Websense Content Gateway.

Warning in system log file when editing vaddrs.config (Linux)

If you edit the `vaddrs.config` file on a Linux system as a non-root user, Websense Content Gateway issues a warning message in the system log file similar to the following:

```
WARNING: interface is ignored: Operation not permitted.
```

You can ignore this message. Websense Content Gateway does apply your configuration edits.

> **Important**
> It is recommended that you always configure virtual IP addresses from Websense Content Manager. Editing the **vaddrs.config** file can lead to unpredictable results.

Non transparent requests fail after enabling always_query_destination

The variable proxy.config.arm.always_query_dest in the `records.config` file configures Websense Content Gateway in transparent mode to ignore host headers and always ask for the IP address of the origin server. When you enable this variable, Websense Content Gateway obtains the origin server's IP address from the existing NAT map list rather than trying to resolve the destination host name with a DNS lookup. As a result, logged URLs contain only IP addresses, not host names. To log domain names, set *proxy.config.arm.always_query_dest* to 0. However, setting *proxy.config.arm.always_query_dest* to 0 does not reduce the number of DNS lookups.

However, explicit requests (non transparent requests, including requests on port 80) fail, as there is no matching map in the NAT list.

> **Note**
> The `always_query_destination` option works only on the primary proxy port.

Websense Content Gateway is running but no log files are created

Websense Content Gateway writes event log files only when there is information to record. If Websense Content Gateway is idle, there may be no log files.

Ensure that you are looking in the correct directory. By default, Websense Content Gateway creates log files in its **logs** directory. Check the location of the log files in Websense Content Manager by examining the **Log Directory** field on the **Configure > Subsystems > Logging > General** tab. Alternatively, you can check the value of the variable *proxy.config.log2.logfile_dir* in the **records.config** file.

Check that the log directory has read/write permissions for the Websense Content Gateway user account. If the log directory does not have the correct permissions, the content_gateway process is unable to open or create log files.

Check that logging is enabled. In Websense Content Manager, examine the **Logging** area on the **Configure > Subsystems > Logging > General** tab. Alternatively, you can check the value of the variable *proxy.config.log2.logging_enabled* in the **records.config** file.

Check that a log format is enabled. In Websense Content Manager, check that a standard format is enabled on the **Configure > Subsystems > Logging > Formats** tab or that the custom format is enabled on the **Custom** tab. In the **records.config** file, you select standard formats or the custom format by editing variables in the **Logging Config** section.

Websense Content Gateway error indicates too many network connections

By default, Websense Content Gateway supports 8000 network connections: half of this number is allocated for client connections and half for origin server connections. A connection throttle event occurs when client or origin server connections reach 90% of half the configured limit (3600 by default). When a connection throttle event occurs, Websense Content Gateway continues processing all existing connections but does not accept new client connection requests until the connection count falls below the limit.

Connection throttle events can occur under the following conditions:

◆ If there is a *connection spike* - if thousands of client requests all reach the proxy at the same time. Such events are typically transient and require no corrective action.

◆ If there is a *service overload* - if client requests continuously arrive faster than the proxy can service them. Service overloads often indicate network problems between Websense Content Gateway and origin servers or indicate that Websense Content Gateway needs more memory, CPU, cache disks, or other resources to handle the client load.

Examine the MRTG graphs to determine the nature of the connection throttle. In particular, check the Client Connections, TCP Connections, and Client Ops Per Second graphs. You can also check error messages in the system log file, error log file, or event log files.

If necessary, you can reset the maximum number of connections supported by the proxy on the **Configure > Networking > Connection Management >Throttling** tab or by editing the value of *proxy.config.net.connections_throttle* in the

records.config file. Do not increase the connection throttle limit unless the system has adequate memory to handle the client connections required. A system with limited RAM might need a throttle limit lower than the default value.

> **Important**
> Do not set this variable below the minimum value of 100.

Low memory symptoms

Under heavy load, the Linux kernel can run out of RAM. The low memory condition can cause slow performance and a variety of system problems. RAM exhaustion can occur even if the system has plenty of free swap space.

Symptoms of extreme memory exhaustion include the following messages in the system log files (**/var/log/messages**):

```
WARNING: errno 105 is ENOBUFS (low on kernel memory),
consider a memory upgrade
kernel: eth0: can't fill rx buffer (force 0)!
kernel: recvmsg bug: copied E01BA916 seq E01BAB22
```

To avoid memory exhaustion, add more RAM to the system or reduce the load on Websense Content Gateway.

Connection timeouts with the origin server

Certain origin servers take longer than 30 seconds to post HTTP requests, which results in connection timeouts with Websense Content Gateway. To prevent such connection timeouts, you must change the value of *proxy.config.http.connect_attempts_timeout* in the **records.config** file to 60 seconds or more.

IBM Web servers do not work with Websense Content Gateway

IBM Web servers do not support the TLS (Transport Layer Security) protocol. For IBM Web servers to work with Websense Content Gateway, you must edit the value of a configuration variable.

1. Open the **records.config** file located in the Websense Content Gateway **config** directory (default location is **/opt/WCG/config**).
2. Edit the following configuration variable:

Variable	Description
proxy.config.ssl.TLSv1	Set this variable to 0 (zero).

3. Save and close the file.

4. From the Websense Content Gateway **bin** directory (default location is **/opt/ WCG/bin**), run `content_line -x` to apply the changes.

Websense Content Gateway does not start (or stop)

Websense Content Gateway starts automatically upon installation. If you must stop the product, the preferred method to stop and restart Websense Content Gateway is to use the `./WCGAdmin start` and `./WCGAdmin stop` commands.

Starting or stopping Websense Content Gateway

1. Become the Websense Content Gateway user called "Websense":

    ```
    su Websense
    ```

2. Change to the Websense Content Gateway **bin** directory. The default location is `/opt/WCG`

3. Start the proxy:

    ```
    ./WCGAdmin start
    ```

    Stop the proxy:

    ```
    ./WCGAdmin stop
    ```

GLOSSARY

ALTERNATES

Different versions of the same Web object. Some origin servers answer requests to the same URL with a variety of objects. The content of these objects can vary, depending on whether a server delivers content for different languages, targets different browsers with different presentation styles, or delivers variable content at different times of the day.

ARM

Adaptive Redirection Module. Used in transparent proxy caching, ARM is a component that redirects intercepted client traffic destined for an origin server to the Websense Content Gateway application. Before the traffic is redirected by the ARM, it is intercepted by an **L4 switch** or router.

CACHE

Stores copies of frequently accessed objects close to users and serves them to users when requested. See also **object store**.

CACHE HIERARCHY

Levels of caches that communicate with each other. All cache hierarchies recognize the concepts of **parent cache** and **child cache**.

CACHE HIT

An object in the cache that can be served directly to the client.

CACHE MISS

An object that is not in the cache or that is in the cache but no longer valid. In both cases, the proxy must get the object from the **origin server**.

CACHING WEB PROXY SERVER

A Web proxy server with local cache storage that allows the proxy to fulfill client requests locally, using a cached copy of the origin server's previous response.

CGI

Common Gateway Interface. A set of rules that describe how an origin server and another piece of software (a *CGI program*) located on the same machine communicate.

CGI-BIN

The most common directory name on an origin server in which **CGI** programs are stored.

CHILD CACHE

A cache lower in a **cache hierarchy** for which Websense Content Gateway is a parent. See also **parent cache**.

CLUSTER

A group of Websense Content Gateway nodes that share configuration information and can act as a single large virtual cache.

CONFIGURE MODE

One of two modes in **Content Manager**. Configure mode lets you configure the Websense Content Gateway system. See also **Monitor mode**.

CONTENT_COP

A Websense Content Gateway process that monitors the health of the **content_gateway** and **content_manager** processes by periodically issuing heartbeat requests to fetch synthetic Web pages.

CONTENT_GATEWAY

A Websense Content Gateway process that is the cache processing engine of the Websense Content Gateway product. content_gateway is responsible for accepting connections, processing requests, and serving documents from the **cache** or **origin server**.

CONTENT MANAGER

Websense Content Gateway's browser-based interface consisting of a series of Web pages that enable you to monitor performance and change configuration settings.

CONTENT_MANAGER

> A Websense Content Gateway process and the command and control facility. content_manager is responsible for launching, monitoring, and reconfiguring the **content_gateway** process. It is also responsible for the administration user interface, the proxy auto-configuration port, and the statistics interface, cluster administration, and **virtual IP failover**.

COOKIE

> A piece of information sent by an origin server to a Web browser. The browser software saves the information and sends it back to the server whenever the browser makes additional requests from the server. Cookies enable origin servers to keep track of users.

DNS

> Domain Name Service. Websense Content Gateway includes a fast, asynchronous DNS resolver to streamline conversion of host names to IP addresses.

EXPLICIT PROXY CACHING

> A Websense Content Gateway configuration option, in which client software (typically a browser) must be specifically configured to send Web requests to the Websense Content Gateway proxy.

FTP

> File Transfer Protocol. A protocol based on TCP/IP for reliable file transfer.

FULL CLUSTERING

> A Websense Content Gateway cluster distributes its cache across its nodes into a single, virtual object store, rather than replicating the cache node by node. See also **management-only clustering**.

HTTP

> Hypertext Transfer Protocol. The client/server protocol upon which the World Wide Web is based.

ICP

> Internet Cache Protocol. A protocol for proxy caches to exchange information about their content.

IP

Internet Protocol. The lowest-layer protocol under TCP/IP responsible for end-to-end forwarding and long packet fragmentation control.

ISP

Internet Service Provider. An organization that provides access to the Internet.

JAVASCRIPT

A scripting language designed to give Web pages the ability to interact with the people viewing them. Examples of such interaction include performing actions in response to mouse movement or mouse clicks and validation of what has been entered into forms.

L4 SWITCH

An Ethernet switch that can control network traffic flow using Level 4 rules. The switch can intercept desired client protocol packets and direct them to a proxy for transparent operation.

MANAGEMENT-ONLY CLUSTERING

A Websense Content Gateway option, in which all nodes in a cluster automatically share configuration information. See also **FULL CLUSTERING**.

MIB

Management Information Base. The set of parameters that an SNMP management station can query in the SNMP agent of a network device (for example, a router). Websense Content Gateway supports MIB-2 (a well-known standard MIB). Websense Content Gateway supports two MIBs: MIB2 (a well-known standard MIB) and the a proprietary MIB, which provides more specific node and cluster information.

MONITOR MODE

One of two modes in **Content Manager**. Monitor mode lets you view statistics about Websense Content Gateway performance and Web traffic. See also **Configure mode**.

MRTG

Multi Router Traffic Grapher. A graphing tool provided with Websense Content Gateway that enables you to monitor Websense Content Gateway performance.

NETSCAPE LOG FORMAT

A standard access log format. Using the Netscape log format, you can analyze Websense Content Gateway access log files with off-the-shelf log analysis scripts. See also **Squid log format**.

OBJECT STORE

A custom high-speed database, on which Websense Content Gateway stores all cached objects.

ORIGIN SERVER

The Web server that contains the original copy of the requested information.

PAC FILE

Proxy Auto-Configuration file. A JavaScript function definition that a browser calls to determine how requests are handled.

PARENT CACHE

A cache higher up in a **cache hierarchy**, to which the proxy can send requests.

PROXY SERVER

See **web proxy server**.

ROUTER

A device that handles the connection between two or more networks. Routers look at destination addresses of the packets passing through them and decide which route to send them on.

SNMP

Simple Network Management Protocol. A set of standards used for communication with devices connected to a TCP/IP network. SNMP-compliant devices (agents) store information about themselves in **MIB**s and provide this information to SNMP Managers.

SOCKS

A circuit-level proxy protocol that provides a tunneling mechanism for protocols that cannot be proxied conveniently.

SQUID LOG FORMAT

A standard access log format. Using the Squid log format, you can analyze Websense Content Gateway event log files with off-the-shelf log analysis scripts. See also **Netscape log format**.

SSL

Secure Sockets Layer. A protocol that enables encrypted, authenticated communications across the Internet. Used mostly in communications between origin servers and Web browsers.

SYSLOG

The UNIX system logging facility.

TCP

Transmission Control Protocol. An Internet standard transport layer protocol. TCP provides reliable end-to-end communication by using sequenced data sent by IP.

TRANSPARENT PROXY CACHING

A configuration option that enables Websense Content Gateway to intercept and respond to Internet requests without requiring users to reconfigure their browser settings. It does this by intercepting traffic destined for an origin server and redirecting that traffic through the proxy cache.

URL

Uniform Resource Locator. The address that defines the route to a file on the Web or other Internet facility.

VIRTUAL IP FAILOVER

An option available to clustered Websense Content Gateway servers, in which WCG maintains a pool of virtual IP addresses that it assigns to the nodes of a cluster. If a node fails, the remaining nodes mask the fault and take over the failed node's virtual interface.

WCCP

Web Cache Control Protocol. A protocol used by Cisco IOS-based routers to redirect traffic during transparent proxy caching.

WEB PROXY SERVER

A proxy server that forwards client requests to **origin server**s. The proxy server may deny requests according to filter rules or security limitations.

WEB SERVER

A computer that provides World Wide Web services on the Internet. See also **origin server**.

WPAD

Web Proxy Auto-Discovery. A protocol that allows clients to automatically locate a Web proxy, providing the benefits of a proxy without the need for explicit client configuration.

Copyrights

DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR COR-RECTION.

Netscape Directory SDK 4.0 for C

Netscape Directory SDK 4.0 for C is available without license fee under the terms of the Netscape ONE SDK End User License Agreement.

Each of the Components is provided on an "AS IS" basis, without warranty of any kind, including without limitation the warranties of merchantability, fitness for a particular purpose and non-infringement. The entire risk as to the quality and performance of the Components is borne by you. Should the Components prove defective or inaccurate, as the case may be, you and not Netscape or its suppliers assume the entire cost of any service and repair. In addition, the security mechanisms, if any, implemented by the Components have inherent limitations, and you must determine that each of the Components sufficiently meets your requirements. This disclaimer of warranty constitutes an essential part of the agreement. SOME JU-RISDICTIONS DO NOT ALLOW EXCLUSIONS OF AN IMPLIED WARRANTY, SO THIS DIS-CLAIMER MAY NOT APPLY TO YOU AND YOU MAY HAVE OTHER LEGAL RIGHTS THAT VARY BY JURISDICTION.

Emanate

Emanate® software is used under license agreement with SNMP Research International Incorporated and the relevant third parties.

Tcl 8.3

Tcl software is copyrighted by the Regents of the University of California, Sun Microsystems, Inc., Scriptics Corporation, and other parties. The following terms apply to all files associated with the software unless explicitly disclaimed in individual files. The authors hereby grant permission to use, copy, modify, distribute, and license this software and its documentation for any purpose, provided that existing copyright notices are retained in all copies and that this notice is included verbatim in any distributions. No written agreement, license, or royalty fee is required for any of the authorized uses. Modifications to this software may be copyrighted by their authors and need not follow the licensing terms described here, provided that the new terms are clearly indicated on the first page of each file where they apply.

IN NO EVENT SHALL THE AUTHORS OR DISTRIBUTORS BE LIABLE TO ANY PARTY FOR DI-RECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OF THIS SOFTWARE, ITS DOCUMENTATION, OR ANY DERIVATIVES THEREOF, EVEN IF THE AUTHORS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. THE AUTHORS AND DISTRIBUTORS SPECIFICALLY DISCLAIM ANY WARRANTIES, IN-CLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FIT-NESS FOR A PARTICULAR PURPOSE, AND NONINFRINGEMENT. THIS SOFTWARE IS PROVIDED ON AN "AS IS" BASIS, AND THE AUTHORS AND DISTRIBUTORS HAVE NO OBLI-GATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFI-CATIONS.

libdb

LIBDB Copyright © 1991, 1993 The Regents of the University of California. All rights reserved. This product includes software developed by the University of California, Berkeley and its contributors.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE AR DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRET, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAM-AGES (INCLUDING LOSS OF USE, DATA, OR PRIFITS; OR BUSINESS INTERRUPTION) HOW-EVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER ARISING IN CONTRACT, STRICT LIABILITY OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, ENVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

INN

Copyright © 1991, 1994, 1995, 1996, 1997, 1998, 1999, 2000, 2001 The Internet Software Consortium and Rich Salz. This code is derived from software contributed to the Internet Software Consortium by Rich Salz Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. All advertising materials mentioning features or use of this software must display the following acknowledgement: This product includes software developed by the Internet Software Consortium and its contributors. 4. Neither the name of the Internet Software Consortium nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE INTERNET SOFTWARE CONSORTIUM AND CON-TRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE INTERNET SOFT-WARE CONSORTIUM OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCI-DENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT

LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

MRTG

Multi Router Traffic Grapher (MRTG) is freely available under the terms of the GNU General Public License. Copyright © 1989, 1991 Free Software Foundation, Inc. 59 Temple Place - Suite 330, Boston, MA 02111-1307, USA

BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

Libregx

Copyright © 1992, 1993, 1994, 1997 Henry Spencer. All rights reserved. This software is not subject to any license of the American Telephone and Telegraph Company or of the Regents of the University of California.

Index