

ISA Server 2000[®]

Caching with Microsoft[®] Internet Security and Acceleration Server 2000

Abstract

Microsoft's Internet Security and Acceleration (ISA) Server 2000 provides an extensible enterprise firewall and Web cache server that integrates with Microsoft [®] Windows[®] 2000 for policy-based security, acceleration, and management of internetworking. This paper focuses on the ISA Web Cache service, which supports forward caching for outgoing requests and reverse caching for incoming requests.

Additionally, the paper examines the advanced routing capabilities of distributed and hierarchical caching. Caching reduces network traffic and improves user response times while retaining freshness of data through ISA Server's advanced caching policies and controls that provide the ability to specify object-level granular routing and cache content rules. ISA Server's Web Cache is an extensible, high-performance Web cache, offering scalability, high availability and ease of management for high volume Internet traffic.

CONTENTS

INTRODUCTION	5
CACHE ARCHITECTURE	7
HTTP Proxy Server	7
Lifetime of Cached Resources	8
Internet Acceleration with ISA Server	8
ISA Server Components	9
Policy Elements	10
Rules for ISA Acceleration	10
Rules Order	12
Authentication and Rules	12
Chained Authentication	13
Active Directory Integration	13
FORWARD CACHING	13
TTL Configuration Options	14
Restricting what Objects are Cached	14
RAM Caching	15
ACTIVE CACHING	15
What Content is Refreshed	15
Tuning Active Caching	15
SCHEDULED CONTENT DOWNLOAD	16
How it Works	16
Configuring a Scheduled Job	16
REVERSE CACHING	17
How it Works	17
Web Publishing Rules	17

Routing Rules	18	
WEB PROXY ROUTING	18	
Routing Requests		18
Using Dial-Up	19	
Caching Requests	19	
SSL Bridging	19	
HTTP REDIRECTOR	21	
CACHE ARRAY ROUTING PROTOCOL.....	21	
How it Works	21	
The Hashing Algorithm		22
Requirements for CARP	25	
Array Routing Scenarios	25	
Web Publishing with Arrays	26	
ISA SERVER SCENARIOS	26	
Medium Sized Business Scenario	26	
Large Enterprise Scenario	27	
SUMMARY	29	
FOR MORE INFORMATION.....	29	

Introduction

Microsoft® Internet Security and Acceleration (ISA) Server 2000 provides an extensible enterprise firewall and Web cache server that integrates with Microsoft® Windows® 2000 for policy-based security, acceleration, and management of internetworking. ISA Server provides three modes—a high-performance Web cache server, a multilayer firewall and an integrated mode that combines the firewall and cache. The cache improves network performance and end-user experience by storing frequently requested Web information, while the multilayer firewall provides enterprise-class security. The firewall screens at the packet, circuit and application layer and controls access policy and routing of traffic. The cache and firewall can be deployed separately on dedicated servers, or used together on the same box.

Sophisticated management tools simplify policy definition, traffic routing, server publishing, and monitoring. ISA Server builds on Windows 2000® security, directory, virtual private networking (VPN), and bandwidth control. Whether deployed as separate cache and firewall servers or in integrated mode, ISA Server can be used to improve Internet access speed, and maximize employee productivity as well as enhance network security, and enforce Internet usage policy, for organizations of all sizes.

This paper examines the Web caching benefits of ISA Server. Web caching is a simple, yet powerful idea. Basically, ISA Server's caching reduces Internet traffic by storing the most frequently accessed Web objects locally. When a request occurs for content previously retrieved by another user, the content is served at local network speed from ISA Server's local cache instead of the destination server. The user experiences a near instantaneous response. Additionally, the organization's Internet connection is relieved from transmitting redundant traffic. An overloaded Internet connection is not the only reason for a Web cache—even when your Internet bandwidth isn't constrained you receive better performance. For example, sluggish Internet response times often result from multiple router hops that negotiate around the temporary outages on the Internet. With ISA Server's caching, popular Web object requests remain on the local high-speed network.

ISA Server supports two basic forms of caching: *forward* and *reverse*. Forward caching is used for outgoing requests, while reverse caching is used for incoming requests. Research establishes the existence of strong groupings of access patterns within an organization or workgroup. In fact, depending on the organization's size and access patterns these findings suggest that between 35 to 50 percent of content requests can be serviced from the ISA Server forward cache. The results for ISA Server's reverse cache are even more impressive—nearly 100 per cent of the content requests are retrieved from the ISA Server cache.

ISA Server implements caching in the *Web Proxy Service*. Rather than requesting content directly from the destination server, Web Proxy clients—or the HTTP Redirector—direct the request to the Web Proxy Service. The Web Proxy Service checks the cache for a local copy of the content. If the content is not available, the Web Proxy service makes a request to the destination server and stores the response in the cache. The content is then served to the client from the local cache.

ISA Server enhances the basic caching technology with the following features:

- **ACTIVE CACHING** ISA Server can be configured to download the most frequently requested content before the

content becomes stale or invalid. This improves response time by increasing the frequency that content is served from ISA Server's local cache.

- **SCHEDULED CONTENT DOWNLOAD** Administrators have the ability to download specific Web content at a scheduled time interval. This extra control allows for optimal use of network bandwidth during periods of network inactivity and provides an accelerated response time for this content during peak hours by serving content from ISA Server's local cache.
- **REVERSE CACHING** Reverse caching improves the performance of internal Web sites published to the Internet by placing ISA Servers in front of the Web Server(s) and caching frequently accessed content. This relieves stress on the Web servers as well as improving user response times. ISA Server can provide reverse caching for Web servers, Web farms, or for placing content closer to different geographical areas. For example, an ISA Server might be implemented for Internet clients in Europe that caches content for the company Web site that is hosted in the United States.
- **TRANSPARENT HTTP REDIRECTOR** The Transparent HTTP Redirector intercepts requests that would normally transfer through the Firewall and SecureNAT layers and redirects the requests to the Web Proxy Service. Internet software not configured to use the Proxy Service will still benefit from caching when the redirector is active.
- **CACHE ARRAY ROUTING PROTOCOL** Proxy Arrays balance the load by distributing requests for cached Web objects across multiple servers. CARP uses a hash-based routing protocol that enables downstream clients to predetermine exactly where content is cached in an array of proxy servers. This reduces the load on the ISA Array by eliminating additional queries being generated within the array itself.
- **WEB PROXY ROUTING** Routing rules may be defined for upstream requests. A routing rule can direct requests to an upstream ISA Server or redirect the request to a different destination host.
- **FTP CACHING** ISA Server extends the benefits of caching to the FTP protocol. The same routing and caching features applied to HTTP content can be applied to FTP content.

These features can be combined to meet the specific requirements of an enterprise network. For example, Scheduled Content Download and Reverse Caching can be combined to deliver content to servers that are closer to the target users on different segments of the Internet. This white paper will provide detail about each of these features and how they can benefit organizations of all sizes.

Cache Architecture

The foundation of the Internet is a set of standards that allow diverse computers to communicate over a global public network. This section will give you an overview of the standards that relate to ISA Server's caching architecture and highlight the innovations that differentiate ISA Server from typical Web caching servers.

A basic understanding of the Hypertext Transfer Protocol (HTTP) is required to best appreciate ISA Server's cache architecture. You are probably familiar with the use of HTTP in retrieving Web pages from servers on the Internet. RFC-2616 describes the HTTP standard as "an application-level protocol for distributed, collaborative, hypermedia information systems." The HTTP protocol describes how to *request* and *transfer* resources over a network. It also specifies rules for caching content on local networks.

The strength of HTTP is in its independence from the information that is transferred using the protocol. Included in the HTTP specification are rules for negotiating the type and encoding of the information requested. This allows the benefits of HTTP to be realized in a number of different applications such as the delivery of XML and HTML data and can be extended to support other applications such as directory access and WebDAV.

A basic HTTP request consists of a *request-line*, a *header* and a *body*. The request-line tells the origin server what method is used in the request and the *URI (Uniform Resource Identifier)* of the resource requested. The header includes information about the request such as the capabilities of the client, authentication information and from where the request was referred. A request body can include any information being transferred to the server. For example, the data in an HTML form filled out by a user is commonly transferred in the request body.

The origin server responds with a similar message. The status of the response is transferred in the response-line. Status examples include 200 – OK, 404 – Not Found, etc. Following the status is the header information, and finally, the body of the response, most commonly an HTML document. In the absence of a proxy server, such as ISA Server, the request and response are sent directly across the network.

An HTTP client can retain a local copy of responses that are determined to be cachable. HTTP specifies the rules for determining whether or not content is cachable. When a cached resource becomes stale, a client can use the If-Modified-Since header field to instruct the origin server to send the information only if it has been modified. You can see that when HTTP is used effectively, it can reduce a significant amount of unnecessary network traffic.

HTTP Proxy Server

An HTTP Proxy Server makes requests to origin servers on behalf of clients. This serves two purposes. First, an HTTP Proxy Server provides a safe path across a firewall without exposing its clients to a public network. An HTTP Proxy Server can also block its clients from accessing inappropriate or dangerous resources on the public network. Second, an HTTP Proxy Server extends the benefit of a cached resource to an entire local network. Once a request is made to an origin server and the response is cached, additional requests for the same Web object made through the HTTP Proxy Server are served from the local network cache. This not only

reduces the congestion of upstream network connections, but also gives clients a faster response.

When you configure a client to use a proxy server, the HTTP requests are sent to the proxy server instead of the origin server. The proxy server inspects the request and generates a new request that is sent directly to the origin server or to another upstream proxy server. When the proxy server receives a response, it caches the resource and generates a response to the client.

Lifetime of Cached Resources

Resources that are cached on the local network will become stale after some amount of time and need to be updated. HTTP specifies the rules for caching resources and how to determine when resources need to be validated. The goal is to be semantically transparent. In other words, the most current resource is delivered to the client. However, the HTTP specification permits some exceptions to semantic transparency where a client may utilize a stale resource.

The freshness of a resource determines whether or not it needs to be revalidated. A resource is determined to be fresh either explicitly or heuristically. The expiration time, or time when the resource becomes stale, is determined explicitly if the origin server specifies the lifetime of the content. The expiration time is determined heuristically if no expiration information is provided. An origin server can also specify that content is not to be cached, either because the information is confidential or the information is immediately stale.

ISA Server uses *Time-To-Live (TTL)* to represent the rules of freshness for HTTP content. Time-To-Live groups the explicit and heuristic rules as well as the no-cache options together for easier administration. Time-To-Live also provides methods for enforcing local rules for content such as a minimum age before content will be considered stale.

Internet Acceleration with ISA Server

ISA Server improves on the standard HTTP Proxy Server by extending the scope of the proxy role. ISA Server accelerates Internet access by introducing features for automatically refreshing stale content resources, intelligently spreading requests among multiple servers in an array, filtering content for special security or cache requirements and implementing caching for the FTP protocol.

ISA Server Enterprise Edition enables the proxy service to be load-balanced among multiple servers in an ISA Array. The Enterprise Edition also integrates with the Microsoft Active Directory™ to allow enterprise policies to be applied to all of the ISA Servers in an organization. This not only reduces the effort of managing many servers, but also increases security by eliminating inconsistencies among ISA Servers.

In an ISA Server environment, when a user makes an initial request for content, the Web browser begins by downloading a configuration file from a nearby ISA server. This configuration file instructs the client where to send HTTP requests for a particular URL. If an ISA Server Array is available, a routing algorithm contained in the configuration file provides the client with the correct server in the array for each requested URL. For example, the URL <http://www.microsoft.com/> might be stored in the cache on one ISA Server while the URL

<http://support.microsoft.com/> might be stored in the cache on a second ISA Server. The algorithm spreads the requests out evenly across the servers in the array. Because this configuration file is downloaded via HTTP, it is assigned a Time-To-Live and is revalidated when it becomes stale.

Once the client has the configuration file, it can now direct HTTP Requests at the appropriate ISA Server. The ISA Server Web Proxy Service receives the request and compares it to the Protocol and Site and Content Rules as defined by the Administrator. If the request violates a rule, a 502 Access Denied response is generated. Otherwise, the request passes through the ISA Server routing rules.

The ISA Server routing rules determine how a request is passed to the server and whether or not the request is cached. The routing rules can specify that requests be directed to an upstream HTTP Proxy Server, ISA Server, or ISA Array. Routing rules may also redirect requests to different destination servers, transmit HTTP requests with SSL security and cache content that normally would not be cached.

By default, the ISA Server checks the local cache in RAM (fastest) and the local cache on disk (fast) for a valid content resource. If no valid content resource is found, a request is generated to the destination server. When the response is received, it is stored in the cache and a response is generated to the client.

ISA Server Components

ISA Server is split into several services each serving different functions in the architecture. Depending on the options selected at installation, all or some of the services will be present.

The ISA Server Control Service handles the starting and restarting of the other ISA Server services as necessary. It manages synchronization of configuration information in ISA Server Arrays, updating client configuration files and deleting any log files older than the configured maximum. The Control Service also generates alerts and executes actions in response to server security and health conditions.

The Firewall Service responds to non-proxy requests made by many different types of network applications. It acts as a secure router between the private network and the Internet. The Firewall Service can also intercept HTTP requests and redirect them to the Web Proxy Service. This service is only present when ISA Server is installed in firewall or integrated mode. The Firewall Service is not discussed further in this white paper.

The Web Proxy Service handles requests from HTTP, FTP and Gopher clients. It makes requests to origin servers on behalf of clients. The Web Proxy Service caches content locally and determines when the requests can be served from the local cache. This service is present when ISA Server is installed in cache or integrated mode.

The Scheduled Cache Content Download Service downloads content into the cache on a schedule predetermined by the administrator. Once the content is in the cache, requests for it can be served from the local network instead of retrieving it across the upstream link. This service is present when ISA Server is installed in either cache or integrated mode.

The H.323 Gateway Service is an optional component that provides a directory and router for internal H.323 clients such as NetMeeting 3.0 or higher.

Policy Elements

You use *Policy Elements* to apply ISA Server policies to groups of objects. For example, all of the computers in the Marketing Department can be grouped in a Client Address Set called "Marketing Computers." Then, policies can be applied to this Client Address Set instead of recalling which computers are part of Marketing each time you want to configure a new policy for that group. In addition, you may add and remove computers to the "Marketing Computers" Client Address Set instead of managing policies by individual computers.

The following Policy Elements are available when ISA Server is installed in cache mode:

- **SCHEDULES** You can create rules that are only in effect at certain times. For example, you might create a schedule for after hours access. Schedule sets define periods during the day and days during the week such as work hours or weekends. A Schedule can be used to restrict access to specific resources during specified times while allowing access at a different time.
- **DESTINATION SETS** You specify destinations by hostname, IP address or range of IP addresses. Destination Sets may also include a path. You can configure a destination set for a specific destination or a group of destinations using wildcards or IP address ranges. You use destination sets to apply rules to resources such as a Routing Rule or Publishing Rule for a specific URL. For example, a destination might be <http://www.microsoft.com/technet>. Perhaps this Destination Set is used for a special caching rule. In addition, third-party companies offer URL filtering products as an add-on to ISA Server. For more information see ISA Server's third-party Web page on Access Control at <http://www.microsoft.com/isaserver/thirdparty/accesscon.htm>.
- **CLIENT ADDRESS SETS** You specify clients by a range of IP addresses. Client Address Sets can represent internal clients for access to Internet resources or external clients for access to internal servers that are published to the Internet. They are used to associate a rule with a group of computers. A Site and Content rule might be restricted to only a specific set of computers defined in a Client Address Set.
- **PROTOCOL DEFINITIONS** These allow you to assign friendly names to IP Ports for restricting access to protocols routed through ISA Server. In cache mode, the only protocol definitions that are significant are FTP, HTTP, S-HTTP and Gopher.
- **CONTENT GROUPS** Content Groups assign friendly names to content types using file extensions and MIME types. Content Groups are used to restrict or grant access to content resources based on their type. For example, you can define a Content Group for MP3 files and control access to that type of resource.
- **DIAL-UP ENTRIES** These elements define dial-up connections to be used for retrieving content. After defining a Dial-up Entry, it may be applied to a Routing Rule to instruct the ISA Server to dial a service provider.

Rules for ISA Acceleration

Rules instruct ISA Server to accept and process requests from internal and external Web clients in a specified manner. ISA Server's rules are processed in the order shown in Figure 1 and Figure 2.

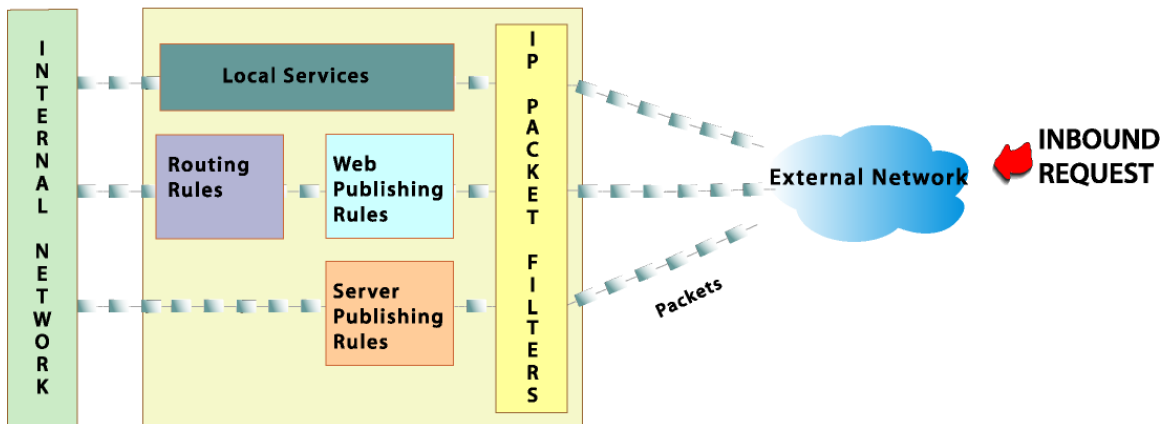


Figure 1 Inbound rule processing order

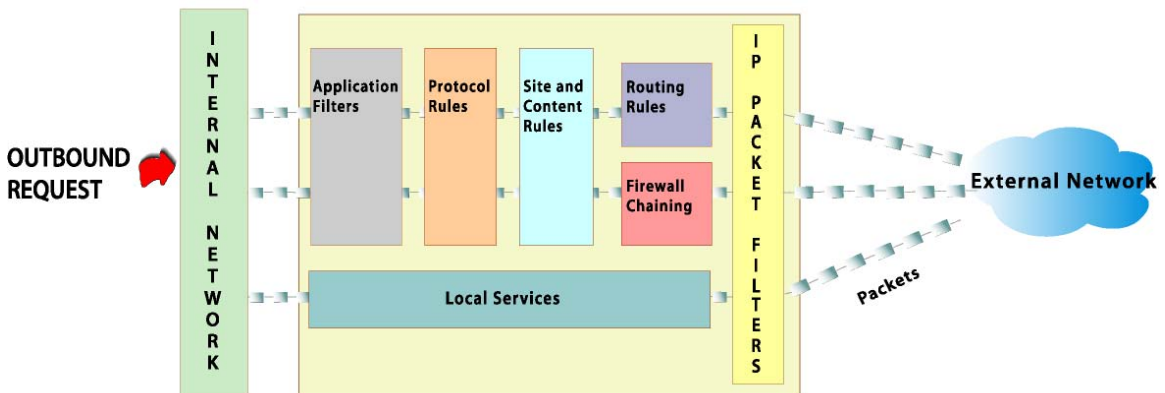


Figure 2 Outbound rule processing order

You can configure the following ISA Server rules:

- PACKET FILTER RULES** Packet Filter Rules allow control over the types of IP packets that are accepted on the external interface. These rules are available in Integrated and Firewall mode only. When enabled, all packets on the external interface are dropped unless you explicitly configure a specific packet type to be accepted. Typically, you create packet filters to control incoming traffic. For outgoing requests, ISA Server will open ports dynamically as they are needed and monitor the ports for responses.
- WEB PUBLISHING RULES** You use Web Publishing Rules to configure ISA Server to forward requests from external clients to internal Web servers. These rules are used in the reverse caching scenario to accelerate access to an organization's Web servers. Web publishing rules also determine security restrictions for incoming requests and how the requests are encrypted when they are forwarded to the internal server.

- **ROUTING RULES** You use Routing Rules to route requests to upstream ISA Servers or redirect requests to alternate destination servers. Routing rules specify what requests are routed, what requests are redirected, and what requests are retrieved directly from the destination server. You use Routing Rules to configure Caching policies specific to Destination Sets. Additionally, you can use Routing Rules to specify whether or not to serve objects from the cache and whether or not to cache the responses from the destination server.
- **BANDWIDTH RULES** Microsoft ® Windows® 2000 Server and Microsoft ® Windows® 2000 Advanced Server include built-in Quality of Service (QoS) functionality for controlling the amount of bandwidth available to a particular application. Bandwidth Rules are used by ISA Server to determine what priority to request from the QoS service. You can use these rules to give a special priority to real-time communications or restrict the bandwidth of broadband communications during office hours.
- **PROTOCOL RULES** You can use Protocol Rules to control access to specific protocols that are allowed to pass through the ISA Server. In cache mode, the only protocols available are HTTP, S-HTTP, FTP and Gopher. In integrated and firewall mode, any protocols can be defined using Protocol Definitions and allowed or denied using Protocol Rules.
- **SITE AND CONTENT RULES** Site and Content Rules control access to specific destination servers and content types by internal clients. You specify destinations using Destination Sets. A Destination Set can be an entire domain, a specific server or specific URL. Content restrictions are first defined using Content Groups and then restricted using the Site and Content Rules.

Authentication and Rules

One way to restrict the application of a rule is by username or security group. Only those requests originating from users that meet the conditions set forth in the “Applies To” settings are applied to the rule.

When a user requests an HTTP resource, the Web browser will always attempt an anonymous connection before authenticating with the ISA Server. When a request is received anonymously, ISA Server will first process it as an anonymous request. If a rule is found that allows the anonymous request to pass through the ISA Server, the user is not asked to authenticate. If no rule is found that allows anonymous access, then ISA Server challenges the user for authentication credentials. Once the user passes the credentials back to ISA Server, the credentials are compared to the rule sets to determine if the user is allowed access to the requested resource.

For example, consider the following settings:

- Outgoing requests are set to allow anonymous.
- **SITE AND CONTENT RULE 1** Allow anonymous access to all Internet destinations.
- **SITE AND CONTENT RULE 2** Deny access to the user Sally.

Site And Content Rule 2 applies only if Sally is required to authenticate. If Sally makes an HTTP request to the Web Proxy Service for a resource on the Internet, the request is allowed because the Web Proxy Service is not configured to ask for authentication. This is true whether or not the Firewall Client is installed.

There are two ways to force ISA Server to authenticate all HTTP requests. The first is to set Outgoing and/or Incoming requests to authenticate using the configuration settings in the ISA Server Properties. The second way is to remove any Site and Content rules that allow anonymous access for the resource that you want to restrict.

Chained Authentication

You use Chained Authentication when ISA Server is challenged for security credentials while routing a request to an upstream server. Chained Authentication is supported for requests routed to upstream servers running Microsoft Proxy 2.0 or ISA Server.

Chained Authentication starts with the downstream server requesting the client to authenticate. While the request is being routed to an upstream server, the upstream server may also request that the client authenticate. ISA Server will pass the client authentication to the upstream server.

If the upstream server cannot identify the client's authentication, the downstream ISA Server may also pass its own security credentials in order to access the requested content. The security credentials to access the upstream server are defined when configuring the Routing Rules on the downstream ISA Server.

Microsoft Active Directory™ Integration

ISA Server Enterprise Edition provides integration with Active Directory for easy and consistent management. Active Directory integration allows administrators to create Enterprise Policies and apply them to all ISA Servers in an organization. When the Enterprise Policy is updated, the settings will be propagated via Active Directory replication to all locations in the network. Each server will apply the Enterprise Policy to its own configuration.

When using two or more ISA Enterprise Servers in an array, the configuration for the ISA Array is stored in the Active Directory so that it is accessible by all of the servers. This eliminates the need for updating all of the servers individually when the array configuration needs to be changed.

Forward Caching

Forward Caching is the principal idea behind ISA Server's Internet acceleration. Acceleration occurs by serving valid Web objects from a location closer to the user. This not only provides a quicker response time, but also reduces the bandwidth usage of your upstream Internet connection.

You can optimize the way the Web Proxy Service caches objects downloaded through the ISA Server. For example, you can enable or disable HTTP Caching or FTP caching functionality. You can also specify how ISA determines the TTL of objects stored in the cache. These settings are made in the properties of the Cache Configuration branch of the management console.

TTL Configuration Options

Typically, ISA Server will follow the rules set by the origin server for how long a cached object is considered valid. This is called *explicit expiration*. However, in some situations the origin server will not specify a TTL, so the expiration of the object must be determined *heuristically*. You can modify how content is heuristically expired by ISA Server.

ISA Server uses a percentage of the time since the object was last modified to determine how long the object should be valid in the cache. ISA Server also applies a minimum and maximum time to the TTL values determined heuristically. There are four ways to configure heuristic expiration.

- **NORMALLY** The TTL for each object is set to 20% of the time since the object was last modified, with a minimum value of 15 minutes and a maximum value of 1 day.
- **FREQUENTLY** Objects are expired immediately unless there is an explicit TTL provided for an object.
- **LESS FREQUENTLY** The TTL for each object is set to 40% of the time since the object was last modified, with a minimum value of 30 minutes and a maximum value of 2 days.
- **SET TIME TO LIVE (TTL) OF OBJECT** Finally, you may configure the percentage, minimum and maximum values through this custom option.

FTP objects do not support a TTL or expiration parameter in the protocol definition. Therefore, a TTL must be specified for all FTP objects that are cached. The TTL for FTP objects may be specified in seconds, minutes, hours, days or weeks.

Restricting what Objects are Cached

You can restrict whether or not the following objects are cached:

- **OBJECT SIZE** Whether or not to cache objects larger than the specified maximum size.
- **UNSPECIFIED LAST MODIFICATION TIME** Whether or not to cache objects without the last modified time specified.
- **INVALID OBJECTS** Whether or not to cache objects that have a response code other than 200 ("OK").
- **DYNAMIC CONTENT** Whether or not to cache objects with question marks in the URL.

At times an origin server on the Internet may become unreachable. If the objects requested from the origin server are in the cache, they may be used even if the TTL has expired. ISA Server provides two methods for controlling the use of expired content. First, you can specify that an expired object be returned only if the object has been expired for a maximum percentage of the original TTL. Second, you can specify a maximum time in minutes since expiration.

RAM Caching

RAM Caching improves cache performance by serving the most frequently requested content from memory rather than disk. You can restrict the size of objects stored in RAM and the percentage of free memory available to the RAM cache.

Active Caching

The caching of content is only the first step to improving performance for repeated requests for the same Web object. The next step is to anticipate what requests will be made and place the Web object in the cache before it is requested. ISA Server implements this idea programmatically with Active Caching.

What Content is Refreshed

Active Caching uses an algorithm to determine which Web objects are the most likely to benefit from being refreshed automatically by the Web Proxy Service. Depending on the current activity of the ISA Server, content that meets the Active Caching criteria is refreshed before the TTL for that content expires. If the activity of the ISA Server is low, the content will be refreshed about half way before the TTL expires. As the ISA Server increases in activity the refreshing of content is put off until just before the TTL expires.

Content is designated for Active Caching in the following way:

A requested resource is downloaded and stored in the cache.

After the initial TTL for that resource expires, the content becomes stale.

If the resource is requested again before a multiple (" n ") of its TTL periods, the resource is placed on the Active Caching list. " n " is determined by tuning the Active Caching settings where $n = 1$ is less frequently, $n = 2$ is normally, and $n = 3$ is frequently.

The resource remains on the Active Caching list provided that it is requested within " n " TTL periods after being refreshed.

If the resource is taken off the Active Caching list, it must meet the original criteria to be placed back on the list.

Tuning Active Caching

You can tune Active Caching to meet the needs of your organization or even disable it altogether. When Active Caching is enabled, you can tune its performance by selecting one of the three available levels: frequently, less frequently, or normally. This adjustment directly impacts the amount of content that is refreshed automatically in the following ways:

- **FREQUENTLY** Content is refreshed often and frequently requested content is more likely to be available in the local cache. The upstream link will be used often to refresh content. This is a good setting if users are likely to

make repeated requests for the same resource.

- **LESS FREQUENTLY** Only the most frequently requested content is refreshed automatically. The upstream link will be used less often to refresh content. This is a good setting if users are often making requests for unique resources or if the upstream link's bandwidth needs to be reserved for other usage.
- **NORMALLY** This option is balanced between the two others and is the default setting. It is optimal for most situations.

Scheduled Content Download

ISA Server cache options include the capability to schedule content for download at a specific time or recurring times. For example, a branch office can schedule an entire intranet site to be downloaded from the corporate network to the branch office's local ISA Server. You can schedule the download to occur during the night so that the upstream link is not affected during the day. Large Web objects, such as reports and graphs contained in the intranet site, are served from the cache and available instantaneously to users.

How it Works

Scheduled content downloads are handled by a service called the *Scheduled Content Download Service*. This service is installed when ISA Server is installed in cache or integrated mode.

The service works by waiting for the time a job is scheduled to run. Then, when it is time for the job to be run, the service retrieves the content at the URL specified in the job and stores the objects in the cache.

Configuring a Scheduled Job

Scheduled Content Download Jobs are configured in the Cache Configuration section of the ISA Management console. You can configure the following options for Scheduled Content Download Jobs:

- **CONTENT TO DOWNLOAD** Each scheduled content job is assigned a starting URL. The URL could be the root page of a Web site, a page, or group of pages within a Web site. By default the scheduled job will traverse all of the links that are contained in the content and continue to recursively traverse links until all of the possible paths are downloaded. However, you can limit the content amount to download as detailed in the following bullet.
- **AMOUNT TO DOWNLOAD** There are several ways you can limit the amount of information downloaded in a scheduled job. For example, you can limit the download to the domain of the specified URL. Any links that are not part of the target domain will be ignored. Another way is to specify a maximum depth of links to traverse.
- **TIME TO LIVE** Some content that is scheduled for download may not have a TTL long enough to be useful to end-users. Therefore, scheduled jobs can be set to override the TTL settings asserted by the origin server. Scheduled content download jobs may also be configured to download and cache dynamic content—even if the origin server specifies that the content is not cacheable.
- **SCHEDULE** You can set the schedules for one-time only download, daily download or weekly download on specified days. Ideally, the time of the download is set for a time when network usage is low.

Reverse Caching

Forward caching provides a substantial performance benefit to your organization's users by serving recurring requests for Web objects from a local network cache. However, this same concept can be applied to users on the Internet or on a business-to-business network accessing your organization's Web servers. By directing incoming requests for your Web server to the ISA Web Proxy Service, the Web objects are served from the ISA Server cache *or* the Web server. The benefits are threefold: External users receive quicker responses from your Web site as content is served from the ISA Server cache; ISA Server offloads the stress from the Web servers; and by using ISA Server's firewall features, your organization's Web servers are more secure from unwanted access. Additionally, a reverse caching ISA Server could use Scheduled Content Download to pull your content from a remote location to your internal network. For example, an organization's Internet users in France would benefit from an ISA Server located in France that is reverse caching the organization's Web site that is located in Canada.

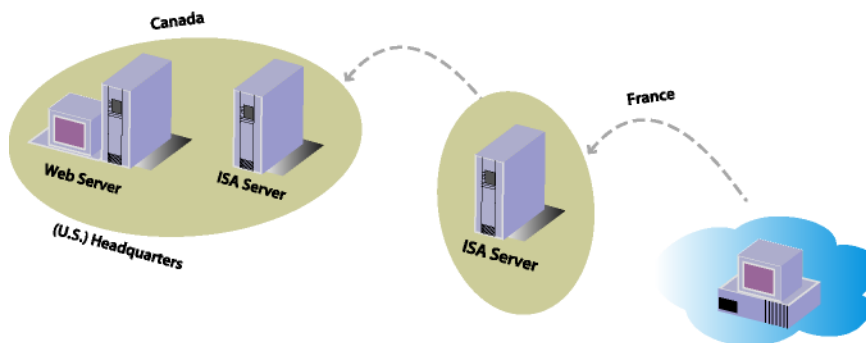


Figure 3 Reverse Caching

How it Works

ISA Server implements reverse caching by impersonating your Web server on the Internet. By configuring the ISA Server with the IP address of your Web site, requests are directed to the Web Proxy Service. The Web Proxy Service answers the request and compares it to the publishing and routing rules to determine what action to take. Web Publishing Rules define which IP addresses or domain names will be answered and redirected to internal Web servers. If the request fits a publishing rule, the request will be passed through the Web Proxy Service, which includes Web Proxy Routing Rules and finally the Bandwidth Rules. When possible, the ISA Server resolves the request from its cache. Otherwise, the requests are forwarded to an internal Web server located behind the ISA Server computer.

Web Publishing Rules

Basically, Web Publishing Rules map incoming requests for Web objects to the appropriate Web servers behind the ISA Server computer. They determine exactly how ISA Server should intercept incoming requests for an internal Web server and how ISA Server should respond on

behalf of the Web server. Web Publishing Rules define what IP addresses or domain names are published using Destination Sets. A *Destination Set* contains a computer name, IP address, or IP range and can include a path. Additionally, a Destination Set might include one or more computers or folders on specific computers or even an entire domain using a wildcard; for example, ***.microsoft.com**.

When a request matches the destination set of a Publishing Rule, the rule applies an action. The action might be to discard the request or redirect the request to an internal Web server. If the rule is to redirect the request, a Web server can be specified by domain name or IP address. Additionally, you can configure requests for the standard HTTP, S-HTTP and FTP ports to something other than the defaults. You can redirect incoming HTTP or SSL requests the internal server as HTTP, SSL or FTP requests. If redirected as an SSL request, you can specify a certificate for authenticating to the SSL Web server. When redirecting FTP requests the result is HTML-rendered pages representing the FTP site.

Also, you can restrict Web publishing rules to specific Client Address Sets or to specific users or security groups. An understanding of how ISA Server applies authentication to rules is important. Authentication is explained in the section *Authentication and Rules*.

Routing Rules

After the Web Publishing Rules processes a request, ISA Server applies the Routing Rules. The Routing Rules determine whether the request is retrieved directly from the origin server, from another proxy server, or redirected to a different destination server. When you redirect a request to another ISA Server, you can specify a backup route in the event that the primary route is unavailable.

Routing Rules also determine how responses are cached for future requests. You can configure the rules to either always serve requests from the cache, or only when a valid object is available in the cache. The specifics of routing rules are discussed in the section "Web Proxy Routing".

Web Proxy Routing

Web Proxy Routing allows requests that would otherwise be forwarded directly to the origin server, to be routed to upstream ISA Servers or redirected to alternate servers. Routing Rules determine whether requests are cached. Web Proxy Routing provide the capability to bridge requests from HTTP to SSL requests or SSL to HTTP.

Routing Requests

Routing Rules provide three methods for routing requests. You may retrieve the requests directly from the destination server; route requests to an upstream ISA Server or an array of ISA Servers; or redirect requests to a hosted site.

- **RETRIEVE THEM FROM THE SPECIFIED DESTINATION** This is the default setting. The Web Proxy Service generates a request to the origin server to retrieve requested resources. If the origin server is unavailable, caching rules determine whether an error response is returned to the client.
- **ROUTING THEM TO A SPECIFIED UPSTREAM SERVER** ISA Server can forward requests to upstream ISA

Servers in a chain. For example, this is useful for placing ISA Servers in remote locations on the network and routing requests through a central ISA Array. When routing requests, an upstream ISA Server or ISA Array is specified along with the HTTP and S-HTTP request ports. For an ISA Array, the routing rule is configured to poll the upstream server for the array configuration. If an upstream server requires authentication, it is specified in the routing rule. Also, you can configure a backup route to use if the upstream server is unavailable. The backup route can be another ISA Server or ISA Array, or it can be a direct connection to the destination server. When no backup route is specified, the request will fail if the primary route is unavailable.

- **REDIRECTING THEM TO A HOSTED SITE** When redirecting requests, ISA Server generates a new request using the hostname of the specified server in the routing rule. Redirected requests can use SSL Bridging. (SSL Bridging defined in the following “SSL Bridging” section)

Using Dial-Up

Dial-Up Entries are defined as Policy Elements in ISA Management. When defining a routing rule, you can specify whether or not Active dial-up entry is used for direct and upstream proxy routes.

Caching Requests

Routing Rules specify whether or not requests are cached and when the cached versions of content are used. When a Routing Rule processes a request, there are three possible settings for serving content from the cache.

- The default option is to use a valid version of the object from the cache. If no valid version of the object exists, a request is made to the origin server for the resource or the request is routed based on the routing rule.
- Another option is to use any version of the object from the cache. This means that even if an object expires, it will still be served from the cache. If the object does not exist in the cache, the resource will be retrieved from the origin server.
- Finally, a Routing Rule can specify to only serve objects from the cache. Any version of the object will be used. If the object does not exist in the cache, the Routing Rule will respond with an error to the client.

When the Web Proxy Service retrieves a resource from an origin server, the Routing Rules specify whether or not the content is stored in the cache. A routing rule can tell the service to store all objects in the cache, only non-dynamic objects or no objects. When a Routing Rule caches all objects, even those that are considered dynamic are cached.

SSL Bridging

ISA Server can encrypt and decrypt Secure Socket Layer (SSL) requests and redirect the request to the destination server. This capability is called *SSL Bridging*. ISA Server can either end or initiate an SSL Connection. There are three possible scenarios for SSL Bridging:

- **HTTP REQUESTS FORWARDED AS SSL REQUESTS** ISA Server encrypts the client’s request for an HTTP object and forwards it to the Web server. The Web server returns the encrypted object to ISA Server. ISA Server decrypts the object and then sends it to the client.

- **SSL REQUESTS FORWARDED AS SSL REQUESTS** ISA Server decrypts the client's request for an SSL object, then encrypts it again and forwards it to the Web server. The Web server returns the encrypted object to ISA Server. ISA Server decrypts the object and then sends it to the client.
- **SSL REQUESTS FORWARDED AS HTTP REQUESTS** ISA Server decrypts the client's request for an SSL object and forwards it to the Web server. The Web server returns the HTTP object to ISA Server. ISA Server encrypts the object and sends it to the client.

For example, SSL requests can be forwarded as HTTP requests in a reverse publishing scenario. In this case the Internet user makes an SSL request for an internally published Web site. Then ISA Server decrypts the request and forwards it as an HTTP request to the internal server.

Internal clients can also benefit from SSL Bridging. Normally, when a client makes an S-HTTP request to a destination server, the ISA Server uses SSL tunneling, which establishes a direct connection between the server and client. For clients that support secure communication with the ISA Server, requests are first decrypted at the ISA server. The ISA Server checks the cache to see if the resource is available locally. If not, the ISA Server makes a new request to the destination server. The new request may be HTTP or S-HTTP, depending on the routing rules.

HTTP Redirector

The Hypertext Transfer Protocol Redirector filters HTTP requests passing through the Firewall and forwards them the Web Proxy Service. The HTTP Redirector by default is enabled when ISA Server is installed in firewall or integrated mode only. The filter can be set to handle requests in three different ways.

- **REDIRECT TO LOCAL WEB PROXY SERVICE** The filter will intercept requests and redirect them to the Web Proxy Service. Requests will pass the Web routing rules and a response will be cached as specified. If the local Web Proxy Service is unavailable the filter can be configured to pass the request directly to the destination server.
- **SEND TO REQUESTED WEB SERVER** Requests are forwarded directly to the destination server. Web Proxy rules will not be processed and responses will not be cached.
- **REJECT HTTP REQUESTS FROM FIREWALL AND SecureNAT CLIENTS** This option will prevent HTTP requests from being passed through the firewall. The only option for a client to make an HTTP request is to direct the request to the Web Proxy Service.

Cache Array Routing Protocol

You can join together multiple ISA Enterprise Servers as a single logical cache in a grouping of ISA Enterprise Servers called an *array*. A queryless protocol, referred to as the Cache Array Routing Protocol, makes efficient caching possible in even the largest of enterprises.

CARP is queryless, making possible unlimited scaling of the array without negatively affecting network performance. CARP's high-speed hashing algorithm allows downstream clients or ISA Enterprise Servers to determine the precise server in an array to store and retrieve a cached object; thus, eliminating the duplication of cached objects.

How it Works

Cache Array Routing Protocol uses the *array membership list* stored in Active Directory to track the individual ISA Enterprise Servers. The individual servers may be configured as an array member during installation or may be promoted to an array at a later time. All ISA array members share the array membership list and are notified when a member is added or removed. You configure clients to request a configuration script from an upstream ISA Enterprise Server that is a member of the array. The configuration script contains both the hashing algorithm and the list of ISA Enterprise Servers that comprise the ISA array. Because a client downloads the script with HTTP, the server can set a TTL for the script. When the TTL expires, the client will update its version of the configuration script. Additionally, the configuration script can specify a backup route to use should the ISA Array become unavailable.

The hashing algorithm is built into the Downstream ISA Servers, so they do not download the configuration script. Instead they simply need the membership list of ISA Servers in the array and the load-balancing factor. This provides quicker computation for Downstream ISA Servers.

If a Web browser is not configured to use CARP, the requests can still be routed correctly. You can configure an ISA Array to resolve requests within the array before retrieving the content from the destination server. This allows the ISA Server to find the cached object within the array and return it to the Web browser.

The Hashing Algorithm

The configuration script contains a subroutine called *FindProxyForURL*. This subroutine contains the hashing algorithm for determining which ISA Enterprise Server in the ISA array will hold the cached object. The hashing algorithm spreads cached objects, and therefore requests, evenly across the array. As a result, all clients and ISA servers will look to the same location for an object as they all have the same configuration script.

While the configuration script determines the location of a cached object using a hashing algorithm, the following example demonstrates the process only. The numbers illustrated here are much smaller than the actual numbers used by the hashing algorithm:

Build table of ISA Servers with hashed values for each server's hostname and a load-balancing factor. The following table shows four ISA Servers in an array for Appliance Science, Inc. Each server is assigned a hash value based on the hostname.

ISA Server	Hash
ScienceApp1	13
ScienceApp2	8
ScienceApp3	5
ScienceApp4	28

Compute a hash value for the URL of the requested object. Here, a hash value is calculated for www.microsoft.com.

www.microsoft.com
19

Combine the hash value of the URL with the hash value of each ISA server.

www.microsoft.com

ISA Server	Hash	19
ScienceApp1	13	5
ScienceApp2	8	9
ScienceApp3	5	7
ScienceApp4	28	4

Multiply each combined hash value by a load factor for its particular ISA server to determine a score value for that ISA server. The diagram now shows the value obtained from combining the hash for www.microsoft.com with each hostname hash value and multiplied by a load factor.

		www.microsoft.com	
ISA Server	Hash	19	
ScienceApp1	13	5	
ScienceApp2	8	9	
ScienceApp3	5	7	
ScienceApp4	28	4	

Sort the ISA server list by highest score value and return the list to the client. ScienceApp2 is the preferred server for the URL www.microsoft.com, then ScienceApp3, ScienceApp1 and ScienceApp4 in order.

		www.microsoft.com	www.lycos.com	www.msn.com	www.compaq.com
ISA Server	Hash	19	14	5	2
ScienceApp1	13	5	6	10	4
ScienceApp2	8	9	2	7	5
ScienceApp3	5	7	4	3	10

ScienceApp4	28	4	7	8	1
-------------	----	---	---	---	---

The client attempts to contact the first server in the list. If the attempt fails, it continues through the list until a successful attempt is made. The following characteristics of the hashing algorithm improve performance:

- **DETERMINISTIC** The hashing algorithm tells the HTTP agent exactly where the object for a particular URL will be cached in an array. No additional queries to the array or among array members need to be made.
- **LOAD-BALANCED** You can adjust Load-balancing factors so that the more capable servers receive the most requests.
- **RELIABLE.** If a particular ISA server becomes unavailable, the HTTP agent falls back to the ISA server with the next highest score. As soon as the original server becomes available, requests are resolved appropriately.
- **SCALABLE** The number of servers in the array may be adjusted without causing network performance problems. The performance of the cache is always proportional to the number of servers in the array.
- **STABLE** Adding and removing servers will not cause all of the cached objects to shift between the servers. If an array is increased from four to five servers, approximately 1/5 of the cached objects will be shifted to the new server.

Requirements for CARP

The *Enterprise Edition* of ISA Server is required to implement an ISA Server Array and the ISA Server computer must be a member of a Windows 2000 domain. Additionally, before you can install any ISA Server as an array member you must install the ISA Server schema to the Active Directory. You install the ISA Server schema in Active Directory by using the Enterprise Initialization utility included with ISA Server.

The Enterprise edition allows ISA Server to store its configuration in the Active Directory. Because configuration information is in the Active Directory, the members of an array can all share the same configuration. All of the routing rules, publishing rules and content filters are shared among the array members. The configuration information includes the array membership list.

Array Routing Scenarios

ISA Server supports three scenarios for routing requests to increase performance for HTTP requests: hierarchical, distributed and combination routing. Hierarchical routing is used when requests are forwarded from downstream ISA Servers to a common ISA Server or ISA Array. Distributed routing uses the CARP algorithm to route requests within an array.

HIERARCHICAL ROUTING In this configuration, ISA Servers are placed close to clients on the network. Requests are first served from the local cache. If the local cache does not contain a valid object for the requested resource, the request is forwarded to a central ISA Server on

the network. The central ISA Server also attempts to serve the request from the local cache. If a valid object is not found in the cache, a request is made to the origin server.

DISTRIBUTED ROUTING ISA Server Arrays implement distributed routing. The route can be determined from the downstream agent or the route can be determined within the array. In other words, if the downstream agent does not use CARP, the ISA Server that accepts the request can determine which server within the array contains the cached object and route the request. The first ISA Server will not cache the response as the object is already contained within the array.

COMBINATION ROUTING This routing scenario implements both hierarchical routing and distributed routing. ISA Servers or ISA Arrays are placed close to clients on the network. When the request is received it is routed to a central ISA Array using CARP. This approach is both scalable on an enterprise level and location level. If a particular segment of the network experiences a high number of requests, that particular segment can be upgraded to an ISA Array.

Web Publishing with Arrays

An ISA Server Array can be used to publish an internal Web site and balance the load using distributed routing and CARP. Requests are received by one of the servers in the array. The receiving server uses the hashing algorithm to determine which server in the array contains the cached item and forwards the request to that server.

ISA Server Scenarios

The acceleration features of Microsoft Internet Security and Acceleration Server combine to provide a wealth of solutions for improving network performance. Depending on your organization's needs, you can choose the features that are appropriate for the specific challenges you face. Following are two scenarios that demonstrate how you can combine several different features of ISA Server Acceleration to meet the specific needs of two different sized organizations.

Medium Sized Business Scenario

Prodigy Marketing, Inc. is a regional marketing firm with headquarters in Columbus, Ohio and three small sales offices located in Cleveland, Cincinnati and Indianapolis. The headquarters has four hundred employees. Each of the three branch offices has approximately twenty employees. The branch offices connect to headquarters by 128K ISDN links. The headquarters connects to the Internet via a 256k link. Typically, the Internet is used for research on industry trade Web sites and sending and receiving e-mail.

Because the sales offices are commonly researching the same Web sites for a project, the IT department implemented a local ISA Server in cache mode for each office. Active Caching was scheduled to update content in the cache during times when the ISDN link was not in use. This spread out the usage of the link over time, reducing network congestion. The IT department also installed an ISA Server in integrated mode at headquarters. The headquarters ISA Server

is used to secure the network from inbound traffic and provide caching features for the main office.

Prodigy Marketing worked on several marketing projects using the Web as an advertising medium. This necessitated the hosting of Web servers internally to support rapid development and content management. To provide faster access to Internet users without increasing bandwidth, Prodigy Marketing located an ISA Server at an Internet Service Provider through a co-location program. This ISA Server is configured to publish the Web servers. The ISA Server is also scheduled to download content from the internally hosted Web servers twice a day. When Internet users make requests for Web content, the ISA Server serves the content from its local cache over the high-speed link of the ISP.

Large Enterprise Scenario

Appliance Science, Inc. is a large manufacturer of a wide range of products having both home and industrial applications. In order to compete in the global market, the operations managers of Appliance Science must integrate a worldwide supply chain. Cost-effective vendors in several countries provide parts and services. Additionally, Appliance Science provides prefabricated parts to other manufacturers as well as pursuing its own retail market strategy. In order to meet revenue goals in a competitively crowded marketplace, it is important that the supply chain operates smoothly and cost-effectively. Consequently, the IT department of Appliance Science decided to implement a Business-to-Business network using ISA Server as the connecting points for secure and accelerated access to their Web-based management system.

Appliance Science provides two methods for connecting to the supply-chain management system. The first option is for end users to connect to the management system using S-HTTP over the Internet. The second method, designed for larger partners, is to implement a private connection—either physically or using VPN technology—directly to the partner network. With the second method, Appliance Science strikes an agreement where it provides and manages the equipment for this connection at some cost to the partner. See Figure 4 Appliance Science's partner network including access methods for the supply-chain.

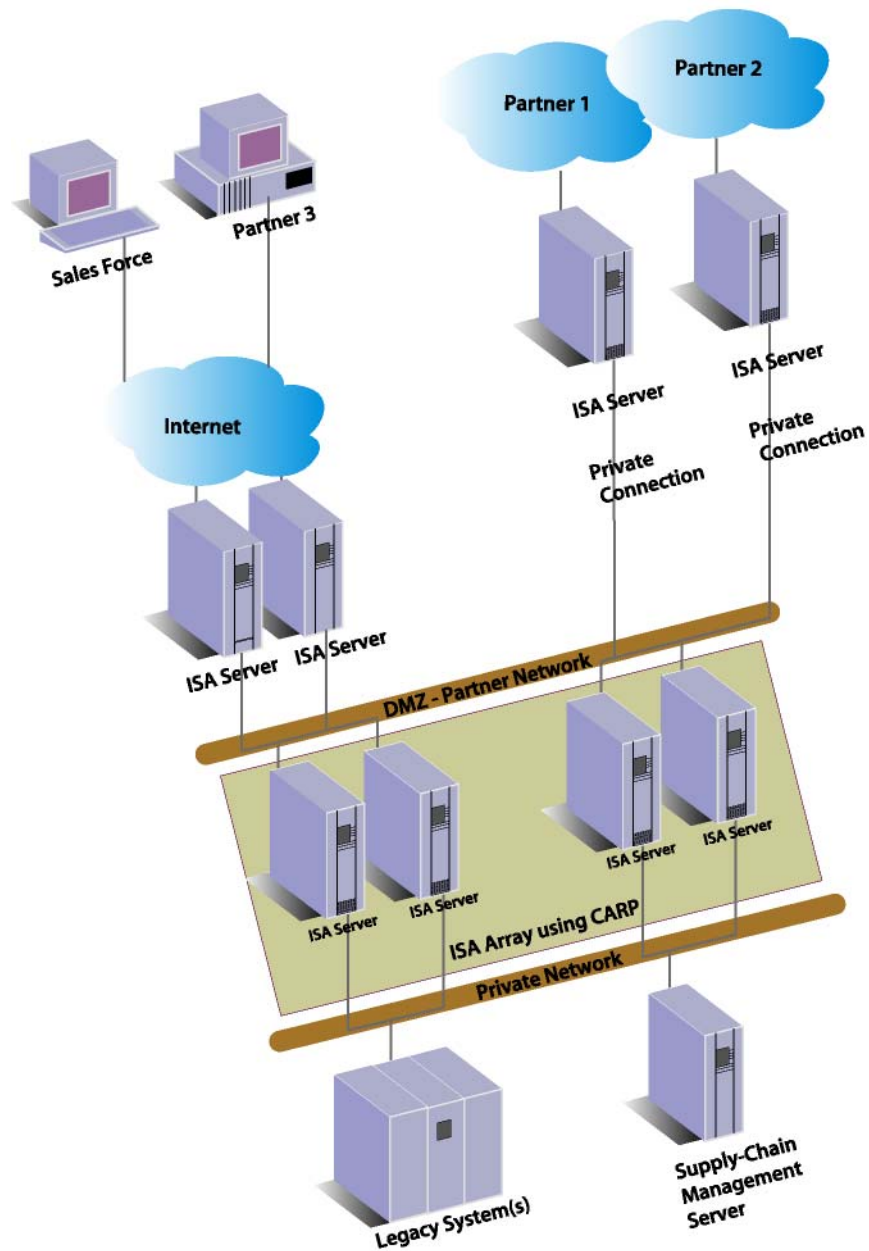


Figure 4 Appliance Science's partner network including access methods for the supply-chain.

Appliance Science implemented the partner network by placing a two-member ISA Server Array on the network with a connection to the Internet. This ISA Server Array is configured to publish the partner Web site to the Internet via Web Publishing Rules. The ISA Server Array will only accept requests that are using SSL and requires authentication before the request will be passed to the partner network. All requests are routed to the four-member ISA Server Array.

Appliance Science installed the two-member ISA Server Array connected to the Internet in integrated mode so it could also act as a firewall between the Internet and the partner network. By using IP packet filtering on the Internet interface and routing only authenticated requests to the Web server, the partner network is protected from unwelcome visitors.

For Appliance Science's larger partners, Appliance Science provides an ISA Server for making a private connection to the partner network. The connection can either be a physical link or a VPN connection. The ISA firewall protects both Appliance Science's partner network and the network of the partner making the connection.

The remote ISA server works similar to the Internet ISA Server by publishing the partner Web site to the local network. When users of the partner network make a request for content, the ISA Server attempts to serve the request from the local cache rather than sending it across the link. This allows Appliance Science to use lower cost links for connecting partners.

When the publishing ISA Servers cannot serve a request from the local cache, the requests are routed using CARP to the partner network ISA Array. The ISA Array not only enables the partner network to grow very large, but also increases the reliability of the partner Web site.

While each partner will have unique needs from the supply-chain Web site, Appliance Science knows that the requests from any particular partner will be very repetitive. Therefore the ISA Servers are configured for the Active Caching of frequently used content. This allows the ISA Servers to retrieve content during times of low network usage. End users find that the response time is consistently fast because most requests are served from the cache.

Additionally, Appliance Science provides large reports through the partner network to the supply-chain. Each partner has the option to schedule the download of these reports on a periodic basis. Many of the partners choose to download reports during the night when the link is not in use. Appliance Science configures a Scheduled Content Download job per the request of the partner.

Summary

Microsoft Internet Security and Acceleration Server 2000 provides value-added services to your organization's Internet solutions. Its wide range of caching options provide the flexibility required so that organizations of any size can accelerate their Internet access. ISA Server Standard Edition has caching features that are a perfect fit for small-to medium-sized organizations. For larger organizations, ISA Server Enterprise Edition provides the innovative benefit of CARP to ISA arrays for blazing-fast performance, along with Active Directory integration for easy configuration and enterprise management. No matter what the size of your organization, ISA Server will accelerate your Internet or intranet connections.

For More Information

www.microsoft.com/isaserver

The information contained in this document represents the current view of Microsoft Corporation on the issues discussed as of the date of publication. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication.

This White Paper is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS OR IMPLIED, AS TO THE INFORMATION IN THIS DOCUMENT.

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

The example companies, organizations, products, people and events depicted herein are fictitious. No association with any real company, organization, product, person or event is intended or should be inferred.

© 2001 Microsoft Corporation. All rights reserved.

Microsoft is a registered trademark of Microsoft Corporation in the United States and/or other countries.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.