Web Security On-prem

v8.5.x

PAC File Best Practices

Revision A

Contents

Introduction

This article looks at Proxy Auto-Configuration (PAC) files and their use and best practices with Forcepoint Web Security, including with Content Gateway, the on- premises web proxy, and with the Forcepoint Web Security Hybrid Module (which combines on-premises and cloud-based web protection).

Topics include:

Related concepts

What is a PAC file? on page 2
How is a PAC file consumed? on page 4
Using a PAC file with Content Gateway on page 6
Using a PAC file with Forcepoint Web Security and the Forcepoint Web Security Hybrid Module on page 7
Sample PAC file on page 7

Related information

PAC file best practices on page 9

What is a PAC file?

A Proxy Auto-Configuration (PAC) file is a JavaScript function definition that determines whether web browser requests (HTTP, HTTPS, and FTP) go direct to the destination or are forwarded to a web proxy server.

PAC files are used to support explicit proxy deployments in which client browsers are explicitly configured to send traffic to the web proxy. The big advantage of PAC files is that they are usually relatively easy to create and maintain.

The use of a PAC file is highly recommended with explicit proxy deployments of Forcepoint Web Security (for the Content Gateway—web proxy—component) and is **required** to support the Forcepoint Web Security Hybrid Module.

Note

These Internet resources provide excellent information, including PAC file JavaScript reference information, PAC file examples, and links to additional information:

This website: (http://www.findproxyforurl.com)

This Wikipedia article: (http://en.wikipedia.org/wiki/Proxy_auto-config)

These Microsoft knowledge base articles:

- An IE10 presentation that focuses on WPAD files: “Enable Automatic Detection and Configuration of Browser Settings” (http://technet.microsoft.com/library/jj647643.aspx)
- An older discussion with more information about PAC files: “Using Automatic Configuration, Automatic Proxy, and Automatic Detection” (http://technet.microsoft.com/en-us/library/dd361918.aspx)

A PAC file is a JavaScript function definition for **FindProxyForURL(url, host)**. The complexity of the function varies with the requirements of each organization.

A PAC file is:

- Flexible and extensible
- Supported by all popular browsers
- Easy to administer and maintain in any size network; however, as this paper explains, PAC files are easiest to administer when the browser is Internet Explorer
- Able to support mobile devices that use standard browsers

A PAC file can:

- Be stored on any server in your network. Small networks may store the file on the proxy itself, but large, enterprise-class networks should use a separate server for storing the PAC file
- Determine where Internet and intranet requests are routed
- Allow for exceptions in the form of bypassing the proxy for specified destinations
- Perform load distribution
- Handle proxy failover

Because PAC files are written in JavaScript, they support the structure, logic, and extensibility of JavaScript. See *JavaScript functions most commonly used in a PAC file*.

Related information
JavaScript functions most commonly used in a PAC file on page 10

Why use a PAC file?

In explicit proxy and Forcepoint Web Security Hybrid Module deployments of Forcepoint Web Security, using a PAC file fulfills several vital functions:

1) The PAC file provides critical security, ensuring that traffic is always proxied when it should be, while allowing secure requests to go direct to the destination.
   - Typically, Internet-bound HTTP, HTTPS, and FTP traffic is sent to the proxy.
   - Typically, intranet traffic goes direct to the destination.

- Exceptions can be made for internal or external sites that, for whatever reason, must go to or bypass the proxy.

2) The PAC file locks down the web browser's LAN egress configuration. The PAC file should be appropriately permission-protected so that end-users cannot change it. This is most easily accomplished when the PAC file is administered with a Group Policy Object. See *How do I configure a Group Policy so that Internet Explorer uses the PAC file?*

3) The PAC file provides a flexible, easy to maintain, script-driven method of controlling the routing of web requests.

4) The PAC file can include code that handles proxy load distribution and failover.

Note

It is important from an organizational security perspective that end users be prohibited from installing unapproved applications on their computers. Without such restrictions, users could install alternate browsers in an attempt to circumvent PAC controls. Within the organizational perimeter, by application of appropriate firewall rules, users should be forced to browse through the designated proxy server(s) only.

Related tasks

How do I configure a Group Policy so that Internet Explorer uses the PAC file? on page 14

How is a PAC file consumed?

Web browsers are configured to look for and read the PAC file each time the browser is started, or at regular intervals when pushed by a Group Policy Object (GPO).

When a URL request is made, the browser calls the **FindProxyForURL(url, host)** function to determine the request's routing disposition (to a proxy or direct to the destination).

Although the PAC file can be placed on each individual client machine, this is an uncommon and inefficient approach. The common arrangement is to host the PAC file on a server that all clients have access to. In smaller deployments, the Content Gateway host system can be used. In large enterprises the PAC file should be hosted on an existing (dedicated) web server that all clients have access to. Alternatively, the

Web Proxy Auto-discovery Protocol (WPAD) can be used to assist browsers in locating and retrieving the WPAD file, which contains the PAC function definition. **WPAD not an option with the Forcepoint Web Security Hybrid Module**; see *Using a PAC file with Forcepoint Web Security and the Forcepoint Web Security Hybrid Module*, below.

Note

Internet Explorer includes a feature called **Automatic Proxy Result Cache**. This feature can cause problems in environments that use a PAC file and have multiple proxy servers. See *What is Internet Explorer Automatic Proxy Result Cache?*

Related concepts
Using a PAC file with Forcepoint Web Security and the Forcepoint Web Security Hybrid Module on page 7
What is Internet Explorer Automatic Proxy Result Cache? on page 15

WPAD: Web Proxy Autodiscovery Protocol

The Web Proxy Autodiscovery Protocol (WPAD) is a method used by web browsers to locate the URL of a PAC file automatically, without manual configuration. WPAD can be used with Content Gateway, but is *not* an option with the Forcepoint Web Security Hybrid Module.

WPAD uses two methods to publish the location of the proxy configuration file—the Dynamic Host Configuration Protocol (DHCP), and the Domain Name System (DNS). A web browser using this method sends a query to the local DHCP server and if it does not send back the desired information, uses DNS.

For more information about using WPAD with Content Gateway, see Content Gateway Manager Help.

Specifying the PAC location

The exact mechanism for configuring a browser to locate and use a PAC file depends on the browser and network environment.

If you are using Microsoft Active Directory and Internet Explorer, you can automate the process via a **Group Policy Object** (GPO). *This is the recommended, best practice.* A GPO is a simple, versatile tool for configuring computers and user settings for members of Active Directory Domain Services.

- For configuration details, see *How do I configure a Group Policy so that Internet Explorer uses the PAC file?*
- For an introduction to Group Policy, see these Microsoft TechNet articles: Group Policy for Beginners and Managing Browser Settings with Group Policy Tools.

Unfortunately, use of GPO to configure other browsers is usually unsupported. Search the Internet for the availability of GPO add-on support for the browser used in your organization.

Users can also set up their browsers manually. The most popular browsers implement this feature in very similar ways.

Related tasks
How do I configure a Group Policy so that Internet Explorer uses the PAC file? on page 14

In Internet Explorer

Steps

1) Navigate to **Tools** > **Internet Options** > **Connections** > **LAN Settings**.

2) Select **Use automatic configuration script** field, and enter the following in the **Address** field:
http://%3CCG_Domain_Name_or_IP_Address%3E:8083/proxy.pac
Ensure everything under Proxy server panel is unchecked.

3) Click **OK**.

In Mozilla Firefox

Steps

1) Navigate to **Tools** > **Options** > **Advanced** > **Network** > **Connection** > **Settings**.

2) Select **Automatic proxy configuration URL** field, and enter:
http://%3CCG_Domain_Name_or_IP_Address%3E:8083/proxy.pac

3) Click **Reload**, and then click **OK**.

In Opera

Steps

1) Navigate to **Tools** > **Preferences** > **Advanced** tab.

2) Select **Network** > **Proxy Servers** > **Use Automatic proxy configuration** (check only this option).

3) Enter the location of the PAC file. For example, `file://c:/proxy.pac`

Next steps

See the documentation for your browser for details.

Using a PAC file with Content Gateway

Content Gateway is the on-premises web proxy component of Forcepoint Web Security.

For smaller enterprises where the user load requires only a single proxy, the Content Gateway host system may also host the PAC file. For larger enterprises that require more scale, the PAC file should be hosted on an existing (dedicated) web server that all of the client machines can access.

The Content Gateway manager provides a page for specifying and maintaining a PAC or WPAD file. In the Content Gateway manager, go to **Configure** > **Content Routing** > **Browser Auto-Config** > **PAC**.

For step-by-step instructions, see Content Gateway Manager Help.

If you have enabled SSL support to inspect HTTPS traffic as it transits Content Gateway, see Content Gateway Manager Help.

Important

If you have configured Content Gateway to use Integrated Windows Authentication to perform user authentication, you must specify the fully qualified domain name (FQDN) of the proxy whenever a request is directed to the proxy in the PAC file. For example:

```
wgc1.example.com:8080
```

Using a PAC file with Forcepoint Web Security and the Forcepoint Web Security Hybrid Module

Forcepoint Web Security with the Forcepoint Web Security Hybrid Module combines on-premises and cloud-based protection as needed. Typically, the on-premises software provides web protection for the main office or campus, while smaller regional offices or satellite locations send their Internet requests through the hybrid service in the cloud. The hybrid service is also useful for users who are off-network, such as telecommuters and those who travel for business.

In the Forcepoint Web Security Hybrid solution, the PAC file used to enable hybrid protection contains a number of global settings, and also allows you to configure sites that users can access directly without sending the request to the hybrid service (for example, intranet sites or organizational web mail).

All users are configured with a single PAC file. This PAC file applies whether the user is inside the network or outside the network. A common PAC file definition will direct users to the on-premises Content Gateway when they are inside the network, and to the hybrid service when they are outside the network. However, other configurations are possible.

For complete information on configuring interactions with the hybrid service, including information about customizing the PAC file, see Administrator Help and the sections it links to.

Sample PAC file

This example PAC file illustrates how to:

- Normalize the requested URL for pattern matching
- Bypass the proxy when the destination is a plain hostname (a hostname that does not include a domain)
- Bypass the proxy for a defined set of local domains
- Bypass the proxy for Windows Update
- Bypass non-routable addresses (RFC 3330)
- Send remaining HTTP, HTTPS, and FTP traffic to a specific proxy

Example:

```
function FindProxyForURL(url, host)
/* Normalize the URL for pattern matching
url = url.toLowerCase(); host = host.toLowerCase();

{
/* Don't proxy local hostnames */
if (isPlainHostName(host))
```

```
{
return 'DIRECT';
}

/* Don't proxy local domains */
if (dnsDomainIs(host, ".example1.com") ||
(host == "example1.com") ||
dnsDomainIs(host, ".example2.com") ||
(host == "example2.com") ||
dnsDomainIs(host, ".example3.com") ||
(host == "example3.com"))
{
return 'DIRECT';
}

/* Don't proxy Windows Update */
if ((host == "download.microsoft.com") ||
(host == "ntservicepack.microsoft.com") ||
(host == "cdm.microsoft.com") ||
(host == "wustat.windows.com") ||
(host == "windowsupdate.microsoft.com") ||
(dnsDomainIs(host, ".windowsupdate.microsoft.com")) ||
(host == "update.microsoft.com") ||
(dnsDomainIs(host, ".update.microsoft.com")) ||
(dnsDomainIs(host, ".windowsupdate.com")))
{
return 'DIRECT';
}
if (isResolvable(host))
{
var hostIP = dnsResolve(host);

/* Don't proxy non-routable addresses (RFC 3330) */
if (isInNet(hostIP, '0.0.0.0', '255.0.0.0') ||
isInNet(hostIP, '10.0.0.0', '255.0.0.0') ||
isInNet(hostIP, '127.0.0.0', '255.0.0.0') ||
isInNet(hostIP, '169.254.0.0', '255.255.0.0') ||
isInNet(hostIP, '172.16.0.0', '255.240.0.0') ||
isInNet(hostIP, '192.0.2.0', '255.255.255.0') ||
isInNet(hostIP, '192.88.99.0', '255.255.255.0') ||
isInNet(hostIP, '192.168.0.0', '255.255.0.0') ||
isInNet(hostIP, '198.18.0.0', '255.254.0.0') ||
isInNet(hostIP, '224.0.0.0', '240.0.0.0') ||
isInNet(hostIP, '240.0.0.0', '240.0.0.0'))
{
return 'DIRECT';
}

/* Don't proxy local addresses.*/
if (false)
{
return 'DIRECT';
}
}
if (url.substring(0, 5) == 'http:' ||
url.substring(0, 6) == 'https:' ||
url.substring(0, 4) == 'ftp:')
{
return 'PROXY wcg1.example.com:8080';
}
return 'DIRECT';
}
```

The following is a simple example of load distribution and failover using DNS. Search the Internet for other methods.

```
{
if (isInNet(myIpAddress(), "10.1.0.0", "255.255.0.0"))
{ return "PROXY wcg1.example.com:8080; " +
"PROXY wcg2.example.com:8080";
}
```

```
if (isInNet(myIpAddress(), "10.2.0.0", "255.255.0.0"))
{ return "PROXY wcg1.example.com:8080; " +
"PROXY wcg2.example.com:8080";
}

if (isInNet(myIpAddress(), "10.3.0.0", "255.255.0.0"))
{ return "PROXY wcg2.example.com:8080; " +
"PROXY wcg1.example.com:8080";
}

if (isInNet(myIpAddress(), "10.4.0.0", "255.255.0.0"))
{ return "PROXY wcg2.example.com:8080; " + "PROXY wcg1.example.com:8080";
}

else return "DIRECT";
}
```

PAC file best practices

Related concepts

JavaScript best practices for PAC files on page 9
How do I restrict the browsers allowed in my network to only those that can be configured with a PAC or WPAD file? on page 13
What is Internet Explorer Automatic Proxy Result Cache? on page 15
How do I specify a URL in a PAC file to bypass Content Gateway? on page 15

Related tasks

How do I configure a Group Policy so that Internet Explorer uses the PAC file? on page 14

Related information

JavaScript functions most commonly used in a PAC file on page 10

JavaScript best practices for PAC files

The JavaScript skills needed for most PAC file development are modest. Occasionally, an advanced understanding is needed. A good Internet resource is the website www.findproxyforurl.com.

Whether you are creating a new PAC file or assuming responsibilities for an existing file, these best practices are worth consideration. The list is inspired by and incorporates many entries from a blog post by Lee Harvey titled “Proxy Automatic Config (PAC) File Tips” (post no longer available online).

- Thoroughly review and understand the PAC file before making changes.
- Use the PAC or WPAD facility in Content Gateway to maintain the PAC or WPAD file. If you choose to edit the file separately, be sure to use a text editor that does not add or change formatting (e.g. vi, notepad, etc.).
- Comment the code consistent with programming best practices. Successors should have no questions about the intent of the code.
- Keep the file as small and efficient as possible.
- Validate support for built-in JavaScript functions before using them.

- Check URL and host parameters before using them.
- Check simple rule exceptions first.
- Place high-probability checks near the top.
- Use efficient regular expressions, and avoid capturing matches that will not be used.
- Because “return” is immediate, avoid using “else” with “if” statements.
- Single-line if() statements do not require begin { and end } brackets.
- Carefully consider the use (overuse) of isResolvable(), dnsResolve(), and isInNet() for potential DNS performance issues.
- Avoid using external or global variables and functions.
- Because .pac files are text and can be downloaded and viewed by anyone, use appropriate file permissions and avoid revealing secrets.
- When possible, sort lists of IP addresses and/or domains to ease future maintenance efforts.
- When possible, group common return values into single conditional if() checks.
- For single proxy server environments, return the proxy’s static IP address to bypass the DNS lookup overhead.
- Test all conditions and exceptions used in your .pac file prior to deployment. Verify that your JavaScript is error-free.

JavaScript functions most commonly used in a PAC file

From “PAC Functions Explained” (http://findproxyforurl.com/pac- functions)

isPlainHostName()

This function returns true if the hostname contains no dots (for example, “http:// intranet”).

It is useful when applying exceptions for internal websites that may not require resolution of a hostname to IP address to determine if they are local.

Example:

```
if (isPlainHostName(host)) return "DIRECT";
```

dnsDomainIs()

Evaluates hostnames and returns true if hostnames match. It is used mainly to match individual host names for exceptions.

Example:

```
if (dnsDomainIs(host, ".google.com")) return "DIRECT";
```

localHostOrDomainIs()

Evaluates hostname and only returns true if an exact hostname match is found. Example:

```
if (localHostOrDomainIs(host, "www.google.com")) return "DIRECT";
```

isResolvable()

Attempts to resolve a hostname to an IP address and returns true if successful. WARNING - This may cause a browser to temporarily hang if a domain is not resolvable.

Example:

```
if (isResolvable(host)) return "PROXY proxy1.example.com:8080";
```

isInNet()

This function evaluates the IP address of a hostname and, if within a specified subnet, returns true. If a hostname is passed, the function will resolve the hostname to an IP address.

Example:

```
if (isInNet(host, "172.16.0.0", "255.240.0.0")) return "DIRECT";
```

dnsResolve()

Resolves hostnames to an IP address. This function can be used to reduce the number of DNS lookups.

Example:

```
var resolved_ip = dnsResolve(host);
if (isInNet(resolved_ip, "10.0.0.0", "255.0.0.0") ||
isInNet(resolved_ip, "172.16.0.0", "255.240.0.0") ||
isInNet(resolved_ip, "192.168.0.0", "255.255.0.0") ||
isInNet(resolved_ip, "127.0.0.0", "255.255.255.0"))
return "DIRECT";
```

myIpAddress()

Returns the IP address of the host machine. Example:

```
if (isInNet(myIpAddress(), "10.10.1.0", "255.255.255.0"))
return "DIRECT";
```

dnsDomainLevels()

This function returns the number of DNS domain levels (number of dots) in the hostname. Can be used to exception internal websites which use short DNS names.

Example:

```
if (dnsDomainLevels(host) > 0)
return "PROXY proxy1.example.com:8080";
else return "DIRECT";
```

shExpMatch()

Attempts to match hostname or URL to a specified shell expression and returns true if matched.

Example:

```
if (shExpMatch(url, "*vpn.domain.com*") ||
shExpMatch(url, "*abcdomain.com/folder/*"))
return "DIRECT";
```

weekdayRange()

Can be used to specify different proxies for a specific day range. Note: the example employs “proxy1.example.com” Monday through Friday.

Example:

```
if (weekdayRange("MON", "FRI"))
return "PROXY proxy1.example.com:8080"; else return "DIRECT";
```

dateRange()

Can be used to specify different proxies for a specific date range. Note: The example employs “proxy1.example.com” January through March.

Example:

```
if (dateRange("JAN", "MAR"))
return "PROXY proxy1.example.com:8080"; else return "DIRECT";
```

timeRange()

Can be used to specify different proxies for a specific time range. Note: The example employs “proxy1.example.com” 8 AM to 6 PM.

Example:

```
if (timeRange(8, 18))
return "PROXY proxy1.example.com:8080";
else return "DIRECT";
```

Potential PAC function issues

From the Wikipedia article titled “Proxy auto-config” (http://en.wikipedia.org/wiki/Proxy_auto-config) with additions from the Forcepoint Knowledge Base.

A PAC file may have the following limitations:

dnsResolve

The function dnsResolve (and similar other functions) performs a DNS lookup that can block your browser for a long time if the DNS server does not respond.

If you cache proxy auto-configuration results by domain name in your browser (such as Microsoft Internet Explorer) instead of the path of the URL, it limits the flexibility of the PAC standard. Alternatively, you can disable caching of proxy auto- configuration results by editing the registry.

It is recommended to always use IP addresses instead of host domain names in the **isInNet** function for compatibility with other Windows components that make use of the Internet Explorer PAC settings, such as .NET 2.0 Framework. For example,

```
if (isInNet(host, dnsResolve(sampledomain) , "255.255.248.0"))
// .NET 2.0 will resolve proxy properly
if (isInNet(host, sampledomain, "255.255.248.0"))
// .NET 2.0 will not resolve proxy properly
```

The current convention is to fail over to the direct connection when a PAC file is unavailable.

When switching quickly between network configurations (for example, when entering or leaving a VPN), dnsResolve may give outdated results due to DNS caching.

For instance, Firefox usually keeps 20 domain entries cached for 60 seconds. This may be configured via the network.dnsCacheEntries and network.dnsCacheExpiration preference variables. Flushing the system's dns cache may also help, (such as by using the **sudo service dns-clean start** in Linux).

myIpAddress

The myIpAddress function has often been reported to give wrong or unusable results (for example, 127.0.0.1, the IP address of the localhost). It may help to remove any lines referring to the machine hostname on the system's host file (such as /etc/hosts on Linux).

Also, when the browser is Firefox 3 or higher, and the operating system has IPv6 enabled, which is the default in Windows 7 and Vista, the myIpAddress function returns the IPv6 address, which is not usually expected nor programed for in the PAC file. For a discussion of solutions, see this Mozilla forum discussion.

Others

Further limitations are related to the JavaScript engine on the local machine.

Note

Some versions of Java have had problems with common proxy PAC file functions such as isInNet(). Please review the Java open issues in the release notes for the versions of Java used by your client browsers.

How do I restrict the browsers allowed in my network to only those that can be configured with a PAC or WPAD file?

If you are using Microsoft Active Directory and Internet Explorer, the recommended approach is to use a **Group Policy Object** (GPO). Windows Group Policy is designed for centralized IT control and configuration of Windows computers that are members of Active Directory Domain Services.

For configuration details, see *How do I configure a Group Policy so that Internet Explorer uses the PAC file?*. For an introduction to Group Policy, see these Microsoft TechNet articles: Group Policy for Beginners and Managing Browser Settings with Group Policy Tools.

Most other browsers will consume a PAC file but do not provide support for GPO. This makes it much more challenging for administrators to control the configuration and use of alternate browser (Firefox 3 offered a GPO

add-on, but Firefox 3 is long gone). Search the Internet for tools and strategies available for your organization's chosen browser.

Note

In addition to controlling which browsers are allowed and managing their configuration, it is essential that proper firewall policy is in place. No traffic should be allowed to go direct to the Internet, bypassing the proxy, unless it is explicitly allowed by policy.

Related tasks

How do I configure a Group Policy so that Internet Explorer uses the PAC file? on page 14

How do I configure a Group Policy so that Internet Explorer uses the PAC file?

Steps

1) Log on to a server in the domain, and, with administrative permissions, open **Start** > **Programs** > **Administrative Tools** > **Active Directory Users & Computers** and expand your domain.

2) Right click the top-level domain or **Organizational Unit** where the policy should be applied, select **Properties**, and then select the **Group Policy** tab.

3) Create a "GPO" and give it a meaningful name.

4) Edit the GPO from the following location: **User configuration** > **Windows Settings** > **Internet Explorer Maintenance** > **Connection** > **Automatic Browser Configuration**

5) Select **Enable Automatic Configuration**.

6) Under **Auto-proxy URL (.JS, .JVS, or .PAC file)**, enter the path to the PAC file.
   - If you are running the Forcepoint Web Security Hybrid module, you will find the path for the PAC file located in the Forcepoint Web Security module of Forcepoint Security Manager under **Settings** > **Hybrid Configuration** > **User Access** > **Proxy Auto-Configuration (PAC)**. It will look something like this:

     http://hybrid-web.global.blackspider.com:8082/proxy.pac?p=

7) In **Automatically configure every field**, specify how often the web browser should query for the auto-configuration. For example, if you enter 240 minutes, every 4 hours the web browser checks for an updated PAC file. If you leave this field blank or set it to “0” the web browser is only configured when it is started.

Note

Forcepoint Security Manager clients using Internet Explorer pick up the settings in this GPO the next time that group policy refreshes, which by default is every 90 minutes for clients, and every 5 minutes for domain controllers (or the next time a user logs off and on again). You can change the refresh interval in the default domain policy, or by going to a particular client and entering the following at the command prompt:

```
gpupdate /force
```

Also note that if the GPO is not applying the settings to the browser, then it is possible that another GPO is being applied that contains different settings; raising the link order for the new GPO should resolve the problem.

What is Internet Explorer Automatic Proxy Result Cache?

With most browsers, the PAC file **FindProxyForURL()** function is called every time a request is made. However, versions of Internet Explorer since 5.5 include a feature called **Automatic Proxy Result Cache** that caches the hostname of the requested URL and the proxy that is returned by the **FindProxyForURL()** function (as <hostname, server>). This has the advantage of minimizing calls to **FindProxyForURL()**, but imposes 2 important limits:

1) Because the Automatic Proxy Result Cache is indexed by hostname, it is impossible for a PAC file to distribute traffic to distinct proxy servers based on any part of a URL in addition to the hostname. In other words, it is impossible to direct traffic to different proxy servers based on the path portion of URLs on a single host.

2) Because Automatic Proxy Result Cache caches the hostname/first_server pair, rather than the full results of the **FindProxyForURL()** function (full URL and multiple servers, if so scripted), the failover from one proxy to another does not occur in the event of a problem, even if the **FindProxyForURL()** function returned a list of proxy servers.

This feature is discussed in more detail in the Microsoft Knowledge Base article titled How to disable automatic proxy caching in Internet Explorer.

How do I specify a URL in a PAC file to bypass Content Gateway?

PAC files are easily modified to specify any number of URLs that will bypass the proxy. Such entries are often referred to as *exceptions*.

Most PAC files already have 1 or more exceptions. A common exception is for internal networks. For example:

```
if (isInNet(host, "192.168.0.0", "255.255.0.0"))
{return "DIRECT";}
```

An entry for an external site might look like:

```
if (shExpMatch(url, "*.webex.com/*"))
{return "DIRECT";}
```

Warning

Some versions of Java have had problems with common proxy PAC file functions such as ***isInNet()***. Please review the Java open issues in the release notes for the versions of Java used by your client browsers.

Published 02 September 2024