Connecting to Citrix MetaFrame Presentation Server through Proxy Servers

Jay Tomlin
Citrix Systems
January 2005

Table of Contents

Overview

This paper describes how to use Citrix MetaFrame Presentation Server and MetaFrame Secure Access Manager when the client is behind a proxy server or the farm is protected by a reverse proxy server.

Working knowledge of Citrix MetaFrame Presentation Server, ICA Clients, Web Interface and Secure Gateway is assumed.

Types of Proxy Servers

There are several different types of proxy servers that perform a range of duties. This section defines the following types of proxy servers and discusses their usability with ICA:

- Forward web proxy
- Secure web proxy
- SOCKS proxy
- Reverse web proxy
- Transparent proxy
- Network Address Translation

Forward Web Proxy

A forward web proxy is used to request web pages on behalf of internal clients who otherwise would have no route to the Internet. The client's HTTP request is sent to the proxy server and terminated by the proxy server, and then the proxy server initiates a new HTTP request on the client's behalf according to access rules defined by the proxy server administrator.

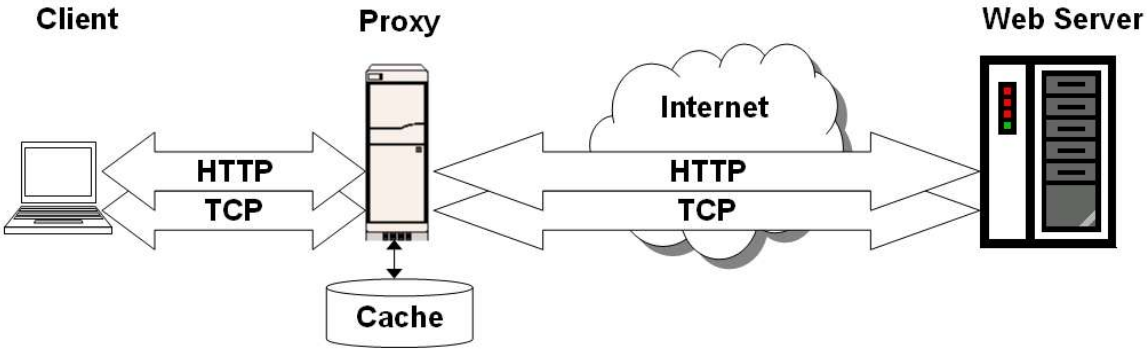


Figure 1 - Forward Web Proxy

Content cache

HTTP content may be cached at the proxy and served locally to clients in order to minimize Internet traffic. For example, suppose 100 internal users all point their web browsers to www.citrix.com at approximately the same time. The proxy server could request the page from citrix.com just once for the first request, store the results in its local cache, and serve the remaining 99 users the result from its cache. The HTTP protocol defines a header called If-Modified-Since which the proxy server could send to the web server to ensure that the content in its cache is up-to-date.

Proxy server acts as HTTP client

The HTTP connection to the target web server is initiated by the proxy server, not by the end user. Proxy servers reproduce the HTTP headers included with the original client request and may insert additional headers informing the target web server that the request is being proxied. For example, the following headers may be appended to the outgoing request from the proxy server to the web server:

Via	A string identifying the proxy server address, platform and version
X-Forwarded-For	The internal IP address of the client on whose behalf the request is being made

At the web server, this information will appear in HTTP server environment variables HTTP_VIA and HTTP_X_FORWARDED_FOR, respectively. For example, suppose a client with an internal address of 10.9.13.21 connects through a Squid proxy server to an IIS web server on the Internet. The IIS session will include the following server variables:

HTTP_VIA: 1.0 squid01.company.net:8080 (squid/2.5.STABLE3-NT-CVS)
HTTP_X_FORWARDED_FOR: 10.9.13.21

Note: These headers will not be present for HTTPS connections.

Popular products that may be used as a forward web proxy include:

- Microsoft Proxy Server 2.0
- Microsoft ISA Server
- Squid
- Netscape Proxy Server
- Apache

A forward web proxy service cannot be used for ICA or ICA/SSL traffic because it is designed specifically to handle HTTP requests.

Secure Web Proxy

A secure web proxy negotiates an end-to-end Layer-4 connection between the client and a web server, allowing an SSL or HTTPS session to be created. Most forward web proxy servers can also act as a secure web proxy, allowing users to connect to external servers using HTTP or HTTPS.

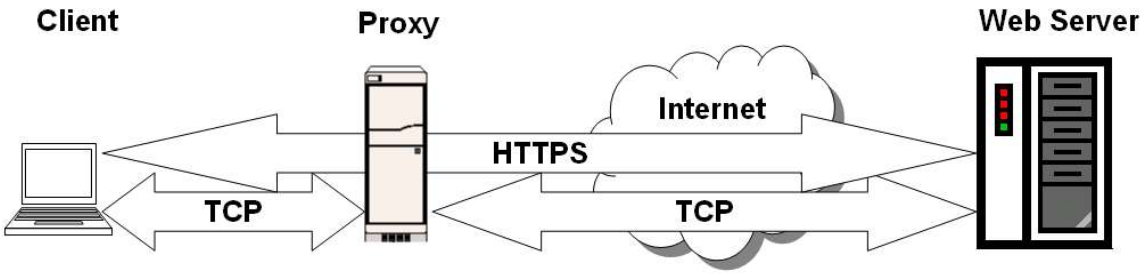


Figure 2 - Secure (HTTPS) Web Proxy

This type of proxy service is also known as an HTTPS proxy or an SSL Tunneling proxy. In order to request a secure tunnel, the client sends a special **CONNECT** request to the proxy server instead of the normal GET request used for HTTP. A CONNECT request contains only the destination server address and desired TCP port, normally 443. If the request is permitted by the proxy server rules, the proxy server initiates a new TCP session to the target server port and routes the client's traffic directly to the target server instead of initiating a new request on the client's behalf. When using the CONNECT method, a secure virtual tunnel is established between the client and the target server; the proxy essentially behaves like a Layer-3 router.

Proxy servers that support the CONNET method include:

- Microsoft Proxy Server 2.0
- Microsoft ISA Server
- Squid
- Netscape Proxy Server
- Apache

This type of proxy service can be used for ICA or ICA/SSL, because the Layer-4 contents are not inspected or cached. The proxy server treats ICA traffic as it would HTTPS traffic.

Allowing the CONNECT method to non-standard ports

Using the CONNECT method bypasses proxy security rules based on URL, and prevents traffic from being meaningfully inspected. For these reasons, most proxy servers limit the outbound ports to which the CONNECT method is allowed. By default, proxy servers generally allow the CONNECT method only when the destination server port is one of the following:

443 – HTTPS – Secure HTTP

563 – NNTPS – Secure NEWS

SOCKS Proxy

SOCKS is a generic TCP wrapping protocol that relays TCP traffic from one network to another.

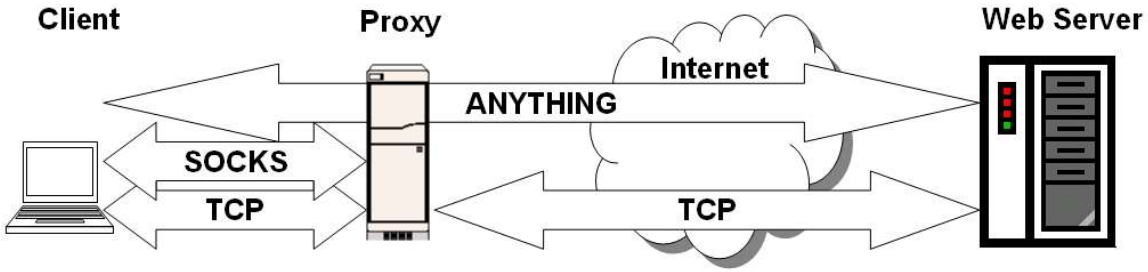


Figure 3 - SOCKS Proxy

Any type of TCP data can be sent through a SOCKS proxy server, including HTTP, HTTPS, ICA, FTP, database queries, e-mail or instant messaging. However, most SOCKS proxy servers do not support the transfer of UDP or ICMP traffic, making it impossible to ping a host located on the other side of a SOCKS proxy.

There are two widely used versions of the SOCKS protocol:

- SOCKS 4 – Does not support authentication
- SOCKS 5 – Supports user authentication and access control lists

Note that Secure Gateway for MetaFrame is a SOCKS proxy server that has been modified to expect SOCKS traffic wrapped in SSL and accompanied by a ticket produced by a Secure Ticket Authority. When communicating with a secure gateway, the ICA client wraps each ICA request into a SOCKS request, encrypts the request with SSL and forwards the request to the secure gateway server.

A SOCKS version 4 or 5 proxy service can be used for ICA or ICA/SSL. A SOCKS proxy server treats ICA traffic as it would any other TCP traffic. Prior to version 6.30, the ICA clients were only able to traverse a SOCKS proxy server. Support for Secure Proxy traversal was introduced in version 6.30. The ICA Client can automatically detect the SOCKS version if necessary.

Reverse Web Proxy

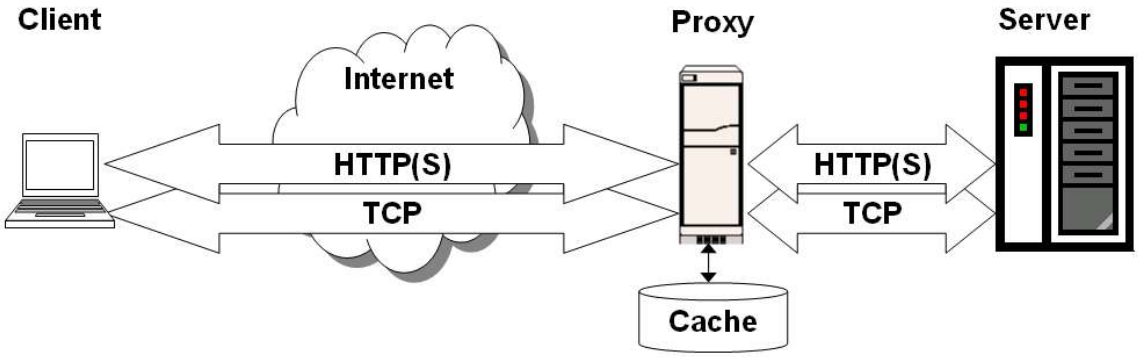


Figure 4 - Reverse Web Proxy

Examples include:

- Microsoft ISA Server
- Apache proxy_mod module
- Sun iPlanet
- Hardware load balancers with "SSL Acceleration" features

Stateful inspection or SSL acceleration routines that assume incoming traffic is HTTPS will cause problems when the traffic is ICA in SSL. Notice in the figure above that the HTTPS connection from the client is terminated at the proxy server, and a new HTTP(S) connection is initiated between the proxy and the target server. This setup implies that the destination server is an HTTP server: neither MetaFrame Presentation Server nor Secure Gateway is an HTTP server.

All-purpose Reverse Proxy (NAT)

A firewall that supports Network Address Translation (NAT) can be thought of as an all-purpose reverse proxy server. Clients connect to the firewall address and their traffic is transparently forwarded to a different host on the secure network. Network Address Translation is powerful because it requires no client logic and works with any protocol. The only requirement is that the client must connect to the firewall address instead of the actual target server address.

NAT offers little or no security, since all network traffic arriving at the firewall is forwarded to the configured destination server. Often, NAT is used when joining disparate trusted networks as an alternative to re-addressing or subnetting.

Transparent Proxy

The term *Transparent Proxy* is used to refer to a proxy service that runs locally on a client machine, intercepts outbound network traffic, and routes the traffic through a proxy server. Applications running on the client device need not be configured with the address of the target proxy server, because the traffic interception takes place at a relatively low layer of the network stack.

For example, the Advanced Gateway Client included with MetaFrame Secure Access Manager 2.2 is a secure transparent proxy that intercepts traffic at the WinSock level. This design allows a client application such as Outlook.exe to route traffic through the proxy without any explicit application configuration. In this example, Outlook is not “aware” that its traffic is being intercepted and redirected through a secure tunnel.

The Citrix Extranet Client for Windows was another example of a transparent localhost proxy.

Configuring a Proxy Server to support ICA or ICA/SSL

Some considerations when configuring a proxy server to accommodate ICA traffic include the following:

- If Secure Gateway is not used, the ICA client will attempt to connect through the proxy server to a destination port 1494 (version 7.x or earlier clients) or 2598 (version 8.x clients with Session Reliability enabled). The proxy server access control rules will need to be modified to allow the CONNECT method on these ports.

ICA is not HTTP

Do not assume that ICA traffic will behave like HTTP traffic. The table below demonstrates some key differences between ICA and HTTP.

Property	HTTP	ICA
Default TCP port	80	1494
TCP Sessions	4 or more	1
Typical session duration	5-30 seconds	1-8 hours
Content	HTML, form data, files, images	Mouse movements, keystrokes, screen drawing instructions, files, clipboard data, print jobs, audio streams, video streams, smart card data
Data characteristics	Plain text (HTML/XML) or binary (images, files)	Always binary
Communication Model	Client requests data, server responds to the request. Requests are typically much smaller than the response.	Client and server both receive and respond to requests.
Packet size	When downloading a large amount of data, Layer-1 frames are filled to capacity	To minimize perceived latency, many small updates are sent immediately resulting in a large number of small Layer-1 packets
Caching	Content may be cached by a proxy server and shared among several clients	All traffic is unique to each session and cannot be cached
Visible descriptors	Server address:port File extension MIME types HTTP Headers Cookies	Server address:port

With so many network appliances and software packages focused on delivering HTTP content, it is important to remember that not all traffic is HTTP traffic. Citrix ICA traffic is not HTTP and should be managed separately from HTTP traffic.

ICA/SSL is not HTTPS

When ICA traffic is encrypted using SSL and forwarded to a Secure Gateway server or the Citrix SSL Relay Service, the target port typically changes from 1494 to 443. However, all of the differences enumerated above still apply; devices intended to manage HTTPS traffic may not perform properly for ICA/SSL traffic.

Here's one example of this: some firewalls will reset a TCP 443 session if no traffic has been sent through that session for 10 minutes. For HTTPS, this is reasonable because HTTP communications are not characterized by 10-minute delays. The client requests a resource and the server responds immediately, then the session can be closed. Within an ICA session, 10 minutes of idle time might typically occur for a variety of reasons, for example if the user switches to a different application or walks away from her desk for a short break. Therefore steps may need to be taken to ensure that either the firewall timeout for port 443 is increased, or keep-alives are enabled that prevent the ICA session from ever becoming complete idle. See CTX435418 – Troubleshooting Disconnected Sessions in Citrix Secure Gateway.

Access Control Rules

When users will connect through a client-side proxy server in order to reach a MetaFrame Presentation Server, the proxy server must be configured to allow access to the desired MetaFrame servers on the appropriate port(s). In all cases, the client will use the CONNECT method for ICA or ICA/SSL traffic. Configuration of access control rules at the proxy will vary according to several variables:

FQDN or IP?

When adding access control rules at the proxy server, the address format of the MetaFrame Presentation Servers or Secure Gateway servers will vary according to the client's ability to resolve the Fully Qualified Domain Name (FQDN) of the target server.

When directed to connect to a fully-qualified domain name, the ICA client by default attempts to resolve that FQDN to an IP address. See the entry for **ProxyUseFQDN** below for details on how to reverse this behavior. If the client is able to resolve the FQDN to an IP address, then the proxy server is asked to connect to the resolved IP address. If the client is unable to resolve the target FQDN, then the proxy server is asked to connect to the FQDN and is expected to resolve the FQDN to an IP address.

Therefore, when client machines are unable to resolve the FQDN of a target MetaFrame or Secure Gateway server, base access control rules on the target FQDN and ensure that the proxy server is able to resolve the target FQDN to an IP address (via DNS or a hosts file entry). Otherwise, base the access control rules on the target server's IP address.

Which destination port?

The default configuration of most client-side proxy servers will allow the **CONNECT** method when the destination port is 443 (HTTPS). For this reason, it is recommended to configure Secure Gateway servers always to listen on port 443. If users will connect through a client-side proxy server to a Secure Gateway server, enable access on port 443.

If users will connect through a client-side proxy server to a MetaFrame Presentation Server without using Secure Gateway, enable access on port 1494 and/or 2598. One of these two TCP ports will be used for ICA connections according to the table below:

Connection	Connection Port
Win32 ICA Client version 8.0 or later with Session Reliability enabled	2598
Win32 ICA Client version 8.0 or later with Session Reliability disabled	1494
Win32 ICA Client version 7.2 or earlier	1494
Non-Windows ICA Clients through version 8.0	1494

Failing 2598, try 1494

If the 8.0 Win32 ICA Client with Session Reliability enabled cannot connect to the Citrix XTE Service on port 2598, the client abandons the session reliability feature and attempts a direct ICA connection to port 1494. Therefore if a proxy server is configured to allow port 1494 but not port 2598, users behind the proxy will still be connected but with the Session Reliability feature disabled. This failover behavior should not cause any visible error messages at the client—only a slight delay in connection time—but may generate “Access denied” entries in the proxy server access logs for each failed connection attempt through port 2598.

For best results, allow the CONNECT method to ports 1494 and 2598 if clients behind the proxy server need to connect directly to MetaFrame Presentation Server 3.0 or later (without using Secure Gateway).

Configuring Program Neighborhood Proxy Settings

To configure proxy settings for an application set or a Custom ICA Connection using the Program Neighborhood client, edit the properties of the application set or connection definition and click **Firewalls…**

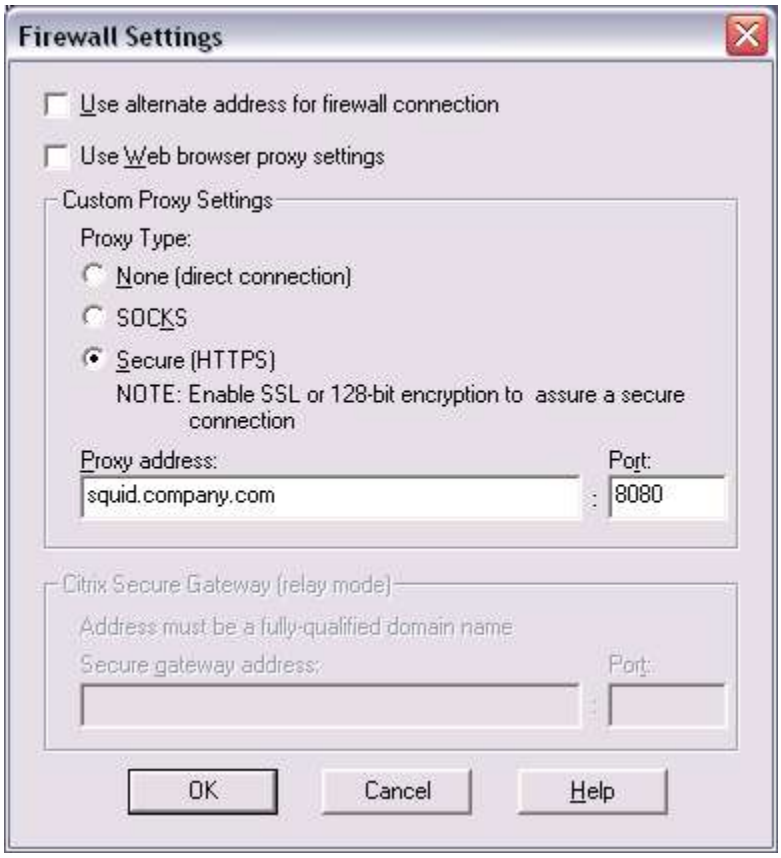


Figure 5 - Firewall settings in the Program Neighborhood client

Use alternate address for firewall connection enables access to published resources behind a NAT firewall. If selected, you must add the external address of the server running MetaFrame Presentation Server to the Address List and you must set the corresponding firewall address on each MetaFrame server using the **ALTADDR /SET** command.

Use Web browser proxy settings applies the proxy server settings configured for your default Web browser. This is the default setting. The process by which the ICA client determines your default browser's proxy settings is described in the Automatic Proxy Detection section below.

Use **Custom Proxy Settings** to override the web browser proxy settings:

- **None (direct connection)** connects to the server using a direct connection, without going through a proxy server.
- **SOCKS** Connects to the server through a SOCKS proxy server. You must enter the IP address and the port number of the SOCKS proxy server. The SOCKS version (4 or 5) will be auto-detected.
- **Secure (HTTPS)** connects to a server through a secure (HTTPS) proxy server. You must enter the IP address and the port number (if other than 8080) of the secure (HTTPS) proxy server. This is the most common type of client-side proxy server.

ICA Client Proxy Parameters

Editing the proxy settings as described in the previous section writes corresponding parameters to the user's ICA client configuration file, **APPSRV.INI**. Entries in APPSRV.INI may be overridden if the same entries appear in an ICA file. The following section describes what proxy-related entries may be written to APPSRV.INI or ICA files.

ProxyType

The type of proxy server that the ICA Client will use for an outbound connection.

Examples:

- ProxyType=Auto
- ProxyType=Secure

Possible values:

- **None** – Do not use a proxy server.
- **Auto** – Detect the proxy settings of the default web browser
- **SOCKS** – Use a SOCKS proxy server and auto-detect the SOCKS version
- **SOCKSv4** – Use a SOCKS version 4 proxy server (no authentication)
- **SOCKSv5** – Use a SOCKS version 5 proxy server (supports authentication)
- **Secure** – Use a Secure Proxy server (a.k.a. SSL Tunneling, uses the CONNECT method to negotiate an end-to-end tunnel between client and server)
- **Script** – Download and interpret the JavaScript Proxy Auto Configuration (PAC) file specified by the ProxyAutoConfigURL parameter

ProxyHost

The location and port of the proxy server.

Examples:

- ProxyHost=squid.company.com:3128
- ProxyHost=10.12.2.1:8080

Notes:

- If the port number is not included, the ICA Client defaults to port 8080 for Secure Proxy servers and port 1080 for SOCKS proxy servers.

ProxyBypassList

A semicolon- or comma-separated list of server addresses for which the client should not use a proxy server.

Examples:

- ProxyBypassList=*.company.net;10.12.*.*

Notes:

- ProxyBypassList parameter is ignored if ProxyType=Auto or ProxyType=None. When ProxyType=Auto, the bypass list should be configured in the default web browser.
- An asterisk may be used as a wildcard character to bypass a group of servers, such as *.company.com or 192.18.*.*; 10.*.*.*

ProxyAutoConfigURL

The location of a Proxy Auto Configuration (PAC) script to be downloaded and interpreted by the ICA Client.

Examples:

- ProxyAutoConfigURL=http://192.168.0.1/proxy.pac
- ProxyAutoConfigURL=http://wpad/wpad.dat

Notes:

- This parameter is ignored unless ProxyType=Script.
- The URL must begin with http:// or https://. URLs that use file:// or other protocols are not supported.
- Anonymous access to the PAC script should be enabled on the web server.

ProxyUsername, ProxyPassword

The username and clear-text password to be sent to the proxy server for automatic authentication.

Example:

- ProxyUsername=jsmith
  ProxyPassword=secret!99

Notes:

- The Win32 client supports NTLMv2 and Basic authentication; all other clients support only Basic authentication.
- When using Basic authentication, usernames and passwords traverse the network as Base64-encoded text. When using NTLM authentication, usernames and passwords are not sent across the network.

ProxyAutoDetectFallback

Specifies whether or not the ICA Client for Java should attempt a direct connection to the server if Internet Explorer is set to automatically detect proxy settings and the WPAD URL (http://wpad/wpad.dat) is not accessible. Values are On or Off.

Examples:

- § ProxyAutoDetectFallback=On – If the WPAD URL cannot be retrieved, attempt a direct connection to the destination server
- § ProxyAutoDetectFallback=Off – If the WPAD URL cannot be retrieved, abort the connection and display error message

Notes:

- § This parameter only affects the ICA Client for Java. Other clients never attempt to reach the WPAD URL.
- § In Java client version 7.1 and earlier the default value is Off; in Java client version 7.2 and later the default value is On.

ProxyUseFQDN

Determines whether the ICA client should attempt to resolve the destination server address or SSLProxyHost address to an IP address when connecting through a proxy server. Values are On or Off. Default value is Off.

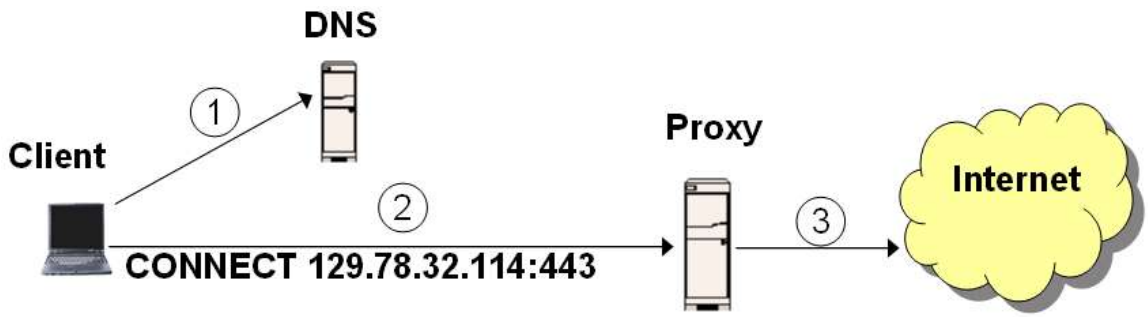


Figure 6 - Sample name resolution behavior when ProxyUseFQDN=Off

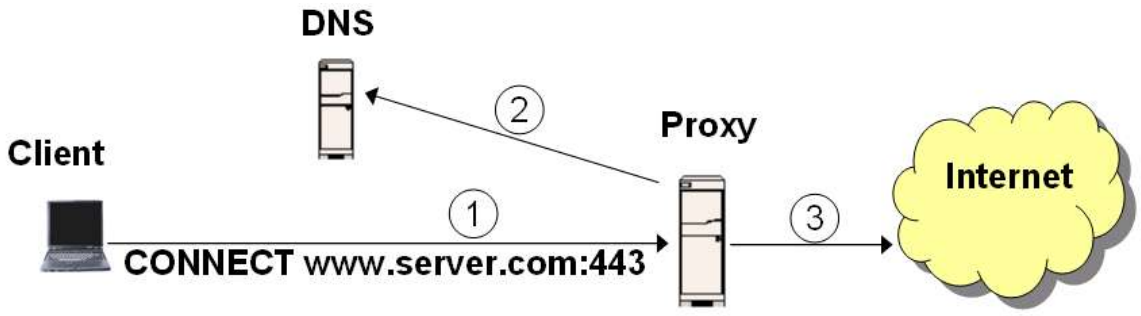


Figure 7 - Sample name resolution behavior when ProxyUseFQDN=On

Example:

- ProxyUseFQDN=On
- ProxyUseFQDN=Off

Notes

- If ProxyType=Auto and ProxyUseFQDN=On, users who are not behind a proxy server will be unable to connect to Secure Gateway. Therefore this parameter should not be added to template.ica on a Web Interface server that is configured to use Secure Gateway.
- If ProxyUseFQDN=Off, the client will attempt to resolve the destination server name and connect by IP address if possible. If the client is unable to resolve the destination server name to an IP address, the client connects by name and the proxy server will have to resolve the name.
- When ProxyUseFQDN=On, the client never attempts to resolve the destination server name to an IP address. The proxy server assumes responsibility for name resolution.
- If the destination server name is configured for DNS round-robin load balancing, setting ProxyUseFQDN=On may result in all clients behind a given proxy server being directed to a single node in the round-robin server group.

ProxyFavorIEConnectionSetting

Determines whether the ICA client should inspect connection-specific proxy settings for dial-up connections in Internet Explorer when ProxyType=Auto and the default web browser is Internet Explorer.

Example

- ProxyFavorIEConnectionSetting=On
- ProxyFavorIEConnectionSetting=Off

Notes

- When ProxyType=Auto and ProxyFavorIEConnectionSetting=Off, the proxy configuration from Internet Explorer LAN Settings will always be used.
- Set ProxyFavorIEConnectionSetting=On to support laptop users who must connect through a proxy server only while on the corporate network. When the users travel outside the office and connect to the Internet through an ISP, the LAN proxy settings will be ignored in favor of the proxy settings for the active dial-up connection.

ProxyFallback

Boolean value that determines whether the client should fail over to a direct connection when a PAC script cannot be retrieved. If true, failed attempts to download a proxy PAC script are ignored and the client attempts a direct connection.

For example, suppose a laptop on the corporate network has Internet Explorer configured to use the proxy auto configuration script located at the URL http://LANSERVER/proxy.pac. When the user takes the laptop home and connects to the Internet through an ISP, LANSERVER is not accessible so the PAC script download fails. By setting ProxyFallback=Yes, the failed attempt to contact LANSERVER is ignored and the client attempts a direct connection.

This parameter was introduced in version 8.1 of the Win32 client. The default value is No.

Example

- ProxyFallback=Yes
- ProxyFallback=No

ICA Client Proxy Parameter Support Matrix

Not all of the proxy parameters and values defined in the previous section are valid on all platform variants of the ICA client. The following table and notes clarify which values may be used on which operating systems.

The minimum required client version for all platforms is 6.30 unless otherwise noted.	Win32	Java	MacOS X	UNIX	WinCE
ProxyType=Secure	●	●	●	●	●
ProxyType=SOCKS, SOCKSv4 or SOCKSv5	●	●	●	●	●
ProxyType=Script, ProxyAutoConfigURL	●	●			
ProxyType=Auto	1	2		3	
Using WPAD		4			
ProxyFavorIEConnectionSetting=On	5				
ProxyUseFQDN=On	6				
ProxyAutoDetectFallback=On		●			
ProxyBypassList	●	●	●	●	●
ProxyUsername, ProxyPassword **Using Basic Authentication**	●	●	●	●	●
Using NTLM Authentication	7				
ProxyFallback	8				

Notes:

1. The Win32 client detects the proxy settings for the workstation's default web browser, which may not be the browser currently in use.
2. When using the Sun JVM, the proxy settings from the Java control panel are always used. If the Java control panel is configured to use Internet Explorer browser settings and Internet Explorer is configured to "Automatically detect settings", the Sun JVM will attempt a direct connection; WPAD will not be used.
3. UNIX clients 6.30 and later support Proxy Auto Detection only with Netscape 4.0 or later, and only if Netscape has been configured with a static proxy server address; auto client proxy detection is not supported if Netscape has been configured to use a PAC script.
4. When running within the Microsoft JVM, the ICA Java client implements minimal support for WPAD. If a PAC script is not found at the URL http://wpad/wpad.dat, the Java client will attempt a direct connection unless you set ProxyAutoDetectFallback=Off. On all non-Win32 clients and on Win32 clients using the Sun JVM, WPAD is not supported.
5. Version 7.2 or later
6. Version 8.0 or later
7. NTLM version 2 only
8. Introduced in version 8.1

Automatic Proxy Detection

The following steps describe how the Win32 ICA Client determines proxy settings automatically:

1. Automatic proxy detection is triggered when **ProxyType=Auto** is found in the current ICA file or Appsrv.ini.
2. The ICA Client reads the registry value **HKEY_CLASSES_ROOT\.htm** to determine the workstation's default browser. This value can be observed by typing **assoc .htm** at a command prompt. The value will be **htmlfile** if Internet Explorer is the default browser or **MozillaHTML** if Netscape is the default browser.
3. The Proxy settings for the default browser are read from the registry or the user's profile according to which browser is the default:
   - For Internet Explorer, proxy settings are retrieved using WinHTTP API calls from WinInet.dll. WinInet stores proxy information in the registry beneath **HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Internet Settings\ProxyServer**
   - For Netscape, proxy settings are read from the user's profile beneath Application Data\Mozilla\Profiles\default in a file named **prefs.js**.
4. If the web browser proxy settings point to a PAC script, the ICA Client performs an HTTP GET request to download the PAC script, then invokes the native Windows JScript interpreter to parse the script. This step requires JScript support to be enabled in Windows and relies on **JSCRIPT.DLL**.
5. If the default browser is Internet Explorer and "Automatically detect settings" is enabled, the setting is ignored and the Win32 ICA client attempts a direct connection.

Web Interface and Proxy Servers

When a user logs into Web Interface and clicks an application icon, Web Interface dynamically renders an ICA file containing the instructions for that user to connect to the chosen application. The ICA file produced by Web Interface may contain proxy instructions for the client. When Web Interface adds proxy instructions to an ICA file, settings in the user's APPSRV.INI file are overridden.

Using the **Web Interface Console**, configure client proxy settings on the **Client-side proxy** page. Start by defining the default proxy behavior for all users under the **Default proxy setting** section:

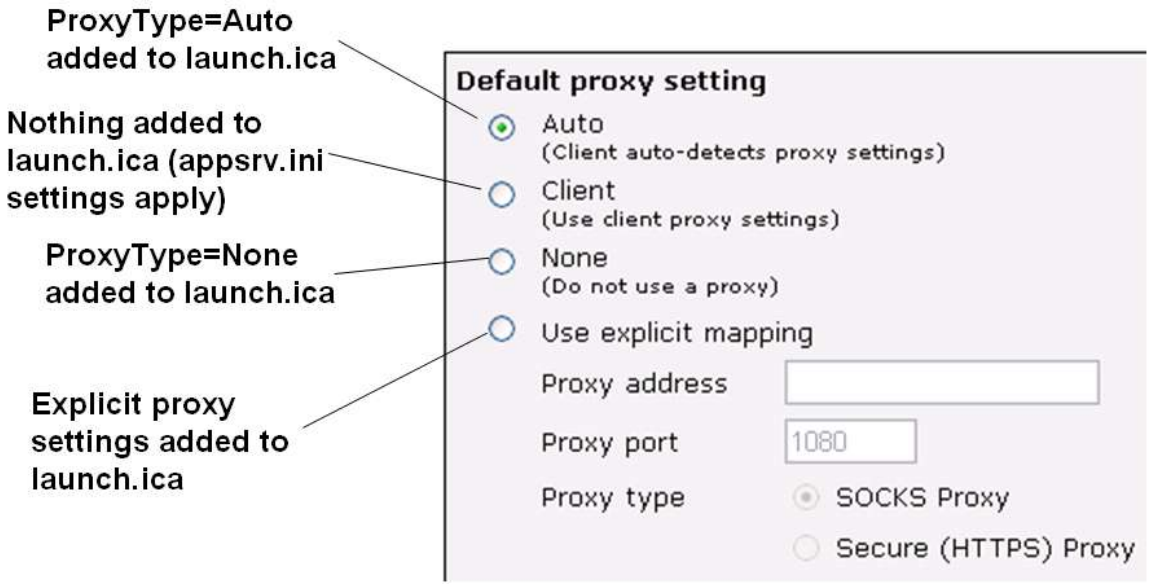


Figure 8 - Web Interface 3.0 Client-Side Firewall Settings: Default Proxy Setting

If necessary, override the default proxy settings for an individual site by adding an exception rule in the **Specific proxy settings** section. Rules added to this section override the default setting above if the client location matches the condition of the rule:

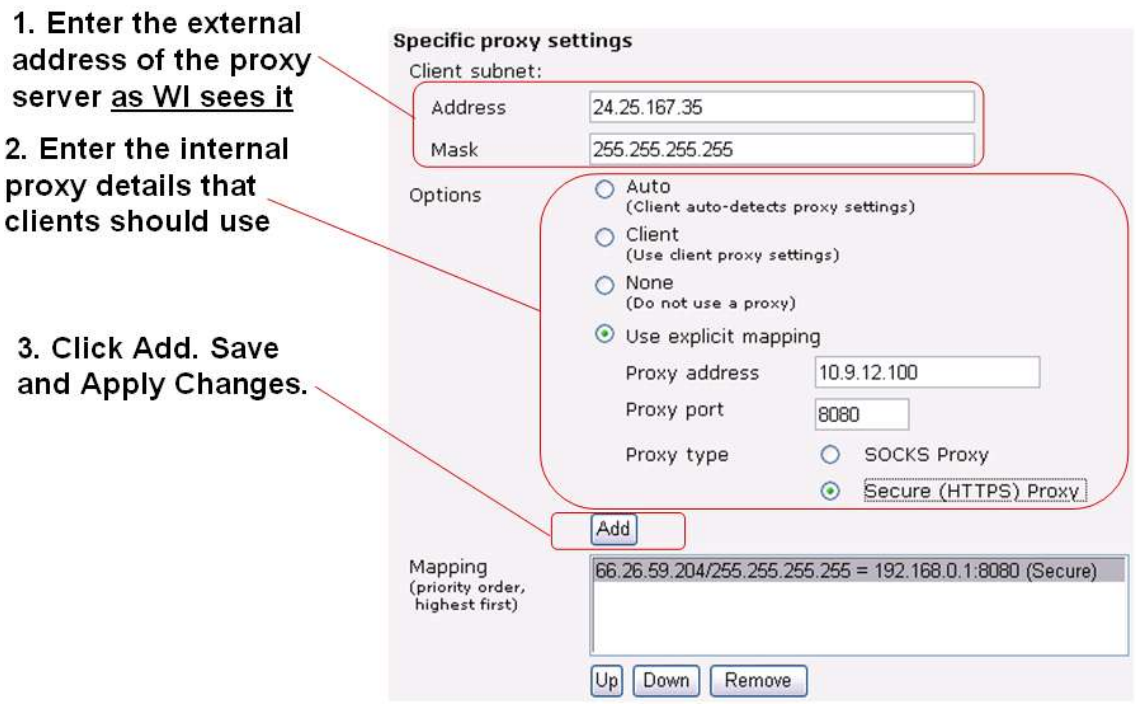


Figure 9 - Web Interface 3.0 Client-Side Firewall Settings: Specific Proxy Settings

Determining the Client Network Address

When configuring Web Interface address translation or client-side firewall settings, it is important to remember that all internet traffic coming through a reverse proxy or a proxy-based firewall will appear to Web Interface as though it originated at the nearest proxy server.

For any HTTP server session, an HTTP server variable called **REMOTE_ADDR** will be present. The value of REMOTE_ADDR is the TCP return address of the traffic that arrived at the Web server. When clients access the Web server directly, REMOTE_ADDR is the actual IP address of the client device. Web Interface and MetaFrame Secure Access Manager rely on the value of this variable to make decisions about address translation and client-side proxy servers based on client location.

When traffic is routed through one or more proxy servers, the value of REMOTE_ADDR will always be the address of the proxy server nearest to the web server, not the true client address. From a networking point of view, a single HTTP(S) session might traverse multiple TCP segments. REMOTE_ADDR will be the return address for the TCP session which finally connects to the target web server:

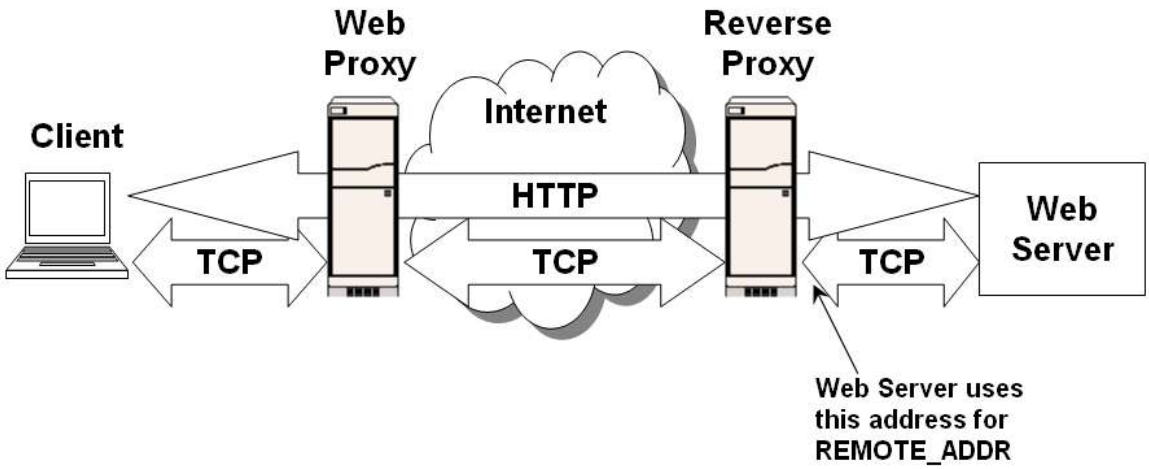


Figure 10 - Client Network Address May Be a Proxy Address

Therefore, any "client network address" settings made in the Web Interface Console (WIAdmin) for Network Address Translation or Client-side Proxy Server settings that are intended to apply to external users must be defined using the IP address of the nearest proxy server. It is this return address which should be entered as the “Client subnet” address in a **Specific proxy settings** rule.

Observing REMOTE_ADDR

You can view the value of REMOTE_ADDR using debug.asp from CTX052061:

CTX052061 - Citrix Web Server Debugging & Analysis Tool
http://support.citrix.com/kb/entry.jspa?externalID=CTX052061

Or use the following line of code in an ASP script:

REMOTE_ADDR is <%= Request.ServerVariables("REMOTE_ADDR") %>

Secure Gateway

Secure Gateway is secure reverse proxy server for SOCKS, HTTP or CGP traffic. CGP stands for Citrix Gateway Protocol, a TCP tunneling protocol developed by Citrix and currently used only by the Gateway Client for Secure Access Manager. A Secure Gateway server will proxy unauthenticated HTTP requests to one web server (referred to as the Logon Agent or Web Interface server), and will proxy authenticated HTTP requests to a different server (usually MetaFrame Secure Access Manager). Any ICA requests arriving at the Secure Gateway server must contain a secure ticket granted by a Secure Ticket Authority (STA). Tickets are requested from the STA for authenticated users by Web Interface or MetaFrame Secure Access Manager.

A convenient feature of Secure Gateway is that it allows Web Interface to be hosted on the same server. HTTPS traffic arriving at the gateway is decrypted and relayed to a web server running on the same machine. This allows Web Interface and Secure Gateway to share a single IP address and SSL certificate.

Problem: Placing Secure Gateway behind Reverse Proxy Causes SSL Error 4

Combining Web Interface and Secure Gateway can create confusion if another reverse web proxy is placed between the client and Secure Gateway. This scenario does not generally cause problems with HTTPS traffic destined for Web Interface, but a reverse web proxy cannot be used for ICA/SSL traffic. When a combination Secure Gateway/Web Interface server is placed behind a reverse web proxy, users are able to log into Web Interface and enumerate application icons (all HTTP communications), but attempting to launch a published application results in SSL Error 4. This happens because the ICA/SSL session is terminated by the reverse web proxy, not the Secure Gateway server:

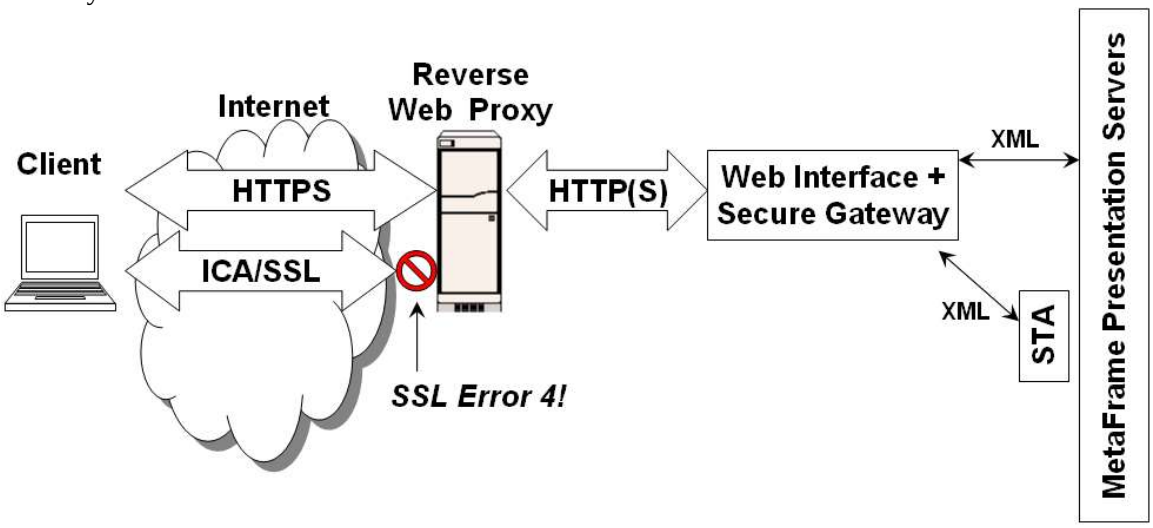


Figure 11 - INCORRECT placement of Secure Gateway behind Reverse Web Proxy

Here the reverse web proxy is viewed by Secure Gateway as a “man in the middle” compromising the integrity of the ICA/SSL network stream. This causes the SSL handshake between the ICA Client and Secure Gateway to fail.

There following sections describe two possible solutions to this problem.

Solution One: Run Secure Gateway Parallel to the Reverse Web Proxy

Separate Web Interface and Secure Gateway onto two machines. Place the server running Web Interface behind the reverse web proxy and place the Secure Gateway server parallel to the reverse web proxy:

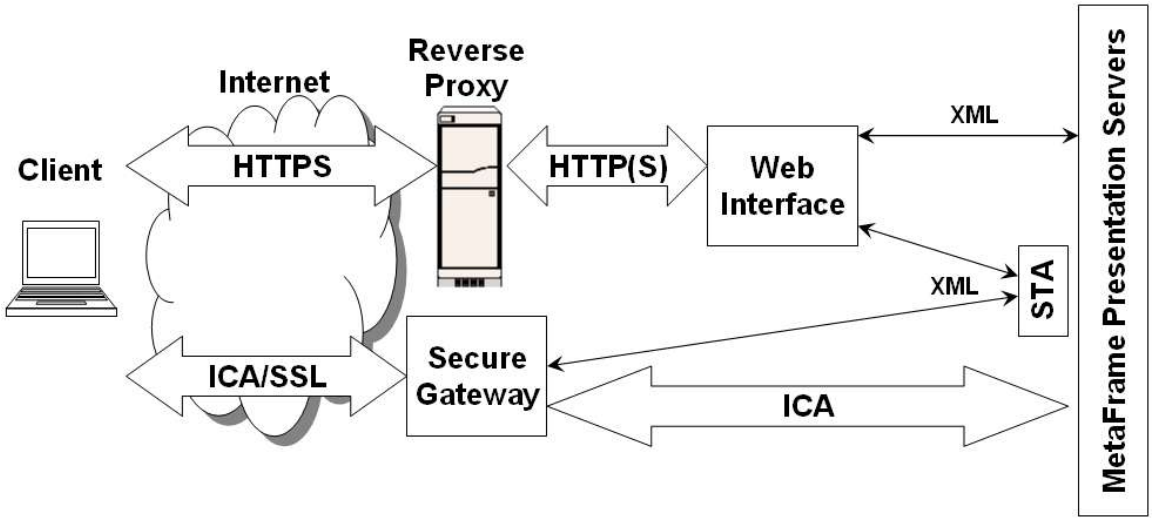
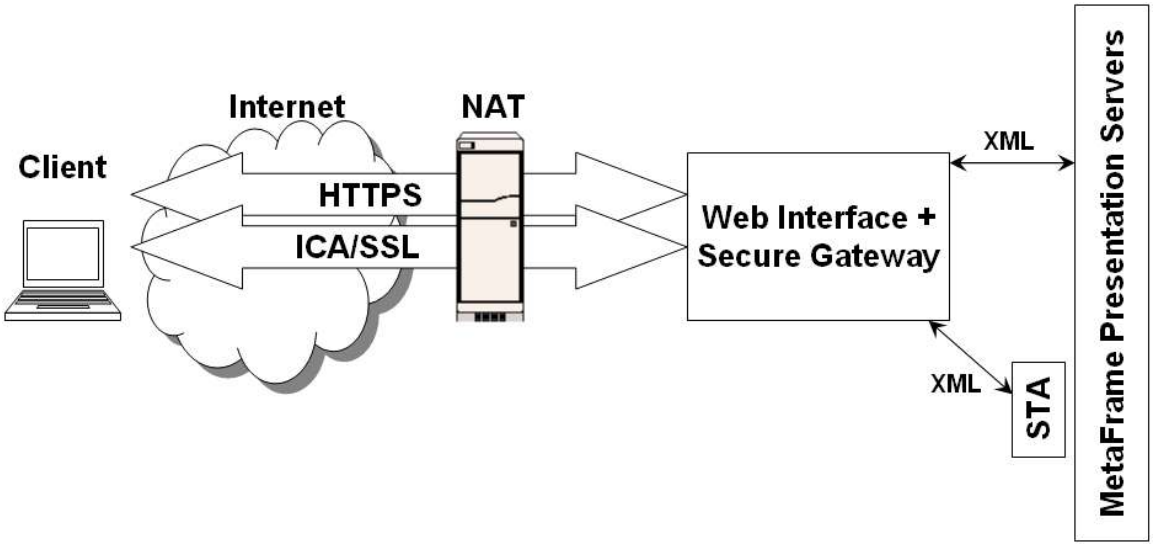


Figure 12 - CORRECT Placement of Secure Gateway Parallel to Reverse Web Proxy

This scenario is still secure, and any security policies defined at the reverse web proxy will still affect all Secure Gateway users. In order to traverse the secure gateway, users must first satisfy the reverse web proxy and log into Web Interface in order to obtain a ticket from the STA. Therefore any access control rules defined at the reverse web proxy will affect users wishing to gain entry through Secure Gateway as well.

Solution Two: Use NAT instead of a Reverse Web Proxy

If the reverse proxy is configured to forward all traffic (not just HTTP traffic) to the combination Secure Gateway/Web Interface server, then SSL is not terminated at the proxy and users are able to connect through Secure Gateway. Different vendors refer to this deployment style in different ways, for example



This approach has the disadvantage that some control must be sacrificed regarding the type of traffic that is permitted to traverse the proxy. Incoming traffic must be routed directly to the Secure Gateway/Web Interface server without being decrypted, authenticated or inspected. From a security standpoint, this is not much different from exposing the Secure Gateway server directly to the Internet. There is a logical SSL “tunnel” between the client and Secure Gateway.

Notes for Specific Proxy Servers

Microsoft ISA Server

Microsoft ISA Server is capable of many different roles. A single ISA Server can act as a forward web proxy, secure proxy, reverse proxy, SOCKS proxy and NAT firewall all at the same time.

When using Microsoft ISA Server as a Forward Web Proxy

When a MetaFrame Presentation Server Client is behind a web proxy such as Microsoft ISA Server (but ISA is not being used as the default gateway), the client will attempt to reach MetaFrame Presentation Servers using the CONNECT method, also known as "SSL Tunneling." By default, Microsoft ISA Server allows the CONNECT method only to ports 443 (HTTPS) and 563 (NNTP). Connections to Secure Gateway should work by default, but connections to a MetaFrame Presentation Server will fail by default.

In order to allow ICA connections through Microsoft ISA Server on ports 1494 or 2598, a script must be run at the ISA Server which modifies the ports for which SSL Tunneling is allowed.

When the following script is executed on a Microsoft ISA Server, ports 1494 (ICA) and 2598 (Session Reliability) are added to the list of ports for which SSL Tunneling is allowed:

Script for Microsoft ISA Server 2000

```
set isa=CreateObject("FPC.Root")
set tpr=isa.Arrays.GetContainingArray.ArrayPolicy.WebProxy.TunnelPortRanges
set tmp=tpr.AddRange("ICA 1494", 1494, 1494)
set tmp=tpr.AddRange("CGP 2598", 2598, 2598)
tpr.Save
```

Script for Microsoft ISA Server 2004

```
set isa=CreateObject("FPC.Root")
set tpr1=isa.Arrays(1)
set tpr=tpr1.ArrayPolicy.WebProxy.TunnelPortRanges
set tmp=tpr.AddRange("ICA 1494", 1494, 1494)
set tmp=tpr.AddRange("CGP 2598", 2598, 2598)
tpr.Save
```

After running this script, restart the Microsoft Web Proxy service (ISA 2000) or Microsoft Firewall Service (ISA 2004) for changes to take effect.

See the following articles from Microsoft for more information about configuring SSL Tunneling for ISA Server:

1. SSL tunneling
   http://www.microsoft.com/resources/documentation/isa/2000/enterprise/proddocs/en-us/isadocs/cmt_authpass.mspx
2. FPCTunnelPortRange Object
   http://msdn.microsoft.com/library/en-us/isa/isaobj3_7gl0.asp

When using ISA as a Reverse Web Proxy

An important distinction exists in ISA terminology between *Web Publishing* and *Server Publishing*. If you use a *Web Publishing* rule to expose a web server to the Internet, all inbound client TCP connections are terminated by the ISA server and then the ISA server connects to the internal server on behalf of the client. This type of rule can be used with Web Interface or MetaFrame Secure Access Manager, but not for ICA or ICA/SSL traffic. If a *Web Publishing* rule is used to grant external access to a server where both Web Interface and Secure Gateway are installed, users will find that browsing for web pages and enumerating application icons will succeed but the final ICA/SSL connection will fail with "SSL Error 4".

For ICA traffic or SSL traffic to traverse an ISA server successfully, a *Server Publishing* rule must be defined instead. When *Server Publishing* is used to expose a service to the Internet, the ISA server does not terminate and re-establish the connection on behalf of the client. This allows for end-to-end connections between the client device and the target server.

Squid

When a MetaFrame Presentation Server Client is behind a web proxy such as Squid, the client will attempt to reach MetaFrame Presentation Servers using the CONNECT method, also known as “SSL Tunneling.” By default, Squid allows the CONNECT method only to port 443 (HTTPS). Connections to Secure Gateway should work by default, but connections to a MetaFrame Presentation Server will fail by default.

In order to allow ICA connections through Squid on ports 1494 or 2598, edit the etc/squid.conf file and locate the following line:

acl SSL_Ports port 443 #https

Add the numbers 1494 and 2598, separated by spaces after the number 443:

acl SSL_Ports port 443 1494 2598 #https

Save the squid.conf file and restart Squid in order for the change to take effect.

NetCache

NetCache supports NTLMv1 authentication, but the Win32 Client requires NTLMv2. Therefore when using a NetCache proxy, only Basic authentication is supported. (CTX103363)

Novell BorderManager

For SSL or ICA connectivity, enable the “Act as a tunnel” checkbox.

EnTrust GetAccess

GetAccess can be used as a reverse proxy for HTTP traffic only. This means it can work for Web Interface but Secure Gateway or ICA connections must bypass the proxy as illustrated in Figure 12 - CORRECT Placement of Secure Gateway Parallel to Reverse Web ProxyFigure 12.

Summary of Recommendations

Citrix offers the following general recommendations to maximize compatibility with the diverse and growing set of proxy servers, Layer-4 switches and firewalls on the market. Please consider these recommendations in the context of your own security needs.

1. Use Secure Gateway for MetaFrame on port 443

To maximize the chances that users will be able to connect to published applications in your MetaFrame Presentation Server farm, expose your farm to the Internet using Secure Gateway for MetaFrame and adhere to the following best practices:

- Purchase a commercial SSL certificate for the Secure Gateway server(s) from a widely trusted certification authority
- Configure the Secure Gateway service to listen on port 443

2. Make changes to ICA template files

ICA files rendered by Web Interface or Secure Access Manager to users on the Internet should contain the following lines in the [WFClient] section:

ProxyType=Auto
ProxyFavorIEConnectionSetting=On

Depending on the version of Web Interface or Secure Access Manager, some or all of these settings may not be included in rendered ICA files. Use the following table to determine what changes need to be made in order for these values to appear in rendered ICA files:

Product version	Notes
Web Interface 3.0	ProxyType=Auto is included by default. Edit all four template.ica files in the /Citrix/MetaFrame/conf folder to add the ProxyFavorIEConnectionSetting values.
Web Interface 2.x	A bug in Web Interface 2.x prevents ProxyType=Auto from appearing until the Client-Side Firewall page of the Web Interface Administration Tool (WIAdmin) has been visited and the administrator clicks “Save” and “Apply changes.” After doing that, edit the template.ica files in the Program Files\Citrix\NFuse folder to add the ProxyFavorIEConnectionSetting.
NFuse Classic 1.71 and earlier	Edit the template.ica files in the Program Files\Citrix\NFuse folder and add the entries.
MetaFrame Secure Access Manager 2.2 and earlier	Edit the icafile.xslt in the bin/Binders folder and add the entries.

3. Bypass reverse web proxy servers

As discussed in the Secure Gateway section above, most SSL VPN appliances, hardware load balancers and concentrators that offer reverse proxy capabilities do so for HTTPS traffic only. Since the ICA client does not use HTTP when communicating with a MetaFrame Presentation Server, these devices in their default configurations tend to prevent connectivity to published applications.

When exposing a MetaFrame Presentation Server farm to the Internet with Secure Gateway, ensure that ICA/SSL traffic can be routed to the Secure Gateway listening port without undergoing any modification above the TCP layer.

Appendix

Proxy-related error messages

The following error messages may be encountered when the ICA client is unable to reach a destination MetaFrame Presentation Server or Secure Gateway server through a proxy server. The notes associated with each error message are intended to be general guidelines only; some errors may occur for a wide variety of reasons.

Error message	Notes
The configured proxy server was unable to establish a connection	The ICA Client connected to a forward proxy server and sent a CONNECT request for the target MetaFrame Presentation Server or Secure Gateway server. The proxy server rules are configured to permit the connection, however the proxy server was unable to complete the connection. Check that the proxy server is able to resolve the FQDN of the destination server and that it can route traffic to the destination IP address.
The configured proxy server refused to establish a connection	The ICA Client connected to a forward proxy server and sent a CONNECT request for the target MetaFrame Presentation Server or Secure Gateway server, but the proxy server rules are configured not to allow the connection. Check to ensure that the proxy server allows the CONNECT method to the destination IP and port number.
There is no MetaFrame server at the specified subnet address **-OR-** ***Protocol driver error***	Clients receive one of these errors if they must pass through a forward proxy server in order to reach the destination MetaFrame Presentation server, but a) the client or ICA file is not configured to use a proxy server or b) the configured proxy server is offline. If using Web Interface or Secure Access Manager, check to ensure that ProxyType=Auto is included in the rendered ICA file.

Proxy Detection Failed: PAC script error. HTTP Download failed. Connection failure.	The ICA Client is configured to use a Proxy Auto Configuration script, but the URL provided for the script could not be reached. This could happen if the web server hosting the PAC script were down or if it requires authentication to download the script. Check the PAC script URL (normally configured via the web browser proxy settings) and ensure that the client can download the script via anonymous HTTP GET. Add ProxyFallback=On to enable a direct connection when the PAC script cannot be located.
Proxy Detection Failed: No JavaScript support	The ICA Client downloaded a PAC script, but was unable to parse the script using the built-in Windows JScript interpreter. Try typing the following command to re-register the JScript library: **regsvr32 %systemroot%\system32\JSCRIPT.DLL**