Secure Gateway 3.0
for Presentation Server

Troubleshooter's Guide

Author	Jay Tomlin
Department	Technical Support
Revision	2.0
Distribution	Public

CITRIX

Table of Contents

About this document 3
1. What's new in Secure Gateway 3.0 4
  1.1 New architecture based on Apache 4
  1.2 Secure Ticket Authority bundled with XML Service 4
  1.3 Common Gateway Protocol 4
  1.4 Support for wildcard certificates 6
  1.5 Support for Relay mode 6
  1.6 What's not included 6
2. Secure Gateway Solution Components 7
  2.1 Secure Gateway Service 7
  2.2 Secure Gateway Proxy 7
  2.3 Secure Ticket Authority 8
3. Deployment Scenarios 10
  3.1 Single-DMZ with Web Interface 10
  3.2 Dual-DMZ with Web Interface 11
  3.3 Secure Proxy on the Trusted Network 11
  3.4 Relay mode 12
4. Secure Gateway Features in Detail 13
  4.1 Configuring Web Interface 4.0 to use Secure Gateway 13
  4.2 Secure Gateway Ticketing 18
    4.2.1 Ticket Types 18
    4.2.2 How it works 18
    4.2.3 How it breaks 20
    4.2.4 Known limitations and issues 21
    4.2.5 Frequently Asked Questions 21
  4.3 Session Reliability through the gateway 22
    4.3.1 Session Reliability without Secure Gateway 22
    4.3.2 Frequently Asked Questions about Session Reliability 23
    4.3.3 Session Reliability through Secure Gateway 3.0 24
    4.3.4 Session Reliability System Requirements 25
  4.4 Relay Mode 25
    4.4.1 How it works 26
    4.4.2 Known limitations and issues 26
    4.4.3 Frequently Asked Questions 26
5. Digital Certificates 27
  5.1 Certificate chain validity requirements 28
    5.1.1 How it works 28
    5.1.2 How it breaks 30
    5.1.3 SSLv3 vs SSLv1 30
  5.2 Certificate renewal and replacement 30
  5.3 SGC Certificates 32
6. Troubleshooting 33
  6.1 Common error messages 33
    6.1.1 Client reports SSL Error 4 33

About this document

This document is intended as a reference for those who need to solve complex Technical Support issues involving Secure Gateway 3.0. It complements—but does not replace—the *Secure Gateway Administrator's Guide*. Focus is on the technical details of how features are implemented and how they tend to break.

Important

This document deals only with Secure Gateway 3.0 deployments that integrate with Web Interface and Presentation Server. It does not include troubleshooting information about using Secure Gateway with MetaFrame Secure Access Manager, Access Gateway, or Access Gateway Enterprise.

1. What's new in Secure Gateway 3.0

1.1 New architecture based on Apache

The code base for the Secure Gateway service has been entirely rewritten for version 3.0. The new code is based on Apache, the open-source HTTP and proxy server. The Secure Gateway 3.0 team started with the Apache source code and modified it to produce functionality that is a superset of the Secure Gateway 2.0 features.

This redesign results in several important changes with respect to troubleshooting:

- The Secure Gateway service reads all configuration settings from the **hidden file httpd.conf** located in the Program Files\Citrix\Secure Gateway\conf directory
- The Secure Gateway Service Configuration tool **writes changes in duplicate** to httpd.conf and also to the registry beneath the gateway service key at HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\CtxSecGwy
- The Secure Gateway Diagnostics tool only reads settings from the registry; the service only reads settings from httpd.conf
- If the settings stored in the registry are ever found to be out of synch with the settings in httpd.conf, the administrator receives a warning upon launch of the Secure Gateway Service Configuration tool
- Manual changes to httpd.conf are not recommended; **any manual changes will be lost** whenever the Secure Gateway Service Configuration tool is run

> ***Just FYI***
>
> The Citrix XTE Service introduced in MetaFrame Presentation Server 3.0 uses the same Apache architecture to deliver the Session Reliability and SSL Relay features.

1.2 Secure Ticket Authority bundled with XML Service

The Secure Ticket Authority (STA), formerly available only as an ISAPI application for IIS, is **bundled with the Citrix XML Service** in Citrix Presentation Server 4.0. An updated standalone installer for IIS will not be made available.

When the STA is delivered by the Citrix XML Service:

- The properties of the STA are governed by **CtxSta.config** located in **Program Files\Citrix\System32**
- The new **AllowedClientIPList** parameter in CtxSta.config can be used to restrict access to the STA for a given list of IP addresses or IP ranges
- The new **SSLOnly** parameter allows the STA to reject any requests that were not directed through the SSL Listener of the Citrix XTE Service (SSL Relay)

1.3 Common Gateway Protocol

Secure Gateway 3.0 introduces full support for the Common Gateway Protocol developed by Citrix. Common Gateway Protocol lays a strong foundation for the remote access capabilities of Citrix Presentation Server and Advanced Access Control.

Common Gateway Protocol is a sophisticated binary protocol designed for efficient tunneling of one or more TCP streams. It bears similarities to SOCKS or SSH in that it is fundamentally a tunneling protocol capable of transferring application data to a remote network.

> ***FYI***
> Common Gateway Protocol is not entirely new. The original MetaFrame Secure Access Manager 2.0 Gateway Client used an early version of Common Gateway Protocol. And when Session Reliability is enabled in MetaFrame Presentation Server 3.0, the 8.0 Win32 ICA client tunnels ICA data through a Common Gateway Protocol connection to the Citrix XTE Service on port 2598.

Like ICA, Common Gateway Protocol is a layered binary protocol with its own reliability, compression, encryption, and reconnect capabilities.

The following table clarifies where Common Gateway Protocol is used by listing the protocols and default TCP ports used by current and previous Citrix components. Protocols and ports will sometimes vary according to whether or not session reliability is available and enabled.

COMPONENT	CONNECTING TO	SESSION RELIABILITY	NETWORK PROTOCOL	TCP PORT
ICA Client Versions 6.3-8.0	MetaFrame XP Feature Release 3 or earlier	Not Available	ICA	1494
Win32 ICA Client version 8.0 or later	Presentation Server 3.0 or later	Enabled	ICA in Common Gateway Protocol	2598
ICA Client versions 6.3-8.0	Secure Gateway 2.0 or earlier	Not Available	ICA in SOCKS in SSL	443
ICA Client 9.0	Secure Gateway 3.0	Enabled	ICA in Common Gateway Protocol in SSL	443
Secure Gateway 3.0	Presentation Server 3.0 or later	Enabled	ICA in Common Gateway Protocol	2598
ICA Client 9.0	Secure Gateway 3.0	Disabled	ICA in Common Gateway Protocol in SSL	443
Secure Gateway 3.0	Presentation Server	Disabled	ICA	1494
Web Browser	Secure Gateway 2.0 or later	N/A	HTTPS	443
Secure Gateway 2.0 or later (DMZ1)	Secure Gateway Proxy 2.0 or later (DMZ2) without SSL	N/A	SOCKS	1080
Secure Gateway 2.0 or later (DMZ1)	Secure Gateway Proxy 2.0 or later (DMZ2) with SSL	N/A	SOCKS in SSL	443

1.4 Support for wildcard certificates

Secure Gateway 3.0 can be configured to use a wildcard certificate, for example, a certificate issued to ***.company.com***. Wildcard certificates are supported by ICA clients version 7.0 and later.

1.5 Support for Relay mode

Relay mode, available in version 1.0 but not in version 2.0, will once again be supported in version 3.0. Relay mode is not as secure as normal mode because no tickets are required from the STA for admission into the trusted network. Program Neighborhood clients are able to traverse a relay mode gateway server without mandating a Web Interface server.

See the section on Relay Mode for more details.

1.6 What's not included

Support for MetaFrame Secure Access Manager 2.x

Secure Gateway 3.0 is not backward-compatible with MetaFrame Secure Access Manager. If your deployment includes MetaFrame Secure Access Manager, continue using Secure Gateway 2.0. Secure Gateway 3.0 is compatible with the Advanced Access Control Option 4.0, which replaces MetaFrame Secure Access Manager.

Support for multiple certificates

It is not possible to create multiple listeners on the gateway and associate a different certificate with each listener.

Logon Agent

The MSI file for Secure Gateway 3.0 no longer includes the Logon Agent component. The Logon Agent is now delivered as a separate MSI file, included with Access Gateway Enterprise 4.0.

Standalone Secure Ticket Authority

Secure Gateway 3.0 does not include a standalone installer for the Secure Ticket Authority. Instead, you should use the XML Service from Citrix Presentation Server 4.0 or the Authentication Service from Access Gateway Enterprise 4.0, both of which include STA functionality.

Secure Gateway 3.0 is backward-compatible with older ticket authorities, but some features are lost depending on the STA version. See the STA section for more details.

2. Secure Gateway Solution Components

This section provides an introduction to all components in the Secure Gateway solution for Web Interface and Presentation Server.

2.1 Secure Gateway Service

In version 3.0, the Secure Gateway service is a Win32 executable based on Apache. The service listens as a reverse proxy for connections on one or more TCP ports (default 443).

All connections are expected to be encrypted with SSL, but once the SSL layer is decoded the gateway can deal with multiple types of traffic. The types of traffic it handles are:

1. **Unauthenticated HTTPS**, which the gateway may forward to the Web Interface server
2. **Authenticated Common Gateway Protocol**, which tunnels any other TCP traffic, usually ICA
3. **Authenticated SOCKS**, which tunnels any other TCP traffic, usually ICA

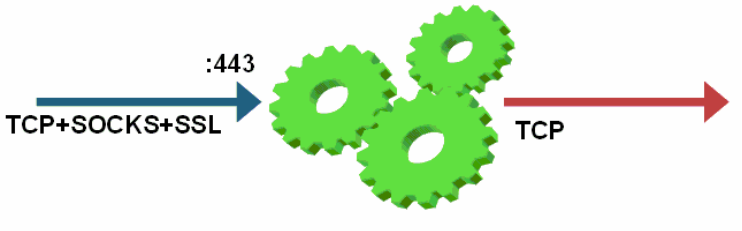
For traffic to be considered authenticated, a ticket from the STA must accompany the initial connection request. For Web Interface deployments, HTTP traffic is always considered unauthenticated. For ICA or other TCP connections, authentication is asserted by the presence of a ticket in the Common Gateway Protocol or SOCKS header. Web Interface is responsible for authenticating the user and then requesting a ticket from the STA.

2.2 Secure Gateway Proxy

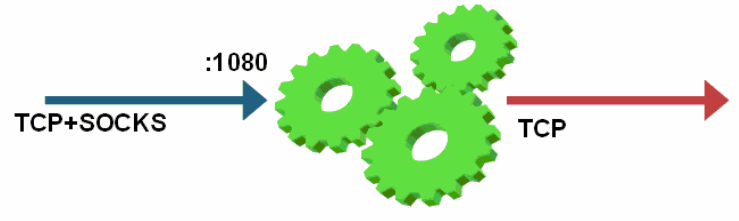
The Secure Proxy service is the same binary executable as the normal Secure Gateway service, but its configuration causes it to accept only one type of traffic:

- **Unauthenticated SOCKS**, which tunnels any other TCP traffic

The Secure Gateway proxy tunnels traffic without requiring any sort of authentication ticket. It therefore should not be exposed directly to any untrusted network such as the Internet. When configured with a certificate, the Secure Gateway proxy expects traffic to be SOCKS+SSL on port 443:



Without a certificate, the Secure Proxy is a simple SOCKS proxy on port 1080:



The Secure Gateway proxy may be used for one of three purposes:

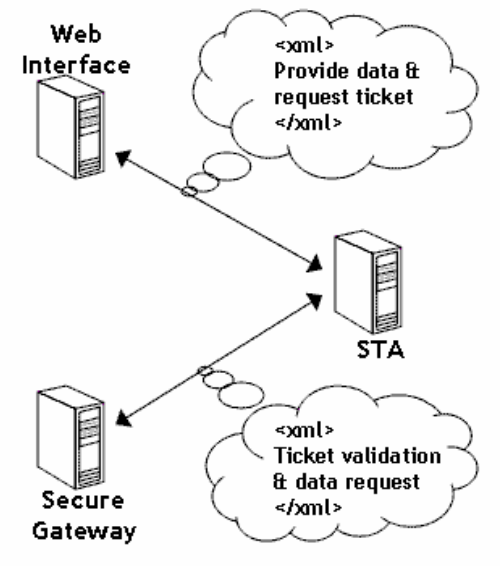
1. **Dual-DMZ deployments**, in which traffic from the Secure Gateway service in DMZ1 must employ a proxy in DMZ 2 in order to reach the trusted network. In this deployment, the Secure Gateway acts as a SOCKS client, using the Secure Gateway proxy as a SOCKS proxy server for all inbound, authenticated traffic. Note that only authenticated traffic is directed to the Secure Proxy server; the Secure Gateway service must have a direct route for unauthenticated HTTP traffic to reach the Web Interface server in DMZ 2.
2. **Firewall simplification**, in which the Secure Proxy resides on the trusted network and acts as the TCP surrogate for all external traffic. This allows the administrator to configure the internal firewall to allow traffic only from the Secure Gateway server in the DMZ to the Secure Proxy on the trusted network. The administrator is thus relieved of configuring firewall rules for each destination server.
3. **Relay mode,** in which unauthenticated ICA clients connect through the Secure Proxy en route to Presentation Server. Web Interface is not required.

2.3 Secure Ticket Authority

Secure Gateway was designed to defer authentication to Web Interface; Secure Gateway permits users to connect only after they have been authenticated by the Web server. A benefit of this design is that the Web server may be protected by third-party authentication requirements and those requirements will logically cascade to Secure Gateway connections. For example, if RSA SecurID tokens are required to gain access to a server running Web Interface, then we can conclude that only users with valid SecurID tokens will be able to traverse the secure gateway.

To implement authentication deferral, a third component was needed to act as a common broker between the Web server and Secure Gateway. This is the role of the Secure Ticket Authority (STA). The STA is an XML web service that produces and validates random gateway tickets (not the same as MetaFrame logon tickets) for each Secure Gateway connection.

During normal operation, a valid ticket is required for each new TCP connection that a client wishes to make through a secure gateway. When initiating a new connection, the client presents a ticket to the gateway and the gateway validates the ticket with the STA that created the ticket.



The STA is implemented as an ISAPI plugin called **CtxSta.dll**. Prior to the release of Citrix Presentation Server 4.0, the STA could only be hosted by IIS. With Presentation Server 4.0, the STA is bundled with the Citrix XML Service on each Presentation Server. Some Secure Gateway 3.0 features are unavailable when using legacy STA versions:

STA Version	Missing Secure Gateway 3.0 Features
1.0	▪ User/application details in Secure Gateway Console ▪ Advanced Gateway Client ▪ Session Reliability through the gateway
2.0	▪ Advanced Gateway Client ▪ Session Reliability through the gateway
2.2	▪ Session Reliability through the gateway
4.0	▪ None

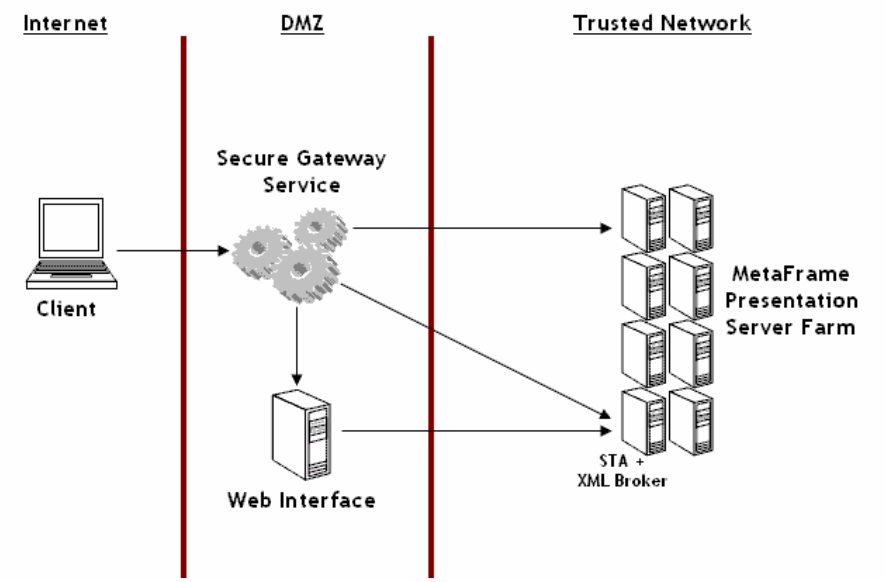
See the Secure Gateway Ticketing section below for more details on how ticketing works.

3. Deployment Scenarios

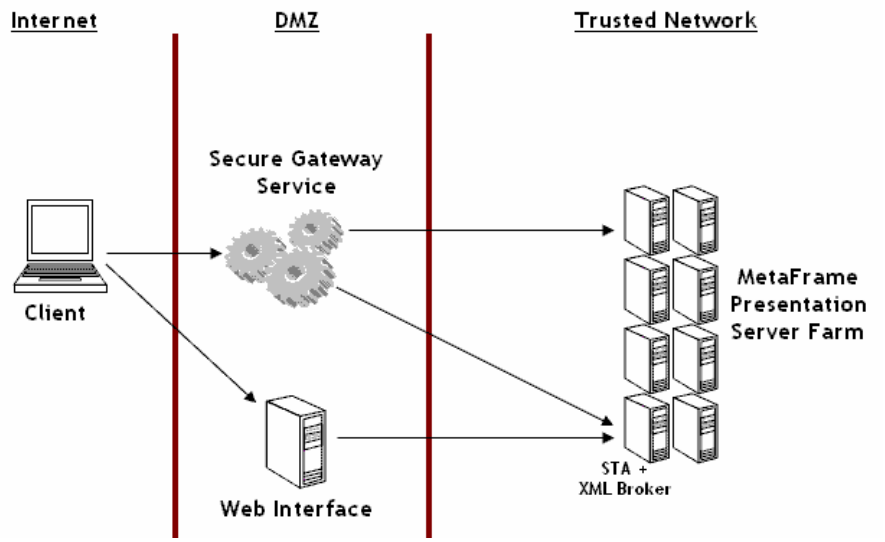
There are a few ways to deploy Secure Gateway for Presentation Server. At the network perimeter, Secure Gateway can be deployed in a single-DMZ (two firewalls) or a dual-DMZ (three firewalls).

The following diagrams introduce some of the more common deployment scenarios, illustrating where each component would reside. These diagrams are examples only, not an exhaustive summary of all deployment scenarios.

3.1 Single-DMZ with Web Interface

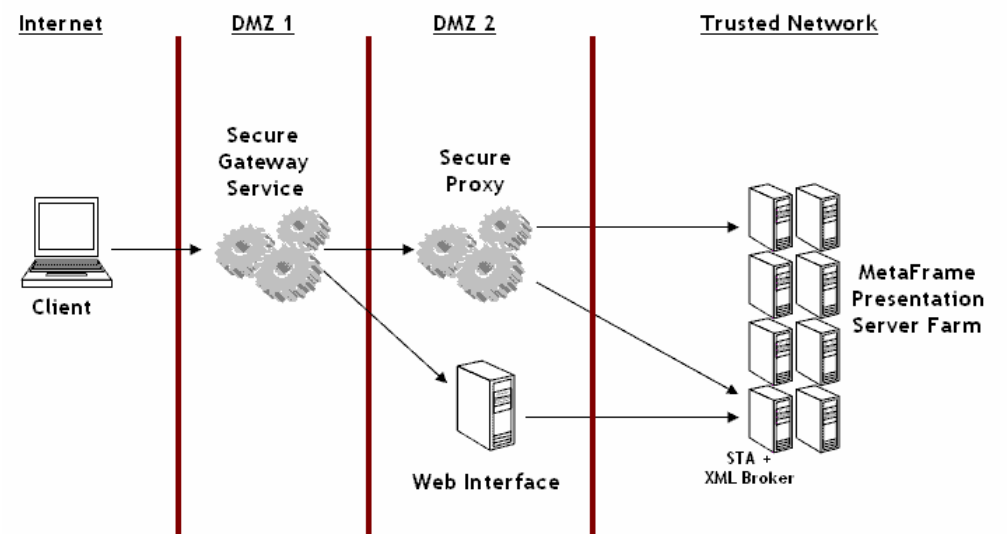


The diagram above depicts Web Interface "behind" the gateway, which means users enter the address of the Secure Gateway service into their Web browser, and the gateway relays the request to Web Interface. An alternative would be a "parallel" deployment where users contact Web Interface directly:



The “behind” deployment is more popular, especially for smaller businesses, because it requires only a single external IP address and a single SSL certificate (both of which cost money). But a “parallel” deployment is required if Web Interface is configured for smart card or desktop credentials pass-through authentication.

3.2 Dual-DMZ with Web Interface

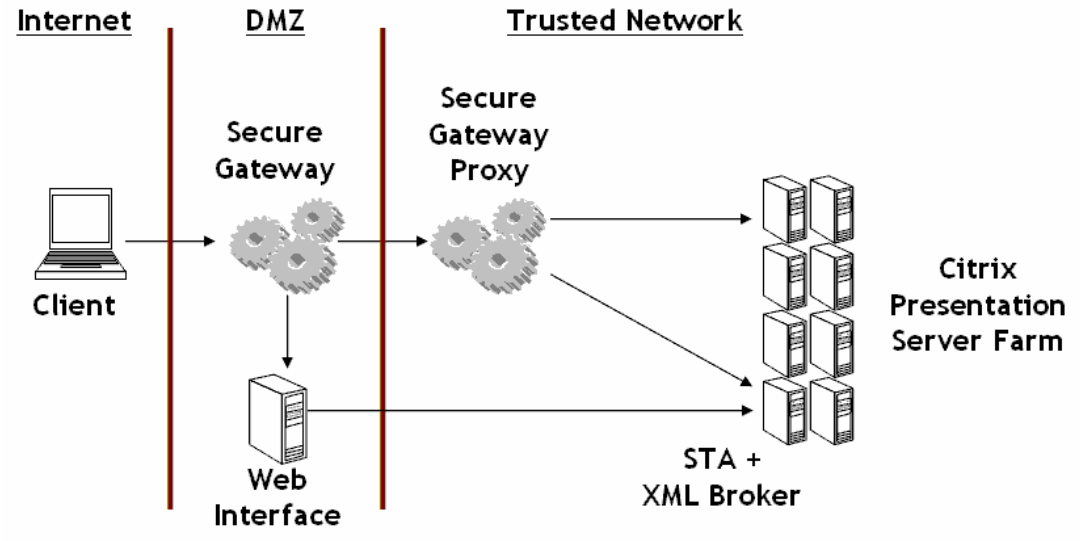


In a dual-DMZ deployment, Web Interface must be placed in the second DMZ where it can contact the STA and Presentation Server farm directly. Dual-DMZ deployments with Web Interface are therefore always “behind” deployments. The gateway service in DMZ 1 uses the Secure Proxy as a SOCKS proxy server for all inbound traffic sent to the trusted network.

Looking at the diagram, you might conclude that the Secure Proxy server in DMZ 2 initiates connections to the XML broker or target MetaFrame Presentation server. But, strictly speaking, this is not the case. The Secure Gateway service in DMZ 1 initiates all inbound connections to the MetaFrame Presentation Server farm but uses the Secure Proxy as a means to reach them. This distinction manifests itself in terms of name resolution and the placement of root certificates when using SSL to secure the internal links. For example, to secure communications to the STA, the Secure Gateway server in DMZ 1—not the Secure Proxy—must be able to resolve the STA FQDN to an IP address and must have the corresponding CA root certificate installed.

3.3 Secure Proxy on the Trusted Network

Like the dual-DMZ deployment, the Secure Gateway sends all authenticated traffic through the Secure Proxy en route to servers on the trusted network. But in this scenario, there is no third firewall and the Secure Proxy resides on the trusted network.

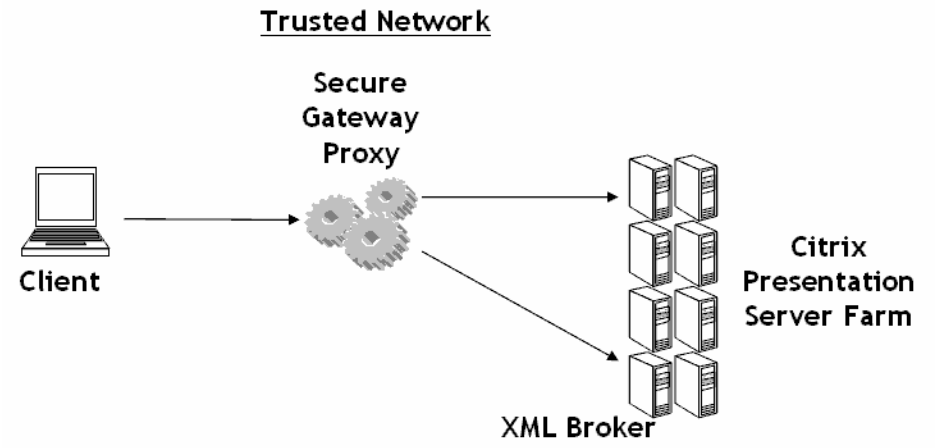


Using the Secure Proxy on the trusted network greatly simplifies the configuration of the inside firewall that separates the DMZ from the Trusted network. Presentation Servers can be added or removed from the farm without having to refine any of the access control lists defined on the firewall that separates Secure Gateway from the trusted network.

IMPORTANT
Web Interface should remain inside the DMZ in this scenario. Secure Gateway does not authenticate inbound HTTPS traffic before relaying it to the Web Interface server. Allowing external traffic to be relayed onto the trusted network effectively allows an attacker to bypass the DMZ.

3.4 Relay mode

The Secure Proxy can act as a "relay mode" server, through which ICA clients can connect to Presentation Server applications without requiring Web Interface or the Secure Ticket Authority.



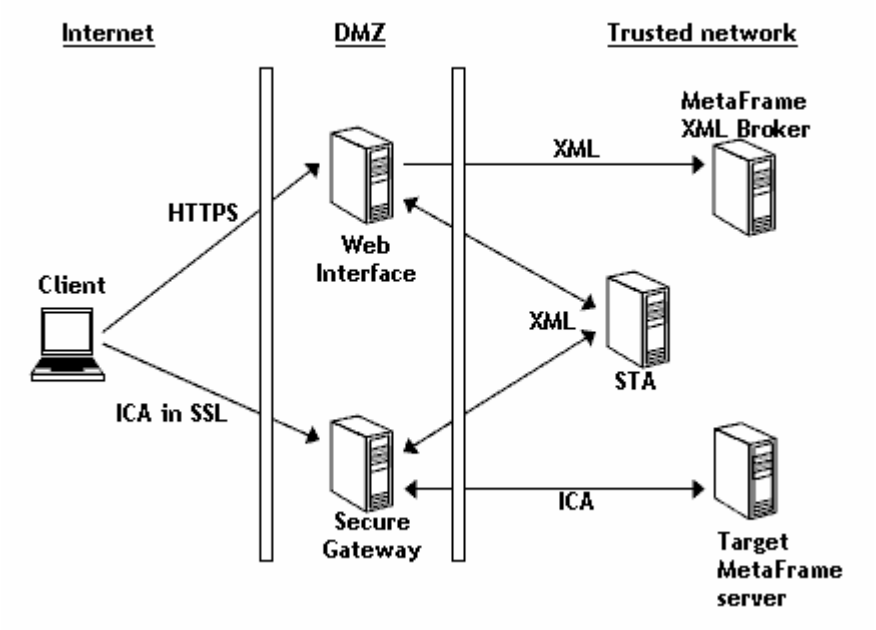
Since no authentication or ticketing takes place in this scenario, this is recommended for internal deployments only. See the Relay Mode section below for more details.

4. Secure Gateway Features in Detail

4.1 Configuring Web Interface 4.0 to use Secure Gateway

Web Interface can render ICA files for users that direct them through a Secure Gateway server en route to their applications.

After authenticating to the Web server and selecting a published application, users connect to a gateway server instead of the target Presentation server. The gateway acts as an ICA traffic relay between the client on an untrusted network and a MetaFrame Presentation Server on the trusted network. ICA traffic between the client and the gateway is encrypted using 128-bit SSL or TLS.

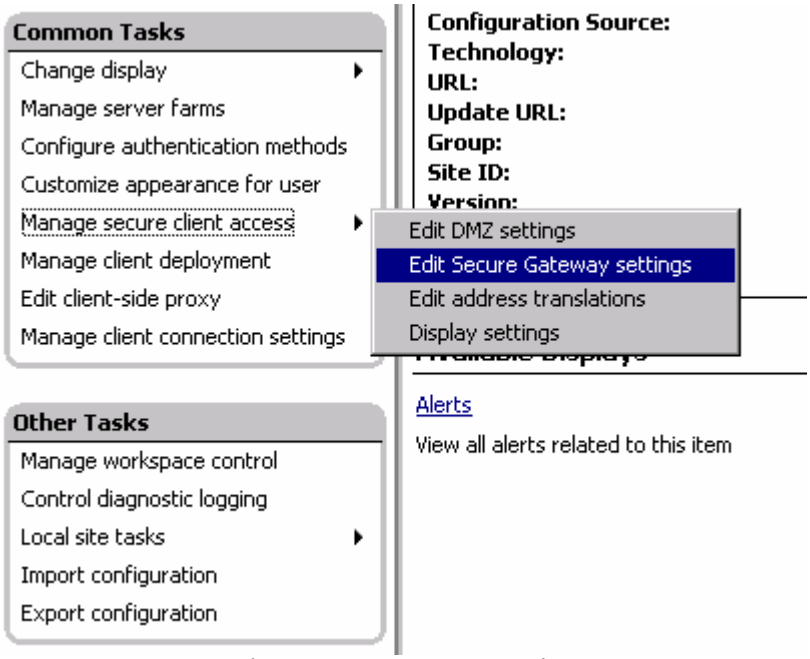


Secure Gateway solution architecture

The *Secure Gateway Administrator's Guide* discusses the secure gateway solution in detail. The purpose of this section is to examine how Web Interface enables secure gateway connectivity.

Note that the Secure Ticket Authority is integrated into the Citrix XML Service with Presentation Server 4.0, and may therefore be consolidated with the MetaFrame XML Broker in the diagram above.

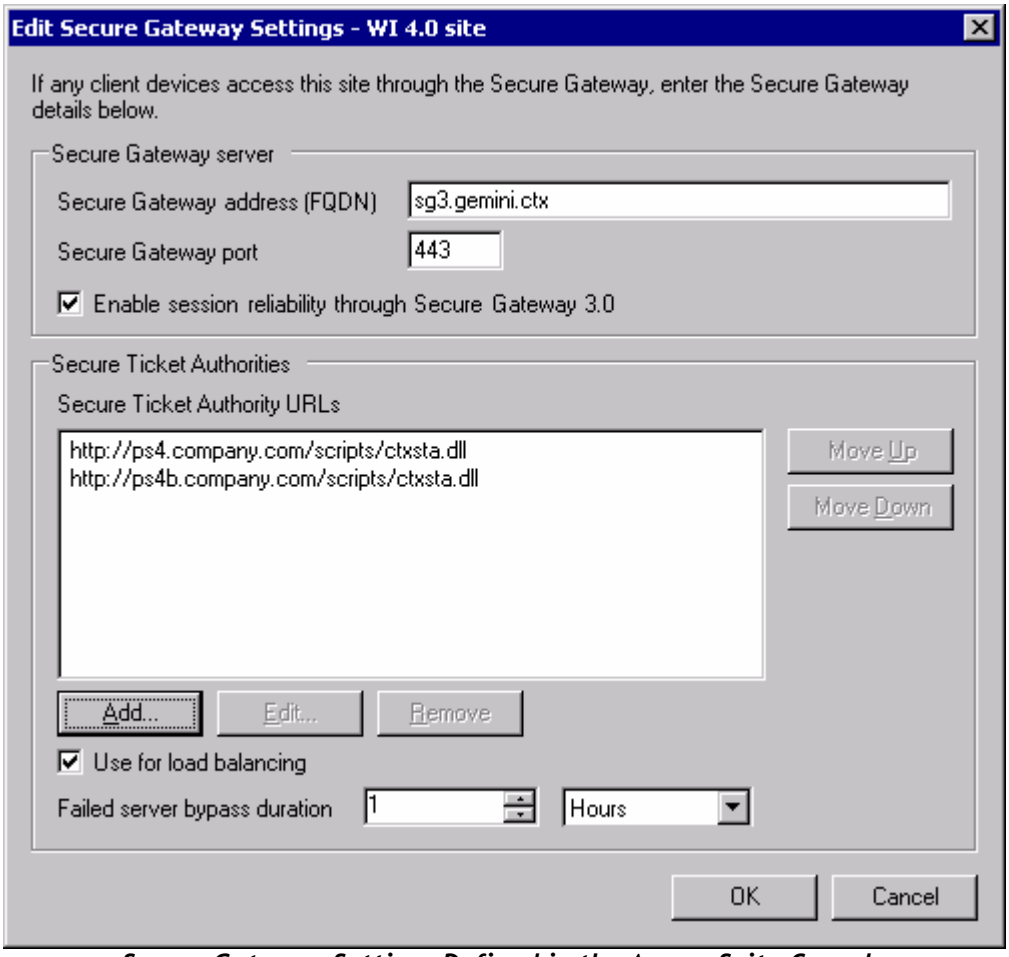
To define the Secure Gateway FQDN and STA location(s) in Web Interface, select Manage Secure Client Access > Edit Secure Gateway Settings in the Access Suite Console:



Edit Secure Gateway Settings

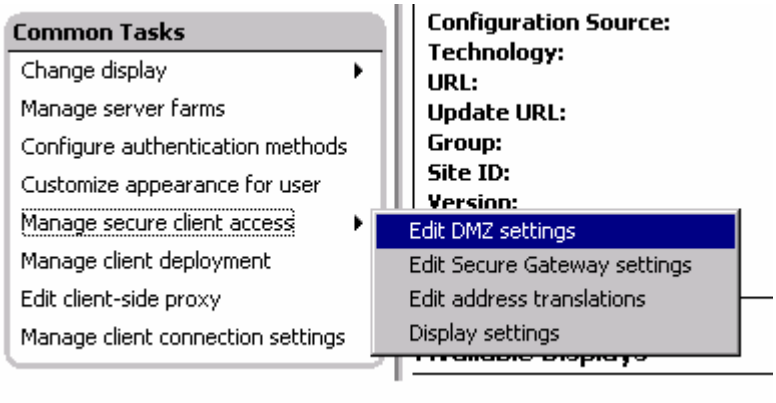
In the dialog that appears, enter the fully qualified domain name (FQDN) of the gateway, which should match the subject of the certificate installed on the gateway. Remote clients must be able to resolve this FQDN to the external IP address of the Secure Gateway server(s). Enter one or more STA URLs of the form http://sta-server/Scripts/CtxSta.dll.

When the STA is hosted by the Presentation Server 4.0 XML service running on a port other than 80, include the port number after the server address in the URL. For example, when the Citrix XML Service is listening on port 8080, the STA URL would be http://ps4.server.org:8080/Scripts/CtxSta.dll.



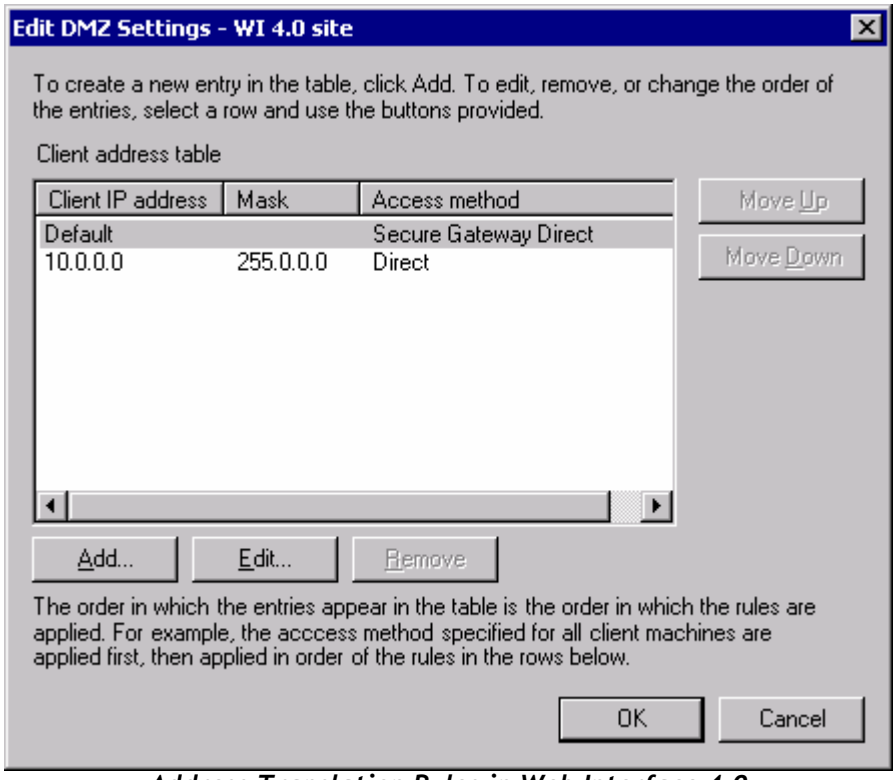
Secure Gateway Settings Defined in the Access Suite Console

Once the gateway FQDN and STA URLs are defined, enable Secure Gateway connectivity by creating address translation rules in the DMZ settings to direct some or all of your users through the gateway. To do so, select Manage Secure Client Access > Edit DMZ Settings:



Edit DMZ settings

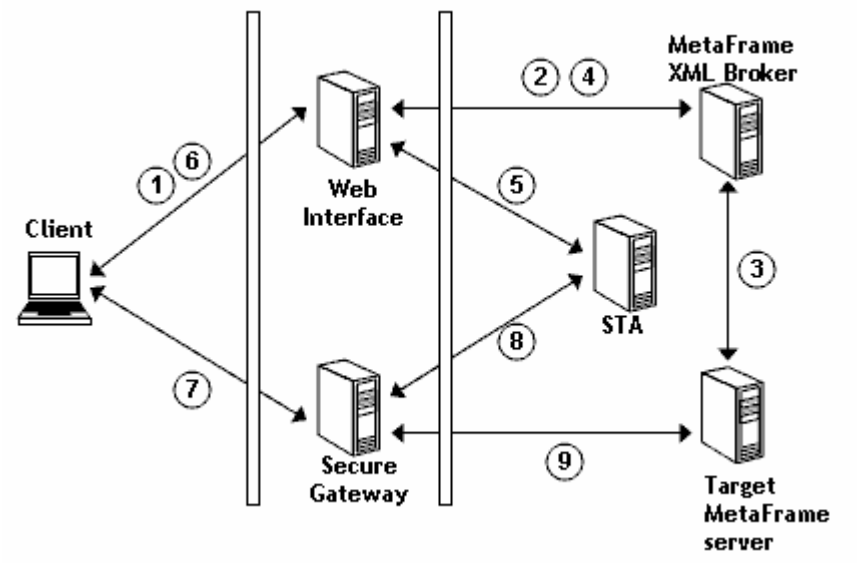
Change the default connection method to Secure Gateway Direct. You may also wish to add an exception rule that allows users on the trusted network to bypass the gateway and connect directly to Presentation Servers:



Address Translation Rules in Web Interface 4.0

How it works

The following diagram illustrates the process by which Web Interface produces an ICA file intended for use with Secure Gateway:



Secure Gateway Connection Process

1. Having authenticated to Web Interface, the user clicks an application icon.

2. Web Interface contacts the XML broker to determine the address of the target MetaFrame server.
3. The XML broker locates the least-busy server for the chosen application and requests a MetaFrame logon ticket for that server.
4. The address of the target MetaFrame server and a corresponding MetaFrame logon ticket are returned to Web Interface.
5. Web Interface sends the target server address, user name, domain name and published application name (collectively referred to as "the data") to the STA and gets a gateway traversal ticket in return.
6. Web Interface renders an ICA file for the user containing the gateway traversal ticket in the Address field. Also included in the ICA file are the following lines that instruct the client to connect to a gateway:

   ```
   SSLEnable=On
   SSLProxyHost=csg.company.com:443
   ```

   The fully-qualified domain name of the Secure Gateway server is drawn from the `CSG_Server` value in WebInterface.conf.
7. The client makes an ICA-in-SSL connection (not HTTPS!) to the gateway server on port 443 and performs an SSL handshake. The gateway server sends its server certificate chain to the client; the client must have the appropriate CA root certificate in order for the SSL handshake to succeed.
8. The gateway server extracts the gateway traversal ticket from the user's ICA file and sends it to the STA for redemption. The gateway receives the data from the STA corresponding to the current ticket. The ticket is then purged from the STA's memory immediately.
9. Having validated the user's ticket, the gateway opens a TCP connection to the MetaFrame server's ICA port and forwards decrypted ICA traffic to the server. A relay is established with the gateway providing encryption/decryption service between the client and the target MetaFrame server. The MetaFrame logon ticket is supplied to initiate the ICA session without re-authentication.

As with non-gateway connections, the role of Web Interface is only to foster the ICA connection. Once an ICA session is established, Web Interface is no longer plays an active role in maintaining the user's ICA session.

How it breaks

Web Interface configured with incorrect gateway FQDN

When configuring Web Interface to support secure gateway connections, the administrator must enter the address of the Secure Gateway server. The address entered must match the FQDN which appears as the Subject of the gateway server certificate. If the FQDN entered does not match the certificate on the gateway, users will receive the following error from the ICA client when an application is launched: "Security alert: The name on the security certificate does not match the name of the server (SSL error 59)."

Known limitations and issues

- Web Interface supports the definition of only a single secure gateway FQDN. The ability to route users through different gateways based on farm, application name or user location is possible only through custom script development.

4.2 Secure Gateway Ticketing

4.2.1 Ticket Types

"Ticket" has become an overloaded term. For the purpose of this paper, we shall use the following names to distinguish among the various ticket types (these are not official Citrix terms):

Name	Issued by	Purpose
Logon Ticket	Presentation Server (MetaFrame 1.8 Feature Release 1 or later)	Authenticate user to MetaFrame session; ticket replaces user credentials
ACR Ticket	MetaFrame Presentation Server	Allow reconnection via Auto Client Reconnect without requiring user to enter credentials
Gateway Traversal Ticket (v1)	STA 2.2 or earlier	Allow ICA connection through Secure Gateway; ticket replaces MetaFrame server address
Common Gateway Protocol Token	Citrix XTE Service	Facilitate secure reconnection to a disconnected Common Gateway Protocol session
Gateway Traversal Ticket (v4)	STA 4.0 or later, or MetaFrame Secure Access Manager 3.0 Authentication Service	Allow ICA connection through Secure Gateway with Session Reliability; ticket replaces server address

The Web server requests a gateway ticket from the STA each time the user launches a new application, and the secure gateway validates the ticket during ICA session initiation. Web Interface versions 3.0 and earlier always request an STA v1 Ticket; Web Interface 4.0 with Session Reliability enabled will attempt to request an STA v4 Ticket. If the v4 request fails, Web Interface tries again with a v1 request.

The following data are included as part of the ticket request sent by the Web server:

- User name and domain name
- Published application name
- Least-busy Presentation Server address

The STA retains this data in memory and issues a random v1 or v4 ticket in exchange. When the user connects to secure gateway, this ticket is presented as part of the ICA file. Secure Gateway sends the ticket back to the STA for validation and receives the data in return.

See CTX101997 for a list of frequently asked questions about the Secure Ticket Authority.

4.2.2 How it works

After authenticating the user and identifying the least-busy MetaFrame Presentation Server for the desired application, the Web server renders an ICA file for the user. If address translation rules on the Web server determine that the user should connect via Secure Gateway, the Web server will send an XML request to the STA as part of the ICA file rendering process. The XML ticket request resembles the following:

```
<?xml version="1.0" encoding="UTF-16"?>
<!DOCTYPE CtxSTAProtocol SYSTEM "CtxSTA.dtd">
<CtxSTAProtocol version="1.0">
    <RequestTicket>
        <AllowedTicketType>STA-v1</AllowedTicketType>
        <AllowedAuthorityIDType>STA-v1</AllowedAuthorityIDType>
        <Data>10.3.1.231:1494</Data>
        <XData>&lt;?xml version=&quot;1.0&quot;?&gt;&lt;!--DOCTYPE
CtxConnInfoProtocol SYSTEM &quot;CtxConnInfo.dtd&quot;--
&gt;&lt;CtxConnInfo
version=&quot;1.0&quot;&gt;&lt;ServerAddress&gt;10.3.1.231:1494&lt;/Server
Address&gt;&lt;UserName&gt;jayt&lt;/UserName&gt;&lt;UserDomain&gt;TOMLIN&l
t;/UserDomain&gt;&lt;ApplicationName&gt;PowerPoint&lt;/ApplicationName&gt;
&lt;Protocol&gt;ICA&lt;/Protocol&gt;&lt;/CtxConnInfo&gt;</XData>
    </RequestTicket>
</CtxSTAProtocol>
```

The <Data> section contains the address of the least-busy Presentation server as determined by Web Interface and the <XData> section contains another encoded XML document summarizing the domain name, user name, server address and published application name. The extended data is consumed by the Secure Gateway service in order to provide details about each gateway session in the Secure Gateway management console.

The above XML document is sent via HTTP POST to the STA URL defined in WebInterface.conf. Typically the STA URL resembles the following:

```
http://10.6.7.8/Scripts/CtxSta.dll
```

Upon receiving this request, the STA will generate a random ticket and return the ticket to the Web server in an XML response document. For example:

```
<?xml version="1.0"?>
<!DOCTYPE CtxSTAProtocol SYSTEM "CtxSTA.dtd">
<CtxSTAProtocol version="1">
  <ResponseTicket>
    <AuthorityID authorityType="STA-v1">STA01</AuthorityID>
    <Ticket ticketType="STA-v1">245489CECBC3CAA3B88446F12FF80B6A</Ticket>
    <TicketVersion>40</TicketVersion>
  </ResponseTicket>
</CtxSTAProtocol>
```

Web Interface receives three pieces of information this response:

1. The version of the ticket authority (40)
2. The unique ID of this ticket authority (STA01)
3. The ticket for the current connection (245489CECBC3CAA3B88446F12FF80B6A)

These data are consolidated into a semicolon delimited string and substituted for the address parameter in the rendered ICA file:

```
Address=;40;STA01;245489CECBC3CAA3B88446F12FF80B6A
```

Additional entries are added to the ICA file, which cause the client to connect to Secure Gateway rather than attempting a direct connection to a MetaFrame Presentation Server:

```
SSLEnable=On
SSLProxyHost=sg.company.com:443
```

After completing the SSL handshake at the Secure Gateway server, the ticket is presented for validation.

The Secure Gateway service must validate the ticket before establishing a relay for ICA traffic. The gateway first resolves the STA ID (for example, STA01) into a URL. If more than one STA is used in a deployment, it is important that each STA have a unique ID so that when validating a ticket the Secure Gateway server always returns to the STA which issued that ticket. The ticket is encapsulated into an XML Data Request document and sent to the STA:

```
<?xml version="1.0"?>
<!DOCTYPE CtxSTAProtocol SYSTEM "CtxSTA.dtd">
<CtxSTAProtocol version="1.0">
 <RequestData>
  <Ticket ticketType="STA-v1">245489CECBC3CAA3B88446F12FF80B6A</Ticket>
  <TicketVersion>40</TicketVersion>
 </RequestData>
</CtxSTAProtocol>
```

The STA validates the ticket and returns the data, including the MetaFrame Presentation Server address, to the gateway:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE CtxSTAProtocol SYSTEM "CtxSTA.dtd">
<CtxSTAProtocol version="1">
    <ResponseData>
      <Data>10.3.1.231:1494</Data>
      <XData>&lt;?xml version=&quot;1.0&quot;?&gt;&lt;!--DOCTYPE
CtxConnInfoProtocol SYSTEM &quot;CtxConnInfo.dtd&quot;--
&gt;&lt;CtxConnInfo version=&quot;1.0&quot;&gt;&lt;ServerAddress&gt;
10.3.1.231:1494&lt;/ServerAddress&gt;&lt;UserName&gt;jayt&lt;/UserName&gt;
&lt;UserDomain&gt;TOMLIN&lt;/UserDomain&gt;&lt;ApplicationName&gt;PowerPoi
nt&lt;/ApplicationName&gt;&lt;Protocol&gt;ICA&lt;/Protocol&gt;&lt;/CtxConn
Info&gt;</XData>
    </ResponseData>
</CtxSTAProtocol>
```

The gateway now knows which Presentation server the user should be relayed to. The gateway connects to MetaFrame and an ICA session is established.

4.2.3 How it breaks

IIS is misconfigured

The XML requests shown in the section above are sent to the STA anonymously. If an IIS server hosting the STA is configured to require authentication, or if for any reason the /Scripts/CtxSta.dll resource is not available, Web Interface returns the following error:

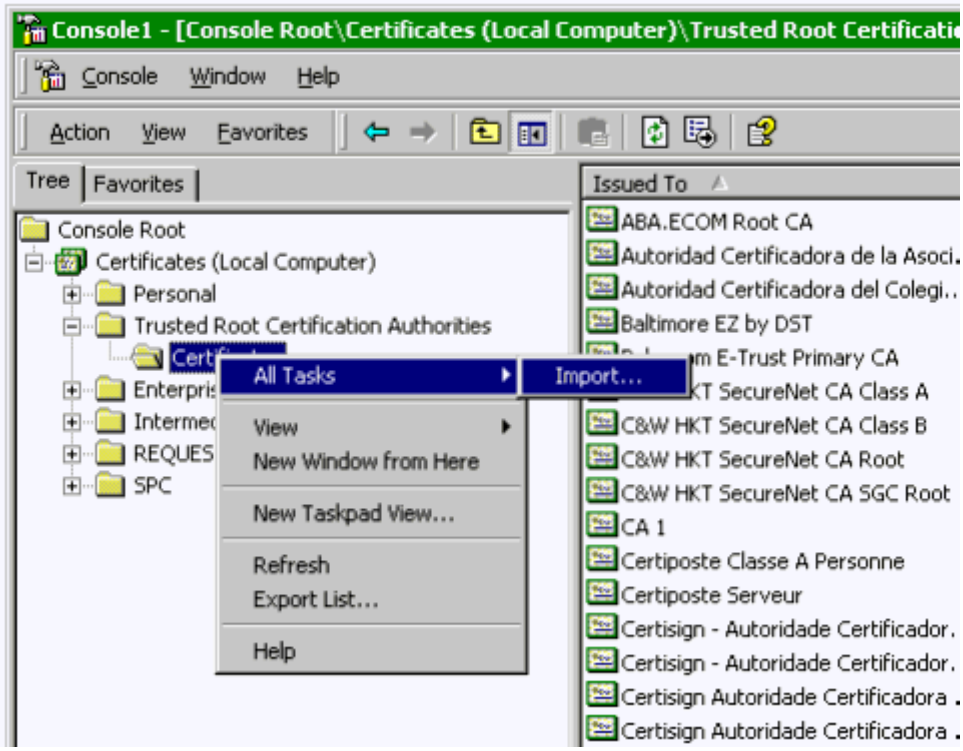
ERROR: None of the configured Secure Gateway for MetaFrame Presentation Server STAs are available

See CTX339681 for a list of reasons why this error message may be encountered.

Web server requires a root certificate

When encrypting traffic between a Web server and the STA, the STA URL must begin with HTTPS and the Web server requires a CA root certificate corresponding to the STA's server certificate. In most cases the STA server certificate will be issued by a private certificate authority so a root certificate must be installed on the Web server.

When installing the root certificate, you must use the MMC Certificates snap-in and import the certificate into the **Certificates (Local Computer) > Trusted Root Certification Authorities > Certificates** store. If you double-click the root certificate in Windows Explorer to launch the certificate import wizard, the root certificate will be installed into your user profile and will be inaccessible to Web Interface. Be certain that the MMC snap-in is used instead:



Use the MMC Certificates snap-in to import root certificates

Note: Under Web Interface for UNIX, ensure that the root certificate is exported to a file and saved beneath the keystore\cacerts folder defined in WebInterface.conf.

4.2.4 Known limitations and issues

- When using Secure Gateway for Solaris or Citrix Secure Gateway for Windows version 1.x, traffic to the STA must not be encrypted with HTTPS. Web Interface 2.0 or later and Secure Gateway 2.0 and later are able to encrypt STA XML requests.
- When the STA is delivered via IIS 6.0, it may be necessary to add the STA as an allowed Web service extension. To do so:
    1. Launch **IIS Manager** and select the **Web Service Extensions** node.
    2. Click **Add a new web service extension…**
    3. For the extension name, enter "STA" or any name you wish.
    4. For Required Files, add the path to **\Inetpub\Scripts\CtxSta.dll**.
    5. Set the extension status to allowed and click **OK**.

4.2.5 Frequently Asked Questions

Does the STA require IIS?

Prior to the release of Presentation Server 4.0, the STA could only be hosted by IIS. In Presentation Server 4.0, the STA is bundled with the Citrix XML Service, which may run

as a separate service (ctxhttp.exe) or "share a port" with IIS, meaning wpnbr.dll, CtxSta.dll and other ISAPI plugins are served by IIS from the Inetpub\Scripts folder.

See CTX101997 for other frequently asked questions about the Secure Ticket Authority.

4.3 Session Reliability through the gateway

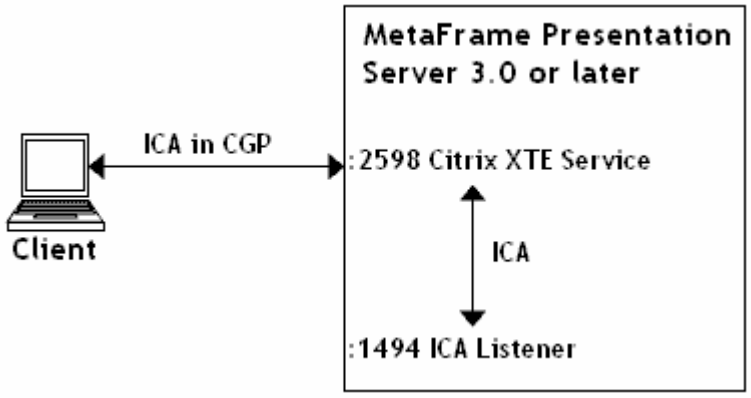
MetaFrame Presentation Server 3.0 introduced the Session Reliability feature, which is part of the "Smooth Roaming" capabilities of Presentation Server. With session reliability enabled, short-term network interruptions are silently repaired without putting the user's session into a disconnected state.

4.3.1 Session Reliability without Secure Gateway

To understand how session reliability works through Secure Gateway, it is helpful to understand how it works in the absence of a gateway. Reliability is provided by the Citrix XTE Service, which adds a Common Gateway Protocol listener on port 2598 of the MetaFrame Presentation Server.

> ***FYI***
> *The XTE Service is architecturally identical to Secure Gateway 3.0: both are based on Apache and support Common Gateway Protocol and SSL connections. In addition to providing the session reliability feature, the XTE Service also replaces the SSL Relay service in MetaFrame Presentation Server 3.0 and later.*

When session reliability is enabled, the ICA Client tunnels its ICA traffic inside the Common Gateway Protocol and sends the traffic to port 2598. The XTE service acts as a relay, unwrapping the Common Gateway Protocol layer and then forwarding traffic to the ICA listener on port 1494:



Likewise, the Presentation Server sends ICA data to the client by way of the XTE service. If the Common Gateway Protocol connection between the client and the XTE service is broken, the ICA listener can continue to send ICA traffic to the XTE service, where it will be buffered until the client reconnects. The user's session does not go into a disconnected state as long as the XTE service is buffering data for the user.

On the client, the user's application will appear frozen while the client attempts to rebuild the Common Gateway Protocol connection. Once the connection is restored, the XTE service flushes the buffered ICA data to the client and the session continues.

4.3.2 Frequently Asked Questions about Session Reliability

How long will the XTE service keep a user's session in an active state, waiting for the client to reconnect?

The default is 180 seconds, and this can be configured on a farm-wide basis by editing the Farm properties in the Presentation Server Console. Changes require a restart to take effect.

What if the client doesn't support session reliability?

The feature is enabled in ICA or appsrv.ini files by setting the parameter **CGPAddress=*:2598**. This tells the client to wrap its outgoing traffic in Common Gateway Protocol and send it to the MetaFrame server on port 2598. Clients which do not support session reliability will ignore this parameter and connect to port 1494 without session reliability as before.

What if the XTE Service isn't started?

If the client is unable to connect to the XTE Service on port 2598, it automatically disables session reliability and attempts a direct ICA connection to port 1494.

I noticed that a Common Gateway Protocol reconnection does not require authentication. What prevents an attacker from reconnecting to my disconnected Common Gateway Protocol session?

When a Common Gateway Protocol session is first established, the XTE server sends a secure Common Gateway Protocol token to the client to be stored in memory. When the client reconnects, this token is required to resume the disconnected session.

How does this affect firewalls and proxy servers?

For session reliability to work, any firewalls between the client and MetaFrame Presentation Server should allow the client to connect on port 2598. Proxy servers need to allow the CONNECT method on port 2598. If this is not the case, clients will fail over to direct ICA connections without session reliability.

Which clients support session reliability?

Only the following clients support session reliability at this time:

- Client for 32-bit Windows version 8.0 or later
- Client for Java version 9.0 or later

How is session reliability different from Auto Client Reconnect?

Session reliability:

- is intended only for short-term network interruptions
- maintains the user's session in an active state at the web server
- maintains the application on the user's screen while rebuilding the connection
- requires the 8.0 Win32 client and MetaFrame Presentation Server 3.0 or later

Auto Client Reconnect:

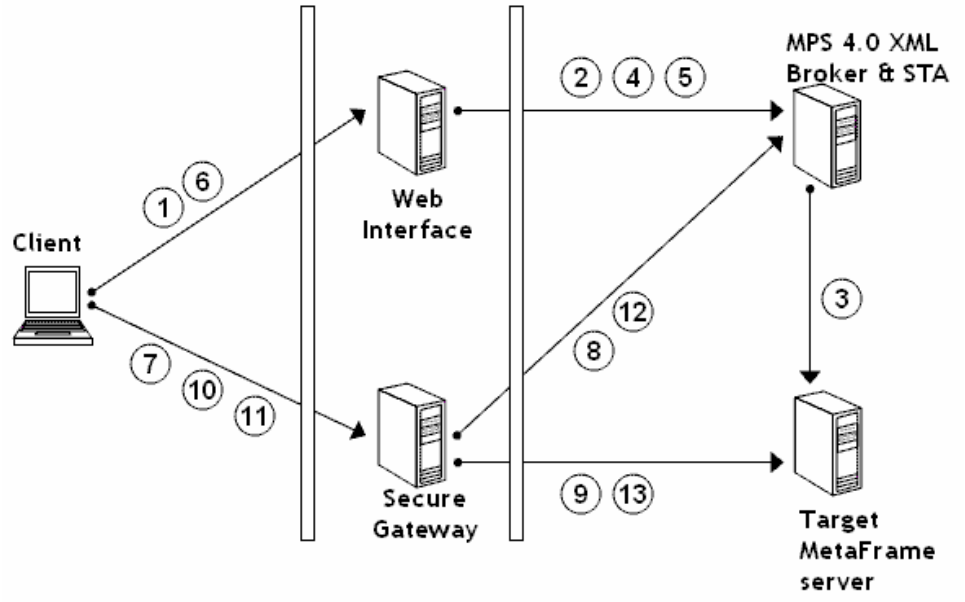
- copes with long-term network interruptions by retrying multiple reconnects
- can only reconnect users to a session that is in a disconnected state at the server
- allows the user's application to disappear and shows a reconnect dialog to inform the user that a reconnect is taking place
- may be configure to require re-authentication

- requires the 6.20.985 Win32 client and MetaFrame XP Feature Release 1 or later

If both session reliability and auto client reconnect are enabled, the features work in sequence: auto client reconnect will engage after the session reliability timer has expired.

4.3.3 Session Reliability through Secure Gateway 3.0

Session reliability can not be used when connecting through Secure Gateway 2.0 or earlier. Secure Gateway 3.0 introduces the ability to use the session reliability (but not auto client reconnect) through the gateway. The following diagram illustrates the procedure for establishing a Common Gateway Protocol tunnel through the gateway with Web Interface 4.0 and how the tunnel is restored after a network interruption. The protocol used in each step appears in [square brackets] after the description. Note that all HTTP or XML connections may optionally be secured using SSL.



1. User clicks an icon to launch an application. [HTTP]
2. Web Interface determines the address of the least-busy MetaFrame Presentation server and requests a MetaFrame logon ticket. [XML]
3. XML broker requests a MetaFrame logon ticket from the target MetaFrame server. [IMA]
4. MetaFrame logon ticket is delivered to Web Interface by the XML broker. [XML]
5. Web Interface sends target server address and other data to STA, receives a V4 Connection Ticket in response. [XML]
6. A rendered ICA file is sent to the user containing the MetaFrame logon ticket and the STA connection ticket. [HTTP]
7. The ICA Client is invoked and connects to Secure Gateway, presenting the STA connection ticket [ICA+CGP+SSL]
8. Secure Gateway validates the connection ticket presented by the client. Immediately after validation, the gateway requests a V4 Refreshable Reconnect ticket from the STA to be used in case the SSL connection is severed. This reconnect ticket is periodically refreshed while the user's session is active. [XML]

9. Secure Gateway tunnels the Common Gateway Protocol connection to MetaFrame Presentation Server. The Citrix XTE Service generates a Common Gateway Protocol token to be used for reconnection in case the Common Gateway Protocol link is severed. [CGP]
10. The Common Gateway Protocol token from step 9 and the reconnect ticket obtained in step 8 are sent to the client and stored in memory. Connection to MetaFrame Presentation Server application is now established. [ICA+CGP+SSL]

    --- Network connection is temporarily severed ---

11. After a network interruption, the client initiates a new SSL connection to the gateway and presents the STA reconnect ticket obtained in step 8. [ICA+CGP+SSL]
12. The gateway validates the reconnect ticket to allow a new TCP session to the MetaFrame server. Immediately after validation, the gateway requests a new V4 Refreshable Reconnect ticket from the STA to be used in case the SSL connection is severed again. This reconnect ticket is periodically refreshed while the user's session is active. [XML]
13. Secure Gateway makes a new connection to the Citrix XTE Service and presents the Common Gateway Protocol token obtained in step 9. The XTE Service associates the token with the user's disconnected Common Gateway Protocol session and restores the Common Gateway Protocol tunnel. [CGP]

4.3.4 Session Reliability System Requirements

- **Presentation Server 3.0 or later** – The XTE Service was introduced in MetaFrame Presentation Server 3.0, and can be used to deliver session reliability with Secure Gateway 3.0.
- **Web Interface 4.0** – earlier versions of Web Interface are unable to request the V4 ticket required for establishing a reliable session through the gateway. Web Interface must also add new parameters into the rendered ICA file in order to enable reliability through SSL.
- **STA version 4.0** – earlier versions are not able to produce or validate refreshable tickets
- **Win32, WinCE, or Java ICA Client version 9.0** – Older clients disable session reliability and use SOCKS instead of Common Gateway Protocol when connecting to a Secure Gateway server. All variants of the 9.0 Win32 client can use session reliability through Secure Gateway 3.0 (Web client, Program Neighborhood Agent or Program Neighborhood).
- **Firewall ports** – Secure Gateway must be able to connect to each MetaFrame Presentation Server on port 2598.

4.4 Relay Mode

When Secure Gateway operates in Relay Mode, ICA clients may traverse the gateway without requiring a ticket from the STA. This obviates the need for Web Interface or any other Web server, enabling use of the gateway by full Program Neighborhood clients. Relay mode is recommended for internal-only deployments, where all traffic is considered trusted by default. Some typical uses of relay mode include:

- **Encrypt wireless traffic** – WiFi clients connect wirelessly to a Secure Proxy server, and then the proxy connects to MetaFrame on the wired network, ensuring all wireless traffic is secured with SSL.
- **Internal firewall traversal** – when a NAT firewall separates two or more segments of a trusted network (multiple divisions of the same company, for example), relay mode can serve as an alternative to defining an alternate address for each MetaFrame Presentation server in the farm.

4.4.1 How it works

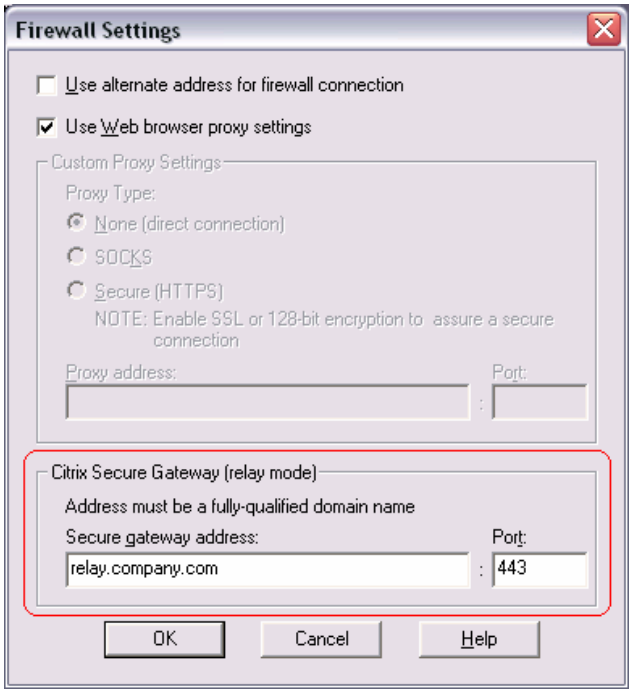
Relay mode is achieved by installing Secure Gateway in Proxy mode with a certificate, which listens on port 443 for SOCKS+SSL connections. Note that a relay mode gateway cannot be used as a reverse Web proxy for Web Interface, and Web Interface will not foster relay-mode connections without a custom change to the launch script or ICA override files.

4.4.2 Known limitations and issues

- Session reliability does not work through a Relay Mode gateway.

4.4.3 Frequently Asked Questions

How do I configure Program Neighborhood to use Secure Gateway in relay mode?
Click the **Firewalls…** button and provide the gateway details:



How do I configure Web Interface to use Secure Gateway in relay mode?
For Web Interface 3.0 and earlier:
Do not enable Secure Gateway support in WIAdmin. Modify the template.ica file(s) and add the following lines after the [NFuse_IcaWindow] tag:

```
SSLEnable=On
SSLProxyHost=csg.server.com:443
```

…where csg.server.com matches the subject of the certificate installed on your relay mode gateway server.

For Web Interface 4.0:
Do not enable Secure Gateway support in the Access Suite Console. Modify default.ica and all other ICA override files and add the following lines to the [Application] section:

```
SSLEnable=On
SSLProxyHost=csg.server.com:443
```

5. Digital Certificates

SSL is responsible for more than just data encryption: connections must be authenticated as well. The cornerstone of SSL authentication is trust. When a client initiates a connection to a server or service, the client must trust that attackers have not compromised the network or that the service to which the client connects is in fact the entity they think it is. The mechanism for creating and exercising this trust is provided by Digital Certificates.

A good analogy for the SSL trust model exists in everyday life: your driver's license. Clear parallels exist when you compare the procedures for obtaining and using a driver's license with those for obtaining and using a digital server certificate:

	Driver's License	**SSL Server Certificate**
Issued by	Department of Motor Vehicles (DMV)	Certifying Authority (CA)
Primary purpose	Personal identification in society and vehicle operation on public roads	Digital identification on the Internet and encryption of data on public networks
Document contains	Your name, address, and photograph	Server name (FQDN) and company name
Proof required during application	Birth certificate, Social Security card, or other official documents provided by the applicant	Business license, verified contact information, or other means of verification provided by the business organization
Document authenticity verified by	DMV hologram or other formatting features imprinted on the license that would be very difficult to forge	Digital signature imprinted on the certificate using a lengthy private key string that would be very difficult to reproduce
Third parties honor the document based on	A common license format and hologram known to all as published by the DMV	A common root certificate made publicly available from the CA
Proof of identity during usage	Photo must match my appearance	FQDN must match the server to which I connect
Validity controlled by	Expiration date	Expiration date, certificate revocation list
Cost to the user	Taxes or fees paid to the DMV	Fees paid to the CA

When someone shows you their driver's license, you trust that they are capable of driving a car on public roads. This feeling of trust is based on the notions that:

- You recognize the format of their driver's license and believe it was issued by the DMV

- You trust the DMV to scrutinize applicants and not to issue a license to someone who is incapable of driving a car

In other words, you trust the license because you trust the DMV.

Digital Certificates operate on this same model of trust-by-proxy. By signaling your trust in certain Certifying Authorities (CAs), you implicitly trust any certificate issued by that CA.

Suppose you are a client wishing to make an SSL-protected connection into an organization. The CA expresses its verification of the organization by issuing to that organization a certificate that has been "signed" by the CA. End-users express their trust in the CA by installing the CA's root certificate on their machine or in their Web browser. During the client's connection request an SSL handshake takes place in which the server sends its signed certificate to the client. Since the certificate contains the signature of a trusted CA, the connection continues.

The Windows operating system and many Web browsers come preconfigured with a set of root certificates from reputable CAs. The list of CAs trusted by default include, but are not limited to:

- Thawte
- VeriSign
- GeoTrust
- EnTrust

These companies operate as public CAs, not unlike public DMV offices. To save money, companies may choose to act as their own CA. To do so, they must install and use their own certificate-generating service. Microsoft provides such a service with Microsoft Certificate Services, an optional Windows component. When using your own certificate server, the onus for distributing your CA root certificate to clients falls upon you.

5.1 Certificate chain validity requirements

There are many different types of certificates: root certificates, client certificates, server certificates, etc, and many possible uses for certificates, such as server authentication, client authentication, email signing, code signing and certificate signing. Well-known standards exist (RFC 3280) which strictly define the actions for which a certificate chain may be used. The Presentation Server client carefully scrutinizes certificate chains upon connection to ensure they adhere to these standards.

5.1.1 How it works

Each certificate in a chain is classified as one of the following:

Identity	The first certificate in a chain.
Client Identity	The first certificate in a Client Authentication chain.
Intermediate	Used to sign a certificate.
Root	Used to sign a certificate and self signed.

The following properties must hold true for each certificate in a chain for the chain to be acceptable:

Enhanced (Extended) Key Usage field

Identity	If present must include one of: ▪ Server Authentication ▪ All Key Usages
Client Identity	If present must include one of: ▪ Client Authentication or ▪ All Key Usages
Intermediate	Must not be critical.
Root	Must not be critical.

Key Usage Field

Identity	If present must include (RSA): ▪ Key Encipherment
Client Identity	If present must include (RSA): ▪ Digital Signature
Intermediate	Must be present and include: ▪ Certificate Signing
Root	If present must include: ▪ Certificate Signing

Basic Constraints Field

Identity	Ignored
Client Identity	Ignored
Intermediate	Must be present Must define CA=true
Root	If present: Must define CA=true

Path Length Limit

Basic Constraints Field not present	No limit.
CA=false	Path Length = 0, unsuitable for ▪ Intermediate ▪ Root
CA=true, no Path Length Limit	No Limit
CA=true, Path Length Limit n	Limit onward chain to n+1 certificates.

Minimum Key Length

No certificate may have public key of length less than 512 bits.

Netscape Certificate Usage Field

Identity	If present must include NETSCAPE_SSL_SERVER_AUTH
Client Identity	If present must include NETSCAPE_SSL_CLIENT_AUTH
Intermediate	If present must include NETSCAPE_SSL_CA_CERT If position in chain >= 2 must include NETSCAPE_SIGN_CA_CERT_TYPE
Root	If present must include NETSCAPE_SSL_CA_CERT If position in chain >= 2 must include NETSCAPE_SIGN_CA_CERT_TYPE

Unrecognized Extensions
No certificate may have an unrecognized extension that is marked critical.

Date Checking
All certificates must be valid at the current time.

5.1.2 How it breaks

Users will receive ***SSL Error 61*** if the certificate chain does not meet the requirements set out above. Citrix Technical Support has encountered two particular certificate faults:

1. The Enhanced Key Usage field is present in an Identity certificate but the field does not contain "Server Authentication."
2. An Intermediate certificate has been used to sign other certificates but does not contain the "Certificate Signing" key usage.

5.1.3 SSLv3 versus SSLv1

The above criteria apply only to SSLv3 certificates. A certificate chain may contain a mix of V1 and V3 certificates so long as all of the V3 certificates in the chain meet the validity requirements outlined above.

5.2 Certificate renewal and replacement

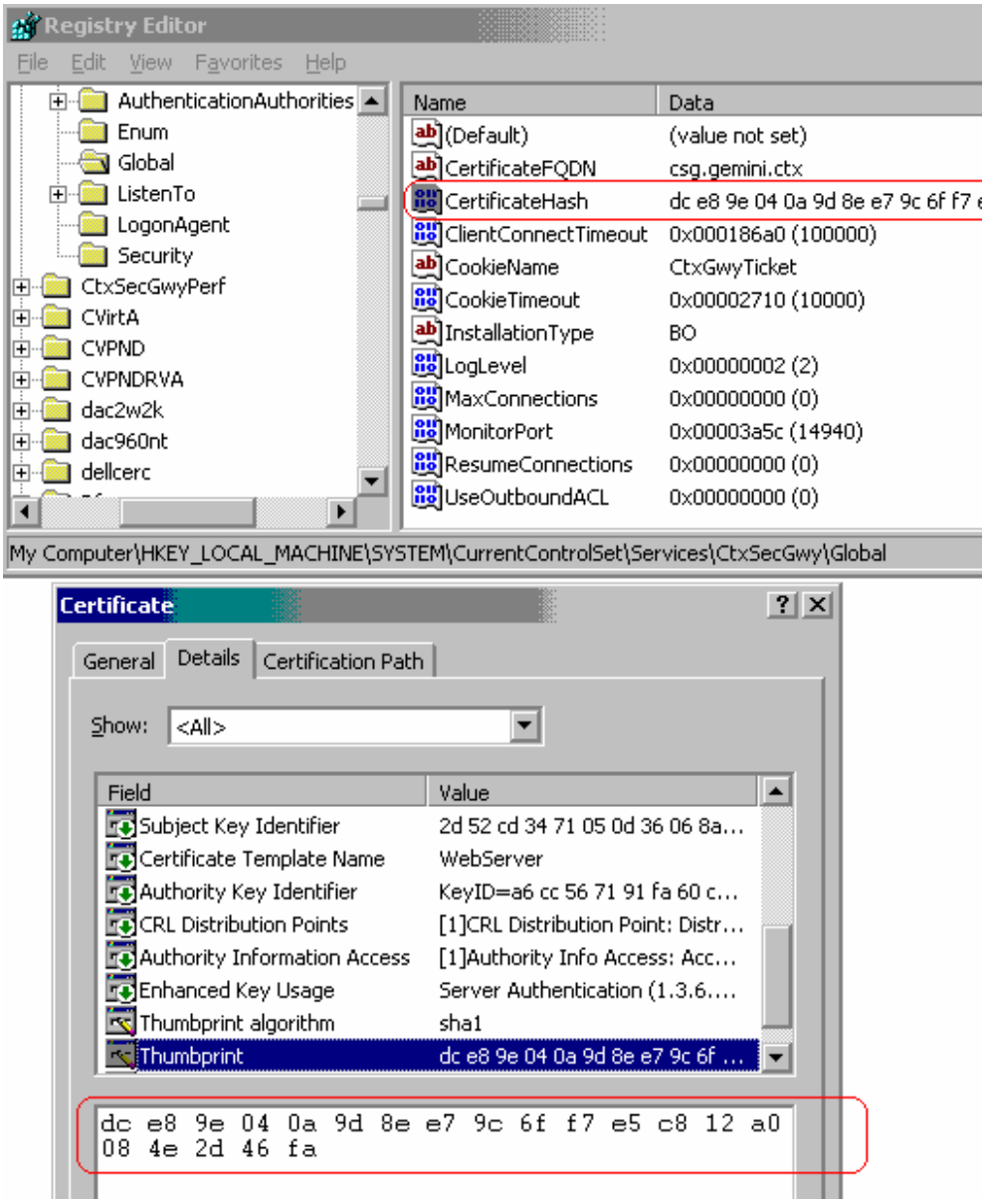
When a Secure Gateway certificate expires, you must request a renewal certificate from your CA or replace the certificate with one from a different CA. Renewal certificates are requested and installed with the same process as original identity certificates; the only real difference is that renewal certificates usually cost less than the original.

The Secure Gateway service reads certificate information when you run the Secure Gateway Configuration tool and records the thumbprint of your chosen certificate into the registry and httpd.conf as ***CertificateHash***:

From httpd.conf:

```
<VirtualHost 192.168.0.104:443>
ServerName gateway.company.com:443
# SSL Params
SSLEngine On
SSLCertificateHash dce89e040a9d8ee79c6ff7e5c812a0084e2d46fa
```

In the registry:



When the Secure Gateway service starts, it looks for a certificate with this thumbprint. Every certificate has a unique thumbprint, which is a hash of all the information contained within the certificate. A renewal certificate, even if it is issued by the same CA to the same server FQDN, will not have a matching thumbprint.

Therefore, after replacing an expired certificate with a renewal certificate of the same name, restarting the Secure Gateway service is not sufficient. The service will fail to start and append the following error to the file Program Files\Secure Gateway\logs\Error_<year>_<month>_<day>.log:

[error] Unable to load SSL Certificate for server gateway.company.com:443 [hint: SSLCertificateHash]

To correct this situation, re-run the Secure Gateway Service Configuration tool and select the new certificate. This updates the httpd.conf file with the thumbprint of the new certificate.

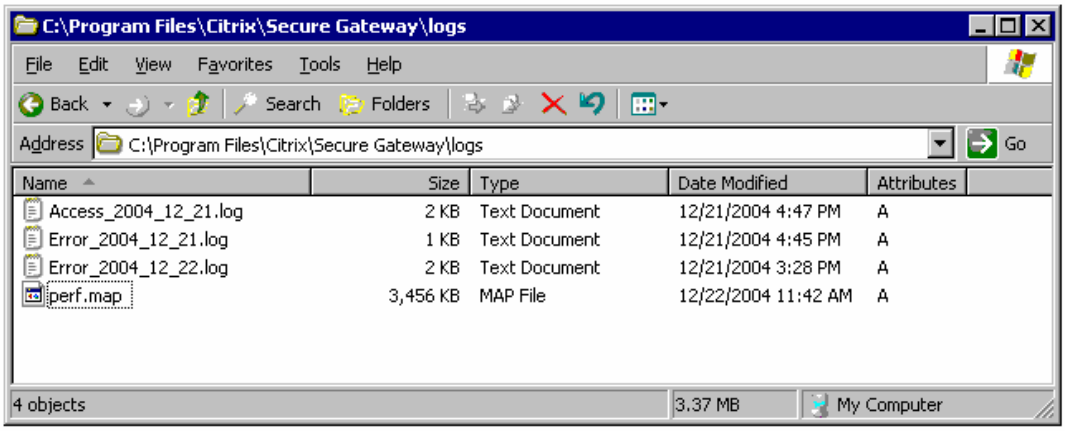
5.3 SGC Certificates

Some certificate vendors sell certificates with a feature called Server Gate Cryptography (SGC). The purpose of SGC is to "step-up" older Web browsers such as Netscape 4.x or Internet Explorer 4.x to use 128-bit encryption when they would otherwise be limited to 40-bit or 56-bit encryption. SGC is a Web-browser-only feature; an older client would not be able to "step-up" the encryption level for an ICA client connection. Therefore SGC certificates are of little use in a Secure Gateway deployment.

However, it's not entirely accurate to say that an SGC certificate is not compatible with the ICA client. As long as the client operating system is natively capable of 128-bit encryption (meaning the step-up process is not necessary), an SGC certificate will work for Secure Gateway connections.

6. Troubleshooting

When the Secure Gateway service encounters a problem, it may write error information to the event viewer, to the httpd error logs in the Program Files directory, or both.



6.1 Common error messages

6.1.1 Client reports SSL Error 4

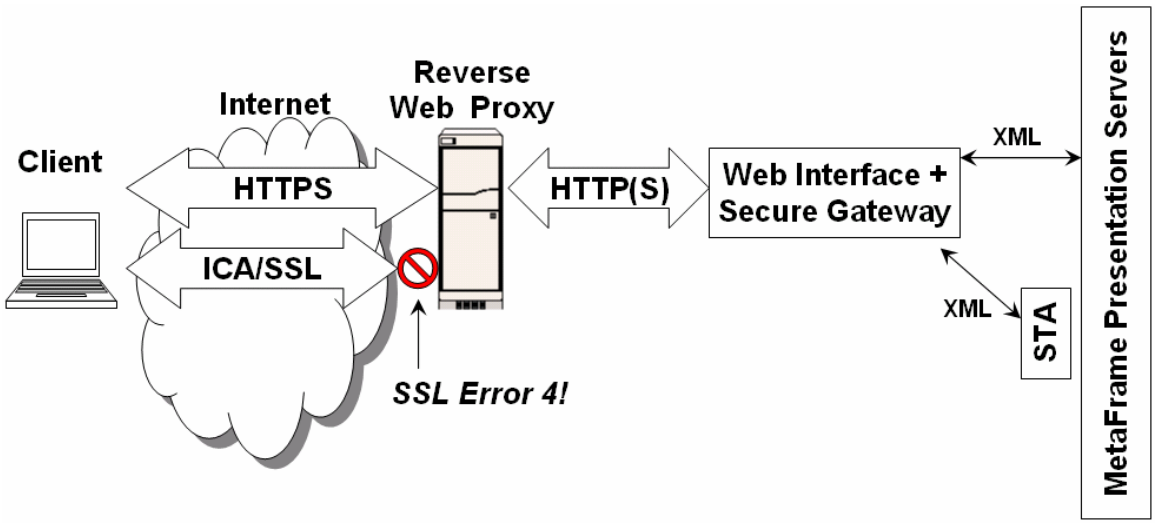
When launching a published application, the client returns SSL Error 4:

Cannot connect to the Citrix MetaFrame server.
A network error occurred (SSL error 4)

SSL Error 4 indicates that the client is connecting to a valid network service, but the service is not Secure Gateway. For example, if the client connects to an IIS listener on port 443 instead of a Secure Gateway listener, SSL Error 4 is reported.

Ensure that the address and port to which the client connects (dictated by the ***SSLProxyHost*** parameter in an ICA file) is a valid Secure Gateway service and not some other service.

This error also commonly occurs when Secure Gateway is placed behind a reverse proxy server such as ISA or Squid that is only able to proxy HTTP(S) traffic:



To correct this condition, the reverse Web proxy should be bypassed when the ICA Client connects to the server Secure Gateway. This may require separating Web Interface and Secure Gateway onto two servers: