



User Guide Part 8

Web Gateway

Contents

- 1 OVERVIEW..... 3**
- 1.1 Gateway Specifications / Requirements 3
- 1.2 Mobile Device Support..... 4
- 1.3 Connecting to the Gateway 4
- 1.4 Architecture..... 5
- 1.5 Supported Features and Limitations..... 5
- 1.5.1 Supported Features 5
- 1.5.2 Supported UI Controls 5
- 1.5.3 Workspaces and Security 10
- 1.5.4 Restrictions 11
- 2 CUSTOMIZING THE LOGIN FOR THE GATEWAY 12**
- 2.1 Setting up Automatic Log In..... 12
- 2.2 Setting a Default Mimic Path 13
- 2.3 Recommended Navigation Method 15
- 2.4 Changing the Gateway Server..... 15
- 2.5 Creating a Custom Login Page..... 17
- 2.6 Moving the Gateway to another IIS Website 18
- 2.6.1 Remove the Gateway from the current IIS Website 18
- 2.6.2 Adding the Gateway into a new IIS Website 18
- 3 REST API 20**
- 3.1 Activating the Service 20
- 3.2 Available Methods..... 21
- 3.2.1 Login 22
- 3.2.2 Logout 22
- 3.2.3 CheckSession 22
- 3.2.4 TranslateBrowsePathToNodeId 23
- 3.2.5 GetAttributes 23
- 3.2.6 GetValue 23
- 3.2.7 SetValue 24
- 3.2.8 GetType 24
- 3.2.9 CreateAsset 24
- 3.2.10 DeleteAsset 25
- 3.2.11 GetNodeIdsProperties 25
- 4 TROUBLESHOOTING 27**
- 4.1 Truncated Streams 28

Figures and Tables

Figure 1 - Specifications/Requirements	4
Figure 2 - IIS Configuration Screen	12
Figure 3 - Application Settings Screen.....	13
Figure 4 - IIS Manager	14
Figure 5 - Default Mimic Path.....	14
Figure 6 - Application Settings	15
Figure 7 - Gateway Path	16
Figure 8 - IIS Manager	20
Figure 9 - Convert to Application Option.....	21
Figure 10 - Convert to Application Dialog	21

1 Overview

This document outlines the user interface experience for the web-based Runtime Client of Status Device Cloud known as the Gateway. Status Device Cloud Gateway is an ASP.NET Application that connects to the Status Device Cloud Server and converts mimics stored as XAML at the server, to HTML5 documents for a web browser. The mimics opened have connections to real-time data. Most major browsers and most mobile devices – including the iPad – support HTML5.

For a list of frequently used terms, definitions, and abbreviations, please see Part 1 – Overview and Concepts.

NOTE: In order for the Gateway application to function, IIS must be installed before Status Device Cloud is installed.

1.1 Gateway Specifications / Requirements

XP	7 & 8
The Gateway requires the .NET 4.5 Framework	
<p>IIS Must Be Installed.</p> <p>*Steps:</p> <ul style="list-style-type: none"> • Open Control Panel > Programs and Features. • Open the “Turn Windows Features On or Off”. • Check the box next to “Internet Information Services”. • Expand Internet Information Services, under that expand World Wide Web Services, under that expand Application Development Features, here you need to make sure you have selected ALL of the following: .NET Extensibility, ASP, ASP.NET, ISAPI Extensions, and ISAPI Filters. • Port 4502 MUST to be open on the firewall for all data to come through. • Click “OK” <p>*If you don’t have permissions to edit Windows Features, contact your network administrator.</p>	
<p>RegIIS Must Be Installed</p> <p>*Steps:</p> <ul style="list-style-type: none"> • Open the Command Prompt (make sure you run as Administrator). • Type: “cd C:\Windows\Microsoft.NET\Framework64\v4.0.30319” with no quotes and press Enter. • Once the Path has Changed type: “aspnet_regiis.exe -i” with no quotes and press Enter. • ASP.NET will start installing and will say finished when done. <p>*If not using a 64bit operating system, replace 'Framework64' with 'Framework'.</p>	

Once IIS and RegIIS are done you can now install Status Device Cloud.

In IIS Manager the application pool should be set as follows:

IIS > ApplicationPools > StatusEnterpriseGateway > Advanced Settings

- **.NET Framework Ver = v4.5**
- **Identity = Network Service**

Supported Browsers:

- **Google Chrome ****Preferred******

Supported Browsers:

- **Internet Explorer***
- **Google Chrome ****Preferred******
- **Safari**
- **Opera**
- **Firefox**

*Internet Explorer 11 and above is recommended. Using previous versions such as Internet Explorer 10 may display incorrect values for certain data types.

Figure 1 - Specifications/Requirements

- When running IIS 8 and Windows Server 2012, it's strongly recommended that you install the WebSocket Protocol from your Server Roles and Features.
- Inbound/Outbound Firewall rules **MUST** be explicitly set for the server executable (StatusEnterpriseServer.exe) for any zone you're expecting to send data over.
- Static js files should use both User-Mode and Kernel-Mode caching to reduce page load times

1.2 Mobile Device Support

The Status Device Cloud Gateway can be accessed through iPads, iPhones, and any mobile device that supports HTML5, allowing direct mobile access to your mimics and live data without the need to connect to a remote client via a VPN.

1.3 Connecting to the Gateway

The first time a user tries to access the Gateway, a check will be performed to ensure that IIS is installed and configured on your system. If IIS is not installed, the user will be prompted that this installation must be completed before using the Status Device Cloud Gateway. If IIS is already installed, the Gateway client will run a check of the IIS Manager to ensure that it is configured correctly. If not, the configuration will be completed automatically before the user is given an opportunity to log in.

Your Network Administrator will provide you the web address to log into the Status Device Cloud Gateway. The default URL is: <http://localhost/StatusEnterpriseGateway> and it uses port 4502. **If Port 4502 is not open on your firewall, data and images will not be displayed.** The login page will ask you

for the username and password you normally use to log into Status Device Cloud. After logging in you will be presented with a navigation system in which you can navigate down through the folders to the document you wish to open – the current path to the folder you are in is presented at the bottom of the screen and a back button on the top left for going back one level.

A username and password can also be hard coded into the Gateway through the “web.config” file in order to have the Gateway log a specific user in automatically. (This would be specifically useful in the case of having a public user account with the Gateway hosted on a web server.) A document path can also be hard coded in the same place so that when a user logs in they are directly routed to that mimic. An additional option to set a default mobile mimic path is also optional.

1.4 Architecture

The Status Device Cloud Gateway logs into the Status Server and maintains the session for your web client when you log in. The Gateway requests the mimic from the server using the document chosen through the navigation system or specified in the “web.config”. When the document is passed to the Gateway, the Gateway converts the document from the mimic’s native format (XAML) to an HTML5 document. The HTML is returned to the client to be displayed in the web browser. A web socket connection is created between the Gateway and the web client. Any graphics in the HTML5 document that contain bindings to live data are updated through the web socket connection to the Gateway.

1.5 Supported Features and Limitations

1.5.1 Supported Features

Mimics running in the Gateway will have nearly all of the functionality they have in the Runtime Client. All navigation controls are supported. All Data Bindings – including Two Way bindings – are supported and will continue to update with live data and accept property writes from the mimic. Trends are also supported – both real time and historical.

1.5.2 Supported UI Controls

XAML Documents that are intended for conversion to HTML must use a sub set of the UI controls available in the Mimic designer. These include:

1.5.2.1 Alarming

- Alarm Control
- Alarm LED
- Alarm Navigation List
- Condition Popup
- Heartbeat LED
- Historical Alarm Control

1.5.2.2 Buttons and Switches

- Flat Toggle Switch
- Gel Toggle Button
- Generic Toggle Button
- Industrial Toggle Button
- Power Toggle Button
- Toggle Switch

1.5.2.3 Charts

- Bar Chart
- Bubble Chart
- Mixed Chart
- Pie Chart
- Scatter Chart.
- Stacked Chart

1.5.2.4 Data Management

- Change Component Button
- Change Component Gel Button
- Collection Grid
- Create Object Button
- Create Object Gel Button
- Data Grid Array Control
- Function Text Block
- Historical Data Control
- Live Tile Control
- Property Grid
- Set Value Button

1.5.2.5 Date and Time

- Analog Clock
- Calendar Control
- Date & Time Display
- Date Picker Control

- Date Time Picker
- Elapsed Time Control
- Time Control

1.5.2.6 Documents

- Document Link
- Document Tree

1.5.2.7 Gauges

- Battery Level
- Compass
- Dual Gauge
- Dynamic Gauge
- High Low Bar
- Horizontal Linear Gauge
- Horizontal Slider
- Parabolic Gauge
- Racing Gauge
- Radial Gauge
- Signal Level
- Simple Gauge
- Temperature Control
- Variable Gauge
- Vertical Linear Gauge
- Vertical Slider

1.5.2.8 Grids and Panels

- Border
- Canvas
- Scroll Viewer
- Stack Panel
- View box
- Wrap Panel

1.5.2.9 Industrial

- Centrifugal Pump
- Connector Pipe
- Cutaway Tank
- Cylinder Tank
- Fitted Elbow Pipe
- Fitted Pipe
- Funnel Tank
- Gear
- Heavy Motor
- Helical Motor
- Indicator Tank
- Ladder Tank
- Metal Fly Wheel
- Mixing Tank
- Paddle Wheel
- Simple Fan
- Small Fly Wheel
- Sprocket
- Square Motor
- Standard Elbow Pipe
- Standard Pipe
- Tire
- Tower Indicator – Three State
- Tower Indicator – Five State
- Universal Pipe Connector
- Valve
- Vessel Tank
- Wind Generator

1.5.2.10 Input

- Data Text Box
- Password Box Control

1.5.2.11 Labels

- Data Label
- Generic Label
- Indicator Label

1.5.2.12 LED Displays

- LED Five By Seven Matrix
- LED Light
- LED Seven Segment
- LED Sixteen Segment
- Level Indicator – Three State
- Level Indicator – Five State

1.5.2.13 Media

- Bindable Image
- Hyperlink
- Web Browser

1.5.2.14 Microsoft

- Check Box
- Combo Box
- Expander
- Label
- Slider
- Tab Control
- Text Block
- Tree View

1.5.2.15 Navigation

- Gel Navigation Back Button
- Gel Navigation Button
- Navigation Back Button
- Navigation Border

- Navigation Button
- Navigation Combo
- Navigation List

1.5.2.16 Pick List

- Enumeration Control
- Radio Button Enumeration Control

1.5.2.17 Shapes

- Capped Line
- Ellipse
- Line
- Path
- Polygon
- Polygon Pencil
- Polyline
- Polyline Pencil
- Rectangle

1.5.2.18 Specialty

- Marquee
- Ratings Control
- Variance Label

1.5.2.19 Trend

- Trend Graph
- Graph Pen
- Pen BaseLine

1.5.3 Workspaces and Security

The Security and Workspace configuration of your Data Model is still applicable when running mimics in the Web Gateway. Your login activity is monitored by the Status Server. Workspaces that you are a member of will be applied to your web login. Properties that you don't have access to based on your workspace access will not update with data should you open a mimic that binds to those properties. Workspace access can also be defined for user controls.

1.5.4 Restrictions

As mentioned above, there are certain controls in the Mimic Designer toolbox that are not compatible with the Web Gateway. A list of compatible controls can be found above in Section 1.5.2.

Using the Gateway to run a mimic that contains unsupported controls will not prevent the mimic from functioning, but you will not see the unsupported controls on the screen. Instead, a generic placeholder will be displayed. All compatible controls will function as normal.

Alarming is not available in the Web Gateway, and Trends are available with limited support.

Mimics in the Gateway can be viewed in any browser that supports HTML5, including the current versions of the following browsers:

- Internet Explorer
- Chrome
- Opera
- Safari

All major mobile browsers have adopted the HTML5 standard, however the compatibility of some features has been reported to vary from one device to another. You may want to test your device's compatibility with HTML5 before using the Web Gateway. At the time of this writing, the following mobile browsers support HTML5:

- Safari iOS
- Android Browser
- Google Chrome
- Amazon Silk
- Blackberry Browser
- Nokia Browser
- Internet Explorer
- Opera Mobile
- Opera Mini

2 Customizing the Login for the Gateway

2.1 Setting up Automatic Log In

This section describes how to set up the Gateway to bypass the login screen using predefined credentials.

Requirements:

- In the Web.config file, the “Username” and “Password” variables must be set to the correct account credentials, and the “AutoLogin” setting needs to be set to a value of “YES”.

Example:

- Open Internet Information Services, select the “StatusEnterpriseGateway” application and then select the “Application Settings” feature.

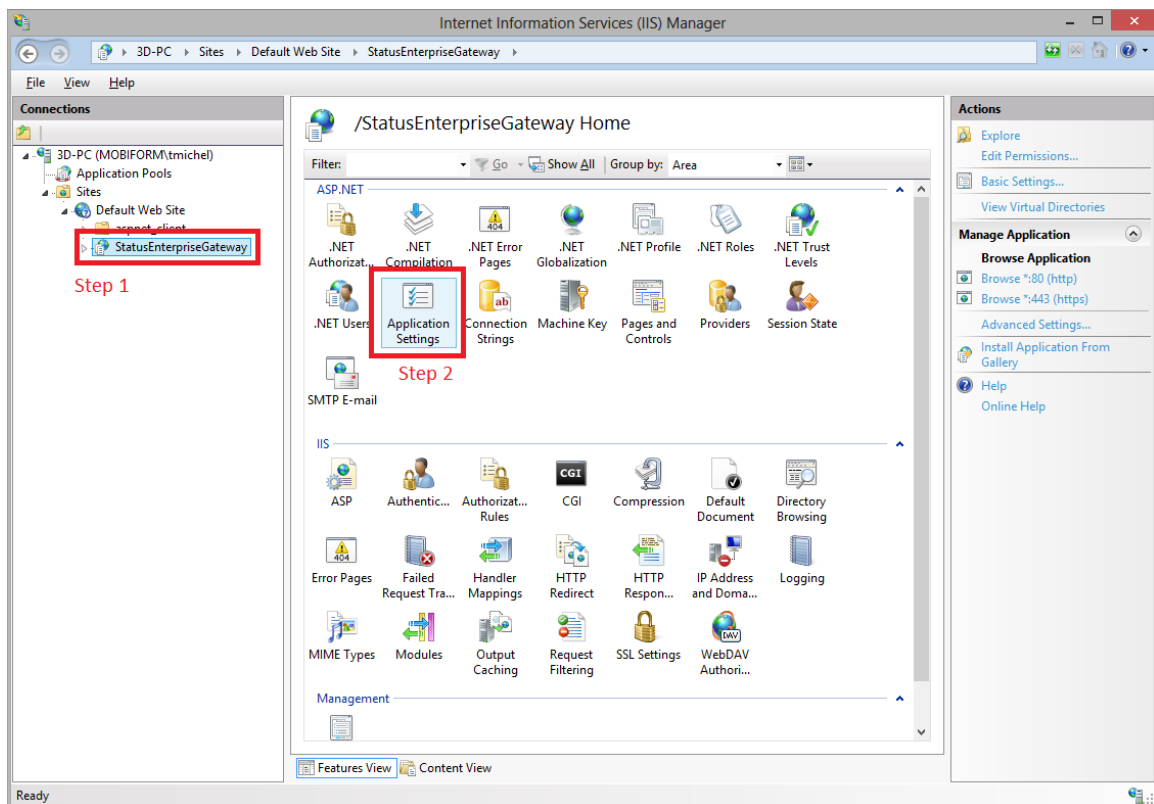


Figure 2 - IIS Configuration Screen

- Set “AutoLogin” to YES, and set the “Username” and “Password” to the preferred credentials.

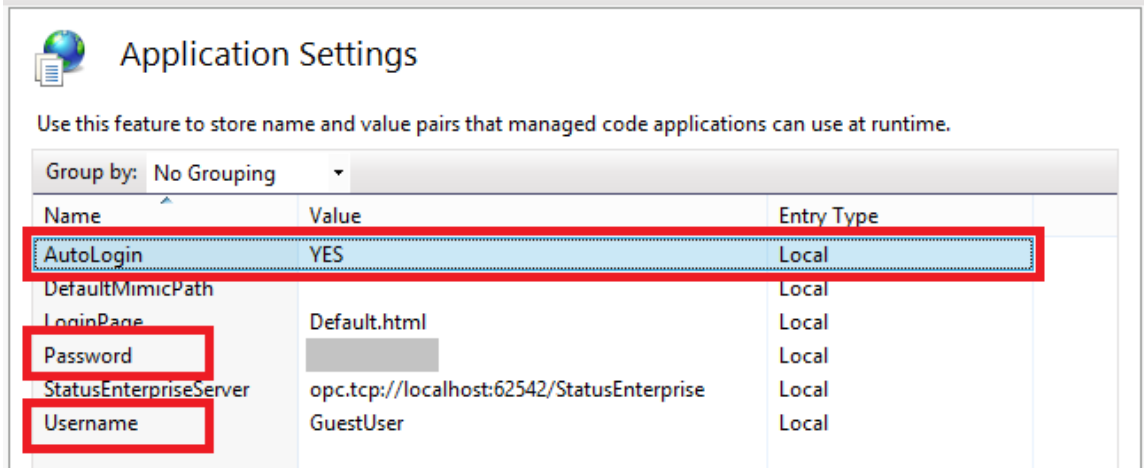


Figure 3 - Application Settings Screen

- Now when you open the Gateway, it will automatically log in the defined user.

2.2 Setting a Default Mimic Path

This section describes how to set the default mimic displayed to the user when they log in instead of the model navigation page.

Requirements:

- The DefaultMimicPath variable in the Web.config file must be set to a valid path.

Example:

- Open Internet Information Services, select the "StatusEnterpriseGateway" application and then select the "Application Settings" feature.

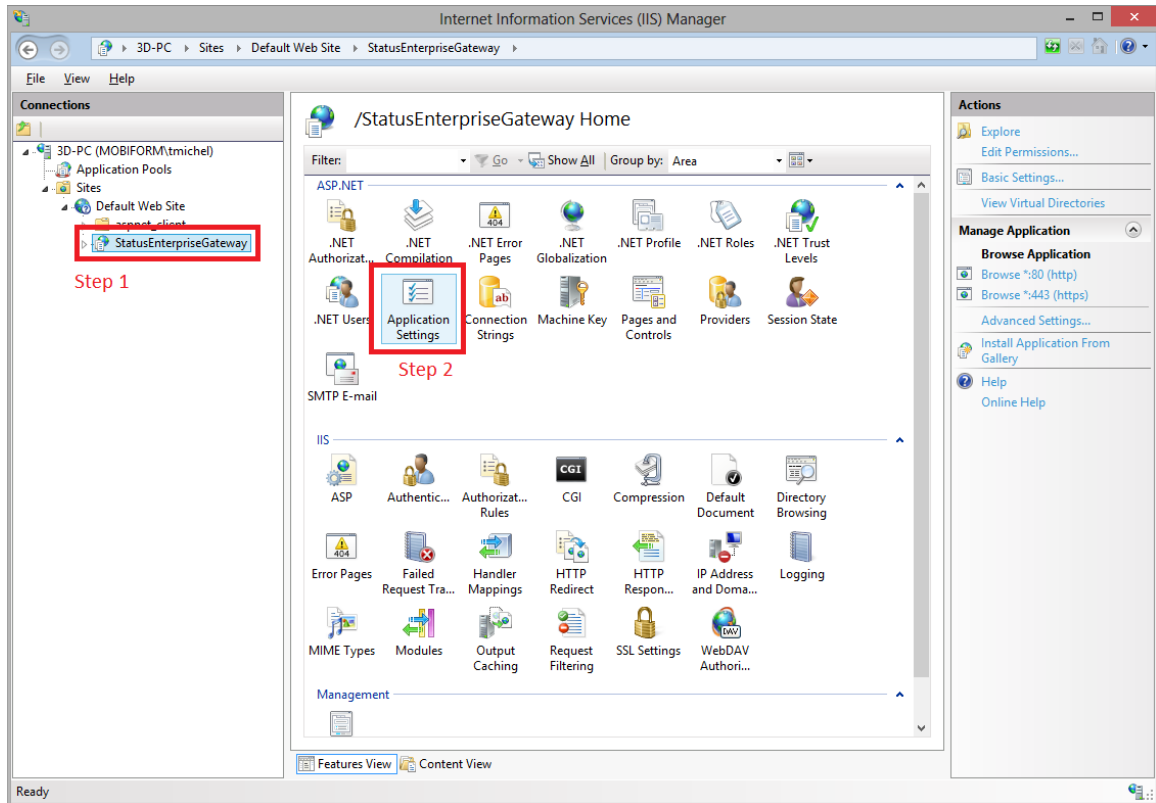


Figure 4 - IIS Manager

- Set "DefaultMimicPath" to a valid browse path for the mimic you would like to display.

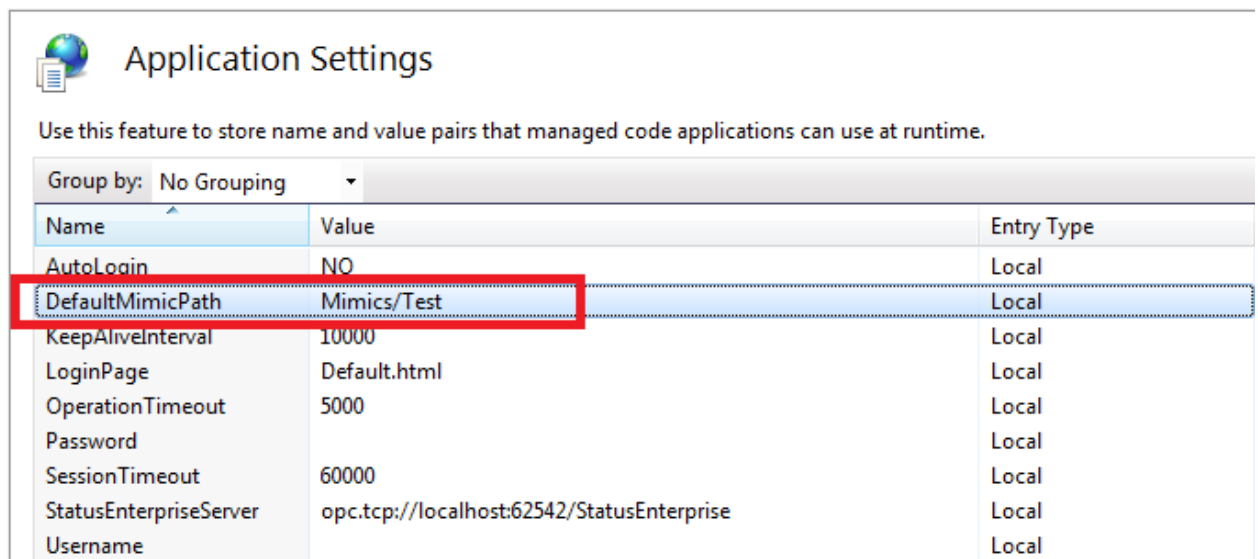


Figure 5 - Default Mimic Path

- Now when you log into the Gateway, this mimic will display instead of the navigation system.

2.3 Recommended Navigation Method

Status Device Cloud Gateway includes a navigation menu. The menu is presented after logon. This menu will provide quick access to all of the Mimics, and Assets, but this is not the preferred or recommended method of navigation. Best practice would be to create a Home Page with navigation button to all other needed Mimics.

2.4 Changing the Gateway Server

This section defines how to specify which Status Server the Gateway is Connecting to.

Requirements:

- The StatusEnterpriseServer variable in the Web.config file must be set to a server path.

Example:

- Open Internet Information Services, select the “StatusEnterpriseGateway” application and then select the “Application Settings” feature.

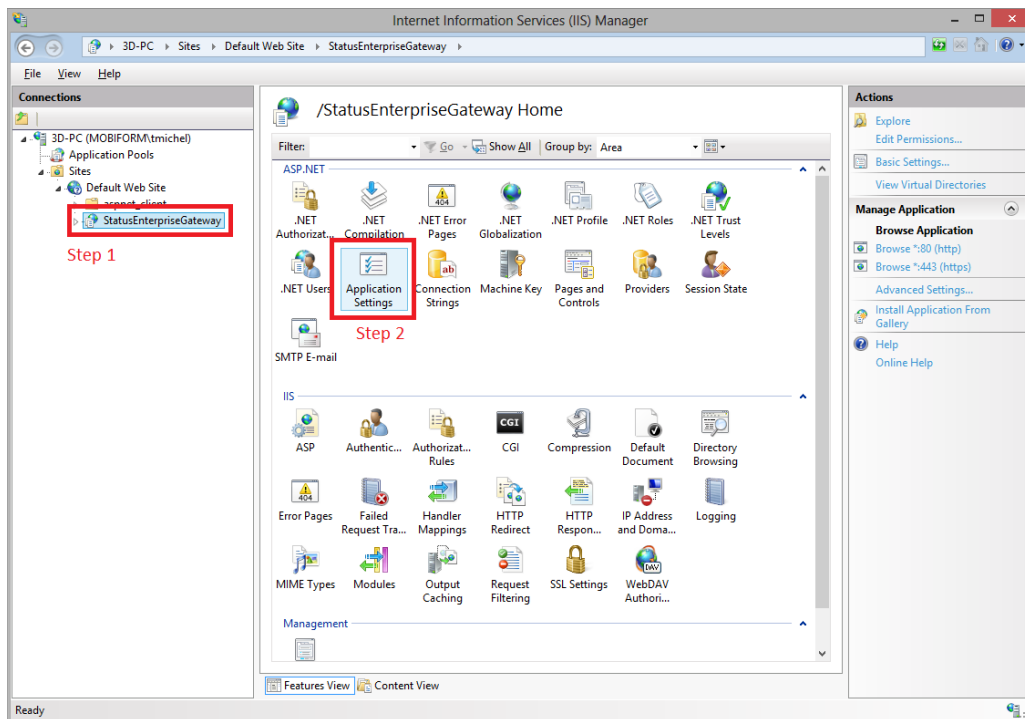


Figure 6 - Application Settings

- Set “StatusEnterpriseServer” to a valid path for a Status Device Cloud Server.

Application Settings

Use this feature to store name and value pairs that managed code applications can use at runtime.

Group by: No Grouping

Name	Value	Entry Type
AutoLogin	NO	Local
DefaultMimicPath		Local
KeepAliveInterval	10000	Local
LoginPage	Default.html	Local
OperationTimeout	5000	Local
Password		Local
SessionTimeout	60000	Local
StatusEnterpriseServer	opc.tcp://192.168.168.108:62542/StatusEnterprise	Local
Username		Local

Figure 7 - Gateway Path

- Now the gateway will connect to a server elsewhere instead of on your local machine.

2.5 Creating a Custom Login Page

Requirements:

- You must have a form with an action of "UserLogin.aspx" and a method of "post".
- Three inputs are required in the form, a username text input, a password text input, and a submit button.
- The username text input needs a name of "username" and the password text input needs a name of "password".
- The third input should be of type "submit".
- To display errors when the user login fails, add a reference to the query_string.js script file and call loadPageVar("error"). This gives you the error returned by the server when the login fails. This can be stored in a variable and displayed on the page in whatever manner necessary.

Examples:

- Form with Inputs –
HTML

```
<form action="UserLogin.aspx" method="post">
  <input type="text" name="username" placeholder="Username" />
  <input type="password" name="password" placeholder="Password" />
  <input type="submit" title="Log In" />
</form>
```

- Storing the error as a variable and displaying it in the page –
Javascript w/ jQuery library referenced and used

```
$(function()
{
  var error = loadPageVar("error");
  if (error != undefined && error != null && error != "")
  {
    var errorBar = $('#errorBar');
    errorBar.text(error);
    errorBar.css('display', 'block');
  }
});
```

- Enabling SSL Encryption
Javascript w/ jQuery library referenced and used

```
$(function()
{
  if (location.href.substring(0,5) != "https")
  {
    Location.href = "https" + location.href.substring(4);
  }
});
```

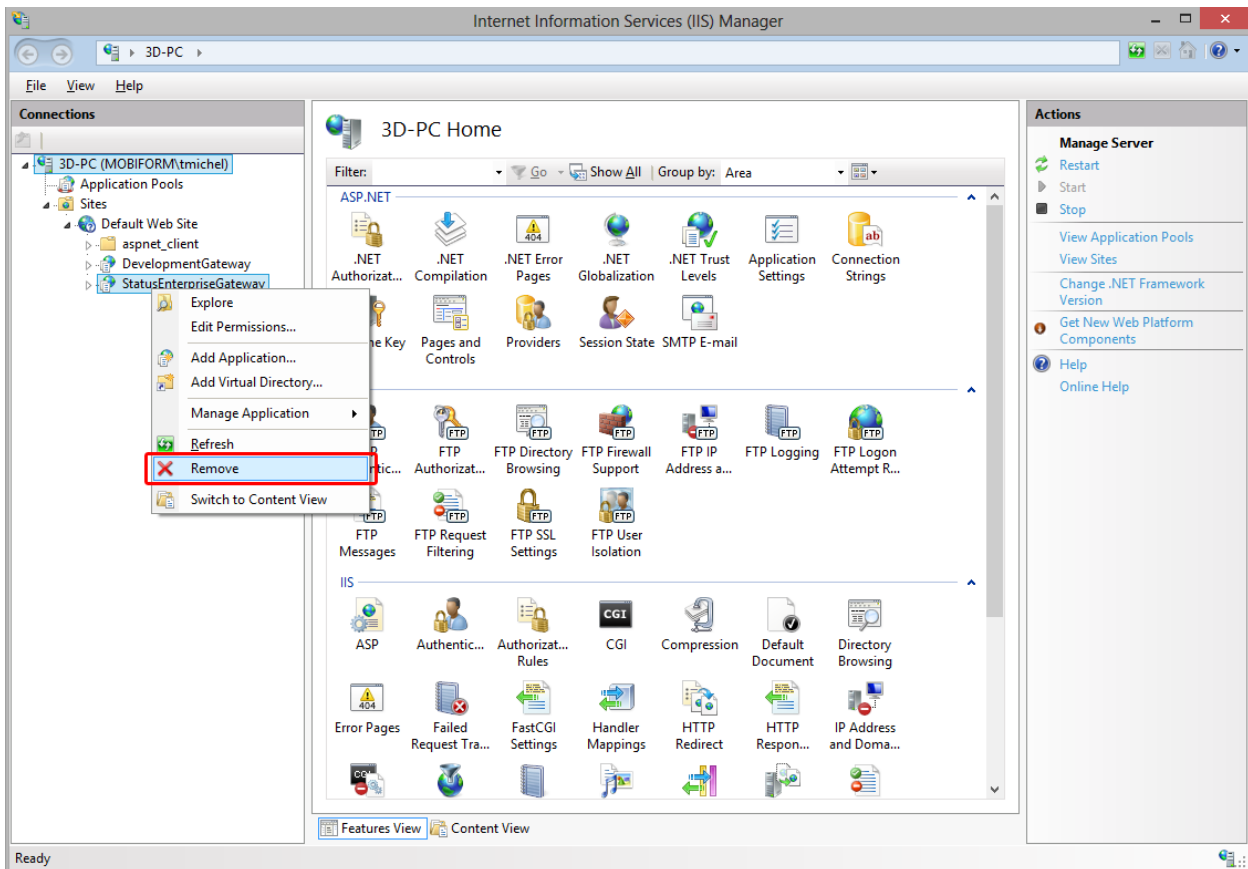
NOTE: If you create a custom login page, you will need to ensure that the Gateway is pointing to the page you've created by updating the "LoginPage" parameter located in the "Application Settings" in IIS.

2.6 Moving the Gateway to another IIS Website

It's possible that you may want to move the Gateway into another IIS site. This section walks you through the steps that need to be taken in order to successfully move the Gateway into another IIS site.

2.6.1 Remove the Gateway from the current IIS Website

To remove the Gateway from IIS, navigate to the site that the Gateway is setup in. If available, the Gateway will be installed to the Default Website. If unavailable, the Gateway will create a new site named StatusEnterpriseGateway. Expand the site, then find the IIS application named "StatusEnterpriseGateway" and right click to bring up the context menu and select "Remove" as shown in the image below:



2.6.2 Adding the Gateway into a new IIS Website

To add the Gateway back into IIS, first select your IIS Website that you want to add the Gateway to, then create a new "Virtual Directory" that points to the Gateway folder in your Status Device Cloud installation directory (C:\Program Files\B-Scada\StatusEnterprise\Gateway by default). Next, convert the Virtual Directory to an Application by using the right click context menu, ensuring that you use Status Device Cloud Gateway as the Application Pool. The Status Device Cloud Gateway requires that the Application Pool being used is running ASP.NET 4.0 using an Integrated Pipeline as well as running as a Network Service.

Once completed, verify that the Gateway loads properly by visiting the URL associated with the site you've added the Gateway to.

NOTE: If you move the Gateway to another site, you must ensure that Index.aspx is added as a default document.

3 REST API

The Status Device Cloud REST API provides software developers the ability to access Status Server programmatically through simple HTTP POST or GET commands.

3.1 Activating the Service

The REST API is automatically installed alongside the Gateway, but is not active by default. To activate it, open Internet Information Services (IIS) Manager and navigate to the StatusEnterpriseGateway site. Expand the site, then expand the API folder and right-click the REST folder.

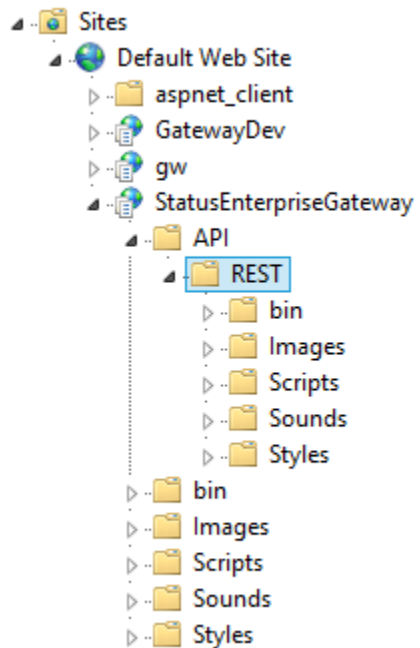


Figure 8 - IIS Manager

Click the **Convert to Application** option then OK in the dialog window to enable the service at the default endpoint of StatusEnterpriseGateway/api/rest/.

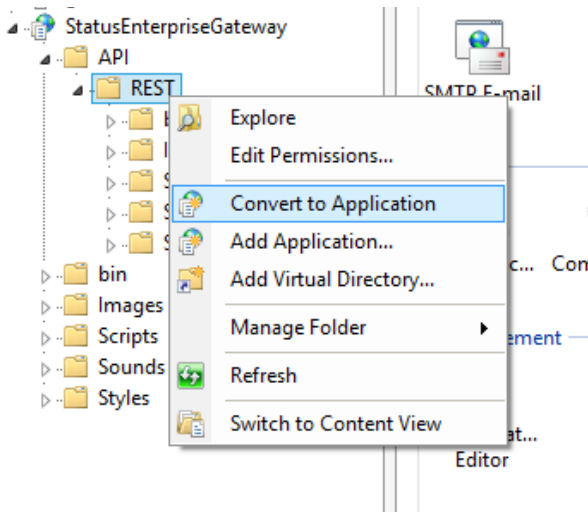


Figure 9 - Convert to Application Option

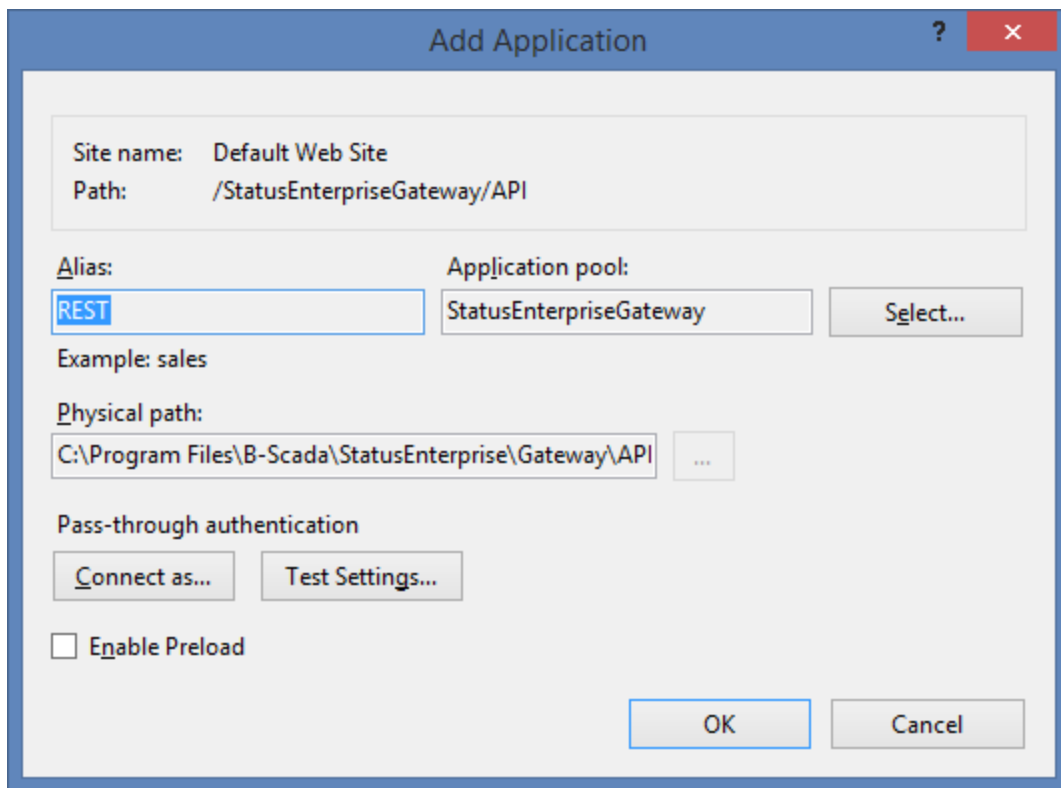


Figure 10 - Convert to Application Dialog

3.2 Available Methods

The following methods are available to perform various tasks in your Status Device Cloud Server installation such as managing assets and updating property values.

3.2.1 Login

This method needs to be called before any others as it generates the token that needs to be passed to the rest of the methods.

Parameters:

Username, Password, KeepAlive, SessionTimeout, OperationTimeout, SubscriptionLifetime

Example:

```
http://your.gateway.com/api/rest/login?username=user&password=pass&keep
alive=60000&sessiontimeout=30000&operationtimeout=5000&subscriptionlife
time=60000
```

Response:

```
<?xml version="1.0" encoding="utf-8"?>
<IoTAPI>
  <Token>1747350660</Token>
  <Result>0</Result>
  <ResultMessage>Success</ResultMessage>
  <Method>Login</Method>
</IoTAPI>
```

3.2.2 Logout

Should be called when no more operations are necessary.

Parameter:

Token

Example:

```
http://your.gateway.com/api/rest/logout?token=1747350660
```

Response:

```
<?xml version="1.0" encoding="utf-8"?>
<IoTAPI>
  <Result>0</Result>
  <ResultMessage>Success</ResultMessage>
  <Method>Logout</Method>
</IoTAPI>
```

3.2.3 CheckSession

This method can be called to determine if the supplied token is still valid, i.e. the connection to the server is still active for a user.

Parameter:

Token

Example:

```
http://your.gateway.com/api/rest/checksession?token=1747350660
```

Response:

```
<?xml version="1.0" encoding="utf-8"?>
<IoTAPI>
  <Result>0</Result>
  <ResultMessage>Success</ResultMessage>
```

```

    <Method>CheckSession</Method>
  </IoTAPI>

```

3.2.4 TranslateBrowsePathToNodeid

This method will need to be called prior to invoking any of the methods that follow unless the Nodeid is already known.

Parameters:

Token, browsePath

Example:

```

http://your.gateway.com/api/rest/translatebrowsepathtonodeid?token=1747350660&browsepath=Types/ObjectTypes/BaseObjectType/StatusObjectType/VirtualVille

```

Response:

```

<?xml version="1.0" encoding="utf-8"?>
<IoTAPI>
  <NodeId>ns=3;s=1AA98C744B0E47238E424233376EC661</NodeId>
  <Result>0</Result>
  <ResultMessage>Success</ResultMessage>
  <Method>TranslateBrowsePathToNodeid</Method>
</IoTAPI>

```

3.2.5 GetAttributes

This method will retrieve the attributes for a given Nodeid.

Parameters:

Token, Nodeid

Example:

```

http://your.gateway.com/api/rest/getattributes?token=1747350660&nodeid=ns=3;s=8FAF814F113A491E8E845656BF0F56FE

```

Response:

```

<?xml version="1.0" encoding="utf-8"?>
<IoTAPI>
  <BrowseName>Virtualville</BrowseName>
  <DisplayName>Virtualville</DisplayName>
  <Description></Description>
  <Datatype>ns=3;s=F0650FE864A14B909BE50DD2BA661004</Datatype>
  <FullPath>Objects/Assets/Virtualville</FullPath>
  <Result>0</Result>
  <ResultMessage>Success</ResultMessage>
  <Method>GetAttributes</Method>
</IoTAPI>

```

3.2.6 GetValue

Gets the value of the property indicated by the supplied Nodeid.

Parameters:

Token, Nodeid

Example:

```
http://your.gateway.com/api/rest/getvalue?token=1747350660&nodeid=ns=3;s=31B18D3D24B94D78A316642D4127EB0C
```

Response:

```
<?xml version="1.0" encoding="utf-8"?>
<IoTAPI>
  <Value>19964</Value>
  <Result>0</Result>
  <ResultMessage>Success</ResultMessage>
  <Method>GetValue</Method>
</IoTAPI>
```

3.2.7 SetValue

Sets the value of the property indicated by the supplied NodeId.

Parameters:

Token, NodeId, Value

Example:

```
http://your.gateway.com/api/rest/setvalue?token=1747350660&nodeid=ns=3;s=31B18D3D24B94D78A316642D4127EB0C&value=19965
```

Response:

```
<?xml version="1.0" encoding="utf-8"?>
<IoTAPI>
  <Result>0</Result>
  <ResultMessage>Success</ResultMessage>
  <Method>SetValue</Method>
</IoTAPI>
```

3.2.8 GetType

Retrieves the Type's NodeId of the given Asset.

Parameters:

Token, NodeId

Example:

```
http://your.gateway.com/api/rest/gettype?token=1747350660&nodeid=ns=3;s=31B18D3D24B94D78A316642D4127EB0C
```

Response:

```
<?xml version="1.0" encoding="utf-8"?>
<IoTAPI>
  <TypeId>ns=2;i=2006</TypeId>
  <Result>0</Result>
  <ResultMessage>Success</ResultMessage>
  <Method>GetType</Method>
</IoTAPI>
```

3.2.9 CreateAsset

Creates an instance of an asset in the model with the given information.

Parameters:

Token, TypeId, OwnerId, BrowseName, DisplayName

Example:

```
http://your.gateway.com/api/rest/createasset?token=1747350660&typeid=ns=3;s=1AA98C744B0E47238E424233376EC661&ownerid=ns=2;i=3002&browsename=Virtualville2&displayname=Virtualville2
```

Response:

```
<?xml version="1.0" encoding="utf-8"?>
<IoTAPI>
  <NodeId>ns=3;s=A341A15388024B6FA8ED9ADF400E4FBD</NodeId>
  <Result>0</Result>
  <ResultMessage>Success</ResultMessage>
  <Method>CreateAsset</Method>
</IoTAPI>
```

3.2.10 DeleteAsset

Deletes an asset from the model.

Parameters:

Token, NodeId, OwnerId

Example:

```
http://your.gateway.com/api/rest/deleteasset?token=1747350660&nodeid=ns=3;s=1AA98C744B0E47238E424233376EC661&ownerid=ns=2;i=3002
```

Response:

```
<?xml version="1.0" encoding="utf-8"?>
<IoTAPI>
  <Result>0</Result>
  <ResultMessage>Success</ResultMessage>
  <Method>DeleteAsset</Method>
</IoTAPI>
```

3.2.11 GetNodeIdsProperties

Retrieves all the properties of the given asset. It is especially useful for getting the NodeId of a property for use in the GetValue/SetValue methods.

Parameters:

Token, NodeId

Example:

```
http://your.gateway.com/api/rest/getnodeidsproperties?token=1747350660&nodeid=ns=3;s=1AA98C744B0E47238E424233376EC661&ownerid=ns=2;i=3002
```

Response:

```
<?xml version="1.0" encoding="utf-8"?>
<IoTAPI>
  <NodeIdType>Asset:Virtualville,Properties ([Property name:Air Quality Index,NodeId :ns=3;s=27B8A0A3D7864F548DEBC8FBE40A0440],[Property
```

```

name:ComfortIndex,NodeId
:ns=3;s=F35083247E99464496A3D9F190382CC0],[Property
name:EnergyAdjustmentFactor,NodeId
:ns=3;s=060459E941264493887CD7A4304E9D98],[Property
name:EnergyCostDiff,NodeId
:ns=3;s=F14D6547B6D14D929D9123D6F2104133],[Property
name:EnergyCostSqFt,NodeId
:ns=3;s=5C1F09F77ED0419ABDC2878B28485ABE],[Property
name:EnergyIndex,NodeId
:ns=3;s=6C3F9479FC5241A49BB50EA9A0FF74A6],[Property
name:EnergyTargetSqFt,NodeId
:ns=3;s=D84B8F97959648058DA3ABEA822EBCD8],[Property
name:FairAirVisibility,NodeId
:ns=3;s=2A3D078D71A04975A8B4FD3356CCB585],[Property
name:GoodAirVisibility,NodeId
:ns=3;s=79BB12EF958843D79C75D594DB88C1BA],[Property
name:Humidity,NodeId
:ns=3;s=812C60E8CB6547B8890BC5B2E0D3826F],[Property
name:IAQIndex,NodeId
:ns=3;s=FDD0318795C24E1CAA8FAD0C53FCD725],[Property name:kWhRate,NodeId
:ns=3;s=757388F074F246598FFAE7F31C237213],[Property
name:MaintenanceCostDiff,NodeId
:ns=3;s=F3EFAE7458E54B859C4CEFCB3D0BB37E],[Property
name:MaintenanceCostSqFt,NodeId
:ns=3;s=E93D265BF77D4CFF86338FC924A31606],[Property
name:MaintenanceTargetSqFt,NodeId
:ns=3;s=7995CE0C4D3848B58B57A92240C70D5C],[Property
name:NoiseLevel,NodeId
:ns=3;s=41C5BA6AD52E4580A17A2F4D2BE6F925],[Property
name:Occupants,NodeId
:ns=3;s=31B18D3D24B94D78A316642D4127EB0C],[Property name:Rate of Power
Consumption,NodeId :ns=3;s=9B39B92AADAD4E7A995BBB6C0AA0C524],[Property
name:Rate of Water Consumption,NodeId
:ns=3;s=CB4FAF255BE646228C420E5C7779E496],[Property
name:SquareFootage,NodeId
:ns=3;s=061CAAB9337743429727E6F994A50963],[Property
name:TargetCostSqFt,NodeId
:ns=3;s=482BBB3EE83F437D9C8564BD7C4A11B4],[Property
name:Temperature,NodeId
:ns=3;s=D27A5D562D6F4CB8BC89A9E638D9AB43],[Property
name:TotalCostSqFt,NodeId
:ns=3;s=512A952AECC74161AE81BFB688D4853B],[Property
name:TotalTax,NodeId
:ns=3;s=DF09A8C404E943C7B2E2DFE3DA689F80)]</NodeIdType>
  <Result>0</Result>
  <ResultMessage>Success</ResultMessage>
  <Method>GetNodeIdsProperties</Method>
</IoTAPI>

```

4 Troubleshooting

Error Code:

Server Error in '/StatusEnterpriseGateway' Application.

Access to the registry key 'Global' is denied.

Description: An unhandled exception occurred during the execution of the current web request. Please review the stack trace for more information about the error and where it originated in the code.

Exception Details: System.UnauthorizedAccessException: Access to the registry key 'Global' is denied.

ASP.NET is not authorized to access the requested resource. Consider granting access rights to the resource to the ASP.NET request identity. ASP.NET has a base process identity (typically {MACHINE}\ASPNET on IIS 5 or Network Service on IIS 6 and IIS 7, and the configured application pool identity on IIS 7.5) that is used if the application is not impersonating. If the application is impersonating via <identity impersonate="true"/>, the identity will be the anonymous user (typically IUSR_MACHINENAME) or the authenticated request user.

To grant ASP.NET access to a file, right-click the file in File Explorer, choose "Properties" and select the Security tab. Click "Add" to add the appropriate user or group. Highlight the ASP.NET account, and check the boxes for the desired access.

Source Error:

```
Line 96:             socketServer.NewMessageReceived += new
SessionEventHandler<WebSocketSession, string>(socketServer_NewMessageReceived);
Line 97:
Line 98:             if (!SocketServerManager.Start())
Line 99:             {
Line 100:                 SocketServerManager.Stop();
```

Source File: c:\Status Device Cloud 2011 Q4\Gateway\Status.Enterprise.Gateway\Global.aspx.cs **Line:** 98

Stack Trace:

```
[UnauthorizedAccessException: Access to the registry key 'Global' is denied.]
  Microsoft.Win32.RegistryKey.Win32Error(Int32 errorCode, String str) +6504925
  Microsoft.Win32.RegistryKey.InternalGetValue(String name, Object defaultValue,
Boolean doNotExpand, Boolean checkSecurity) +361
  Microsoft.Win32.RegistryKey.GetValue(String name) +62
  System.Diagnostics.PerformanceMonitor.GetData(String item) +148
  System.Diagnostics.PerformanceCounterLib.GetPerformanceData(String item) +255
  System.Diagnostics.PerformanceCounterLib.get_CategoryTable() +172
```

```

System.Diagnostics.PerformanceCounterLib.CounterExists(String category, String
counter, Boolean& categoryExists) +35
System.Diagnostics.PerformanceCounterLib.CounterExists(String machine, String
category, String counter) +95
System.Diagnostics.PerformanceCounter.InitializeImpl() +392
System.Diagnostics.PerformanceCounter..ctor(String categoryName, String
counterName, String instanceName, Boolean readOnly) +373
System.Diagnostics.PerformanceCounter..ctor(String categoryName, String
counterName, String instanceName) +14
SuperSocket.SocketEngine.SocketServerManager.StartPerformanceLog() +252
SuperSocket.SocketEngine.SocketServerManager.Start() +498
Status.Enterprise.Gateway.Global.StartSuperWebSocketByConfig() in c:\Status
Device Cloud 2011 Q4\Gateway\Status.Enterprise.Gateway\Global.asax.cs:98

```

```

[HttpException (0x80004005): Access to the registry key 'Global' is denied.]
System.Web.HttpApplicationFactory.EnsureAppStartCalledForIntegratedMode(HttpConte
xt context, HttpApplication app) +12864673
System.Web.HttpApplication.RegisterEventSubscriptionsWithIIS(IntPtr appContext,
HttpContext context, MethodInfo[] handlers) +175
System.Web.HttpApplication.InitSpecial(HttpApplicationState state, MethodInfo[]
handlers, IntPtr appContext, HttpContext context) +304
System.Web.HttpApplicationFactory.GetSpecialApplicationInstance(IntPtr
appContext, HttpContext context) +404
System.Web.Hosting.PipelineRuntime.InitializeApplication(IntPtr appContext) +475

[HttpException (0x80004005): Access to the registry key 'Global' is denied.]
System.Web.HttpRuntime.FirstRequestInit(HttpContext context) +12881540
System.Web.HttpRuntime.EnsureFirstRequestInit(HttpContext context) +159
System.Web.HttpRuntime.ProcessRequestNotificationPrivate(IIS7WorkerRequest wr,
HttpContext context) +12722601

```

Solution:

1. Open Information Internet Services(IIS) Manager.
2. Expand the local machine in the left Connections panel.
3. Click on 'Application Pools' in the Connections panel.
4. Right Click on StatusEnterpriseGateway in the Application Pools panel and select 'Advanced Settings...'
5. Under the heading 'Process Model' select 'Identity'.
6. Click the '...' button on the right to see the available values.
7. Select Built-in account:
NetworkService

4.1 Truncated Streams

Turning on WCF data logging can result in lost data. See 'Enabling WCF logging causes response corruption in webHttp service that returns a stream': <http://social.msdn.microsoft.com/Forums/en-US/wcf/thread/2bc5e5a6-1971-46cb-8b4f-f1c46130faca>