



User Guide Part 14

REST API

Contents

- 1 OVERVIEW..... 3**
- 1.1 Activating the Service 3
- 1.2 Available Methods..... 5
 - 1.2.1 Login 5
 - 1.2.2 Logout 5
 - 1.2.3 CheckSession 5
 - 1.2.4 TranslateBrowsePathToNodeId 6
 - 1.2.5 GetAttributes 6
 - 1.2.6 GetValue 7
 - 1.2.7 SetValue 7
 - 1.2.8 GetType 7
 - 1.2.9 CreateAsset 8
 - 1.2.10 DeleteAsset..... 8
 - 1.2.11 GetNodeIdsProperties 8

Figures and Tables

Figure 1 - IIS Manager	3
Figure 2 - Convert to Application Option	4
Figure 3 - Convert to Application Dialog	4

1 Overview

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

1.1 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.

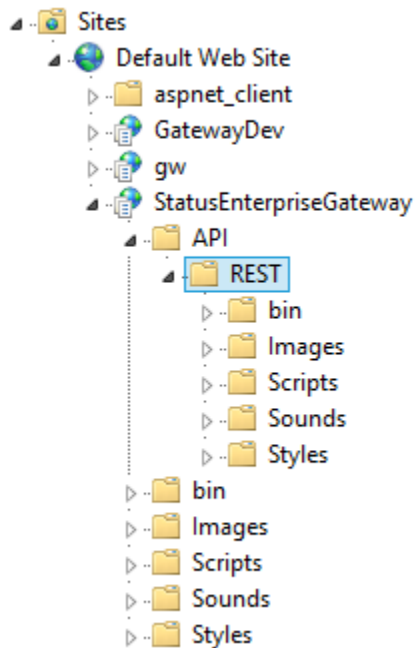


Figure 1 - IIS Manager

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

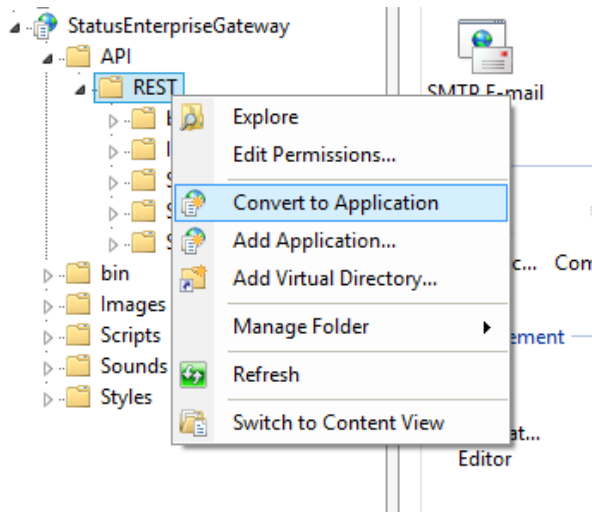


Figure 2 - Convert to Application Option

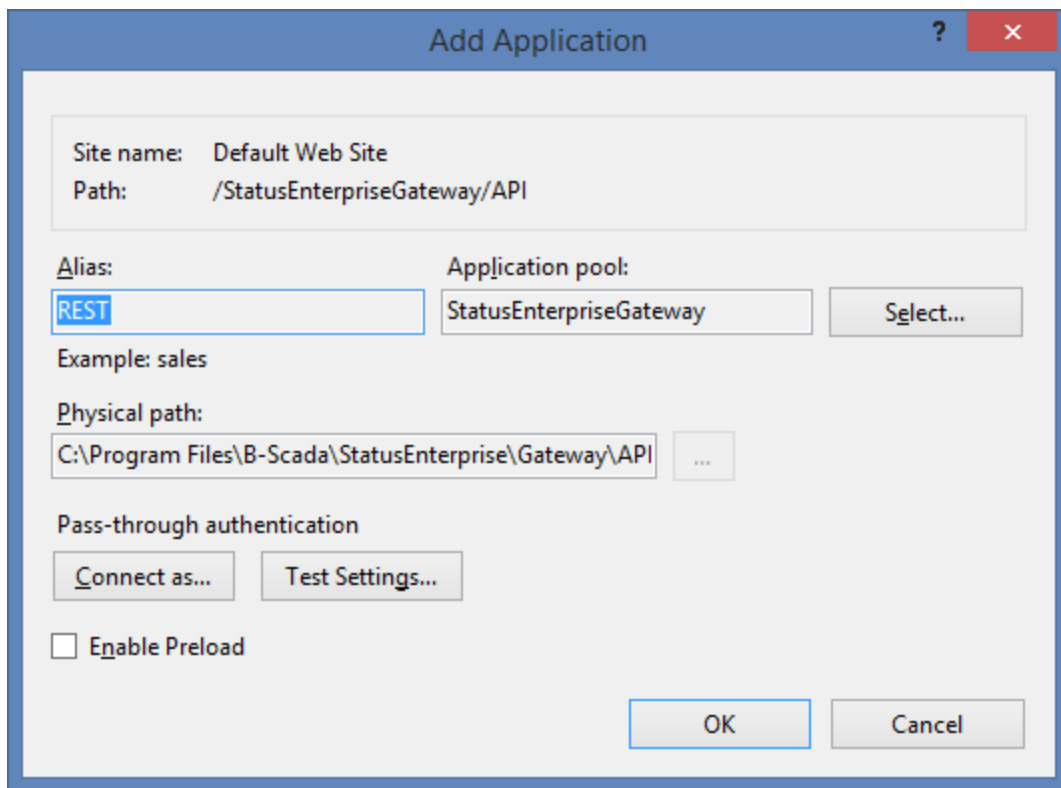


Figure 3 - Convert to Application Dialog

1.2 Available Methods

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

1.2.1 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/login?username=user&password=pass&keepalive=60000&sessiontimeout=30000&operationtimeout=5000&subscriptionlifetime=60000
```

Response:

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

1.2.2 Logout

Should be called when no more operations are necessary.

Parameter:

Token

Example:

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

Response:

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

1.2.3 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/checksession?token=1747350660
```

Response:

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

1.2.4 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/translatebrowsepathtonodeid?token=1747350660&browsepath=Types/ObjectTypes/BaseObjectType/StatusObjectType/VirtualV
ille
```

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>
```

1.2.5 GetAttributes

This method will retrieve the attributes for a given NodeId.

Parameters:

Token, NodeId

Example:

```
http://your.gateway.com/api/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>
```

1.2.6 GetValue

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

Parameters:

Token, NodeId

Example:

```
http://your.gateway.com/api/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>
```

1.2.7 SetValue

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

Parameters:

Token, NodeId, Value

Example:

```
http://your.gateway.com/api/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>
```

1.2.8 GetType

Retrieves the Type's NodeId of the given Asset.

Parameters:

Token, NodeId

Example:

```
http://your.gateway.com/api/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>
```

1.2.9 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/createasset?token=1747350660&typeid=ns=3;s=
1AA98C744B0E47238E424233376EC661&ownerid=ns=2;i=3002&browsename=Virtual
ville2&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>
```

1.2.10 DeleteAsset

Deletes an asset from the model.

Parameters:

Token, NodeId, OwnerId

Example:

```
http://your.gateway.com/api/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>
```

1.2.11 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/getnodeidsproperties?token=1747350660&nodei
d=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>
```