

Provisioning Integration With Unity Licencing Portal

The Kakapo Systems licensing API is a .NET Web Service that allows all actions relating to management of enterprises and service providers [collectively identified as “resellers” within the Kakapo portal], groups and users, as well as license assignment.

The URL for the web service is <https://portal.unityclient.com/KakapoProvision/Provision.asmx>

You will see the various web methods available through the API, as shown below

Provision

The following operations are supported. For a formal definition, please review the [Service Description](#).

- **AddGroup**
Add a group to a reseller. The system provider and reseller must already exist in the system.
- **AddGroupLicenseSubPurchase**
Add a sub-purchase to a group. This depends on sufficient licenses being available in the provider and reseller license pool, as well as reseller license oversell permissions. Application names are UnityDesktop, UnityAgent, UnitySupervisor, UnityWallboard and UnityReception. License Types are Lite, Pro, Standard and Enterprise.
- **AddGroupLogo**
Assign a logo to the group, which overrides the current logo. The size of the logo cannot be more than 200 pixels in width and 70 pixels in height, and must be either a bmp, png, jpg/jpeg or gif file. If specifying the URL of a logo [in the cloud], it will be downloaded locally before being saved, in order to ensure the file meets these logo requirements.
- **AddReseller**
Add a reseller to a system provider. The system provider must already exist in the system.
- **AddResellerLicenseSubPurchase**
Add a sub-purchase to a reseller. This depends on sufficient licenses being available in the provider license pool. Application names are UnityDesktop, UnityAgent, UnitySupervisor, UnityWallboard and UnityReception. License Types are Lite, Pro, Standard and Enterprise.
- **AddResellerLogo**
Assign a logo to the reseller, which overrides the current logo. The size of the logo cannot be more than 200 pixels in width and 70 pixels in height, and must be either a bmp, jpg/jpeg or gif file. The logo will be downloaded locally before being saved, in order to ensure the file meets these logo requirements.
- **AddSystemProvider**
Create a child system provider which can be managed through the portal by the parent provider.
- **AddSystemProviderLicenseSubPurchase**
Add a sub-purchase to a child system provider. This depends on sufficient licenses being available in the parent provider license pool. Application names are UnityDesktop, UnityAgent, UnitySupervisor, UnityWallboard and UnityReception. License Types are Lite, Pro, Standard and Enterprise.
- **AddSystemProviderLogo**
Assign a logo to the provider, which overrides the current logo. The size of the logo cannot be more than 200 pixels in width and 70 pixels in height, and must be either a bmp, jpg/jpeg, png or gif file. The logo will be downloaded locally before being saved, in order to ensure the file meets these logo requirements.
- **AddUser**
Add a user to a group. The system provider, reseller and group must already exist in the system.
- **AddUserLicense**
Assign a license to a user. The system provider, reseller, group and user must already exist in the system. Temporary licenses cannot be explicitly assigned and the system provider must have the required licenses available. If the license is already assigned it will not be overridden but any other permanent application currently assigned to the user will be unassigned. This suits an upgrade or downgrade scenario. Application names are UnityDesktop, UnityAgent, UnitySupervisor and UnityReception. License Types are Lite, Pro, Standard and Enterprise. Use the web method GetUserLicenses to get current license assignments.
- **AddUserLicenseWithTrial**
Assign a license to a user, and also assign a trial license for the same Unity product. The trial license will be used while it is valid, after which the assigned license will be used. This provides an upsell opportunity whereby the end user starts with all functionality, then reverts to the functionality offered by the trial license. The system provider, reseller, group and user must already exist in the system. The system provider must have the required licenses available (not applicable for trial licenses). If the license is already assigned it will not be overridden but any other permanent licenses for the same application currently assigned to the user will be unassigned. If the trial license has already been assigned it will not be assigned again or reset. Application names are UnityDesktop, UnityAgent, UnitySupervisor and UnityReception. License Types are Lite, Pro, Standard and Enterprise. Use the web method GetUserLicenses to get current license assignments.
- **AddUserWithHierarchy**
Add a user to the system. The reseller and group do not have to exist in the system as anything that does not yet exist will be created within the system provider.
- **DeleteGroup**
Delete a group. This also deletes all users within the group.

Each web method provides a description and when you click into the web method it will provide details of all required parameters as well as expected outputs, as below

Provision

Click [here](#) for a complete list of operations.

AddSystemProvider

Create a child system provider which can be managed through the portal by the parent provider.

Test

To test the operation using the HTTP POST protocol, click the 'Invoke' button.

Parameter	Value
loginUserName:	<input type="text"/>
loginPassword:	<input type="text"/>
newSystemProviderServerHostName:	<input type="text"/>
newSystemProviderName:	<input type="text"/>

SOAP 1.1

The following is a sample SOAP 1.1 request and response. The **placeholders** shown need to be replaced with actual values.

```
POST /KakapoProvision/Provision.asmx HTTP/1.1
Host: portal.unityclient.com
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://kakaposystems.com/AddSystemProvider"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <AddSystemProvider xmlns="http://kakaposystems.com">
      <loginUserName>string</loginUserName>
      <loginPassword>string</loginPassword>
      <newSystemProviderServerHostName>string</newSystemProviderServerHostName>
      <newSystemProviderName>string</newSystemProviderName>
    </AddSystemProvider>
  </soap:Body>
</soap:Envelope>

HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length
```

Authentication details must be provided with each web method call, please note that these are not the same login details that you use to log into the Kakapo portal. We will create a special "Interface User" login which must be provided when calling all web methods in the web service. Please send an email to ineedhelp@kakaposystems.com if you would like an Interface Only login.

Whenever using the web methods you must make sure that the input parameters match Broadworks, for example in the below screenshot for the AddUserWithHierarchy web method the following values are required

loginUsername/loginPassword: These will be provided by Kakapo Systems on request and will be the same for all web methods

serverHostName: This is the server hostname that is entered into Unity when connecting

resellerName: This is the ID of the enterprise or service provider in Broadworks

groupName: This is the ID of the group in Broadworks

broadworksUserId: This is the login ID [including domain] of the user in Broadworks

userName: This is the first name and last name of the user in Broadworks, if this changes it is automatically updated by the Unity client as part of the license request

userPhone: This is the VoIP number of the user in Broadworks, if this changes it is automatically updated by the Unity client as part of the license request

Provision

Click [here](#) for a complete list of operations.

AddUserWithHierarchy

Add a user to the system. The reseller and group do not have to exist in the system as anything that does not yet exist will be created within the system provider.

Test

To test the operation using the HTTP POST protocol, click the 'Invoke' button.

Parameter	Value
loginUsername:	<input type="text"/>
loginPassword:	<input type="text"/>
serverHostName:	<input type="text"/>
resellerName:	<input type="text"/>
groupName:	<input type="text"/>
broadworksUserId:	<input type="text"/>
userName:	<input type="text"/>
userPhone:	<input type="text"/>
<input type="button" value="Invoke"/>	

If the values do not match those in Broadworks directly, the licensing server will not be able to match the account so will either create a new account with the correct details, or could reject the license request.

In fact the AddUserWithHierarchy web method as outlined above is the easiest way to quickly add a user, because it will add the enterprise/service provider and group if they don't already exist. If they do already exist they will not be created again and an error will not occur. There are separate web methods to add a new reseller and group, but we recommend using this web method whenever you want to add a new user account.

The next most important web method is AddUserLicense, as below.

Provision

Click [here](#) for a complete list of operations.

AddUserLicense

Assign a license to a user. The system provider, reseller, group and user must already exist in application currently assigned to the user will be unassigned. This suits an upgrade or downgra

Test

To test the operation using the HTTP POST protocol, click the 'Invoke' button.

Parameter	Value
loginUsername:	<input type="text"/>
loginPassword:	<input type="text"/>
serverHostName:	<input type="text"/>
broadworksUserId:	<input type="text"/>
applicationName:	<input type="text"/>
licenseType:	<input type="text"/>
<input type="button" value="Invoke"/>	

The applicationName and licenseType values must both be one of several constants, and the licenseType depends on applicationName, because different Unity applications have different licenses.

The different permutations are below

ApplicationName	LicenseType
UnityDesktop	Lite
UnityDesktop	Pro
UnityDesktop	Enterprise
UnityAgent	Standard
UnityAgent	Enterprise
UnitySupervisor	Standard
UnitySupervisor	Enterprise
UnityReception	Standard
UnityReception	Enterprise
UnityWallboard	Standard
UnityMobile	Standard