



## Document Revision History

Revision	Date	Reason
A	August 2017	Created for the 4.2 release.

## Trademarks

All intellectual property is protected by copyright. All trademarks and product names used or referred to are the copyright of their respective owners. No part of this document may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, chemical, photocopy, recording or otherwise without the prior written permission of Thales Group.

## Disclaimer and Copyrights

All information herein is either public information or is the property of and owned solely by Gemalto NV. and/or its subsidiaries who shall have and keep the sole right to file patent applications or any other kind of intellectual property protection in connection with such information.

Nothing herein shall be construed as implying or granting to you any rights, by license, grant or otherwise, under any intellectual and/or industrial property rights of or concerning any of Gemalto's information.

This document can be used for informational, non-commercial, internal and personal use only provided that:

- The copyright notice below, the confidentiality and proprietary legend and this full warning notice appear in all copies.
- This document shall not be posted on any network computer or broadcast in any media and no modification of any part of this document shall be made.

Use for any other purpose is expressly prohibited and may result in severe civil and criminal liabilities.

The information contained in this document is provided "AS IS" without any warranty of any kind. Unless otherwise expressly agreed in writing, Gemalto makes no warranty as to the value or accuracy of information contained herein.

The document could include technical inaccuracies or typographical errors. Changes are periodically added to the information herein. Furthermore, Gemalto reserves the right to make any change or improvement in the specifications data, information, and the like described herein, at any time.

Gemalto hereby disclaims all warranties and conditions with regard to the information contained herein, including all implied warranties of merchantability, fitness for a particular purpose, title and non-infringement. In no event shall Gemalto be liable, whether in contract, tort or otherwise, for any indirect, special or consequential damages or any damages whatsoever including but not limited to damages resulting from loss of use, data, profits, revenues, or customers, arising out of or in connection with the use or performance of information contained in this document.

Gemalto does not and shall not warrant that this product will be resistant to all possible attacks and shall not incur, and disclaims, any liability in this respect. Even if each product is compliant with current security standards in force on the date of their design, security mechanisms' resistance necessarily evolves according to the state of the art in security and notably under the emergence of new attacks. Under no circumstances, shall Gemalto be held liable for any third party actions and in particular in case of any successful attack against systems or equipment incorporating Gemalto products. Gemalto disclaims any liability with respect to security for direct, indirect, incidental or consequential damages that result from any use of its products. It is further stressed that independent testing and verification by the person using the product is particularly encouraged, especially in any application in which defective, incorrect or insecure functioning could result in damage to persons or property, denial of service or loss of privacy.

©Gemalto 2017. All rights reserved. Gemalto, the Gemalto logo, are trademarks and service marks of Gemalto and are registered in certain countries

**Product Version:** Sentinel EMS 4.2

**Document Number** : 007-013916-001, Revision A

**Release Date**: August 2017

# CONTENTS

<b>PREFACE</b> .....	<b>5</b>
Audience .....	5
Support Contacts .....	5
<b>Introduction</b> .....	<b>6</b>
Namespace .....	7
License Model .....	7
Features .....	8
Products .....	8
Entitlement .....	8
Activation .....	8
<b>Prerequisites</b> .....	<b>9</b>
Scenario .....	9
Primary Preparations .....	10
<b>Login Web Service to Create a Session</b> .....	<b>11</b>
<b>Define Namespace</b> .....	<b>12</b>
<b>Define Features</b> .....	<b>13</b>
<b>Define a Product</b> .....	<b>15</b>
<b>Create Entitlements</b> .....	<b>17</b>
Create a Basic Entitlement with Product 'Print-Scan' .....	17
<b>Modify an Entitlement</b> .....	<b>20</b>
<b>Create and Complete Activation</b> .....	<b>22</b>

# PREFACE

## Audience

---

This guide is meant for first time user of Sentinel EMS Web services where individuals are well versed with Java , XML and REST framework.

## Support Contacts

---

You can contact us using any of the following options:

- **Business Contacts**  
To find the nearest office or distributor, use the following URL:  
<https://sentinel.gemalto.com/contact-us-sm/>
- **Technical Support**  
To obtain assistance in using Gemalto Sentinel Products, feel free to contact our Technical Support team:
  - Customer Support Portal (preferred):  
<https://supportportal.gemalto.com/csm?id=sentinel>
  - Phone:  
AMER: 800-545-6608 (US toll free), International: +1-410-931-7520  
EMEA/APAC: <https://supportportal.gemalto.com/csm?id=sentinel>  
Click “Contact us”
  - E-mail (only if you cannot submit the technical issue via the portal)
  - [technical.support@gemalto.com](mailto:technical.support@gemalto.com)
- **Downloads**  
To download installers and other updated components:  
<https://sentinelcustomer.gemalto.com/sentineldownloads/>

# Introduction

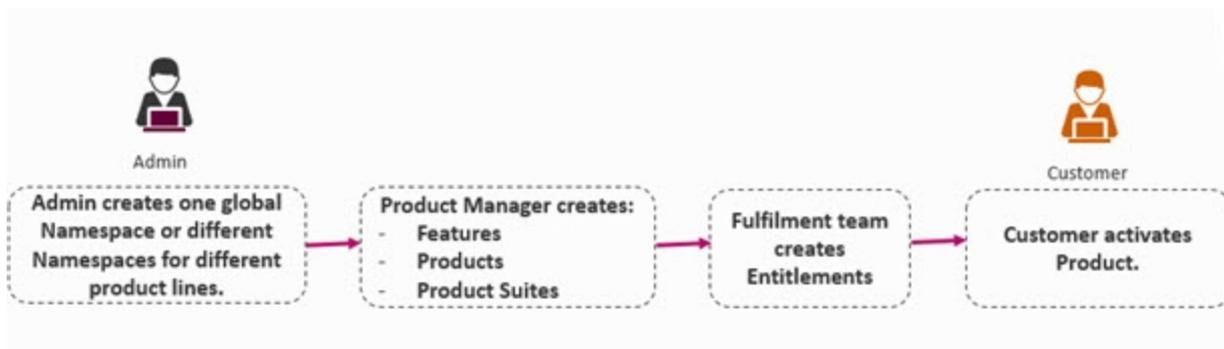
Sentinel EMS is a Web-based solution that provides you with a centralized interface for all your license and Entitlement management functions. Sentinel EMS provides several Web services to work with entities such as Namespace, License Models, Features, Products, Entitlements, and Activation.

The Getting Started Guide to Web Services:

- Introduces the entities Namespace, License Models, Features, Products, Entitlements, and Activation.
- Provides a scenario where the user will log in and create entities such as Features, Products, and Entitlements based on parameters defined in the scenario.
- Discusses the prerequisites for this scenario.
- Provides sample XML input for each step to fulfill the defined scenario.

The activities described in this document are typically performed by different users:

- The admin creates one global Namespace. If he wants to prevent Users in one Namespace (division or region) from being able to view or edit Products in another Namespace, then he can create multiple Namespaces.
- A product manager creates Features and associates License Models with them. Then he creates Products and associates Features with them.
- The fulfillment team can then generate Entitlements for the Products.
- A customer activates products using the Entitlement details.



---

**Note:** Sentinel EMS provides a pre-defined administrative User (**admin**). After installing Sentinel EMS, the admin first needs to log on to the portal using his User ID and password.

---

Following are some definitions that are used in this document.

## Namespace

Namespace acts as a logical partition for your EMS workspace. If you have different Product lines, you can maintain a different Namespace for each and define multiple Products in each Namespace. Namespaces are mutually exclusive. That is, a Product manager cannot pick Features from different Namespaces and club into a Product. Similarly, he cannot pick Products from different Namespaces and club them into a Suite.

## License Model

A license model defines the way in which a feature can be used and its restrictions. For example, a stand-alone license model enables you to restrict the use of software for a specific machine. You can configure usage terms to meet your company's individual requirements. You can choose from a variety of built-in license models. You can also create your own license model by copying and modifying existing ones.

In Sentinel EMS web services, each license model has a numeric database ID, called `licenseModelId`. When you create a new license model, the next available numeric value is assigned as its ID. If you plan to use a predefined license model, you can use the following license models defined in Sentinel RMS.

Following predefined license models are available.

<b>licenseModelId</b>	<b>Value</b>
1	Trial Standalone
2	Trial Network
3	Subscription
4	Network Node Locked
5	Standalone Node Locked
6	Commuter
7	Auto Checkout
8	Trial Standalone Node Locked
9	Trial Network Node Locked
10	Pooled Capacity Node Locked
11	Non-Pooled Capacity Node Locked
12	Redundant Node Locked
13	Flexible License Model

---

## Features

---

A feature is a distinct functionality of a software application that can be independently licensed. Features are the basic building blocks of a product. One or more features can be bundled into a product according to your product monetization model. In other words, you can break down your products to different usable features. By doing this, you can give your customers flexibility to purchase only those features and capabilities of your product that they actually need. A feature is not a sale-able entity, and it needs to be part of a product.

In Sentinel EMS, a Feature may be an entire application, a module or a specific functionality such as Print, Save, or Draw. Features are combined to form Products. License models are defined for each Feature in a Product.

---

## Products

---

A Product is a sale-able entity that can contain one or many features. It is a combination of individual features and is associated with one or more license models. It can be a software application that can be ordered or licensed according to your licensing plan.

As you cannot sell a feature directly, you will need to bundle it in a product. License models are defined for each feature in a product. The implementation scenario can be one of the following:

- You create a common product with all the related features bundled. The order taker can switch on/off features while producing the order.
- You can define a number of products with combination of different features by understanding the market requirements and keep them ready for order.

---

## Entitlement

---

An Entitlement refers to the rights of an end-user to use a software package. It includes information like who can use it and for what duration. After the end-user obtains these rights he can activate, which generates a license to use the software conforming to the defined terms. A license is the final outcome of an entitlement.

---

## Activation

---

Product activation refers to a method by which the machine-specific **lock code** of a computer is used to generate a license code or file specific to that computer. The license code ensures that the Product can be used in the system(s) to which it is locked.

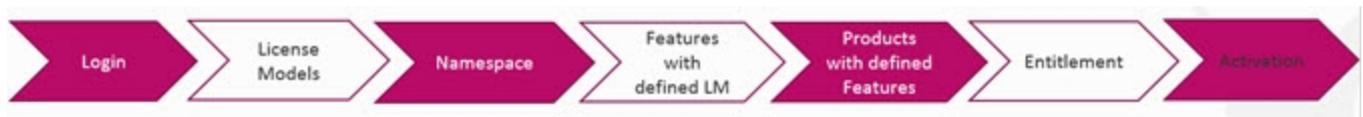
# Prerequisites

## Scenario

As the administrator, our requirement is to create a Product named Printer-Scanner that will contain two Features named Print and Scan with 'Trial Standalone' and 'Trial Standalone Node Locked' as the license models. We will create a basic entitlement first using the product created earlier and then activate the product.

In order to define an Entitlement, some resources should already be present in your Sentinel EMS system. Following is a list of resources/functions that are required before defining an Entitlement:

- Login
- Predefined License Models
- Namespace (workspace to place features, products, entitlements)
- Features with Defined LM
- Products with Defined Features
- Customers



To fulfill the scenario defined above, you will have to:

1. Login as the admin.
2. Create a Namespace, 'Global'.
3. Create a Features, 'Print' and 'Scan'. Both the Features should be of License Model type - 'Trial Standalone' and 'Trial Standalone Node Locked'. 'Trial Standalone' should be marked as the default License Model.
4. Create a Product, 'Printer-Scanner' to include both features defined.
5. Create an Entitlement with 'Printer-Scanner' as a line item.
6. Modify the Entitlement.
7. Activate the Entitlement.

The table given below summarizes the entities and parameters being used in the scenario:

Entity	Value
User	admin
License Model	Trial Standalone Trial Node Locked

Entity	Value
Namespace	Global
Feature	Print Scan
Product	Printer-Scanner
Entitlement	System generated Entitlement ID
Activation	System generated Activation ID

## Primary Preparations

1. Download HttpClient and its dependencies to your system and include them in your system's classpath. (For example, you can use the link: <https://hc.apache.org/downloads.cgi>)
2. Create a property file to read different EMS settings like EMS server name, http port, https port, etc. You can use the property file present in the emsWSDemo tool and modify the required parameters. The following parameters should be defined in your property file:
  - server = localhost:8080
  - sslServer = localhost:8443
  - trustStore = c:/cacerts
  - trustStorePass = changeit
3. Write Java code to read values like server URL, trust store location, and password from property file. See Web Services Guide for details.
4. Set `javax.net.ssl.trustStore` and `javax.net.ssl.trustStorePassword` to authenticate the EMS Server specified in server property.

# Login Web Service to Create a Session

To use any Sentinel EMS web service, you need to first establish a session. You can use verifyLogin.xml for this purpose. When a client application invokes the verifyLogin method it passes in a user name and password as user credentials. Upon invocation, the EMS server authenticates the credentials and returns a sessionId for the session to use in all subsequent web service calls. You will require the following from your EMS administrator:

- A user name in Sentinel EMS (with adequate permission to create Features, Products, and Entitlements)
- A password. Otherwise, you can use the default admin account to log on. The default password for this account is admin. However, your administrator might have changed the password.

Default Protocol	Method Type	URI
HTTPS	POST	/ems/verifyLogin.xml

The parameters used in the Login Web service are:

Parameter	Description	Type
userName	(Required) Login User Name	String
password	(Required) Login password associated with the specified user name	String

## Sample Response

When you establish a session after providing a valid user name and password, a session ID is returned as response:

```
<?xml version="1.0" encoding="UTF-8"?>
<EMSResponse>
  <sessionId>7D8B62A40233770B476C61A0DEF96878</sessionId>
  <stat>ok</stat>
</EMSResponse>
```

# Define Namespace

You can define a namespace to create a workspace for the features, products, entitlements being defined. Create a namespace, with the namespaceName as 'Global':

You can create namespace using the Namespace.xml web service.

Default Protocol	Method Type	URI
HTTP	POST	/ems/addNamespace.xml

The parameters used in the Namespace Web service are:

Parameter	Description	Type
namespaceName	(Required) The namespace name.	String
namespaceDescription	(Optional) Description of the namespace.	String
refId1	(Optional) Identifier 1 to store the unique record ID as used by external ERP/CRM systems.	String
refId2	(Optional) Identifier 2 to store the unique record ID as used by external ERP/CRM systems.	String

## Sample Response

```
<?xml version="1.0" encoding="UTF-8"?>
<EMSResponse>
  <namespaceId>1</namespaceId>
  <stat>ok</stat>
</EMSResponse>
```

# Define Features

You can define Features in your application that you want to license individually. For our tutorial, we will create two Features named Print and Scan with 'Trial Standalone' and 'Trial Standalone Node Locked' license model.

First, create a Feature named Print with the following parameters:

Parameter	Value	Meaning
featureName	Print	Name of the Feature
enforcementId	1	Sentinel RMS, v9
licenseModelId	1	Trial Standalone
	8	Trial Standalone Node Locked. Note: See definition of License Model in Introduction to know about predefined and custom-made license models.
isDefaultLM	True	Trial Standalone is the default license model. At least one License Model has to be added to a Feature. If multiple License Models are added to a Feature, one of them has to be marked as default.

You can create features using the feature.ws web service.

Default Protocol	Method Type	URI
HTTP	PUT	v4_1/ws/feature.ws

Use the following XML to create a Feature, Print:

```
<feature>
  <namespaceIdentifier>
    <namespaceId>1</namespaceId>
    <namespaceName>Global</namespaceName>
  </namespaceIdentifier>
  <externalFtrId></externalFtrId>
  <featureIdentifier></featureIdentifier>
  <featureName>Print</featureName>
  <refId1></refId1>
  <refId2></refId2>
  <description></description>

  <enforcement> /* License model 1*/
    <enforcementIdentifier>
      <enforcementId>1</enforcementId>
    </enforcementIdentifier>
    <featurelicenseModel>
      <licenseModelIdentifier>
```

```

        <licenseModelId>6</licenseModelId>
        <licenseModelName></licenseModelName>
    </licenseModelIdentifier>
    <isDefaultLm>True</isDefaultLm>
</featurelicenseModel>
</enforcement> /* end */

<enforcement> /* License model 2*/
    <enforcementIdentifier>
        <enforcementId>1</enforcementId>
    </enforcementIdentifier>
    <featurelicenseModel>
        <licenseModelIdentifier>
            <licenseModelId>8</licenseModelId>
            <licenseModelName></licenseModelName>
        </licenseModelIdentifier>
        <isDefaultLm>False</isDefaultLm>
    </featurelicenseModel>
</enforcement> /* end */
</feature>

```

Similarly, create another Feature named Scan with all other parameters remaining same:

Parameter	Value
featureName	Scan
enforcementId	1
licenseModelId	1
	8
isDefaultLM	True

You can use the previous XML, by changing the featureName parameter to Scan, for example, `<featureName>Scan</featureName>`.

## Sample Response

### Success

HTTP Status Code: 201 (Created)

Header-Location: {feature Id}

### Failure

HTTP Status Code: 400 (Bad Request)

Response Body: Feature {feature name} already belongs to this namespace. Try again with a different name and version.

# Define a Product

You can define new Products by mixing and matching Features at any time. For this tutorial, we will define a Product named Printer-Scanner. The Product will include the two Features mentioned above. The Products that we define now will be included in an Entitlement that we will define later.

Parameter	Value	Meaning
productName	Printer-Scanner	Name of the Product.
productVersion	1	Version is a must to create a Product.
lifeCycleStage	COMPLETE	You can: - either create the Product as COMPLETE - or create as DRAFT and COMPLETE later
namespaceId	1	The Namespace to which the Product should belong.
familyID	6	The Product Family to which the Product should belong.
activationMethod	SAOT	SAOT enables you to specify the activation method while defining the Entitlement.
isFixed	False	Specify whether the License Model associated with the Feature is fixed when Entitlement is created.

You can create products using the product.ws web service.

Default Protocol	Method Type	URI
HTTP	PUT	v4_1/ws/product.ws

Use the following XML sample to create a Product, Printer-Scanner:

```
<product>
  <productName>Printer-Scanner</productName>
  <productVersion>1</productVersion>
  <lifeCycleStage>COMPLETE</lifeCycleStage>
  <namespace>
    <namespaceId>1</namespaceId>
  </namespace>
  <family>
    <familyId>6</familyId>
  </family>
  <enabled>true</enabled>
  <activationMethod>SAOT</activationMethod>
  <fixedQuantity>0</fixedQuantity>
/*Insert Feature 'Print' with License Model 'TSNL' here */
```

```

<productFeatureRef>
  <featureIdentificationInfo>
    <featureId>3</featureId>
  </featureIdentificationInfo>
  <productFeatureState>Optional_DefaultOn</productFeatureState>
  <productFeatureLicenseModel>
    <licenseModel>
      <licenseModelId>8</licenseModelId>
    </licenseModel>
  </productFeatureLicenseModel>
</productFeatureRef>
/*Out of the two license model IDs 6 and 8 available in the feature, we have used 8 */
</licenseModel>
<default>
  <isFixed>>false</isFixed>
</default>
</productFeatureLicenseModel>
</productFeatureRef>
/*End Feature 'Print' */
/*Insert Feature 'Scan' with License Model 'TSNL' here */
<productFeatureRef>
  <featureIdentificationInfo>
    <featureId>4</featureId>
  </featureIdentificationInfo>
  <productFeatureState>Optional_DefaultOn</productFeatureState>
  <productFeatureLicenseModel>
    <licenseModel>
      <licenseModelId>8</licenseModelId>
    </licenseModel>
  </productFeatureLicenseModel>
</productFeatureRef>
/*Out of the two license model IDs 6 and 8 available in the feature, we have used 8 */
</licenseModel>
<default>
  <isFixed>>false</isFixed>
</default>
</productFeatureLicenseModel>
</productFeatureRef>
/*End Feature 'Scan' */
</product>

```

If you do not know the featureId, you may use the search feature web service to get it using some other known parameter. (URL format: /ems/v4\_1/ws/searchFeature.ws?featureName=&namespaceName=&sortOrder=&pageSize=&sortCol=&externalId=&pageIndex=&refId2=&refId1=&featureVersion=)

# Create Entitlements

In this section, we will create a basic entitlement first using the product created earlier. We will discuss some more scenarios later.

## Create a Basic Entitlement with Product 'Print-Scan'

In this section we will create an Entitlement with minimum prerequisites. Let us use the product Printer-Scanner without any modification.

Parameter	Value	Meaning
startDate	yyyy-mm-dd format	Entitlement start date.
endDate	yyyy-mm-dd format	Entitlement end date. A value of 2500-12-31 signifies 'never expires'
customerId	1	(Integer) Identifier of the customer in EMS. This value may be different in your EMS environment. Use Get Customer web service to get the actual customerId.
contactId	1	(Integer) Identifier of the contact associated with the customer in EMS. This value may be different in your EMS environment. Use Get Contact web service to get the actual contactId.
emailId	demo@democust.com	Email of the customer.
userRegistration	OPTIONAL	Specify whether end-user registration is REQUIRED or OPTIONAL during activation.
activationAllowed	TRUE	Whether you want to allow activation for this Entitlement? Also, you can modify this later.
revocationAllowed	TRUE	Whether you want to allow revocation for this Entitlement? Also, you can modify this later.
lifeCycleStage	COMMITTED	The state in which you want to create the Entitlement. The default value is DRAFT. <ul style="list-style-type: none"><li>- DRAFT - The Entitlement is not yet ready. It can be modified or deleted.</li><li>- COMMITED – The Entitlement is complete. It can be activated.</li></ul>

Parameter	Value	Meaning
sendNotification	TRUE	Specify whether you want to send a notification to the contact specified.
totalQuantity	10	The number of activations permitted.
productId	1	(Integer) Identifier of the Product (Print-Scan) in EMS. This value may be different in your EMS environment. Use Get Product web service to get the actual productId.

You can create Entitlements using the entitlement.ws web service.

Default Protocol	Method Type	URI
HTTP	PUT	v4_1/ws/entitlement.ws

As we are using the product Printer-Scanner (without any modification) in this entitlement, we just need the productId within the <Item></Item> tags. You can use the following XML to create an Entitlement for the Product, Printer-Scanner.

```
<entitlement>
  <startDate>2015-10-23</startDate>
  <endDate>2500-12-31</endDate>
  <customerIdentifier>
    <customerId>3</customerId>
  </customerIdentifier>
  <contactIdentifier>
    <contactId>2</contactId>
    <emailId>Demo@democust.com</emailId>
  </contactIdentifier>
  <userRegistration>OPTIONAL</userRegistration>
  <activationAllowed>true</activationAllowed>
  <revocationAllowed>true</revocationAllowed>
  <action>COMMIT</action>
  <lifeCycleStage>COMMITTED</lifeCycleStage>
  <sendNotification>true</sendNotification>
  <productKey>
    <Item>
      <enforcementIdentifier>
        <enforcementName>Sentinel RMS</enforcementName>
        <enforcementVersion>9.0</enforcementVersion>
      </enforcementIdentifier>
      <totalQuantity>10</totalQuantity>
      <product>
        <productId>2</productId>
      </product>
    </Item>
  </productKey>
</entitlement>
```

## Sample Response

### Success

HTTP Status Code: 201 (Created)

Header-Location: {entitlement Id}

### Failure

HTTP Status Code: 400 (Bad Request)

Response Body: External ID 1 already exists. Try again with a different External ID.

## Exercise - Create an Entitlement with 'Flexible/Leased' License

At this point you can try and create another entitlement for a leased license. Leased licenses require to be renewed at defined intervals by connecting to a cloud server. The benefits of using this license model include reducing cost for users who use your software infrequently.

Let us use the Product, Printer-Scanner and create an Entitlement with the following modifications.

1. Create a new Namespace, 'DemoSpace'
2. While defining Features Print and Scan (as described earlier), use the following parameter for license model ID.

licenseModelId	13	Flexible License Model
----------------	----	------------------------

3. While defining the Product Printer-Scanner, use this license model ID (i.e., 13).
4. Create the Entitlement using the XML provided in the previous section without changing any parameter.

# Modify an Entitlement

You can modify an Entitlement if it is in Complete state. In this section we will modify an Entitlement previously created. Use GET Entitlement web service to get the Entitlement XML. Get the entId of the entitlement and use it within the <linkedEntId> </linkedEntId> tags, change the following parameters.

Parameter	Action	Value	Meaning
startDate		yyyy-mm-dd format	Entitlement start date. You cannot change the end date of an Entitlement to a date prior to the Product's start date.
endDate	Renew	yyyy-mm-dd format	Entitlement end date. A value of 2500-12-31 signifies 'never expires'
customerId		1	(Integer) Identifier of the customer in EMS. This value may be different in your EMS environment. Use Get Customer web service to get the actual customerId.
contactId		1	(Integer) Identifier of the contact associated with the customer in EMS. This value may be different in your EMS environment. Use Get Contact web service to get the actual contactId.
emailId		demo@democust.com	Email of the customer.
userRegistration		OPTIONAL	Specify whether end-user registration is REQUIRED or OPTIONAL during activation.
activationAllowed		TRUE	Whether you want to allow activation for this Entitlement? Also, you can modify this later.
revocationAllowed		TRUE	Whether you want to allow revocation for this Entitlement? Also, you can modify this later.
sendNotification		TRUE	Specify whether you want to send a notification to the contact specified.
totalQuantity	Recharge	5	The number you provide here will be added to the total quantity. If you specify a negative

Parameter	Action	Value	Meaning
			value (for example, -5), the amount will be deducted from the total quantity.

You can modify Entitlements using the entitlement.ws web service. The Web Service, entitlement.ws, can be used to modify Entitlements by introducing the <linkedEntId>EntID</linkedEntId> tag in the XML input. The <action></action> tag specifies the action to be performed on the Entitlement.

Default Protocol	Method Type	URI
HTTP	PUT	v4_1/ws/entitlement.ws

Use the following XML to modify the Entitlement created in the previous section:

```
<entitlement>
<linkedEntId>18</linkedEntId>
<startDate>2015-10-23</startDate>
<endDate>2500-12-31</endDate>
<customerIdentifier>
<customerId>1</customerId>
</customerIdentifier>
<contactIdentifier>
<contactId>1</contactId>
<emailId>Demol@democust.com</emailId>
</contactIdentifier>
<userRegistration>MANDATORY</userRegistration>
<activationAllowed>true</activationAllowed>
<revocationAllowed>true</revocationAllowed>
<lifeCycleStage>COMMITTED</lifeCycleStage>
<sendNotifcation>true</sendNotifcation>
<productKey>
<Item>
<ItemId>14</ItemId>
<activationMethod>FIXED</activationMethod>
<totalQuantity>1</totalQuantity>
<fixedQuantity>1</fixedQuantity>
<product>
<productIdentifier>
<productId>3</productId>
</productIdentifier>
</product>
<enforcementIdentifier>
    <enforcementName>Sentinel RMS</enforcementName>
    <enforcementVersion>9.0</enforcementVersion>
</enforcementIdentifier>
</Item>
</productKey>
</entitlement>
```

# Create and Complete Activation

In this section, we will generate a license for the product using entitlement ID that was generated as a response of creating an entitlement for the product.

Default Protocol	Method Type	URI
HTTP	PUT	v4_1/ws/activation.ws

## Activation of Entitlement Created for Product

Use the following sample XML input to generate a license for the product, 'Printer Scanner'

```
<groupActivation groupActivationId="0">
  <userInput>
    <sendNotification>false</sendNotification>
    <activatorEmailId></activatorEmailId>
    <activateeEmailId></activateeEmailId>
    <comments>test</comments>
    <executedBy>admin</executedBy>
    <lineItem>
      <aid>51415822-f23b-w4727-8fc5-16a325776bb1</aid>
      <externalId></externalId>
      <lineItemIdentifier>
        <lineItemId>2</lineItemId>
        <productKey>97032f71-c504-4640-8ee5-6ada09cef74b</productKey>
      </lineItemIdentifier>
      <activationQuantity>1</activationQuantity>
      <activationAttributes>
        <attributeGroup groupName="Locking Info">
          <attribute>
            <attributeName>MANUAL_LIBRARY_SERIAL_NUMBER.target_id</attributeName>
            <attributeValue>2</attributeValue>
            <readOnly>false</readOnly>
          </attribute>
        </attributeGroup>
      </activationAttributes>
    </lineItem>
    <customAttribute>
      <name>66565</name>
      <value></value>
    </customAttribute>
  </userInput>
</groupActivation>
```

## Sample Response

### Success

HTTP Status Code: 201 (Created)

Header-Location: {ActivationId}

**Failure**

HTTP Status Code: 400 (Bad Request)

Response Body: Invalid XML string provided. Provide a valid XML.