

Licensee Secret

- [Introduction](#)
- [Validation flow](#)
- [Examples](#)
 - [Predefined mode](#)
 - [Client mode](#)

Introduction

Licensee Secret is an additional optional security feature. It adds globally a *node-locking* capability, and may be used in combination with any supported licensing models. Although using it with some licensing models, e.g. [Floating](#), does not make much sense, there is no restriction, feel free to find your own creative way of using Licensee Secret.

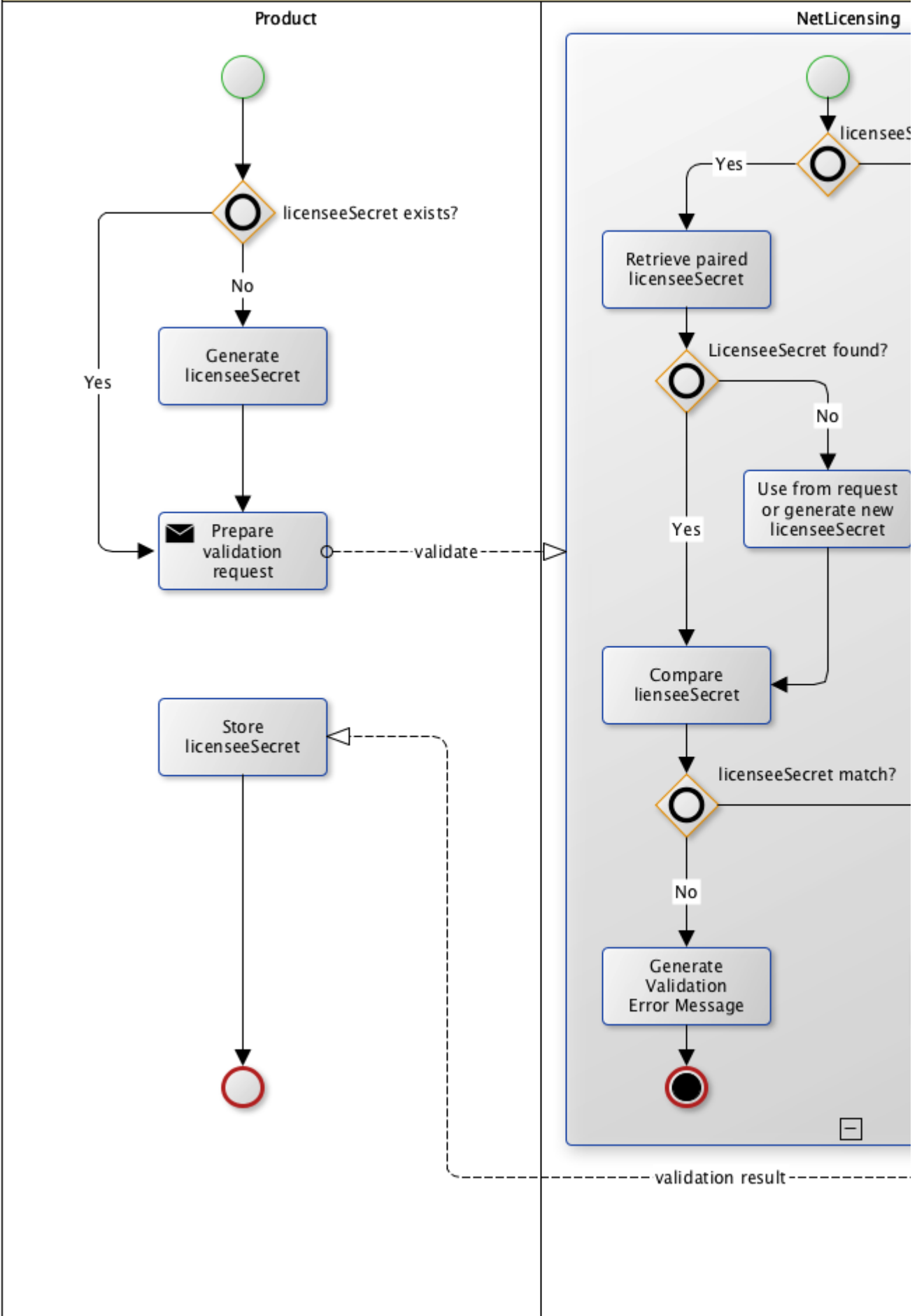
Licensee Secret feature has the following modes of operation, configurable in Product settings:

Mode	Description
Disabled	Validation parameter <i>licenseeSecret</i> will be ignored. Previously stored value retained, but is not accessible until the mode is changed.
Predefined	Secret is generated by vendor and passed on to the licensee (e.g. in a hardware dongle, or text-based encrypted key). The value must be explicitly set to the <i>licenseeSecret</i> property of the corresponding Licensee entity (using either Console or API). Licensee validation will only succeed if the value provided in <i>licenseeSecret</i> validate parameter matches the stored one. Since <i>licenseeSecret</i> must be set to the Licensee in advance, this mode can't be combined with "Auto-create Licensee" mode.
Client	This mode is similar to the Predefined mode, but unlike Predefined, the secret must not be set to the Licensee in advance, instead it will be set using the value provided in <i>licenseeSecret</i> parameter on first validate call. Once set, it works same way as <i>Predefined</i> mode. This will allow automatic locking to any secret value provided by the licensee, typically hash of some HW related data (e.g. MAC address, CPU serial number, etc). This mode can be used with hardware dongle too: the dongle holds the secret, but it is only bound to the specific licensee on the first validate .

Validation flow

Here is the licensee secret workflow diagram for the "**Client**" mode:

LicenseeSecret Pairing & Validation Flow



Examples

Use these examples to construct API calls when using licensee secret.

Predefined mode

▼ Create or Update Licensee

Request

```
POST
https://go.netlicensing.io/core/v2/rest/licensee/ITEST-DEMO?licenseeSecret=c99c12016c357200771b21645be64157
Accept: application/xml
```

Response

```
<netlicensing xmlns="http://netlicensing.labs64.com/schema/context">
  <infos/>
  <items>
    <item type="Licensee">
      <property name="number">ITEST-DEMO</property>
      <property name="active">true</property>
      <property name="productNumber">PTEST-DEMO</property>
      <property
name="licenseeSecret">c99c12016c357200771b21645be64157</property>
      <property name="name">Try & Buy licensing model demo
customer</property>
      <property name="inUse">true</property>
    </item>
  </items>
</netlicensing>
```

▼ Subsequent validate, secret matches

Request

```
POST https://go.netlicensing.io/core/v2/rest/licensee/ITEST-DEMO/validate
Accept: application/xml
Content-Type: application/x-www-form-urlencoded

licenseeSecret=c99c12016c357200771b21645be64157
```

Response

```
<netlicensing xmlns="http://netlicensing.labs64.com/schema/context"
ttl="2016-11-26T12:43:44.052Z">
  <infos/>
  <items>
    <item type="ProductModuleValidation">
      <property name="productModuleNumber">MTEST-DEMO</property>
      <property name="evaluation">true</property>
      <property name="valid">true</property>
      <property name="productModuleName">Module licensed under Try &
Buy licensing model</property>
      <property
name="evaluationExpires">2016-12-25T12:43:44.057Z</property>
      <property name="licensingModel">TryAndBuy</property>
    </item>
  </items>
</netlicensing>
```

Subsequent validate, secret mismatch

Request

```
POST https://go.netlicensing.io/core/v2/rest/licensee/ITEST-DEMO/validate
Accept: application/xml
Content-Type: application/x-www-form-urlencoded
```

```
licenseeSecret=WRONG_SECRET
```

Response

```
<netlicensing xmlns="http://netlicensing.labs64.com/schema/context"
ttl="2016-11-26T13:01:25.753Z">
  <infos>
    <info id="licenseeSecretMismatch" type="error">Licensee secret
mismatch</info>
  </infos>
  <items>
    <item type="ProductModuleValidation">
      <property name="productModuleNumber">MTEST-DEMO</property>
      <property name="evaluation">true</property>
      <property name="valid">false</property>
      <property name="productModuleName">Module licensed under Try &
Buy licensing model</property>
      <property
name="evaluationExpires">2016-12-25T12:43:44.057Z</property>
      <property name="licensingModel">TryAndBuy</property>
    </item>
  </items>
</netlicensing>
```

Client mode

First validate and Subsequent validate with matching Secret

Request

```
POST https://go.netlicensing.io/core/v2/rest/licensee/ITEST-DEMO/validate
Accept: application/xml
Content-Type: application/x-www-form-urlencoded

licenseeSecret=c99c12016c357200771b21645be64157
```

Response

```
<netlicensing xmlns="http://netlicensing.labs64.com/schema/context "
ttl="2016-11-26T13:05:59.255Z">
  <infos/>
  <items>
    <item type="ProductModuleValidation">
      <property name="productModuleNumber">MTEST-DEMO</property>
      <property name="evaluation">true</property>
      <property name="valid">true</property>
      <property name="productModuleName">Module licensed under Try &
Buy licensing model</property>
      <property
name="evaluationExpires">2016-12-25T12:43:44.057Z</property>
      <property name="licensingModel">TryAndBuy</property>
    </item>
  </items>
</netlicensing>
```

Subsequent validate, secret mismatch

Request

```
POST https://go.netlicensing.io/core/v2/rest/licensee/ITEST-DEMO/validate
Accept: application/xml
Content-Type: application/x-www-form-urlencoded

licenseeSecret=WRONG_SECRET
```

Response

```
<netlicensing xmlns="http://netlicensing.labs64.com/schema/context"
ttl="2016-11-26T13:01:25.753Z">
  <infos>
    <info id="licenseeSecretMismatch" type="error">Licensee secret
mismatch</info>
  </infos>
  <items>
    <item type="ProductModuleValidation">
      <property name="productModuleNumber">MTEST-DEMO</property>
      <property name="evaluation">true</property>
      <property name="valid">false</property>
      <property name="productModuleName">Module licensed under Try &
Buy licensing model</property>
      <property
name="evaluationExpires">2016-12-25T12:43:44.057Z</property>
      <property name="licensingModel">TryAndBuy</property>
    </item>
  </items>
</netlicensing>
```