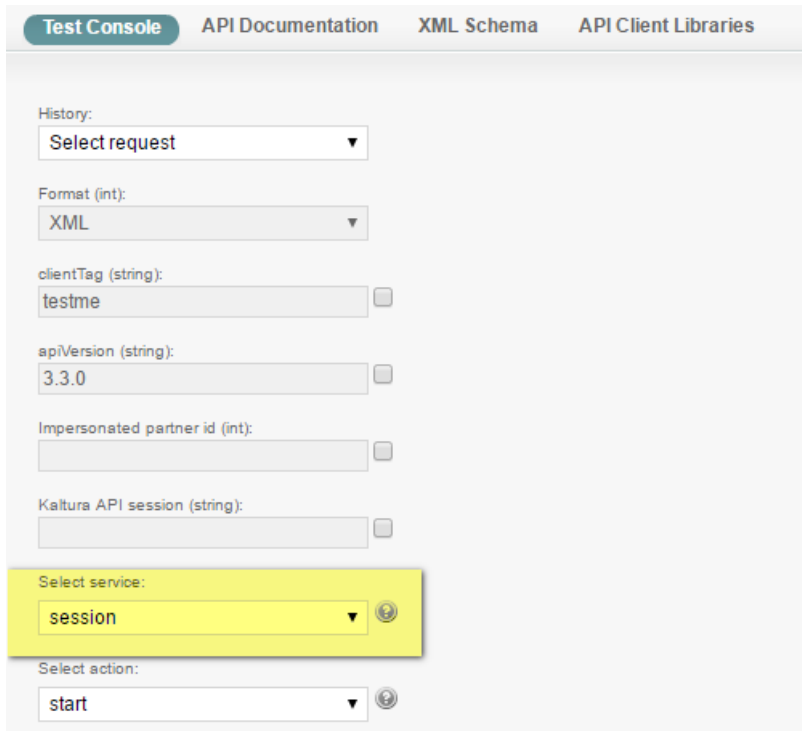


How to Start a Kaltura Session using the TestMe Console

Last Modified on 11/28/2021 1:32 pm IST

☰ To start a Kaltura Session

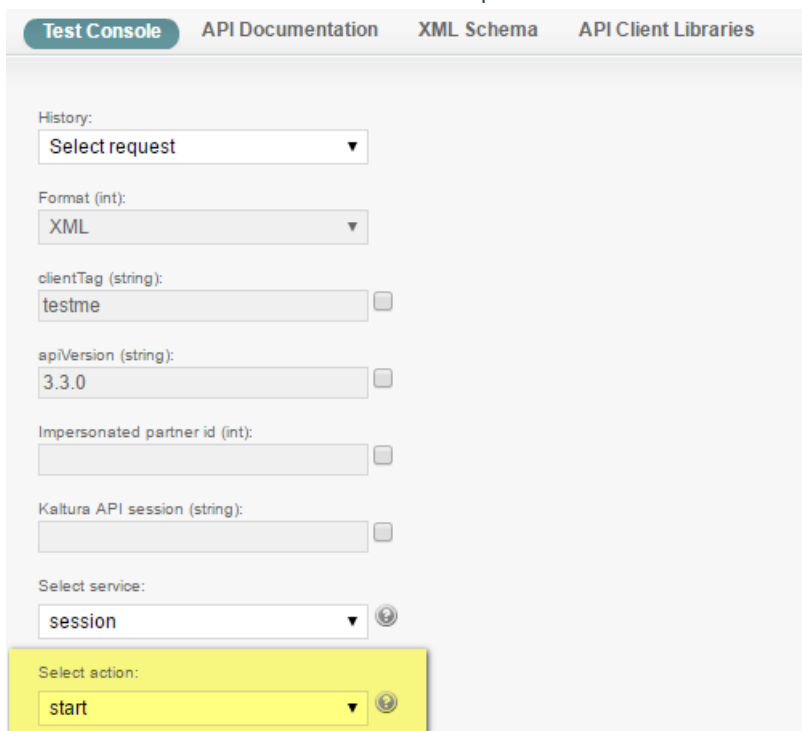
1. Go to: http://www.kaltura.com/api_v3/testme
2. Select **Session** from the **Select Service** drop-down list.



The screenshot shows the 'Test Console' interface with the following fields:

- History: Select request
- Format (int): XML
- clientTag (string): testme
- apiVersion (string): 3.3.0
- Impersonated partner id (int):
- Kaltura API session (string):
- Select service: session (highlighted in yellow)
- Select action: start

3. Select **Start** from the **Select action** drop-down list.



The screenshot shows the 'Test Console' interface with the following fields:

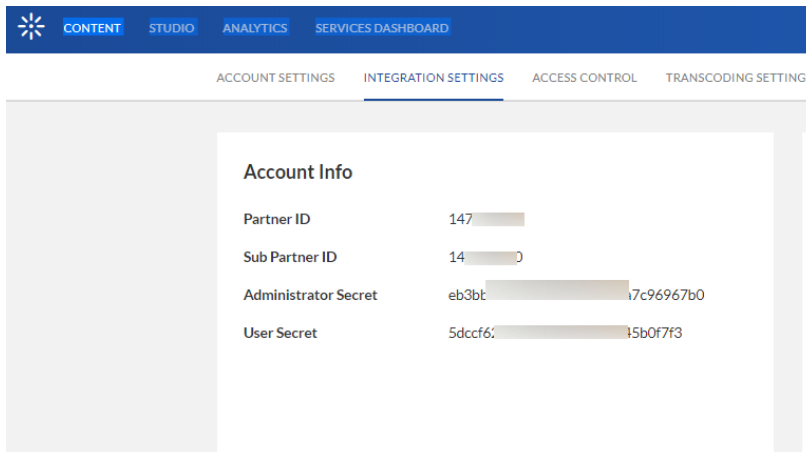
- History: Select request
- Format (int): XML
- clientTag (string): testme
- apiVersion (string): 3.3.0
- Impersonated partner id (int):
- Kaltura API session (string):
- Select service: session
- Select action: start (highlighted in yellow)

Log in the KMC and retrieve the Partner ID and Administrator Secret.

☰ To retrieve information from the KMC

1. Click the **Settings** button and then **Integration settings**.
2. Save the following information:(actual information is blurred intentionally)
 - a. **Partner ID.**
 - b. **Administrator Secret.**

The numbers and letters are different for each Partner.



3. Return to the TestMe Console and paste the Partner ID and Administrator Secret information, and change the type field from **USER** to **ADMIN**.

Test Console
API Documentation
XML Schema
API Client Libraries

History:
 Select request ▼

Format (int):
 XML ▼

clientTag (string):

apiVersion (string):

Impersonated partner id (int):

Kaltura API session (string):

Select service:
 session ▼

Select action:
 start ▼

secret (string):
 ✓

userId (string):

type (KalturaSessionType):
 ADMIN ▼ ✓

partnerId (int):
 ✓

expiry (int):

privileges (string):

Send

PHP
Java
C#
Python
Javascript
Show Code Example

4. Click Send.

The output results should contain the KS, a Kaltura Session key.

```

1 - <xml>
2   <result>Mjk4NjF1MT...</result>
3   <executionTime>2.288818359375E-5</executionTime>
4 </xml>
5
  
```

5. Copy the KS string and put it in the Kaltura API session (string) field.

That's it! Your API console is ready to make new API calls now.

Notes

The Kaltura API is [REST](http://en.wikipedia.org/wiki/Representational_state_transfer) (http://en.wikipedia.org/wiki/Representational_state_transfer) based and [stateless](http://en.wikipedia.org/wiki/Stateless_protocol) (http://en.wikipedia.org/wiki/Stateless_protocol). Every call (request) made to the Kaltura API requires an authentication key, the Kaltura Session (aka KS), identifying the account on which the action is to be carried, the authenticated user and its role.

A Kaltura Session can be initiated in 3 different ways:

1. Calling the `session.start` action, providing the account Partner ID and an Admin or User secret key.

```
$ks = $client->session->start ($adminSecret, $anyUserName, KalturaSessionType::ADMIN, $partnerId);  
$client->setKS($ks);
```

2. Calling the `user.loginByLoginId` action, providing a User login ID (email), its Password and the account Partner ID.

```
$ks = $client->user->loginByLoginId($loginId, $password, $partnerId);  
$client->setKS($ks);
```

3. Creating a session locally - Combine all the Kaltura Session details, and sign them using the shared secret.

```
$ks = $client->generateSession($adminSecret, $userId, $type, $partnerId);  
$client->setKS($ks);
```

The above examples are using the [Kaltura PHP5 Client Library](http://www.kaltura.com/api_v3/testme/client-lib.php) (http://www.kaltura.com/api_v3/testme/client-lib.php).

Important Security Notes:

1. **ADMIN type KS provides super admin privileges to the Kaltura account. If you're creating an application where the session will be exposed to the end-user, make sure that you are using a USER type KS and not ADMIN type. Exposing an ADMIN type KS in non-administrative context will expose your Kaltura account to risks of being used by malicious users with unrestricted access.**
2. **Sharing the account API secret keys with 3rd party vendors should be avoided, as secret keys can not be regenerated or blocked for access. Kaltura API based application developers and 3rd party application vendors should build their application to leverage the `user.loginByLoginId` API and ask the publisher for their email, password and account Id (aka partnerId). Users can be easily created, removed or blocked and their password can easily be changed.**

- To learn more about the KS and its usage read: [Kaltura's API Authentication and Security](https://developer.kaltura.com/api-docs/VPaaS-API-Getting-Started/Kaltura_API_Authentication_and_Security.html) (https://developer.kaltura.com/api-docs/VPaaS-API-Getting-Started/Kaltura_API_Authentication_and_Security.html): [The](#)



Kaltura Session (<https://developer.kaltura.com/api-docs/VPaaS-API-Getting-Started/how-to-create-kaltura-session.html>).

- To use the Kaltura API, you will need a Publisher Account with API access. Start a [free Kaltura.com trial](http://corp.kaltura.com/free-trial) (<http://corp.kaltura.com/free-trial>) or [download Kaltura CE](http://www.kaltura.org/project/community_edition_video_platform) (http://www.kaltura.org/project/community_edition_video_platform).
-