Kaltura Video Plugin for Jive Deployment Guide

Version: March 2018

KALTURA

Kaltura Business Headquarters

250 Park Avenue South, 10th Floor, New York, NY 10003

Tel.: +1 800 871 5224

Contents

Preface

This preface contains the following topics:

About this Guide

This guide describes how to deploy the Kaltura Video Plugin for Jive and the deployment of the NodeJS middleware application. This guide pertains to version 9.0.1.0-1 of the Kaltura Video Plugin for Jive.

NOTE: Please refer to the official and latest product release notes for last-minute updates.
Technical support may be obtained directly from: Kaltura Customer Care.

Contact Us:

Please send your documentation-related comments and feedback or report mistakes to knowledge@kaltura.com. We are committed to improving our documentation and your feedback is important to us.

Audience

This guide is intended for administrators deploying the Kaltura Video Plugin for Jive.

Document Conventions

Kaltura uses the following admonitions:

- Note
- Workflow

NOTE: Identifies important information that contains helpful suggestions.

Workflow: Provides workflow information.

1. Step 1
2. Step 2

SECTION 1

Software Versions

This guide pertains to version 9.0.1.0-1 of the Kaltura Video Plugin for Jive.

The application is developed for the following versions:

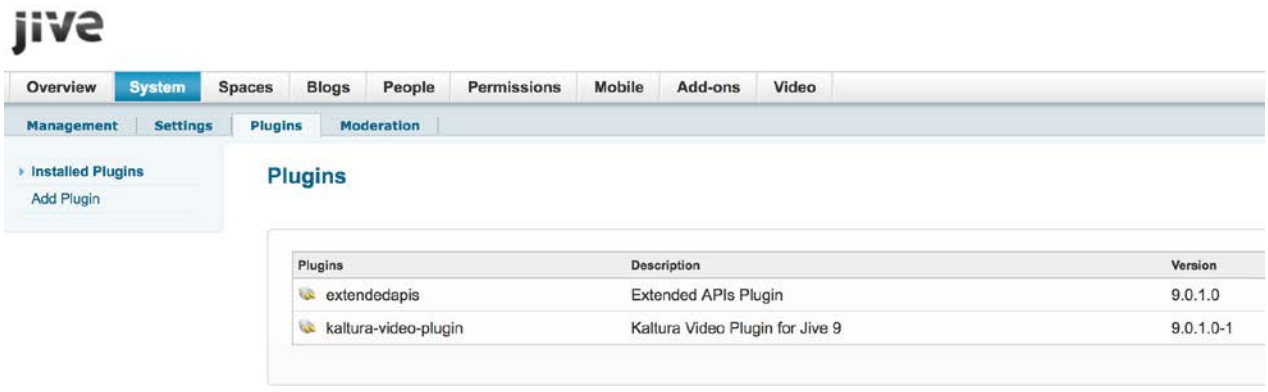
- Jive SDK: Jive Node SDK version 0.2.17
- NodeJS: v6.11.3 or or v9.3.0
- Jive SBS on premise: 9.0.1.0

Uploading the Kaltura Video for Jive Plugin Deployment

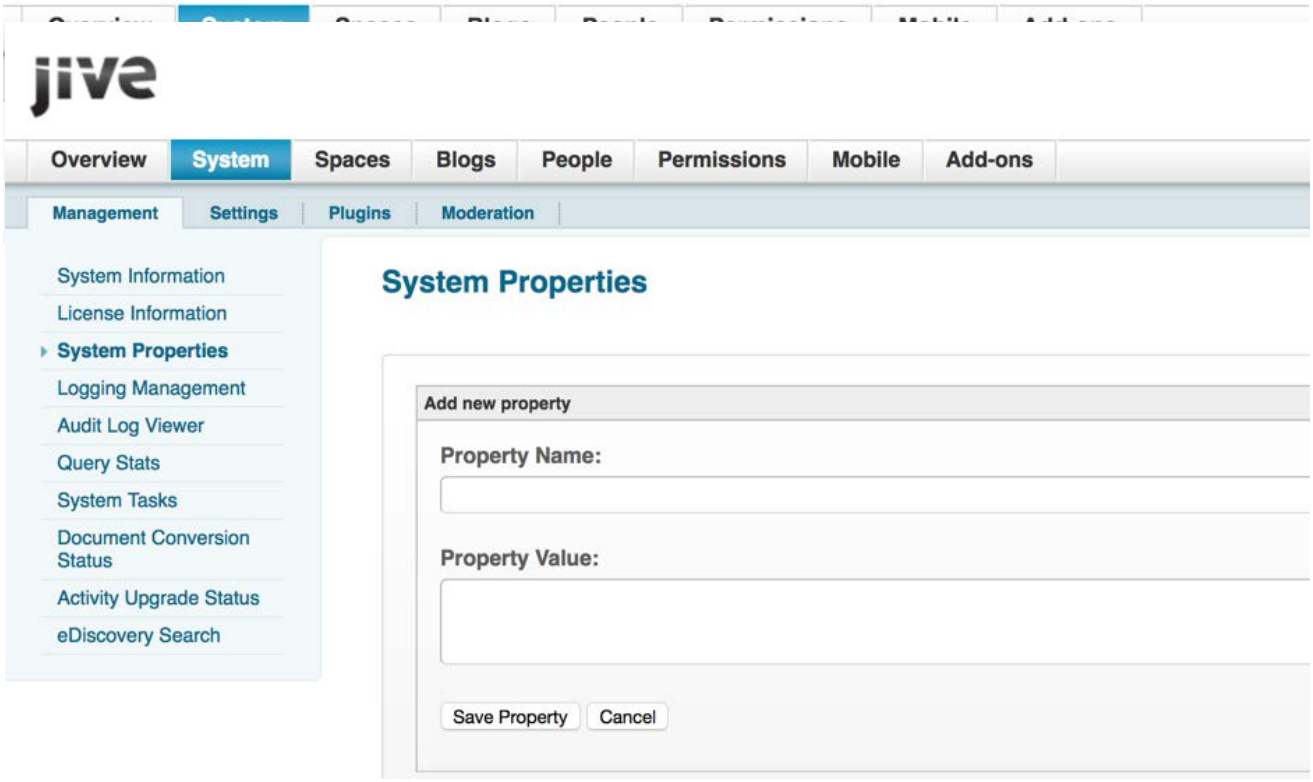
Upload the Plugin

To Upload the Plugin

1. Log into the Admin Console at http://community.mydomain.com/admin with a privileged user account.
2. Click on the main “System” tab, then the “Plugins” tab.



The available Plugins screen is displayed.

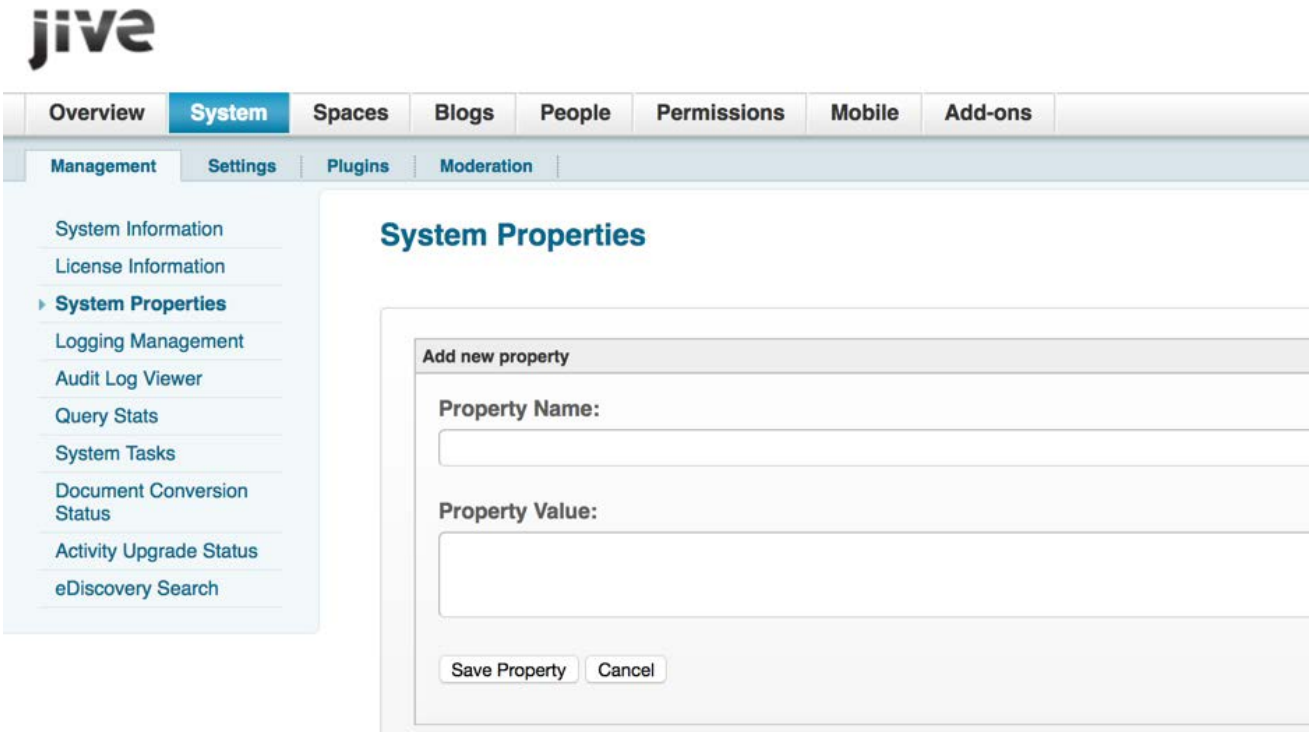


3. Click “Add Plugin” in the left-hand column.
4. Upload the Kaltura Video Plugin.

Configure the Plugin Settings

1. 1. In the Admin Console, click on the main "System" tab, then the "Management" tab.
2. Click "System Properties" in the left-hand column.

2. Click "System Properties" in the left-hand column.



3. Create a new system property with:
   - Property Name: kaltura.partnerid
   - Property Value: a number that identifies your Kaltura account
4. Click the "Save Property" button
5. Repeat the process and create a new system property with:
   - Property Name: kaltura.instance.url
   - Property Value: The KAF URL at which Kaltura can be contacted.

   The value of this parameter is likely similar to: http://[instanceid].mediaspace.kaltura.com
6. Click the "Save Property" button
7. Repeat the process and create a new system property with:
   - Property Name: kaltura.admin.secret
   - Property Value: The secret token that authorizes the Jive system to interact with Kaltura services.
8. Click the "Save Property" button
9. (Optional) Repeat the same process and create new optional system property with:
   - Property Name: kaltura.service.url
   - Property Value: the Kaltura service URL

   This property is not required for most installations, its default value is: http://www.kaltura.com

Restart the Jive Application Server

The Jive Application must be restarted to load the new plugin.

Deploying the Add-on

This guide pertains to version 0.9 of the Kaltura Add-on for Jive.

Database Setup

To set up the database

1. Create a Mongo database.
2. Copy the DB connection string into jiveclientconfiguration.json.
   For additional details on connection strings see:
   https://docs.mongodb.com/manual/reference/connection-string/

The jiveclientconfiguration.json

Install the package

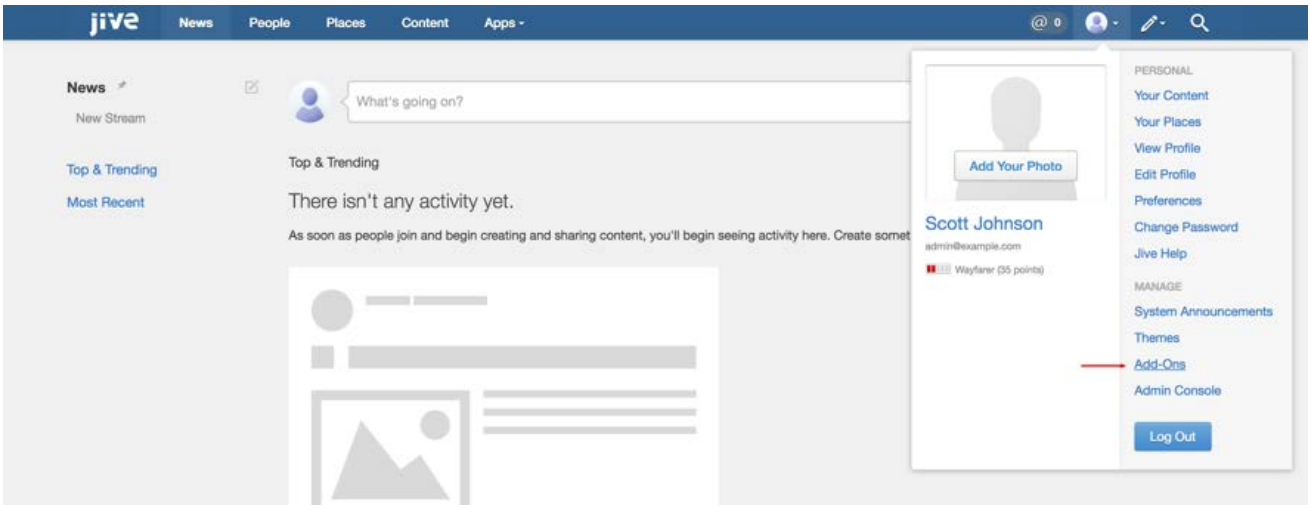
To install the package

1. Unpack the kaltura-connector-addon-1.0.0.tgz into the desired directory.
2. Check if the script fix-extension.sh is executable, if not, use chmod +x to do so

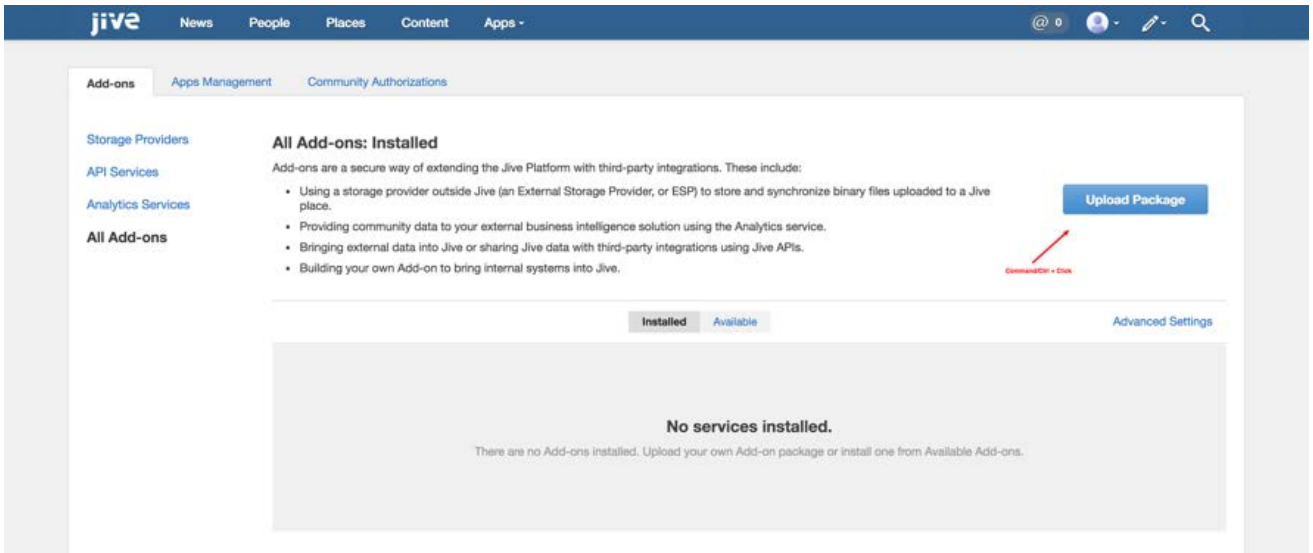
Addon Signing

Use the jiveclientconfiguration.json to configure and deploy the add-on.

1. Navigate to Add-on configuration page in Jive.



2. Command+Click (Mac) or Ctrl+Click (Windows) on "Upload Package".



3. Paste the UUID found in jiveclientconfiguration.json into the Sign Package field and click Sign.
4. Copy generated signature into jiveclientconfiguration.json 'jiveServiceSignature' field.

Note: Multiple signatures representing different Jive instances can be added with spaces separating the signatures.

Set the Middleware Server URL

1. Determine the URL of the server you will be hosting the middleware server on.
2. Copy this URL into jiveclientconfiguration.json in the 'clientUrl' field.
3. The default port for the add-on is 8090, however, if you will be proxying to standard 80/443 http ports, you should set the 'clientUrlExcludesPort' to ‘true'.

Configure the Proxy Server (Optional)

If a Proxy server is deployed between the Node server and the Kaltura service, support for the Proxy server is included.

- Add the following to the main JSON object in jiveclientconfiguration.json file to include parameters in this format:

```
proxy: {
    host: "myproxy.domain.com",
    port: 3812
}
```

Start the Server

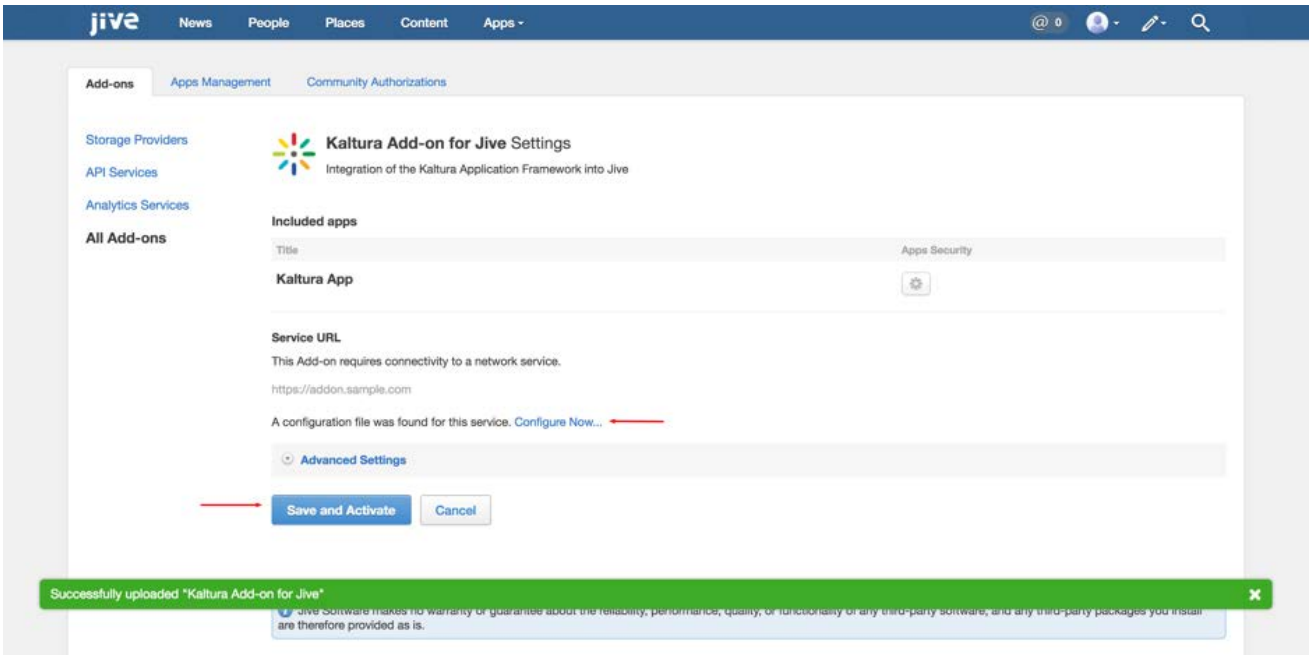
It is no longer required to run ‚npm update', the package will ship with all of its dependencies.

- Run the server by running `node app`.

Install the Add-on

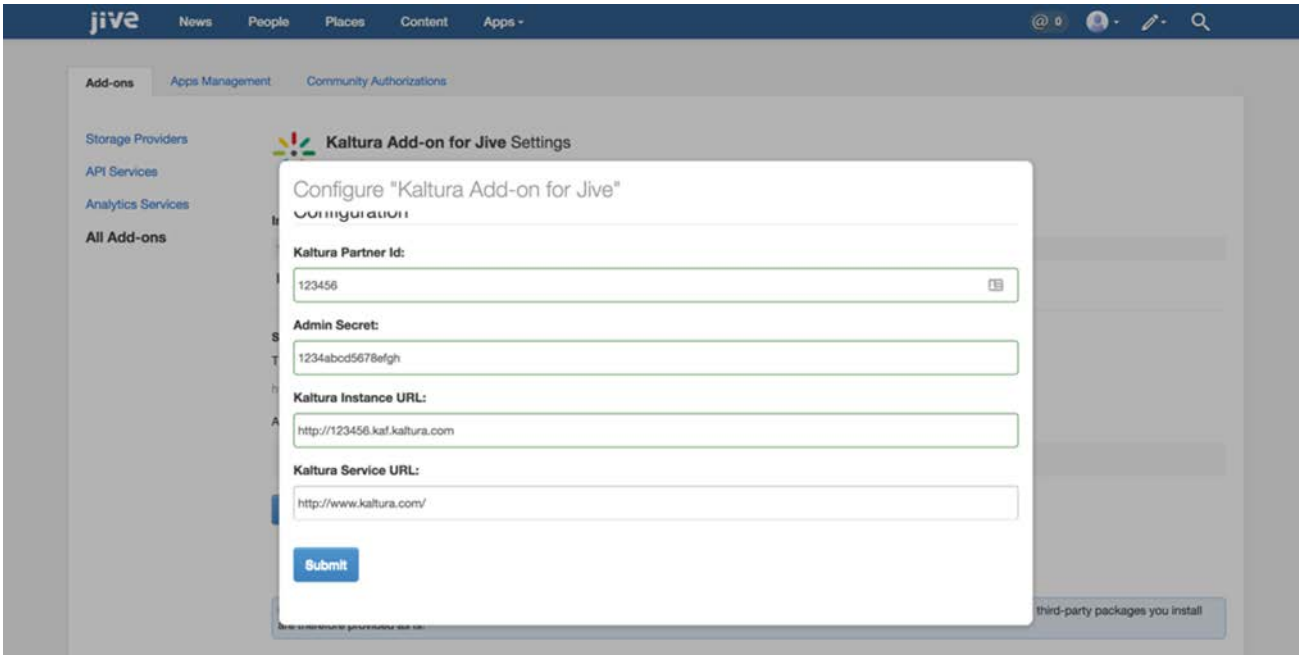
1. Execute the fix-extension.sh script.
2. Upload the file kaltura-connector-addon.zip to the Upload Package screen.

   This should indicate a success and show you the Kaltura Add-on settings page. You should also see the URL of your middleware server noted on that screen.



Addon Configuration

1. Before clicking "Save and Activate" configure the Add-on by clicking "Configure Now".



2. Enter the configuration details for your Kaltura account and click Submit.
3. After you submit the Configuration details for the Kaltura Add-on for Jive configuration click "Save and Activate“.

SECTION 3

Setting up HTTPS

If you plan to run the NodeJS application, you must place the certificate file and the certificate key file in a directory that is accessible for the process owner of the node application.

Open the configuration.json and change the paths for certificate files according to your environment.

It is not required to have different ports for HTTP and HTTPS. The application tests if there is a certificate available. If so, the application starts a HTTPS listener on the configured port. If not, it starts a HTTP listener on the configured HTTP port.