Synology Audio Station

Lyrics Module Development Guide

Synology

2013-01-11

2013-01-11 | ® 2013 Synology Inc. All Rights Reserved.

Synology

Synology Inc.

® 2013 Synology Inc.

All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, mechanical, electronic, photocopying, recording, or otherwise, without prior written permission of Synology Inc., with the following exceptions: Any person is hereby authorized to store documentation on a single computer for personal use only and to print copies of documentation for personal use provided that the documentation contains Synology's copyright notice.

The Synology logo is a trademark of Synology Inc.

No licenses, express or implied, are granted with respect to any of the technology described in this document. Synology retains all intellectual property rights associated with the technology described in this document. This document is intended to assist application developers to develop applications only for Synology-labeled computers.

Every effort has been made to ensure that the information in this document is accurate. Synology is not responsible for typographical errors.

Synology Inc.

3F-3, No. 106, Chang-An W. Rd. Taipei 103, Taiwan

Synology and the Synology logo are trademarks of Synology Inc., registered in the United States and other countries.

Marvell is registered trademarks of Marvell Semiconductor, Inc. or its subsidiaries in the United States and other countries.

Freescale is registered trademarks of Freescale Semiconductor, Inc. or its subsidiaries in the United States and other countries.

Other products and company names mentioned herein are trademarks of their respective holders.

Even though Synology has reviewed this document, SYNOLOGY MAKES NO WARRANTY OR REPRESENTATION, EITHER EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT, ITS QUALITY, ACCURACY, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. AS A RESULT, THIS DOCUMENT IS PROVIDED "AS IS," AND YOU, THE READER, ARE ASSUMING THE ENTIRE RISK AS TO ITS QUALITY AND ACCURACY. IN NO EVENT WILL SYNOLOGY BE LIABLE FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES RESULTING FROM ANY DEFECT OR INACCURACY IN THIS DOCUMENT, even if advised of the possibility of such damages.

THE WARRANTY AND REMEDIES SET FORTH ABOVE ARE EXCLUSIVE AND IN LIEU OF ALL OTHERS, ORAL OR WRITTEN, EXPRESS OR IMPLIED. No Synology dealer, agent, or employee is authorized to make any modification, extension, or addition to this warranty.

Some states do not allow the exclusion or limitation of implied warranties or liability for incidental or consequential damages, so the above limitation or exclusion may not apply to you. This warranty gives you specific legal rights, and you may also have other rights which vary from state to state.

Table of Contents

Introduction 4

Lyrics Module for Audio Station 4

AUM Module Content 4

Content of INFO 5

Lyric.php 5

Methods to Add Search Results 8

Pack the AUM module 9

Apply Your Lyrics Module 9

Restricted Runtime Environment 11

Minimal Privilege 11

Restricted File Access 11

Document Revision History 12

2013-01-11 | ® 2013 Synology Inc. All Rights Reserved.

Introduction

On the Synology DiskStation with Audio Station 5.0 and onward, 3rd-party developers are allowed to develop lyrics modules for Audio Station to search lyrics from external sites.

This document is to provide guidelines to develop lyrics modules for Audio Station. Developers are encouraged to have PHP programming experience before reading this document.

Lyrics Module for Audio Station

AUM Module Content

On Audio Station 5.0 and onward, users are allowed to install lyrics modules onto Audio Station with .aum file extension. An aum module is actually a normal unix .tgz or .tar.gz archive. For example, if you have an aum module named **lyricwiki.aum**, after extracting the module, you will see the following files in the current directory:

```
DiskStation> tar zxf lyricwiki.aum

DiskStation> ls -l

-rwxr-xr-x    1 root     root           200 Nov 28 20:59 INFO

-rwxr-xr-x    1 root     root          1912 Nov 28 20:59 lyric.php
```

INFO

This is a mandatory file for module configuration. It defines the name of the module and describes how to locate the actual PHP module to search lyrics.

lyric.php

This is the actual PHP module which searches lyrics when being loaded by Audio Station.

2013-01-11 | ® 2013 Synology Inc. All Rights Reserved.

Content of INFO

The content of INFO is described by a JSON format using UTF-8 encoding and usually looks like below:

```
{
"name": "lyricwiki",
"displayname": "LyricWiki",
"description": "lyrics.wikia.com",
"version": "1.0",
"module": "lyric.php",
"type": "lyric",
"class": "LyricWiki"
}
```

Table 1. INFO file definition

Key	Description	Remarks
`name`	The unique name of the lyrics module among all Audio Station modules. The `name` must be one and only. If the `name` of the new aum conflicts with that of the existing one, the newly-added aum cannot be installed.	Mandatory
`displayname`	Texts displayed in the **Lyrics Plugin** settings of Audio Station.	Mandatory
`description`	The description shown in the **Lyrics Plugin** settings of Audio Station.	Recommended
`version`	The version number of the lyrics module.	Mandatory
`module`	The file name of the PHP module that performs lyrics search action. Usually it is `lyric.php`, but you can use other file names.	Mandatory
`type`	The module type. Currently only "lyrics" is supported.	Mandatory
`class`	The unique PHP class name within `lyric.php` among all lyrics modules. The `class` name must be one and only. If the `class` of the new aum conflicts with that of the existing one, the newly-added aum cannot be installed.	Mandatory

Lyric.php

Lyric.php is the PHP script that defines a PHP class performing lyrics search action. The class should include two mandatory methods which are used to communicate with Audio Station. Table 2, “Mandatory methods in lyric.php and arguments” explains the details of these two mandatory methods.

The content of lyric.php looks like below:

2013-01-11 | ® 2013 Synology Inc. All Rights Reserved.

```
<?php
class LyricWiki {
    private $apiUrl = 'http://lyrics.wikia.com/api.php';
    public function __construct() {
    }
    public function getLyricsList($artist, $title, $info) {
        $count = 0;

        $searchUrl = sprintf(
            "%s?artist=%s&song=%s&fmt=realjson",
            $this->apiUrl,
            urlencode($artist),
            urlencode($title)
        );

        $content = $this->getContent($searchUrl);

        $obj = json_decode($content, TRUE);
        if ($obj['lyrics'] !== 'Not found') {
            $id = $obj['lyrics'];
            $info->addTrackInfoToList(
                $obj['artist'],
                $obj['song'],
                $id,
                $obj['lyrics']
            );

            $count = 1;
        }
```

2013-01-11 | ® 2013 Synology Inc. All Rights Reserved.

```
        return $count;
    }
    public function getLyrics($id, $info) {
        $lyric = $this->getLyricById($id);
        $info->addLyrics($lyric, $id);
        return true;
    }
    private function getContent($url) {
        $result = '';
        // using curl to get content from url
        return $result;
    }
    private function getLyricById($id) {
        $lyric = '';
        // using id to get lyric content
        return $lyric;
    }
}
```

2013-01-11 | ® 2013 Synology Inc. All Rights Reserved.

Table 2. Mandatory methods in lyric.php and arguments

Method	Argument	Description
`getLyricsList`	`$artist`: string Artist name from the music info. `$title`: string Song title from the music info. `$info`: object The plugin class instance. Return value: integer Returns the number of results successfully parsed and added of the plugin.	The method is called to search lyrics by the given artist and title. Usually, this method will send an HTTP request to get lyrics search result, parse the result, and add it by using `addTrackInfoToList()`.
`getLyrics`	`$id`: string Id from the search result. `$info`: object The plugin class instance. Return value: boolean Shows whether the lyrics are successfully retrieved or not.	The method is called after results are returned from `getLyricsList()`. The `id` value filled in `getLyricsList()` will be passed to this method. Usually, this method will send an HTTP request to get lyrics content, parse it to a human readable text, and add the lyrics by using `addLyrics()`.

Methods to Add Search Results

Audio Station provides methods to add the lyrics search result. Please see Table 3, “Methods to add results into final result set” for detailed method explanation.

Table 3. Methods to add results to Audio Station’s final result set

Method	Argument	Description

`$info`
`->`
`addTrackInfoToList`
	`$artist`: string
	Artist of the song
	`$title`: string
	Title of the song
	`$id`: string
	Id of the song will be passed to `getLyrics()` later. Usually this `id` value would be set as a URL of the lyrics page.
	`$partialLyrics`: string
	Optional. Some lyrics sites provide partial lyrics for preview in the search result. This value will be displayed when you search lyrics manually via the **Song information** dialog.
		This method is used to add lyrics search result to the list.
		Usage
		`$info` -> `addTrackInfoToList` (`$artist`, `$title`, `$id`, `$partialLyric`)

`$info`
`->`
`addLyrics`
	`$lyric`: string
	Human readable text of lyrics.
	`$id`: string
	Song id passed from `getLyrics()`
		This method is used to set the parsed lyrics content.
		Usage
		`$info` -> `addLyrics` (`$lyric`, `$id`)

2013-01-11 | ® 2013 Synology Inc. All Rights Reserved.

Pack the AUM module

Module developers can use the following command to pack the .aum module:

```
linux> tar zcf mymodule.aum INFO lyric.php
```

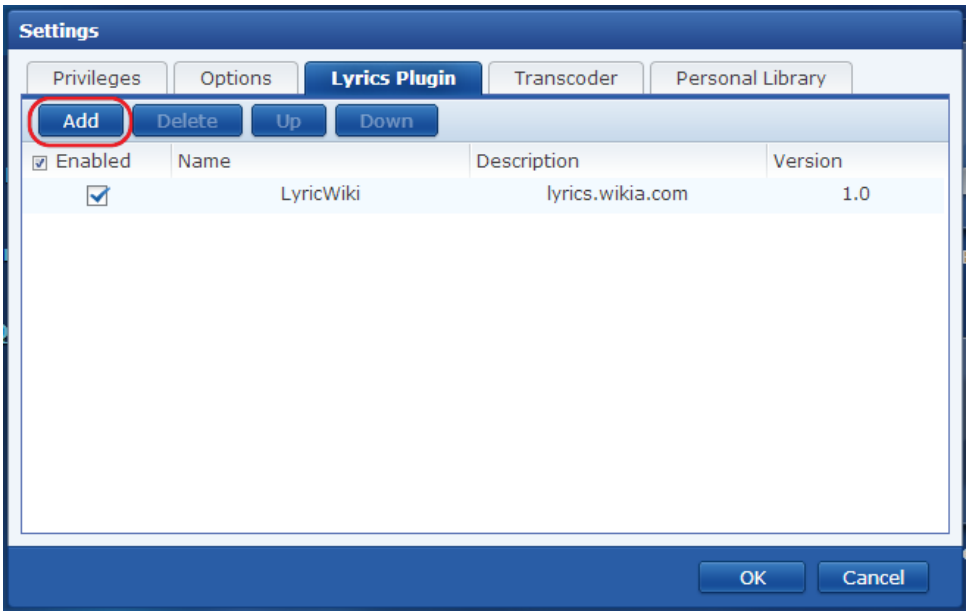
Apply Your Lyrics Module

When the .aum module is completed, follow the steps below to add it to your Audio Station.

1 Open Audio Station, and click the **Settings** button.

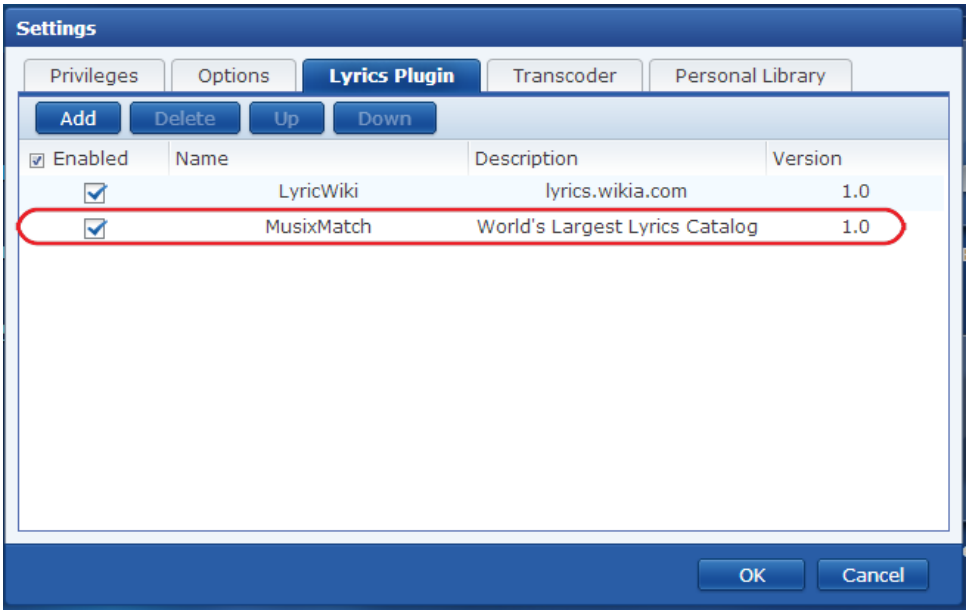
2013-01-11 | ® 2013 Synology Inc. All Rights Reserved.

2 In the window that appears, click **Lyrics Plugin** > **Add**.



3 Specify the path of your .aum file, and click **Add** to add the lyrics module to the list.

4 The lyrics module is added to the list. Now you can search lyrics with the newly-added lyrics module.



2013-01-11 | ® 2013 Synology Inc. All Rights Reserved.

Restricted Runtime Environment

PHP scripts in Audio Station search module are run within a very restricted environment to protect Synology DiskStation from malicious scripts. The following restrictions apply to any PHP scripts run by Audio Station:

Minimal Privilege

Audio Station lyrics module is run using "nobody" privilege.

Restricted File Access

Audio Station lyrics module cannot access files stored in shared folders and system configuration directories.

2013-01-11 | ® 2013 Synology Inc. All Rights Reserved.

Document Revision History

This table describes the changes to the Synology Audio Station Lyrics Module Development Guide.

Date	Description
2013-01-08	First creation of this document

2013-01-11 | ® 2013 Synology Inc. All Rights Reserved.