

Audio Lock API

Overview

Environment: SOAP

Unique API Key: AudioLock.NET -> My Account -> API Access

Returns: All functions return error strings in the 'error' key of the response array, if error is not set, then the operation was successful and the results / output is returned in the 'response' key of the response array.

Example Usage (PHP):

```
$api_key = 'XXXX';

$soap = new SoapClient('http://www.audiolock.net/soapServerv2.php?wsdl');

$response = $soap->getRecipients(
    array(
        'api_key'=> $api_key
    )
);

$recipients = unserialize($response->response);
```

Example Usage (Python):

```
#!/usr/bin/env python

from suds.xsd.doctor import Import, ImportDoctor
from suds.client import Client

imp = Import('http://www.w3.org/2001/XMLSchema',
             location='http://www.w3.org/2001/XMLSchema.xsd')

imp.filter.add('http://tempuri.org/')

client = Client('http://www.audiolock.net/soapServerv2.php?wsdl', doctor=ImportDoctor(imp))
parameters = {'api_key' : 'XXXX'}
recipients =client.service.getRecipients(parameters).response
```

Example Usage (.NET):

Simply use Visual Studio's 'Add Web Reference' feature and allow it to digest the web service for you.

API Functions

- [addAlbum](#)
- [addRecipient](#)
- [addTrack](#)
- [analyseTrack](#)
- [createGroup](#)
- [createPackage](#)
- [createTask](#)
- [deleteAlbums](#)
- [deleteGroup](#)
- [deletePackages](#)
- [deleteRecipients](#)
- [deleteTasks](#)
- [deleteTracks](#)
- [editAlbum](#)
- [editGroup](#)
- [editRecipient](#)
- [editTrack](#)
- [generateDownloadLink](#)
- [generateZipDownloadLink](#)
- [getAlbum](#)
- [getAlbums](#)
- [getAlerts](#)
- [getGroups](#)
- [getPackages](#)
- [getPackageStats](#)
- [getRecipient](#)
- [getRecipients](#)
- [getSERAlerts](#)
- [getTasks](#)
- [getTrack](#)
- [getTracks](#)
- [getWaveform](#)
- [schedulePackage](#)
- [sendPackage](#)
- [updateAlerts](#)
- [updateTaskStatus](#)

addAlbum

- Usage: Used to add an album to your account
- Return: The id of the added album, used in such functions as [editAlbum](#)
- Parameters
 - o name (required), the title of the album
 - o artist (optional), the artist of the album, e.g. "Various Artists" or "Paul van Dyk"
 - o catalog_number (optional), the catalog number of the album, e.g. VAN1234D
 - o tracks (optional), the unique identifier or array of unique identifiers of the tracks you wish to send, see [addTrack](#)
 - o artwork_file_name (optional), the file name of the artwork, 'e.g. picture001.jpg'
 - o artwork_file_contents (optional), base64 encode of the artwork file's contents
 - o release_date (optional), timestamp of the album's release date

addRecipient

- Usage: Used to add a recipient to your account
- Return: The id of the added recipient, used in such functions as [createPackage](#)
- Parameters
 - o email (required), the recipients email address
 - o name (optional), the recipient's name
 - o address (optional), the recipients postal address
 - o postcode (optional), the recipients postcode or zip
 - o group (optional), id of an existing group, see [createGroup](#).
 - o fail_on_dup (optional), boolean, default is false, if set to true it will fail if recipient already exists in your account otherwise it will return the existing recipient id as the response.

addTrack

- Usage: Used to add a track to your account
- Return: The unique identifier of the added track, used in such functions as [createPackage](#), [editTrack](#)
- Parameters
 - o artist (required)
 - o track_name (required), your reference to the track
 - o mix_name (optional), the specific mix of the track
 - o track_file_name (required), the file name of the track, e.g. 'PVD – 45rpm.mp3'
 - o track_file_contents (required), publically accessible URL of the source file, this is downloaded immediately and **should be deleted** once addTrack returns the UID of the newly added track / success.
 - o artwork_file_contents (optional), base64 encode of the artwork file's contents
 - o isrc (optional), International Standard Recording Code for the track, with or without the dashes
 - o release_date (optional), timestamp of the track's release date
 - o encode (optional), boolean, default is true, if set to false when this track is delivered it will **not** be watermarked.

analyseTrack

- Usage: Used to analyse a potential infringement
- Return: Information about the infringement and infringer if found
- Parameters
 - o track (required), the unique identifier for the track, see [addTrack](#)
 - o file_name (required), the file name of the track, e.g. 'PVD – 45rpm.mp3'

- file_contents (required), base64 encode of the the track file's contents
- comment (optional), a note to log against this analysis, perhaps where you found it or why you are analysing it.

createGroup

- Usage: Used to create a group that a recipient can be placed in.
- Return: The id of the created group, used in such functions as [addRecipient](#)
- Parameters
 - group_name (required), your reference to the group, e.g. 'House DJs'
 - add_recipients (optional), the id or an array of existing recipient ids to add to this group, see [addRecipient](#)

createPackage

- Usage: Used to create a package of tracks to send to recipients
- Return: success
- Parameters
 - recipients (optional), the id or array of ids of existing recipients for the recipients you wish to send the package to, see [addRecipient](#)
 - groups (optional), the id or array of ids of existing groups of recipients you wish to send the package to, see [createGroup](#), [addRecipient](#)
 - tracks (optional), the id or array of ids of existing tracks you wish to send with the package to, see [addTrack](#)
 - albums (optional), the id or array of ids of existing albums you wish to send with the package to, see [addAlbum](#)
 - title (optional), internal name / reference for the package
 - subject (optional), the subject of the email sent to each recipient
 - message (optional), a custom message included in the email sent to each recipient
 - stream (optional), boolean, default false, if set to true recipients will only be able to stream and listen to tracks via the link sent to them and unable to download
 - save (optional), Boolean, default false, if set to true the package will be saved not sent and you will be able to edit it later.
 - package (optional), the id of an existing unsent package that you are editing
 - attachments (optional), the url or array of urls of attachments to include in the outgoing email, e.g. press pack, pdf.
 - expires (optional), float, 7 – 30, default 7, the number of days after sending that the package links expire, only applies if save = false.

createTask

- Usage: Create task (CD)
- Return: success
- Parameters
 - recipient_id (optional), the id or array of ids of existing recipients for the recipients you wish to send the track(s) to, see [addRecipient](#)
 - group_id (optional), the id or array of ids of existing groups of recipients you wish to send the track(s) to, see [createGroup](#), [addRecipient](#)
 - album_id (required), the album_id of the album you wish to create a task for, see [addAlbum](#)
 - copies (optional), the number of copies per recipient you want to create, default 1.
 - publisher (optional), the company creating the CDs, either yourself (specify your own user ID), or MasterPiece, default.

deleteAlbums

- Usage: Delete one or more albums
- Return: success
- Parameters
 - o albums (required), the id or array of ids of existing albums, see [addAlbum](#)

deleteGroup

- Usage: Delete a group, and optionally it's recipient contents
- Return: success
- Parameters
 - o group_id (required), id of an existing group, see [createGroup](#).
 - o delete_recipients (optional), boolean, default is false, if true deletes recipients in this group

deletePackages

- Usage: Delete one or more unsent packages
- Return: success
- Parameters
 - o packages (required), the id or array of ids of unsent packages, see [createPackage](#)

deleteRecipients

- Usage: Delete one or more recipients
- Return: success
- Parameters
 - o recipient_id (required), id or array of ids of existing recipients, see [addRecipient](#)

deleteTasks

- Usage: Delete one or more tasks, this will only work for tasks "not started"
- Return: success
- Parameters
 - o tasks (required), id or array of ids of existing tasks, see [createTask](#)

deleteTracks

- Usage: Delete one or more tracks
- Return: success
- Parameters
 - o tracks (required), unique identifier or array of unique identifiers, see [addTrack](#)

editAlbum

- Usage: Update album information
- Return: success
- Parameters
 - o album_id (required), id of an existing album, see [addAlbum](#).
 - o name (optional), the title of the album
 - o artist (optional), the artist of the album, e.g. "Various Artists" or "Paul van Dyk"
 - o catalog_number (optional), the catalog number of the album, e.g. VAN1234D
 - o tracks (optional), the unique identifier or array of unique identifiers of the tracks you wish to send, see [addTrack](#)

- artwork_file_name (optional), the file name of the artwork, 'e.g. picture001.jpg'
- artwork_file_contents (optional), base64 encode of the artwork file's contents
- release_date (optional), timestamp of the album's release date

editGroup

- Usage: Update group information
- Return: success
- Parameters
 - group_id (required), id of an existing group, see [createGroup](#).
 - group_name (optional), the group's new name, whilst this is optional, since this is the only thing you can currently change on a group using this function. It would be wise to include it otherwise the call will do nothing.
 - add_recipients (optional), the id or an array of existing recipient ids to add to this group, see [addRecipient](#)
 - remove_recipients (optional), the id or an array of existing recipient ids to remove from this group, see [addRecipient](#)

editRecipient

- Usage: Update recipient information
- Return: success
- Parameters
 - recipient_id (required), id of an existing recipient, see [addRecipient](#)
 - email (optional), the recipients updated email address
 - name (optional), the recipients updated name
 - address (optional), the recipients updated postal address
 - postcode (optional), the recipients updated postcode or zip
 - group (optional), the id or an array of existing groups, see [createGroup](#), this user should be a member of, to remove from all groups send an empty array, if not specified previous memberships are preserved.

editTrack

- Usage: Update a tracks information
- Return: success
- Parameters
 - track (required), the unique identifier for the track, see [addTrack](#)
 - artist (optional), the artist of the track
 - mix_name (optional), the specific mix of the track
 - label (optional), the label for the track
 - track_name (optional), your reference to the track
 - artwork_file_contents (optional), base64 encode of the artwork file's contents
 - isrc (optional), International Standard Recording Code for the track, with or without the dashes
 - release_date (optional), timestamp of the track's release date
 - encode (optional), boolean, default is true, if set to false when this track is delivered it will **not** be watermarked.

generateDownloadLink

- Usage: Generates a download link, that when clicked streams the file to the end users browser
- Return: The download link

- Parameters
 - track (required), the unique identifier for the track, see [addTrack](#)
 - format (required), the id of the format required (see footer notes)
 - recipient_id (required), the of an existing recipients, see [addRecipient](#), alternatively a two part array containing name + email of the recipient e.g. array('John Doe', 'john.doe@google.com'). This contact will subsequently be added to your account
 - expires (optional), float, 7 – 30, default 7, the number of days after generating that the link expires

generateZipDownloadLink

- Usage: Generates a download zip link, containing specified files
- Return: The download link
- Parameters
 - tracks (required), array of unique identifiers of tracks, see [addTrack](#)
 - format (required), the id of the format required (see footer notes)
 - recipient_id (required), the of an existing recipients, see [addRecipient](#), alternatively a two part array containing name + email of the recipient e.g. array('John Doe', 'john.doe@google.com'). This contact will subsequently be added to your account
 - expires (optional), float, 7 – 30, default 7, the number of days after generating that the link expires
 - zip_name (optional), the name of the zip file when downloaded

getAlbum

- Usage: Used to get information of a particular album, see [addAlbum](#)
- Return: Serialised object of album info
- Parameters
 - album_id (required), id of an existing album, see [addAlbum](#)

getAlbums

- Usage: Used to get your albums, see [addAlbum](#)
- Return: Serialised array of your results
- Parameters
 - offset (optional), the start position of the returned results
 - limit (optional), the maximum number of results returned
 - sort (optional), what field to sort by (date_added, album_name, tracks, length)
 - direction (optional), sort direction (asc, desc)

getAlerts

- Usage: Gets a list of all the alerts in your account
- Return: Serialised array of the results
- Parameters:
 - track (optional), the unique identifier for the track, see [addTrack](#)
 - album (optional), the unique identifier for the album, see [addAlbum](#)
 - status (optional), the current status of an alert (unchecked, positive match, not a match)
 - offset (optional), the start position of the returned results
 - limit (optional), the maximum number of results returned

getGroups

- Usage: Used to get your recipient groups, see [createGroup](#)
- Return: Serialised array of your results
- Parameters
 - o None

getPackages

- Usage: Used to get your packages, see [createPackage](#)
- Return: Serialised array of your results
- Parameters
 - o match (optional), a string to match against, e.g. "PVD"
 - o recipient (optional), the id of a recipient the package was (or due to be) sent to, see [addRecipient](#)
 - o offset (optional), the start position of the returned results
 - o limit (optional), the maximum number of results returned
 - o sort (optional), what field to sort by (date_added, album_name, tracks, length)
 - o direction (optional), sort direction (asc, desc)

getPackageStats

- Usage: Used to get information about a package, see [createPackage](#)
- Return: Serialised array of your results
- Parameters
 - o package (required), the id of an existing package, see [createPackage](#)
 - o recipient (optional), the id of an existing recipient, see [addRecipient](#)

getRecipient

- Usage: Used to get information of a particular recipient, see [addRecipient](#)
- Return: Serialised object of recipient info
- Parameters
 - o recipient_id (required), id of an existing recipient, see [addRecipient](#)

getRecipients

- Usage: Gets a list of all recipients in your account
- Return: Serialised array of the results
- Parameters
 - o match (optional), a string to match against, e.g. "Richard"
 - o track (optional), the unique identifier for the track they have been sent, see [addTrack](#)
 - o offset (optional), the start position of the returned results
 - o limit (optional), the maximum number of results returned
 - o sort (optional), what field to sort by (contact_name, contact_email)
 - o direction (optional), sort direction (asc, desc)
 - o group (optional), the id of an existing group, see [createGroup](#)

getSERAlerts

- Usage: Gets all Search Engine Removal requests for your content
- Return: Serialised array of the results
- Parameters:
 - o track (optional), the unique identifier for the track, see [addTrack](#)
 - o album (optional), the unique identifier for the album, see [addAlbum](#)

- offset (optional), the start position of the returned results
- limit (optional), the maximum number of results returned

getTasks

- Usage: Used to get all tasks (CDs) currently assigned to you, see [createTask](#)
- Return: Serialised array of your results
- Parameters
 - offset (optional), the start position of the returned results
 - limit (optional), the maximum number of results returned

getTrack

- Usage: Used to get information of a particular track, see [addTrack](#)
- Return: Serialised object of track info
- Parameters
 - track_id (required), id of an existing track, see [addTrack](#)

getTracks

- Usage: Gets a list of all tracks in your account
- Return: Serialised array of the results
- Parameters
 - offset (optional), the track record to start at
 - limit (optional), the maximum number of tracks to return
 - sort (optional), what field to sort by (artist, track_name, length, date_added)
 - direction (optional), sort direction (asc, desc)

getWaveform

- Usage: Used to get a transparent waveform png of a track, see [addTrack](#)
- Return: Serialised object of track info
- Parameters
 - track_id (required), id of an existing track, see [addTrack](#)
 - color (optional), hex ref (no hash) for the colour of the waveform, default '000000'

schedulePackage

- Usage: Schedules an unsent package to be sent at a particular date/time
- Return: success
- Parameters
 - package (required), the id of an existing unsent package, see [createPackage](#)
 - send (required), a datetime of when to send package, e.g. 2011-12-25 08:00:00'

sendPackage

- Usage: Sends an unsent package
- Return: success
- Parameters
 - package (required), the id of an existing unsent package, see [createPackage](#)
 - expires (optional), float, 7 – 30, default 7, the number of days after sending that the package links expire.

updateAlerts

- Usage: Update the status of alert(s)
- Return: success
- Parameters:
 - o alert (required), the id of the alerts you wish to update, see [getAlerts](#)
 - o status (required), the new status of an alert (unchecked, positive match, not a match)
 - o dmca (optional), boolean, if status is set to 'positive match' and this is set to true a DMCA will be issued for this file.

updateTaskStatus

- Usage: Update the status of a task / CD
- Return: success
- Parameters
 - o task_id (required), id of an existing task, see [createTask](#).
 - o status (optional), the task's new status, allowed values are "not started", "in progress", "completed", "erroneous"
 - o error_message (optional), the error message if applicable for this task, if supplied this will automatically set "status" to "erroneous"

Formats

AudioLock.NET currently supports 4 output formats

- mp3 – 320kbps (format: 1)
- wav (format: 2)
- mp4 – 256kbps (format: 3)
- mp3 – 128kbps (format: 4) – recommended for streaming / previewing.

Errors

General API Errors	
100	No API key provided
101	This API function is not publically accessible
102	Required field(s) missing: ##MISSING_FIELDS##
102 Ext.	Email address required in '##FIELD##', e.g. '##FIELD##' => array('John Doe', 'john.doe@gmail.com'), alternatively specify the 'recipient_id' of an existing recipient
103	Invalid '##FIELD##' specified: '##VALUE##'
103 Ext.	The duration of this track is not supported, please upload a track longer than ##TRACK_MIN## seconds but no longer than ##TRACK_MAX## minutes
103 Ext.	Accepted formats are: ##FORMATS_ACCEPTED##
103 Ext.	Accepted statuses are: ##STATUSES_ACCEPTED##
Account Errors	
200	Sorry, your track limit of ##TRACK_LIMIT## has been reached. Please delete a track before adding another or contact us for an upgrade
201	You have insufficient delivery credits to ##ACTION_TRIED##. You need ##CREDITS_NEEDED## credits. Get More
Method Specific Errors	
300	AudioLock.NET protection not detected [analyseTrack]
301	Could not add recipient: ##SPECIFIC_ERROR## [addRecipient]
302	The specified 'album_id' has no tracks in [createTask]
303	The specified 'album_id' is too long (##ALBUM_LENGTH##). It needs to be 80 minutes or less [createTask]
304	You cannot modify the contents of this album until all CDs for this album are completed or cancelled [editAlbum]
305	This package has already been sent [sendPackage, schedulePackage]
306	You cannot change the status of an alert once a DMCA has been issued [updateAlert]
System Errors	
900	How embarassing, something went wrong
901	There was a problem storing the file you are trying to add
902	Original source file could not be found, please contact support
999	Unknown Error, please contact support