

# WebService MailUpImport

About MailUpImport	WS_MailUpImport Methods
<ul style="list-style-type: none"><li>• <a href="#">Overview</a><ul style="list-style-type: none"><li>• <a href="#">MailUpImport WSDL location</a></li><li>• <a href="#">Sample ReturnCode</a></li></ul></li><li>• <a href="#">Glossary</a></li><li>• <a href="#">MailUp API status</a><ul style="list-style-type: none"><li>• <a href="#">Manual Activation</a></li><li>• <a href="#">Activation via API (WSActivation.aspx)</a><ul style="list-style-type: none"><li>• <a href="#">Sample WSActivation response</a></li><li>• <a href="#">Response Codes (ReturnCode)</a></li></ul></li></ul></li><li>• <a href="#">Authentication</a><ul style="list-style-type: none"><li>• <a href="#">Sample WS_MailUpImport authentication</a></li><li>• <a href="#">Error codes</a></li></ul></li><li>• <a href="#">Note about Data Fields</a></li></ul>	<ul style="list-style-type: none"><li>• <a href="#">WS_MailUpImport.GetIdWsUser method</a></li><li>• <a href="#">WS_MailUpImport.GetIdUser method</a></li><li>• <a href="#">WS_MailUpImport.GetNIListsUser method</a></li><li>• <a href="#">WS_MailUpImport.GetNILists</a></li><li>• <a href="#">WSMailUpImport.CreateGroup</a></li><li>• <a href="#">WSMailUpImport.NewImportProcess</a></li><li>• <a href="#">WSMailUpImport.GetProcessDetails</a></li><li>• <a href="#">WSMailUpImport.StartProcess</a></li><li>• <a href="#">WSMailUpImport.StartImportProcesses</a></li></ul>

## Overview

The MailUpImport web service allows you to load into your MailUp account - using groups and lists - contacts coming from your database. It also allows you to retrieve details about lists, groups and the status of import operations, and also to segment the database. Among the tasks that you can perform using the MailUpImport Web Service:

- Get all the distribution Lists and the Groups contained in a given MailUp account;
- Create a new Group within an existing List;
- Upload a list of subscribers in order to create a new import process;
- Start the import process;
- Check on the import process status

Each method returns a XML string containing a "MailUp Message", which includes a "ReturnCode" and the requested information (when available).

The ReturnCode is a number, with the following values:

- < 0 = error code
- 0 = call successful

## MailUpImport WSDL location

The MailUpImport Web Services is described at the following URL:

`http://<MAILUP_CONSOLE_URL>/Services/WSMailupImport.asmx?WSDL`

... where <MAILUP\_CONSOLE\_URL> is the domain of your MailUp admin console. If you don't have that information, simply log into your MailUp admin console and look at the browser URL to locate the domain).

## Sample ReturnCode

```
<mailupMessage>
<mailupBody>
<ReturnCode>0</ReturnCode>
...
</mailupBody>
</mailupMessage>
```

### Authentication Required



Please note: if you receive a generic 500 error when attempting to use MailUpImport, the reason is typically authentication: Remember to authenticate yourself: this can be done either with an API call (see below) or manually in the admin console.

## Glossary

- **Customer:** a company/business subscribing to the MailUp service
- **Console URL / NL\_URL:** MailUp defines a dedicated URL for each Customer. For this reason, external systems must know the exact URL to which to send the request (e.g. <http://mailing.myCompany.com/Services/WSMailUpImport.aspx>). The URL is also used for links, hosted pages, and for the admin console. You can use a custom URL if your account has the [Private Labeling](#) feature turned on.
- **WS\_MailUpImport:** a Web Service designed to import recipients from a third party application, retrieve details on lists & groups, and check the status of an import process
- **MailUp API Account:** credentials used specifically with the MailUp API, different from the credentials used to log into your MailUp account. They consist of a username and a password. The API account credentials are automatically created when a new MailUp account is created for a new customer, while for existing customers they are manually created via the MailUp console under *Manage > Web Services*
- **WS:** Web Service

## MailUp API status

Once the API account has been created, the MailUp APIs must be activated in order to be able to receive requests (they are disabled by default for safety reasons): they can be activated manually using the MailUp admin console or automatically by calling a specific URL. Deactivation is also done manually through the MailUp admin console. The IP address of the application using the web service must be registered (manually in the admin console or by performing the activation call).

## Manual Activation

To manually activate and configure the MailUp API in your MailUp admin console:

1. Log into your account
2. Navigate to *Settings > Account settings > Developer's corner* and select the "Webhooks" tab
3. Select *WSMailUpImport* from the drop-down at the top
4. Set a password and store it securely: you will need it in your API calls
5. Enable the Web services
6. Add the necessary IP addresses to allow for requests coming from those locations.

## Activation via API (WSActivation.aspx)

A third party application can enable web services via a call to:

```
http://<MAILUP_CONSOLE_URL>/frontend/WSActivation.aspx?usr=<usr>&pwd=<pwd>&nl_url=<nl_url>&ws_name=<ws_name>
```

MailUp provides an ASP.NET Web Form to allow third party applications to activate (if not already active) and register (store) the IP address of the caller. The form supports the parameters described below, both in GET and in POST modes:

- **usr:** MailUp API Account username (e.g. a7132)
- **pwd:** MailUp API Account password
- **nl\_url:** Console URL (e.g abcd.sp09.com, omitting "http://" )
- **ws\_name:** Web service name (specify: "WSMailUpImport")

MailUp will register the IP address of the computer calling, and therefore authorize future calls coming from that IP address. If the caller's IP address changes, WSActivation web service must be called again.

## Sample WSActivation response

```
<mailupMessage>
<mailupBody>
<ReturnCode>0</ReturnCode>
<WS_Activation>
<WS_Name> WS_MailUpImport </WS_Name>
<DateOfRequest>2008-01-16</DateOfRequest>
<User>admin</User>
</WS_Activation>
</mailupBody>
</mailupMessage >
```

## Response Codes (ReturnCode)

Code	Description
0	request execution succeeded
-2	ws name has not been specified <sup>(1)</sup>
-4	user name has not been specified <sup>(1)</sup>

-8	password has not been specified <sup>(1)</sup>
-16	nl url has not been specified <sup>(1)</sup>
-101	mismatch between List ID and List GUID (where applicable)
-1000	unrecognized error
-1001	the account is not valid
-1002	the password is not valid
-1003	suspended account
-1004	inactive account
-1005	expired account
-1006	the web service is not enabled
-1007	the web service is not active
-1008	the web service is already active
-1009	web service activation error
-1010	IP registration error
-1011	IP is not registered
-1012	IP is registered but has the "deny access" flag
-1013	Missing authentication data
-1014	Login error: contract not signed

<sup>(1)</sup> ReturnCode can be a combination of these

## Authentication

WS\_MailUpImport checks the following parameters in order to authenticate requests:

- WS username
  - It is predefined for a specific MailUp account
  - It cannot be modified for that account
- WS password
  - It can be modified using the admin console (*Manage > Web Services*)
- IP address
  - MailUp check the IP address from which the request is coming and checks it against a list of IP addresses that have been provided

Authentication fails if:

- The MailUp API is disabled in your MailUp account (*Manage > Web Services*);
- username and/or password are incorrect;
- the IP address of the third party application has changed since the last successful access. The IP address can be manually changed in the MailUp admin console (*Manage > Web Services*).

The WS\_MailUpImport web service requires for every call an authentication via username and password; credentials must be passed in the header of the SOAP message.

Account credentials vs. API credentials



Note that the API credentials are different from the credentials you use to log into your MailUp account.

## Sample WS\_MailUpImport authentication

```

<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
<soap:Header>
<Authentication xmlns="http://tempuri.org/">
<User>a21351</User>
<Password>URY256872UYT</Password>
</Authentication>
</soap:Header>
<soap:Body>
...
</soap:Body>
</soap:Envelope>

```

## Security Information



Please Note: in the current release of the API the credentials passed in the authentication request are not encrypted.

## Error codes

ReturnCode values in case authentication fails are shown below:

Error Code	Description
-1000	unrecognized error
-1001	the account is not valid
-1002	the password is not valid
-1003	suspended account
-1004	inactive account
-1005	expired account
-1006	the web service is not enabled
-1007	the web service is not active
-1011	IP is not registered
-1012	IP is registered but has the "deny access" flag
-1013	Missing authentication data
-1014	Contract not signed

All the methods requiring *idList e listGuid* at input execute a check on the input values and return one of the following:

Error Code	Description
-100	unrecognized error
-101	verification failed
-102	list Guid format is not valid

## Note about Data Fields

The WS\_MailUpImport methods support up to 20 personal data fields. Should it be necessary to import more than 20 fields, there are other procedures that can be used. For example:

- Several consecutive calls until all the fields are imported;
- Batch import from file;
- Synchronized import from file, using the [MailUpSend.SendMessageNL](#) web service.