

Ferraro Ltd.		
		Support:
		support@trashmail.net

TrashMail.net API Reference

Created for: TrashMail.net Add-On & Script Developer
Made by: Stephan Ferraro

2012-01-22
Version: 3.0.0

Introduction

TrashMail.net disposable email addresses can be created over an API. This API is currently used by the ExtJS frontend on the main page at www.trashmail.net, as too in Firefox Add-On Version 2.5.7 (currently available at <https://ssl.trashmail.net/files/trashmailnet-2.5.7-fx.xpi>).

The idea is that all Add-Ons could use the same API as the web frontend.

The communication works over a REST API where some requests & responses are sent and received in a JSON format. All requests are **forced to use the HTTPS protocol**.

To debug there are multiple possibilities:

1. Curl is a command line tool available for UNIX as too for Windows systems which gives the possibility to try the API manually:

```
$ curl -d 'api=1&lang=en&cmd=login&fe-login-user=saf&fe-login-pass=foobar'
```

```
https://ssl.trashmail.net/
```

```
{"success":false,"msg":"Login not registered or invalid password."}
```

2. The Firefox Add-On has a hidden debug mode with which all sent and received HTTPS requests can be shown. Its possible so to understand easier the API by doing reverse engineering.

The debug mode can be enabled by the following steps:

- Type in the URL field **about:config**
- set the field **extensions.trashmail.debugMode** to **2**
- open in the Firefox menu Tools -> Error console

The communication between the Add-On and the TrashMail.net server appears now in the error console.

TrashMail.net API



Firefox Fehlerkonsole

API-Properties

TrashMail.net uses to log in a login/password authentication. For mostly all operations like getting a list of current disposable email address, its needed to be logged in. Creating disposable email addresses is possible without being logged in.

For an Add-On it would be the best to first authenticate by logging in, then after each execution of a command to verify the returned error code. If there is written in the error code response that the user has been logged out or the session has expired, then the Add-On should agin log in automatically without that the user needs to do an action. This is what is currently doing the TrashMail.net Add-On version 2.5.7.

Log into manager

Login with the user „saf“ and the password „foobar“:

HTTPS POST DATA:

```
api=1&lang=en&cmd=login&fe-login-user=saf&fe-login-pass=foobar
```

Parameters:

api: 1

lang: en OR de OR fr

Successfull response:

TrashMail.net API

```
{"success":true,"msg":{"url":"https://ssl.trashmail.net/?lang=en&cmd=manager","real_email_list":{"cont  
act@ferraro.net":true,"stephan@ferraro.net":true}}
```

Failed response:

```
{"success":false,"msg":"Error message to popup to the customer as an alert kind window."}
```

Response parameters:

success: true OR false

msg: url: This parameter could be ignored. It has as function to redirect a web browser which reads this parameter.

msg: real_email_list: Contains a list of real email addresses which are assigned to this user.

Additionally the server answers at log in with a **trashmail_session** cookie wvvhich needs to be saved to continue to communicate with the server.

Receiving disposable email address list

URL:

https://ssl.trashmail.net/?api=1&cmd=read_dea

The **trashmail_session** cookie needs to be sent with each request.

Successfull response:

```
{"success":true,"data":[{"id":"3036220","disposable_name":"apache","disposable_domain":"kurzepost.de",  
,"destination":"contact@ferraro.net","forwards":"-1","expire":"-1","website":null,"cs":"3","notify":"1"},{"id":  
"2655028","disposable_name":"apache","disposable_domain":"trashmail.at","destination":"contact@ferra  
ro.net","forwards":"-1","expire":"-1","website":null,"cs":"3","notify":"1"},
```

...

Creating disposable email address

URL:

https://ssl.trashmail.net/?api=1&cmd=create_dea

HTTPS POST DATA in JSON format:

```
{"data":{"id":-1,"disposable_name":"abplilma","disposable_domain":"trashmail.net","destination":"contact  
@ferraro.net","forwards":2,"expire":7,"website":"","cs":0,"notify":1}}
```

TrashMail.net API

Successfull response:

```
{"success":true,"message":"Created disposable address","data":{"id":6343071,"destination":"contact@ferraro.net","cs":"0","disposable_name":"abplilma","disposable_domain":"trashmail.net","expire":"7","forwards":"2","notify":"1","website":""}}
```

Deleting a disposable email address

URL:

https://ssl.trashmail.net/?api=1&cmd=destroy_dea

POST DATA in JSON:

```
{"data":"3036220"}
```

or deleting multiple addresses with one request:

```
{"data":[6343074,6343075]}
```

The ID numbers could be obtained by executing before the command „cmd=read_dea“ (in section Receiving disposable email address list).

Modifying a disposable email address

URL:

https://ssl.trashmail.net/?api=1&cmd=update_dea

POST DATA in JSON:

```
{"data":{"destination":"contact@ferraro.net","id":6343077}}
```

Explanation:

The disposable email address with the ID 6343077 has been updated to forward now emails to contact@ferraro.net.

Its possible to make multiple changes with one request:

```
{"data":[{"forwards":3,"id":6343084}, {"expire":31,"id":6343085}]}
```

Explanation:

On the first disposable email address, the number of possible forwards will be set to 3, at the second the expiration date will be set to 31 days.

TrashMail.net API

Register new user

This function does not need a **trashmail_session** cookie.

URL:

`https://ssl.trashmail.net/index.php?lang=de`

POST DATA:

`api=1&lang=en&cmd=register_account&user=test123&pass=foobar42&pass-cfrm=foobar42&email=foob
ar%40email.tld&newsletter=on`

Successfull response:

```
{"success":true}
```

Failed response::

```
{"success":false,"msg":"Sorry, the name has already been taken. Please choose another name."}
```

msg: Contains the localized error message from the server. This could be shown for example as popup in an Add-On.

Notice: The „lang“ parameter needs to be set to the correct language for all requests. In our example the language was set to „en“ (english language). Currently the following languages are 100% supported: Englisch (en), German (de), French (fr).

Error Codes

In some cases, the failed responses have a field „error_code“ which has an error code number. It helps the Add-Ons developer and the software to know what kind of error has occurred to understand why something failed. In some case its possible that the error code is 0 but the field „success“ is false. In this case its a „bug“ in the server code. The error codes are not yet fully implemented in all cases of requests.

Error code	Name	Description
0	ERROR_CODE_SUCCESS	Success
1	ERROR_CODE_UNDEFINED	Failed, not yet an error code defined.
2	ERROR_CODE_NOT_LOGGED_IN	Not yet logged in
3	ERROR_CODE_INCORRECT_USERNAME_OR_PASSWORD	Incorrect username or password
4	ERROR_CODE_LOGIN_WAIT_FOR_CONFIRMING	Username & password was correct. However the user has not yet confirmed its confirmation email after first registration.
5	ERROR_CODE_INCORRECT_INVALID_PASSWORD	Password string missing or invalid.
6	ERROR_CODE_MISSING_PARAMETERS	Some required parameters for this API call are missing.
7	ERROR_CODE_REQUIRES_TRASHMAIL_PLUS	This API requests requires TrashMail Plus for this user.
8	ERROR_CODE_INVALID_PARAMETERS	One or more parameters have an invalid value.
9	ERROR_CODE_PERMISSION_DENIED	User is not allowed to perform this operation.
10	ERROR_CODE_INVALID_SECURITY_TOKEN	Invalid key value for security token.