



# Ether Mailer

## Ether Mailer API

*Documentation of the Ether Mailer API*

Ether Mailer API Doc

Sept. 2017

# Preface

## About the API

Ether Mailer exposes some API endpoints to the public to allow integration to external clients, such as CRM, Website Frameworks, Eshop Frameworks, external Web Services and many more.

Via the API it is possible for an external client to add / edit / delete contacts, add / edit / delete contact groups, get extra contact fields, get sender profiles, and more.

The target audience of this document is mainly Software Engineers and Web Developers who are familiar with REST APIs.

The API is not final and will never be. We are constantly updating it and adding new endpoints.

# Basic Information

## REST

The Ether Mailer API is based on simple **REST** architecture. The payloads and responses are in **JSON** format only.

## HTTP Headers

All API calls require the following HTTP headers:

```
Content-Type: application/json
Accept: application/json
Authorization: Api xxxxxxxxxxxx
```

## Paginated list of content

Usually the API endpoints that respond with a list of content use pagination. For example, list of contacts and list of groups are paginated. In order to control the pagination you will have to use the parameters **limit** and **offset**. The offset starts with zero and is incremented by the limit (offset = offset + limit). For example, let's say that we want to get a list with limit 50. The offsets will be:

Page 1: offset 0, limit 50

Page 2: offset 50, limit 50

Page3: offset 100, limit 50

Page 4: offset 150, limit 50

Page 5: offset 200, limit 50

## Rate Limiting (Throttling)

In order to secure the application from abuse and DDOS attacks, we apply rate limits. The rate limit is **3 requests / second** for each URI.

# Authentication

## Authenticate your API calls

Each API call has to be authenticated. You will have to get an API KEY from Ether Mailer and use it in every API call. The API KEY has to be put in the **Authorization** header.

More specific:

1. Login to Ether Mailer and browse to Integrations > Api Keys. Direct link: <https://app.ethermailer.com/#/app/user/api-keys>
2. Create a new Api key and copy it to your application
3. Keep the API KEY safe, at the backend of your application. Do not use it in a javascript file, or any frontend file, that anybody can view!
4. In every API call add the following header:

```
Authorization: Api your_api_key
```

Example:

```
Authorization: Api 911222492809633792-e2d5f707-0363-4a71
```

# Error Responses

## Structure of error responses

When submitting POST or PUT requests to modify data (eg add a contact or rename a contact group), the system responds either with 200 OK or 404 Not Found or 400 Bad Request. In case of 400 Bad Request, a response entity is returned that explains the bad request.

### Structure of Bad Request response

```
{
  "errors": [
    {
      "objectName": "xxxxxxxx",
      "constraintName": "xxxxxxxx",
      "description": "xxxxxxxx"
    },...
  ]
}
```

**objectName** contains the offending JSON field name. **constraintName** contains the error code. **description** contains a textual description of the error code.

### Example

Here is an example when trying to add a contact with an email address that already exists in the contact list

```
{
  "errors": [
    {
      "objectName": "email",
      "constraintName": "AlreadyExists",
      "description": "A contact with the email test@test.com
already exists"
    }
  ]
}
```

# User Information

Get basic user information

## Request

```
GET https://api.ethermailer.com/user
```

## Response

```
{
  "username": "xxxxxxxx",
  "company": "xxxxxxxx",
  "email": "xxxxxxxx",
  "apiKey": "xxxxxxxx",
  "active": true,
  "verified": true,
  "createdAt": xxxxxxxx
}
```

# Email Settings

Get the list of Sender Profiles

## Request

```
GET https://api.ethermailer.com/user/email-settings/sender-profiles
```

## Response

```
{
  "items": [
    {
      "id": "xxxxxxxx",
      "name": "xxxxxxxx",
      "email": "xxxxxxxx",
      "verified": true,
      "spf": true,
      "dkim": true
    },...
  ]
}
```

# Contact Groups

## List Contact Groups

Get a paginated list of Contact Groups.

### Request

```
GET https://api.ethermailer.com/user/contact-groups/limit/{findLimit}/offset/{offset}
```

### Parameters

```
findLimit: The limit of each page. Max is 50
offset: The offset to be used. Starts with 0.
```

### Response

```
{
  "items": [
    {
      "groupId": "xxxxxxxx",
      "name": "xxxxxxxx",
      "canDelete": xxxxxxxx,
      "contactsCount": xxxxxxxx
    }, ...
  ],
  "totalItems": xxxxxxxx
}
```



## Create a Contact Group

Create a new Contact Group.

### Request

```
POST https://api.ethermailer.com/user/contact-groups
```

### Payload

```
{
  "name": "xxxxxxxx"
}
```

### Response

The response **text** contains the **ID** of the new Contact Group.

```
{
  "text": "xxxxxxxx"
}
```

## Rename a Contact Group

Rename an existing Contact Group.

### Request

```
PUT https://api.ethermailer.com/user/contact-groups/{groupId}/name
```

### Parameters

```
groupId: The group id
```

### Payload

```
{  
  "name": "xxxxxxxx"  
}
```

### Response

The response **text** contains the **ID** of the updated Contact Group.

```
{  
  "text": "xxxxxxxx"  
}
```

## Delete a Contact Group

Delete a Contact Group.

### Request

```
DELETE https://api.ethermailer.com/user/contact-groups/{groupId}
```

### Parameters

```
groupId: The group id
```

### Response

An empty response.

# Extra Contact Fields

## List Extra Contact Fields

Get a paginated list of Extra Contact Fields

### Request

```
GET https://api.ethermailer.com/user/extra-fields/limit/{findLimit}/offset/{offset}
```

### Parameters

```
findLimit: The limit of each page. Max is 50  
offset: The offset to be used. Starts with 0.
```

### Response

```
{  
  "items": [  
    {  
      "fieldId": "xxxxxxxx",  
      "name": "xxxxxxxx",  
      "required": false,  
      "type": "xxxxxxxx",  
      "options": [  
        {  
          "optionId": "xxxxxxxx",  
          "text": "xxxxxxxx"  
        }, ...  
      ]  
    }, ...  
  ],  
  "totalItems": xxxxxxxx  
}
```

### Values of “type”

```
DATE, INTEGER, BOOLEAN, TEXT, OPTIONS
```

# Contacts

## List Contacts

Get a paginated list of Contacts.

### Request

```
GET https://api.ethermailer.com/user/contact/limit/{findLimit}/offset/{offset}
```

### Parameters

```
findLimit: The limit of each page. Max is 50  
offset: The offset to be used. Starts with 0.
```

### Response

```
{  
  "items": [  
    {  
      "contactId": "xxxxxxxx",  
      "email": "xxxxxxxx",  
      "name": "xxxxxxxx",  
      "surname": "xxxxxxxx",  
      "deleted": false,  
      "deletedAtDate": 0  
    },...  
  ],  
  "totalItems": xxxxxxxx  
}
```

# Get a single Contact

## Request

```
GET https://api.ethermailer.com/user/contact/{contactId}
```

## Parameters

```
contactId: The contact id
```

## Response

```
{
  "contactId": "xxxxxxxx",
  "email": "xxxxxxxx",
  "name": "xxxxxxxx",
  "surname": "xxxxxxxx",
  "groups": [
    {
      "groupId": "xxxxxxxx",
      "name": "xxxxxxxx"
    },...
  ],
  "extraFieldValues": [
    {
      "fieldId": "xxxxxxxx",
      "name": "xxxxxxxx",
      "required": false,
      "type": "xxxxxxxx",
      "options": [
        {
          "optionId": "xxxxxxxx",
          "text": "xxxxxxxx"
        },...
      ],
      "value": "xxxxxxxx"
    },...
  ]
}
```

## Add a single Contact

### Request

```
POST https://api.ethermailer.com/user/contact
```

### Payload

```
{
  "email": "xxxxxxxx",
  "name": "xxxxxxxx",
  "surname": "xxxxxxxx",
  "groups": [
    {
      "groupId": "xxxxxxxx"
    },...
  ],
  "extraFieldValues": [
    {
      "fieldId": "xxxxxxxx",
      "value": "xxxxxxxx"
    },...
  ]
}
```

### Response

The response **text** contains the **ID** of the added Contact.

```
{
  "text": "xxxxxxxx"
}
```

# Batch import Contacts

## Request

```
POST https://api.ethermailer.com/user/contact/import
```

## Payload

```
[
  {
    "email": "xxxxxxxx",
    "name": "xxxxxxxx",
    "surname": "xxxxxxxx",
    "groups": [
      {
        "groupId": "xxxxxxxx"
      },...
    ],
    "extraFieldValues": [
      {
        "fieldId": "xxxxxxxx",
        "value": "xxxxxxxx"
      },...
    ]
  },...
]
```

## Response

```
{
  "totalCreated": xxxxxxxx,
  "totalUpdated": xxxxxxxx,
  "totalErrors": xxxxxxxx,
  "contactImportResponses": [
    {
      "contactEmail": "xxxxxxxx",
      "constraintViolationErrorList": [
        {
          "objectName": "xxxxxxxx",
          "constraintName": "xxxxxxxx",
          "description": "xxxxxxxx"
        },...
      ]
    },...
  ]
}
```



## Register with double optin a single Contact

An optin email will be sent to the given email address using the specified sender profile.

### Request

```
POST https://api.ethermailer.com/user/contact/register
```

### Payload

```
{
  "senderProfileId": "xxxxxxxx",
  "email": "xxxxxxxx",
  "name": "xxxxxxxx",
  "surname": "xxxxxxxx",
  "groups": [
    {
      "groupId": "xxxxxxxx"
    },...
  ],
  "extraFieldValues": [
    {
      "fieldId": "xxxxxxxx",
      "value": "xxxxxxxx"
    },...
  ]
}
```

### Response

```
{
  "doubleOptinEmailSent": true
}
```

# Update a single Contact

## Request

```
POST https://api.ethermailer.com/user/contact/{contactId}
```

## Parameters

```
contactId: The contact id
```

## Payload

```
{
  "email": "xxxxxxxx",
  "name": "xxxxxxxx",
  "surname": "xxxxxxxx",
  "groups": [
    {
      "groupId": "xxxxxxxx"
    },...
  ],
  "extraFieldValues": [
    {
      "fieldId": "xxxxxxxx",
      "value": "xxxxxxxx"
    },...
  ]
}
```

## Response

The response **text** contains the **ID** of the updated Contact.

```
{
  "text": "xxxxxxxx"
}
```

## Delete a list of Contacts

### Request

```
DELETE https://api.ethermailer.com/user/contact/multiple
```

### Query Parameters

A list contact ids to delete. Max is 1000 contact ids per request.

```
contactIds: List of contact ids to delete
```

### Example

```
DELETE https://api.ethermailer.com/user/contact/multiple?  
contactIds=xxxxxxxx&contactIds=xxxxxxxx&contactIds=xxxxxxxx
```

### Response

An empty response.

## Restore a list of Contacts from deletion

### Request

```
PUT https://api.ethermailer.com/user/contact/multiple/restore
```

### Query Parameters

A list contact ids to restore. Max is 1000 contact ids per request.

```
contactIds: List of contact ids to restore
```

### Example

```
PUT https://api.ethermailer.com/user/contact/multiple/restore?  
contactIds=xxxxxxxx&contactIds=xxxxxxxx&contactIds=xxxxxxxx
```

### Response

An empty response.