# SWIFT EMAIL VERIFIER API DOCUMENTATION V5.0

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT US FOR A COPY.

IN NO EVENT SHALL SWIFT OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING,WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF SWIFT OR ITS AFFILIATTES HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

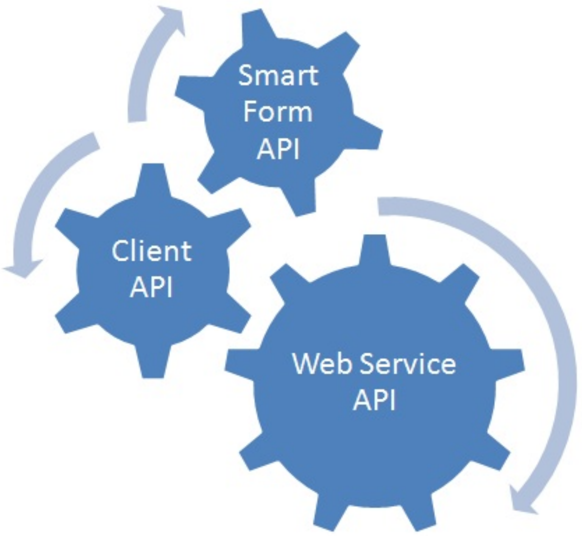
SWIFT EMAIL VERIFIER: API Documentation V5.0
2021 Swift Software & Services.

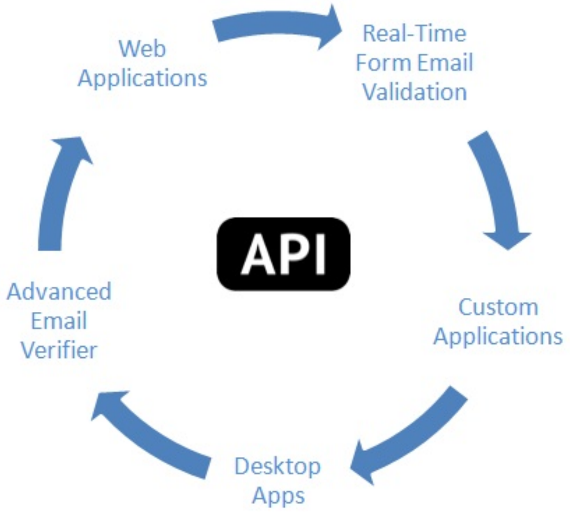
# Contents

# HOW IT WORKS

Swift email validation API is a RESTful API that allows consumers and developers to verify or scrub email lists thereby enabling users to maintain a clean mailing list and reduce bounces which can significantly improve sender reputation. Our email validation service performs a complete and full verification by checking if an email address actually exist (actual mailbox existence) and can receive email (deliverable) at that specific moment in time.

Where we are unable to conclusively verify that an email address actually exists AND can receive email at that instant it was verified, we will respond back with an "Unknown" status. Unknown email address status does not attract any charges.





## OVERVIEW OF EMAIL VALIDATION API

Our real-time email validation APIs allows you to check if an email address really exists and if it can receive messages. For every email address checked, a specific status is presented which tells you if the email address is “valid” or “invalid” or whether it is damaging or undesirable (“bad”) for your email marketing including over 10 status codes for investigating the reason of a specific email validation failure.

### What is Checked by Email Validation API (In progressive order):

✓ **Syntax, Typo and Fake Email Pattern Match:** This checks for syntax and for typo errors including fake pattern

✓ **Mail Server Existence Check:** This checks the availability of the email address domain using DNS MX records

✓ **Mail Existence Check:** This checks if the email address really exists and can receive email

✓ **Catch-All Domain Email Check:** This checks if the email domain will receive all of the email messages addressed to that domain, even if their addresses do not exist in the mail server.

✓ **Disposable Email Address Check:** This checks if the email is provided by a known Disposable Email Address (DEA) provider such as Mailinator, 10MinuteMail, GuerrillaMail and about 2000 more.

## EMAIL VALIDATION STATUS AND STATUS CODES

Our email validation API is a web service API and uses status codes to indicate API success or errors. The status codes provide further information regarding the result of the validation and indicate why the validation of an email may have failed.

The API defines the validity of an email address as follows using only 3 statuses and each of these statuses have their corresponding status codes.

| **Status** | **Description/Meaning** |
|---|---|
| **Valid** | Mailbox exists and not handled by Catch-all domains or known to be a DEA |
| Invalid | Mailbox does not exists |
| Unknown | Mailbox could not be verified or is determined to be handled by a Catch-all domain, DEA, Greylisted,, SMTP/Mailbox  timeouts, Temporary mailbox unavailability. Specific reason for failure is provided in the status codes. |

Each of these Statuses is linked to the following status Codes:

| **Status Codes** | **Meaning** |
|---|---|
| **Mailbox Exists and Active** | The email was successfully verified as Valid |
| **Bad** | The email was determined to be potentially toxic or damaging to your email campaign such as known publicly harvested email, bots, fakes, spamtraps, honeypots, Typos, Curse words, all known disposable email domains etc.<br>Note: This status code does not mean that the email address is invalid or cannot receive messages. |
| **Error** | This status code indicates that an error has occurred in the request such as an email that is not syntactically correct such as john@@yahoo.com |
| **Mailbox Not Found** | This failure means that the mailbox for the provided email address does not exist. |
| **InvalidToken** | An invalid API key was used. Please check the API key and make sure it is correct |
| **NoMoreQueries** | The allocated # of queries or requests for the API key has been exhausted. |
| **InternalError** | There was an unexpected error on our server. |
| **InternalDBError** | This indicates that there was a database connection error from our API server |
| **Invalid JSON Response** | This error indicates that an error was received in the output of the results during the API call. |

## RESOURCE URL

The API URL request is a simple GET request with 3 parameters as follows:

- **API Key:** This is the unique key to authenticate to the API. Each key has a pre-allocated requests quota

- **API Server IP:** This is the API server IP that was provided to you.

- **Email:** The email address to verify

All resources are exposed according to the following URL structure:

***http://<IP>/CheckEmail/<api_key>/<email_address>***

To ensure data privacy and secure the key from eavesdropping, clients may pass the API request over HTTPS. Please note that however, HTTPS is currently not supported on the API server.

## HOW TO USE THE EMAIL VALIDATION/SCRUBBING API:

To make an API call to the Email Validation API, you will need:

1. Your Email Validation API Key (APIKey)
2. The API server IP (IPs)
3. The Email Address to check (Email_Address)

**Note:** The API server IP(s) will be provided in the welcome email after an API key order is made

The API call must be made using the URL format below:

***http://<IP>/CheckEmail/<api_key>/<email_address>***

**Parameters:**

•***Email_Address****:* This is the email address to be verified.

Example> johndoe@yahoo.com

•***APIKey***: This is your API Key to authenticate to the API

•***IP***: This is the API server IP

**Example:**

http://198.58.101.227/CheckEmail /t3dxuewqjuv/johndoe@yahoo.com

## AUTHENTICATION

Clients must authenticate to the API by providing their API key. Care must be taken to secure the key from unauthorized access. It is your responsibility to keep your API key secure at all times and ensure that unauthorized users do not have access to it.

## HTTP STATUS CODES

Swift Email Verifier uses conventional HTTP status codes to indicate success or failure of a request. Clients must examine the HTTP status code of server replies before attempting to interpret their content. Below are the possible HTTP status codes which the API may return:

| **Status Code** | **Meaning** |
|---|---|
| 200 OK | The request was well formed, contains no error and the reply contains the result |
| 404 Bad Request/Not Found | The request URL is invalid or not available |
| 503 Service Unavailable | The API server is temporarily down. Please try the alternate URL or request for support |

Note: Unless something goes wrong, the HTTP status code will always be 200 regardless weather the email is valid or invalid.

## REPRESENTATIONS

The data interchange format for the API is JSON. When validating server replies, clients should ensure that all expected fields are present in the responses. Each response will contain the following data.

- **Email Address:** This is the email address that was verified
- **Status**: This is the validation status for the email as provided here
- **StatusCode**: This is the corresponding status code for the email status as provided here

Some sample request responses are as follows:

```
  {"Address":"lamboo7000@yahoo.com","Status":"Valid","StatusCode":"Mailbox
Exists and Active"}


  {"Address":"lamboo7000@yahoyyo.com","Status":"Invalid","StatusCode":"Bad"}

  {"Address":"anonyproz@gmail.com","Status":"Valid","StatusCode":"Mailbox Exists and Active"}

  {"Address":"john@gmail.com","Status":"Invalid","StatusCode":"Bad"}


  {"Address":"service@gmail.com","Status":"Invalid","StatusCode":"Bad"}
```

## API KEY REQUEST QUOTA CHECK

By default, each API key is issued with a specific limit of allowed queries or requests. If your query limit or key quota has been exhausted, you can top-up the API key with additional quota. To get information about your current request quota, you can make the following API call in the formats below:

**Email Validation API Remaining Quota Check URL:** https:// *server_host/path_to_resource* /{APIKey}/TokenInfo

**Sample response:** {"ActualQueryCount":995,"InitalQueryCount":1000,"Status":"Ok","Token":"50000"}

To learn more about our pricing and to top-up your API key quota, please go to our website link below:

https://webemailverifier.com/#prices

## GETTING SUPPORT

If you have any questions or encounter any issues while implementing the API, you can reach our support center by visiting: https://www.gondorland.com/supportsuite/

To request for information about your current request quota, you can send us an email or support ticket. Please make sure you provide your API key when requesting for the information

To place order for API keys, please go to our website link below:

https://www.gondorland.com/member/signup.php

To learn more about our pricing and to top-up your API key quota, please go to our website link below:

https://www.webemailverifier.com

# Known Limitations with Email Validation Technology

Due to the fact that our email verification engine employs DNS and SMTP protocol functionalities to perform email address validations without actually sending any email message, many SMTP servers will not cooperate or sometimes give false positive answers as an anti-spam protection.

Therefore it is not possible to guarantee a 100% email validation success rate or accuracy due to multiple factors beyond our control such as the remote SMTP server inbound connection policy, reputation of the incoming connecting IP and many other factors. In particular, some email service providers will not cooperate with email validation techniques because emails cannot be validated without sending a real message.

While most major ISPs such as Gmail, Hotmail, AOL etc give bounce replies (rejects invalid emails) on the wire (real-time) during the SMTP session when validating the emails, there are some that do not give real-time bounce replies but will send the bounce messages via email. Therefore, for such ISPs, it is impossible to accurately detect if an email address from such ISPs is valid or not because you will not be able verify if the emails are valid until they bounce. In addition, most ISPs will accept to deliver messages to all recipients whether they are valid or not instead of giving bounce replies during the SMTP session. One notable example of such ISPs is YAHOO.

When our email validation system is unable to conclusively verify the status of an email address, it will return an “Unknown” result. However, we do not charge credits for unknown results. Thus users are free re-check the status of the unknown emails at any time since it’s possible that some of the unknown results may be temporary.

However, please be aware that some emails which return unknown results could be valid. Examples of such emails which are flagged unknown by our email validation system and which may be valid are:

- Disposable Email Addresses from email address providers, like Mailinator, 10MinuteMail, GuerrillaMail,etc
- Catch-all email addresses
- Temporarily Unavailable emails (Greylisting)
- Soft bounces

The document aims to highlight all the known issues or limitations that a user of our email validation service may experience and provides the necessary recommendations or possible solutions to them. These issues are as follows:

## False Positive Results

In rare cases, some email addresses marked as Valid by the verifier may not be Valid in reality. This false positive is caused by a strict anti-spam technology employed by some ISPs notably free email providers. In most cases, deactivated/suspended/expired emails are marked as Valid by the verifier because these ISPs regards these emails as valid since they are “still available on their databases” even though such emails are not active and cannot receive emails. Hence when you attempt to send your campaigns to such emails, they will bounce because the emails are not active.

## Temporary SMTP Server Issues

Since a majority of the unknown results from our email validation system are caused by temporary issues (Soft Bounces) such as SMTP server timeout or downtime, Greylisting, Mailbox size Exceeds quota, temporary mailbox suspension/deactivation, and temporary blockings due to IP reputation, it is strongly recommend to re-validate the entire unknown list again at a later time. We do not recommend deleting the unknown list immediately after running your verification job. Chances are that previous emails which previously tested as unknown could test valid after re-validating the list.

## Greylisting

Our email validation service comes with a powerful automatic Greylisting detection and handling otherwise known as email temporary mailbox unavailability which is technology that reduces spam by rejecting initial email delivery attempts. Greylisting works by returning a temporary failure response ("Temporarily Unavailable") to the first attempt to deliver an email, but accepts it on the second attempt. Thus every proper email server will attempt to redeliver a message after a temporary failure response.

Since greylisting is a temporary issue from the remote SMTP server, we do not recommend deleting the unknown results due to greylisting immediately after running your verification job. Chances are that previous emails which previously tested as unknown could test valid after re-validating the list. It is strongly recommend to re-validate the entire unknown list again at a later time.

## Catchall Email Domains

Our email validation service has the capability to automatically detect Catch-all emails which is a mailbox on an email domain that will receive all of the email messages addressed to that domain, even if their addresses do not exist in the mail server. Thus emails that return “ServerisCatchAll" status may be VALID or INVALID. The emails could not be conclusively determined as VALID because the email server under test accepts fake, non-existent, email addresses; therefore the provided email address MAY be inexistent too. In some cases, these Catch-all domains are now setup by ISPs and ESPs as Catch-all Spam Trap domains specifically targeted to catch spammers using Dictionary Spam Attacks.

Therefore, it is impossible to verify conclusively whether the email address is good or not. You won't know definitively until the message bounce. We recommend bounce processing be used to take care of unknown emails.

## Non-cooperating Email Service Provider (ESP)

Some unknown results can also result from the inability to verify the emails by simulating a message sending to the recipient email server because the recipient email server requires that a REAL message is sent. Thus, it is impossible to verify whether the address is good or not. You won't know definitively until the message bounce because these mail servers won't cooperate or cannot be checked without sending a real message to them. For such email service providers who would not cooperate with email validation techniques by simulating a real message, we recommend the use of bounce processing. Alternatively, our scrubbing API service can be used for email addresses from such ESPs.