



SMTPMaster Email Logs API Documentation

The SMTPMaster Email Logs API allows you to retrieve individual email delivery records from your SMTPMaster account.

You can use the API to view delivered messages, hard bounces, soft bounces, sender and recipient information, timestamps, and email subjects. The API also supports date filtering, sender and recipient filtering, category filtering, and cursor-based pagination

Documentation for Email-Logs

This guide explains how to authenticate, enable access, read response bodies, paginate logs, and fetch the APIs from Python, Go, PHP, Laravel, and Java.

Quick Summary- Email Logs:

GET /user-email-logs - cursor-paginated message rows filtered by category, date/date range, sender, and recipient.

Base URL :

Use your production backend domain as the base URL.

- <https://api.smtpmaster.com/api/user-email-logs>

Method

GET

Authentication:

Use the API key in the request header.

X-API-Key: YOUR_API_KEY

Accept:application/json

Important: API access requires an active eligible plan. Expired, upcoming, inactive, suspended, and cancelled plans are rejected.

Where Can You Find the API Key?

After logging in to <https://app.smtpmaster.com/>, navigate to the **Email Config** section. There, you will find the Sending API option, where you can access the API key required for integration.

Note: If the Sending API option is not visible in your account, please contact the SMTPMaster Support Team for assistance.

Endpoint : Get User Email Logs

GET /user-email-logs

Returns cursor-paginated message logs.

Query parameters

- date - conditional. Single date in YYYY-MM-DD format. Use this alone; do not send start_date or end_date.
- start_date and end_date - conditional date range. end_date must be after or equal to start_date.
- from - optional sender filter. Full email matches exactly; domain text like example.com matches sender domain.
- to - optional recipient filter. Full email matches exactly; domain text like gmail.com matches recipient domain.

Category values

ALL = Delivered + hard bounce + soft bounce

D = Delivered only

HB = Hard bounce only

SB = Soft bounce only

Request Examples

GET /user-email-logs?category=ALL&date=2026-08-11&limit=50

GET /user-email-logs?category=HB&start_date=2026-08-01&end_date=2026-08-11

GET /user-email-logs?category=D&date=2026-08-11&from=example.com&to=gmail.com

GET /user-email-logs?category=SB&date=2026-08-11&cursor=NEXT_CURSOR_VALUE

Successful response body

```
{
  "success": true,
  "user_id": 123,
  "has_more": true,
  "next_cursor": "eyJpZCI6MTAwMjN9",
  "data": [
    {
    },
    {
      "status": "Delivered",
      "sender": "sender@example.com",
      "recipient": "client@gmail.com",
      "time": "2026-08-11 10:15:22",
      "subject": "Welcome to SMTPMaster"
    },
    {
      "status": "Hard Bounce",
      "sender": "sender@example.com",
      "recipient": "bad@example.net",
      "time": "2026-08-11 10:17:44",
      "subject": "Campaign update"
    }
  ]
}
```

Pagination Pattern for Logs

GET /user-email-logs?category=ALL&date=2026-08-11&limit=100 GET /user-email-logs?
category=ALL&date=2026-08-11&limit=100&cursor=NEXT_CURSOR_VALUE

Fetch Examples

Python

```
import requests
```

```
BASE_URL = "https://api.smtpmaster.com/api/"
```

```
API_KEY = "YOUR_API_KEY"
```

```
headers = {  
    "X-API-Key": API_KEY,  
    "Accept": "application/json"  
}
```

```
logs = requests.get(  
    f"{BASE_URL}/user-email-logs",  
    headers=headers,  
    params={  
        "category": "ALL",  
        "date": "2026-08-11",  
        "limit": 50  
    },  
    timeout=30,  
)
```

```
print(logs.status_code)
```

```
print(logs.json())
```

**SMTP
Master**

Go

```
package main
```

```
import (  
    "fmt"  
    "io"  
    "net/http"  
)
```

```
func main() {
```

```
    url := "https://api.smtpmaster.com/api/user-email-logs?category=ALL&date=2026-08-11&limit=50"
```

```
    req, err := http.NewRequest("GET", url, nil)  
    if err != nil {  
        panic(err)  
    }
```

```
    req.Header.Set("X-API-Key", "YOUR_API_KEY")  
    req.Header.Set("Accept", "application/json")
```

```
    res, err := http.DefaultClient.Do(req)  
    if err != nil {  
        panic(err)  
    }
```

```
    defer res.Body.Close()
```

```
    body, err := io.ReadAll(res.Body)  
    if err != nil {  
        panic(err)  
    }
```

```
    fmt.Println("Status Code:", res.StatusCode)  
    fmt.Println(string(body))
```

PHP

```
<?php
```

```
$url = "https://api.smtpmaster.com/api/user-email-logs?category=HB&date=2026-08-11&limit=50";
```

```
$ch = curl_init($url);
```

```
curl_setopt_array($ch, [  
    CURLOPT_RETURNTRANSFER => true,  
    CURLOPT_HTTPHEADER => [  
        "Accept: application/json",  
        "X-API-Key: YOUR_API_KEY",  
    ],  
]);
```

```
$response = curl_exec($ch);  
$status = curl_getinfo($ch, CURLINFO_HTTP_CODE);
```

```
curl_close($ch);
```

```
echo "Status Code: " . $status . PHP_EOL;  
echo $response . PHP_EOL;
```



Laravel

The original mixed documentation contains a Laravel example for Email Stats rather than Email Logs. The following uses the same documented authentication pattern with the Email Logs endpoint.

<?php

```
use Illuminate\Support\Facades\Http;
```

```
$response = Http::withHeaders([
    "Accept" => "application/json",
    "X-API-Key" => config("services.smtpmaster.api_key"),
])->get("https://api.smtpmaster.com/api/user-email-logs", [
    "category" => "ALL",
    "date" => "2026-08-11",
    "limit" => 50,
]);
```

```
if ($response->successful()) {
```

```
    $logs = $response->json("data");
    $hasMore = $response->json("has_more");
    $nextCursor = $response->json("next_cursor");
```

```
} else {
```

```
    logger()->warning("SMTPMaster Email Logs API failed", [
        "status" => $response->status(),
        "body" => $response->json(),
    ]);
```

```
}
```

Java

```
import java.net.URI;
import java.net.http.HttpClient;
import java.net.http.HttpRequest;
import java.net.http.HttpResponse;

public class SmtplibMasterEmailLogsExample {

    public static void main(String[] args) throws Exception {

        HttpRequest request = HttpRequest.newBuilder()
            .uri(URI.create(
                "https://api.smtplibmaster.com/api/user-email-logs?category=ALL&date=2026-08-11&limit=50"
            ))
            .header("Accept", "application/json")
            .header("X-API-Key", "YOUR_API_KEY")
            .GET()
            .build();

        HttpResponse<String> response =
            HttpClient.newHttpClient().send(
                request,
                HttpResponse.BodyHandlers.ofString()
            );

        System.out.println("Status Code: " + response.statusCode());
        System.out.println(response.body());
    }
}
```

Common Error Responses

The Email Stats API may return the following HTTP errors.

401 – Invalid API Credentials

```
{
  "success": false,
  "message": "Invalid API credentials"
}
```

Check that the Sending API key is correct and included in the request header.

403 – Email Logs API Disabled

```
{
  "success": false,
  "message": "User email logs API is disabled by admin"
}
```

The Email Stats API is not currently enabled for the account. Contact SMTPMaster Support for assistance.

403 – Active Paid Plan Required

```
{
  "success": false,
  "message": "Active paid plan required"
}
```

The account does not currently meet the plan requirement for API access.

422 – Validation Failed

```
{
  "success": false,
  "message": "Validation failed",
  "errors": {
    ...
  }
}
```

This usually means one or more request parameters are missing or invalid. Check the category, date/date-range format, limit, cursor, sender, and recipient parameters.

Dates must use:

YYYY-MM-DD

When requesting a range, both start_date and end_date must be provided, and end_date must be greater than or equal to start_date.

500 – Unexpected Error

```
{
  "success": false,
  "message": "Unexpected error occurred",
  "error": "..."}
}
```

An unexpected server-side error occurred. Try the request again or contact SMTPMaster Support if the problem continues.

Important Notes

- Use the Sending API Key in every API request.
- Always include Accept: application/json.
- Use only the /user-email-stats endpoint for Email Stats.
- Use date when requesting statistics for one specific day.
- Use start_date and end_date when requesting statistics for a date range.
- Do not use date together with start_date or end_date.
- API access requires the applicable active plan and Email Stats API permission.
- Never expose your Sending API key in frontend or client-side code.

For More Information Visit at
<https://smtpmaster.com>

Contact : +91 74396 80211
Email : info@smtpmaster.com

