



Email Sending API Documentation

This documentation explains how to send plain text and HTML emails using the hosted Email Sending API. The API supports single recipients, multiple recipients, dynamic variables, HTML templates, and secure API authentication.

1. Overview

This API allows applications to send emails through our hosted SMTP infrastructure. Supported email types:

- Plain text emails
- HTML emails
- Multi-recipient emails
- Dynamic placeholder emails The API automatically handles SMTP delivery using the SMTP account configured by the administrator.

2. Endpoint

Method: POST

Endpoint: <https://api.smtpmaster.com/api/send-email>

Required Headers

HeaderDescription: Content-Typeapplication/json and X-API-Key API key

X-API-Key: You can generate this from <https://app.smtpmaster.com/email-config>

3. Authentication

Every request must include both authentication headers:

- X-API-Key If either value is invalid, the API returns HTTP 401
- Unauthorized.

4. Request Fields

Field Required Description: from_email :Yes (Sender email address) , from_name:No (Sender display name) ,to_email:Yes (Single email or array of emails) ,cc:No (Array of CC recipients) ,bcc:No (Array of BCC recipients), subject:Yes (Email subject line), body:No (Plain text or auto-detected HTML), html_body:No (HTML email body), plain_body:No (Plain text fallback), variables:No (Dynamic placeholder values).

5. Plain Text Example

```
{
  "from_email": "noreply@example.com",
  "from_name": "Example App",
  "to_email": "user@example.com",
  "subject": "Welcome",
  "plain_body": "Hello and welcome to Example App"
}
```

6. HTML Example

```
{
  "from_email": "noreply@example.com",
  "from_name": "Example App",
  "to_email": "user@example.com",
  "subject": "Welcome",
  "html_body": "<h1>Welcome</h1><p>Your account is ready.</p>",
  "plain_body": "Welcome - Your account is ready."
}
```

7. Multiple Recipients Example

```
{
  "from_email": "team@example.com",
  "to_email": [
    "user1@example.com",
    "user2@example.com"
  ],
  "cc": [
    "manager@example.com"
  ],
  "bcc": [
    "audit@example.com"
  ],
  "subject": "Team Update",
  "plain_body": "Monthly report attached."
}
```

8. Dynamic Variables

```
{  
  "subject": "Hello {{first_name}}",  
  "html_body": "<h1>Hello {{first_name}}</h1>",  
  "variables": {  
    "first_name": "YourFirstName"  
  }  
}
```

9. Important JSON Rule For HTML

Incorrect:

```
{  
  "html_body": "<h1 class='title'>Hello</h1>"  
}
```

Correct:

```
{  
  "html_body": "<h1 class='\"title\"'>Hello</h1>"  
}
```

Recommended:

- JavaScript → JSON.stringify()
- PHP → json_encode()

10. Response Examples

Success:

```
{  
  "message": "Email sent successfully."  
}
```

401 Unauthorized:

```
{  
  "message": "Invalid mail username or API key."  
}
```

422 Validation Error:

```
{  
  "message": "Validation failed.",  
  "errors": {  
    "to_email": ["The to_email field is required."]   
  }  
}
```

502 Send Failed:

```
{  
  "message": "Email could not be sent."  
}
```

11. Best Practices

- Use `html_body` for HTML emails.
- Always send `plain_body` as a fallback.
- Use JSON encoders to safely escape HTML.
- Keep your API key secret.
- Use BCC for hidden recipients.
- Never send SMTP passwords to the API.
- Use only allowed sender addresses.

12. Language Examples

TypeScript / Node.js

```
const response = await fetch("https://api.smtpmaster.com/api/send-email", {
  method: "POST",
  headers: {
    "Content-Type": "application/json",
    "X- API-Key": "your_api_key"
  },
  body: JSON.stringify({
    from_email: "noreply@example.com", to_email: "user@example.com",
    subject: "Welcome",
    html_body: "<h1>Hello</h1>", plain_body: "Hello"
  })
});
```

Java

```
HttpRequest request = HttpRequest.newBuilder()
.uri(URI.create("https://api.smtpmaster.com/api/send-email"))
.header("Content-Type", "application/json") .header("X-API-Key",
"your_api_key") .POST(HttpRequest.BodyPublishers.ofString("""
{
"from_email":"noreply@example.com", "to_email":"user@example.com",
"subject":"Welcome", "plain_body":"Hello"
}

```

PHP

```
$payload = [
    "from_email" => "noreply@example.com", "to_email" =>
    "user@example.com", "subject" => "Welcome",
    "html_body" => "<h1>Hello</h1>", "plain_body" => "Hello"
];
$ch = curl_init("https://api.smtpmaster.com/api/send-email");
curl_setopt($ch, CURLOPT_POST, true); curl_setopt($ch,
CURLOPT_HTTPHEADER, [ "Content-Type: application/json",
    "X-API-Key: your_api_key"
]);
curl_setopt($ch, CURLOPT_POSTFIELDS, json_encode($payload));
curl_exec($ch);
```

Ruby

```
require "net/http" require "json"
uri= URI("https://api.smtpmaster.com/api/send-email")
payload = {
    from_email: "noreply@example.com", to_email: "user@example.com",
    subject: "Welcome",
    plain_body: "Hello"
}
http = Net::HTTP.new(uri.host, uri.port) http.use_ssl = true
request = Net::HTTP::Post.new(uri) request["Content-Type"] =
"application/json" request["X-API- Key"] = "your_api_key"
request.body = payload.to_json
response = http.request(request) puts response.body
"")) .build();
```

For More Information Visit at : smtpmaster.com

Contact : +91 74396 80211

Email : info@smtpmaster.com

