sinch

Sinch E-Mail 365, SMTP specification

Version 3.0 – February 2022

Revision History

Version	Date	Description
1	19.05.19	First Draft
2	20.08.25	Updated to include bounce API information
3	22.02.08	URL Change- SMTP Specifications

Table of Tables

Table of Contents

1 Service Description for SMTP E-Mail

With SMTP E-Mail service emails can be sent directly from business applications. This is done by connecting the Customer's infrastructure to an SMTP server via established interfaces.

- High availability and system stability
- Local processing
- No internal email infrastructure needed
- Connection via SMTP-Adapter
- Encryption via Enforced TLS per sender domain

1.1 Basic configuration

Basic configuration consists of access data for an endpoint to an SMTP server in a data center. The setup includes a sender domain/address, default job parameters, IP routing, SPF record, and DKIM encryption. The account is activated upon complete and successful setup and an interface description is provided.

1.1.1 Technical Specifications

Communication Interface (job request)

- Transfer Protocol: SMTP
- Security: TLS

Authentication

- Basic Auth (Username / Password)
- Registered Sender Domain
- Registered Sender Address
- SPF / DKIM Validation Tag.

2 Parameters required for SMTP Account setup

- SMTP server name / URL :
- Port: 25
- Security : TLS
- Authentication : basic
- Account credentials
  - Username
  - Password
- Valid and registered Sender Domain & address

NOTE   Account credentials and URL to be provided to the customer during the onboarding step

---

3 SMTP Commands:

- EHLO: To start the conversation with server followed by (server name or IP)
- STARTTLS: to start the Transport Layer Security
- AUTH: AUTH command is used to authenticate the client to the server, the service supports PLAIN
- MAIL FROM: This command begins the operations; the sender states the source email address in the "From" field and actually starts the email transfer
- RCPT TO: It identifies the recipient of the email; if there are more than one, the command is simply repeated address by address.
- DATA: This command begins the transfer of email content; server response is 354 which provides the permission to start transmission.
- SIZE: This command informs the remote server about the estimated size (in terms of bytes) of the attached email. It can also be used to report the maximum size of a message to be accepted by the server
- QUIT: It terminates the SMTP conversation.

4 Get notification Status by id

This will pull all the events for a single notification ID. If multiple recipients are passed in a request; this will pull events for all recipients for that request/ notification ID.

NOTE Data is only available for 7 days. If the notification was sent more than 7 days back, an empty object will be returned.

4.1 Sample Resource URL:

GET

https://eu.email.sdi.sinch.com/v1/caas_email12345/notifications/id/10000000902077 2

4.2 Resource Information

Request Authentication   Authorization: Basic

Response Content-type   application/json

Response Object   Notification

4.3 URL Parameters

Class	Field	Required	Description
	accountId	Required	Customer account identifier
	notificationId	Required	notificationId returned in POST Notification response
	recipient	Optional	Email address in format

			"email:<xxx>@<domain.com>"

Table 1: GET URL parameters

4.4 Examples

4.4.1 Sample API call to query status of notification

GET

https://eu.email.sdi.sinch.com/v1/caas_email12345/notifications/id/100000009020772

Authorization Basic sdfjlkerwyvnnewrlj==

4.4.2 Example: Response to status of notification

```
"Notification": {
        "notificationId":"100000009020772",
        "status" : [
                {
                "statusCode": "CAAS_RECEIVED",
                "statusText": "Successful",
                "channel" "email",
                "recipient": "email:name1@customer.com"]",
                "timestamp": "2015-03-08 14:58:30.252"
                },
                {
                "statusCode": "CAAS_SENT",
                "statusText": "Successful",
                "channel" "email",
                "recipient": "email:name1@customer.com"]",
                "timestamp": "2015-03-08 14:59:10.252"
                },
```

```
            {
            "statusCode": "CH_RECEIVED",
            "statusText": "Successful",
            "channel" "email",
            "recipient": "email:name1@customer.com"]",

            "timestamp": "2015-03-08 14:59:20.252"

            },
            {
            "statusCode": "CH_SENT",
            "statusText": "Successful",
            "channel" "email",
            "recipient": "email:name1@customer.com"]",

            "timestamp": "2015-03-08 14:59:30.252"

            },
        ]
    }
}
```

4.4.3 Sample API call to query status of notification with specific recipient

GET

https://eu.email.sdi.sinch.com/v1/caas_email12345/notifications/id/100000009020772?recipient=email:name1@customer.com?recipient=email:name1@customer.com

This gives the same result as above. This would be useful if you sent to multiple recipients.

5 SMTP reply codes

SMTP servers respond to commands with a variety of numerical reply codes in the format of x.y.z where:

- X indicates whether the command was good, bad, or incomplete.
- Y indicates the kind of response that was sent.
- Z provides additional information about the command

When a response is received by the server that opened the connection, it can tell whether the remote server accepted the command and is ready for the next one, or if an error occurred.

The first digit (X) is particularly important to understand because it indicates the success or failure of the command that was sent. Here are its possible values, and their meanings.

Reply code	Meaning
2.y.z	The command that was sent was successfully completed on the remote server. The remote server is ready for the next command.
3.y.z	The command was accepted but the remote server needs more information before the operation can be completed. The sending server needs to send a new command with the needed information.
4.y.z	The command wasn't accepted by the remote server for a reason that might be temporary. The sending server should try to connect again later to see if the remote server can successfully accept the command. The sending server will continue to retry the connection until either a successful connection is completed (indicated by a 2.y.z code) or fails permanently (indicated by a 5.y.z code). An example of a temporary error is low storage space on the remote

Reply code	Meaning
	server. Once more space is made available, the remote server should be able to successfully accept the command.
5.y.z	The command wasn't accepted by the remote server for a reason that is isn't recoverable. The sending server won't retry the connection and will send a non-delivery report back to the user who sent the message. An example of an unrecoverable error is a message that's sent to an email address that doesn't exist.

Table 2: SMTP Reply codes