SocketLabs

SMTP Quick Start Guide

Your Complete Guide to SMTP

An Email **Best Practices** Whitepaper

Why Read This Guide

Simple Mail Transfer Protocol (SMTP) is the most widely supported standard for sending email. Most development frameworks support SMTP without the need for any extra libraries, making SMTP the quickest way to start sending transactional email from your web application.

Despite the fact that setting up your application to send emails using SMTP couldn’t be any easier, there are still many pitfalls that you could run into along the way. These issues range from SMTP connection problems to deliverability issues — all of which will cost you time, money, and even degrade the experience of your application.

This guide will show you how to get started with SMTP while avoiding some common pitfalls along the way.

What You Will Learn:

- What the term SMTP means: We'll explain the differences between the SMTP protocol, an SMTP Server, and an SMTP Relay Service
- The differences between SMTP, IMAP, and POP3
- Which port SMTP uses – and we'll also show you what to do if your main port is blocked
- How to test SMTP connection issues – we'll even point you to a free tool that will help you test your SMTP connection in a matter of minutes
- The 5 questions to ask yourself before you DIY or use an open source SMTP solution
- All about Gmail's SMTP service: including Gmail's SMTP limits, when to use it, and when to avoid it
- The 5-point checklist for choosing the right SMTP relay service
- SMTP & email delivery 101 – do you know when an email is actually considered delivered? (The answer may surprise you!)
- How to know when SMTP transmissions are failing before customers tell you that they never received your emails
- 21 SMTP response codes that will give you a view into the performance of your SMTP server

Ready? Let's do this.

Contents

CHAPTER 1

What Is SMTP

[A Crash Course]

When it comes to SMTP, it's important to understand the basics — specifically the differences between SMTP, an SMTP server, and an SMTP relay service.

What is an SMTP?

SMTP stands for **Simple Mail Transfer Protocol**. It's responsible for delivering outgoing email and it does not accept incoming email. Think of SMTP as the protocol for email sending, which is made possible with the use of an SMTP Server.

What is an SMTP Server?

A simple way to think about an SMTP server is that it's a machine, or collection of machines, built to send, receive, and/or relay messages between email senders and receivers. You can either build and manage an SMTP server yourself, or you can use an SMTP service to do the heavy lifting for you.

What is an SMTP Service?

An SMTP relay service (also referred to as an SMTP provider) is a service that helps senders deliver transactional and bulk email, by routing the email through a trusted 3rd party, like SocketLabs. **The SMTP relay service** provides all of the underlying technology and expertise to help businesses deliver email over SMTP.

Use an SMTP relay service when you need to:

- Enable email sending capabilities for an app or website
- Simplify email headaches, get your mail flowing again, and improve email deliverability (your ability to reach the inbox)
- Send email from hardware like a printer, scanner, fax machine, or IoT device
- Overcome ISP limitations — for example, when your ISP (i.e. Gmail) has put a cap on how many SMTP relays it can conduct per day

CHAPTER 2

SMTP, IMAP, & POP3: The Differences Between Each

SMTP is just one of three email protocols — there's also IMAP and POP3. Each of which plays an important role in sending and retrieving email. Let's take a closer look at each.

SMTP

As explained earlier, SMTP stands for **Simple Mail Transfer Protocol**. It's responsible for delivering outgoing email. It does not accept incoming email.

IMAP

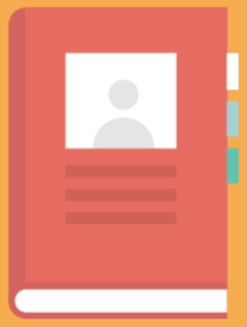
IMAP (Internet Access Message Protocol) is an email protocol that deals with managing and retrieving email messages from the receiving server. Since IMAP deals with message retrieval, you will not be able to use the IMAP protocol to send email. Instead, IMAP is used for receiving messages.

POP3

In addition to IMAP, there's also another protocol for receiving email — it's called POP3.

POP stands for Post Office Protocol.

And the number three stands for "version 3," which is the latest version and the most widely used — hence the term "POP3."



Continue reading for SMTP, IMAP, and POP3 Use Cases

LEARN MORE ABOUT PORTS

CHAPTER 3

What's An SMTP Port? What Port Should I Use?

Let’s start with the basics — what’s an SMTP port?

SMTP relies on domain names and internet addresses to know where to send messages.

You’re probably already familiar with the textual version of an internet address like www.socketlabs.com.

And you may also be aware that the easy-to-remember textual address translates to an IP address like 104.20.20.29 — which also includes a port number. The port numbers appear after the IP address with a colon separating the two. For example, if your IP address is 104.20.20.29 and you see an entry for 104.20.20.29:2869, it means port 2869 is open.

An “SMTP port” refers to the specific part of the internet address that’s used to transfer email.

So, now that you have a better understanding of what an SMTP port is, let’s discuss what ports you should be using.

What SMTP port should I use?

This is a common question for anyone configuring an app or mail system to relay email. After all, you might see information that tells you to connect to ports 25, 225, 587, 465, or some other port number.

At SocketLabs, we use port 25 as the standard port for SMTP transmission. This is because port 25 is the most widely used port for SMTP relaying.

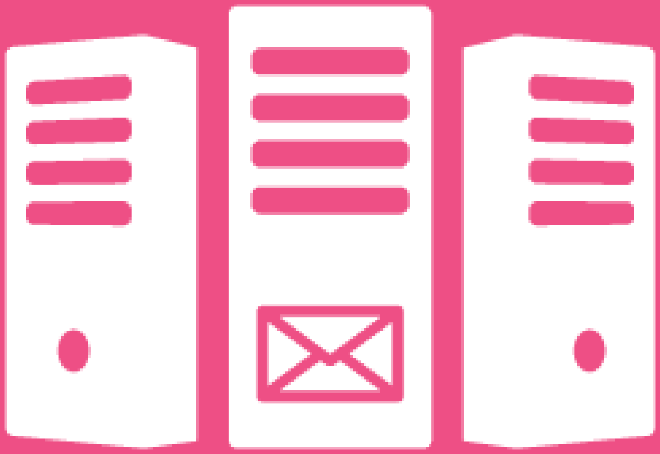
What should I do if port 25 is blocked?

Since port 25 is the standard and most widely used port, some ISPs will block this port to prevent abuse from malicious senders. If you discover that Port 25 is blocked, then you can switch to an alternative port like 2525, 587, or 465.

Read More About Ports »

CHAPTER 4

Testing for Common SMTP Connection Issues



If you’re experiencing problems connecting to your SMTP server, then knowing how to test your SMTP relay connection will allow you to quickly diagnose the issue.

By testing your SMTP connection, you can answer questions like:

- Is my SMTP server up and running?
- Why is my SMTP server not sending?
- Is something blocking communication with my server? Possibly a firewall?
- What caused the connection to my SMTP server to fail?
- And much more...

Before we show you how to run an SMTP check, let's first discuss some common causes of SMTP connection issues.

1) Antivirus Software

A first step in troubleshooting an SMTP connection issue is to take a look at the antivirus software. While this security feature is extremely important, it can sometimes block email and cause connection issues. To see if your antivirus software is the issue, try disabling the software and connect to your SMTP server again. If this works, then you know that you'll need to make changes to your antivirus software.

2) Firewalls

In addition to antivirus software, firewalls are another common cause of SMTP connection issues. If you think that your firewall is stopping your mail from sending via SMTP, then try turning it off and reconnecting. If that works, it means that you need to update your firewall settings.

3) ISP Restrictions or Blocks

If you find that your SMTP connection issue is not the result of your antivirus software or firewall, then the next step is to ensure that your Internet Service Provider allows for SMTP transmission on the SMTP port that you're using. For example, if your ISP is blocking port 25, then try a different port, such as 2525 or 587. At SocketLabs, we support ports 25, 2525, and 587.

Deeper Issues: Use This Free Tool

ⓘ If you're still having trouble connecting to SMTP, then your connection issue could be the result of a deeper issue with your SMTP server.

When all else fails, it's safe to assume that your SMTP connectivity issue is the result of a deeper problem — most likely caused by a compatibility issue with your application.

At SocketLabs, we created a **Free SMTP Server Connection Diagnostic Tool** to enable you to troubleshoot SMTP server issues.

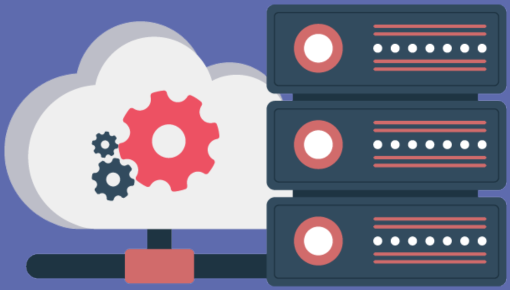
Learn more about testing your SMTP Server Connection Issues »

Download Our Free SMTP Server Connection Diagnostic Tool

DOWNLOAD NOW

CHAPTER 5

SMTP Options: Should You Build Your Own SMTP Server?



Is Building Your Own Server Worth It?

When it comes to building an SMTP server, there are a couple of routes that you can take.

You can use a hosted SMTP relay service that provides scalable email relaying capabilities right out of the box.

Or, you can build on top of an open source SMTP solution like Postfix SMTP. But, here's the big question…

… Is building your own SMTP server really worth it?

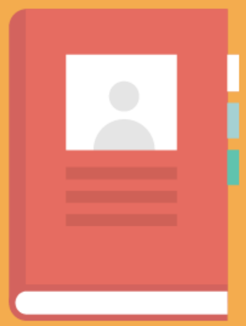
Read on to Find Out

Our answer:

While building and managing your own SMTP server is certainly acceptable in many cases (in fact SocketLabs has helped companies do this for well over a decade) it may not always be the best option — especially when you're looking to maximize your deliverability over the long term.

If you want to setup an SMTP server to send emails, then here are the **top 5 questions** that you should ask yourself before you DIY.

1. How much time will I need to invest into SMTP server setup?
2. Do I have the resources and knowledge to setup my own email infrastructure?
3. Will setting up my own SMTP server positively or negatively impact my deliverability?
4. What's my sending volume?
5. How do I configure my infrastructure to relay mail?



Dive Deeper Into the 5 Questions You Should Ask Before Building Your Own SMTP Server:

LEARN MORE

CHAPTER 6

Thinking About Using Gmail’s SMTP Solution? Know the Limits

When it comes to sending over SMTP, you have a few different options:

- Build an SMTP server using an open source SMTP server solution like Postfix SMTP (discussed above)
- Quickly setup an SMTP relay server using a **relay service** that provides a hosted SMTP service (easy to maintain but usually not free after you reach a certain sending volume)
- Or you can setup your own SMTP server using Gmail (easy to setup and usually free)

Most developers and IT pros turn to Gmail's SMTP server because it's easy to configure. While Gmail's SMTP service is certainly an acceptable solution for low volume senders, it comes with the following **limitations**:

1. **Daily sending limits:** You're restricted to 500 emails per day for a free Gmail account and 1,000 emails per day if you're paying for G-suite.
2. **Reporting and analytics:** Gmail lacks even basic analytics to help you optimize the delivery of your outgoing email.
3. **Limitations of Gmail's API:** Gmail doesn't offer complex bulk sending, mail merge, and does not provide programmatic access to APIs for tracking, reporting, and parsing email messages.

If you're building an application and need a quick way to test sending capabilities, then it could make sense to use Gmail's SMTP server. However, we recommend switching to an optimized SMTP relay service like SocketLabs, before your app goes live. This way you can confidently send your time-critical transactional emails knowing that email issues will not degrade your app's user experience.

CHAPTER 7

Need an SMTP Relay Service?

[Use This 5 Point Checklist]

If you're convinced that using a hosted SMTP relay service is the way to go, then here's a 5-Point Checklist to help you choose the right ESP the first time around.

Save Time

The last thing that any IT pro or developer wants is to spend hours looking for the right SMTP relay service, only to change providers at a later date because a configuration isn't working as desired. Or worse, because the outbound mail stops flowing.

Avoid Email Deliverability Issues

When choosing an SMTP relay, it's important to understand that the delivery of your email over SMTP is directly impacted by your service provider.

For example, if your provider has reputation or infrastructure problems, then your ability to reach inboxes could be affected.

Maintain Your Professional Reputation

Choosing an SMTP service is not only an important technical decision, but it's a professional decision as well.

A reliable ESP will provide you with at least these five things:

1. Ease of integration into your application or website
2. High-touch support from in-house email experts
3. Features built around authentication, security, deliverability, and integration. See a full list of features here »
4. High deliverability with built-in analytics to help you monitor and optimize your email streams
5. Competitive pricing

See the full checklist »

CHAPTER 8

Measuring SMTP: When Is An Email Actually Considered Delivered?

Have you ever wondered what it actually means for an email to be considered delivered? For example: Is an email delivered when it reaches the inbox? Or, is an email delivered when it is opened and read?

When Is an Email Delivered? (3 Common Answers)

If you ask someone when an email is considered 'delivered,' don't be surprised if you receive a number of different responses. Here are three of the most common answers to this question:

1) The email is accepted by the receiving mail server:

In this case, the message leaves the sender's mail client and is successfully received by the recipient's mail server. However, there's still more work to be done on behalf of the receiving mail server to determine how to properly handle the message.

2) The message is placed in the appropriate mailbox on the receiving mail server:

In this situation, the message was not only received by the mail server, but the message was also placed in the appropriate mailbox.

3) The recipient opens and reads the message:

In this case the recipient actually opens and reads the message.

Since the meaning of "delivered" varies depending upon the situation, context, and email service provider (ESP), all three answers are correct. However, from a technical perspective, there is only one correct answer.

The Technically Correct Answer for When an Email Is Delivered:

Technically speaking, a message is considered delivered when the sender's mail server receives a **"250 OK"** SMTP response at the end of an SMTP transmission.

CHAPTER 9

What is a Failed Message & How to Monitor Email Delivery Errors

When it comes to SMTP monitoring and delivery errors, it can be hard to identify the meaning of each delivery error. This is because there are a lot of them. And to make things more challenging, each email service provider maintains their own set of error codes, many of which require additional research to understand their meaning — **here's a list of 21 Common SMTP Error Codes.**

With that said, before we dive deeper into email delivery errors, it's important to understand what a failed message is in the first place.

What Is a Failed Message? (Synchronous Vs Asynchronous Failures)

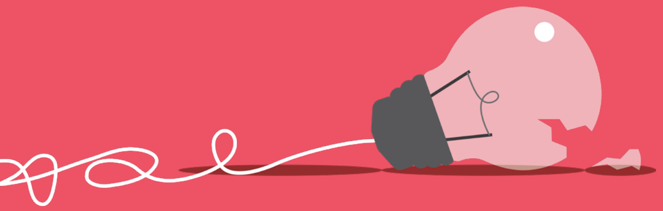
A failed message simply means that the message did not reach the intended recipient.

There are two types of email delivery failures that you may encounter when sending outbound email. They are synchronous and asynchronous failures.

Synchronous failures occur when the remote mail server rejects the message. This happens during the initial conversation between the **SMTP server** and the receiving mail server.

An asynchronous failure occurs when the receiving remote mail server accepts the message and then later returns it by sending an NDR (Non Delivery Report) to the return path of the message. Since the receiving mail server initially accepted the message, it's easy to believe that the message was successfully delivered. It is not until we later receive a failure notification from the remote server that we know that the outbound message has failed. This is known as an **email bounce**, which can be either hard or soft.

Now let's dive deeper into SMTP failures by taking a look at three common delivery error messages and what they are telling you about a failed SMTP transmission.



Delivery Error Messages

1. Permanent Errors

A permanent error, or permanent failure, is exactly how it sounds — the message was returned by the recipient mail server and no further attempt will be made to deliver the message.

The most common reason for a permanent error is because the domain does not exist, or because the recipient is unknown. This is also referred to as a hard bounce.

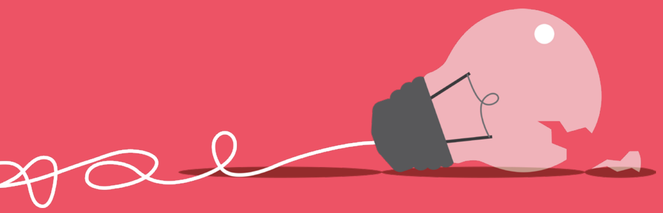
2. Temporary Errors

A temporary delivery error means that the message was not accepted by the receiving mailbox, but the message will remain in queue to be retried at a later time.

There are many reasons why an SMTP transmission could fail due to a temporary error. For example, you could receive a temporary error if the recipient's mailbox is full or if the Mail Transfer Agent (MTA) is unable to connect to the MX records for the receiving domain.

3. Email Greylisting

Greylisting is a temporary rejection of a message that forces the sender to attempt to send the message again in the near future. This prevents bots and spammers from flooding an inbox with unwanted mail because legitimate senders will attempt to resend the message, while spammers and bots will not make a second attempt. For this reason, a greylisting most commonly occurs on new IP addresses, mail systems that send low volumes, and for senders that have a low sender reputation.



Why Monitor Failed Messages

The problem with failures, especially an asynchronous hard bounce email, is that constant and repeated attempts to deliver hard bounces will damage your sender reputation, which will negatively impact your ability to reach the inbox. Messages that land in spam will not only hurt your reputation and cost you potential revenue, but will result in user experience and customer service issues as well.

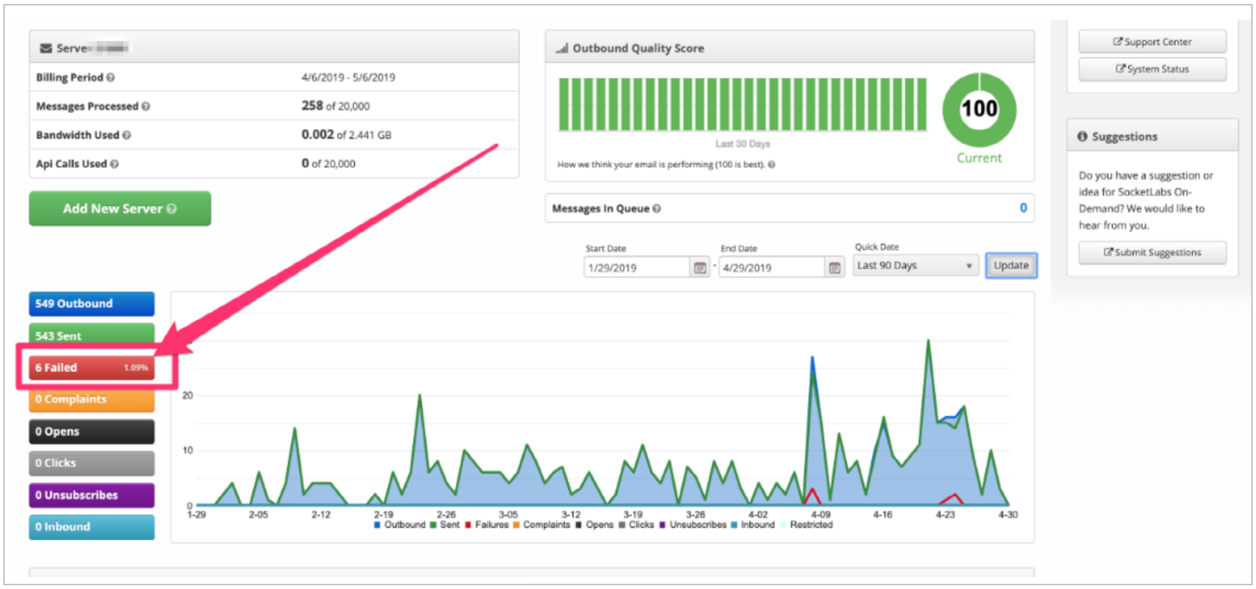
How to Monitor Failures?

How do you know when messages are failing?

Monitoring and reporting upon message failures is a complex task, especially for asynchronous failures. This is because the SMTP protocol requires asynchronous failures to be sent to the return path of the message as an NDR (Non Delivery Report), rather than the original sending server. After this happens, a complex process needs to run to gather, parse, and analyze NDRs.

If you're sending transactional email from a custom-built or open-source SMTP server, then you would need to build a very complicated system to capture and process failures. However, many email services like SocketLabs will automatically do this for you.

One of the major benefits of using an SMTP Relay Service is that we provide a central location where you can quickly see the status of your messages, including those that failed delivery.



Serve[redacted]

Billing Period	4/6/2019 - 5/6/2019
Messages Processed	258 of 20,000
Bandwidth Used	0.002 of 2.441 GB
Api Calls Used	0 of 20,000

Outbound Quality Score

How we think your email is performing (100 is best).

Messages In Queue 0

Start Date 1/29/2019 - End Date 4/29/2019 Quick Date Last 90 Days

Suggestions

Do you have a suggestion or idea for SocketLabs On-Demand? We would like to hear from you.

549 Outbound

543 Sent

6 Failed 1.09%

0 Complaints

0 Opens

0 Clicks

0 Unsubscribes

0 Inbound

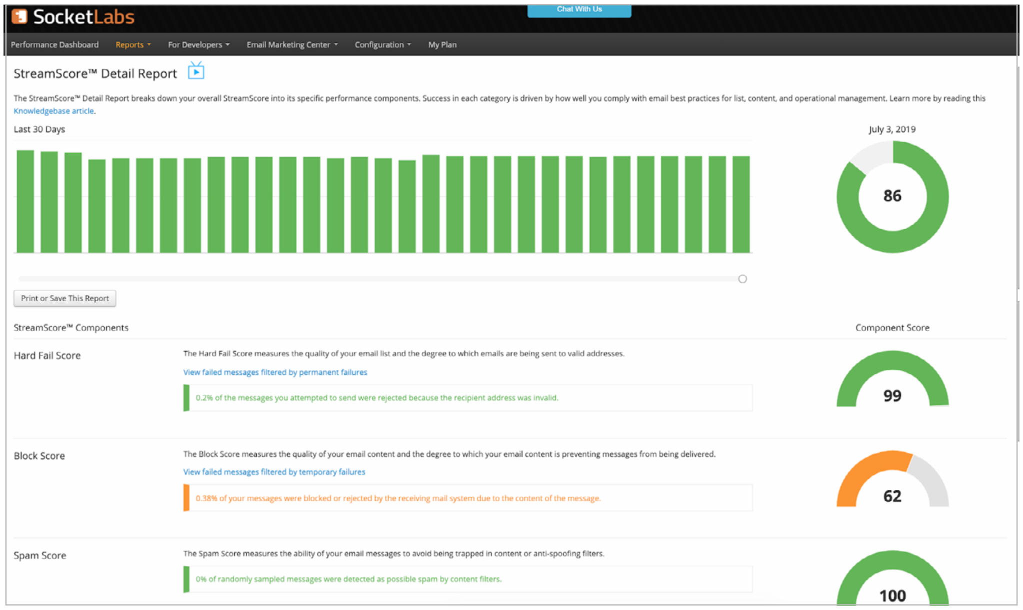
Monitoring The Impact of Failures On Your Reputation

If you’re concerned about failures damaging your reputation, then you can use our StreamScore™ feature to monitor the performance of your outbound SMTP mail.

StreamScore™ is a measure of how well your content is performing in the eyes of mailbox providers. The StreamScore™ Detail Report breaks down your overall StreamScore into its specific performance components. Success in each category is driven by how well you comply with email best practices for list, content, and operational management.

With StreamScore™ you can drill down into your results and see a breakdown of your email streams based on specific performing metrics including:

- Hard failures
- Blocks
- Complaints
- Spam Scoring
- User Engagement



StreamScore™ Detail Report

The StreamScore™ Detail Report breaks down your overall StreamScore into its specific performance components. Success in each category is driven by how well you comply with email best practices for list, content, and operational management. Learn more by reading this Knowledgebase article.

StreamScore™ Components — Component Score

Hard Fail Score — 99

The Hard Fail Score measures the quality of your email list and the degree to which emails are being sent to valid addresses.

View failed messages filtered by permanent failures

0.2% of the messages you attempted to send were rejected because the recipient address was invalid.

Block Score — 62

The Block Score measures the quality of your email content and the degree to which your email content is preventing messages from being delivered.

View failed messages filtered by temporary failures

0.38% of your messages were blocked or rejected by the receiving mail system due to the content of the message.

Spam Score — 100

The Spam Score measures the ability of your email messages to avoid being trapped in content or anti-spoofing filters.

0% of randomly sampled messages were detected as possible spam by content filters.

A drop in your StreamScore™ may indicate that your reputation is being negatively impacted as a result of message failures. If this is the case, then the issue will be clearly identified in your StreamScore™ Detail Report.

You can then dive deeper into your failed message reports to determine why emails are failing so you can isolate and fix delivery issues.

Learn More About SMTP Monitoring & Fixing Message Failures

VIEW MORE

CHAPTER 10

21 SMTP Response Codes That You Need To Know

Response Codes: What You Need to Know

In the previous section, we talked about SMTP monitoring, specifically failed messages.

When troubleshooting message failures you may come across SMTP response codes. By knowing the meaning of the SMTP response codes, you'll be able fix email delivery issues.

Here are some of the most important SMTP response codes that you need to know:

- **550 —** The requested command failed because the user's mailbox was unavailable. For example, it was not found, or the command was rejected for policy reasons.
- In addition, SMTP response code 550 is also commonly used to indicate additional instances of permanent failures. For example, "550 The mail server detected your message as spam and has prevented delivery."
- **551 —** The recipient is not local to the server. The server then gives a forwarding address to try. This is commonly used as a strategy for spam prevention.
- **552 —** The action was aborted due to exceeded storage allocation. This is usually due to the recipient's mail server being too full. This could either be because the recipient doesn't check their email, or in some more extreme situations, the recipient is a victim of **mail bombing**.
- **553 —** The command was aborted because the mailbox name is invalid. In this case, the mailbox was unable to verify the email address. Check to ensure that all the email addresses that you're sending to are correct.
- **554 —** Sorry, your message cannot be delivered. This mailbox is disabled. If you receive SMTP code 554, then this is just a normal invalid address response. Check the email address and try again.

Continue Reading to See all 21 of the Most Common SMTP Response Codes ››

Conclusion

Phew! We put A TON of work into this guide and we hope you enjoyed it. As you can see, SMTP is a complex topic and there are many pitfalls that you can run into along the way. While we barely scratched the surface of the SMTP topic, it's our hope that this guide can put you on the right path to getting started with and optimizing your SMTP mail streams.

Have SMTP Questions?

Simply reach out to us at https://www.socketlabs.com and one of our email experts will get back to you. Here's how to reach us: Live Chat / support@socketlabs.com

Try Our SocketLabs' SMTP Relay Service

Learn more about the SocketLabs SMTP Relay Service »

Test our SMTP Relay Service with a Free Trial of SocketLabs »

See our SMTP relay service plans & pricing »

TRY SOCKETLABS FOR FREE TODAY!

SocketLabs