

How to test email server (SMTP) using Telnet command

As a Linux administrator, you may find it necessary to troubleshoot or test your Simple Mail Transfer Protocol (SMTP) server to ensure it is working properly.

As a Linux administrator, you may find yourself needing to troubleshoot or test your Simple Mail Transfer Protocol (SMTP) server to make sure it is working properly. One of the most efficient ways to do this is to use the telnet command.

Telnet allows you to manually connect to an SMTP server, send commands, and view live responses from the server, helping you diagnose problems or verify configurations. This guide will walk you through the process of testing an SMTP server using the telnet command.

Step 1: Open a Telnet session

To start testing the SMTP server, open a Terminal on your Linux system and start a Telnet session to the SMTP server.

```
telnet smtp.example.com 25
```

Replace *smtp.example.com* with the domain or IP address of the SMTP server and 25 with the appropriate port number if different.

Step 2: Understand the SMTP server greeting

After connecting, the SMTP server will respond with a greeting message, usually displaying the server's hostname and indicating that the server is ready to receive commands. This message will look something like this:

```
220 smtp.example.com ESMTPE postfix
```

If you receive this message, it means the connection was successful and the SMTP server is ready to accept commands.

Step 3: Start an SMTP conversation

EHLO Command : Start the conversation by identifying yourself to the SMTP server. Use the EHLO command followed by the domain or any placeholder like localhost.

```
EHLO localhost
```

The server will respond with a list of supported extensions and features:

```
250-smtp.example.com Hello localhost [127.0.0.1] 250-SIZE 10485760 250-PIPELINING
```

HELO Command : Alternatively, you can use the older HELO command instead of EHLO. However, EHLO is recommended as it indicates support for extended features.

```
HELO localhost
```

The response is usually simpler:

```
250 smtp.example.com Hello localhost [127.0.0.1]
```

Step 4: Specify the sender

Next, specify the sender's email address using the MAIL FROM command:

```
MAIL FROM:
```

The server will confirm the command:

```
250 2.1.0 Ok
```

Step 5: Specify recipients

Now, specify the recipient's email address using the RCPT TO command:

```
RCPT TO:
```

If the recipient is accepted, you will see:

```
250 2.1.5 Ok
```

If the recipient is invalid or not accepted by the server, you may see an error like the following:

```
550 5.1.1
      : Recipient address rejected: User unknown in local recipient table
```

Step 6: Send message data

To send email data, use the DATA command:

```
DATA
```

The server will respond with a message indicating that it is ready to receive data:

```
354 End data with .
```

Now you can enter the body of your email. Start with the subject:

```
Subject: Test Email From: sender@example.com To: recipient@example.com This is a
```

After writing the email, end the data entry by typing a period (.) on a new line and pressing **Enter** :

```
.
```

The server will respond:

```
250 2.0.0 Ok: queued as ABC123DEF456
```

This indicates that the email has been accepted and queued for sending.

Step 7: End session

To end an SMTP session, use the QUIT command:

```
QUIT
```

The server will close the connection:

```
221 2.0.0 Bye
```

Step 8: Explaining common SMTP responses

During your interaction with the SMTP server, you will receive various response codes. Here are some common response codes:

1. 220: Server is ready.
2. 250: Requested mail action, completed.
3. 354: Start typing letter; end with .
4. 421: Service unavailable, channel closed.
5. 450: Requested mail action not performed: Mailbox unavailable.
6. 550: Requested action not taken: Mailbox unavailable.

Testing an SMTP server using Telnet is a valuable skill for any Linux administrator. It allows you to manually send commands and observe the server's response, making it easier to diagnose problems such as connection problems, incorrect SMTP configuration, or problems sending and receiving email. While Telnet is useful for testing, remember that it transmits data in plain text, so it should only be used in a secure, controlled environment. For production use, always ensure that your SMTP communications are encrypted using STARTTLS or other secure protocols.

You finished reading the article "**How to test email server (SMTP) using Telnet command**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.