|  |
| --- |
| **Task 25: Send a sample Email in PHP using SES**  **GOAL: send an email to the verified address using aws-SES with php**  1. Verify the recipient address inside simple email service[SES].  2. Get your smpt credentials ( from SES console)  3. Install php and other required dependencies in your machine  4. Try to send mail by creating and executing a php file [ refer aws docs for the information about the file and its contents ] |

**Amazon Simple Email Service (AWS SES)**

Amazon Simple Email Service (AWS SES) is a highly scalable and cost-effective email-sending service designed for developers. It allows you to send transactional and marketing emails reliably and efficiently.

Using AWS SES with PHP, you can leverage the SMTP protocol to send emails from your application. SMTP (Simple Mail Transfer Protocol) is a standard communication protocol for email transmission, and AWS SES supports this protocol, allowing seamless integration with various programming languages, including PHP.

### **Why Use AWS SES?**

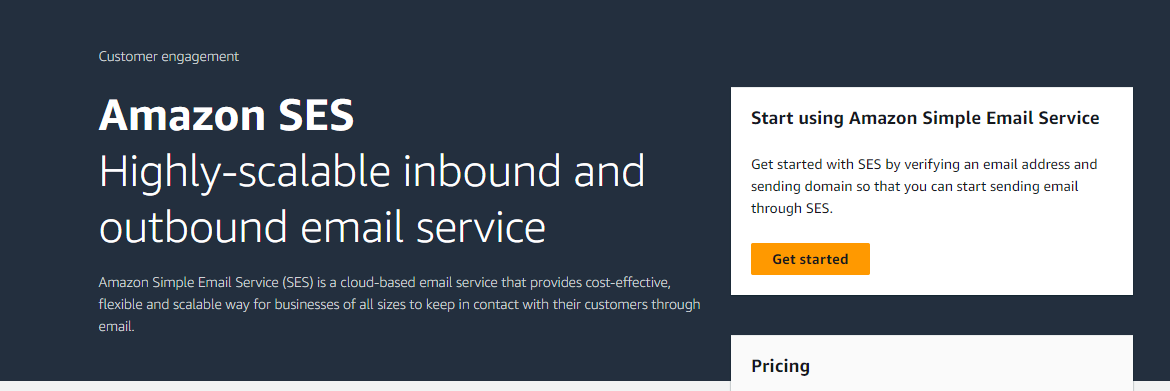
* **Scalability:** AWS SES can handle large volumes of emails, making it ideal for growing businesses.
* **Cost-Effective:** You only pay for what you send, with no upfront costs or minimum fees.
* **Deliverability:** AWS SES integrates with other AWS services to ensure high email deliverability, including reputation management and feedback loops.
* **Security:** It offers features like domain verification, DKIM (DomainKeys Identified Mail), and SPF (Sender Policy Framework) to ensure that your emails are trusted.

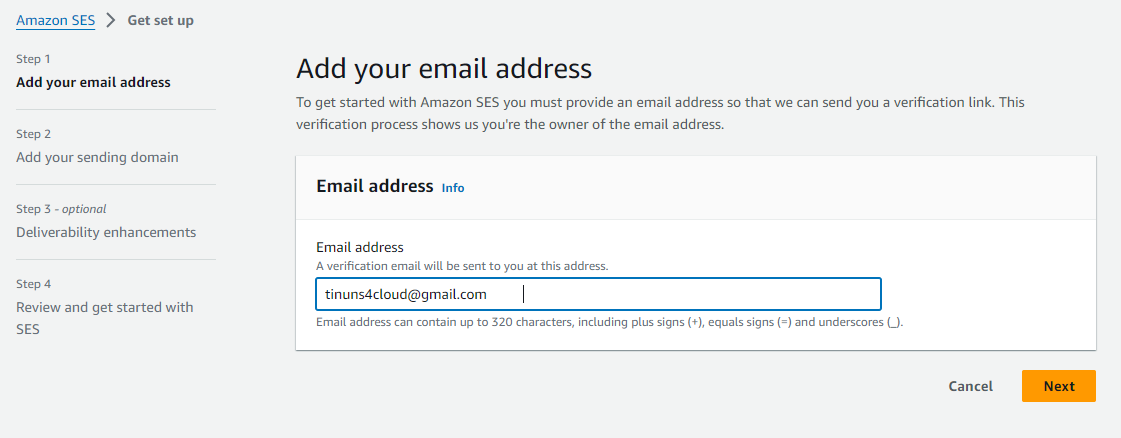
### **Overview of the Process**

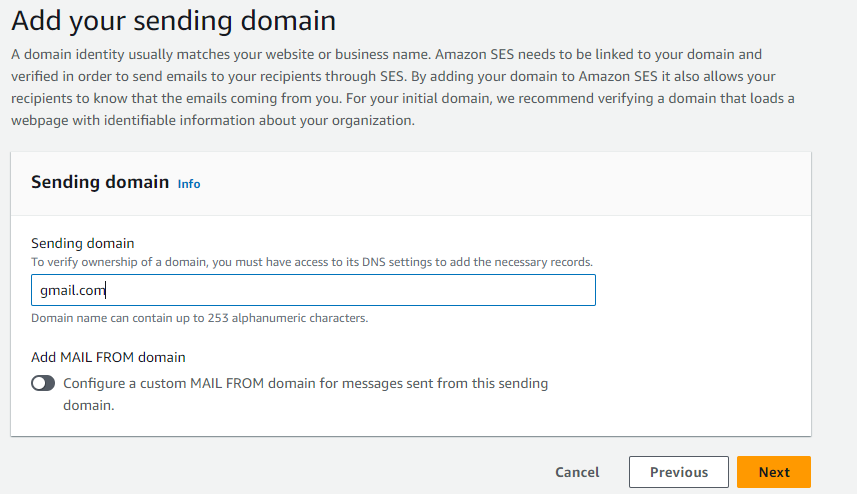
1. **Verify the Recipient Address or Domain:** Before sending emails, you must verify the email addresses or domains to ensure you're authorized to send emails from them.
2. **Obtain SMTP Credentials:** AWS SES provides SMTP credentials that you can use to authenticate your email-sending requests via the SMTP interface.
3. **Set Up PHP Environment:** Install PHP and necessary dependencies on your server or local environment to execute the email-sending script.
4. **Install PHPMailer:** PHPMailer is a popular PHP library that simplifies sending emails via SMTP. It provides a flexible and secure way to send emails from PHP applications.
5. **Send an Email:** Using PHPMailer, connect to AWS SES via SMTP and send emails to your verified recipients. You can customize the email content, including HTML formatting, attachments, and more.

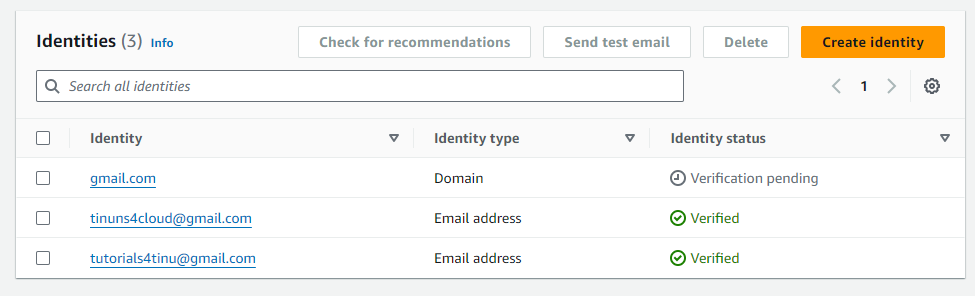
### **Benefits of Using AWS SES with PHP**

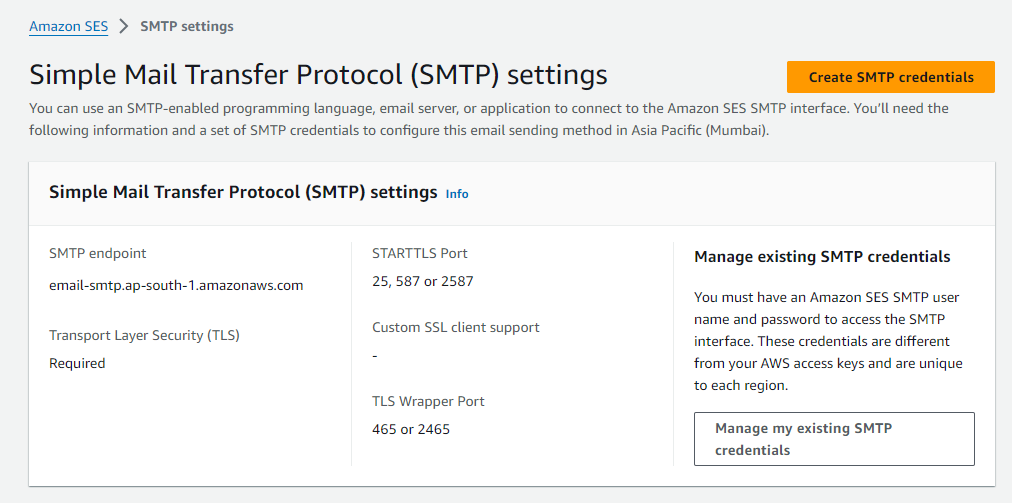
* **Ease of Integration:** With PHPMailer and SMTP support, integrating AWS SES into your PHP application is straightforward.
* **Reliable Delivery:** AWS SES's infrastructure is designed to ensure that your emails reach their intended recipients.
* **Customizability:** PHPMailer allows you to easily customize email headers, body content, and attachments.

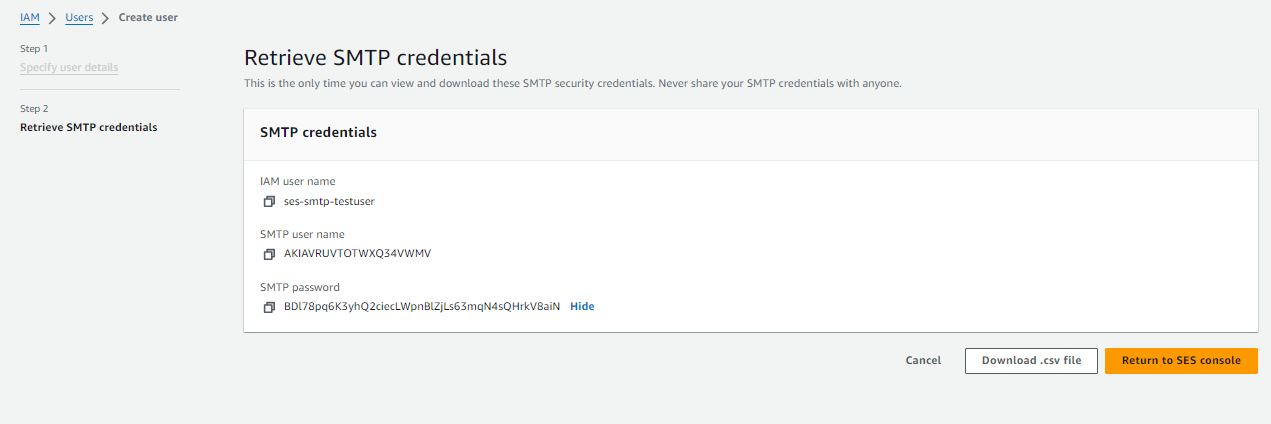












Link: https://getcomposer.org/download/

This example uses the PHPMailer class to send email through Amazon SES using the SMTP interface.

Before you perform the following procedure you must complete the tasks in [Setting up Amazon Simple Email Service](https://docs.aws.amazon.com/ses/latest/dg/setting-up.html). In addition to setting up Amazon SES you must complete the following prerequisites to sending email with PHP:

###### **Prerequisites:**

* **Install PHP** – PHP is available at [http://php.net/downloads.php](https://php.net/downloads.php). After you install PHP, add the path to PHP in your environment variables so that you can run PHP from any command prompt.
* **Install the Composer dependency manager** – After you install the Composer dependency manager, you can download and install the PHPMailer class and its dependencies. To install Composer, follow the installation instructions at <https://getcomposer.org/download>.

**Command-line installation**

To quickly install Composer in the current directory, run the following script in your terminal. To automate the installation, use [the guide on installing Composer programmatically](https://getcomposer.org/doc/faqs/how-to-install-composer-programmatically.md).

php -r "copy('https://getcomposer.org/installer', 'composer-setup.php');"

php -r "if (hash\_file('sha384', 'composer-setup.php') === 'dac665fdc30fdd8ec78b38b9800061b4150413ff2e3b6f88543c636f7cd84f6db9189d43a81e5503cda447da73c7e5b6') { echo 'Installer verified'; } else { echo 'Installer corrupt'; unlink('composer-setup.php'); } echo PHP\_EOL;"

php composer-setup.php

php -r "unlink('composer-setup.php');"

This installer script will simply check some php.ini settings, warn you if they are set incorrectly, and then download the latest composer.phar in the current directory. The 4 lines above will, in order:

* Download the installer to the current directory
* Verify the installer SHA-384, which you can also [cross-check here](https://composer.github.io/pubkeys.html)
* Run the installer
* Remove the installer

Most likely, you want to put the composer.phar into a directory on your PATH, so you can simply call composer from any directory (*Global install*), using for example:

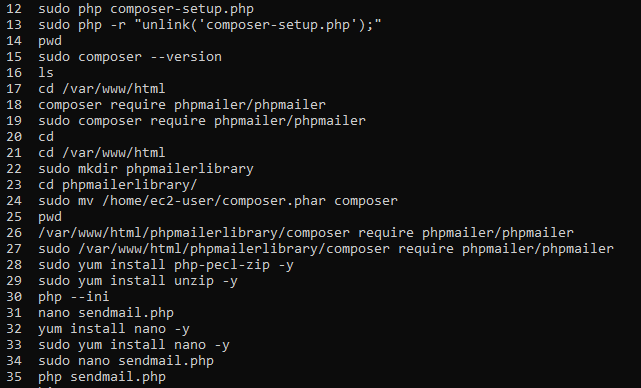
sudo mv composer.phar /usr/local/bin/composer

* **Install the PHPMailer class** – After you install Composer, run the following command to install PHPMailer:

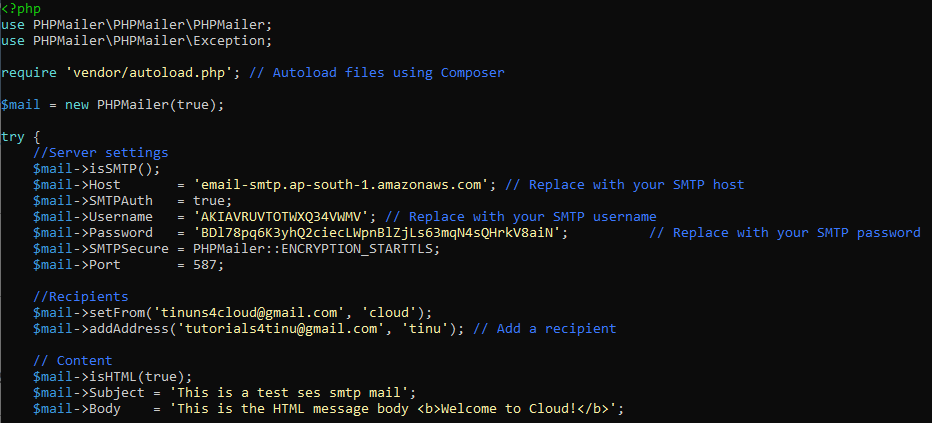
path/to/composer require phpmailer/phpmailer











**Commands Explanation:**

1. **Install PHP and Required Extensions**:

**sudo yum install -y php php-cli php-fpm php-json php-common php-mbstring php-xml php-gd php-zip**

* Installs PHP and its extensions required for running PHP scripts and working with various data types, files, and web content.

1. **Install Composer**: Composer is a dependency manager for PHP, allowing you to manage libraries and dependencies easily.

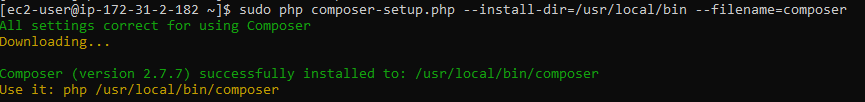
* **Download Composer Installer**:

**sudo php -r "copy('https://getcomposer.org/installer', 'composer- setup.php');"**

* Downloads the Composer installer script (composer-setup.php).
* **Install Composer**:

To install composer in **dir=/usr/local/bin**

**sudo php composer-setup.php --install-dir=/usr/local/bin --filename=composer**

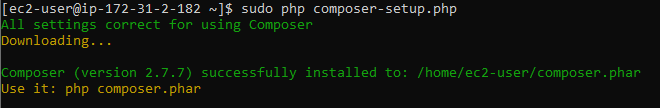


* Runs the installer to install Composer globally on your system. --install-dir=/usr/local/bin specifies the directory, and --filename=composer sets the command name for Composer.

**OR**

To install composer in home directory of ec2-user

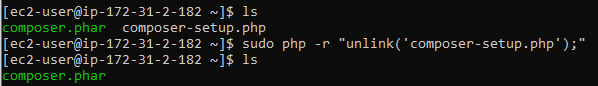
**sudo php composer-setup.php**



* **Remove Installer**:

**sudo php -r "unlink('composer-setup.php');"**

* Deletes the installer script after Composer is installed.



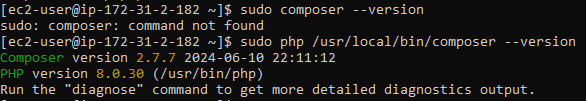
* **Verify Installation**:

**sudo composer –version**

**or**

**sudo php /usr/local/bin/composer --version**

* Checks if Composer was installed successfully by displaying its version.



1. **Install PHPMailer Using Composer**: PHPMailer is a popular library for sending emails from PHP applications, with support for SMTP and other protocols.

* **Navigate to the Web Root Directory**:

**cd /var/www/html**

Changes the directory to your web server’s root, where your PHP application resides.

1. **Create Directory for PHPMailer Library (Optional)**:

**sudo mkdir phpmailerlibrary**

**cd phpmailerlibrary/**

* Creates a directory to organize and store the PHPMailer library files.

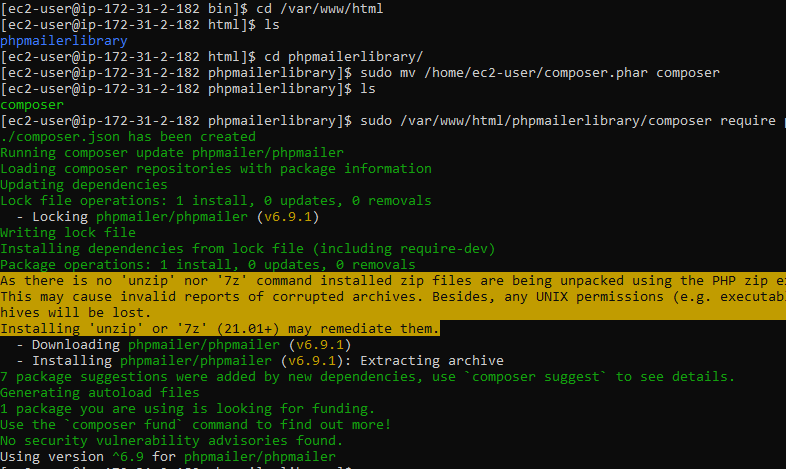
**sudo mv /home/ec2-user/composer.phar composer**

* Moves the Composer PHAR file into the current directory and renames it to composer.

1. **Install PHPMailer in Specific Directory (Optional)**:

**sudo /var/www/html/phpmailerlibrary/composer require phpmailer/phpmailer**

* Installs PHPMailer inside the phpmailerlibrary directory, though this step is optional if PHPMailer is already installed in the project root.



1. **Install Zip Extension**:

**sudo yum install php-pecl-zip -y**

* Installs the PHP zip extension, which is sometimes required by Composer or PHP scripts that need to work with zip files.

1. **Unzip Utility**:

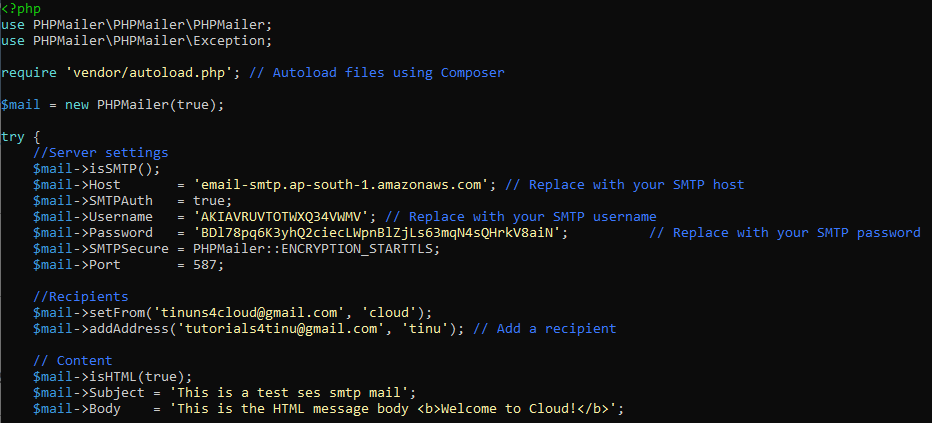
**sudo yum install nano -y**

**sudo yum install unzip -y**

* Installs the unzip utility, which may be needed if you are working with compressed files.

1. **Create and Edit PHP Script**:

**sudo nano sendmail.php**



**<?php**

**use PHPMailer\PHPMailer\PHPMailer;**

**use PHPMailer\PHPMailer\Exception;**

**require 'vendor/autoload.php'; // Autoload files using Composer**

**$mail = new PHPMailer(true);**

**try {**

**//Server settings**

**$mail->isSMTP();**

**$mail->Host = 'email-smtp.ap-south-1.amazonaws.com'; // Replace with your SMTP host**

**$mail->SMTPAuth = true;**

**$mail->Username = 'AKIAVRUVTOTWXQ34VWMV'; // Replace with your SMTP username**

**$mail->Password = 'BDl78pq6K3yhQ2ciecLWpnBlZjLs63mqN4sQHrkV8aiN'; // Replace with your SMTP password**

**$mail->SMTPSecure = PHPMailer::ENCRYPTION\_STARTTLS;**

**$mail->Port = 587;**

**//Recipients**

**$mail->setFrom('tinuns4cloud@gmail.com', 'cloud');**

**$mail->addAddress('tutorials4tinu@gmail.com', 'tinu'); // Add a recipient**

**// Content**

**$mail->isHTML(true);**

**$mail->Subject = 'This is a test ses smtp mail';**

**$mail->Body = 'This is the HTML message body <b>Welcome to Cloud!</b>';**

**$mail->AltBody = 'This is the body in plain text for non-HTML mail clients';**

**$mail->send();**

**echo 'Message has been sent';**

**} catch (Exception $e) {**

**echo "Message could not be sent. Mailer Error: {$mail->ErrorInfo}";**

**} ?>**

* Opens the sendmail.php file in the nano text editor for editing, where you will write or paste the PHP script to send an email using SES and SMTP.

1. **Run PHP Script**:

**php sendmail.php**

* Executes the sendmail.php script, which will use PHPMailer to send an email through AWS SES.
* 

**Summary:**

* **Composer**: A dependency manager for PHP, used to install libraries like PHPMailer.
* **PHPMailer**: A library to send emails from PHP using various protocols, including SMTP (with AWS SES).
* **Commands**: The commands involve setting up PHP, installing Composer, using Composer to manage PHPMailer, and running the PHP script to send an email.

