

PHP - Sending Emails

The provision of sending emails is one the commonly required features of a typical PHP powered web application. You would like to send emails containing notifications, updates and other communications to your registered users, through your PHP application itself, instead of a different mail service. You can add this capability to your PHP application by adopting the techniques described in this chapter.

PHP has a built-in mail() function to send an email. However, you need configure properly the "php.ini" settings to be able to do so. First, you must know the SMTP domain of the web hosting platform that you are using. For example, if your website is being hosted on GoDaddy hosting service, the SMTP domain is "smtp.secureserver.net", which you should use in the configuration.

If you use Windows based hosting of GoDaddy, you should ensure that two directives are enabled in php.ini file. The first is called SMTP that defines your email server address. The second is called sendmail_from which defines your own email address.

The configuration for Windows should look something like this –

```
[mail function]
; For Win32 only.
SMTP = smtp.secureserver.net

; For win32 only
sendmail_from = webmaster@tutorialspoint.com
```

Linux users simply need to let PHP know the location of their sendmail application. The path and any desired switches should be specified to the sendmail_path directive.

The configuration for Linux should look something like this –

```
[mail function]
; For Win32 only.
SMTP =

; For win32 only
sendmail_from =

; For Unix only
sendmail_path = /usr/sbin/sendmail -t -i
```



mail() function in PHP requires three mandatory arguments that specify the recipient's email address, the subject of the message and the actual message additionally there are other two optional parameters.

```
mail( to, subject, message, headers, parameters );
```

Parameters

- **to** – Required. Specifies the receiver / receivers of the email
- **subject** – Required. Specifies the subject of the email. This parameter cannot contain any newline characters
- **message** – Required. Defines the message to be sent. Each line should be separated with a LF (\n). Lines should not exceed 70 characters
- **headers** – Optional. Specifies additional headers, like From, Cc, and Bcc. The additional headers should be separated with a CRLF (\r\n)
- **parameters** – Optional. Specifies an additional parameter to the send mail program

Multiple recipients can be specified as the first argument to the mail() function in a comma separated list.

Sending HTML Email

When you send a text message using PHP then all the content will be treated as simple text. Even if you will include HTML tags in a text message, it will be displayed as simple text and HTML tags will not be formatted according to HTML syntax. But PHP provides option to send an HTML message as actual HTML message.

While sending an email message you can specify a Mime version, content type and character set to send an HTML email.

Example

The following example shows how to send an HTML email message to "xyz@somedomain.com" copying it to "afgh@somedomain.com". You can code this program in such a way that it should receive all content from the user and then it should send an email.

It should receive all content from the user and then it should send an email.

[Open Compiler](#)

```
<?php
    $to = "xyz@somedomain.com";
    $subject = "This is subject";

    $message = "<b>This is HTML message.</b>";
    $message .= "<h1>This is headline.</h1>";

    $header = "From:abc@somedomain.com \r\n";
    $header .= "Cc:afgh@somedomain.com \r\n";
    $header .= "MIME-Version: 1.0\r\n";
    $header .= "Content-type: text/html\r\n";

    $retval = mail ($to,$subject,$message,$header);

    if( $retval == true ) {
        echo "Message sent successfully...";
    }else {
        echo "Message could not be sent...";
    }
?>
```

It will produce the following **output** –

```
Message could not be sent...
sh: 1: /usr/sbin/sendmail: not found
```

Explore our [latest online courses](#) and learn new skills at your own pace. Enroll and become a certified expert to boost your career.

Sending Email from Localhost

The above method of calling PHP mail() may not work on your localhost. In that case, there is an alternate solution to sending email. You can use PHPMailer to send email using SMTP from localhost.

PHPMailer is an open-source library to connect SMTP to send emails. You can download it from PEAR or Composer repositories, or download it from <https://github.com/PHPMailer/PHPMailer>. Download the ZIP file from here, and copy the contents of the PHPMailer folder into one of the include_path directories specified in your PHP configuration and load each class file manually.

Example

Use the following PHP script to send email with PHPMailer library –

Phpmailer.php

```
<?php
use PHPMailer\PHPMailer\PHPMailer;
use PHPMailer\PHPMailer\SMTP;
use PHPMailer\PHPMailer\Exception;

require_once __DIR__ . '/vendor/phpmailer/src/Exception.php';
require_once __DIR__ . '/vendor/phpmailer/src/PHPMailer.php';
require_once __DIR__ . '/vendor/phpmailer/src/SMTP.php';
require 'vendor/autoload.php';

$mail = new PHPMailer;
if(isset($_POST['send'])){

    // getting post values
    $fname=$_POST['fname'];
    $toemail=$_POST['toemail'];
    $subject=$_POST['subject'];
    $message=$_POST['message'];
    $mail->isSMTP(); // Set mailer to use SMTP
    $mail->Host = 'smtp.gmail.com';
    $mail->SMTPAuth = true;
    $mail->Username = 'myID@gmail.com'; // SMTP username
    $mail->Password = 'mypassword'; // SMTP password

    // Enable TLS encryption, 'ssl' also accepted
    $mail->SMTPSecure = 'tls';
    $mail->Port = 587;
    $mail->setFrom(myID@gmail.com', 'My_Name');
    $mail->addReplyTo(myID@gmail.com', 'My_Name');
    $mail->addAddress($toemail); // Add a recipient
    $mail->isHTML(true); // Set email format to HTML
    $bodyContent=$message;
    $mail->Subject =$subject;
    $body = 'Dear'.$fname;
    $body .='<p>'.$message.'</p>';
    $mail->Body = $body;
```

```

    if(!$mail->send()) {
        echo 'Message could not be sent.';
        echo 'Mailer Error: ' . $mail->ErrorInfo;
    } else {
        echo 'Message has been sent';
    }
}
?>

```

Use the following HTML form to compose the mail message. The form is submitted to the above phpmail.php script

Email.html

```

<h1>PHP - Sending Email</h1>
<form action="PHPmailer.php" method="post">
    <label for="inputName">Name</label>
    <input type="text" id="inputName" name="fname" required>

    <label for="inputEmail">Email</label>
    <input type="email" id="inputEmail" name="toemail" required>

    <label for="inputSubject">Subject</label>
    <input type="text" id="inputSubject" name="subject" required>

    <label for="inputMessage">Message</label>
    <textarea id="inputMessage" name="message" rows="5" required></textarea>

    <button type="submit" name="send">Send</button>
</form>

```

Sending Attachments with Email

To send an email with mixed content you should set Content-type header to multipart/mixed. Then text and attachment sections can be specified within boundaries.

A boundary is started with two hyphens followed by a unique number which can not appear in the message part of the email. A PHP function md5() is used to create a 32 digit hexadecimal number to create unique number. A final boundary denoting the email's final section must also end with two hyphens.

Example

Take a look at the following example –


[Open Compiler](#)

```
<?php
```

```
// request variables
$from = $_REQUEST["from"];
$emaila = $_REQUEST["emaila"];
$filea = $_REQUEST["filea"];

if ($filea) {
    function mail_attachment ($from , $to, $subject, $message, $attachment){
        $fileatt = $attachment;          // Path to the file
        $fileatt_type = "application/octet-stream"; // File Type

        $start = strrpos($attachment, '/') == -1 ?
            strrpos($attachment, '//') : strrpos($attachment, '/')+1;

        // Filename that will be used for the file as the attachment
        $fileatt_name = substr($attachment, $start,
            strlen($attachment));

        $email_from = $from;          // Who the email is from
        $subject = "New Attachment Message";

        $email_subject = $subject; // The Subject of the email
        $email_txt = $message;      // Message that the email has in it
        $email_to = $to;           // Who the email is to

        $headers = "From: ".$email_from;
        $file = fopen($fileatt,'rb');
        $data = fread($file,filesize($fileatt));
        fclose($file);

        $msg_txt="\n\n You have recieved a new attachment message from $from";
        $semi_rand = md5(time());
        $mime_boundary = "==Multipart_Boundary_x{$semi_rand}x";
        $headers .= "\nMIME-Version: 1.0\n" . "Content-Type: multipart/mixed;\n"
            boundary=\"{$mime_boundary}\"";
    }
}
```

```

$email_txt .= $msg_txt;

$email_message .= "This is a multi-part message in MIME format.\n\n" .
"--{$mime_boundary}\n" . "Content-Type:text/html;
charset = \"iso-8859-1\"\n" . "Content-Transfer-Encoding: 7bit\n\n" .
$email_txt . "\n\n";

$data = chunk_split(base64_encode($data));

$email_message .= "--{$mime_boundary}\n" . "Content-Type: {$fileatt_type}
" name = \"{$fileatt_name}\" \n" . //"Content-Disposition: attachment;\n"
//" filename = \"{$fileatt_name}\" \n" . "Content-Transfer-Encoding:
"base64\n\n" . $data . "\n\n" . "--{$mime_boundary}--\n";

$ok = mail($email_to, $email_subject, $email_message, $headers);

if($ok) {
    echo "File Sent Successfully.";
    // delete a file after attachment sent.
    unlink($attachment);
} else {
    die("Sorry but the email could not be sent. Please go back and try again");
}
}
move_uploaded_file($_FILES["filea"]["tmp_name"],
'temp/'.basename($_FILES['filea']['name']));

mail_attachment("$from", "youremailaddress@gmail.com",
"subject", "message", ("temp/".$_FILES["filea"]["name"]));
}
?>
<html>
<head>
<script language = "javascript" type = "text/javascript">
function CheckData45() {
    with(document.filepost) {
        if(filea.value != "") {
            document.getElementById('one').innerText = "Attaching File ... Please wait";
        }
    }
}
}

```

```

</script>
</head>
<body>
  <table width = "100%" height = "100%" border = "0"
    cellpadding = "0" cellspacing = "0">
    <tr>
      <td align = "center">
        <form name = "filepost" method = "post"
          action = "file.php" enctype = "multipart/form-data" id = "file">
          <table width = "300" border = "0" cellspacing = "0"
            cellpadding = "0">
            <tr valign = "bottom">
              <td height = "20">Your Name:</td>
            </tr>
            <tr>
              <td><input name = "from" type = "text" id = "from" size = "30"
            </tr>
            <tr valign = "bottom">
              <td height = "20">Your Email Address:</td>
            </tr>
            <tr>
              <td class = "frmtxt2"><input name = "emaila" type = "text" id
            </tr>
            <tr>
              <td height = "20" valign = "bottom">Attach File:</td>
            </tr>
            <tr valign = "bottom">
              <td valign = "bottom"><input name = "filea" type = "file" id
            </tr>
            <tr>
              <td height = "40" valign = "middle">
                <input name = "Reset2" type = "reset" id = "Reset2" value
                <input name = "Submit2" type = "submit" value = "Submit" c
              </td>
            </tr>
          </table>
        </form>
      <center>
        <table width = "400">
          <tr>
            <td id = "one"></td>
          </tr>

```



```
</table>
</center>
</td>
</tr>
</table>
</body>
</html>
```

It will produce the following **output** –

| | |
|--------------------------------------|---|
| Your Name: | <input type="text"/> |
| Your Email Address: | <input type="text"/> |
| Attach File: | <input type="button" value="Choose File"/> No file chosen |
| <input type="button" value="Reset"/> | <input type="button" value="Submit"/> |