

Mail

Introduction

The [mail\(\)](#) function allows you to send mail.

Installing/Configuring

Requirements

For the Mail functions to be available, PHP must have access to the *sendmail* binary on your system during compile time. If you use another mail program, such as *qmail* or *postfix*, be sure to use the appropriate *sendmail* wrappers that come with them. PHP will first look for *sendmail* in your *PATH*, and then in the following: */usr/bin:/usr/sbin:/usr/etc:/etc:/usr/ucblib:/usr/lib*. It's highly recommended to have *sendmail* available from your *PATH*. Also, the user that compiled PHP must have permission to access the *sendmail* binary.

Installation

There is no installation needed to use these functions; they are part of the PHP core.

Runtime Configuration

The behaviour of these functions is affected by settings in *php.ini*.

Mail configuration options

Name	Default	Changeable	Changelog
SMTP	"localhost"	PHP_INI_ALL	
smtp_port	"25"	PHP_INI_ALL	Available since PHP 4.3.0.
sendmail_from	NULL	PHP_INI_ALL	
sendmail_path	"/usr/sbin/sendmail -t -i"	PHP_INI_SYSTEM	

For further details and definitions of the `PHP_INI_*` constants, see the [php.ini directives](#).

Here's a short explanation of the configuration directives.

SMTP [string](#)

Used under Windows only: host name or IP address of the SMTP server PHP should use for mail sent with the [mail\(\)](#) function.

smtp_port [int](#)

Used under Windows only: Number of the port to connect to the server specified with the *SMTP* setting when sending mail with [mail\(\)](#); defaults to 25. Only available since PHP 4.3.0.

sendmail_from [string](#)

Which "From:" mail address should be used in mail sent from PHP under Windows. This directive also sets the "Return-Path:" header.

sendmail_path [string](#)

Where the *sendmail* program can be found, usually */usr/sbin/sendmail* or */usr/lib/sendmail*. *configure* does an honest attempt of locating this one for you and set a default, but if it fails, you can set it here. Systems not using *sendmail* should set this directive to the *sendmail* wrapper/replacement their mail system offers, if any. For example, [» Qmail](#) users can normally set it to */var/qmail/bin/sendmail* or */var/qmail/bin/qmail-inject*. *qmail-inject* does not require any option to process mail correctly. This directive works also under Windows. If set, *smtp*, *smtp_port* and *sendmail_from* are ignored and the specified command is executed.

Resource Types

This extension has no resource types defined.

Predefined Constants

This extension has no constants defined.

Mail Functions

ezmlm_hash

ezmlm_hash -- Calculate the hash value needed by EZMLM

Description

int **ezmlm_hash** (string \$addr)

[ezmlm_hash\(\)](#) calculates the hash value needed when keeping EZMLM mailing lists in a MySQL database.

Parameters

addr

The email address that's being hashed.

Return Values

The hash value of *addr*.

Examples

Example #1 - Calculating the hash and subscribing a user

```
<?php

$user = "joecool@example.com";
$hash = ezmlm_hash($user);
$query = sprintf("INSERT INTO sample VALUES (%s, '%s')", $hash, $user);
$db->query($query); // using PHPLIB db interface

?>
```

mail

mail -- Send mail

Description

```
bool mail ( string $to, string $subject, string $message [, string $additional_headers [,
string $additional_parameters ] ] )
```

Sends an email.

Parameters

to

Receiver, or receivers of the mail. The formatting of this string must comply with [» RFC 2822](#). Some examples are:

- user@example.com
- user@example.com, anotheruser@example.com
- User <user@example.com>
- User <user@example.com>, Another User <anotheruser@example.com>

subject

Subject of the email to be sent.

Caution
This must not contain any newline characters, or the mail may not be sent properly.

message

Message to be sent. Each line should be separated with a LF (\n). Lines should not be larger than 70 characters.

Caution
(Windows only) When PHP is talking to a SMTP server directly, if a full stop is found on the start of a line, it is removed. To counter-act this, replace these occurrences with a double dot. <?php \$text = str_replace("\n.", "\n..", \$text); ?>

additional_headers (optional)

String to be inserted at the end of the email header. This is typically used to add extra headers (From, Cc, and Bcc). Multiple extra headers should be separated with a CRLF (\r\n).

Note

When sending mail, the mail *must* contain a *From* header. This can be set with the *additional_headers* parameter, or a default can be set in *php.ini*.

Failing to do this will result in an error message similar to *Warning: mail(): "sendmail_from" not set in php.ini or custom "From:" header missing*. The *From* header sets also *Return-Path* under Windows.

Note

If messages are not received, try using a LF (\n) only. Some poor quality Unix mail transfer agents replace LF by CRLF automatically (which leads to doubling CR if CRLF is used). This should be a last resort, as it does not comply with [» RFC 2822](#).

additional_parameters (optional)

The *additional_parameters* parameter can be used to pass an additional parameter to the program configured to use when sending mail using the *sendmail_path* configuration setting. For example, this can be used to set the envelope sender address when using sendmail with the *-f* sendmail option. The user that the webserver runs as should be added as a trusted user to the sendmail configuration to prevent a 'X-Warning' header from being added to the message when the envelope sender (-f) is set using this method. For sendmail users, this file is */etc/mail/trusted-users*.

Return Values

Returns **TRUE** if the mail was successfully accepted for delivery, **FALSE** otherwise.

It is important to note that just because the mail was accepted for delivery, it does NOT mean the mail will actually reach the intended destination.

ChangeLog

Version	Description
4.3.0 (Windows only)	All custom headers (like From, Cc, Bcc and Date) are supported, and are not case-sensitive. (As custom headers are not interpreted by the MTA in the first place, but

	are parsed by PHP, PHP < 4.3 only supported the Cc header element and was case-sensitive).
4.2.3	The <code>additional_parameters</code> parameter is disabled in <code>safe_mode</code> and the <code>mail()</code> function will expose a warning message and return FALSE when used.
4.0.5	The <code>additional_parameters</code> parameter was added.

Examples

Example #2 - Sending mail.

Using `mail()` to send a simple email:

```
<?php
// The message
$message = "Line 1\nLine 2\nLine 3";

// In case any of our lines are larger than 70 characters, we should use
wordwrap()
$message = wordwrap($message, 70);

// Send
mail('caffinated@example.com', 'My Subject', $message);
?>
```

Example #3 - Sending mail with extra headers.

The addition of basic headers, telling the MUA the From and Reply-To addresses:

```
<?php
$to      = 'nobody@example.com';
$subject = 'the subject';
$message = 'hello';
$headers = 'From: webmaster@example.com' . "\r\n" .
'Reply-To: webmaster@example.com' . "\r\n" .
'X-Mailer: PHP/' . phpversion();

mail($to, $subject, $message, $headers);
?>
```

Example #4 - Sending mail with an additional command line parameter.

The `additional_parameters` parameter can be used to pass an additional parameter to the program configured to use when sending mail using the `sendmail_path`.

```
<?php
mail('nobody@example.com', 'the subject', 'the message', null,
    '-fwebmaster@example.com');
?>
```

Example #5 - Sending HTML email

It is also possible to send HTML email with [mail\(\)](#).

```
<?php
// multiple recipients
$to = 'aidan@example.com' . ', ' . 'wez@example.com';
$to .= 'wez@example.com';

// subject
$subject = 'Birthday Reminders for August';

// message
$message = '
<html>
<head>
  <title>Birthday Reminders for August</title>
</head>
<body>
  <p>Here are the birthdays upcoming in August!</p>
  <table>
    <tr>
      <th>Person</th><th>Day</th><th>Month</th><th>Year</th>
    </tr>
    <tr>
      <td>Joe</td><td>3rd</td><td>August</td><td>1970</td>
    </tr>
    <tr>
      <td>Sally</td><td>17th</td><td>August</td><td>1973</td>
    </tr>
  </table>
</body>
</html>
';

// To send HTML mail, the Content-type header must be set
$headers = 'MIME-Version: 1.0' . "\r\n";
$headers .= 'Content-type: text/html; charset=iso-8859-1' . "\r\n";

// Additional headers
$headers .= 'To: Mary <mary@example.com>, Kelly <kelly@example.com>' .
"\r\n";
$headers .= 'From: Birthday Reminder <birthday@example.com>' . "\r\n";
$headers .= 'Cc: birthdayarchive@example.com' . "\r\n";
$headers .= 'Bcc: birthdaycheck@example.com' . "\r\n";
```

```
// Mail it
mail($to, $subject, $message, $headers);
?>
```

Note

If intending to send HTML or otherwise Complex mails, it is recommended to use the PEAR package » [PEAR::Mail_Mime](#).

Notes

Note

The Windows implementation of [mail\(\)](#) differs in many ways from the Unix implementation. First, it doesn't use a local binary for composing messages but only operates on direct sockets which means a *MTA* is needed listening on a network socket (which can either on the localhost or a remote machine).

Second, the custom headers like *From:*, *Cc:*, *Bcc:* and *Date:* are *not* interpreted by the *MTA* in the first place, but are parsed by PHP.

As such, the *to* parameter should not be an address in the form of "Something <someone@example.com>". The mail command may not parse this properly while talking with the *MTA*.

Note

Email with attachments and special types of content (e.g. HTML) can be sent using this function. This is accomplished via MIME-encoding - for more information, see this » [Zend article](#) or the » [PEAR Mime Classes](#).

Note

It is worth noting that the [mail\(\)](#) function is not suitable for larger volumes of email in a loop. This function opens and closes an SMTP socket for each email, which is not very efficient.

For the sending of large amounts of email, see the » [PEAR::Mail](#), and » [PEAR::Mail_Queue](#) packages.

Note
The following RFCs may be useful: » RFC 1896 , » RFC 2045 , » RFC 2046 , » RFC 2047 , » RFC 2048 , » RFC 2049 , and » RFC 2822 .

See Also

- [imap_mail\(\)](#)
- [» PEAR::Mail](#)
- [» PEAR::Mail_Mime](#)