I'm not robot

reCAPTCHA

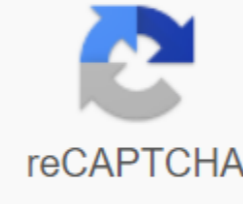**Continue**

Inbox mass mailer php

New Additions $225.00 $175.00 New Additions $850.00 $650.00 New Additions $225.00 $175.00 New Additions $175.00 $150.00 New Additions $250.00 $175.00 New Additions $95.00 $60.00 New Additions $125.00 $95.00 New Additions $250.00 $175.00 New Additions $95.00 $65.00 New Additions $125.00 $85.00 New Additions $100.00 $85.00 New Additions $125.00 $95.00 New Additions $150.00 $125.00 New Additions $125.00 $99.00 New Additions $175.00 $125.00 New Additions $199.00 $125.00 New Additions $199.00 $129.00 New Additions $40.00 $30.00 New Additions $100.00 $75.00 New Additions $79.00 $69.00 New Additions $199.00 $149.00 New Additions $125.00 $95.00 27,989,126 Latin America Emails 3,254,114 Hungary Emails (.HU Domains) 115,000 Petrolium & Gas Agency Emails 8,107,674 Indonesia Gmail Emails 11,544,380 Indonesia Emails (Tokopedia) 815,075 Zing Vietnam Emails 2,009,961 Yahoo Vietnam Domain Emails (yahoo.com.vn) 3.5 Million Spain Email Address List 750,000 Chile Email Address List 3 Million Argentina Email Address List 1,165,993 Austria Emails (. AT Domains Only) 911.202 Portugal Emails (.PT Domains Only) 261,855 US Insurance Agent Records 600,000 Sweden Emails (.SE) 6,217,359 USA Business Leads Database (Full Details) 1,577,564 Uzbekistan Emails (.UZ) Maxprog Bulk Email Verifier V3.7 6 10709 Singapore Business Emails 961,228 Apple iCloud Email 16,803,570 Mail.Ru Email 31,650,578 United Kingdom Emails (.UK) 5,156,969 Italy Emails (.IT) 1,180,664 Brazilian Emails (.BR) Got A Question? We are available on WhatsApp On +917359077550, ICQ On 704859458 & Skype On phpinboxmailer@yahoo.com 100% encrypted payment protection. We will never share any of your details with anyone at competitive prices on all our guaranteed products, up to 70% discount, instant delivery guarantee. Delphi ist die leistungsstarke RAD-Umgebung für die schnelle Entwicklung nativer und plattformübergreifender Anwendungen. Die visuellen Designtools und deintegrierten Tools, bringen Entwickler und deren Teams schneller zum Ziel.Testen Sie, wie einfach native Desktop-Anwenungen oder mobile Apps erstellt werden können. Desktop.com's Early Access release allows individuals, businesses, and teams to effectively explain their digital environment by keeping all web apps and bookmarks organized, tagged, and searchable under a single icon! 1 2 3 4 5 Practice-Web has served the dental community since 1988. Our team of experienced and friendly support professionals are here to help you throughout your practice life. We have over 1,700 very successful customers across the country. All this at a fraction of the cost you will pay for a similar solution (or multiple solutions) elsewhere! 6 7 8 9 A simple, effective PHP-oriented object based on mass mailing scenarios for membership websites, affiliate marketing programs, clubs and associations. GNU General Public License version 2.0 (GPLv2) Delphi ist dead RAD-Umgebung für die schnelle Entwicklung nativer und plattformübergreifender Anwendungen. Die visuellen Designtools und deintegrierten Tools, bringen Entwickler und deren Teams schneller zum Ziel.Testen Sie, wie einfach native Desktop-Anwenungen oder mobile Apps erstellt werden können. Login to evaluate this project details additional French, English based web PHP MySQL 2009-04-21 Similar business software reporting inappropriate content Page 2 A simple, effective object oriented PHP based on mass mailing scenarios for member sites, affiliate marketing programs, clubs and associations. GNU General Public License version 2.0 (GPLv2) Erstellen Sie aus einer einzigen Codebasis schnell native Anwendungen für Windows, Linux, MacOS, iOS, Android, Cloud und IoT. Entwerfen, Erstellen, Ausführenund Verteilen von Anwendungen, die für alle Plattformen optimiert sind. Mit RAD Studio können Sie alle Daten auf jeder Ebene im Unternehmen oder in der Cloudsicher verbinden und einbetten. Testen Sie kostenfrei, wie einfach native Desktop-Anwenungen oder mobile Apps erstellt werden können. Login to evaluate this project details additional French project, English based web PHP MySQL 2009-04-21 Similar business software reporting inappropriate content This is advice, not an answer: You have much, much better using dedicated mailing list software. mailman is an oft-used example, but something as simple as mlmmj can be enough. Mass mailing is actually a more difficult task than it actually appears to be. Not only do you have to send messages, you must also monitor dead addresses to avoid your messages, or worse, your mail server, marked as spam. You must deal with unscincted people for many of the same reasons. You can do these things yourself, but especially handling bounces is hard work and unrewarding. Using the mailing list manager will make things much easier. As for how to make your mail palatable to yahoo, it's a complete problem. For all its bugs, they seem to put big stock in SPF and DomainKey. You will probably have to implement them, which will require cooperation from your mail server administrator. mail ($to string, $subject string, $message string [, $additional_headers [, $additional_parameters] string] : bool to receive, or receive messages. The format of this string must comply with » RFC 2822. Some examples are: user@example.com user@example.com, anotheruser@example.com User <user@example.com> <anotheruser@example.com>sent. Caution Subject must satisfy » RFC 2047. message message to be sent. Each line should be separated with a CRLF (\r). Lines must not be larger Characters. Caution (windows only) When PHP is talking directly to an SMTP server, if a full stop is found on top of a line, it is</anotheruser@example.com> </user@example.com> </user@example.com> </user@example.com> To counter-this effect, replace these appearances with double dots. <?php$text = str_replace(., .., $text);? > additional_headers or array inserted at the bottom of the email title. This is often used to add additional headlines (From, Cc, and Bcc). Many additional headlines should be separated by CRLF (\r). If external data is used to compose this title, the data should be disinfected so that no unwanted title can be injected. If an array is passed, its keys are the title name and its values are the corresponding title values. Note: Before PHP 5.4.42 and 5.5.27, reppectively, additional_headers no letter title injection protection. Therefore, users must ensure that the specified title is secure and contains only the title. that is, never start the message content by placing multiple new lines. Note: When sending a message, the message must contain the from title. This can be set with additional_headers parameters, or the default can be set in php.ini. Not doing this will result in an error message similar to Warning: mail(): sendmail_from is not set in php.ini or custom From: missing title. From the set title also returns the path when sent directly via SMTP (Windows only). Note: If the message isn't received, try using only LF (). Some Unix (most notably qmail) mail transfer agents replace LF with automatic CRLF (resulting in double CR if CRLF is used). This should be a last-minute method, as it does not comply » RFC 2822. additional_parameters (optional) additional_parameters parameters can be used to pass additional flags as command line options for program configurations to be used when sending messages, as defined by the configuration settings sendmail_path. For example, this can be used to set the envelope sender address when using sendmail with the -f sendmail option. This function is exited by internal escapeshellcmd() to prevent the command from being implemented. escapeshellcmd() prevents command execution. But escapeshellcmd() is applied automatically, some characters allowed such as the email address of internet RFCs cannot be used. Mail() cannot allow such characters, so in programs where the use of such characters is necessary, alternative means of sending emails (such as using a framework or a library) are recommended. Web server users run as should be added as a trusted user to configure sendmail to prevent an 'X-Alert' heading from being added to the message when the sender envelope (-f) is set up using this method. For sendmail users, this file is /etc/mail/trusted-users. Returns TRUE if the message was successfully accepted for sending, FALSE if It is important to note that only because the message has been accepted for it does not mean that the message will actually reach the intended destination. Send #1, for example. Use mail() to send a simple email: <?php the message$message = Line 1\rLine 2\rLine 3\/ In case any of our lines are larger than 70 characters, we should use wordwrap()$message = wordwrap($message, 70, \r/ Sendmail('caffeinated@example.com', 'My Subject', $message;?? > #2 send messages with additional headlines. The addition of basic headlines, for buying words and replying to addresses: For example #3 Mailings with additional headlines such as array This example will be <?php$to = 'nobody@example.com';$subject = 'the subject;$message = 'hello';$headers = 'From: webmaster@example.com'. \r . 'Reply-To: webmaster@example.com'. \r . 'X-Mailer: PHP. phpversionmail($to, $subject, $message, $headers);? =?> sends messages similar to the immediate example above, but surpasses additional headlines such as arrays (available as of PHP 7.2.0). <?php$to = 'nobody@example.com';$subject = 'the subject;$message = 'hello';$headers = array('From' =?>, 'webmaster@example.com', 'Reply-To' => 'webmaster@example.com', 'X-Mailer' => 'PHP/' . phpversionmail ($to, $subject, $message, $headers);? > #4 mailing with an additional command line parameters. The additional_parameters can also be used to pass an additional parameters for the program configured to use when sending messages using sendmail_path. <?phpmail('nobody@example.com', 'the subject', 'the message', null, '-fwebmaster@example.com');?? > #5 HTML email senders Can also send HTML email with mail(). <?php multiple recipients$to =?><html><head> <title>Lời nhắc sinh nhật cho tháng Tám</title></head><body> <p>Dưới đây là những ngày sinh nhật sắp tới trong tháng Tám!</p> <table> <tr> <th>Người</th><th>Ngày</th><th>tháng</th><th>năm</th> </tr> <tr> <td>Johny</td><td>Thứ 10</td><td>Tháng 8</td><td>1970</td> </tr> <tr> <td>Sally</td><td>Thứ 17</td><td>Tháng 8</td><td>1973</td> </tr> <table >&lt;/body></html>'// To send HTML messages, the Content type title must be caption$[] = 'MIME-Version: 1.0';$headers[] = 'Content-type: text/html; charset=iso-8859-1'// Additional title$title[] = 'To: Mary <mary@example.com>, Kelly <kelly@example.com>';$headers[] = 'From: Birthday Reminder <birthday@example.com>';$headers[] = 'Cc: birthdayarchive@example.com';$headers[] = 'Bcc: birthdaycheck@example.com'// Mail itmail($to, $subject, $message, explosion(\r, $headers));? > note: If you intend to send HTML or otherwise</birthday@example.com> </kelly@example.com> </mary@example.com>pear package » PEAR::Mail_Mime. Note: The SMTP (Windows only) implementation of mail() varies in many ways from doing sendmail. First, it does not use a local binary to compose messages but only works on sockets directly which means that an MTA is needed listening on a network socket (which can either be on localhost or a remote computer). Secondly, custom messages such as From:, Cc:, Bcc: and Date: are not explained by the MTA in the first place, but are analyzed syntax by PHP. As such, the parameters should not be an address in the form of Something <someone@example.com>. The message command may not analyze this syntax properly while talking to the MTA. Note: It should be noted that the mail() function does not match the larger volume of emails in a loop. This function opens and closes an SMTP socket for each email, which is not very efficient. To send a large number of emails, see » PEAR::Mail and » PEAR::Mail_Queue package. Note: The following RFCs can be useful: » RFC 1896, » RFC 2045, » RFC 2046, » RFC 2047, » RFC 2048, » RFC 2049, and » RFC 2822. php at simoneast dot net ¶ Often it is useful to find the correct error message triggered by the mail(). Although the function does not provide errors directly, you can use error_get_last() when mail() returns false. <?php$success == mail('example@example.com',= 'my= subject',= $message');if= (!$success)= { = $errormessage=error_get_last()['message'?? > (Successful testing on Windows uses SMTP by default, but sendmail on Linux/OSX may not provide the same level of detail.) Thanks to Unknown Person ¶ If you notice characters that are mis displayed in the email, it's because you need to put the correct Content Type and Charset in the email's header: Owner <?php$headers== 'content-type: text/plain;= charset=utf-8' .= \r;? =?> weak, UTF-8 is your best option. You can set a custom title with the fourth parameters of the mail() function. To make the whole thing waterproof, add the following header too: Now you can use the combination of <?php$headers.='Content-Transfer-Encoding: base64' .= \r;? =?>; UTF-8 and Base64 to properly encode the subject line and recipient name like this: <?php$subject == '=? UTF-8? B?' .= base64_encode('test= email= with= german= umlauts= öäüß')= .= '?=';$recipient = ' =?utf-8?b?' =. .= base64_encode('margret= müller')= .= '?=?> <recipient@domain.com>';'? >;And don't forget Base64 encrypts too: All references are taken <?php$message == base64_encode('this= email= contains= german= umlauts= öäüß.');? =? > from: Anonymous ¶ Security advice: Although it is not recorded, for parameters $to and $subject the mail() function changes to at least \r and space. So these parameters are safe lại tiêm</recipient@domain.com> </someone@example.com> </someone@example.com> </someone@example.com> Title. But you may want to $to commas because of these separate addresses, and you may not want to send them to multiple recipients. The important part is $additional_headers. This parameters cannot be cleaned using the mail() function. Therefore, it is up to you to prevent unwanted \r or be inserted into the values you put in it. Otherwise, you just need to create a potential spam distributor. chris at dot ocproducts com ¶ The 'sendmail' enforcement that PHP uses on Linux/Mac (not Windows) hopes as a line separation. This implementation is a standard, and simulated by other MTAs. confirmed necessary for qmail and postfix, probably also for sendmail and exim but I have not tested. If you come across using \r as a split it may appear to work, but your email will be delicately broken and some middleware may break. It only works because some systems will clean up your mistakes. If you are performing DKIM must be very careful, as DKIM test will fail (at least on the popular confirmation tool) if you screw this up. DKIM must be calculated using \r but then you must convert them all so that when using php mail function. However, on Windows, you should use \r because PHP is using SMTP in this situation, and therefore the normal rules of the SMTP protocol (not the normal rules of unix pipelines) apply. charles dot fisher at arconic dot com ¶ I moved an application to a platform without a local transport agent (MTA). I didn't want to configure an MTA, so I wrote this xxmail function to replace mail() with calls to a remote SMTP server. Hopefully it's from $to, $subject, $body, $headers. $smtp = stream_socket_client ('tcp://smtp.yourmail.com:25', $eno, $estr, 30); $B = 8192; $c = \r; $s = 'myapp@someserver.com'; fwrite($smtp, 'helo ' . _ENV ['HOSTNAME'] . $c); $junk = fgets ($smtp, $B); Envelope($smtp, 'mail from: ' . $s . $c); $junk = fgets ($smtp, $B); fwrite($smtp, 'rcpt to: ' . $to . $c); $junk = fgets ($smtp, $B); fwrite($smtp, 'data' &c . $junk = fgets ($smtp, $B); The title fwrite($smtp, 'To: ' . $to . $c); if(strlen($subject)) fwrite($smtp, 'Theme: ' . $subject . $c); if (strlen($headers)) fwrite($smtp, $headers); Must be \r (divide) fwrite($smtp, $headers . $c); The body if (strlen($body)) fwrite($smtp, $body . $c); fwrite ($smtp, $c . '.' . $c); $junk = fgets ($smtp, $B); Close fwrite ($smtp, 'quit' $c); $junk = fgets ($smtp, $B); fclose($smtp);} Ben Cooke ¶ Note that there is a big difference between the behavior of this function on windows system compared to UNIX system. On Windows it delivers directly to an SMTP server, while on a UNIX system it uses a local command to hand over to the system's own MTA. The upshot of all this is on a Windows messages and your title must use the standard line ends \r in accordance with the email specifications. On a UNIX system of MTA sendmail sendmail Assuming that the data received will use unix end lines and will turn any \r, so you must provide only to mail() on a UNIX system to avoid hypercorrecting the MTA to \r\r. If you use the old bronze on a Windows system, some MTAs will get a little annoying. qmail in particular would refuse completely to accept any message there a lonely without a jim comes at jimbrooking dot net ¶ I recently changed the hosting company and a day trying to see why an email script has been working for years failed on the new server. The answer is that the old hosting company's email server has accepted multiple CC lines: in the additional title sequence and the new line does not. So on the new server ... $add_hdr .= CC: . $email 1. PHP_EOL;$add_hdr .= CC: . $email 2. PHP_EOL;... did not work, but ... $add_hdr .= CC: . $email 1. , . $email 2. PHP_EOL;... worked. In both cases, php's mail() function returned no error, but until I put both emails, commas separated, in the same line I received the following error: 550 Messages should have one or no title Cc, not 2.Hope this will help someone. Porjo ¶ Make sure you include \r in quotes (not single quotes!) so PHP can translate that into the exact linefeed code pavel.lint at vk.com ¶ This is a small handy function I use to send emails in UTF-8. <?phpfunction mail_utf8($to, $from_user, $from_email, $subject = 'No subject), $message = "$from_user = =? UTF-8? B?. base64_encode($from_user).? =; $subject = =? UTF-8? B?. base64_encode($subject).? =$headers = From: $from_user <$from_email>\r. MIME version: 1.0 . \r . Content type: text/html; charset=UTF-8 . \rmail($to, $subject, $message, $headers? >; ABOMB ¶ I've had delivery issues from this function to Gmail, Yahoo, AOL, etc. I've used the notes here to figure that you need to be setting your return-path to a valid email to catch bounce. There are two additional distribution gotchas on it:1) The domain in the email is used in the -f option in the sendmail php.ini parameters or in the additional headers() parameters field, a valid SPF record is required for the domain (in DNS as a solid TXT record type, and add additional SPF types record if possible). Why? That's the title field that's being used to test spam.2) You should also use a domain key or DKIM. The secret here is to lock the domain name / DKIM is the sensitive case! I use Cpanel to create my domain key which automatically uses all lowercase domains in the primary creation. I found when emailing and using a camel case -f account@MyDomainHere.Com option, my key was not accepted. However it was accepted when I used -f account@mydomainhere.com. There are many other factors that can contribute to uns received messages inbox, including your own you Test attempts failed, so I recommend referencing each site's instructions and not asking me for help. These are just technical issues the couple helped my case. I hope this saves someone some time and headaches ... eeeugeneee ¶ Send a message with a minimum request from email services.<?php $encoding = utf-8$subject_preferences = array. $encoding, output-charset => $encoding, line-length => 76, line-break-chars =>| $header = Content-type: text/html; charset=.$encoding. \r; $header = From: .$from_name. <.$from_mail.> \r; $header .= MIME-Version: 1.0\r; $header .= Content-Transfer-Encoding: 8bit \r; $header .= Date: .date(r (T)). \r; $header .= iconv_mime_encode(Subject, $mail_subject, $subject_preferencesmail($mail_to, $mail_subject, $mail_message, $header);?

> John dot php at yaph dot org dot uk ¶ I found I need email@address to use the additional parameters -f$envelope_sender to send registered messages to manage my email list (Carlotta's enemies, YMMV). So to sign up mylist@example.com a PHP script I callmail (mylist-subscribe@example.com, $emailSubject, $body', -f$email_to_subscribe); jerold at pangzlab dot com ¶ Send mail with sendMail function attachments($mailTo, $message, $subject = Your audience, $fromMail = your@emialaddress.com, $fromName = from sender, $replyTo = no reply, $filePath = path){ $LE = \r; $uid = md5(uniqid(time())); $withAttachment = ($filePath !== NULL &file_exists($filePath));   if($withAttachment){ $fileName = base name ($filePath); $fileSize = file size($filePath); $handle = fopen($filePath, r); $content = fread($handle, $fileSize);   fclose ($handle); $content = chunk_split (base64_encode($content));   } $header = From: .$fromName. <.$fromMail.>$LE; $header.= Reply-To: .$replyTo. $LE; $header .= MIME version: $1.0 LE; $header .= Content type: multiple parts/mixtures; =\.$uid. \$LE$LE; $header .= This is a multiple-part message in MIME.$LE format; $header .= --.$uid. $LE; $header .= Content-type:text/html; charset=UTF-8$LE; $header .= Content-Transfer-Encoding: $7bit LE$LE; $header .= $message. LELE;   if($withAttachment){ $header .= --.$uid. $LE; $header .= Content type: application/octet-stream; name=\.$fileName. \$LE; $header .= Content layout: attachments; file name=\.$fileName. \$LE$LE; $header .= $content. LELE; $header .= --.$uid. --; } returns the message ($mailTo, $subject, $header);} rexlorenzo at gmail dot com ¶ Be careful not to put more space $headers popularity. For example, this does not work on our server:$headers = From: $from \r Bcc: $bcc \r; But this did:$headers = From: $from\rBcc: $bcc\r; Notice the removal of the surrounding space first \r. Max AT ¶ To determine message sensitivity, you must place this line in the title:<?php $headers = MIME-Version: 1.0 ; $headers .= Content-Type: text/html; charset=\iso-8859-1\$headers .= Sensitive: Personal$status = mail($to, $subject, $message.$headers);? > Possible Options:Sensitive: Normal, Personal, Private, and Company-ConfidentialThese will be recognized and processed in Outlook, Thunderbird, and others. Edward ¶ Currently my hosting service is on Godaddy. When trying to use the message function without a fifth parameters that contain -f, my message title will not work. Whenever your message title doesn't work, just try using the fifth parameters: <?php mail($to, $subject, $message, $headers, -femail.address@example.com) ?> chris at ocproducts dot com ¶ Correction to my earlier note:' confirmed necessary for qmail and postfix, perhaps also for sendmail and exim but I have not tested.'It only affects qmail. jimmytrojan009 at gmail dot com ¶ I tried many tutorials online to get mail() function working in windows, until I stumbled upon this website really grasped to change few directives in php.ini and sendmail.iniChanges requested in sendmail.inismtp_server = smtp.gmail.comsmtp_port = 587error_logfile = error.logdebug_logfile = debug.logauth_username = your-gmail-id@gmail.comauth_password = your-gmail-passwordforce_sender = your-gmail-id@gmail.comChange required in php.iniSMTP=smtp.gmail.comsmtp_port=587sendmail_from = your-gmail-id@gmail.comsendmail_path = \C:\xampp\mailtodisk\mailtodisk.exe\ - t;sendmail_path = C:\xampp\mailtodisk\mailtodisk.exe ittasks at gmail dot com ¶ When dealing with mail headlines and \r * sometimes * makes a big difference. Once our CentOs servers have been reinstalled, all the headlines such as: $headers = MIME-Version: 1.0\r; $headers.= Content type: text/html; charset=iso-8859-1\r; ... became part of the message content instead of the titleT I was able to fix this by replacing \r only with Systemx ¶ Bare LFs in SMTP Use < this?php $message = preg_replace(#(?<!\r)#si, \r, $message$headers = preg_replace(#(?<!\r)#si, \r, $headers); ?> msheldon at desertraven dot com ¶ Just a comment on some examples, and as a note to those who may not know. SMTP RFC 822 is very clear in saying that \r is the only acceptable line breaking format in the title, despite being a little vague about the message body. when many MTAs will deal with only, I've run through a lot of them will show interesting behaviors when this happens. These strict MTAs in compliance will inevitably break when the headlines are terminated with only. They will also most likely break if the contents of the message contain more than 1000 characters in a row without a \r.*Note that RFC 821 is a little clearer in determining:line A a string of ASCII characters ending in a <CRLF>. RFC 821 makes no distinction between line title and message content stream, since both are actually transmitted in the data phase. Bottom line, the best practice is to make sure to convert any bare characters in a message to \r.* The maximum total length of a line of text included <CRLF>both is 1000 characters (RFC 82 richard at richard-sumilang dot com ¶ If you are using sendmail applications from an exim package or something you do not really need to change the normal parameters PHP gives it (-t -i) as other articles are described. I just added -f myemail@example.com and it worked. One thing that I was stuck for a few hours tried to figure out why the back path was set up as the user (user run php) and not what I set it up with the option -f then I found there to strongly set the path back to the user account running commands must be in the trusted user configuration of exim! It helps to add trusted_groups well then everything works well :) - Richard Sumilang molotster on google mail com ¶ Note, the only line should not contain more than 78 characters, but is not allowed to contain more than 998 characters. The possible consequences are: Over 78 - customers are allowed to display notifications in a more difficult way to read. On 998 - clients and servers are allowed to drop messages or cause any buffer limit errors.See: part 2.1.1. f dot touchard at laposte dot net ¶ ***Encoding plain text as quoted-printable in MIME email*** If you don't want to install IMAP and use imap_8bit() to encode plain text or html messages as quoted-printable (friendly French special character encoding :-) in mime email, try this function. I have not completely tested it (as with microtime with long mail). I send html messages like 7-bit, so I have not tried yet with html. If you have good html practice, you don't really need to encode html as the picture is because it only uses 7-bit characters F.Touchard <?php function= qp_encoding($messagefor= ($i=0;? > <127; $i++)= {= = = = = $charlist[$i]= /.chr($i+128)./;= = = = = $hexlist[$i]== .strtoupper(dechex(chr($i+128$message=str_replace(=, =3D, $message);= = $message=preg_replace($CharList, $hexlist.= $message$messagelines=split(, $message);= = = $message_qp= ;= while(list(,= $line)=each($MessageLines) {= = = = if = (strlen($line)=>75 { $Pointer = 0; while ($Pointer strlen($Line)) { strlen($line))= {= = = = = = =></= strlen($Line)) { > <./127;> </CRLF> </CRLF> </CRLF> $Offset = 0;   if (preg_match(/^=(3D|([8-9A-F] {1} [0-9A-F] {1} $/, substr($Line, ($Pointer+73), $))) $Offset=-2;   substr($Line, ($Pointer+74), 3))) $Offset=-1; $Message_qp.= substr($Line, $Pointer, (75+$Offset)). =;   if (strlen($Line) - ($Pointer+75)) <= 75 { $Message_qp.= substr($Line, ($Pointer+75+$Offset)).;   break $1 Pointer+= 75+$Offset $Message_qp= $Line. $Message_qp?> bburch at bouncingpixel dot com ¶ When using php mail() functionality with IIS 6 on Windows Server 2003, check your Relay installed on SMTP Virtual Server in IIS. If you grant access to 127.0.0.1 and then set up your PHP.ini SMTP to the same IP address (along with setting up the same port 25), you should have succeeded in sending the message. I am using PHP 5.3 and have been successful with this configuration and do not need to specify sendmail_from in our php.ini file. debis at those dot rr dot com ¶ This is for Windows Server 2003, IIS 6.0 with SMTP virtual server. The problem I did not init_set for SMTP servers, I think the SMTP definition in the SMTP IIS virtual server configuration will work. When I send messages manually, this is not a problem. Additionally, $mail_sent = @mail ($to, $subject, $message, $headers); will not work but $mail_sent = mail($to, $subject, $message, $headers); did. The lack date_default_timezone_set() only causes a warning because php guess what it should be. This worked;<?php$to = 'nobody@example.com';$subject = 'theme';$message = 'hello';$headers = 'From: webmaster@example.com'. \r .   'Reply-To: webmaster@example.com'. \r .   'X-Mailer: PHP/' . phpversion();ini_set (SMTP, smtp-server.example.com); date_default_timezone_set $to, New_York $subject, $message, $headers and just so you can fix the problem, which worked when sending messages from command lines/manually. CLI works even if the php code without init_set feature will not work. You'll notice that large and large rcpt fields may seem redundant, but if both are not used, the mail's arrival field will be empty.---------------------------telnet www.example.com 25helomail from: webmaster@example.comrcpt to: someone@example.comdata: someone@example.comsubject: check look again to make sure this is my message .quit---------------------------- martin dot farrow at versacloud dot com ¶ I've noticed that on some versions of PHP occasionally mail() returns blank strings for success, rather than right or wrong. The string is empty to evaluate false.if you use the likeif (mail) structure { # do something here on success } this wont work consistently.so you need code { $ret == " || $ret) { # do something here}to get consistent results. Alex Jaspersen ¶ For qmail users, I wrote a live talk function with qmail-queue, instead of going through the sendmail wrapper used by mail(). Therefore, it allows more direct control over the message (for example, you can adjust the function to display the recipient not disclosed in the title To:). It also performs careful confirmation of the e-mail addresses that are transferred to it, making it harder for the mailer to exploit your script. Please note that this function differs from the mail() function at the word address level that must be passed as the _separate_. It is automatically included in the message header and _does not_ needs to be included in $additional_headers.$to works per user or a single address contained in a string.$message should not contain any return shipping characters – only linefeeds. No confirmation is made $additional_headers. This is mostly unnecessary because qmail will ignore any supplements To: headlines injected by a malicious user. However if you have some strange mail settings it can be a problem. The function returns false if the message cannot be authenticated or rejected by the qmail queue and returns true on success.<?phpfunction qmail_queue($to, $from, $subject, $message, $additional_headers = $cmd = /var/qmail/bin/qmail-queue; $hostname = trim(file_get_contents(/var/qmail/control/meif(is_scalar($to)) $to = array($to$e = /^[-+\\.10-9=a-z_]+@([-0-9a-z]+\\\\.) +([0-9a-z]){2.4}$/i;   if(!preg_match($e, $from)) returns the wrong;   foreach($to as $rcptpreg_match($e, $rcpt)) returns false(if(!preg_match(/^[\\040-\\\\176]$/, $subject)) returns false$dspec pipe, r), array(pipe, r)); $pipes = array(); $proc = proc_open ($cmd, $dspec, $pipes);   if(!is_resource($proc)) returns false if(!empty($additional_headersfwrite($pipes[0], $additional_headers. fwrite($pipes), To: . $to[0]); for($i = 1; $i < sizeof($to); $i++) { fwrite($pipes[0], , $to[$ifwrite($pipes[0], Theme: $subject .); fwrite($pipes[0], From: . $from $pipes . . $hostname .   fwrite($pipes[0], $message);   fwrite($pipes[0],);   fclose($pipes0fwrite($pipes[1], F . $from . \0);   foreach($to as $rcptfwrite($pipes[1], T. $rcpt. \0fwrite($pipes[1], \0);   fclose($pipes[1] proc_close($proc) == 0?> Erich at gasboysnospam dot net ¶ if your mail is failing (returns false) be aware that many servers are configured to kill mail going out with a bcc or cc header. The ideal solution is to use the smtp functionality that the server allows because of its better audit trail. Also call the message function multiple times. I just spent about four hours trying worked out what I did wrong!! Paul ¶ My mime multipart / alternative message went ok, until I switched to qmail with php .. after years of painful searching, I came across this on life with Qmail 'Gotchas' section: G.11. Back shipping/Linefeed (CRLF) non-workqmail-injection breaking lines and local injection mechanisms such as sendmail do not work as soon as the mail is injected with dos-style back/linefeed shipping (CRLF) line breaking. Unlike Sendmail, qmail requires that messages be injected locally to use unix newlines (LF only). This is a common problem with PHP scenarios. So now, I can go back to sending emails with text and html components :) yarik dot bohatsky at gmail dot com ¶ If you want to send HTML UTF-8 messages, you need to mention charset twice:1) In the message header: <?php$headers .='Content-type: text/html; charset=utf-8' .= \r;? =?>gt; 2) In html title:<?php$message ==?>gt;<html><head> <meta http-equiv=Content-Type content=text/html; charset=utf-8> <title>Fillon soutient à fond le retour d'un Grand Prix de France Fillon soutient le retour d' un Grand Prix de France</title><head><body> <p>Le Premier ministre François Fillon, passionné d' car et pilote à ses heures an apporté un soutien appuyé au retour d'un Grand Prix de France au calendrier 2013 de la Formule 1, en faisant un passage-éclair vendredi sur le circuit Paul Ricard dan le Var.</p><body></html>'In this case Outlook will also understand that messages are encrypted using UTF-8. php dot net at schrecktech dot com ¶ When emailing MIME make sure you follow the documentation with 70 characters per line ... you can end up with missing characters... and that's really hard to track ... shuitest at gmail dot com ¶ If you use mutt, do the following, /usr/bin/mutt -s '$subject' -f/dev/null -e 'set copy=no' -e 'set from = {$GLOBALS[cfg][email_from]}' -a '$attach_file_full_path' '{$GLOBALS[cfg][email_to]}' & 1; umangberi at gmail dot com ¶ Outlook 2007 seems to be a bit finicky to me to have returned shipping in the title. Therefore, any \r leads to messages that have defaulted apache messages sent through to me. As soon as I removed \r from all the headlines, the script started working fine. I hope that helps. shenyqwilliam ¶ If you are sending a large attachment, you may experience overflow messages. AFAIK, two general limits may be responsible.1. Postfix message size limit. Edit /etc/postfix/main.cf . Change the value of message_size_limit.2. Apache limits the value for the script. Edit /etc/php.ini . Change the value of memory_limit.//Then reload (or restart) Postfix and Apache.//Empirically, sending 200MB attachments that require 500MB of memory. Carefully! High memory limits can cause unexpected consequences, and are therefore not accepted. Recommended alternatives include:* Package and divide attachments into multiple emails.* Include only links to files. Recipients can download it later.* Use server (e.g. Dovecot). pangz dot lab at gmail dot com ¶ * Send email with sendMail $fileAttachment, string $mailMessage = MAIL_CONF [mailMessage], string $subject = MAIL_CONF [subject], string $toAddress = MAIL_CONF [toAddress], string $fromMail = MAIL_CONF [fromMail]): bool { $fileAttachment = trim($fileAttachment); $from = $fromMail; $pathInfo = pathinfo ($fileAttachment); $attchmentName = attachment ._date(YmdHms). ((((isset($pathInfo['extension'])) ? .. $pathInfo ['extension'] :); $attachment = chunk_split (base64_encode (file_get_contents($fileAttachment)); $boundary = PHP-mixed-.md5(time()); $boundWithPre = --.$boundary; $headers .= Answer: $from; $headers .= Content type: multiple parts/mixtures; =\.$boundary. \; $message = $boundWithPre; $message .= Content type: text/simplicity; charset=UTF-8; $message .= $mailMessage; $message = $boundWithPre; $message .= Content type: application/octet-stream; name=\.$attchmentName. \; $message .= Content-Transfer-Encoding: base64; $message .= Content layout: attachment; $message .= $attachment; $message .= $boundWithPre. --;   return messages ($toAddress, $subject, $message, $headers);} * Send email in htmlfunction sendHtmlMail(string $mailMessage = MAIL_CONF [mailMessage], string $subject = MAIL_CONF [subject], array $toAddress = MAIL_CONF [address], string $fromMail = MAIL_CONF [fromMail]): bool { $to = implode(,, $toAddress); $headers[] = 'MIME Version: 1.0'; $headers[] = 'Content type: text/html; charset=iso-8859-1'; $headers[] = 'To: '.$to; $headers[] = 'From: '.$fromMail;   return message($to, $subject, $mailMessage, implode(\r, $headers)); $headers));}