В общих настройках Joomla сайта, настройки почты находятся на вкладке «Сервер», в блоке «Настройка почты

**Отправка почты.** Этим переключателем можно выключить возможность отправки почты с сайта. Причины могут быть разные: или вообще не используется почта сайта, или сайт взломан и с него осуществляется рассылка спама и т. д. По умолчанию функция почты сайта включена.

**Способ отправки.** Данная настройка определяет способ отправки сообщений. Существует три способа отправки сообщений с сайта: «PHP Mail», «Sendmail», «SMTP».

**E-mail сайта.** В это поле вводится адрес используемого ящика электронной почты.

**Отправитель письма.** Текст, введённый здесь, будет отображаться у получателя в поле «От». Например, можно ввести «Администрация сайта такого-то» или «Интернет магазин такой то» и т. д.

**Disable Mass Mail.** Название этого переключателя переводится как «Отключить массовую рассылку сообщений». Если на вашем сайте не предусмотрена регистрация пользователей или вы не собираетесь рассылать сообщения пользователям вашего сайта, то эту опцию можно отключить. Если же нет, то вы сможете отправлять сообщения зарегистрированным пользователям или группе пользователей сайта. По умолчанию возможность массовой рассылки включена.

Это самый предпочтительный способ отправки почты. Есть некоторые тонкости. Они касаются хостинга.

PHP Mail

Это самый предпочтительный способ отправки почты, на cms Joomla, тк есть компоненты которые не работают с SMTP. Есть некоторые тонкости. Они касаются хостинга.

Если сайт находится на «боевом» хостинге, то сначала на хостинге, в своём домене нужно создать почтовый ящик. Адрес этого почтового ящика надо ввести в поле «E-mail сайта». И всё! Но адрес именно почтового ящика созданного на хостинге, а не какой-нибудь произвольный.

SMTP

Этот способ отправки почты позволяет использовать для сайта произвольный, существующий почтовый ящик. Выглядит это так.

Заполняем поля

**Авторизация на SMTP-сервере.** Подавляющее число SMTP-серверов использует авторизацию для отправки сообщений. Поэтому можно сразу включить эту опцию.

**Защита SMTP.** О поддержке защиты лучше узнать у поставщика услуг электронной почты. Поддерживается ли защищённая отправка сообщений, и какая технология (SSL, TLS) для этого используется.

**Порт SMTP-сервера.** Обязательное поле. В подавляющем большинстве случаев используется порт 25.

**Имя пользователя SMTP.** Для разных серверов может использоваться: или короткое имя пользователя, или полное. Короткое имя пользователя – это часть адреса электронной почты до символа «@». Полное имя пользователя совпадает с адресом электронной почты. О том, какой формат имени пользователя используется можно узнать у своего провайдера электронной почты. Чаще используется полное имя пользователя.

**Пароль для SMTP.** Пароль для указанного почтового ящика.

**SMTP-сервер.** Полное имя SMTP-сервера. Можно узнать у провайдера электронной почты.

**Отправка почты через mail.ru**

**E-mail сайта** - указываете действующий почтовый ящик на mail.ru, например - [test@mail.ru](mailto:test@mail.ru)  
**Отправитель письма** - имя отправителя (или название сайта, к примеру), которое будет отображаться у адресата в поле "От.."  
**Авторизация на SMTP-сервере** - ДА  
**Защита SMTP** - SSL  
**Порт SMTP-сервера** - 465  
**Имя пользователя SMTP** - это ваш действующий email на mail.ru, например - [test@mail.ru](mailto:test@mail.ru)  
**Пароль для SMTP** - пароль от вашего почтового ящика  
**SMTP-сервер** - smtp.mail.ru

**Отправка почты через gmail.com**

**E-mail сайта** - указываете действующий почтовый ящик на mail.ru, например - [test@gmail.com](mailto:test@gmail.com)  
**Отправитель письма** - имя отправителя (или название сайта, к примеру), которое будет отображаться у адресата в поле "От.."  
**Авторизация на SMTP-сервере** - ДА  
**Защита SMTP** - SSL  
**Порт SMTP-сервера** - 465  
**Имя пользователя SMTP** - это ваш действующий email на gmail.com, например - [test@gmail.com](mailto:test@gmail.com)  
**Пароль для SMTP** - пароль от вашего почтового ящика  
**SMTP-сервер** - smtp.gmail.com

**Отправка почты через Yandex**

**E-mail сайта** - указываете действующий почтовый ящик на yandex.ru, например - [test@yandex.ru](mailto:test@yandex.ru)  
**Отправитель письма** - имя отправителя (или название сайта, к примеру), которое будет отображаться у адресата в поле "От.."  
**Авторизация на SMTP-сервере** - ДА  
**Защита SMTP** - SSL  
**Порт SMTP-сервера** - 465  
**Имя пользователя SMTP** - это ваш действующий email на aol.com, например - [test@yandex.ru](mailto:test@yandex.ru)  
**Пароль для SMTP** - пароль от вашего почтового ящика  
**SMTP-сервер** - smtp.yandex.ru

**Отправка почты через Rambler**

**E-mail сайта** - указываете действующий почтовый ящик на yandex.ru, например - [test@rambler.ru](mailto:test@rambler.ru)  
**Отправитель письма** - имя отправителя (или название сайта, к примеру), которое будет отображаться у адресата в поле "От.."  
**Авторизация на SMTP-сервере** - ДА  
**Защита SMTP** - SSL  
**Порт SMTP-сервера** - 465  
**Имя пользователя SMTP** - это ваш действующий email на aol.com, например - [test@rambler.ru](mailto:test@rambler.ru)  
**Пароль для SMTP** - пароль от вашего почтового ящика  
**SMTP-сервер** - smtp.rambler.ru

**Порт SMTP-сервера** лучше проверить перед настройкой \*

Часто встречающиеся ошибки

Joomla smtp connect failed

Ошибка вида «joomla smtp connect failed» зачастую вызвана неправильным заполнением поля «SMTP Security». В этом случае необходимо уточнить, как их рекомендует заполнять ваш почтовый сервер.

Ошибки авторизации

Ошибки связанные с авторизацией возникают из-за неправильного заполнения юзернейма и пароля. Обратите внимание в яндексе в поле Username мы указываем содержимое нашего почтового ящика до собаки, в других почтовых серверах возможно необходимо указывать почту целиком.

Sendmail

Этот способ отправки почты устарел. Он сохранился ещё с тех времен, когда на хостингах отправка сообщений осуществлялась программой Sendmail. Сейчас этот способ устарел и используется крайне редко. Поэтому мы не будем его рассматривать.

Подробнее о том как работает почта на joomla

### Объект для работы с почтой

Ссылка на глобальный объект электронной почты JMail получается с помощью класса JFactory. Это создание нашего объекта JMail.

$mailer =& JFactory::getMailer();

В результате мы получим объект класса JMail, который объявлен по адресу /libraries/joomla/mail/mail.PHP и является потомком класса PHPMailer. Все необходимые файлы подключаются сами по необходимости.

**Библиотека PHPMailer** — Это внешняя библиотека для отправки почты расположенная по адресу /libraries/phpmailer. Служит для передачи электронной почты, позволяющий, в частности, вкладывать в сообщение несколько файлов, посылать копии CC и BCC, устанавливать атрибут REPLY-TO, использовать в сообщениях [HTML](http://joomla-book.ru/term/html), работать с резервными SMTP-серверами, переносить строки по словам. Он может посылать сообщения с через sendmail, PHP mail() или непосредственно по протоколу SMTP.

### Установка отправителя

Отправитель письма устанавливается с помощью метода setSender. Функция принимает массив с адресом электронной почты и именем в качестве аргумента. Мы получили имя и адрес электронной почты из глобальной конфигурации сайта. Они устанавливаются в административной части сайта (Общие настройки / Настройки почты). От этого имени будет отправлена почта.

$config =& JFactory::getConfig();

$sender = array($config->getValue('config.mailfrom'), $config->getValue('config.fromname') );

$mailer->setSender($sender);

### Установка получателя

Вы устанавливаете получатель электронной почты с помощью метода addRecipient. Для того что-бы задать e-mailадрес текущего авторизованного пользователя, мы возьмём его из объекта пользователя.

$user =& JFactory::getUser();

$recipient = $user->email;

$mailer->addRecipient($recipient);

Если у нас есть несколько адресатов, мы можем отправить их в виде массива.

$recipient = array('person1@domain.com', 'person2@domain.com', 'person3@domain.com');

$mailer->addRecipient($recipient);

### Создание письма

Мы должны указать тему письма и создать содержимое. Тема письма устанавливается с помощью метода JMail->setSubject.

Простой способ создания сообщения в виде строки текста. Используйте функцию JMail->setBody для добавления сообщения к письму. Вы можете так же прикрепить файл JMail->addAttachment. В качестве аргумента может быть имя одного файла или массив имён нескольких.

$body = "Your body string\nin double quotes if you want to parse the \nnewlines etc";

$mailer->setSubject('Your subject string');

$mailer->setBody($body);

// Optional file attached

$mailer->addAttachment(PATH\_COMPONENT.DS.'assets'.DS.'document.pdf');

#### HTML-формат письма

Если вы предпочитаете HTML-формат письма, вы должны это указать. Это делается с помощью метода JMail->IsHTML. Темы и любые вложения обрабатываются, как указано выше, за исключением изображений встроенных в HTML. Они обрабатываются методом JMail->AddEmbeddedImage.

$body = '<h2>Our mail</h2>'

.'<div>A message to our dear readers'

.'<img src="cid:logo\_id" alt="logo"/></div>';

$mailer->isHTML(true);

$mailer->setBody($body);

// Optionally add embedded image

$mailer->AddEmbeddedImage(PATH\_COMPONENT.DS.'assets'.DS.'logo128.jpg', 'logo\_id', 'logo.jpg', 'base64', 'image/jpeg');

Как правило, вы отправляете все изображения на сервер и вставляете обычным HTML-тегом изображения, для уменьшения размера почты и времени отправки.

### Дополнительные опции для писем

Это редкоиспользуемые методы, но в редких случаях бывают необходимы и добавляют гибкости при отправке почты. Всем перечисленным методам ниже можно указывать несколько значений в виде массива или строкой.

* **JMail->addReplyTo** — добавить адрес почты для ответа. Пример одного адреса — array(’person1@domain.com’, ’username’). Несколько адресов соответственно — это вложенный массив.
* **JMail->addBCC** — Добавить получателей слепой точной копии (BCC).
* **JMail->addCC** — Добавить получателей точной копии (CC).
* **JMail->addAttachment** — Добавить файл(ы) к письму. Указывать нужно абсолютный путь до файла на вашем сервере.

Т.к JMail является потомком PHPMailer, то доступны все методы и из него, прочитать о нем подробно можно по ссылкам ниже.

### Отправка почты

Письмо отправляется с помощью метода Send. Он возвращает true при успешной отправке или объект ошибки JError. Все письма будут отправляться в кодировке UTF-8. Это необходимо иметь ввиду, иначе письмо будет затруднительно прочитать.

$send =& $mailer->Send();

if ($send !== true) {

echo 'Error sending email: '.$send->message;

} else {

echo 'Mail sent';

}

Во всяком случае лучше всего написать собственный обработчик ошибок с помощью класса JError.

### Альтернативный способ отправки писем

Существует класс JUtility, который без явного создания объекта поможет отправить письмо. Это просто немного упрощенная версия того что описано выше. Делает это примерно так.

jimport('joomla.utilities.utility');

$fromEmail = 'admin@mysite.ru'; // отправитель, почта

$fromName = 'Test Server'; // отправитель, имя

$email = 'admin@example.com'; // кому

$subject = 'Test Mail Message'; // тема письма

$convertedBody = '<p>TEST MESSAGE</p>'; // сообщение в html

$filename = JPATH\_BASE.DS.'images'.DS.'stories'.DS.'articles.jpg'; // прикрепляемый файл

// отправляем

JUtility::sendMail($fromEmail, $fromName, $email, $subject, $convertedBody, true, null, null, $filename );

### Почему не нужно использовать функцию mail()?

Отправляя письма через mail(), вы сами должны следить за всеми заголовками, конвертировать необходимые вещи через base64, учитывать кодировку и многое другое. Более того, JMail может отправлять письма различными способами: SMTP, sendmail и функция mail(). Настроить это можно в Общих настройках / Сервер.

Так что лучше использовать готовый набор библиотек, где все уже учтено и отлажено и готово к работе.

### Примеры в компонентах Joomla

Объект JMail используется для отправки почты в компоненте контактов (joomla/components/com\_contact/controller.PHP).

http://www.dia.org.ua/stati/faq/119-nastrojki-smtp-na-joomla-3-dla-razlichnyh-pochtovyh-serverov.html

http://www.aitishnik.ru/joomla-3-nastroyka-elektronnoy-pochti-sayta.html

http://joomla-book.ru/development/classes/jmail-send-email-message