BatPost

Установка, начальная настройка и запуск сервера

Оглавление

Глава 1

1 Введение

Данное руководство поможет Вам установить почтовый сервер BatPost и настроить его "с нуля". Даже не обладая специальными знаниями вы сможете с помощью пошаговых инструкций получить работоспособный почтовый сервер в базовой конфигурации.

Когда потребуются специальные настройки у Вас уже будет твердая почва под ногами для смелых экспериментов. Можно будет сосредоточится на тонкостях, вместо того чтобы думать сразу обо всем. Изменяя небольшое количество настроек, Вы всегда сможете вернуть все "как было", если что-то пойдет не так.

Настройка почтового сервера может оказаться непростой задачей, если Вы никогда раньше этим не занимались. Возникает масса вопросов ответы на которые не так легко найти. Из данного руководства Вы сможете узнать базовые сведения об электронной почте необходимые для понимания принципов работы почтового сервера.

Почтовый сервер может использоваться в самых разных условиях. Подключение к Интернету может быть постоянным или периодическим (по телефонной линии), а может и вообще отсутствовать. Поэтому перед установкой сервера нужно решить для себя ряд важных вопросов. Это позволит сэкономить время на переконфигурирование сервера.

В разделе "Вопросы и ответы" собраны сведения о наиболее часто возникающих проблемах и даны рекомендацию по их решению.

Глава 2

2 Базовые сведения об электронной почте

Электронная почта (e-mail) была в числе первых услуг предоставляемых сетью Интернет. В то время сообщения были чисто текстовыми и сама возможность их передачи на огромные расстояния за считанные секунды впечатляла. Это было намного быстрее и удобнее телеграмм и уж тем более обычных писем. Но в то же время стало очевидным, что обычного текста часто совершенно не достаточно для многих применений. С тех пор формат сообщений неоднократно расширялся и теперь мы можем оформлять сообщения по своему вкусу разными шрифтами и цветом, вставлять в текст картинки, прикреплять к сообщению файлы, использовать шифрование и электронную подпись.

Все это стало возможным благодаря развитию почтовых клиентов. Из простейшего текстового редактора с функцией отправки и приема они превратились в мощные программы с массой полезных возможностей. Теперь сообщения можно красочно оформлять в HTML формате, можно использовать шаблоны сообщений для ускорения работы, принимаемые сообщения можно автоматически сортировать по папкам, появились мощные адресные книги. Но что же происходит когда мы нажимаем в почтовом клиенте кнопку "Отправить" или "Принять" сообщение? Вот тут и вступает в дело почтовый сервер.

Почтовый клиент ничего не знает о том как доставить сообщение адресату. Его дело грамотно оформить сообщение, прописать правильные адреса отправителя и получателя и передать сообщение почтовому серверу. Входящие сообщения он также получает от почтового сервера. Почтовый же сервер совершенно не интересуется оформлением сообщения, зато знает как доставить его по адресу. В этом смысле ситуация очень похожа на реальное почтовое отделение и приходящих туда людей. В роли почтового отделения выступает почтовый сервер, а в роли людей туда приходящих выступают почтовые клиенты.

Казалось бы, зачем вообще нужен почтовый сервер? Почему почтовые клиенты не могли бы сразу отправить письмо получателю? Ведь Интернет, в отличие от реальной почты, не имеет территориальных границ и передать информацию на другой конец земного шара не сложнее чем в соседнюю комнату. Но представьте себе, что Вы хотите отправить письмо, а адресат в это время уже лег спать или просто выключил компьютер. Или Вам пришло письмо, а Вы в этот момент в отпуске. Вряд ли Вам или тому кто Вам пишет захочется дожидаться возможности передать письмо, тем более что наступить эта возможность может очень не скоро. Гораздо проще передать сообщение своему почтовому серверу, а он уже позаботится о том, чтобы оно было доставлено. Причем доставлено не самому адресату, а его почтовому серверу, который заботливо положит сообщение в ящик пользователя. Там оно и будет дожидаться когда пользователь проверит свою входящую почту.

Но как же узнать какому серверу нужно передать письма адресованные данному пользователю? Ответ на это вопрос дает адрес назначения. Почтовый адрес состоит из двух частей разделенных символом "@". То что находится после этого символа является именем домена, а то что до - именем пользователя внутри этого домена. Чтобы доставить письмо по назначению необходимо определить какой сервер отвечает за почту для данного домена. Для этого используется DNS (Domain

Name System - Система Доменных Имен), каждому почтовому домену там может быть приписан набор серверов, которые отвечают за почту в него адресованную.

2.1 Почтовый клиент

Современный почтовый клиент предлагают множество функций, однако с точки зрения электронной почты необходимо, чтобы он умел принимать и отправлять почту. В зависимости от типа сервера и качества Интернет канала могут применяться следующие стили работы с почтой:

1. Соединяемся с Интернетом, скачиваем письма, отключаемся от Интернета. Просматриваем входящую почту, отвечаем на письма, но ответы пока храним локально.
   Соединяемся с Интернетом, отправляем исходящую почту, возможно скачиваем новую входящую, отключаемся от Интернета.
   Такой стиль работы характерен при отсутствии постоянного подключения к Интернету, например, через модем по телефонной линии. При этом для приема почты обычно используется протокол POP3, хотя возможно и использование IMAP.
2. Подключаемся к серверу, скачиваем входящую почту, отключаемся от сервера. Просматриваем письма, отвечаем на них и ответы сразу отправляем.
   Такой стиль характерен при наличии постоянного подключения к Интернету. Для приема почты могут использоваться как протокол POP3, так и протокол IMAP.
3. Подключение к серверу постоянное.
   Вся почта хранится на сервере и скачивается по необходимости. Сервер уведомляет нас о приходе новых сообщений.
   Для такого стиля работы необходим хороший Интернет канал, например локальная сеть. Также должен использоваться протокол IMAP.

Каждый из перечисленных стилей имеет как свои преимущества, так и недостатки. Так, например, первый метод самый нетребовательный к качеству Интернет канала, но привязан к данному рабочему месту, так как вся почта хранится локально. Третий метод позволяет работать с одним и тем же ящиком из нескольких мест, но зато очень требователен к каналу, постоянно использует ресурсы сервера и поэтому зависит от его возможностей и загруженности.

2.2 Почтовый сервер

Почтовый сервер сам по себе не обладает пользовательским интерфейсом. Он только предоставляет набор протоколов, при помощи которых с ним можно общаться.

POP3

Предназначен для проверки входящих сообщений. Хотя у пользователя может быть много папок на сервере, в данном протоколе они не поддерживаются и, по сути, работа ведется только с папкой INBOX. Протокол очень прост и поддерживается всеми почтовыми клиентами. Он позволят получить список сообщений на сервере с их размерами, можно скачать или удалить указанное сообщение. Также возможно частичное скачивание сообщений (заголовок сообщения и указанное количество строк тела письма).

IMAP

Также позволяет проверять входящие сообщения, но предоставляет значительно более широкие возможности. Появился в значительной мере как развитие протокола POP3, для него характерно перенесение многих функций с клиента на сервер. Это позволяет использовать так называемые "тонкие" клиенты, которые сами не предоставляют особых возможностей, а полагаются на функциональность сервера.
Пользователь может иметь на сервере сложную структуру папок, а уже в них хранятся сообщения. Сообщения разбираются сервером на составные части и можно получить доступ к любой из этих частей. Так, например, можно скачать только прикрепленный к сообщению файл.
В отличие от протокола POP3, можно работать сразу с набором сообщений и запрашивать только нужную информацию. Так, например, можно запросить отправителей, получателей и размеры всех сообщений с 5-го по 10-е включительно.
Также есть возможность поиска сообщений в папке с указанием сложных условий. К примеру, можно найти все сообщения от данного отправителя за последний месяц.
Помимо самого стандартного протокола, было также принято множество расширений, еще более расширяющих возможности протокола. Такое обилие возможностей привело к тому,что протокол получился достаточно сложным. Не все почтовые клиенты и сервера его поддерживают, и практически никто не поддерживает в полной мере.

SMTP

Применяется для отправки сообщений от пользователя на сервер, а также для передачи сообщений от одного сервера к другому. При этом отдельно указываются адрес отправителя и адреса каждого из получателей. Это позволяет вернуть ошибку на любом из этапов. Так, например, сервер может отказаться принять сообщение от данного отправителя (если, к примеру, такой адрес не существует), либо для кого-то получателей (если такой пользователь не найден).
Первоначально протокол SMTP не предусматривал возможности авторизации, но в связи с появлением спама (нежелательных сообщений часто рекламного характера) такая возможность была добавлена.
При отправке сообщений на внешние адреса часто требуется авторизация, иначе была бы возможна рассылка сообщений через данный сервер (open relay), что открывает возможности для злоупотребления. Вместо авторизации сервер может вводить ограничения по IP адресам. Так, например, отправка на внешние адреса может быть позволена только из внутренней подсети.
Обмен сообщениями между серверами чаще всего происходит без авторизации , так как сервера обычно ничего друг о друге не знают. Ограничением является то, что сервер-получатель должен быть конечным пунктом для сообщения (по крайней мере, с точки зрения сервера-отправителя). Исключением является случай, когда небольшая организация использует сервер провайдера как промежуточный для отсылки своих сообщений. В этом случае должна использоваться либо авторизация, либо какие-то другие ограничения (например на основе IP адреса).

2.3 DNS

Электронный адрес (e-mail) дает серверу полную информацию о том, кому предназначено данное сообщение и как его надо доставить. Предположим, что серверу надо доставить письмо по адресу user@example.com. Здесь "example.com" - имя почтового домена, а "user" - имя пользователя внутри этого домена. Прежде всего нужно определить какой сервер (или сервера) отвечает за почту адресованную в домен "example.com", затем соединиться с ним по протоколу SMTP и передать письмо, а дальше уже будет его задача доставить письмо пользователю. Для того, чтобы найти ответственный сервер используется DNS (Domain Name System - Система Доменных Имен). DNS-сервер хранит для имен доменов наборы записей различных типов и по запросу возвращает записи запрошенных типов для заданного домена. С точки зрения электронной почты нас интересуют записи типа MX и записи типа A.

Записи типа MX

В записях типа MX хранится число определяющее приоритет данной записи после которого следует имя хоста, отвечающего за почту для данного домена. Хосты с меньшим значением приоритета должны быть использованы раньше других. Хосты с одинаковым приоритетом могут использоваться в любом порядке.
Если сервер обнаруживает свое собственно имя в списке хостов отвечающих за почту для данного домена, то он должен соединяться только с хостами с меньшим значением приоритета, чем его собственный. Это необходимо, чтобы избежать ситуации когда письмо начинает ходить "по кругу".
Так, например, в нашем случае могли быть получены следующие MX записи:

```
example.com  MX  10 mx1.example.com
example.com  MX  20 mx2.example.com
example.com  MX  20 mx3.example.com
```

В данном случае сервер должен сначала попытаться отдать почту хосту "mx1.example.com", а в случае неудачи хостам "mx2.example.com" и "mx3.example.com" (в любом порядке).

Записи типа A

Записи типа MX дают нам только имя хоста, а для того чтобы определить его IP-адрес используются записи типа A. В записях типа A хранится IP-адрес данного домена. Для одного домена может быть несколько A записей, т.е. один и тот же домен может иметь несколько IP-адресов. При отправке почты мы должны попытаться соединиться с каждым из IP-адресов (в любом порядке).
В нашем случае мы могли бы получить для хоста "mx1.example.com" следующие A записи:

```
mx1.example.com  A  192.5.19.31
mx1.example.com  A  192.5.25.5
```

Тогда мы должны были бы попытаться соединиться с адресами 192.5.19.31 и 192.5.25.5 в любом порядке и, в случае неудачи, перейти к следующей MX записи.

Такая организация записей позволяет реализовывать:

- Кластер серверов. Крупные почтовые порталы для нормальной работы могут параллельно использовать несколько серверов (кластер). Это повышает надежность системы - в случае отказа одного сервера, остальные продолжают работать. Также это распределяет нагрузку между серверами, что увеличивает быстродействие системы.

Такая структура может быть организована при помощи нескольких MX записей (с одинаковыми приоритетами), либо при помощи нескольких A записей. Поскольку почтовые сервера обычно обрабатывают записи в том порядке в котором их вернул DNS-сервер важно чтобы он возвращал их в случайном порядке, тогда нагрузка будет распределяться равномерно.

- Резервирование серверов. В такой структуре есть главный сервер, который реально хранит почту и резервные сервера, которые могут ее временно принять, если главный сервер не отвечает. Для того, чтобы это реализовать заводится несколько MX записей с разными приоритетами. Главный сервер имеет самое меньшее значение приоритета (с ним будут соединяться в первую очередь), а резервные сервера имеют большее значение приоритета (с ними будут соединяться, если главный сервер не отвечает).
  В нашем примере "mx1.example.com" мог бы быть главным сервером, а "mx2.example.com" и "mx3.example.com" могли бы быть резервными серверами.

Замечание. Последняя схема может также применяться в случае, когда компания решает сама обслуживать свой почтовый домен, который до этого обслуживался сервером провайдера. Нужно добавить запись MX, ссылающуюся на сервер внутри компании. Причем значение приоритета у нее должно быть меньше (главный сервер), чем у сервера провайдера (резервный сервер).
Это позволит сравнительно безболезненно настраивать сервер внутри компании - в крайнем случае почта все равно будет принята сервером провайдера. Когда же настройка будет завершена и внутренний сервер перейдет в полностью рабочий режим MX на сервер провайдера можно будет убрать (или оставить в резервных целях).

Глава 3

3 Установка сервера

В этом разделе описана последовательность установки и первоначальной настройки сервера. Даны пошаговые инструкции позволяющие получит работоспособный сервер в базовой конфигурации. Также даны рекомендации по дальнейшей настройке сервера.

3.1 Перед установкой

Поскольку сервер может работать в нескольких режимах, желательно перед его установкой и настройкой определиться, что же именно от него требуется. Некоторые технические данные (IP-адрес DNS, параметры почтового сервера провайдера) Вы сможете получить у вашего Интернет-провайдера.

При рассылке писем почтовые сервера широко используют систему доменных имен (DNS). DNS позволяет серверу определить, какой почтовый сервер отвечает за прием почты для данного домена (для этого служат записи MX). Таких серверов может быть несколько и каждому из них выставлен приоритет. Это позволяет значительно повысить общую надежность почтовой подсистемы, так как в случае отказа одного сервера почту может временно принять другой. Чтобы еще больше повысить гибкость, в записях MX указываются не IP адреса почтовых серверов, а имена их хостов. При помощи DNS можно определить какие IP-адреса соответствуют данному имени хоста (для этого служат записи A).

Прежде всего нужно решить будет ли компьютер, на который планируется установить сервер, иметь постоянный доступ в Интернет или подключаться к нему лишь периодически. Этот вопрос очень важен, так как для доступа к DNS чаще всего требуется постоянное подключение к Интернету. Бывают случаи, когда сервер не имеет доступа к Интернету, но тем не менее имеет доступ к DNS. Это возможно когда сервер используется исключительно в локальной сети для каких-то внутренних нужд, например, для организации документооборота внутри компании. Хотя в данном случае доступ к DNS может быть вообще необязателен, так как все операции могут происходить только в пределах одного единственного сервера.

Таким образом возможны три сценария работы почтового севера:
- имея постоянный доступ к Интернету и DNS
- периодически подключаясь к Интернету, используя dial-up
- вообще не имея доступа к Интернету и работая только в локальной сети

BatPost достаточно гибок, чтобы справиться с любым из данных сценариев. Достигается это изменением всего лишь нескольких настроек сервера. Рассмотрим каждый из вариантов отдельно и в каждом случае дадим некоторые рекомендации.

Постоянная связь с Интернетом
Если связь с Интернетом постоянная, то лучше чтобы сервер сам рассылал почту, используя DNS. В данном случае Вам надо выяснить IP адрес DNS сервера. Эти данные можно получить у системного администратора или у Интернет провайдера. Однако в случае самостоятельной рассылки может потребоваться завести свой собственный почтовый домен, так как некоторые сервера производят расширенные проверки и отказываются принимать почту от серверов, не зарегистрированных в

DNS. Если нет планов регистрировать свой домен, то рассылку можно производить через сервер провайдера или любой другой промежуточный сервер (relay).

Периодическое подключение к Интернету
Если сервер подключается к Интернету на непродолжительные промежутки времени, то использование DNS и прямая рассылка писем становятся неэффективными. В этом случае гораздо удобнее использовать сервер провайдера (или любой другой подходящий сервер) как для получения входящей, так и для рассылки исходящей почты.
Для получения входящей почты могут использоваться технологии Remote POP (RPOP), либо ETRN/ATRN. В первом случае BatPost будет подключаться к внешнему серверу по протоколу POP3 и забирать сообщения. Затем заголовки этих сообщений будут анализироваться, чтобы определить адресатов. Во втором случае будет использоваться специальный протокол (ETRN/ATRN) для того чтобы забрать сообщения из очереди отправки внешнего сервера. Второй вариант более предпочтителен, так как в этом случае получатели сообщения определяются однозначно, но он требует чтобы внешний сервер был настроен особым образом, что не всегда возможно.
Для отсылки исходящей почты, сервер провайдера будет использоваться в качестве промежуточного сервера (relay).
Этот случай является наиболее сложным по количеству настроек. В любом случае все подробности нужно обсудить с провайдером, и получить у него необходимые настроечные данные.

Связь с Интернетом отсутствует
В случае, когда связь с Интернетом вообще не предполагается, в локальной сети может быть внутренний DNS. Он позволит иметь во внутренней сети несколько почтовых серверов. При этом записи DNS специфичны для данной сети и не имеют смысла вне ее. Гораздо чаще DNS тоже отсутствует и тогда вся работа с письмами будет вестись в пределах одного почтового сервера. В этом случае, хотя реально почта рассылаться вообще не будет, в качестве способа рассылки нужно указать использование промежуточного сервера (relay). Настройки промежуточного сервера в данной ситуации несущественны.

Установка сервера
Когда все необходимые решения будут приняты и настроечные данные будут известны, можно приступать к установке сервера. Процедура установки BatPost на Ваш компьютер состоит из следующих этапов:
1. Установка сервера
2. Начальная настройка сервера
3. Создание глобальных администраторов
4. Создание доменов
5. Создание пользователей
6. Запуск и проверка работоспособности

В случае, когда предполагается, что сервер будет периодически подключаться к Интернету, необходимо также произвести некоторые дополнительные настройки.

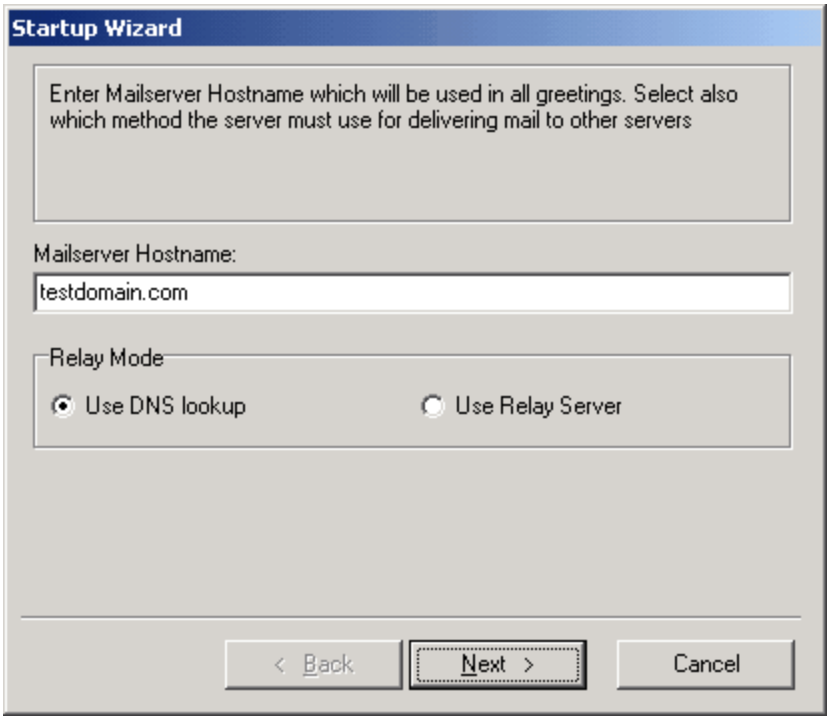
3.2 Установка сервера

Дистрибутив сервера представляет собой исполняемый файл BatPost_v221xx.exe. В процессе установки Вам будет предложено выбрать каталог, в который будут скопированы файлы, и программную группу, в которую будут установлены иконки. В простейшем случае можно принять значения, предлагаемые по умолчанию, и ограничиться только нажиманием кнопки "Next". Помимо полной установки есть выборочная, которая позволяет выбрать компоненты, которые будут установлены. Серверные компоненты необходимы для работы самого сервера, а административные - для его настройки, конфигурирования и управления. Если нужно удаленно конфигурировать сервер, то на компьютере, с которого планируется удаленное конфигурирование, нужно будет установить только административные компоненты.

Если использовались параметры, предложенные по умолчанию, то после установки появится программная группа "BatPost", в которой будут иконки для запуска: Конфигуратора, Монитора и процедуры удаления сервера с Вашего компьютера.

3.3 Начальная настройка сервера

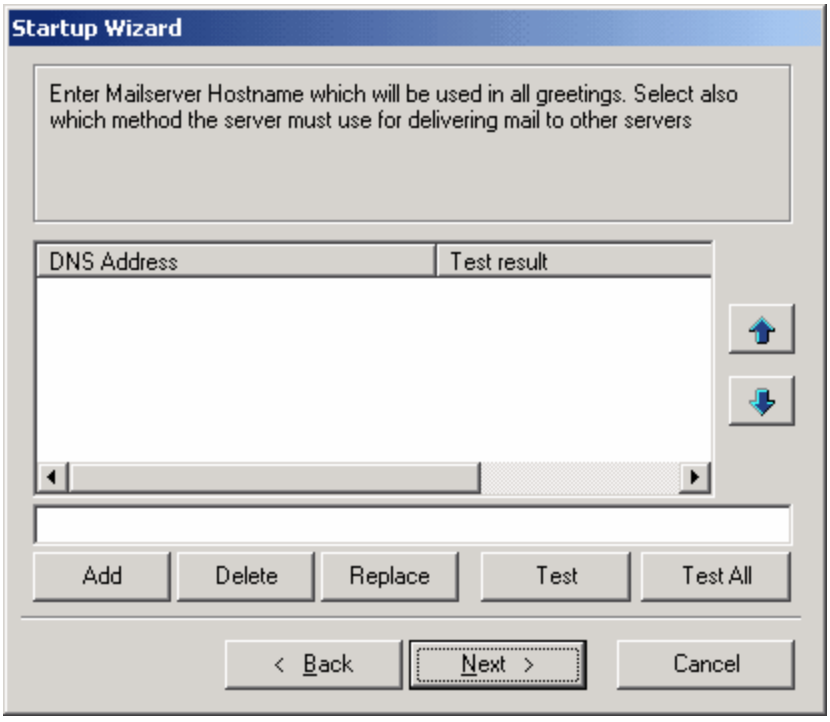
После успешной установки будет запущен мастер начальной настройки сервера. Он позволит задать основные параметры, необходимые для нормальной работы сервера.



В первом окне мастера начальной настройки нужно задать имя сервера, под которым он будет представляться в сети. Это имя обычно совпадает с именем главного домена почтового сервера. Также нужно выбрать метод рассылки исходящих сообщений. По поводу выбора метода можно прочитать в разделе "

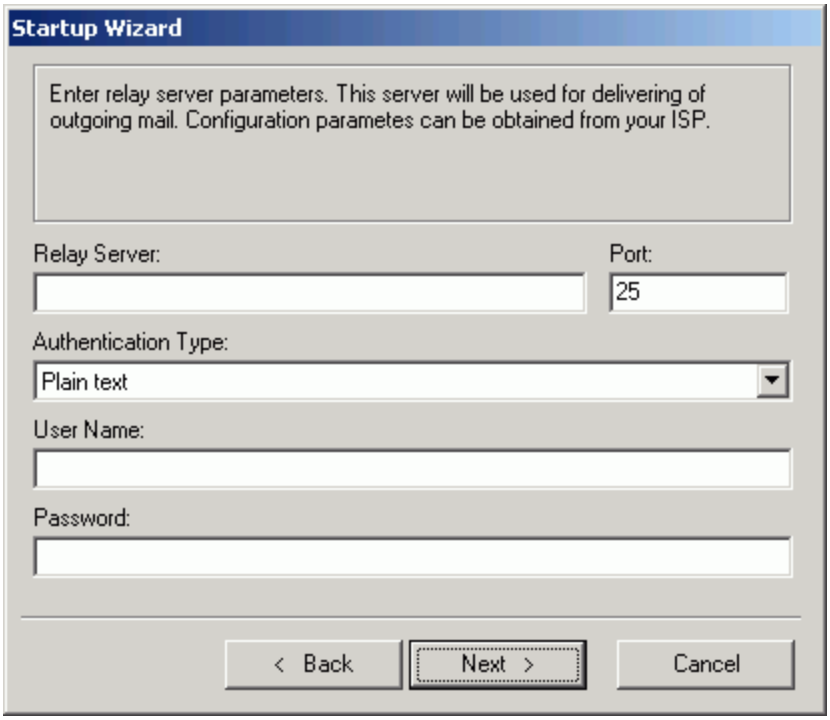
Перед установкой".

В случае, когда для рассылки почты будет использоваться DNS появляется окно позволяющее указать DNS сервера:



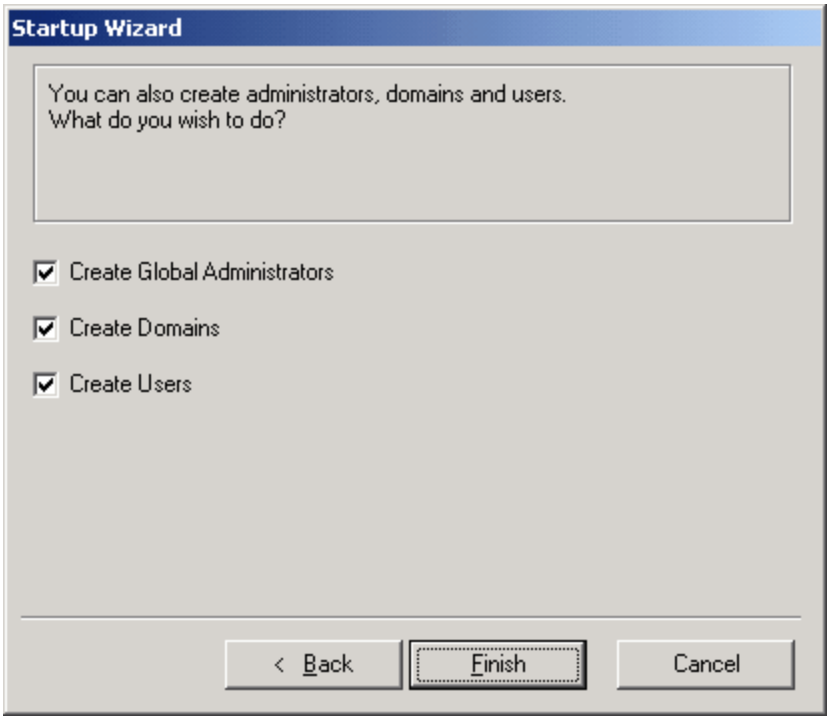
BatPost позволяет указать несколько DNS серверов. При работе он будет стараться использовать наиболее быстрый DNS сервер. В данном окне можно также протестировать работоспособность указанных DNS серверов.

Если же для рассылки исходящей почты был выбран промежуточный сервер (relay), то появится окно для настройки его параметров:



В данном окне можно указать имя хоста или IP адрес промежуточного сервера (relay), номер порта для исходящих SMTP сессий, а также тип и параметры аутентификации. Если аутентификация не требуется, то имя пользователя и пароль нужно оставить пустыми.

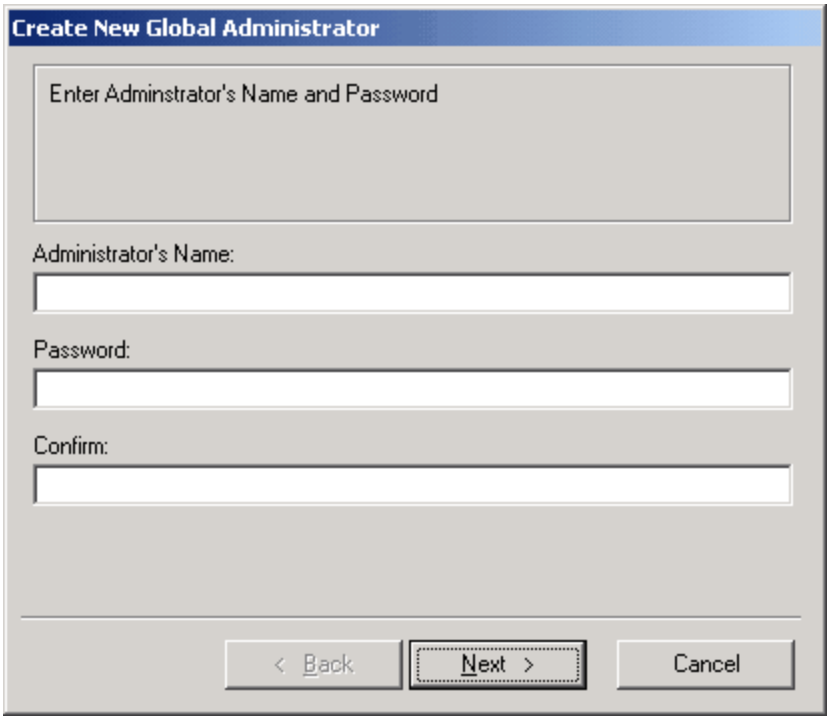
После нажатия на кнопку Next появится заключительное окно мастера начальной настройки:



В данном окне можно выбрать нужно ли создавать глобальных администраторов, домены и пользователей.

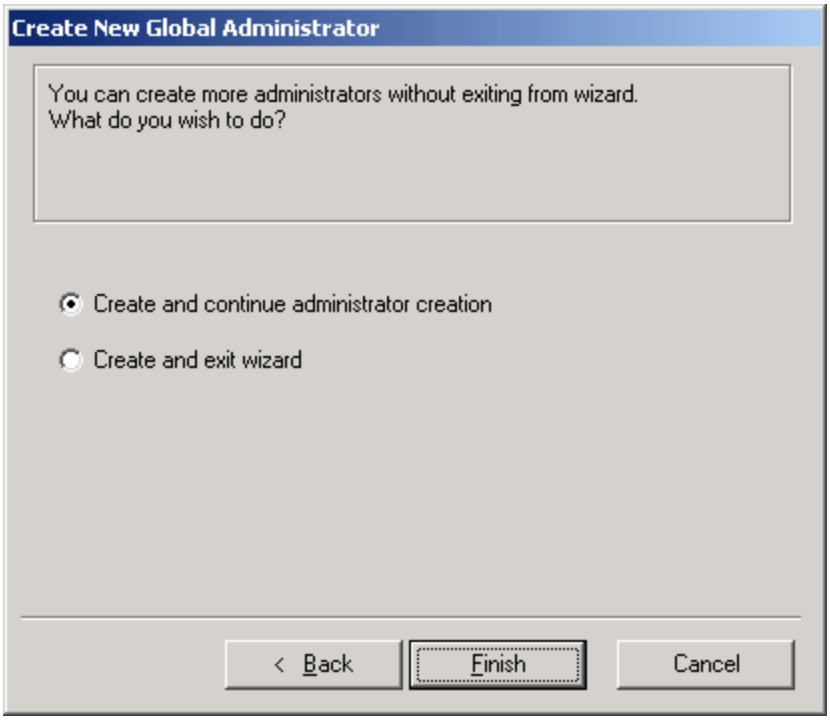
3.4 Создание глобальных администраторов

Глобальные администраторы имеют право осуществлять мониторинг сервера и менять любые его настройки. Данные операции можно делать не только локально, но и с другого компьютера. Если предполагается удаленная работа с сервером, то нужно создать хотя бы одного глобального администратора.



Для глобального администратора требуется указать только имя и пароль.

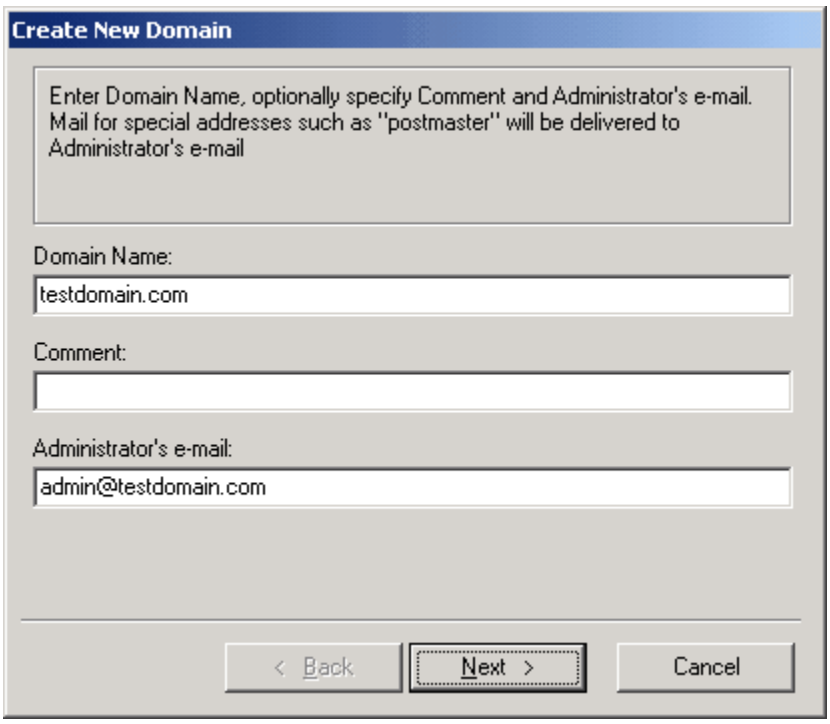
Следующее окно позволяет либо завершить мастер, либо продолжить создание глобальных администраторов:



3.5 Создание доменов

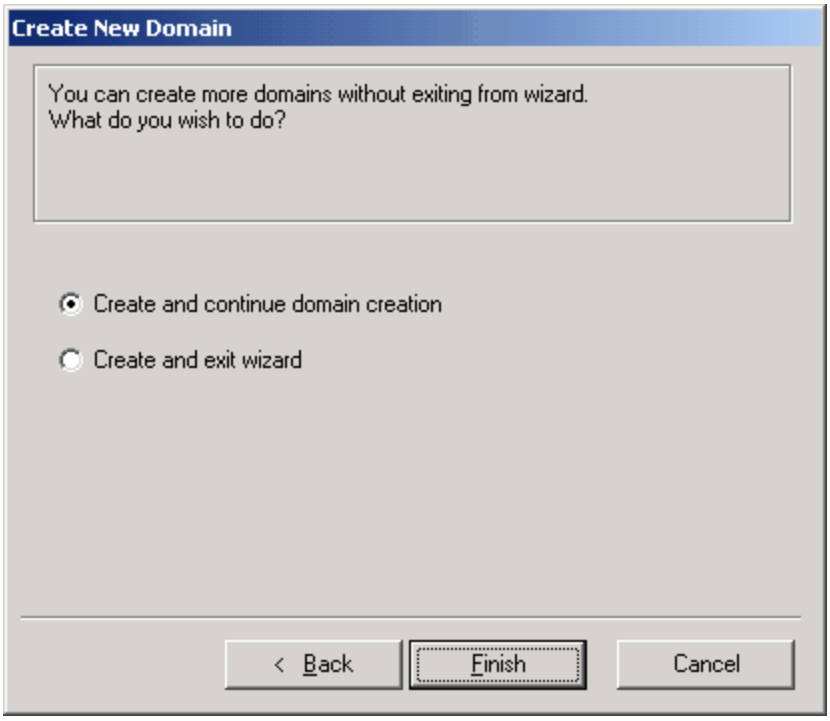
Для почтового сервера домен является сущностью, объединяющей множество пользователей. С точки зрения Интернета домен является символическим адресом ресурса в сети. Для нас же эта двойственность проявляется в том, что почтовый сервер должен быть правильно прописан в DNS. То есть для данного имени домена должна существовать одна или более записей MX. Для каждого хоста прописанного в записях MX должна существовать хотя бы одна запись A, позволяющая определить его IP адрес. Это позволит другим серверам правильно присылать почту адресованную в наш домен.

На сервере домен должен быть прописан в точности под тем же самым именем, что и в DNS.



При создании домена нужно указать его имя, необязательный комментарий и почтовый адрес администратора. Правила создания почтовых подсистем требуют, чтобы существовали некоторые служебные адреса, такие как POSTMASTER. Можно создать на сервере пользователей с нужными именами, но чаще всего достаточно перенаправить письма, приходящие на служебные адреса администратору.

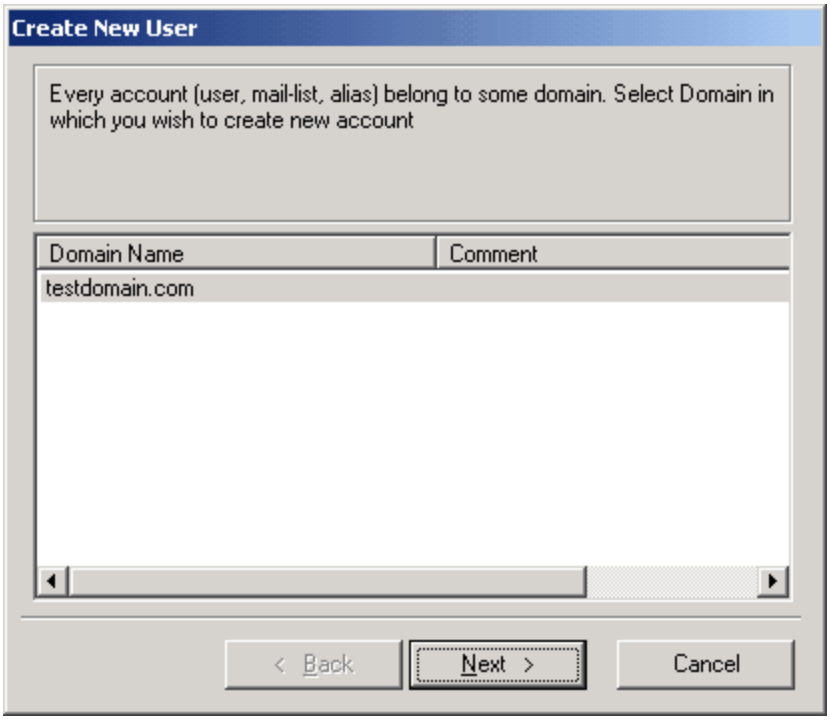
Следующее окно позволяет либо завершить мастер, либо продолжить создание доменов:



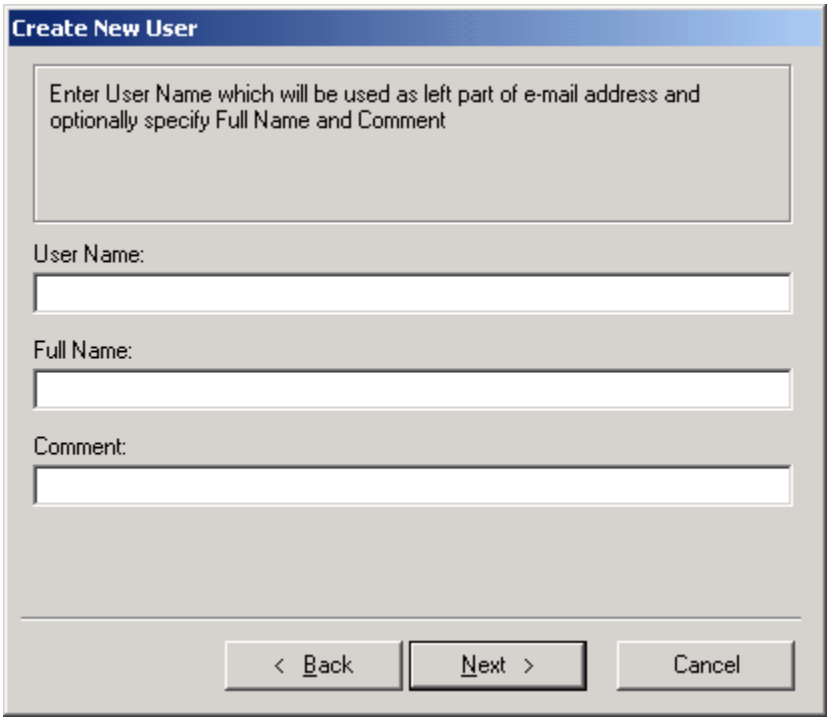
3.6 Создание пользователей

Почтовый сервер BatPost поддерживает несколько типов пользователей (user) или, как их еще можно назвать, учетных записей (account). Письма, приходящие на разные виды учетных записей обрабатываются по разному. Здесь мы рассмотрим "обычных" пользователей, письма которых складываются в их личный почтовый ящик.

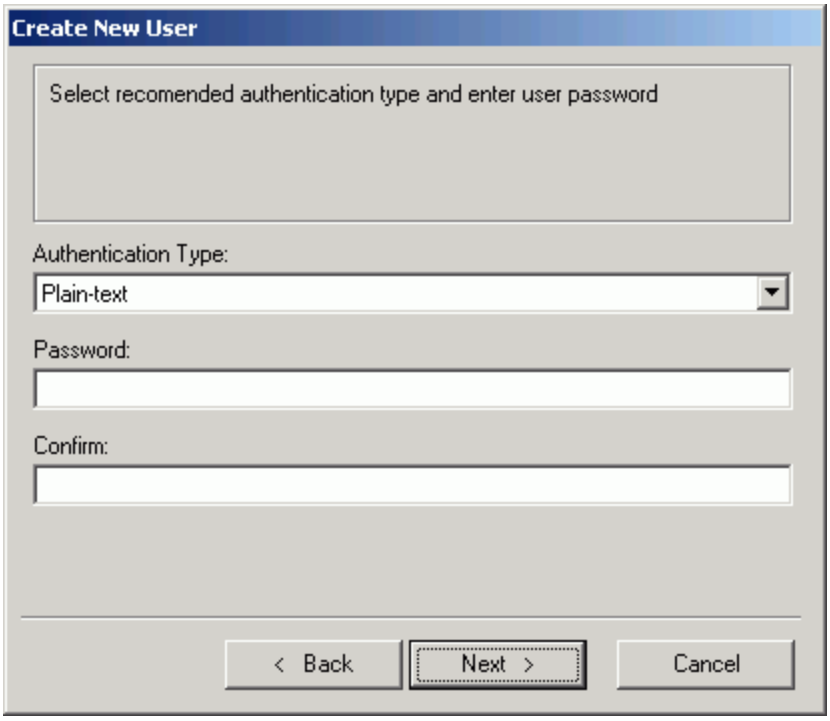
При создании пользователя сначала будет предложено выбрать домен, к которому он будет принадлежать:



Далее появится окно, позволяющее задать основные параметры пользователя:

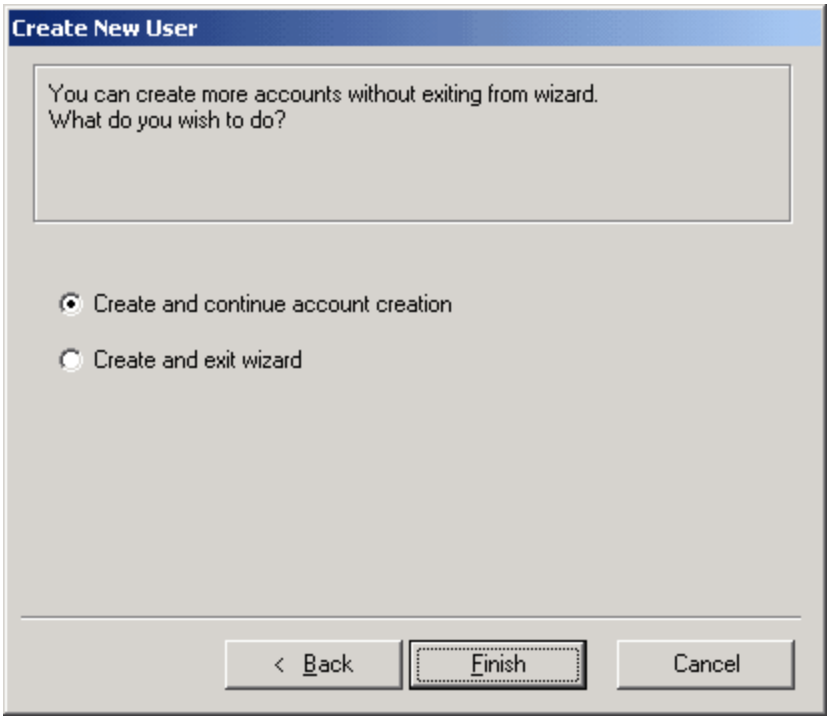


Нужно обязательно задать имя пользователя на сервере. Полное имя и комментарий являются необязательными параметрами.



Следующее окно позволяет задать параметры аутентификации. Если будут использоваться почтовые клиенты, поддерживающие "сильные" методы аутентификации (например, The Bat!), то рекомендуется выбрать метод CRAM-MD5. Если же планируется использовать Outlook, то рекомендуется выбрать метод аутентификации NTLM (MSN).

Следующее окно позволяет либо завершить мастер, либо продолжить создание пользователей:

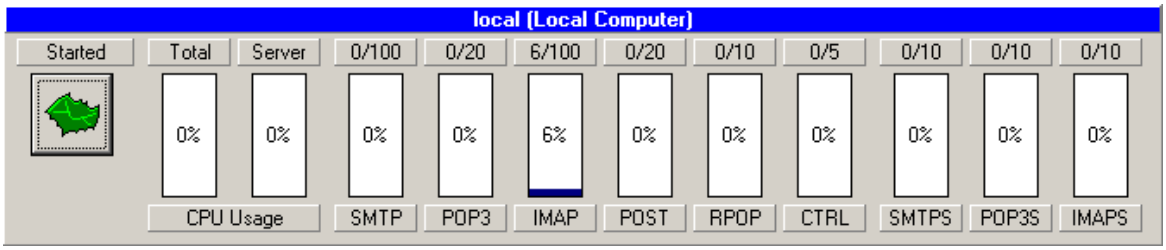


3.7 Запуск и проверка работоспособности

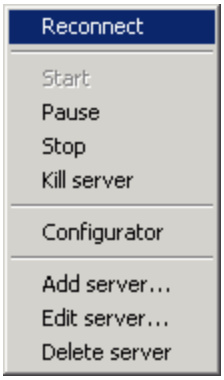
На данном этапе производится пробный запуск сервера, с целью убедиться в его работоспособности. После завершения мастеров начальной настройки сервер будет запущен автоматически.

Управление работой сервера производится при помощи модуля Монитора (BatPostM.exe). Он позволяет производить запуск и останов сервера, наблюдать за его активностью и загрузкой, а также просматривать журналы событий. Монитор позволяет наблюдать одновременно за несколькими серверами.

Верхняя часть окна Монитора отвечает за управление серверами и за отображение их загрузки по протоколам:

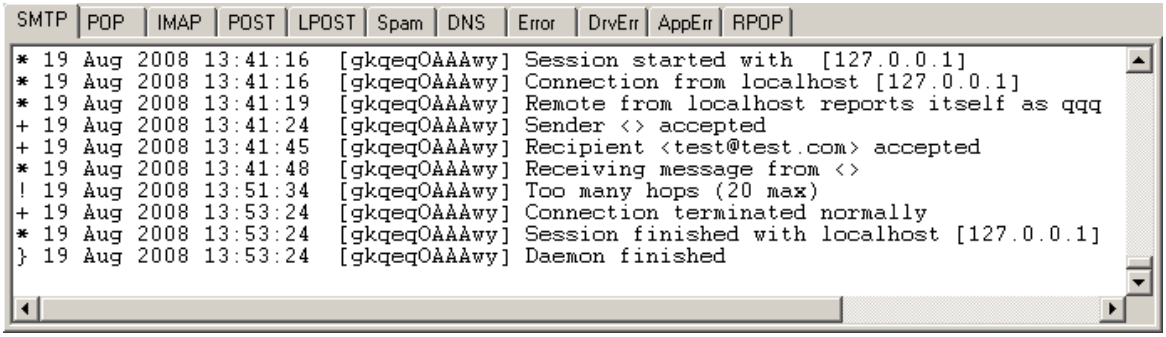


Кнопка с изображением летучей мыши показывает текущее состояние сервера. Это же состояние в тестовом виде отображается над кнопкой. При нажатии на кнопку появляется меню управления сервером:



Оно позволяет выполнить все операции по управлению сервером, а также работать со списком серверов за которыми ведется наблюдение.

В нижней части окна показываются журналы событий:



Просмотр журналов может очень помочь при выявлении неисправностей в работе сервера. События отсортированы по категориям. Наиболее важными являются Error.log, который показывает общие ошибки при работе сервера, и DrvErr.log, который показывает ошибки при работе с данными.

Если установка и начальная настройка сервера были проведены успешно, то сервер должен быть уже запущен. Если же этого не произошло, то скорее всего сервер был неправильно настроен. Подробности следует искать в Error.log. Так, например, сообщение вида:

```
! 14 Jun 2008 09:59:47  DNS server not specified. Server stopped
```

говорит о том, что в процессе настройки не был задан хотя бы один адрес DNS сервера. Более подробно различные ошибки рассмотрены в разделе "Решение проблем".

Если сервер запустился, то настало время проверить его работу с различными протоколами. В дальнейшем предполагается, что на сервере существует домен "testdomain.com" и что в этом домене существует пользователь "test".

Для начала необходимо проверить работу сервера локально. Для этого запустите на том же компьютере, на котором запущен сервер, почтовый клиент. В настройках почтового клиента в качестве почтового адреса следует указать "test@testdomain.com", а в качестве имени пользователя ввести "test". Также нужно указать правильный пароль данного пользователя. В полях для задания POP3 и

SMTP серверов следует ввести "127.0.0.1", это означает, что в качестве сервера будет использован текущий компьютер.

Теперь отправим от имени "test@testdomain.com" тестовое письмо на этот же адрес и примем почту. Если все прошло правильно, то в папке для входящих писем должно быть тестовое письмо, то есть сервер успешно настроен для обмена сообщениями между локальными пользователями.

В случае возникновения ошибок следует обратиться к журналам событий для выяснения того, что же собственно пошло не так и, в случае необходимости, исправить настройки сервера. При проблемах с отправкой письма надо смотреть журнал SMTP.LOG, а при проблемах с приемом - POP.LOG.

Разрешить доставку по внешним адресам можно несколькими методами:
- поставить галочку для проверки по mail-from, но при этом необходимо помнить о возможности подделки
- указать шаблоны адресов или имен хостов, которым разрешена удаленная доставка
- обязать всех клиентов делать аутентификацию при отправке почты

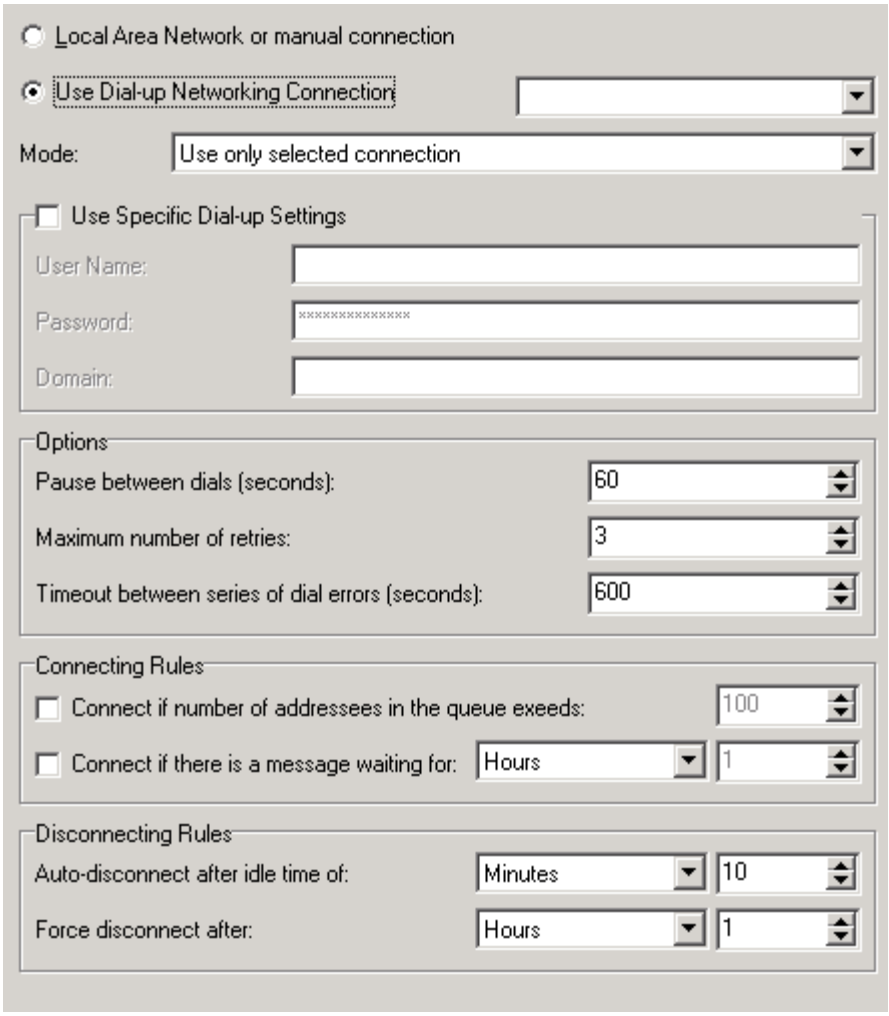
Наиболее предпочтительным является последний метод и именно он используется по умолчанию.

Для проверки удаленной доставки нужно послать тестовое письмо на какой-нибудь сторонний адрес и проверить, что оно было успешно доставлено. В случае ошибок, нужно смотреть SMTP.LOG для того, чтобы убедится, что письмо было успешно принято сервером, и Post.LOG, для того, чтобы убедиться, что письмо было успешно доставлено удаленному серверу.

3.8 Дополнительные настройки

В случае, когда постоянное подключение отсутствует, сервер может периодически подключаться через телефонную линию (dial-up) для приема входящей и отсылки исходящей почты.

Прежде всего необходимо настроить способ доступа к Интернету в разделе "Опции > Сеть" ("Options > Networking"):



В нашем случае нужно выбрать режим "Использовать подключение через Dial-up" ("Use Dial-up Networking Connection") и выбрать из списка одно из существующих подключений. Если в момент, когда серверу нужен доступ к Интернету, соединение уже существует, возможны разные режимы работы:

- Не использовать существующие подключения (Don't use existing connections)
- Использовать только выбранное подключение (Use only selected connection)
- Использовать любое существующее подключение (Use any existing connection)

По умолчанию используются настройки указанные при создании подключения, однако можно указать специфичные имя пользователя, пароль и домен.

При попытке дозвонится могут возникать различные ошибки. Поэтому сервер использует следующую стратегию: он производит серию звонков и, если дозвонится не удалось, выдерживает паузу и повторяет серию. Таким образом имеются следующие настройки: пауза между попытками внутри серии, количество попыток в серии и пауза между сериями.

Сервер будет отсылать исходящую почту как только появится подключение к Интернету. Для того, чтобы в очереди не скапливалось слишком много писем,

сервер может соединятся с Интернетом когда количество писем в очереди или время ожидания отправки превысит заданные значения.

Поскольку, при использовании телефонной линии, нахождение в Интернете приводит к существенным финансовым затратам, необходимо как-то ограничить его по времени. Для этого предусмотрены настройки автоматического отключения после заданного времени бездействия или "силового" отключения после истечения заданного промежутка времени.

Так как сервер не находится в Интернете постоянно, он не может отвечать за бесперебойный прием входящей почты для своих доменов. Чтобы работа почтовой подсистемы не нарушалась, необходимо чтобы кто взял на себя ответственность за прием почты приходящей на наши домены. Чаще всего это почтовый сервер провайдера. Он принимает почту, но никак ее не обрабатывает, а держит в каком-то особом хранилище. В качестве такого хранилища может выступать специально созданный для этих целей почтовый ящик, либо очередь отправки сервера (spool).

Если вся почта складывается в особый почтовый ящик, то для того, чтобы ее забрать, должна использоваться технология Remote POP (RPOP). Это означает, что наш сервер должен по протоколу POP3 соединится с сервером провайдера и забрать из этого ящика все письма. При этом, для того чтобы определить отправителя и получателя, анализируются заголовки писем.

Если же сервер временно хранит письма в своей очереди отправки, то должна использоваться технология ETRN/ATRN. Это означает, что наш сервер соединяется с сервером провайдера по протоколу SMTP и выдает особую команду о том, что собирается забрать почту для заданного домена (или доменов). После этого, в зависимости от того, какая команда использовалась сервер провайдера либо устанавливает с нашим сервером новое входящее SMTP соединение, либо текущее соединение из исходящего преобразуется во входящее. После этого происходит доставка почты так, как если бы наш сервер имел постоянное подключение к Интернету.

В любом случае нужно посоветоваться с провайдером и узнать какой метод он может предоставить. Поскольку анализ заголовков писем далеко не всегда может дать достоверную информации об отправителе и получателе сообщения, предпочтительнее использовать второй метод.

Глава 4

4 Вопросы и ответы

В этом разделе даны ответы на наиболее часто задаваемые вопросы. Прежде чем задавать вопросы в службу поддержки, внимательно ознакомьтесь с данным разделом и убедитесь что в нем нет ответа на Ваш вопрос.

4.1 Настройка сервера

В настройках пользователя есть поле "Тип аутентификации" ("Authentication type"). Какое значение в нем указывать?
В данном поле указывается самый слабый тип аутентификации, который будет допустим для данного пользователя. Т.е. пользователь сможет зайти на сервер используя данный метод или более безопасный.
Поддерживаются следующие типы аутентификации:

- Открытым текстом (Plain-text)
- NTLM (MSN)
- CRAM-MD5

Типы аутентификации перечислены в порядке возрастания их безопасности. Использование защищенного SSL (TLS) соединения повышает безопасность метода, и даже если указан CRAM-MD5, допустимо авторизоваться открытым текстом через защищенное соединение.
Аутентификация открытым текстом передает пароль по сети в открытом виде и поэтому ее не рекомендуется использовать кроме как по защищенному SSL (TLS) соединению. Другие типы авторизации используют криптографические методы для того, чтобы подтвердить подлинность пользователя без раскрытия его пароля. По возможности рекомендуется использовать CRAM-MD5, как более безопасный метод, однако не все почтовые клиенты его поддерживают. Если планируется использовать MS Outlook или MS Outlook Express, то нужно использовать NTLM (MSN), как единственный безопасный метод поддерживаемый данными клиентами.

На сервере есть пользователи с одинаковыми именами, но находящиеся в разных доменах. При авторизации клиенты получают странные результаты. Как правильно авторизоваться?
Если в настройках клиента указано только имя пользователя, то сервер пытается найти пользователя с таким именем во всех доменах и использует первого, которого удалось найти. Поэтому может оказаться так, что первым попадется пользователь из другого домена. В этом случае авторизация будет неудачной, если пароли не совпадают или пользователь увидит письма совершенно другого пользователя, если у них еще и одинаковые пароли.
Для того, чтобы решить эту проблему, рекомендуется указывать в качестве имени пользователя его полный электронный адрес (e-mail). Т.е. вместо "user" указывать "user@example.com". В этом случае неоднозначности не возникнет.

Как правильно настроить "Удаленный POP" ("Remote POP")?
"Удаленный POP" ("Remote POP") предназначен для того, чтобы забрать почту из внешнего ящика по POP3 протоколу и переложить ее во внутренние ящики сервера или переслать на заданный адрес. Для правильной работы нужно указать параметры внешнего сервера и создать расписание для периодического запуска этого задания. Расписание должно быть добавлено в список используемых на закладке "Расписания" ("Schedules").

Есть два режима работы:

- "Удаленная учетная запись" ("Remote Account") - в этом режиме письма из внешнего ящика переносятся в заданный локальный ящик или пересылаются по заданному адресу.
- "Домен POP" ("Domain POP") - в этом режиме анализируются заголовки писем, чтобы определить адресата.

Поскольку любые изменения в настройках требуют перезапуска сервера, чтобы они вступили в силу, удобно пользоваться кнопкой "Запустить!" ("Run Now!") для немедленного запуска задания с новыми настройками. Результаты работы нужно смотреть в журнале RPop.LOG - это единственный способ узнать причину ошибок или убедиться, что задание отработало верно. После того, как настройки будет проверены, нужно убедиться что расписание также настроено и перезапустить сервер.

Для чего нужно поле "Переслать на" ("Forward To") в настройках "Удаленный POP" ("Remote POP") и нужно ли его заполнять?

В зависимости от режима поле "Переслать на" ("Forward To") действует по разному. В режиме "Удаленная учетная запись" ("Remote Account") в этом поле указывается адрес на который будут пересылаться все письма. В режиме "Домен POP" ("Domain POP") в этом поле указывается адрес на который будут пересылаться письма для которых не удалось определить адресата.

Это поле рекомендуется заполнять в обоих режимах, так как иначе на сервере может оставаться часть писем. Эти письма будут в дальнейшем скачиваться в каждой сессии до тех пор, пока не будут удалены вручную. Если таких писем на сервере будет много, то это может привести к замедлению работы "Удаленный POP" ("Remote POP") и значительным потерям трафика.

На нашем локальном сервере должна быть прописана только часть пользователей домена, а остальных пользователей обрабатывает другой сервер. Как это правильно настроить?

Для этого нужно создать домен и прописать в нем только тех пользователей, которые должны храниться локально. А в настройках домена на закладке "Расширенные настройки" ("Advanced") нужно выбрать режим "Рассматривать неизвестных пользователей как внешних" ("Treat unknown users as remote").

Как запретить некоторым пользователям отправку писем на внешние адреса?

В настройках пользователя, на закладке "Базовые настройки" ("Basic") есть галочка "Права при отправке почты через SMTP" ("SMTP mail sending rights"). Если ее включить, то можно персонально для данного пользователя установить права при отправке на локальные и внешние адреса. Зеленые квадратики означают разрешение, красные - запрет. Чем больше квадратиков выставлено, тем сильнее запрет или разрешение. Это может быть важно, так как могут одновременно действовать другие правила (подробности проверки правил можно прочитать в "Руководстве Администратора"). Для гарантированного запрета нужно установить четыре красных квадратика.

Мы зарегистрировали новое имя почтового домена, но на старый все еще приходят письма. Как сделать так, чтобы на сервере было два разных домена, но чтобы пользователи в них были одни и те же?

Как создать несколько доменов с одинаковым содержимым (главный домен и его псевдонимы)?
При создании домена на закладке "Расширенные настройки" ("Advanced") в поле "Тип содержимого" ("Contents Type") можно выбрать значение "Содержимое по ссылке" ("Referral contents") и в поле "Ссылка" ("Referral") указать домен, из которого будут браться пользователи для вновь созданного домена. Данные при этом не дублируются, просто при обращении к пользователям нового домена, будут использоваться пользователи домена указанного в ссылке.

В настройках домена на закладке "Протоколы" ("Protocols") есть поле "IP адрес" ("IP address"). Что нужно указывать в этом поле?
В большинстве случаев это поле нужно оставлять пустым. Только если сервер имеет несколько сетевых интерфейсов (сетевых карт), которые находятся в разных подсетях и нужно, чтобы соединения принимались только из конкретной подсети, то нужно указать IP адрес данной подсети. Если это поле оставлено пустым, то соединения будут приниматься из всех существующих подсетей.

Изменение каких настроек требует перезапуска сервера?
Перезапуска сервера требуют изменения в настройках доменов, расписаний, антивируса и анти-спама, а также к настроек в разделе "Опции" ("Options").

4.2 Регистрация

Для чего нужно регистрировать программу?
После регистрации будут сняты ограничения испытательной версии на количество пользователей. Можно будет также получать бесплатную техническую поддержку.

Куда вводить регистрационный код?
Для того чтобы ввести регистрационный код нужно в Конфигураторе выбрать пункт меню **Помощь > Ввести Регистрационный Код** (**Help > Enter Registration Code**). Если выбрать в меню **Помощь > О программе** (**Help > About**), то можно увидеть список всех регистраций. При помощи кнопок **Добавить** (**Add**) и **Удалить** (**Delete**) можно добавлять новые и удалять существующие регистрации. Количество лицензий (максимальное количество пользователей) по всем регистрациям суммируется.

Замечание. Для того, чтобы зарегистрировать программу под Windows Vista, нужно запустить Конфигуратор с правами администратора (Run as Administrator).

Какие данные высылаются при регистрации?
В регистрационном письме должны быть:
- Ключ (Key)
- Контрольная сумма (Key Checksum)
- Пароль (Key Password)

Пароль используется для защиты регистрационного ключа от кражи и указывается в письме только если был назначен дилером. Если же вы сами указывали пароль при регистрации, то в письме его не будет.

Кем назначается пароль для ключа регистрации?
Пароль назначается либо самим пользователем при заполнении регистрационной формы, либо дилером. В последнем случае дилер должен сообщить пароль

пользователю.

Какие ограничения у незарегистрированной версии?
В испытательной версии можно создать не более 20 учетных записей (пользователей) во всех доменах. Испытательный период составляет 30 дней. В течение испытательного периода доступны все функции программы.

Что произойдет после истечения испытательного периода, если программа не была зарегистрирована?
Истечение испытательного срока вовсе не означает, что программа перестанет работать. Просто появятся дополнительные ограничения - максимальное количество соединений по каждому из протоколов станет равным 2 вне зависимости от того, что указано в настройках.

4.3 Решение проблем

При попытке отправить почту сервер говорит "We don't relay". Что это значит?
Настройки по умолчанию требуют, чтобы при отправке на внешние адреса обязательно использовалась SMTP авторизация. Это необходимо, чтобы через сервер нельзя было рассылать нежелательные письма (спам).
Для того, чтобы избавиться от этой ошибки, нужно в настройках почтового клиента включить авторизацию при отправке почты.

При запуске сервера в журнале Error.log появляются сообщения вида "! 15 Oct 2007 15:57:06 POP addresses configuration error" или "! 15 Oct 2007 15:57:06 SMTP addresses configuration error". Что настроено не так?
Эти сообщения означают, что есть конфликт между доменами в настройках протоколов. Скорее всего, для указанного протокола в одном домене был указан IP адрес конкретной подсети, а в другом домене это поле было оставлено пустым. Прием соединений на заданном порту возможен либо из заданной подсети, либо из всех подсетей, но не то и другое сразу - поэтому возникает конфликт.
Скорее всего поле "IP адрес" ("IP address") было заполнено по ошибке. Чаще всего требуется принимать соединения из всех подсетей и это поле должно оставаться пустым.

При запуске сервера в журнале Error.log появляются сообщения вида "Could not bind to address "0.0.0.0", port 25".
Это означает, что 25 порт (SMTP) уже занят какой-то другой программой. Скорее всего запущен какой-то другой почтовый сервер или SMTP сервис системы.
Для того, чтобы выяснить кем занят порт, нужно из командной строки выполнить команду:

```
telnet localhost 25
```

В приглашении сервера обычно упоминается его название.

При запуске сервера в журнале Error.log появляются сообщения вида "Could not bind to address "192.168.101.1", port 25".
Скорее всего по ошибке было заполнено поле "IP адрес" ("IP address") в настройках протоколов домена или этот адрес не принадлежит данному компьютеру. Для того, чтобы выяснить IP адреса данного компьютера нужно из командной строки

выполнить команду:

```
ipconfig
```

Для получения полной информации о сетевых настройках нужно выполнить команду:

```
ipconfig /all
```

При попытке отправить почту сервер выдает ошибку "5.7.1 - unable to relay".
Это сообщение принадлежит явно не BatPost. Либо при попытке отправить почту был по ошибке использован какой-то другой сервер, либо кроме BatPost запущен еще какой-то сервер, который перехватил SMTP порт.
Для того, чтобы быть точно уверенным, нужно запустить на сервере из командной строки команду:

```
telnet localhost 25
```

и убедиться, что в приглашении сервера присутствует слово "BatPost".

Очень долго происходит доставка писем (даже локальная).
Большая загрузка процессора.
Антивирусные мониторы могут пытаться проверять каждое приходящее письмо. При большом количестве писем это может существенно затормозить работу сервера. Нужно попробовать отключить антивирус и, если проблема исчезнет, значит он просто не справляется с таким потоком писем.
Для того, чтобы определить какое приложение грузит процессор, можно использовать Task Manager. В некоторых случаях он показывает, что основную загрузку дает процесс System. Для того, чтобы получить более подробную информацию, лучше использовать Process Explorer. Скачать его можно здесь: http://technet.microsoft.com/en-us/sysinternals/bb896653.aspx

При попытке забрать почту сервер выдает сообщение "Authentication failed. Mailbox locked".
Протокол POP3 требует, чтобы почтовый ящик был заблокирован на время работы с ним. Если попытаться зайти в почтовый ящик, в то время как к нему уже кто-то обращается, будет получено это сообщение.
Нужно убедиться, что с данным ящиком по протоколу POP3 в каждый момент времени работают только из одного клиента. Так, например, вполне допустимо когда человек проверяет свой ящик на работе и дома, но делает это в разное время. Если же требуется одновременный доступ, то должен использоваться протокол IMAP.

Серверу не удается отправить почту на внешние адреса. В чем может быть проблема?
Прежде всего нужно изучить журнал Post.LOG, там могут быть сообщения об ошибках, которые прояснят ситуацию. Если для доставки почты на внешние адреса используется DNS, то нужно убедиться, что его адрес указан правильно и он работает. Если же используется промежуточный сервер (relay), то нужно убедится, что сообщение было успешно ему передано.
При использовании DNS возможна ситуация, когда удается успешно соединится с внешним сервером, но он отказывается принимать почту. Это может быть вызвано тем, что на внешнем сервере используются расширенные проверки для борьбы с нежелательной почтой (спамом). Так, может проверяться:

- Прописан ли наш сервер в DNS (это подразумевает, что у нас статический IP

адрес)
- Прописана ли для нашего сервера PTR запись в обратной зоне DNS. Такая проверка характерна для сервера mail.ru - в случае, если PTR записи нет, он возвращает ошибку "550 Unroutable address".

Иногда единственным способом решения проблемы может быть использование промежуточного сервера (relay) для отправки почты во внешний мир.

Почта из внешнего мира до нас не доходит. В чем может быть проблема?
Есть два пути по которым к нам может попадать почта из внешнего мира:
- Если у нас есть свой почтовый домен и в DNS прописано, что наш сервер отвечает за почту для этого домена, то другие сервера сами будут с нами соединяться и передавать нам почту. В этом случае для нашего сервера должны быть прописаны A запись, описывающая наш IP адрес и MX запись ссылающаяся на A запись.
- Мы можем сами соединяться с другими серверами и забирать почту нам предназначенную. В этом случае мы можем даже не иметь своего домена и забирать письма с конкретных адресов через "Удаленный POP" ("Remote POP"). Если же у нас есть почтовый домен, то он может указывать на сервер который временно принимает почту для нашего домена (например, сервер провайдера). Этот промежуточный сервер может хранить письма в своей очереди сообщений и отдавать их через ETRN/ATRN или складывать письма в специальный ящик, чтобы мы могли их забрать через "Удаленный POP" ("Remote POP"). Первый вариант предпочтительнее, поскольку в этом случае не теряется информация об истинном адресате письма, но не все сервера могут предоставлять такую возможность. В любом случае решение остается за владельцем промежуточного сервера.

Не проходит авторизация. В журналах пишется "Incorrect password", хотя я уверен, что пароль указан верно. В чем проблема?
Скорее всего в настройках почтового клиента было указано только имя пользователя без указания домена, а на сервере пользователь с таким именем существует в нескольких доменах. В этом случае сервер будет искать пользователя во всех доменах и проверит первого с совпадающим именем. Для того, чтобы исключить неоднозначность, нужно в почтовом клиенте указать полное имя пользователя (совпадает с его электронным адресом). Т.е. вместо "user" указать "user@example.com".

Не проходит авторизация. В журналах пишется "User not found", хотя такой пользователь точно существует. В чем проблема?
Есть две основные причины:
- Был создан новый домен, а в нем пользователи, но сервер не перезапускался. Создание, удаление и изменение настроек доменов требуют обязательного перезапуска сервера.
- В домене, которому принадлежит этот пользователь, отключен нужный протокол. Либо этот протокол настроен неправильно. Нужно убедиться, что в настройках домена на закладке "Протоколы" ("Protocols") стоит галочка соответствующая нужному протоколу и что в журнале Error.log нет сообщений об ошибках связанных с данным протоколом. После изменения настроек сервер обязательно нужно перезапустить.

Изменения настроек не подхватываются. В Конфигураторе изменения видны, но сервер работает по старому. Что нужно сделать?
Изменение некоторых настроек требует перезапуска сервера, для того чтобы эти изменения вступили в силу. Это относится к настройкам доменов, расписаний, антивируса и анти-спама, а также к настройкам в разделе "Опции" ("Options").

4.4 Журналы

Какие журналы ведутся на сервере и каково их назначение?
Сервер, сам по себе, не имеет визуального интерфейса и начинает свою работу еще до того как пользователь зайдет (login) под какой-то учетной записью. Поэтому единственным способом слежения за работой сервера и диагностики проблем является ведение журналов работы. Для удобства разные виды событий записываются в разные журналы. По умолчанию запись ведется в текстовые файлы, однако возможно ведение журналов в базе данных. Нужно, однако, учитывать что работа с базой данных происходит намного медленнее, чем с текстовыми файлами, и может затормозить работу сервера.
На сервере ведутся следующие журналы:

- SMTP.LOG - здесь записываются события, происходящие в SMTP сессиях. SMTP используется почтовыми клиентами и другими серверами для отправки почты
- POP.LOG - здесь записываются события, происходящие в POP сессиях. POP используется для приема почты.
- IMAP.LOG - здесь записываются события, происходящие в IMAP сессиях. IMAP используется для приема почты.
- Post.LOG - здесь записываются события, происходящие при доставке писем на другие сервера.
- LPost.LOG - здесь записываются события, происходящие при локальной доставке писем.
- Session.LOG - здесь записывается содержание сессий, для которых установлена опция записи "Запись сессий" ("Log Sessions") в разделе "Опции > Журналы" ("Options > Logs") на закладке "Параметры" ("Parameters").
- RPOP.LOG - здесь записываются события, происходящие в сессиях "Удаленный POP" ("Remote POP").
- Control.LOG - здесь записываются события, происходящие в сессиях управляющего протокола. Управляющий протокол используется Конфигуратором для настройки сервера.
- PCP.LOG - здесь записываются события, происходящие в PCP сессиях. PCP (Password Change Protocol) используется для изменения пароля пользователя. Это протокол поддерживается почтовым клиентом The Bat.
- Spam.LOG - здесь записываются события, связанные с работой анти-спам механизмов.
- DNS.LOG - здесь записываются события, связанные с обращением к DNS (Domain Name System) серверам.
- Error.LOG - здесь записываются события, связанные с ошибками в работе сервера. Эти ошибки требуют пристального внимания, так как могут означать либо проблемы с конфигурацией, либо ошибки в самом сервере, о которых нужно сообщить разработчикам.
- DrvErr.LOG - здесь записываются события, связанные с ошибками в работе драйвера доступа к данным. Они также требуют пристального внимания, так как

скорее всего связаны с ошибками в драйвере доступа к данным или с повреждением баз.
- AppErr.LOG - в настоящее время не используется.
- Virus.LOG - здесь записываются события, связанные с работой антивирусных механизмов.
- Dialup.LOG - здесь записываются события, связанные с работой dial-up ("звонилки"). Если для доступа к Интернету используется телефонная линия, то здесь будут события связанные с установлением и разрывом телефонного соединения.

Для чего нужны журналы исключительных ситуаций?

Исключительная ситуация чаще всего сигнализирует об ошибке в какой-то из компонент сервера. При возникновении исключительной ситуации создается журнал вида

```
<имя_компонента>_Exceptions.log
```

Так, например, если исключение возникло в модуле BatPost.exe, то будет создан журнал

```
BatPost_Exceptions.log
```

Искать эти журналы для большинства компонент нужно в папке

```
%allusersprofile%\Application Data\BatPost\
```

Для драйвера доступа к данным этот журнал будет в папке

```
%allusersprofile%\Application Data\BatPost\Drivers\
```

Значение переменной `%allusersprofile%` зависит от системы. Так, например, в Windows XP она обычно имеет значение

```
C:\Documents and Settings\All Users\
```

Для того, чтобы узнать ее значение в вашей системе, нужно выдать с командной строки:

```
echo %allusersprofile%
```

Эти журналы позволяют разработчикам более точно понять причину и место возникновения ошибки. Поэтому, сообщая разработчикам об ошибке, очень важно приложить к обращению соответствующий журнал.

Журналы очень быстро увеличиваются в размерах. Что с этим можно сделать?

Поскольку в журналах регистрируется масса событий из жизни сервера, они могут расти очень быстро. Насколько быстро это происходит зависит от загруженности сервера, но рано или поздно можно столкнуться с ситуацией, когда место на диске кончилось, а сервер ушел в непонятное состояние. Даже если места на диске достаточно, работать с гигантскими файлами журналов может быть не очень удобно.

Для того, чтобы избежать этих ситуаций, сервер поддерживает автоматическое разбиение и чистку журналов. Настраивается это в разделе "Опции > Журналы" ("Options > Logs") на закладке "Разбиение/Очистка" ("Splitting/Purging"). При разбиении создается отдельный файл в имени которого присутствует дата к которой он относится. Так, например, при разбиении POP.LOG может быть создан файл

```
POP20080602.LOG
```

который относится к 2008 году 06 месяцу (июнь) и 02 дню. Такой порядок указания даты позволяет при сортировке по имени файлов получить список файлов, упорядоченный по дате к которой они относятся.

Журналы можно разбивать на части: ежедневно, еженедельно или при достижении

заданного размера. Если журнал растет не слишком быстро, то вполне подойдет еженедельное разбиение, для быстрорастущих журналов (например, SMTP.LOG) больше подойдет ежедневное разбиение.
Разбиение журналов решает проблему роста их размеров, но не спасает от переполнения диска. Для решения этой проблемы используется автоматическая очистка. Журналы, которые старше чем заданное количество дней, могут автоматически удаляться с диска или переноситься в указанную папку. При использовании удаления рекомендуется хранить журналы хотя бы за последний месяц, так как иногда приходится решать проблемы которые возникли намного раньше, чем были замечены. Если же журналы переносятся в специальную папку, то должно использоваться резервное копирование старых журналов на диски или ленту, чтобы не произошло переполнения уже по новому месту.

Какой программой лучше просматривать журналы?
Монитор можно использовать для просмотра последних записей в основных журналах. Этого достаточно для оценки текущей работы сервера, но для более серьезной работы нужно работать напрямую с файлами журналов. Для этого подойдет любая программа умеющая быстро просматривать большие файлы и имеющая функцию поиска строк. Такую возможность предоставляют многие файловые менеджеры (Far, Total Commander), а также некоторые текстовые редакторы. Файловый менеджер в этом плане более предпочтителен, так как позволяет видеть все файлы журналов, искать в нескольких файлах и исполнять команды из командной строки.

Как найти в журналах нужное?
Поскольку журналы могут иметь очень большие размеры, найти нужную часть информации не всегда просто. Однако, если придерживаться определенной стратегии, то то поиск не составит большой сложности.
Прежде всего нужно определиться с ключевой фразой, которая поможет нам найти нужное. Так, например, если мы ищем нечто связанное с пользователем "user@example.com", то это и будет нашей ключевой фразой.
Далее нужно определиться к какой дате относится разыскиваемое событие. Если используется разбиение журналов, то можно будет сосредоточиться на журналах, относящихся к этой дате. Иногда бывает, что точная дате неизвестна - в этом случаем мы можем воспользоваться поиском по нескольким файлам. Например, мы можем искать в папке "Logs" указав файловую маску "*.*", чтобы искать по всем файлам или указать маску "SMTP*.*", чтобы искать только в файлах относящихся к SMTP протоколу.
Когда ключевая фраза найдена, нас обычно интересуют события относящиеся к той же сессии. Вот как выглядит типовая POP сессия:

```
{ 03 Jun 2008 17:38:01  [gURXxDAAAgA] Daemon started
* 03 Jun 2008 17:38:01  [gURXxDAAAgA] Session started with  [127.0.0.1]
* 03 Jun 2008 17:38:01  [gURXxDAAAgA] Connection from localhost [127.0.0.1]
+ 03 Jun 2008 17:38:01  [gURXxDAAAgA] User test <test@test.com> logged on
(Plain-text)
+ 03 Jun 2008 17:38:01  [gURXxDAAAgA] User test <test@test.com> logged off
+ 03 Jun 2008 17:38:01  [gURXxDAAAgA] Connection terminated normally
* 03 Jun 2008 17:38:01  [gURXxDAAAgA] Session finished with localhost
[127.0.0.1]
} 03 Jun 2008 17:38:01  [gURXxDAAAgA] Daemon finished
```

Можно заметить, что все записи имеют одинаковый уникальный идентификатор (ID) сессии. В данном случае это "gURXxDAAAgA". Этот идентификатор можно

использовать как путеводитель для поиска записей данной сессии. В простых случаях это можно сделать визуально, в случае же когда записи разных сессий сильно перемешаны можно выполнить из командной строки команду:

```
grep gURXxDAAAgA POP.LOG
```

для вывода строк сессии "gURXxDAAAgA" прямо на экран, либо команду:

```
grep gURXxDAAAgA POP.LOG >Ses.txt
```

для вывода их в файл "Ses.txt".

Замечание. По умолчанию утилита grep может отсутствовать в Вашей системе. Изначально она была создана для UNIX и в Windows отсутствует. Однако ее можно скачать в составе пакета UnxUtils с сайта http://unxutils.sourceforge.net Пакет распространяется в виде zip-архива, который нужно разархивировать в какую-то папку на вашем диске (например, C:\UnxUtils). После этого эту папку желательно добавить в переменную окружения PATH, чтобы доступ к утилитам был возможен из любого места.

Как проследить прохождение письма через сервер?

При попадании на сервер каждому письму присваивается уникальный идентификатор (ID). Не следует путать его с уникальным идентификатором сессии, специально для этого они имеют различный вид. Этот идентификатор можно увидеть в журналах везде где происходит обращение к сообщению. Вот как это может выглядеть в журнале SMTP.LOG:

```
> 02 Jun 2008 17:05:45  [g0Q9fIAAAAA] Message 4843FD8700000001 (4438 bytes)
received
```

здесь 4843FD8700000001 - это уникальный идентификатор присвоенный принятому сообщению.

А вот как это может выглядеть в журнале POP.LOG при прочтении и удалении того же сообщения:

```
+ 02 Jun 2008 17:05:52  [g0Q9fIAAAAB] Message 4843FD8700000001 (4438 bytes)
was read
+ 02 Jun 2008 17:05:52  [g0Q9fIAAAAB] Message 4843FD8700000001 (4438 bytes)
marked as deleted
+ 02 Jun 2008 17:05:52  [g0Q9fIAAAAB] Message 4843FD8700000001 (4438 bytes)
permanently deleted
```

Также этот идентификатор прописывается в заголовке самого сообщения в поле Received:

```
Received: from userdomain.com (userdomain.com [192.168.101.1])
     by testdomain.com with BatPost v2.21r4 ESMTP Daemon
     id 4843FD8700000001
     for <test@testdomain.com>; Mon, 2 Jun 2008 17:05:45 +0300
```

Поле Received обычно прописывается каждым сервером, через который проходило письмо, поэтому нужно найти среди них то, которое было прописано сервером BatPost.

Поскольку уникальный идентификатор прописывается в журналах при каждом действии с данным письмом, можно проследить жизненный цикл письма на сервере. Так в журнале SMTP.LOG можно найти запись о том, как письмо было принято на сервер, а в журнале POP.LOG (или IMAP.LOG) - запись о том как оно было прочитано пользователем и, возможно, удалено.

Если из поля Received в заголовке письма известен идентификатор сообщения, то можно произвести поиск по всем журналам, чтобы найти все записи относящиеся к данному сообщению и получить полное представление о том, что происходило с данным сообщением на сервере.

4.5 Обслуживание сервера

Где сервер хранит свои файлы?
До версии 2.21 сервер по умолчанию хранил все свои файлы в папке %programfiles %\BatPost. Начиная с версии 2.21 программные файлы хранятся в папке % programfiles%\BatPost, а файлы данных в папке %allusersprofile%\Application Data\ BatPost. Такое разделение сделано для совместимости с Window Vista.
Для совместимости с предыдущими версиями сервер проверяет местонахождение файла Server.ini - если он находится в папке %programfiles%\BatPost, то будет предполагаться, что файлы данных находятся там же. Если же его там нет, то будет предполагаться что файлы данных находятся в папке %allusersprofile% \Application Data\BatPost.
В корне структуры данных сервера находятся следующие файлы:

- Server.ini - в котором хранятся общие настройки сервера.
- GlobalAdmins - в котором хранится список глобальных администраторов. Глобальные администраторы имеют право удаленного мониторинга и конфигурирования сервера.

Также там хранятся следующие подпапки:

- Archive - здесь может храниться архив сообщений.
- DB - здесь хранится конфигурация сервера - группы, домены, пользователи, папки, сообщения, а также правила, "черные" и "белые" списки.
- Drivers - здесь хранятся настройки драйверов доступа к данным.
- Logs - здесь хранятся журналы работы сервера.
- PEMs - здесь могут храниться сертификаты сервера.
- Spool - здесь хранится очередь отправки сервера. В ней временно хранятся сообщения (MSG-файлы), до того как попадут в нужную базу или будут отосланы внешнему серверу. Здесь также хранится состояние их отправки (QUE-файлы).
- Stream - здесь хранятся временные файлы (стримы) слишком большие для того чтобы держать их в памяти.

Место хранения архива сообщений можно настроить в Конфигураторе в разделе "Опции > Архив&Аудит" ("Options > Archive&Audit").
Места, где хранятся журналы, очередь отправки и стримы могут настраиваться в Конфигураторе в разделе "Опции > Общие" ("Options > Common").

Замечание. Для того, чтобы изменить место хранения конфигурации недостаточно изменить настройку в разделе "Опции > Общие" ("Options > Common"). Нужно также вручную изменить в файле \Drivers\DefDrv.ini параметр RootDir в секции Common.

Место хранения сертификатов настраивается в разделе "Опции > Безопасность" ("Options > Security").

Как перенести сервер на другой компьютер?
Благодаря тому, что сервер хранит все свои настройки в виде файлов на диске и не используется для этих целей реестр Windows, перенос сервера сводится к копированию файлов с настройками.
Сначала нужно на новом месте установить сервер. В конце установки можно отказаться от настройки сервера, так как конфигурация сервера все равно будет скопирована. Затем нужно перенести на новое место файлы Server.ini и GlobalAdmins, а также содержимое папок DB, Drivers, Spool и PEMs (если есть сертификаты). Содержимое остальных папок можно тоже, по желанию, перенести

на новое место.
После этого нужно вручную запустить сервер и убедиться, что перенос конфигурации на новое место прошел успешно.

Как правильно обновить сервер на более новую версию?
Для правильного обновления нужно обязательно закрыть Конфигуратор и Монитор (если запущены). После этого запустить пакет установки новой версии. В конце установки будет сообщено, что сервер уже сконфигурирован и будет предложено эту конфигурацию изменить - на это можно ответить отказом.

4.6 Прочие вопросы

Как установить на сервер web-интерфейс?
Сам по себе сервер не имеет web-интерфейса, но есть пакеты сторонних разработчиков, которые позволяют добавить его. Такие пакеты обычно обращаются к серверу по протоколу IMAP. Многие из этих решений требуют для своей работы, чтобы был установлен web-сервер Apache (http://www.apache.org/) с установленным PHP (http://www.php.net/).
В качестве примера подобных решений можно назвать следующие пакеты:
- SquirrelMail (http://www.squirrelmail.org/)
- Horde IMP (http://www.horde.org/imp/)
- RoundCube (http://roundcube.net/)

Какие антивирусы поддерживаются сервером?
Сервер может использовать антивирусные модули расширения от программы The Bat (http://www.ritlabs.com/ru/products/thebat/plugin.php). Некоторые модули расширения идут в составе самих антивирусов. Не все модули корректно работают на сервере. Это связано с тем, что сервер может одновременно производит проверку множества писем на вирусы и не все антивирусные модули на это рассчитаны. Из проверенных можно назвать модуль расширения для NOD32.

Какие анти-спам решения поддерживаются сервером?
Сервер имеет встроенную поддержку SpamAssassin (http://spamassassin.apache.org/) - одного из наиболее мощных анти-спам решений. SpamAssassin может быть установлен как на том же компьютере, где запущен сервер, так и на отдельном компьютере (например, под управлением Unix). Обращение к SpamAssassin производится через модуль spamd, поэтому он тоже обязательно должен быть установлен. Установка и, особенно, настройка под Windows может оказаться непростой задачей. Достаточно подробное описание этого процесса (на английском) есть здесь: http://wiki.apache.org/spamassassin/UsingOnWindows.
Сервер также поддерживает анти-спам модули расширения от программы The Bat (http://www.ritlabs.com/ru/products/thebat/plugin.php). Некоторые из них слишком ориентированы на The Bat и с сервером работать отказываются. Из проверенных можно назвать Bayes Filter Plugin.

Bayes Filter Plugin v2.0.4 for The Bat!
http://www.ritlabs.com/download/files3/the_bat/plugins/antispam/bayesfilter2.0.4.exe

Параметр "No Report Dialog Below" нужно установить равным 0, чтобы отключить экранные сообщения. Пути к файлам лучше указать абсолютные, чтобы быть

уверенным в их местонахождении.
Статистика на закладке "About" обновляется не сразу, изменения становятся видны после перезагрузки. Скорее всего это связано с тем, что для ускорения работы информация записывается в базу не сразу, а сначала кэшируется в памяти. Поскольку в Конфигураторе загружается другой экземпляр модуля, чем в сервере, изменение статистики видно не сразу.

Для нормальной работы многих анти-спам решений требуется предварительная тренировка. Т.е. нужно передать им для обучения наборы писем которые, по мнению пользователя, являются спамом и не спамом. Это нужно для выделения характерных признаков таких писем, чтобы в дальнейшем можно было отличать их друг от друга. Тренировку желательно производить регулярно, так как новые виды спама появляются постоянно. Для тренировки есть два специальных адреса: $SPAM и $NONSPAM на которые нужно посылать, соответственно, спам и не спам. Чтобы отправлять письма на эти адреса обязательно нужна авторизация. Пользователь при этом должен входить в группу с правом тренировки анти-спам модулей.

Как сделать для сервера сертификат?
Для того, чтобы можно было использовать безопасное SSL/TLS подключение необходимо, чтобы на сервере был установлен сертификат. Для хранения сертификата и соответствующего ему секретного ключа может использоваться подпапка PEMs.
Сервер поддерживает SSL/TLS при помощи библиотеки OpenSSL (http://www.openssl.org). Поэтому для хранения сертификатов и секретных ключей используется pem-формат. Этот формат позволяет хранить сертификаты в текстовом виде, что удобно при их пересылке по почте.
Можно использовать как самоподписанный сертификат, так и сертификат полученный от сертификационного центра. В случае самоподписанного сертификата его нужно будет добавлять в список доверенных на почтовом клиенте, если же сертификат получен от сертификационного центра, то скорее всего он сразу будет доверенным, так как корневые сертификаты сертификационных центров уже находятся в списке доверенных.
Для работы потребуется утилита openssl, которая входит в состав библиотеки OpenSSL. Можно собрать библиотеку из исходных кодов, а можно скачать уже в собранном виде. Версию для Windows можно найти здесь: http://www.openssl.org/related/binaries.html.

Прежде всего нужно сгенерировать секретный ключ RSA и запрос на сертификацию. Для этого нужно сначала создать файл .rnd в который записать произвольные случайные данные, на основе которых будет генерироваться ключ. Можно скопировать в него файл с оцифрованным шумом или текстовый файл со случайно набранными словами и фразами. Размер файла значения не имеет. После этого нужно будет выполнить из командной строки:

```
openssl req -newkey rsa:1024 -keyout key.pem -out req.pem -config openssl.cnf
```

Эта команда создаст 1024-битный секретный ключ RSA и запишет его в файл key.pem. Ключ должен выглядеть примерно так:

```
-----BEGIN RSA PRIVATE KEY-----
MIIBOwIBAAJBALtv55QyzG6i2PlwZ1pah7++Gv8L5j6Hnyr/uTZE1NLG0ABDDexm
q/R4KedLjFEIYjocDui+IXs62NNtXrT8odkCAwEAAQJAbwXq0vJ/+uyEvsNgxLko
nWmM1KvqnAo5uQIhALqEADu5U1Wvt8UN8UDGBRPQulHWNycuNV45d3nnskWPAiAw
ueTyr6WsZ5+SD8g/Hy3xuvF3nPmJRH+rwvVihlcFOg==
```

```
-----END RSA PRIVATE KEY-----
```

При создании ключа будет также запрошен пароль для него. Этот пароль впоследствии надо будет ввести в настройках сервера.

Также будет создан запрос на сертификацию. В запросе содержится описательная информация будущего сертификата: Страна, Штат/Область, Город, Организация, Подразделение, Доменное имя сервера и т.д. В процессе выполнения команды будут запрошены атрибуты будущего сертификата. Не забудьте в качестве атрибута "Common Name" указать доменное имя сервера. Эта команда создаст на основе секретного ключа из файла key.pem и данных введенных пользователем запрос на сертификацию и запишет его в файл req.pem. Запрос должен выглядеть примерно так:

```
-----BEGIN CERTIFICATE REQUEST-----
MIIBGzCBxgIBADBjMQswCQYDVQQGEwJBVTETMBEGA1UECBMKUXVlZW5zbGFuZDEa
MBgGA1UEChMRQ3J5cHRTb2Z0IFB0eSBMdGQxIzAhBgNVBAMTGkNsaWVudCB0ZXN0
2NNtXrT8odkCAwEAATANBgkqhkiG9w0BAQQFAANBAC5JBTeji7RosqMaUIDzIW13
oO6+kPhx9fXSpMFHIsY3aH92Milkov/2A4SuZTcnv/P6+8klmS0EaiUKcRzak4E=
-----END CERTIFICATE REQUEST-----
```

Теперь у Вас есть секретный ключ в файле key.pem и запрос на сертификацию в файле req.pem. Обязательно сохраните секретный ключ в безопасном месте. Он потребуется для нормального функционирования сервера при использовании безопасных соединений. Без секретного ключа сертификат будет практически бесполезен. По соображениям безопасности, сертификационные центры не смогут выписать вам новый сертификат (например, при изменении атрибутов) если у Вас нет доступа к секретному ключу.

Теперь настало время послать Ваш запрос на сертификацию сертификационному центру. Это может быть VeriSign (http://www.verisign.com) или Thawte (http://www.thawte.com). После проверки ваших данных сертификационный центр пришлет Вам сертификат или сообщит причину отказа. Сертификат нужно поместить в файл cert.pem. Он должен выглядеть примерно так:

```
-----BEGIN CERTIFICATE-----
MIICLjCCAZcCAQEwDQYJKoZIhvcNAQEEBQAwWzELMAkGA1UEBhMCQVUxEzARBgNV
BAgTClF1ZWVuc2xhbmQxGjAYBgNVBAoTEUNyeXB0U29mdCBQdHkgTHRkMRswGQYD
dp7jnmWZwKZ9cXsNUS2o4OL07qOk2HOywC0YsNZQsOBu1CBTYYkIefDiKFL1zQHh
8lwwNd4NP+OE3NzUNkCfh4DnFfg9WHkXUlD5UpxNRJ4gJA==
-----END CERTIFICATE-----
```

Если нужно создать самоподписанный сертификат, то нужно выполнить из командной строки:

```
openssl req -new -key key.pem -out cert.pem -x509 -config openssl.cnf
```

Такой сертификат больше подходит для использования в тестовых целях. Можно также получить тестовые сертификаты от VeriSign по адресу http://digitalid.verisign.com/test_server_ids.html (выберите "C2Net (Apache-SSL-US)") или от Thawte по адресу https://www.thawte.com/cgi/server/test.exe (выберите "Generate an X.509v3 certificate & Use the most basic format").

По умолчанию сервер ищет секретный ключ и сертификат в файле ".\PEMs\server.pem". Можно скопировать их в этот файл при помощи текстового редактора или при помощи команды:

```
copy key.pem+cert.pem server.pem
```

Либо можно указать нужные файлы в Конфигураторе в разделе "Опции > Безопасность" ("Options > Security"). Также обязательно нужно указать пароль для

расшифровки секретного ключа.