

# Лабораторная работа № 6

## Передача почтовых сообщений по протокол SMTP

### 6.1 Цель работы

Ознакомиться с технологией работы SMTP клиента и реализация его с базовым набором команд.

### 6.2 Задание на лабораторную работу

Разработать приложение, позволяющее отправлять электронные письма с вложенными файлами, используя протокол SMTP.

### 6.3 Методические указания

**SMTP** (англ. Simple Mail Transfer Protocol — простой протокол передачи почты) — это сетевой протокол, предназначенный для передачи электронной почты в сетях TCP/IP.

#### 6.3.1 Обзор протокола

Широкое распространение SMTP получил в начале 1980-х годов. До него использовался протокол UUCP, который требовал от отправителя знания полного маршрута до получателя и явного указания этого маршрута в адресе получателя, либо наличия прямого коммутируемого или постоянного соединения между компьютерами отправителя и получателя.

Взаимодействие в рамках SMTP строится по принципу двусторонней связи, которая устанавливается между отправителем (клиент SMTP) и получателем (сервер SMTP) почтового сообщения. При этом отправитель инициирует соединение и посылает запросы на обслуживание, а получатель - отвечает на эти запросы. Фактически отправитель выступает в роли клиента, а получатель - сервера.

Протокол был разработан для передачи только текста в кодировке ASCII, кроме того, первые спецификации требовали обнуления старшего бита каждого передаваемого байта. Это не даёт возможности отсылать текст на национальных языках (например, кириллице), а также отправлять двоичные файлы (такие как изображения, видеофайлы, программы или архивы). Для снятия этого ограничения был разработан стандарт MIME, который описывает способ преобразования двоичных файлов в текстовые. В настоящее время большинство серверов поддерживают 8BITMIME, позволяющий отправлять двоичные файлы так же просто, как текст.

### 6.3.2 MIME

**MIME** (*Multipurpose Internet Mail Extensions* - многоцелевые расширения почты Интернет) - спецификации, определяющие дополнения в формате почтовых сообщений для

- пересылки восьмибитных текстов и полностью двоичных данных;
- поддержки сложных сообщений, состоящих из нескольких разделов (возможно, содержащих данные разных типов).

Формирование и разбор сообщений в соответствии со спецификациями MIME производится пользовательскими почтовыми агентами. Описание MIME содержится в RFC 2045-2049.

**Таблица 6.1**

#### Наиболее часто встречающиеся поля

Название поля	Назначение
Return-Path:	Служит для указания адреса возврата письма. Как правило, совпадает с адресом отправителя. Адрес заключается в угловые скобки.
Received:	Содержит путь прохождения письма по почтовым серверам. Полей обычно несколько и каждое такое поле обозначает один сервер в маршруте. Серверы указываются в обратном хронологическом порядке, то есть каждый новый сервер вставляет свою подпись сверху. Формат таков: Received: from <домен> - отправитель, by <домен> - получатель, via <путь> - физический путь, with <протокол> - почтовый протокол, id <идентификатор> - идентификатор сообщения у получателя, for <адрес> - финальный адрес получателя, <дата/время> - время получения сообщения. Обязательно только указание времени, остальная информация может быть пропущена, хотя обычно присутствует большинство из описанных данных.
Message-ID:	Уникальный идентификатор сообщения. Как правило, идентификатор составляется из текущей даты, времени, адреса компьютера и некоей случайной величины. Записывается идентификатор в угловых скобках.

**Продолжение табл. 6.1**

From:	Автор сообщения. Указывается адрес электронной почты автора в угловых скобках, но нередко указывается еще и реальное имя.
Sender:	Лицо или служба, отправившая сообщение, в том случае, если не совпадает со значением поля "From". Довольно редко встречающееся поле, попадает иногда в различных спам-письмах.
To:	Основной получатель. Указывается адрес электронной почты в угловых скобках. Можно указать еще и реальное имя, которое впрочем, будет с успехом проигнорировано почтовыми службами. Можно указать несколько получателей через запятую.
Cc:	Carbon Copy. Дополнительные получатели, синтаксис такой же, как для поля "To".
Bcc:	Blind Carbon Copy. Дополнительные получатели, адреса которых не должны быть видны другим получателям. Очень полезное поле! Используется когда, например, надо известить многих лиц об одном и том же, но не нужно, чтобы содержание вашей адресной книги попало ко всем корреспондентам. Синтаксис аналогичен полям "To" и "Cc".
Date:	Дата написания письма, хотя некоторые системы выставляют дату отправки письма. В стандартной ситуации это все-таки дата написания письма, а дата отправки определяется из первого поля "Received".
Subject:	Тема письма.
MIME-Version:	Наличие этого поля говорит о том, что текст письма форматирован в соответствии со стандартом MIME. Значение поля говорит об используемой версии MIME. Подробное описание стандарта MIME см. в RFC 1521. Например: "MIME-Version: 1.0".
In-Reply-To:	Ссылка на письмо, ответом на которое является настоящее письмо. Каждое письмо имеет свой уникальный идентификатор, который проставляется в поле "Message-ID".
References:	Ссылки, на другие письма, относящиеся к данному. Является обобщением поля "In-Reply-To".
X-Mailer:	Информация о почтовой программе, использованной для написания письма.

**Продолжение табл. 6.1**

Priority:	Приоритет письма. Бывает "обычным", "срочным" и "не срочным". Как правило, используются фразы: "Normal", "Urgent", "Non-urgent". Может оказывать влияние на скорость обработки и передачи письма различными промежуточными почтовыми системами.
X-Priority:	Тоже приоритет письма, но, в отличие от "Priority", обозначается цифрами.
Importance:	Пометка от отправителя получателю о важности сообщения. Может принимать значения "High", "Normal" или "Low". В отличие от поля "Priority", никак не влияет на скорость пересылки.
Content-Type:	Формат содержимого письма. Определяется тип информации в письме и способ ее представления. В частности задается кодировка письма, если используется какой-либо национальный набор символов.
Content-Transfer-Encoding:	Метод кодирования, используемый в письме, согласно стандарту MIME.

Приведенная краткая таблица не претендует на абсолютную полноту. Существует масса устаревших или, напротив, новых (экспериментальных) полей заголовка e-mail. Например, Microsoft использует поля "X-MimeOLE" и "X-MSMail-Priority", которые являются нестандартными. Строго говоря, все поля, начинающиеся с "X-" (так называемые, private-поля), не являются стандартными, а используются по внутрифирменным соглашениям. Существуют редко используемые стандартные поля, например: "Keywords", "Comments", "Content-Description" - служащие для описания содержимого письма. Но в подавляющем большинстве случаев вышеприведенной таблицы вполне достаточно для прочтения и понимания содержания заголовка. Впрочем, для полноты картины эту таблицу желательно дополнить более подробным описанием стандарта MIME (Multipurpose Internet Mail Extensions), поскольку многие поля так или иначе относятся именно к MIME, информацию по которому можно найти в RFC 1521 и RFC 1522.

Итак, MIME предназначен для задания способов передачи посредством электронной почты различной нетекстовой информации, а также больших объемов информации, которые необходимо посылать в нескольких письмах, разбив на кусочки. Следует заметить, что под нетекстовой информацией надо понимать не только бинарные файлы, но и тексты в нестандартной кодировке, то есть в отличной от ASCII, описанной в ANSI X3.4-1986.

Одно из основных полей, касающихся MIME, - это "Content-Type". Возможные его значения приведены в таблице.

**Таблица 6.2**

Название поля	Назначение
text	Текстовая информация в виде набора символов или описания текста на стандартном языке.
multipart	Служит для объединения нескольких частей (возможно, разных типов) сообщения в одно письмо.
application	Служит для пересылки бинарных файлов приложений или данных посредством электронной почты.
message	Указывает на инкапсуляцию другого письма.
image	Служит для пересылки изображений.
audio	Служит для пересылки звуковой информации.
video	Служит для пересылки видеоинформации.

В поле "Content-Type" нужно указать подтип информации. Например, для поля "text" используются в основном подтипы "plain" - означающий, что текст обычный и неформатированный, и "html" - указывающий на то, что сообщение в формате HTML. Если отправляется сообщение в формате HTML, то правильная почтовая программа создаст две части письма: в "text/plain" и "text/html", чтобы в случае непонимания подтипа "html" на получающей стороне не возникло неудобств. С другой стороны, правильная почтовая программа получателя должна показать письмо в максимально информативном виде и поэтому, пропустив "text/plain"-часть сообщения, отобразит "text/html". В общем случае, текст может быть форматирован любым способом, понимаемым обеими сторонами. Это декларировано в RFC 1341. В поле "text" также указывается кодовая таблица. Например: "Content-type: text/plain; charset=us-ascii".

Для поля "multipart" можно указывать подтипы: "alternative" - на случай, если несколько частей представляют одно и то же, а надо выбрать одну из них для отображения; "parallel" - если надо отобразить части одновременно; "digest" - если каждая часть имеет тип "message"; и некоторые другие подтипы. Для поля "message" основной подтип - "rfc822". Еще пара используемых подтипов: "partial" - для посылки части сообщения и "External-body" - для передачи, например, объемной информации путем ссылки на внешний источник данных.

Подтип поля "image" задает графический формат, в котором пересылается изображение. Основные форматы - jpeg и gif. Поле "audio" имеет основной подтип "basic". Поле "video" - основной подтип "mpeg".

Поле "application" имеет основной подтип "octet-stream" в случае обычных бинарных данных. Если посылается файл известного типа, то указывается его тип. Так, для MS Word документов Outlook пишет "application/msword", а для MS Excel "application/vnd.ms-excel".

В соответствии с RFC 822, все письма, по умолчанию, передаются простым текстом в кодировке US-ASCII, что соответствует "Content-type: text/plain; charset=us-ascii".

Следующее важнейшее поле, касающееся MIME, - это "Content-Transfer-Encoding". Служит оно для указания способа кодирования в случаях, когда посылаемое письмо содержит что-либо не уместяющееся в рамках US-ASCII. Стандартизованы следующие способы:

**Таблица 6.3**

Название поля	Назначение
7bit	Считается, что длина одной строки сообщения не более 1000 символов, а сами строки состоят только из US-ASCII-символов.
8bit	Строки также ограничены по длине, но могут использоваться символы из старшей половины кодовой таблицы.
binary	Нет ограничений ни на длину, ни на содержание.
base64	Письмо кодировано в соответствии со стандартом "base64".
quoted-printable	Письмо кодировано в соответствии со стандартом "quoted-printable".

Разработчики почтовых систем могут использовать свои (оригинальные) способы кодирования, с тем лишь условием, чтобы принимающая сторона смогла корректно декодировать информацию.

Вторая задача MIME - это стандартизация разбиения больших писем на несколько кусочков. Для этого в поле "Content-Type" после значения "multipart/" указывается строка - уникальный ограничитель фрагментов "boundary=". А затем перед каждым фрагментом пишется эта строка, предваренная двумя минусами, а в конце фрагментации еще одна, завершающаяся такими же двумя минусами.

### 6.3.3 Пример письма

Received: from ada.vvsu.ru (ada.vvsu.ru [212.16.195.70]) by maria.vvsu.ru  
(8.8.3/8.8.3) with SMTP id SAA04920 for <fire@maria.vvsu.ru>;  
Tue, 18 Jan 2000 18:58:00 +1000 (VVO)

Message-ID: <38842ABB.39CC@vvsu.ru>

Date: Tue, 18 Jan 2000 18:56:27 +1000

From: Maxim Mamayev <m2@vvsu.ru>

X-Mailer: Mozilla 3.0 (Win95; I)

MIME-Version: 1.0

To: fire@maria.vvsu.ru

Subject: [Fwd: test mixed]

Content-Type: multipart/mixed; boundary="-----20EADB04695"

This is a multi-part message in MIME format.

-----20EADB04695

Content-Type: text/plain; charset=koi8-r

Content-Transfer-Encoding: 8bit

Текст, добавленный при форвардинге

-----20EADB04695

Content-Type: message/rfc822

Content-Transfer-Encoding: 8bit

Content-Disposition: inline

Message-ID: <388412B0.3522@vvsu.ru>

Date: Tue, 18 Jan 2000 17:13:53 +1000

From: Maxim Mamayev <m2@vvsu.ru>

X-Mailer: Mozilla 3.0 (Win95; I)

MIME-Version: 1.0

To: fire@maria.vvsu.ru

Subject: test mixed

Content-Type: multipart/mixed; boundary="simple boundary"

This is a multi-part message in MIME format.

--simple boundary

Content-Type: text/plain; charset=koi8-r

Content-Transfer-Encoding: 8bit

Текст исходного сообщения

--simple boundary

Content-Type: image/jpeg; name="test.jpg"

Content-Transfer-Encoding: base64  
Content-Disposition: inline; filename="test.jpg"

YWJjZGVm  
--simple boundary

-----20EADB04695--

#### **6.3.4 Команды SMTP**

Сервер SMTP — это конечный автомат с внутренним состоянием. Клиент передает на сервер строку команда<пробел>параметры<перевод строки>.



Таблица 6.4

## Команды простого протокола передачи почты (SMTP)

Команда	Обязательна	Описание
<b>HELO</b> <i>hostname</i>	X	первая команда сеанса, <i>hostname</i> - доменное имя вызывающего хоста (клиента). Идентифицирует модуль-передатчик для модуля-приемника (hello).
<b>MAIL FROM:</b> <i>email_адрес_отправителя</i>	X	обратный адрес
<b>RCPT TO:</b> <i>email_адрес_получателя</i>	X	адрес получателя (в случае нескольких адресатов команда повторяется для каждого адресата). Идентифицирует получателя почтового сообщения (recipient).
<b>DATA</b>	X	начало ввода текста сообщения; сервер посылает промежуточный положительный отклик 354 и рассматривает все последующие строки в качестве текста (тела) сообщения; конец ввода - строка состоящая из одной точки. Перед текстом сообщения вводятся поля заголовка (см. выше п. Структура email-сообщения). Каждое поле заголовка должно начинаться с новой строки. Между заголовком и текстом сообщения должна быть <i>одна пустая строка</i> .
<b>RSET</b>		сброс сеанса к начальному состоянию (как после ввода HELO). Прерывает текущую почтовую транзакцию (reset).

Продолжение таблицы 6.4

Команда	Обязательна	Описание
<b>NOOP</b>		Требует от получателя не предпринимать никаких действий, а только выдать ответ ОК. Используется главным образом для тестирования.(No operation).
<b>QUIT</b>		Требует выдать ответ ОК и закрыть текущее соединение.
<b>VRFY</b>		выдается положительный отклик (250,251 или 252), если сервер может <i>попытаться</i> доставить сообщение по указанному адресу; иначе выдается отрицательный отклик 550. Если <i>email_адрес</i> - не локальный адрес на сервере, то положительный отклик не обязательно означает, что этот адрес существует.
<b>HELP</b>		Команда SMTP-серверу вернуть сообщение-справку о его командах.
<b>TURN</b>		Команда SMTP-серверу либо сказать ОК и поменяться ролями, то есть стать SMTP- клиентом, либо послать сообщение-отказ и остаться в роли SMTP-сервера.

### **6.3.5 Коды ответов SMTP**

Сервер отвечает на каждую команду строкой, содержащей код ответа и текстовое сообщение, отделенное пробелом. Код ответа — число от 100 до 999, представленное в виде строки, трактующийся следующим образом:

2XX — команда успешно выполнена

3XX — ожидаются дополнительные данные от клиента

4XX — временная ошибка, клиент должен произвести следующую попытку через некоторое время

5XX — неустраняемая ошибка

Текстовая часть ответа носит справочный характер и предназначена для человека, а не программы

### 6.3.6 Пример простейшей SMTP-сессии

Таблица 6.5

Пример простейшей SMTP-сессии

1	RECEIVER	220 mail.ru ESMTP Thu, 19 Nov 2009 23:22:50 +0300
2	SENDER	HELO user
3	RECEIVER	250 mx75.mail.ru Hello user [158.259.135.56]
4	SENDER	AUTH LOGIN
5	RECEIVER	334 VXNlcm5hbWU6
6	SENDER	Логин, закодированный base64
7	RECEIVER	334 UGFzc3dvcmQ6
8	SENDER	Пароль, закодированный base64
9	RECEIVER	235 Authentication succeeded
10	SENDER	MAIL FROM: Smith@mail.ru
11	RECEIVER	250 OK
12	SENDER	RCPT TO: Jones@mail.ru
13	RECEIVER	250 Accepted
14	SENDER	DATA
15	RECEIVER	354 Enter message, ending with "." on a line by itself
16	SENDER	Subject: This is mail theme
17	SENDER	TO: Jones@mail.ru
18	SENDER	X-Mailer: mail
19	SENDER	Текст письма
20	SENDER	.
21	RECEIVER	250 OK id=1NBJYW-000GKw-00
22	SENDER	QUIT
23	RECEIVER	221 mx75.mail.ru closing connection

Как видно из строки 1, когда SMTP-клиент устанавливает TCP-соединение с портом протокола 25, SMTP-сервер отвечает кодом 220. Это означает, что соединение успешно установлено:

**1. RECEIVER: 220 mail.ru ESMTP Thu, 19 Nov 2009 23:22:50 +0300**

После установления соединения, первой командой, согласно спецификации, должна быть команда HELO. Как указано в строке 2, SMTP-клиент передает HELO, указывая имя своего компьютера в качестве аргумента. Другими словами, он сообщает: <Привет, я - user>. Команда HELO употребляется с аргументом, как показано ниже:

**2. SENDER: HELO user**

В ответ на HELO приемник выдает код 250, сообщая передатчику о том, что команда принята и обработана:

**3. RECEIVER: 250 mx75.mail.ru Hello user [158.259.135.56]**

После приветствия необходимо авторизироваться, для этого нужно отправить команду

**4. SENDER: AUTH LOGIN**

После, нужно последовательно ввести логин и пароль, закодированные base64(п 6, 8).

После авторизации SMTP-клиент приступает к почтовой транзакции. Для начала он выполняет команду:

**10. SENDER: MAIL FROM: Smith@mail.ru>**

MAIL <пробел> FROM:<reverse-path> <carriage-return line-feed>

Аргумент <обратный путь> (reverse path) указывает серверу, кому в случае ошибки отослать соответствующее сообщение. Мы еще рассмотрим его подробнее. На данный момент для нас важно, что в аргументе содержится адрес источника сообщения (в нашем случае, Smith@mail.ru). После того как сервер выдал код ответа 250 (строка 11), согласившись обработать сообщение от Smith@mail.ru необходимо указать получателя сообщения. Это делается при помощи команды RCPT. Команда RCPT имеет аргумент - имя получателя. На одну команду приходится только одно имя, поэтому, если получателей несколько, команда RCPT выдается несколько раз. В нашем примере команды RCPT выполняются в строке 12. Синтаксис RCPT похож на синтаксис команды MAIL:

RCPT <пробел> TO:<forward-path> <CRLF>

Однако, в отличие от MAIL, аргумент RCPT начинается со слова <TO:>. Содержимое аргумента - путь передачи сообщения (forward path), а не обратный путь. На данный момент для нас важно, что в пути передачи сообщения указано имя почтового ящика получателя. Выдав команду RCPT, МТА-клиент ожидает получить ответ с кодом 250.

## **12. SENDER: RCPT TO: Jones@mail.ru**

После того как посланы все команды RCPT, клиент начинает передачу данных при помощи команды DATA. В строке 14 показано, как МТА-клиент (передатчик) высылает команду DATA, в строке 15 - как сервер отвечает кодом 354. Этот код означает, что передача данных разрешена и должна заканчиваться комбинацией CRLF-<точка>-CRLF (новой строкой, содержащей только точку).

## **14. SENDER: DATA**

## **15. RECEIVER: 354 Enter message, ending with "." on a line by itself**

После того как получен код 354, клиент может начать передачу данных. Заполняем тему письма

## **16. SENDER Subject: This is mail theme**

Еще раз вводится адрес получателя, но уже используется только команда TO

## **17. SENDER: TO: Jones@mail.ru**

И поле X-Mailer вводится информация о почтовой программе, использованной для написания письма (если это поле будет не заполнено, то письмо будет помечено как спам и не будет отправлено!)

## **18 SENDER X-Mailer: mail**

Пишем тело письма. Сервер не высылает никаких ответов до тех пор, пока не получит комбинацию CRLF-точка-CRLF от клиента, означающую конец передачи данных. Как показано в строках 16 и 17, в ответ на полученную комбинацию CRLF-<точка>-CRLF, сервер выдает код 250. Как мы уже говорили, код ответа 250 означает успешное окончание операции:

## **20. SENDER: .**

## **21. RECEIVER: 250 OK id=1NBJYW-000GKw-00**

Для того чтобы закончить почтовую транзакцию, клиент, по правилам SMTP, обязан послать команду QUIT. Сервер, в свою очередь, отвечает кодом 221. Этот код подтверждает клиенту, что соединение будет закрыто:

## **22. SENDER: QUIT**

## **23. RECEIVER: 221 mx75.mail.ru closing connection**

В любой момент во время транзакции клиент может использовать команды NOOP, HELP, EXPN и VRFY. В ответ на каждую команду сервер высылает клиенту определенную информацию. Конечно, в зависимости от ответа клиент может предпринять определенные действия, однако спецификация SMTP ничего не говорит по этому поводу. Например, клиент-МТА может передать команду VRFY для того, чтобы убедиться, что имя пользователя действительно. Если сервер ответит, что данного имени не существует, клиент МТА может не передавать почту для этого пользователя. В спецификации SMTP, однако, на этот счет нет никаких указаний - клиент может ничего не делать в ответ на команду VRFY. МТА-клиент может ничего не делать также в ответ на команды NOOP, HELP и EXPN - ответственность целиком лежит на разработчике конкретной реализации МТА.

**ESMTP** — расширяемый протокол, в отличие от SMTP. При установлении соединения сервер объявляет о наборе поддерживаемых расширений (в качестве ответа на команду EHLO). Соответствующие расширения могут быть использованы клиентом при работе. Необходимо помнить, что если сессия начинается с команды HELO (используемой в «классическом» SMTP, RFC 821), то список расширений выводиться не будет.

#### **6.4 Контрольные вопросы и задания**

1. Объяснить назначение протокола POP3.
2. Описать отличия протокола ESMTP от SMTP.