

Електронна поща

проф. д-р инж. Венета Алексиева

ОСНОВНИ МОМЕНТИ

- Електронна поща
- SMTP
- POP3
- IMAP
- Конфигуриране

Електронната поща преди

- 1965г. се появява за копиране съдържанието на файл от една машина на друга като го добавя в т.н. *пощенска кутия (mailbox)* на получателя. Позволява трансфер само на текст.
- 1971г. в ARPANET се създава 1-во приложение за електронна поща – знакът **@** **отделя името на потребителя от името на машината**
- 1977г.- RFC733- за унифициране на е-пощата
- 1982г.- RFC822 – за унифициране имената на домейни
- 1982г.- RFC821(RFC5321 от 2008г.) – SMTP, който заменя FTP за обмен на е-поща
- 1996г.- RFC2045- RFC2049 - MIME (**Multipurpose Internet Mail Extensions**)

Електронната поща сега

- Електронната поща не може да функционира без DNS система
- Всяко електронно съобщение включва заглавни части (хедъри) и тяло
- Протоколите SMTP, POP и IMAP използват команди в ASCII текст за управление на обмена
- Използва се порт 25

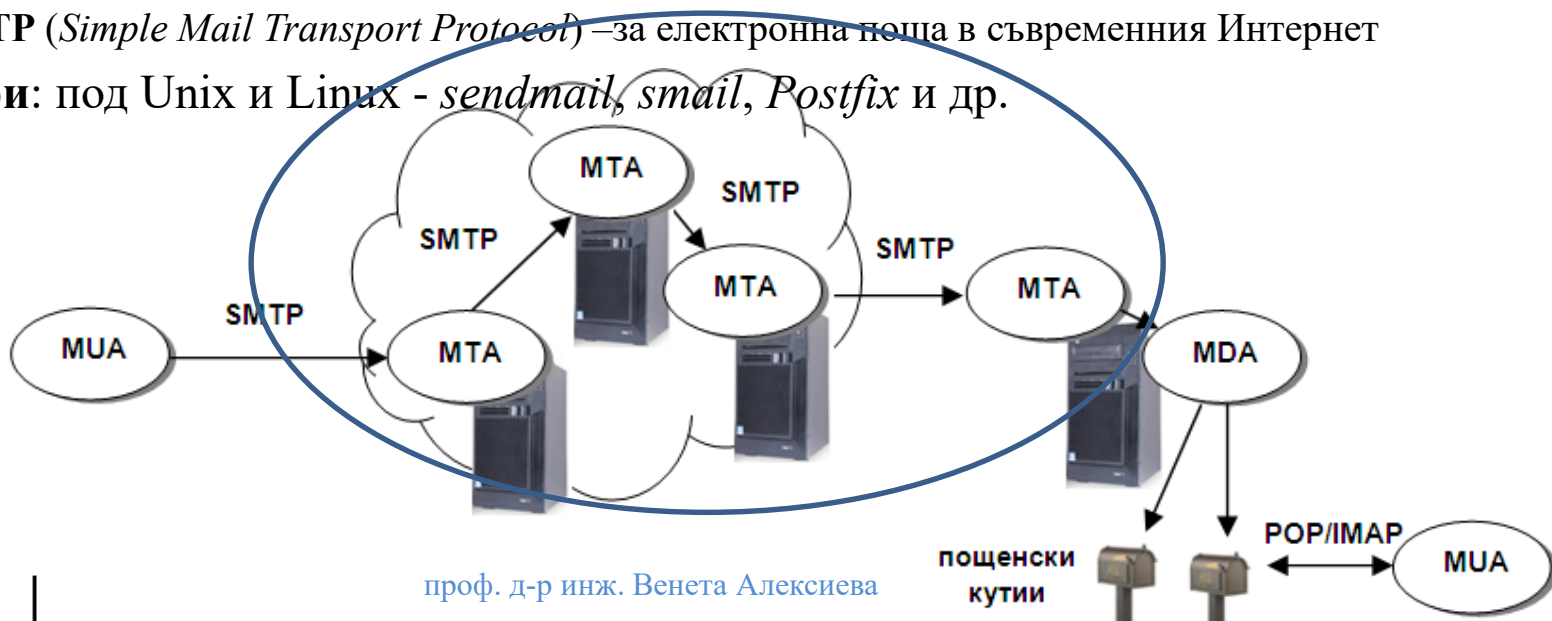
ОСНОВНИ КОМПОНЕНТИ

- **MUA** (*Mail User Agent*)
 - При предаване-формиране на e-mail съобщение и изпращането му към друга програма(MTA), която да осъществи неговото доставяне.
 - При получаване - прочитане на съобщенията от пощенската кутия на потребителя, класифициране и обработка.
- **Примери:** *mail* и *pine* под Linux, *Pegasus Mail* и *Outlook Express* под Windows.



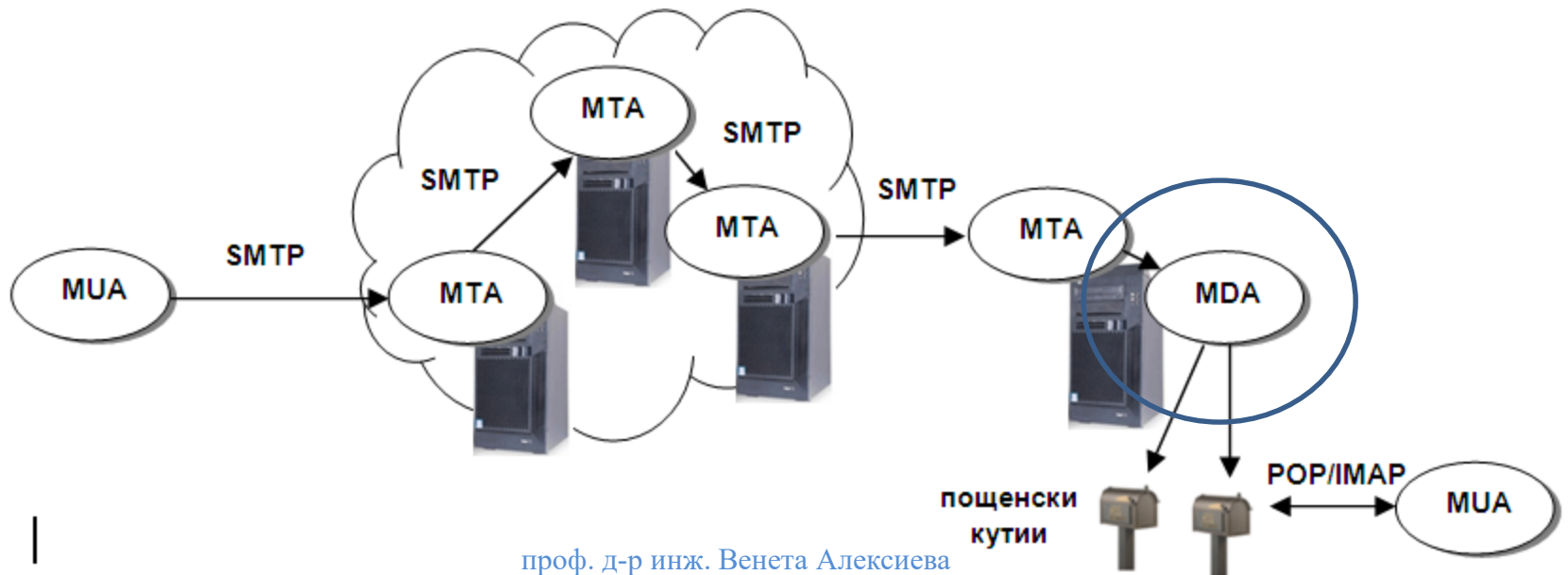
ОСНОВНИ КОМПОНЕНТИ

- **MTA** (*Mail Transport Agent*) е специализирана програма, осъществяваща транспортирането на електронната поща между машините и нейната доставка до получателя.
 - получава подготвените съобщения от MUA,
 - определя дестинацията,
 - препраща писмата (*relay*) през други междинни MTA до машината на получателя – т.н *пощенски сървър* (*Mail server*).
- **Протоколи:**
 - **UUCP** (*Unix to Unix Copy*) – за обмен на електронна поща между компютърни системи под Unix;
 - **X.400** – за електронна поща в системи, базирани на X.25 мрежови протокол;
 - **SMTP** (*Simple Mail Transport Protocol*) – за електронна поща в съвременния Интернет
- **Примери:** под Unix и Linux - *sendmail, smail, Postfix* и др.

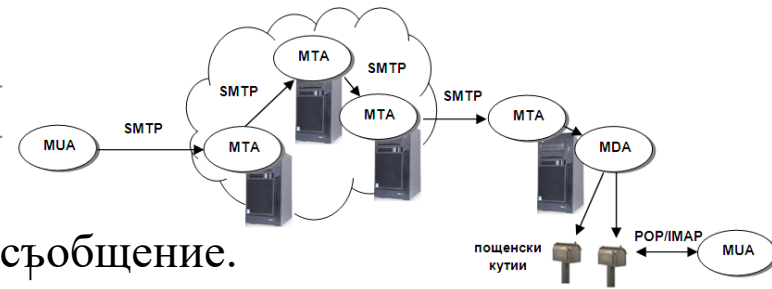


ОСНОВНИ КОМПОНЕНТИ

- **MDA** (*Mail Delivery Agent*) осигурява доставянето на полученото от пощенския сървър писмо в пощенската кутия на конкретния получател.
 - използване на псевдоними (*aliases*)
 - пренасочване (*forwarding*)
 - списъци с получатели на електронна поща (*mail lists*)
- **Примери:** под Linux - *mail*.



СТЪПКИ



- **Създаване на електронната поща.**
 - MUA (от компютъра на потребителя) формира съобщение.
- **Изпращане на съобщението.**
 - MUA изпраща готовото писмо на съответния MTA по SMTP (RFC2822) или по неговото разширение ESMTP.
- **Маршрутизиране на съобщението.**
 - Ако получателят е на същата машина, MTA извършва локално доставяне, с използване на услугите на локалния MDA.
 - Ако получателят е на отдалечена машина, MTA осигурява предаването на съобщението до дестинацията по SMTP или ESMTP.
- **Разпределяне на съобщенията.**
 - След получаване на електронната поща в крайния пощенски сървър, тя се обработва от съответния MDA и се записва в пощенската кутия на получателя.
- **Прочитане на съобщението.**
 - Потребителят прочита полученото писмо от своята пощенска кутия с **POP3** (*Post Office Protocol ver.3*), чрез който съдържанието на пощенската кутия на потребителя се прехвърля на локалния му компютър или **IMAP4** (*Internet Message Access Protocol ver.4*), чрез който потребителят може освен това и да класифицира, изтрива и поддържа своите електронни писма динамично на пощенския сървър.

Адреси при електронната поща

Адрес на получателя:

идентификатор_на_локалния_потребител @ описание_на_домейн

- *идентификатор_на_локалния_потребител* (изпращача или получателя)
 - потребителското име на съответната машина (*login name*)
 - Реалното име във формат *ПървоИме.ПоследноИме*
 - псевдоним, който ще бъде преобразуван в потребителско име
 - списък от потребители
- *описание_на_домейн* (пощенския сървър на изпращача или на получателя)
 - това е пълното квалифицирано име на домейн (FQDN) във вида *host.domain*.
 - Това обозначение не описва пътя до крайната машина.

Структура на съобщение за електронна поща

Административна част:

Envelope:

Адреси на източника и на
получателя

Mail header: темата на съобщението,
списък от множество получатели,
времето на изпращане на
съобщението и др

Празен ред

Body

```
Return-Path: <bounces+6683617-16b3-valeksieva=tu-varna.bg@em9200.varnaparking.bg>
Delivered-To: <valeksieva@tu-varna.bg>
Received: from mail.tu-varna.bg
    by mail.tu-varna.bg (Dovecot) with LMTP id uV2+BrCjG19qFwAAAbfQXw
    for <valeksieva@tu-varna.bg>; Fri, 24 Jul 2020 12:02:40 +0300
Received-SPF: Pass (sender SPF authorized) identity=mailfrom; client-ip=208.117.55.132;
helo=o1.f.az.sendgrid.net; envelope-from=bounces+6683617-16b3-valeksieva=tu-
varna.bg@em9200.varnaparking.bg; receiver=valeksieva@tu-varna.bg
DKIM-Filter: OpenDKIM Filter v2.11.0 mail.tu-varna.bg 3343120865
Authentication-Results: mail.tu-varna.bg;
    dkim=pass (1024-bit key) header.d=varnaparking.bg header.i=@varnaparking.bg
    header.b="i+RBROSD"
Received: from o1.f.az.sendgrid.net (o1.f.az.sendgrid.net [208.117.55.132])
    by mail.tu-varna.bg (Postfix) with ESMTP id 3343120865
    for <valeksieva@tu-varna.bg>; Fri, 24 Jul 2020 12:02:38 +0300 (EEST)
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/relaxed; d=varnaparking.bg;
    h=content-type:from:subject:content-transfer-encoding:mime-version:to;
    s=s1; bh=S9ALPLn2BnLmUipwIjAUsm3AGDX3FhlB1/zkO+3qNM=;
    b=i+RBROSDzRBMccNy5k5JJHpuyU0f/BV7926LtgWc2iAMQkYZsOTB2KS3EZ0e8bSsSv+d
    jhkrhzb1KctNxB4G13w9uBpcJ/ZU7l0uSNkYznByZsv7PfjocQuq1tbYy6daLRI7B1tAI
    eEWypI8vj+wCVbVcKHg3WD88GEaKwabrS=
Received: by filterdrecv-p3mdw1-75c584b9c6-hrjh5 with SMTP id filterdrecv-p3mdw1-
75c584b9c6-hrjh5-19-5F1AA3AD-14
    2020-07-24 09:02:37.194152638 +0000 UTC m=+2389979.532371553
Received: from [127.0.0.1] (unknown)
    by ismtpd0005p1lon1.sendgrid.net (SG)
    with ESMTP id YPMCUDq1QGq16uXGIGmORg
    for <valeksieva@tu-varna.bg>;
    Fri, 24 Jul 2020 09:02:37.036 +0000 (UTC)
Content-Type: text/html; charset=utf-8
From:
    =?utf-8?q?=D0=A1=D0=B8=D0=BD=D1=8F_=D0=B7=D0=BE=D0=BD=D0=B0_=D0=92=D0=B0?=
    =?utf-8?q?=D1=80=D0=BD=D0=B0?= <noreplay@varnaparking.bg>
Subject: =?UTF-8?B?0J7QtNC+0LHRgNC10L1vINC30LDRj9Cy0LvQtdC90LjQtQ==?=
    =?UTF-8?B?INC30LAG0L/QsNGA0LrQuNGA0LDQvdC1?=
Message-ID: <68f33271-911a-1bd8-d9c7-fdc9a608313@varnaparking.bg>
Content-Transfer-Encoding: base64
Date: Fri, 24 Jul 2020 09:02:37 +0000 (UTC)
MIME-Version: 1.0
X-SG-EID:
```

Полета

- Подател
 - **From:** - Адрес на автора на писмото;
 - **Sender:** - Потребителския софтуер, използван за съставяне на писмото;
 - **Reply-To:** - Адрес на отговора, често използван от мейл листи;
- Получатели
 - **To:** - Основен списък от получатели, разделени с ‘,’;
 - **Cc:** - Списък с други получатели, към които ще се изпратят копия на писмото;
 - **Bcc:** - Списък с други получатели, към които ще се изпратят копия на писмото, без обаче те да бъдат информирани за останалите получатели;
- По време на изпращането
 - **Date:** - Датата на изпращане на писмото;
 - **Message-Id:** - Уникален идентификатор на съобщението, генериран от МТА на подателя;
 - **Subject:** - Темата на съобщението;
- По време на маршрутизирането
 - **Received:** - Всеки междинен МТА, които получава писмото, добавя в това поле името на своята машина, датата и идентификатора на съобщението, откъде е получено и името на софтуера, който го е транспортирал;
 - **Status:** - Индикация за статуса на всяко получено писмо.

MIME (Multipurpose Internet Mail Extension)

- Изпращане на разнообразна информация през Интернет;
- Кодиране и прикачане на двоични (не-ASCII) данни към съобщенията;
- Използване на много типове данни в един и същ документ;
- Средства за идентифициране на типа на съдържанието;
- Неограничена дължина на съобщението.

Content-type: type/subtype [;parameters]

MIME дефинира типа данни посредством 3 описателни части:

- **Тип** - общата категория на съдържанието,
- **Подтип** - задава специфичния формат на данните
- **Опционни параметри** на съдържанието-доуточняват подробностите на данните.

Тип	Значение	Подтип
text	текст	plain: не включва форматираща информация
audio	звук	basic
image	статично изображение	формат на изображението, например, jpg, gif, tiff
video	анимирано изображение	формат на изображението, например, mpeg, quicktime
application	данни за приложение	octet-stream, postscript
message	данните представляват пакетизирано e-mail съобщение	rfc822, partial, external-body
multipart	съобщение с множество типове данни. Подтипът определя как тези части се интерпретират	mixed, alternative, digest, parallel

MIME прекодира данните

```
Content-Type: text/html; charset=utf-8
From:
      =?utf-8?q?=D0=A1=D0=B8=D0=BD=D1=8F_=D0=B7=D0=BE=D0=BD=D0=B0_=D0=92=D0=B0?=
      =?utf-8?q?=D1=80=D0=BD=D0=B0?= <noreplay@varnaparking.bg>
Subject: =?UTF-8?B?0J7QtNC+0LHRgNC10L1vINC30LDRj9Cy0LvQtdC90LjQtQ==?=
      =?UTF-8?B?INC30LAG0L/QsNGA0LrQuNGA0LDQvdC1?=-
Message-ID: <68f33271-911a-1bd8-d9c7-fdcd9a608313@varnaparking.bg>
Content-Transfer-Encoding: base64
Date: Fri, 24 Jul 2020 09:02:37 +0000 (UTC)
MIME-Version: 1.0
```

- Данните в съобщенията са единствено ASCII символи.
- MIME прекодира данните. Използваният метод за прекодиране се задава чрез полето **Content-Transfer-Encoding**. Възможните стойности са:
 - **7bit** – за 7-битови US-ASCII символи без прекодиране;
 - **8bit** – за 8-битови текстови данни. САМО АКО и подателят и получателят MTA поддържат 8-битово MIME прекодиране;
 - **Quoted-printable** – за ASCII текст, в който само малка част от символите са не-ASCII (например, скандинавските символи в стандарта ISO8859-1). Процесът на прекодиране игнорира ASCII символите и прекодира единствено останалите;
 - **Base64** – Прилага се при двоични данни. Алгоритъмът прекодира множество от 3 октета от данни в четири 6-байтови ASCII символи като въвежда увеличение на обема на съобщението от около 33%.

Примери

- Съдържание в обикновен ASCII текст:

```
Mime-Version:      1.0
Content-type:      text/plain; charset="us-ascii"
...
```

- Съдържание хипертекстов документ, символно множество iso-8859-1 и начин на прекодиране quoted-printable

```
Mime-Version:      1.0
Content-type:      text/html; charset="iso-8859-1"
Content-Transfer-Encoding:  quoted-printable
...
```

- Прикачане на документ в pdf формат и начина на прекодиране – base64

```
Mime-Version:      1.0
Content-type:      application/pdf; name="manual.pdf"
Content-Transfer-Encoding:  base64
...
```

MIME – multipart разширение

- Отделните части се разграничават с последователност от символи, които трябва да се укажат в полето **Content-type**.
- Тези ограничители се разполагат и в началото, и в края на тялото на електронното съобщение.
- В зависимост от необходимия начин на интерпретиране, отделните части могат да бъдат от един от следните подтипове:
 - **mixed** – Всяка част представя самостоятелна информация, като отделните части се разглеждат в определената последователност;
 - **alternative** – Всяка част се явява различно представяне на една и съща информация. Системите трябва да изберат “най-добрия” тип за извеждане на базата на локалните променливи на средата;
 - **parallel** – Всяка част представя самостоятелна информация, но редът на следване не е от значение. Най-честото приложение на този тип е едновременното изобразяване на съдържанието на всичките части от съответния софтуер или хардуер;
 - **digest** – Всяка част е RFC2822 съобщение, използвано от пощенски списък.

Пример

Следващият пример илюстрира съобщение, включващо две части: обикновен текст и приложение. Частите са разграничени чрез низа '*frontier*'. Последната секция се указва с "--".

```
Content-type: multipart/mixed; boundary="frontier"  
MIME-version: 1.0
```

```
--frontier
```

```
Content-type: text/plain
```

```
This is the body of the message.
```

```
--frontier
```

```
Content-type: application/octet-stream
```

```
Content-transfer-encoding: base64
```

```
gajwO4+n2Fy4FV3V7zD9awd7uG8/TITP/vIocxXnnf/5mjgQjcipBUL1b3uyLwAVtBLOP4n  
V  
LdIAhSz1ZnyLAF8na0n7g6OSeej7aqI13NIXCfxDsPsY6NQjSvV77j4hWEjlF/ag1S6ghfj  
u  
FgRr+OX8QZMI1OmR4rUJUS7xgoknalqj3HJvaOpeb3CF1NI9VGZYZ6H6zuQBOWZzNB8glwp  
C  
--frontier--
```

Пример

Return-Path: <hristo@tu-varna.bg>
Received: from mail.tu-varna.bg
by mail.tu-varna.bg (Dovecot) with LMTP id r3N4ILA0GF9YdwAAAbfQXw
; Wed, 22 Jul 2020 15:45:15 +0300
Received: by mail.tu-varna.bg (Postfix, from userid 48)
id 1933820AF1; Wed, 22 Jul 2020 15:45:15 +0300 (EEST)
DKIM-Filter: OpenDKIM Filter v2.11.0 mail.tu-varna.bg 1933820AF1
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/relaxed; d=tu-varna.bg;
s=default; t=1595421915;
bh=dNweJznHfTmsYL5xrAW3Hcc2FP4pSOZAjKq/Gi7DwBs=;
h=To:Subject:Date:From:From;
b=zyH3pIXwcvb7r2bK8PG0e+dw0kBE0vSaEtI8JY5cX3fPFMGJdb5joiXB0SiXd0dFl
5PtzNvSv+761bcYC+Zs1AF1pG9ZTipWIUJZOocGWA4hNvPYBaQtKfD8BsX6MEgPiwi
Yw+ksnjBCxrXpaRCNpwyCJW/yGBjUt5emykjTmoQmyX/I8Nwlh6pHj5eQz60sF4pDu
gJPghDbq47yihAzUpAr9ZhRgpw3u74Bm1OX37GGGNTUggpm29LVaVrfzycDdbfIZTZ
9pmpm/AcnvtnOXLAI1R8bL2AQthKHjJSb0l1L4QP1/dgZogB2078DV8DLZnFhPYfC+i
00nPNahgVLC8A==
To: valeksieva@tu-varna.bg, v.nikolov@tu-varna.bg, nikolov_vn@tu-varna.bg,
g.spasova@tu-varna.bg, gmarinova@tu-varna.bg, zh_viv@tu-varna.bg, ivailo.penev@tu-
varna.bg, lig@tu-varna.bg, angelovmg@tu-varna.bg, mkarova@tu-varna.bg, slava_y@tu-
varna.bg, tganchev@tu-varna.bg, hristo@tu-varna.bg, yulka.petkova@tu-varna.bg,
i.boychev@tu-varna.bg, aydin.mehmed@tu-varna.bg, galina.naydenova@tu-varna.bg,
plamena.edreva@tu-varna.bg, p.genchev@tu-varna.bg, k.vangelova@tu-varna.bg,
prolet_deneva@tu-varna.bg
Subject: Fwd: =?UTF-8?Q?=D0=97=D0=B0=D0=BF=D0=BE=D0=B2=D0=B5=D0=B4=20=33?= =?UTF-8?Q?
=34=33/=31=37=2E=30=37=2E=32=30=32=30?=
X-PHP-Originating-Script: 0:rcube.php
MIME-Version: 1.0
Content-Type: multipart/mixed;
boundary="= b51ddcc636f72d03d57ea0f93580a969"
Date: Wed, 22 Jul 2020 15:45:14 +0300
From: Hristo Valchanov <hristo@tu-varna.bg>
Organization: Technical University of Varna
Message-ID: <a22181f821d3fe622039d1c18a692f1c@tu-varna.bg>
X-Sender: hristo@tu-varna.bg

MIME-Version: 1.0

Content-Type: multipart/mixed;
boundary="= b51ddcc636f72d03d57ea0f93580a969"

Date: Wed, 22 Jul 2020 15:45:14 +0300
From: Hristo Valchanov <hristo@tu-varna.bg>
Organization: Technical University of Varna
Message-ID: <a22181f821d3fe622039d1c18a692f1c@tu-varna.bg>
X-Sender: hristo@tu-varna.bg
User-Agent: Roundcube Webmail/1.0.12

--= b51ddcc636f72d03d57ea0f93580a969

Content-Transfer-Encoding: 8bit

Content-Type: text/plain; charset=UTF-8;
format=flowed

Здравейте колеги,

искам да благодаря на всички за организацията и провеждането на
дипломните защиты.

Associate Professor Hristo Valchanov, Ph.D.

Head of Computer Science and Engineering Department
Technical University of Varna
1 Studentska Str.
9010 Varna
Bulgaria

--= b51ddcc636f72d03d57ea0f93580a969

Content-Transfer-Encoding: base64

Content-Type: application/pdf;

name="=?UTF-8?Q?=D0=B7=D0=B0=D0=BF_=E2=84=96343=2Epdf?="

Content-Disposition: attachment;
filename*=UTF-8' '%D0%B7%D0%B0%D0=BF%20%E2%84%96343.pdf

Маршрутизиране на пощата

user@alabala.com

- След формиране на писмото от MUA и изпращането му към MTA, MTA изгражда списък със сайтовете на получателите и поставя по едно писмо в т.н. плик за всеки сайт-получател.
- За всеки сайт MTA трябва да определи метода за маршрутизация на базата на адреса на получателя.
- Първо се определя хоста, към който трябва да се изпрати съобщението, след което то се доставя директно към този хост.
- За анонсирането на мейл услуга, в локалната DNS система на даден сайт за пощенския сървър трябва да има специален ресурсен запис – **MX** (*Mail Exchanger*).
- Задължително се указва приоритет. Предпочита се този, който е с по-малка стойност на приоритета.

```
alabala.com IN MX 5      mail.alabala.com  
alabala.com IN MX 10     mail2.alabala.com  
mail.alabala.com IN A 10.10.10.10
```
- Ако не е открит MX запис за домейна получател, допуска се MTA да провери дали домейна има асоцииран IP адрес и да се опита да достави пощата директно към съответния хост.

Протокол SMTP

- От 1971г.
- Протоколът SMTP позволява клиентите да доставят електронната поща до съответните пощенски сървъри.
- Сървърите използват също SMTP, за да препратят съобщенията между други пощенски сървъри до окончателното доставяне на писмото в пощенската кутия на получателя.
- Стартиране на SMTP сесия се извършва винаги от страна на MUA, докато транспортният агент очаква получаването на SMTP съобщения на порт 25.

ESMTP (Extended Simple Mail Transport Protocol)

- От ноември 1995
- RFC1869
- транспортира 8-битови ASCII символи (RFC 6152)
- поддържа MIME
- има индикация за максималната големина на съобщението (RFC 1870)
- Има автентикация (RFC 4954)
- Сървъри, които го поддържат:
 - IceWarp
 - Postfix
 - Sendmail
 - HMailServer
 - Exim

Команди на SMTP

- Командите дефинират предаването на писмата или функциите, изисквани от потребителя.
- Някои команди изискват и съответен аргумент.
- Отговорът на дадена команда е трицифрен код.
- Командите са символни низове, завършващи с <CRLF>.
- Ако командата има параметри, те се разделят от името с <SP>.

command arguments <CRLF>

Основни команди на SMTP

- **HELO (HELLO)** - първата команда в сесията. Използва се за идентифициране на SMTP изпращача от SMTP получателя. Аргументът съдържа името на домейна на SMTP изпращача. Отговор **OK** е индикация, че SMTP изпращачът и получателят са в началния стадий на обмен.
- **EHLO (HELLO)** - Алтернативна команда за стартиране на диалог по разширения протокол ESMTP.

Основни команди на SMTP

- **MAIL FROM** -Указва изпращача на писмото. Всеки върнат e-mail ще бъде изпратен към този адрес. Командата инициира мейл транзакция, в която данни се доставят до една или повече пощенски кутии. Аргументът може да съдържа и **reverse-path** - опционен списък от хостове, който се използва като път от източника да върне съобщения при невъзможност за доставка към подателя.
- **RCPT TO (RECIPIENT TO)** - Указва получателя на писмото. Множество получатели могат да се зададат чрез няколко такива команди. Аргументът може да съдържа и **forward-path** – опционен списък от хостове. Когато има списък с хостове, това е път на източника и указва, че мейлът трябва да се релейва към следващ хост от списъка. Ако SMTP-получателят не реализира релейване се изпраща код за грешка 550.

Основни команди на SMTP

- Когато мейлът се релейва, релей хоста трябва да се премахне от началото на **forward-path** и да се постави в началото на **reverse-path**.
- Когато мейлът достигне до крайната дестинация (**forward-path** ще съдържа само пощенската кутия на получателя), SMTP-получателят поставя съобщението в пощенската кутия.

Пример:

FROM:<USERX@HOSTY.ARPA>

TO:<@HOSTA.ARPA, @HOSTB.ARPA:USERC@HOSTD.ARPA>

Мейлът ще бъде релейван към хост В (HOSTB) с аргументи:

FROM:<@HOSTA.ARPA:USERX@HOSTY.ARPA>

TO:<@HOSTB.ARPA:USERC@HOSTD.ARPA>

Основни команди на SMTP

- **DATA** - Задава данните, които ще формират тялото на писмото. Краят на данните се указва със символа ‘.’ на отделен ред, т.е. последователността от символите ‘<CRLF>.<CRLF>’. Релейваните мейли ще имат множество такива редове.
- **QUIT**- Указва завършване на диалога между изпращача и получателя. Получателят трябва да изпрати отговор **OK** и тогава да затвори комуникационния канал. Получателят не трябва да затваря канала докато не получи и отговори на **QUIT** (дори и да има грешка). Подателят не трябва да затваря канала докато не изпрати **QUIT** и получи отговор (дори и да има отговор за грешка от предишна команда). Ако конекцията се затвори неочаквано, получателят трябва да действа сякаш е получил команда **RSET** (спиране на текущите транзакции, без връщане на предишните завършили), а подателят – както ако по време на транзакция е получен код за грешка 4xx.

Основни команди на SMTP

- **RSET (RESET)**- Преустановяване на текущия трансфер на поща. Всички буфери трябва да се изчистят.
- **VRFY (VERIFY)** -Използва се за потвърждение от получателя на наличието на потребител за дадения e-mail адрес. Ако съществува такъв потребител, като отговор се връща пълното име на потребителя и пълното име на пощенската кутия.
- **EXPN (EXPAND)**- Използва се за потвърждение от получателя на указания e-mail списък. Ако съществуват потребители, в последователност от множество отговори се връщат пълното име на потребителя и пълното име на пощенската му кутия.
- **SEND** - Използва се за изпращане на съобщение към терминал на получател вместо към пощенска кутия.
- **SAML (SEND AND MAIL)** - Използва се за изпращане на съобщение към терминал и пощенска кутия на получатели. За всеки потребител данните се изпращат към неговия терминал на съответния хост (ако потребителят е активен) и до неговата пощенска кутия.
- **SOML (SEND OR MAIL)**- Използва се за изпращане на съобщение към терминал или пощенска кутия на получатели. За всеки потребител данните се изпращат към неговия терминал на съответния хост (ако потребителят е активен), в противен случай - към неговата пощенска кутия.

Отговорите на изпратените команди

- 1-та цифра – успех (1,2,3), незавършен (4) или провал (5)
- 2-та цифра – категорията на съобщението за грешка (0- синтактична; 2- проблем с връзката; 5- проблем с мейла)
- 3-та цифра – специфичното съобщение в тази категория

Примери:

250 – изисканото действие е изпълнено

550 – пощенската кутия на получателя не може да бъде открита

452 – препълнена е паметта на пощенския сървър

Получаване на поща

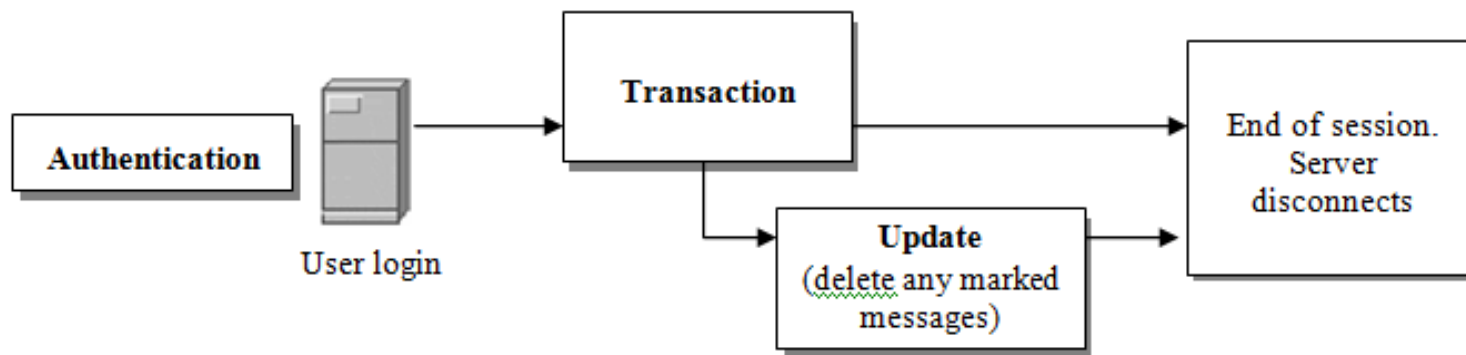
3 базови модела:

- Off-line (POP3, IMAP4) – изтеглят се съобщенията на локалния компютър.
- On-line (IMAP4) - съобщения се манипулират директно на пощенския сървър без необходимостта от тяхното изтегляне на локалния компютър.
- Disconnected use (IMAP4) – свързва се със сървъра, синхронизира съобщенията като изтегля в локалния кеш новите и приключва връзката.

Получаване на поща – POP3

- RFC1939- POP3(*Post Office Protocol*)
- Използва се от пощенски клиенти за получаването на електронната поща от пощенските кутии на потребителя.
- Той е текстово базиран.
- Отговорите от сървъра съдържат обозначение за успешно изпълнение на командата (**+OK**) или за грешка (**-ERR**).
- Клиентите се свързват към пощенските сървъри по порт 110.

Основни фази на обмен



- **Authentication** – клиентът се автентичира с потребителско име и парола преди да може да изтегли своята поща. По време на процеса на автентикация, пощенската кутия на потребителя се блокира (locked) и получените в този период писма не могат да се изтеглят преди приключване на сесията. В даден момент само един потребител може да се свърже към определена пощенска кутия;
- **Transaction** - клиентът изпраща последователност от команди, които сървърът обработва и на които отговаря съобразно протокола;
- **Update** -заклучителна фаза при комуникацията между клиента и сървъра, при която връзката се затваря. След затварянето на връзката, съдържанието на пощенската кутия се обновява за да отрази промените, които клиентът е направил по време на последната сесия.

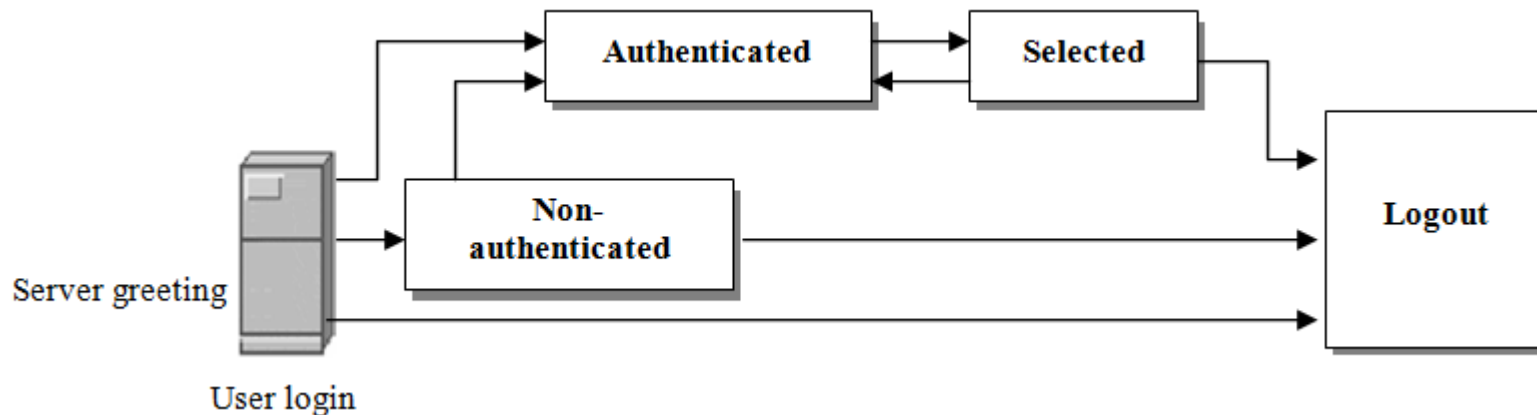
Команди на POP3 клиент

Команда	Значение
USER	Име на потребител
PASS	Паролата на потребителя
STAT	Връща информация за броя и размера на всички съобщения в пощенската кутия
LIST	Връща информация за номера и размера на всяко съобщение в пощенската кутия
RETR msg	Връща съдържанието на указаното с номера си съобщение. Това включва заглавието, празния ред между него и тялото, самото тяло на писмото и завършващия символ ‘.’
DEL msg	Изтрива указаното с номера си съобщение. Актуалното изтриване не се извършва до изпращане на командата QUIT
TOP msg n	Връща заглавията на съобщението с указан номер msg плюс n на брой редове от съдържанието му. Ако n = 0, връща се само заглавието. Това е удобен начин за филтриране на писмата по техните заглавия без необходимостта от изтегляне на самото им съдържание.
QUIT	Затваря сесията и изтрива всички съобщения, маркирани с DEL

Получаване на поща – IMAP4

- RFC2060- IMAP4 (*Internet Message Access Protocol*)
- Слуша на порт 143
- Работи с текстови команди, изпращани от клиента и връщане на данни и статуса на изпълнението им от сървъра
- Характеристики:
 - ефективно управление на папки - създаване, преименоване и изтриване
 - разширено търсене на съобщения по техния размер, заглавни част и податели.
 - отделяне на прикачените файлове от заглавната част и текста на съобщението – полезно при MIME съобщения с много части.
 - Паролата на потребителя за автентикация не се изпраща в чист текст. Същевременно, сървърът периодично изпраща проверяващи съобщения (*challenges*) за удостоверяване идентичността на свързания потребител.

Основни фази на обмен



- **Non-authenticated** - При стартиране на връзката клиентът трябва да предаде своите креденциали (име, парола) преди повечето команди да бъдат разрешени. Ако конекцията е била предварително автентизирана, директно се преминава към следващото състояние;
- **Authenticated** - След като клиентът се е автентизирал, сесията преминава в това състояние. Клиентът първо трябва да избере пощенска кутия за достъп, преди командите за обработка на съобщенията да бъдат разрешени.
- **Selected** - Сесията преминава в това състояние когато успешно е избрана пощенска кутия;
- **Logout** - Конекцията се прекратява от сървъра след съответна заявка от страна на клиента или едностранно от сървъра.

Команди на IMAP4 клиент

- Всяка изпратена команда съдържа префикс с идентификатор, наречен *tag* (*tag*).
- Клиентът генерира различен таг за всяка команда, например A0001, A0002 и т.н.
- Връщаните данни от сървъра към клиента и отговорите за статус при неизпълнени команди се маркират със символа '*’.
- Сървърът може да върне три типа отговори:
 - **OK** (успешно изпълнение на команда),
 - **NO** (индикация за грешка),
 - **BAD** (индикация за грешка в протокола).
- За всяко състояние командите са различни.

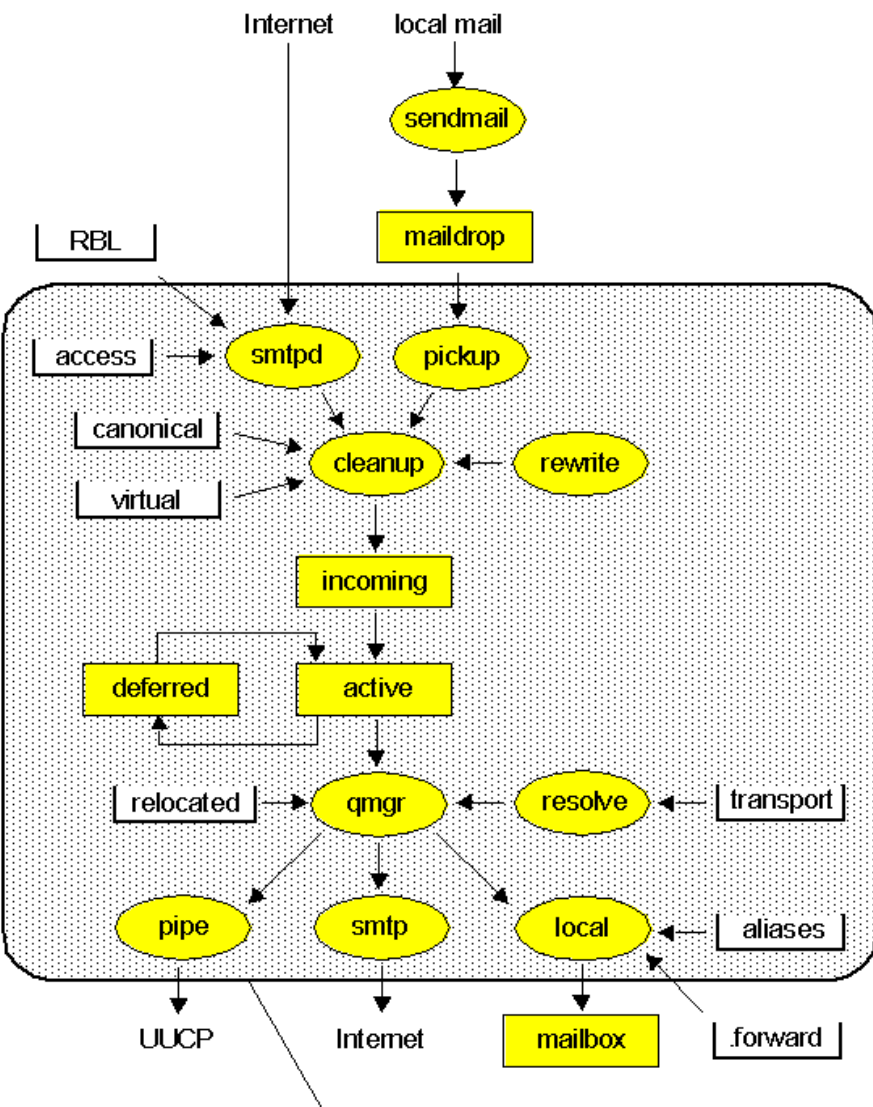
Postfix mail сървър

- Sendmail е една програма с монолитна структура. Служи за обслужване на пощенските съобщения.
- Postfix е базиран на полу-резидентни, съвместно взаимодействащи процеси (демони), всеки от които изпълнява определени задачи, без взаимна зависимост от типа родител-наследник.
- Разделя множеството функции на отделни програми (демони).
- Първо се стартира т.н. **master** демон, който извиква други демони при необходимост.
- Master демонът използва конфигурация от */etc/postfix/master.cf* и */etc/postfix/main.cf* файловете.

Компоненти на Postfix-опашки

Опашки:

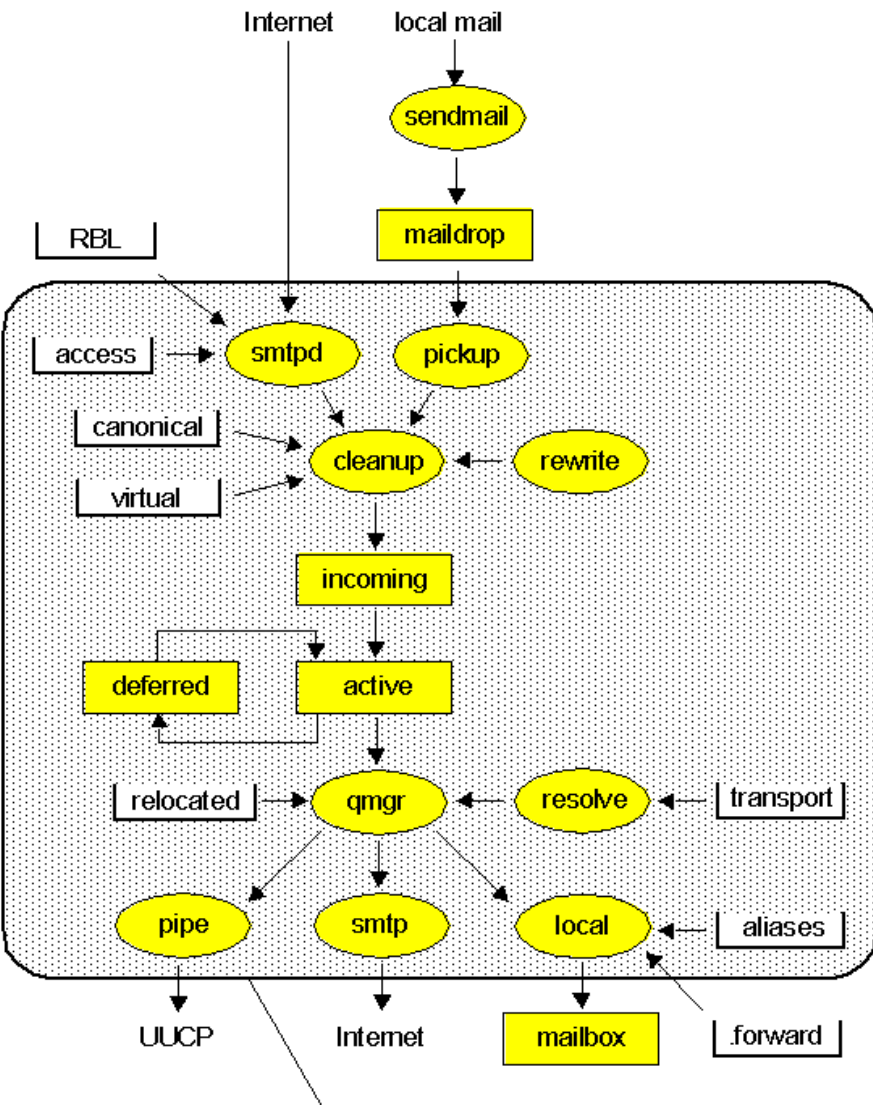
- **Maildrop**- за локално изпращаните мейли, копират в **incoming** след евентуална обработка.
- **Incoming** - мейли, които са току що получени или все още не са погледнати от queue manager (QM)
- **Active** - е с ограничен размер, и е за мейли, които QM е отворил за доставка. При наличие на място в опашката, QM вмъква едно съобщение от **incoming** и **deferred** опашките.
- **Deferred**- мейлите, които не могат да се доставят.



Компоненти на Postfix-демони

Демони за входящи съобщения:

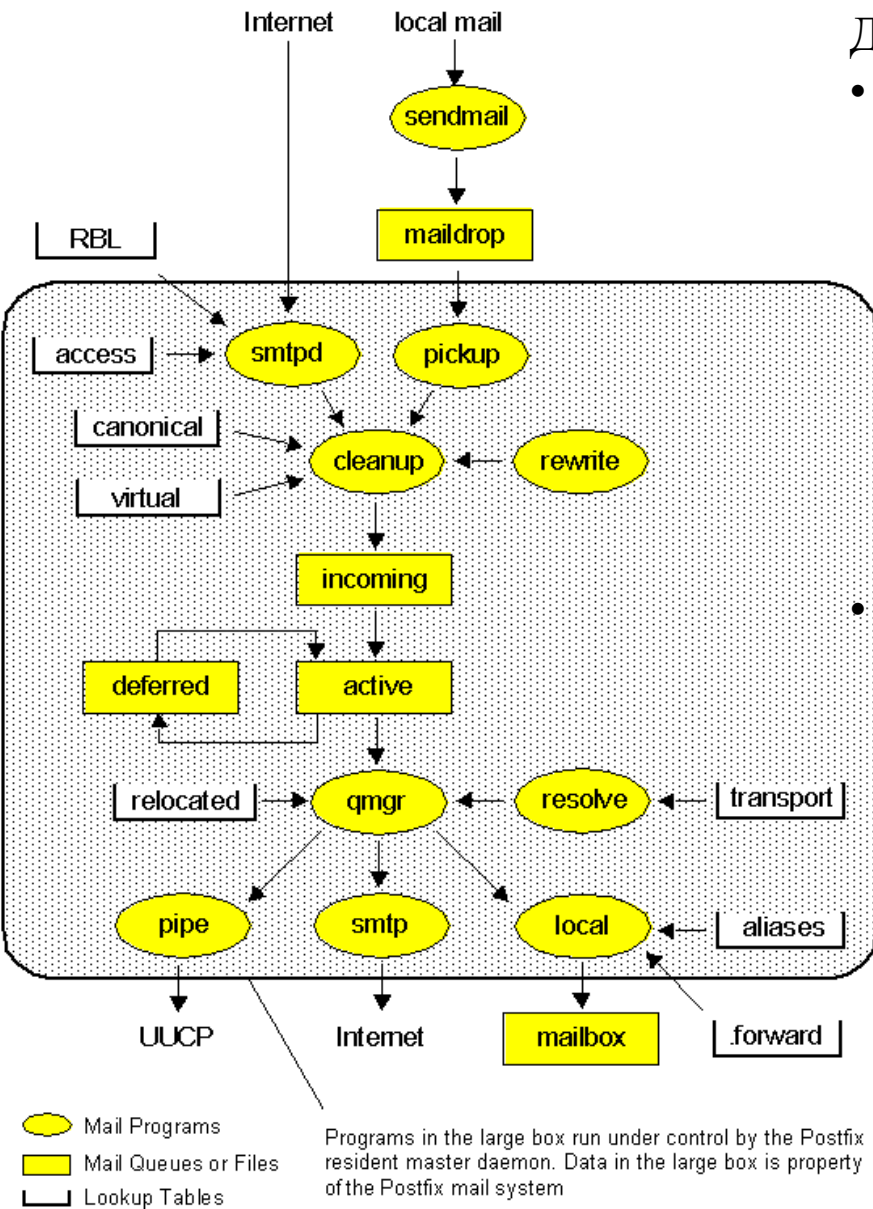
- **Sendmail**- интерфейс към програмата sendmail
- **Pickup**- очаква съобщения в **maildrop** и ги подава на демона **cleanup**. Стартира се от главния процес **master** с *superuser* привилегии и не контактува с външния свят.
- **Smtpd**- Приема конекции от мрежата и изпълнява SMTP транзакции за всяка от тях. Всяко получено съобщение се подава на демона **cleanup** през **pipe**, след което се поставя в опашката **incoming**. (**smtpd** може да обслужва конекции, създадени от стандартния вход и да депозира съобщения директно в опашката **maildrop**. В този stand-alone режим, SMTP сървърът може да приема поща даже ако няма функционираща мейл система.)



Компоненти на Postfix - демони

Демони за обработка:

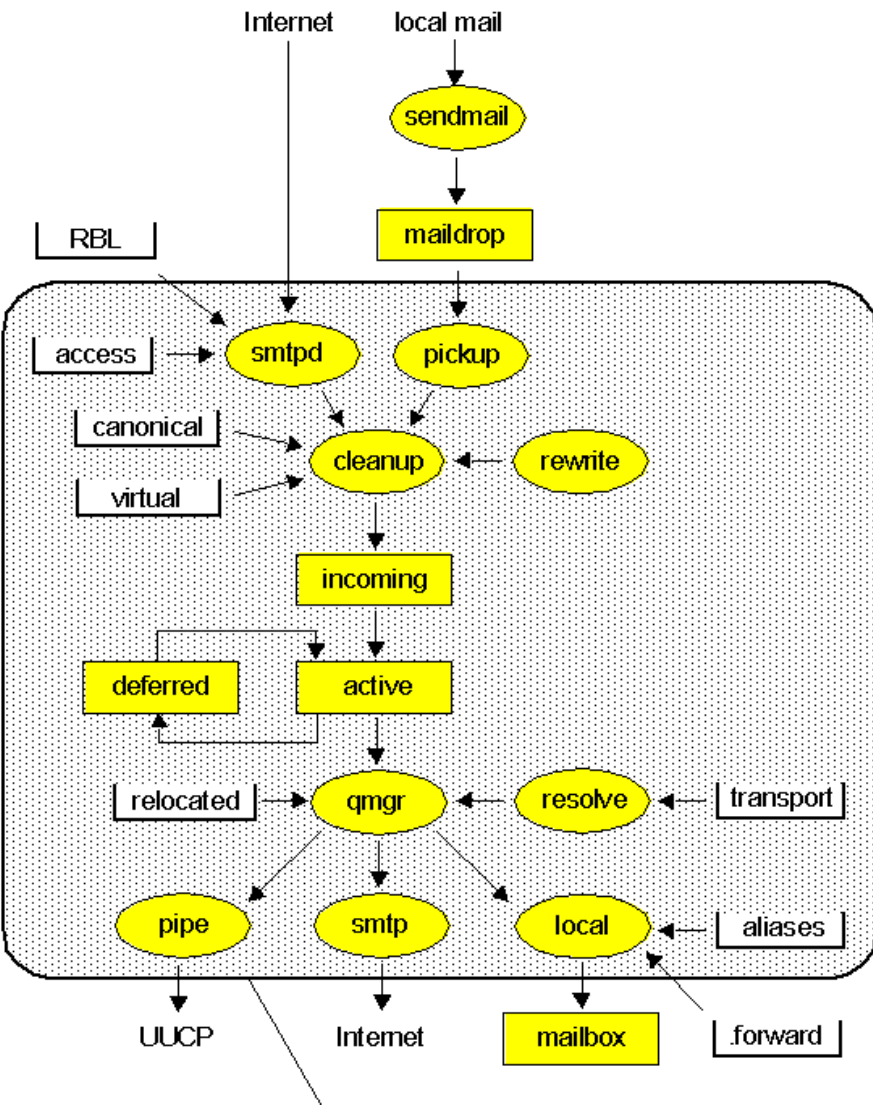
- **Cleanup** - Реализира окончателната обработка за нов мейл. Добавя липсващи хедъри като **From:**, оформя адресите във вида *user@fully.qualified.domain* и опционално извлича адреси от хедърите на мейла. Поставя резултата в опашката **incoming** и уведомява QM за наличието на нов мейл. Трансформирането на адресите се извършва на базата на таблиците **canonical** или **virtual**.
- **Qmgr** - Контактва с агентите за доставка на пощата **local**, **smtp** или **pipe**. Демонът изпраща заявки за доставка с информация за път на файл, адрес на изпращача на мейла, хоста за доставяне ако дестинацията е отдалечена и един или повече адреса на получатели. QM поддържа отделна опашка **deferred** за мейли, които не могат да се доставят. По този начин голям обем от недоставена поща не забавя нормалния достъп до опашките.



Компоненти на Postfix - демони

Демони за резолване и съобщения за грешки:

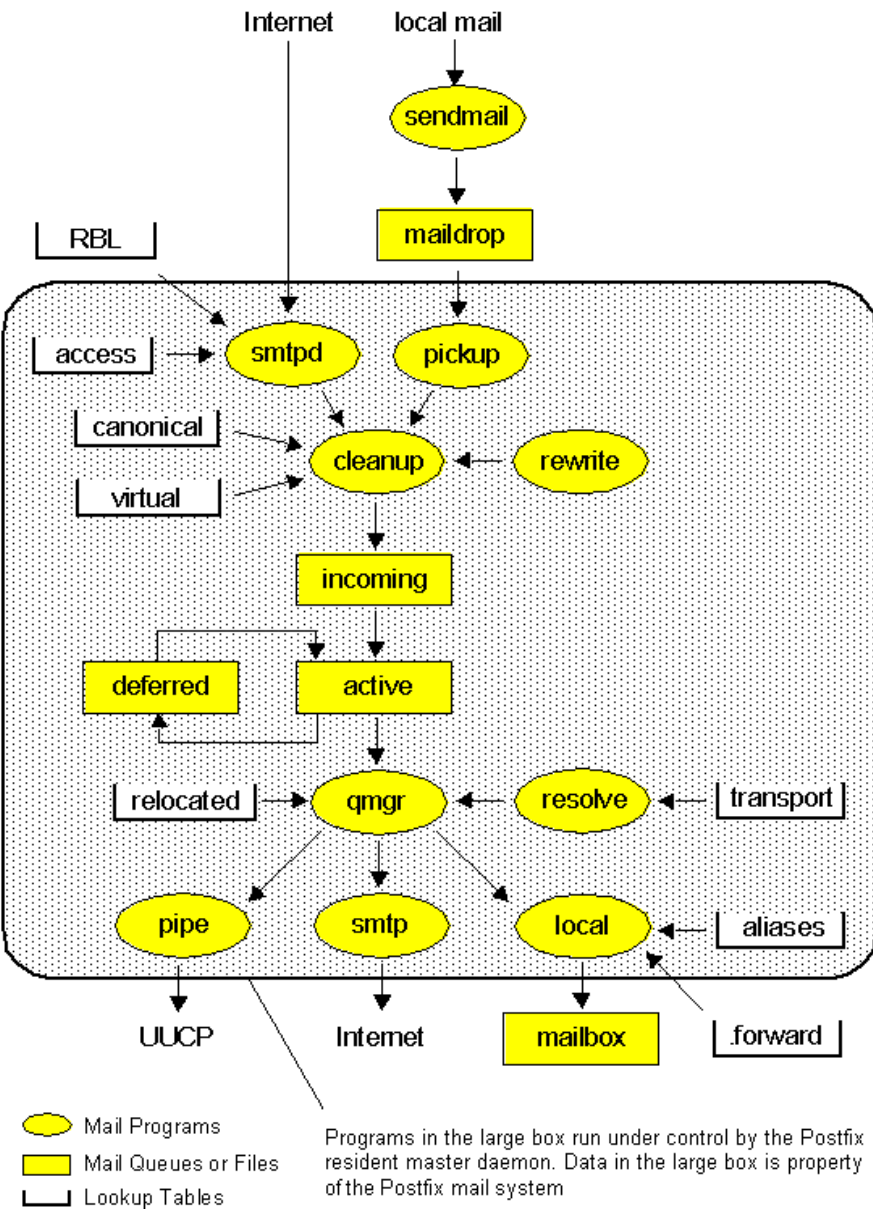
- **Resolve-** Разрешава дестинацията по заявка на QM. По подразбиране се разграничават локалните от отдалечените дестинации. Допълнителна информация за маршрутизиране на пощата може да опише чрез опционната таблица **transport**.
- **Deferred-** По заявка на QM, демонът генерира отчет за недоставяне при невъзможност за доставка на пощата (невъзстановима грешка; недостижима дестинация за дълъг период от време...)



Компоненти на Postfix - демони

Демони за доставка на съобщения:

- **Local** - за локално доставяне.
Поддържа Unix-style mailboxes (файл), sendmail-style бази данни с псевдоними (*alias databases*) и sendmail-style файл **.forward** за потребител.
- **Smtп** - Това е SMTP клиент, който извлича списък с MTA (MX записите от DNS) за дестинацията, сортира го и се опитва да се свърже с всеки от списъка. Ако Postfix системата е натоварена, могат да се видят множество SMTP клиентски процеси, изпълняващи се паралелно.
- **Pipe** - Този демон е изходен интерфейс за други типове транспорт на поща (sendmail входен интерфейс), например като UUCP (Unix to Unix copy).



Конфигуриране на Postfix (1)

```
# ./configure  
# make  
# make install
```

Преди стартиране да се определи дали машината е конфигурирана с FQDN (HOSTNAME). Пробва се с:

```
# hostname
```

Ако не е конфигурирана, то трябва да се установи параметъра **myhostname**:

```
# postconf -e  
myhostname=name.example.com
```

Конфигуриране на Postfix(2)

Да се създаде база данни */etc/aliases.db*

Добре е да се копира базата от */etc/postifx* в */etc* и да се стартира програмата

```
# newaliases
```

Стартиране на сървъра:

```
# postfix start
```

Ако се променят конфигурационните файлове:

```
# postfix reload
```

Спиране на сървъра:

```
# postfix stop
```

Конфигуриране на Postfix(3)

/etc/postfix/main.cf

- Всеки параметър се задава във вида
`<parameter> = <value>`
- Използването на неговата стойност изисква задаване на името му, предхождано от символа „\$“:
`new_param = $myparam`
- Стойностите на параметрите се извеждат с:
`postconf -d` (извежда стойностите по подразбиране)
`postconf -n` (извежда тези, които не са по подразбиране)

Параметри за конфигуриране

- **queue_directory** – задава местоположението на опашката със съобщения на сървъра
- **daemon_directory** – задава местоположението на програмите-демони на сървъра;
- **mail_owner** – задава собственика на опашката и повечето от процесите. Този потребител (по принцип *postfix*) трябва да бъде създаден на машината, където е инсталиран сървърът. Този потребител (и неговата група) трябва да се използва само за това и не трябва да притежава никакви файлове или директории;
- **myhostname** – указва FQDN на машината, където е инсталиран сървърът;
- **mydomain** – указва локалното Интернет име на домейн;
- **myorigin** – указва домейна, от който локалната поща се получава;
- **inet_interfaces** – дефинира адресите на мрежовите интерфейси, по които сървърът ще очаква мейл съобщения;
- **mydestination** – указва списък от домейни, за които машината счита, че е крайна дестинация;
- **unknown_local_recipient_reject_code** – задава кода на отговора на сървърът ако се получи мейл, на който частта от адреса след „@“ съвпада със зададеното в **mydestination**, но името на получателя не е открито;
- **mynetworks** – задава списък от мрежови адреси, които се третираат като „доверени“ SMTP клиенти. За тези клиенти сървърът ще препредава техните мейл съобщения към всяка дестинация;
- **relay_domains** – задава разрешените отдалечени дестинации, за които ще се препредават мейлите, получени от клиенти, извън доверените мрежи.

Пример за базова конфигурация

```
queue_directory = /var/spool/postfix
daemon_directory = /var/libexec/postfix
mail_owner = postfix
myhostname = mail.alabala.com
mydomain = alabala.com
myorigin = $myhostname
inet_interfaces = all
mydestination = $myhostname, localhost.$mydomain
unknown_local_recipient_reject_code = 550
mynetworks = 200.100.50.32/27, 127.0.0.0/8
relay_domains =
```

Използване на псевдоними

- Разрешава пощата да бъде насочена към пощенски кутии, които са алтернативни имена на потребители на машината на местоназначението.
- Псевдонимите се съхраняват във файла */etc/aliases*. Postfix се допитва до този файл, когато определя начина за обработка на входящо пощенско съобщение. Ако се открие елемент в този файл, съвпадащ с целевия потребител в пощенското съобщение, сървърът пренасочва съобщението там, където елементът указва.

- Псевдонимите се задават във вида:

alias: recipient [, recipient]

- **alias** - задава псевдонима, към който е адресирана пощата,
- **recipient** - е потребителско име, име на друг псевдоним или пълен е-мейл адрес, име на машина, списък от множество получатели.

База данни на псевдонимите

- */etc/aliases.db* - вътрешна база от данни с определен формат
- След промяна на файла *aliases* е необходимо да се укаже повторното ѝ изграждане, след което да се рестартира сървъра:

```
# newaliases  
# postfix reload
```


Пример

```
# /etc/postfix/aliases
#
postmaster: root      # задължителни псевдоними
#
#
happy: Bill          # псевдоним за един потребител
spider: Tom, John    # псевдоним за списък от
    потребители
#
master: theboy@gmail.com # препраща пощата
    към друг сървър
#
staff:    /var/log/staff # получавана поща се
    записва във файл
```

Въпроси ?

Благодаря за вниманието !