

(۱) مقدمه

پس از آشنایی مقدماتی با مبانی برنامه نویسی شبکه و مفهوم برنامه های سرویس دهنده / مشتری ، باید سرویس دهنده های استاندارد و مشهوری را که کاربرد فراگیر و جهانی دارند ، معرفی کرده و از دیدگاه مهندسی اینترنت به تحلیل پروتکل های حاکم بر آنها پردازیم. یکی از کاربردی ترین و عمومی ترین کاربردهای شبکه جهانی اینترنت سیستم پست الکترونیکی است که حدود سه دهه قدمت دارد و در خلال این سالها متحول شده و پیشرفتهای چشمگیری داشته است . در این فصل سیستم پست الکترونیکی و پروتکل های مرتبط را معرفی خواهیم کرد.

کلاً سیستم پست الکترونیکی در دو برنامه مجزا سازماندهی می شود:

- **کارگزار کاربر^۱** : امکان خواندن ، نوشتن و ارسال و دریافت نامه را برای کاربر فراهم می کند .
- **کارگزار انتقال پیام^۲** : انتقال نامه ها را از مبدا به مقصد برعهده دارد.

کارگزار کاربر یک برنامه محلی است که محیطی را برای نوشتن ، خواندن ، ویرایش و نهایتاً تقاضای ارسال یا دریافت نامه فراهم می نماید. این برنامه ها به چند صورت عرضه شده اند :

الف) برنامه های مبتنی بر خط فرمان^۳ : در این محیطها کاربر موظف است با دستوراتی که در خط فرمان اجرا میشوند اقدام به ارسال یا دریافت نامه هایش نماید. مثلاً کاربر در محیط یونیکس دستور mail را در خط فرمان صادر کرده و پیامی را پس از نوشتن ارسال می نماید. استفاده از این روش بعلت عدم راحتی کاربران منسوخ شده است.

ب) برنامه های مبتنی بر منوها^۴ : در این برنامه ها کاربران برای انجام عملیات خود بجای استفاده از فرامین ، از منوهای چند گزینه ای استفاده می کنند . هر چند این محیطها بسیار ساده تر هستند ولی کمتر مورد استفاده قرار می گیرند.

ج) برنامه های گرافیکی : در این گونه برنامه ها کاربر تمام فرامین و عملیات مورد نیاز را با استفاده از موس و فشاردادن روی گزینه ها و شکلک ها^۵ انجام داده و

^۱ User Agent

^۲ Message Transfer Agent

^۳ Command Based

^۴ Menu Based

^۵ Icons

محیط برنامه صد درصد گرافیکی و صفحه آرایی شده است لذا استفاده از آنها بسیار ساده و راحت است و امروزه بصورت فراگیر از آنها استفاده می شود. (برنامه Outlook Express یا Eudora از این دسته هستند)

کلاً هر سیستم پست الکترونیکی حداقل باید امکانات پنج گانه زیر را فراهم نماید:

- **امکان ایجاد و نگارش نامه**^۱: این عملیات به منظور کمک به کاربران برای نوشتن و ویرایش نامه هایشان طراحی میشود و به آنها کمک می کند تا آدرسها و فیلدهای لازم را تنظیم نمایند .
- **امکان انتقال**^۲: به پروتکل یا مجموعه عملیاتی که ارسال یک نامه را از مبدا به مقصد تضمین کند اطلاق میشود. دقت کنید که اساس این عملیات بر خدمات لایه سوم از شبکه اینترنت بنیان نهاده شده است؛ این عملیات شامل برقراری یک ارتباط TCP، ارسال اطلاعات، تأیید دریافت آن از طرف مقابل و سپس ختم ارتباط می باشد. این وظائف به صورت خودکار انجام می شود و کاربر هیچ اطلاعی از چگونگی آن ندارد.
- **امکان گزارش گیری**^۳: به مجموعه عملیاتی اطلاق می شود که به شخص ارسال کننده نامه اطلاعاتی را در مورد سرنوشت نامه اش می دهد؛ آیا نامه پذیرفته شد یا آنکه بدلائلی از دست رفت؛ همچنین دلائل هر مشکل احتمالی گزارش می شود.
- **امکان نمایش**^۴: به مجموعه عملیاتی اطلاق می شود که به کاربر این امکان را می دهد که نامه هایش را بخواند و اگر ضمائم مانند صدا و تصویر به همراه آن است به نحوی ملاحظه و بررسی شود.
- **امکان تصمیم گیری**^۵: به مجموعه عملیاتی گفته می شود که به کاربر این امکان را می دهد تا در مورد نامه هایش تصمیم بگیرد؛ نامه ای قبل از خواندن بصورت خودکار حذف شود، نامه ای بعد از خواندن بصورت خودکار حذف شود؛ برخی از نامه ها هیچگاه حذف نشوند و برخی از نامه ها پس از دریافت بطور خودکار پاسخ داده شوند.

^۱ Composition
^۲ Transfer
^۳ Reporting
^۴ Displaying
^۵ Dispositoin

امکانات پنج گانه‌ای که به آنها اشاره شد حداقل امکانات یک سیستم پست الکترونیکی هستند و امروزه امکانات سطح بالاتری از یک سیستم پست الکترونیکی انتظار می‌رود. بعنوان مثال یک شرکت می‌خواهد لیستی از مشتریان و همچنین لیستی از تولیدکنندگان را تهیه کرده و آنها را در یک فهرست پستی^۱ سازماندهی کند تا هر گاه نامه‌ای را به یکی از این فهرستها ارسال کرد سیستم پست الکترونیکی بطور خودکار یک نسخه از آن نامه را برای تمام اعضای آن فهرست ارسال نماید.

از امکانات مورد انتظار دیگر می‌توان به موارد زیر اشاره کرد:

- ارسال رونوشت یک نامه به دیگران که در نامه‌های الکترونیکی^۲ Cc نامیده می‌شود. یعنی فرستنده یک نامه به غیر از آدرس فرد گیرنده نامه، آدرس اشخاص دیگری را که باید نسخه‌ای از آن نامه را دریافت کنند، درج می‌کند و سیستم پست الکترونیکی بصورت خودکار نسخه‌ای از آن نامه را برای گیرندگان رونوشت ارسال می‌کند.

- ارسال رونوشت یک نامه به دیگران بدون اطلاع دریافت کنندگان آن از آدرس سایر گیرندگان که اختصاراً^۳ Bcc نامیده می‌شود.

- پست الکترونیکی رمز شده برای ارسال نامه‌های محرمانه و سرّی

نکته مهمی که در مورد یک نامه الکترونیکی بایستی بدانید آنست که نامه معمولی بصورت متنی نوشته می‌شود و اطلاعاتی که سرویس دهنده پست الکترونیکی برای ارسال لازم دارد آنها بصورت متنی به ابتدای نامه اضافه خواهد شد. بنابراین یک سیستم پست الکترونیکی باید بتواند بین مجموعه‌ای از رشته‌های متنی که بعنوان نامه دریافت می‌کند مشخصات سرآیند نامه را که شامل اطلاعات بسیار مهمی برای رساندن آن به مقصد است از بدنه نامه که آنها متنی است تمیز بدهد.

بنابراین یک نامه الکترونیکی شامل دو قسمت است :

الف : اطلاعات سرآیند^۴ که شامل اطلاعاتی نظیر آدرس گیرنده، آدرس فرستنده، موضوع و... است.

ب : قسمت پیام که بدنه نامه نامیده می‌شود.

^۱ Mailing list
^۲ Carbon copy
^۳ Blind Carbon copy
^۴ Header

قبل از هر کاری ابتدا استاندارد مربوط به قالب یک نامه الکترونیکی را بررسی می‌کنیم .

دقت کنید هر گونه خطایی در تنظیم قالب یک نامه الکترونیکی ممکن است سرنوشت یک نامه را در رسیدن به مقصد تغییر بدهد، بنابراین شناخت کامل از قالب یک نامه الکترونیکی بسیار مهم است. در ادامه استاندارد قدیمی پست الکترونیکی در مورد قالب نامه‌ها را بررسی کرده و پس از آن به روشهای جدیدتر خواهیم پرداخت.

۲) استاندارد RFC 822 : تبیین قالب یک نامه ساده الکترونیکی

در این استاندارد یک نامه الکترونیکی بصورت زیر سازماندهی می‌شود:

- تعدادی فیلد مشخص و تعریف شده که برای عملیات انتقال نامه لازم است. این قسمت سرآیند نامه را تشکیل می‌دهد. (این فیلدها متنی هستند)
- یک سطر خالی (به عنوان مرز قسمت سرآیند و بدنه نامه)
- بدنه پیام (شامل متن اصلی نامه)

فیلدهائی که در سرآیند نامه قرار می‌گیرند (و بعضی الزامی و برخی دیگر اختیاری هستند) در جدول (۱-۹) آمده است . دقت کنید که ترتیب کوچک و بزرگ بودن حروف هر فیلد برای سرویس دهنده های پست الکترونیکی مهم است و باید دقیقاً بگونه‌ای باشد که در جدول درج شده است؛ هر گونه تغییر در آن ممکن است منجر به بروز خطا شود.

فیلد	شرح
To:	آدرس پست الکترونیکی گیرنده اصلی نامه
Cc:	آدرس پست الکترونیکی گیرنده یا گیرندگان ثانویه
Bcc:	آدرس پست الکترونیکی گیرنده یا گیرندگان ثانویه بدون اطلاع از آدرس یکدیگر
From:	آدرس پست الکترونیکی صاحب اصلی (نویسنده) نامه
Sender:	آدرس پست الکترونیکی فرستنده اصلی نامه
Received:	خطی که توسط سیستمهای پست الکترونیکی در بین مسیر اضافه می‌شود.
Return-path:	مسیر برگشت نامه را تعریف می‌کند.

جدول (۱-۹) فیلدهای سرآیند در استاندارد RFC 822

فیلدهای جدول (۱-۹) را جداگانه توضیح می‌دهیم. ذکر این نکته ضروری است که هر فیلد باید در یک سطر مجزا قرار بگیرد و نباید بین آنها سطر خالی وجود داشته باشد.

- فیلد **“To:”**: در جلوی این فیلد (البته پس از یک فاصله خالی) آدرس شخص گیرنده نامه قرار می‌گیرد؛ مثلاً:

To: erlina@cs.uto.edu

همانگونه که در مبحث سیستم DNS اشاره شد، قسمت سمت راست آدرس پست الکترونیکی (یعنی پس از علامت @) نام ماشین سرور دهنده پست الکترونیکی است که باید به آدرس IP ترجمه شود و سمت چپ آن نام فردی قرار می‌گیرد که به عنوان یک مشترک بر روی آن ماشین تعریف شده و باید نامه را دریافت نماید.

- فیلد **“Cc:”**: در جلوی این فیلد آدرس پست الکترونیکی شخص دیگری درج می‌شود که باید رونوشتی از این نامه را دریافت نماید.

- فیلد **“Bcc:”**: این فیلد دقیقاً همانند قبلی است با این تفاوت که گیرندگان نامه از این موضوع که شخص دیگری این نامه را دریافت کرده مطلع نخواهند شد چرا که در مقصد محتوای این فیلد نشان داده نمی‌شود. دقت کنید که هیچ تفاوتی بین نامه‌ای که هر یک از گیرندگان دریافت می‌کنند وجود نخواهد داشت یعنی اگر جای آدرسها در فیلد **To:** و فیلد **Bcc:** (یا **Cc:**) عوض شوند اتفاق خاصی نخواهد افتاد.

- فیلد **“From:”**: در جلوی این فیلد آدرس پست الکترونیکی نویسنده و صاحب اصلی نامه درج می‌شود.

- فیلد **“Sender:”**: جلوی این فیلد آدرس پست الکترونیکی کسی که حقیقتاً نامه را ارسال کرده است قرار می‌گیرد. بعنوان مثال فرض کنید رئیس یک شرکت به منشی خود دستور می‌دهد نامه‌ای را برای یک موسسه نوشته و ارسال نماید. بنابراین باید در فیلد **From:** آدرس رئیس شرکت و در جلوی فیلد **Sender:** آدرس پست الکترونیکی منشی شرکت قرار بگیرد. اگر نویسنده اصلی نامه و ارسال کننده آن هر دو یکی باشند نیازی به فیلد **Sender:** نخواهد بود.

- فیلد **“Received:”**: گاهی بین سیستم پست الکترونیکی فرستنده نامه و سیستم گیرنده نامه، سرور دهنده‌های دیگری بعنوان واسطه‌های انتقال^۱ وجود دارد. این

^۱ Transfer Agent

نمایندگیها پس از دریافت نامه، سطری حاوی فیلد: **Received**، هویت و آدرس خود، ساعت و تاریخ دریافت پیام و هرگونه اطلاعاتی که می‌تواند برای ردیابی مسیر مفید باشد به آن اضافه می‌نمایند. برای مثال فرض کنید یک دانشگاه دارای سیستم پست الکترونیکی است که تمام نامه‌ها را دریافت و بین سرویس دهنده‌های پست الکترونیکی در هر دانشکده توزیع می‌کند. سرویس دهنده کل دانشگاه که با دنیای خارج در ارتباط است بعنوان واسطه انتقال شناخته می‌شود.

• فیلد **“Return-path”**: آخرین واسطه انتقال این فیلد را به نامه اضافه می‌نماید و مشخص می‌کند که پاسخ نامه چگونه باید به فرستنده آن برگردد. از لحاظ تئوری این اطلاعات را می‌توان از فیلد **Received** استخراج کرد.

به غیر از فیلدهائی که توضیح داده شد فیلدهای اضافی دیگری هم وجود دارد که توسط سیستم انتقال استفاده و پردازش نمی‌شود بلکه فقط برای اطلاع شخص خواننده نامه یا برنامه‌ای که کاربر برای خواندن نامه‌اش از آن استفاده می‌کند مفید خواهد بود و وجود و عدم وجود آنها مهم نیست. این فیلدها در جدول (۲-۹) معرفی شده‌اند که در ادامه آنها را معرفی می‌کنیم.

فیلد سرآیند	شرح
Date:	تاریخ و زمان ارسال پیام (نامه)
Reply-To:	آدرس پست الکترونیکی کسی که باید پاسخ این نامه را دریافت نماید.
Message-Id:	یک شماره منحصر بفرد برای آنکه بتوان بعداً به آن شماره استناد کرد.
In-Reply-To:	شماره نامه‌ای که این نامه در پاسخ به آن نامه می‌باشد.
References:	شماره‌های دیگری که این نامه با آنها مرتبط است.
Keywords	برخی از کلمات کلیدی از مضمون نامه که توسط نویسنده نامه انتخاب می‌شود.
Subjects	موضوع نامه (خلاصه بسیار کوتاهی از نامه فقط در یک خط)

جدول (۲-۹) فیلدهای اختیاری در استاندارد RFC 822

- فیلد **“Date:”**: در جلوی این فیلد تاریخ و زمان ارسال نامه قرار می‌گیرد. معمولاً تاریخ و زمان بر حسب ساعت گرینویچ^۱ (GMT) درج می‌شود.
- فیلد **“Reply-To:”**: این فیلد زمانی بکار می‌رود که نویسنده اصلی نامه تمایلی به دریافت پاسخ آن نامه نداشته باشد بلکه بخواهد پاسخ نامه ارسالی توسط شخص ثالثی دریافت شود. بعنوان مثال مدیر بازاریابی یک شرکت برای مشتریان خود نامه‌ای را تنظیم کرده و منشی او آنرا ارسال می‌نماید. پاسخ نامه‌های ارسالی باید به مدیر فروش شرکت ارجاع داده شود بنابراین در جلوی فیلد **Reply-To:** آدرس مدیر فروش درج می‌شود.
- فیلد **“Message-Id:”**: در جلوی این فیلد یک شماره منحصر به فرد و اختیاری قرار می‌گیرد که بتوان برای نامه‌های پیرو که بعداً ارسال می‌شود به آن استناد کرد. (دقیقاً مثل شماره نامه‌های اداری)
- فیلد **“In-Reply-To:”** در جلوی این فیلد شماره نامه‌ای قرار می‌گیرد که نامه فعلی در پاسخ به آن ارسال شده است.
- فیلد **“References:”**: در جلوی این فیلد شماره نامه‌های دیگری قرار می‌گیرد که نامه فعلی به موضوع آن نامه‌ها مرتبط است.
- فیلد **“Keywords:”**: برخی از کلمات کلیدی که با متن و موضوع نامه مرتبط است و برنامه نامه‌خوان می‌تواند آنها را ملاک دسته‌بندی یا جستجو قرار بدهد.
- فیلد **“Subject:”**: یک کلمه یا جمله کوتاه که مضمون نامه را برای خواننده آن مشخص می‌نماید.

اگر چه فیلدهای فوق اختیاری هستند ولی اگر کمی به آنها دقت کنید متوجه خواهید شد که در یک شرکت که ممکن است روزی صدها نامه الکترونیکی دریافت کند این فیلدها چقدر می‌توانند مفید واقع شوند و یک نرم‌افزار نامه‌خوان برای دسته‌بندی، جستجو، تصمیم‌گیری برای سرنوشت نامه‌ها یا بایگانی آنها، چگونه می‌تواند از آنها استفاده کند.

در استاندارد RFC 822 شخص کاربر می‌تواند فیلدهایی را برای استفاده خودش یا نرم‌افزاری که برای خواندن نامه‌هایش استفاده می‌کند تعریف نماید. تمام این فیلدها که همانند بقیه فیلدها در قسمت سرآیند نامه درج می‌شوند بایستی حتماً با دو حرف **X-** شروع شوند. این

^۱Greenwich Mean Time

فیلدها فقط برای یک نرم‌افزار خاص یا شخص کاربر مفید خواهد بود و سرویس دهنده‌های پستی که وظیفه انتقال و ذخیره نامه‌ها را بر عهده دارند آنها را نادیده می‌گیرند. مثلاً فرض کنید یک موسسه خوش ذوق بخواد در یک فیلد از هر نامه‌ای که ارسال می‌کند (بطور خودکار و توسط نرم‌افزار) یک بیت شعر یا ضرب‌المثل قرار بدهد. در این حالت در قسمت سرآیند نامه، فیلد زیر قابل تعریف است:

X-Proverb: ضرب‌المثل

(البته باید نرم‌افزار نامه‌خوان شخص گیرنده بتواند این فیلد را تمیز داده و نشان بدهد)

یادآوری این نکته بسیار مهم خواهد بود که هر فیلد و رشته‌ای که جلوی آن نوشته می‌شود باید در یک سطر مجزا قرار بگیرد و بین سطرهای سرآیند سطر خالی نباشد چرا که سطر خالی به منزله خاتمه قسمت سرآیند نامه و شروع متن (بدنه) نامه تلقی می‌شود و نرم‌افزار نامه‌خوان درون بدنه نامه را پردازش نخواهد کرد.

استاندارد RFC 822 تقریباً کامل و جامع است ولی فقط زمانی بکار می‌آید که هدف ما ارسال نامه‌های هائی باشد که مطلقاً متنی و به زبان انگلیسی هستند. حال فرض کنید بخواهید نامه‌ای جهت تبریک سال نو به زبان فارسی به‌مراه یک فایل تصویر بعنوان کارت پستال برای یک دوست ارسال نمایید یا بخواهید قطعه کوتاهی از فیلم جشن تولد فرزندتان به نامه شما ضمیمه شود. در چنین مواردی استاندارد RFC 822 که مربوط به دو دهه قبل است جوابگو نخواهد بود. استاندارد جدید MIME ضمن پشتیبانی از RFC 822 این نواقص را برطرف کرده است.

۳) استاندارد MIME : سیستم نامه (سانی توسعه یافته در اینترنت

در این استاندارد ایده اصلی آن بوده است که بدون پشت پا زدن به استاندارد RFC 822 (که در آن زمان بسیار محبوبیت داشت) روشی ابداع شود تا بتوان فایل‌های غیر آسکی همانند فایل‌های اجرایی، صدا و تصویر بگونه‌ای در بدنه نامه قرار گیرد که براساس سرویس‌دهنده‌های قدیمی قابل ارسال و دریافت باشد؛ در این صورت بدون نیاز به تغییر سرویس‌دهنده‌های قبلی، فقط باید برنامه نامه‌خوان در سمت کاربر عوض شود که هزینه کمی را به کاربر تحمیل می‌کند.

استاندارد MIME پنج فیلد جدید در سرآیند نامه تعریف کرده است که این فیلدها در جدول (۳-۹) معرفی شده‌اند.

سرآیند	توضیح
MIME-Version:	شماره نسخه MIME
Content-Description:	یک سطر که مضمون کلی نام را مشخص می‌نماید.
Content-Id:	یک مشخصه یا شماره منحصر به فرد
Content-Transfer-Encoding:	طریقه کدگذاری محتوای نام
Content-Type:	نوع و محتوای نام

جدول (۳-۹) سرآیندهای جدید در استاندارد MIME

- فیلد **"MIME-Version"**: این فیلد به برنامه نامه‌خوان در سمت کاربر تفهیم می‌کند که این نامه الکترونیکی با استاندارد MIME سازماندهی و ارسال شده است؛ در ضمن نسخه استاندارد MIME را نیز مشخص می‌نماید. نامه هائی که این فیلد را در سرآیند نامه نداشته باشند (همانند نامه های قدیمی با استاندارد RFC 822) بصورت نامه‌ای تماماً متنی با کدهای آسکی تلقی می‌شوند.
- فیلد **"Content-Description"**: متنی که در جلوی این فیلد قرار می‌گیرد مضمون و محتوای نامه را مشخص می‌کند. گیرنده نامه با استفاده از این فیلد می‌تواند تشخیص بدهد که آیا رمزگشائی و خواندن پیام ارزشمند است یا نه.
- فیلد **"Content-Id"**: این فیلد که مشابه فیلد Message-Id در استاندارد RFC 822 است شماره یا رشته‌ای است منحصر به فرد که می‌توان به عنوان شماره نامه در نامه های بعدی به آن استناد کرد.
- فیلد **"Content-Transfer-Encoding"**: در جلوی این فیلد عبارتی قرار می‌گیرد که به برنامه نامه‌خوان در سمت کاربر تفهیم می‌کند که چه قاعده‌ای را برای دیکود^۱ کردن بدنه نامه بکار برد. بگونه‌ای که اشاره شد بر خلاف استاندارد RFC 822 در بدنه نامه های مبتنی بر استاندارد MIME می‌تواند کدهای غیر آسکی، فایل‌های صدا، تصویر یا کلاً هر فایل دودویی قرار بگیرد. بنابراین در مقصد قبل از نمایش محتوای نامه، باید قسمت بدنه آن پردازش و دیکود شود. اگر بدون رمزگشائی، نامه را نگاه کنید یکسری کاراکترهای نامفهوم خواهید دید. انواع کدگذاری^۲ در استاندارد MIME به شرح زیر است:

^۱ Decoding
^۲ Encoding

◀ کدهای ASCII ساده که باید کدهای بین صفر تا ۱۲۷ باشند. (یعنی حروف، علائم و کاراکترهای صرفاً انگلیسی) تنها محدودیتی که وجود دارد آنست که هر خط از متن نامه نبایستی از ۱۰۰۰ کاراکتر تجاوز کند.

◀ کدهای ASCII توسعه یافته^۱ که می‌تواند تمام ۲۵۶ کاراکتر جدول آسکی را شامل بشود. نکته‌ای که وجود دارد آنست که چون استاندارد معینی برای کاراکترهای بالای جدول آسکی وجود ندارد لذا ارسال نامه‌هایی که از این کاراکترها استفاده می‌کنند ممکن است در مقصد با مشکل نمایش مواجه شوند. برای روشن شدن قضیه فرض کنید کسی نامه‌ای را بنویسد که در آن از حروف فارسی با کدهای بالای ۱۲۸ استفاده کرده باشد. حال وقتی این نامه ارسال شد ممکن است در مقصد بصورت یک متن کاملاً نامفهوم و مبهم دیده شود چرا که ماشین مقصد کدهای بالای ۱۲۸ در جدول آسکی را به صورت دیگری تفسیر کرده است. در اینجا هم حداکثر طول هر خط ۱۰۰۰ کاراکتر است.

◀ کد گذاری base64: این روش که به آن ASCII Armor هم گفته می‌شود در مواقعی کاربرد دارد که بخواهید یک فایل دودویی (مثل یک فایل اجرایی یا فایل تصویر) را در بدنه نامه جا سازی نمایید. در چنین مواقعی بهترین راه حل ممکن آن است که فایل به نحوی به کاراکترهای ASCII تبدیل شده و درون متن نامه قرار بگیرد. (برای روشن شدن قضیه فرض کنید نامه‌ای نوشته و به آن فایل تصویر ضمیمه می‌کنید؛ این فایل تصویر بصورت کدهای ASCII در متن نامه قرار می‌گیرد و شما می‌توانید آن کدها را توسط یک ویرایشگر ساده مثل Notepad ببیند) روش تبدیل فایل‌های باینری به کدهای ASCII بشرح زیر انجام می‌شود:

از درون فایل دودویی سه بایت سه بایت جدا می‌شود. (سه بایت مجموعاً ۲۴ بیت خواهد شد) این ۲۴ بیت به چهار قسمت شش بیتی تقسیم می‌گردد. هر قسمت شش بیتی مجموعاً ۶۴ حالت دارد (از صفر تا ۶۳) که برای حالت صفر کاراکتر 'A' و برای ۱ کاراکتر 'B' و به همین ترتیب تا ۲۵ که 'Z' قرار می‌گیرد؛ برای ۲۶ تا ۵۱ به ترتیب 'a' تا 'z' و برای ۵۲ تا ۶۱ به ترتیب کاراکترهای '0' تا '9' قرار می‌گیرد؛ برای ۶۲ و ۶۳ به ترتیب کاراکترهای '+' و '/' جایگزین می‌شود.

ممکن است تعداد بایتهای یک فایل دودویی ضربی از ۳ نباشد. بنابراین در دسته آخر ممکن است فقط یک یا دو کاراکتر باقی مانده باشد. در این حالت اگر در گروه

^۱ Extended ASCII

آخر یک کاراکتر باقی مانده باشد پس از تبدیل دو علامت '=' و اگر دو کاراکتر باقی مانده است یک علامت '=' قرار می‌گیرد. طرح یک مثال بسیار روشن‌گر خواهد بود:
فرض کنید بخواهیم پنج بایت زیر را طبق روش فوق به کدهای ASCII Armor تبدیل نمائیم:

0x20	0x12	0x12	0xE1	0xA1	-
00100000	00010010	00010010	11100001	10100001	00xxxxxx

001000	000001	001000	010010	111000	011010	000100	xxxxxx
I	B	I	S	4	a	E	=

با روش فوق هر فایل دودویی به حالت متنی تبدیل می‌شود. در مقصد برنامه نامه‌خوان وقتی گزینه زیر را در متن ببیند به راحتی آنرا به حالت اصلی برخواهد گرداند:

Content-Transfer-Encoding: base64

دقت کنید هر فایل که به روش فوق کد شود فقط شامل حروف کوچک و بزرگ انگلیسی، کاراکترهای '0' تا '9' و علامتهای '+، /' و '=' خواهد بود و در مقصد هر کاراکتر که به غیر از کاراکترهای ذکر شده لابلای آن وجود داشته باشد به سادگی حذف خواهد شد؛ چون در استاندارد MIME هر سطر می‌تواند حداکثر هزار کاراکتر باشد لذا پس از تبدیل یک فایل دودویی به حالت ASCII Armor می‌توان در هر جای متن کد '\n' را اضافه کرد تا هر سطر زیر هزار کاراکتر باشد؛ این کدها در مقصد حذف خواهند شد. این نکته نیز قابل توجه است که طول یک فایل دودویی پس از تبدیل به حالت ASCII Armor حداقل با ضریب $\frac{4}{3}$ افزایش می‌یابد.

◀ کد گذاری quoted-printable: استفاده از روش قبل برای تبدیل فایل‌هایی که تعداد کمی کاراکتر با کد بالای ۱۲۸ دارند راه مناسبی نیست چون طول فایل بیهوده افزایش می‌یابد. در این روش برای کاراکترهایی که کد آنها زیر ۱۲۸ است خود کاراکتر ولی برای آنهایی که کدشان بین ۱۲۸ تا ۲۵۵ است ابتدا علامت '=' و بعد دو کاراکتر معادل کد مبنای ۱۶ آن درج می‌شود. به عنوان مثال چهار بایت زیر به روش بالا کد شده‌اند:

01000110	10100111	11110011	01000001
E	=A7	=F3	A

روش کدگذاری quoted-printable فقط زمانی مفید است که نسبت کاراکترهای بالای ۱۲۸ در متن بسیار کم باشد چرا که هر کاراکتر بالای ۱۲۸، پس از تبدیل با سه کاراکتر جانشین خواهد شد.

◀ کد گذاری تعریف شده توسط کاربر: اگر هیچکدام از روشهای کدگذاری قبل را نپسندیدید می‌توانید خودتان یک سبک کدگذاری ابداع کنید بشرطی که گیرنده نامه شما قادر باشد متن شما را از حالت کدگذاری خارج نماید و بنابراین بایستی نرم‌افزار نامه‌خوان یا نرم‌افزار واسطه‌ای برای این کار طراحی نمایید.

• فیلد "Content-Type": آخرین فیلد سرآیند در استاندارد MIME یکی از کاربردی‌ترین فیلدها خواهد بود و مشخصات محتوای نامه را تشریح خواهد کرد. بعنوان مثال در قسمت سرآیند یک نامه این فیلد می‌تواند بصورت زیر تنظیم شده باشد:

Content-Type: Video/Mpeg

به معنای آنکه محتوای بدنه فایلی است ویدئویی با قالب MPEG و بالطبع نرم‌افزار نامه‌خوان باید قادر باشد ضمن استخراج آن از متن، ابزار نمایش آنرا هم بارگذاری نماید.

حال ببینیم محتوای نامه الکترونیکی چه انواعی را در برمی‌گیرد. انواع محتویات متن یک نامه الکترونیکی با استاندارد MIME در جدول (۴-۹) مشخص شده است.

◀ نوع Text: به معنای آنست که نامه از نوع متنی است. انواع آن عبارتست از

◆ Text/plain: نامه معمولی با کاراکترهای ASCII

◆ Text/Richtext: نامه در قالب یک زبان نشانه گذاری (همانند HTML) ارسال

شده است، بنابراین نرم‌افزار نامه‌خوان بایستی قبل از نمایش نامه، متن را تفسیر نماید. در حقیقت در این نوع نامه در لابلای محتوای آن برجسبهایی برای صفحه آرایی و قالب بندی قرار داده می‌شود.

نوع کلی	نوع دقیق	شرح
Text	Plain	متن ساده معمولی
	Richtext	متن حاوی دستورات قالب بندی
Image	Gif	فایل تصویر با قالب GIF
	Jpeg	فایل تصویر با قالب JPEG
Audio	Basic	فایل صوتی با قالب snd
Video	Mpeg	فایل ویدئویی با قالب MPEG
Application	Octet-stream	دنباله‌ای از بایت‌های تفسیر نشده
	Postscript	متن تنظیم شده در پست اسکریپت
	Rfc822	متن تنظیم شده در استاندارد RFC822
Message	Partial	متن به منظور انتقال تکه تکه شده است.
	External-body	متن پیام باید از شبکه اینترنت بارگذاری شود.
	Mixed	متن دارای چند قسمت است که ترتیب مشخص دارد.
Multipart	Alternative	متن دارای چند قسمت با قالب‌های متفاوت است.
	Parallel	قسمت‌های مختلف متن باید همزمان ملاحظه شود.
	Digest	متن شامل چند قسمت است و هر قسمت از نوع RFC822 است.

جدول (۴-۹) انواع محتویات متن در استاندارد MIME

◀ نوع **Image**: به معنای آن است که درون متن، یک تصویر جاسازی شده است. نوع تصویر می‌تواند یکی از دو قالب زیر باشد:

◆ **Image/Gif**: محتوای بدنه، تصویری با قالب GIF می‌باشد.

◆ **Image/Jpeg**: محتوای بدنه، تصویری با قالب JPEG می‌باشد

◀ نوع **Audio**: متن نامه محتوی فایلی از نوع صداست. (با قالب snd)

◀ نوع **Video**: متن نامه محتوی فایلی از نوع ویدئو با قالب MPEG است.

◀ نوع **Application**: متن نامه محتوی فایلی کاربردی است و خودش به دو نوع دیگر تقسیم میشود:

◆ **Application/Octet-stream**: مجموعه‌ای از بایت‌های متوالی که می‌تواند یک فایل اجرایی یا هر فایل داده باشد.

◆ **Application/postscript**: یک فایل قابل چاپ با قالب پست اسکریپت

◀ نوع **Message**: بدین معناست که درون بدنه یک پیام دیگر جاسازی شده است و بنابراین بایستی برای استخراج بقیه قسمت‌ها عملیات اضافه تری انجام شود انواع آن به شرح ذیل است:

◆ **Message/Rfc822**: درون بدنه نامه یک نامه دیگر با استاندارد RFC 822 جاسازی شده است.

◆ **Message/Partial**: بدین معناست که نامه اصلی چون بزرگ بوده به چندین قسمت شکسته شده است و برنامه نامه‌خوان در سمت کاربر باید آنها را کنار هم قرار داده و بازیابی نماید. (در حقیقت نامه اصلی قطعه قطعه شده و باید هر قطعه شماره‌ای داشته باشد تا بتوان آنها را بازسازی کرد)

◆ **Message/External-body**: با این نوع می‌توان به نرم‌افزار نامه‌خوان تفهیم کرد که باید یک فایل طولانی را با استفاده از پروتکل‌های دیگر مثل FTP از روی شبکه استخراج نماید. با این روش بجای آنکه یک فایل حجیم صدا و تصویر را به نامه ضمیمه کرده و داده‌های آنرا در متن نامه قرار بدهیم می‌توان آدرس URL آنرا به خواننده نامه ارائه کرده تا در صورت تمایل آنرا روی سیستم خود بار نماید. نامه‌خوان باید امکان بارگذاری چنین فایل‌هایی را از روی شبکه داشته باشد.

◀ نوع **Multipart**: این گزینه شاید مهمترین نوع نامه‌های الکترونیکی به شمار بیاید چرا که با این گزینه می‌توان در بدنه نامه تمام انواع متن، صدا، تصویر و فایل‌های اجرائی را جاسازی کرد بگونه‌ای که ابتدا و انتهای هر بخش دقیقاً مشخص بوده و نامه‌خوان بتواند آنها را از هم تفکیک نماید. در انتهای این بخش یک مثال از این نوع را خواهیم داشت. انواع گزینه **Multipart** به شرح ذیل است:

◆ **Multipart/Mixed**: بدین معناست که نامه خودش شامل چندین نامه دیگر با استاندارد RFC 822 است.

◆ **Multipart/Alternative**: بدین معناست که نامه دارای چندین قسمت متفاوت شامل متن، صدا، تصویر یا فایل‌های دودویی می‌باشد. در این نوع نامه مرز هر قسمت از نامه با بقیه قسمت‌ها دقیقاً مشخص می‌شود؛ در مثالی چگونگی آنرا تشریح خواهیم کرد.

◆ **Multipart/Parallel**: این گزینه همانند قبلی است یعنی نامه شامل چندین بخش مختلف است ولی تفاوت عمده آن با گزینه قبلی در آن است که نرم‌افزار

نامه‌خوان موظف است قسمتهای متفاوت را بطور همزمان دیکود و اجرا نماید. مثالی از این نوع نامه آنست که محتوای یک نامه شامل یک قطعه انیمیشن و یک قطعه صدا باشد و از نرم‌افزار نامه‌خوان انتظار داشته باشیم که قطعه صدا همزمان با نمایش انیمیشن به اجرا دربیاید.

◆ Multipart/Digest: این گزینه وقتی مورد استفاده قرار می‌گیرد که پیامها با یکدیگر ترکیب شده و پیام بزرگتری را ایجاد می‌کند بعنوان مثال برخی از گروههای خبری در اینترنت پیامهای دسته‌ای از مشترکین را جمع‌آوری کرده و سپس آنها بعنوان یک نامه Multipart/Digest ارسال می‌نمایند.

```
From: erlinor@abc.com
To: carolyn@xyz.com
MIME-Version: 1.0
Message-Id: <0704760941 AA00747@abc.com>
Content-Type: multipart/alternative; boundary=qwertyuiopasdfghjklzxcvbnm
Subject: Earth orbits sun integral number of times
```

This is preamble. The user agent ignore it. Have a nice time.

```
--qwertyuiopasdfghjklzxcvbnm
```

```
Content-Type: text/richtext
```

Happy birthday to you

Happy birthday dear <bold>Carolyn</bold>

```
--qwertyuiopasdfghjklzxcvbnm
```

```
Content-Type: message/external-body;
```

```
Access-type="anon-ftp";
```

```
Site="bicycle.abc.com";
```

```
Directory="pub";
```

```
Name="birthday.snd";
```

```
Content-Type: audio/basic
```

```
Content-Transfer-Encoding: Base64
```

```
--qwertyuiopasdfghjklzxcvbnm
```

داده‌های صدا در ادامه نامه قرار می‌گیرد که در این مثال حذف شده است.

مثال (۵-۹) قالب یک نامه الکترونیکی با استاندارد MIME

در مثال (۵-۹) قالب یک نامه الکترونیکی با استاندارد MIME آورده شده است. اگر به این مثال دقت شود در خط پنجم، عبارت ذیل دیده می‌شود:

Content-Type: Multipart/alternative;boundary=qwertyuiopasdfghjklzxcvbnm

این فیلد دو موضوع را تبیین می‌کند:

اول آنکه نامه دارای چندین بخش مجزا و متوالی است و هر قسمت از آن، نوع و محتوای متفاوتی دارد.

دوم آنکه نامه‌خوان موظف است ابتدا و انتهای هر قسمت را با رشته کاراکتری زیر جدا کرده و هر قسمت را مجزا پردازش و دیکود نماید:

--qwertyuiopasdfghjklzxcvbnm

رشته‌ای که بعد از دو کاراکتر -- آمده است همان رشته‌ای است که بعنوان متمایز کننده بخشها انتخاب شده است. دقت کنید که معمولاً این رشته طولانی و تصادفی است تا درون متن مشابه آن یافت نشود. دو کاراکتر -- نیز نشاندهنده خط متمایز می‌باشد و باید دقیقاً در ابتدای سطر ظاهر شود.

نکته نهائی آنکه فقط وقتی مجبور خواهید بود فیلدها و سرآیندهای معرفی شده را بکار ببرید که بخواهید ساختار نامه را توسط یک ویرایشگر ساده مثل Notepad ایجاد کنید یا خودتان نرم‌افزار نامه‌خوان بنویسید و گرنه این فیلدها بصورت خودکار توسط نرم‌افزار نامه‌خوان شما ایجاد خواهد شد.

۱۴) پروتکل ساده انتقال نامه های الکترونیکی: SMTP^۱

پس از تحلیل ساختار متنی یک نامه الکترونیکی باید ببینیم که برای ارسال نامه یا دریافت آن، از دیدگاه برنامه سرویس دهنده و مشتری چه اتفاقاتی می‌افتد. مشهورترین سرویس دهنده پست الکترونیکی، SMTP نام دارد که روند عملیات آن بسیار ساده است:

ماشین مبداء (یعنی ماشینی که می‌خواهد نامه نوشته و تنظیم شده‌ای را ارسال کند) با پورت شماره ۲۵ از ماشین مقصد که سرویس دهنده SMTP روی آن اجرا شده یک ارتباط TCP برقرار میکند. بنابراین براحتی می‌توانید در ذهن خود مجسم کنید که برنامه سرویس دهنده یک برنامه سوکت است که به پورت ۲۵ گوش می‌دهد (این

^۱ Simple Mail Transfer Protocol

برنامه در محیط یونیکس به نام دایمون SMTP معروف است. دایمونها برنامه هایی هستند که در حالت انتظار می مانند و با یک سیگنال شروع به انجام عملیات خود می نمایند) این برنامه ارتباطات TCP به پورت ۲۵ را می پذیرد.

پس از برقراری ارتباط و پذیرش آن توسط سرویس دهنده، شروع کننده ارتباط (یعنی نرم افزار مشتری یا همان نامه خوان) باید آنقدر صبر کند تا سرویس دهنده مقصد با ارسال یک پیغام اعلام آمادگی نماید. روند اعلام آمادگی و بقیه مراحل مبادله نامه بصورت زیر است:

- سرویس دهنده با ارسال یک رشته متنی که معمولاً بصورت زیر است به برنامه مبدا اعلام آمادگی می نماید:

SMTP service ready آدرس نام حوزه خود 220

مثال :

220 xyz.com STMP service ready

- پس از اعلام آمادگی (کد 220 بمعنای اعلام آمادگی است) برنامه مبدا با ارسال یک رشته که حاوی کلمه HELO (مخفف کلمه سلام) و همچنین آدرس نام حوزه خودش می باشد هویت خود را برای سرویس دهنده آشکار می کند. مثال :

HELO abc.com

- پس از آنکه سرویس دهنده هویت فرستنده پیام را ارزیابی کرد در صورتی که تمایل به دریافت نامه داشته باشد با کد ۲۵۰ و رشته ای که در ادامه آن می آید اعلام آمادگی می نماید. مثال:

250 xyz.com says hello to abc.com

- سرویس دهنده صاحب نامه را بررسی کرده و در صورتی که منعی برای دریافت نامه چنین شخصی وضع نشده باشد مجدداً با کد ۲۵۰ و رشته ای که در ادامه می آید اعلام آمادگی می کند. مثال:

250 sender ok

- برنامه مبدا گیرنده نامه را معرفی می کند. مثال:

PCPT TO:<carolyn@xyz.com>

- بار دیگر سرویس دهنده، گیرنده نهایی نامه را ارزیابی کرده و بررسی می کند که آیا چنین شخصی (در مثال بالا Carolyn) وجود دارد یا خیر. در صورتی که امکان

دریافت نامه وجود داشته باشد برای بار سوم با کد ۲۵۰ مطابق مثال زیر اعلام آمادگی می‌شود:

250 recipient ok

- برنامه مبداء اعلام می‌کند که برای ارسال داده‌ها که کلاً کاراکترهای اسکی با کد زیر 128 هستند آماده است؛ کلمه DATA بدون هیچ حرف اضافه به عنوان اعلام آمادگی برای ارسال است. مثال:

DATA

- سرویس دهنده ضمن اعلام آمادگی جهت دریافت داده‌ها به مبداء اعلام می‌کند که پس از آخرین سطر نامه یک خط که فقط شامل تک کاراکتر '۰' است ارسال کند تا انتهای نامه مشخص باشد. مثال:

354 Send mail; end with "." on a line by itself

- مبداء، نامه‌ای را که با استاندارد RFC822 یا MIME تنظیم شده است، ارسال می‌کند؛ در انتهای نامه خطی که شامل تک حرف '۰' است به معنای خاتمه نامه، ارسال می‌شود.
- سرویس دهنده دریافت موفقیت‌آمیز نامه را اعلام می‌کند؛ این اعلام بصورت زیر است:

250 message accepted

- مبداء با ارسال رشته QUIT اعلام خروج می‌کند. (البته می‌تواند مجدداً از مرحله دوم شروع کرده و نامه دیگری را ارسال کند.)

QUIT

- فرستنده ضمن تأیید خروج و معرفی مجدد خود اعلام می‌کند که ارتباط TCP را قطع خواهد کرد، در این جا کار انتقال خاتمه یافته است. مثال:

221 xyz.com closing connection

به یک انتقال واقعی پیام در مثال (۶-۹) دقت نمایید. در این مثال نامه الکترونیکی مثال (۵-۹) طبق روال فوق بین مبداء و مقصد مبادله شده است. در این مثال داده هائی که توسط مبداء ارسال شده با C: و آنهائی که توسط سرویس دهنده ارسال شده با S: نشان داده شده است. (S: , C: ارسال نمی‌شود)

سیستم پست الکترونیکی SMTP در مواردی بهبود داده شده که مشخصات آن در RFC-1425 آمده است.

```
S: 220 xyz.com service ready
C: HELO abc.com
S: 250 xyz.com says hello to abc.com
C: MAIL FROM: <erlinor@abc.com>
S: 250 sender ok
C: RCPT TO: <carolyn@xyz.com>
S: 250 receipient ok
C: DATA
S: 354 Send mail; end with "." on a line by itself
C: From: erlinor@abc.com
C: To: carolyn@xyz.com
C: MIME-Version: 1.0
C: Message-Id: <0704760941 AA00747@abc.com>
C: Content-Type: multipart/alternative; boundary=qwertyuiopasdfghjklzxcvbnm
C: Subject: Earth orbits sun integral number of times
C:
C: This is preamble. The user agent ignore it. Have a nice time.
C: --qwertyuiopasdfghjklzxcvbnm
C: Content-Type: text/richtext
C:
C: Happy birthday to you
C: Happy birthday dear <bold>Carolyn</bold>
C:
C: --qwertyuiopasdfghjklzxcvbnm
C: Content-Type: message/external-body;
C: Access-type="anon-ftp";
C: Site="bicycle.abc.com";
C: Directory="pub";
C: Name="birthday.snd";
C:
C: Content-Type: audio/basic
C: Content-Transfer-Encoding: Base64
C: --qwertyuiopasdfghjklzxcvbnm
C: .
S: 250 message accepted
C: QUIT
S: 221 xyz.com closing connection
```

مثال (۶-۹) مراحل انتقال یک نامه الکترونیکی

(۵) تمویل نهائی نامه^۱

برای شروع این بخش شما را با یک سوال روبرو می‌کنیم: کاربری که بطور نامنظم و پراکنده به شبکه اینترنت متصل می‌شود چگونه می‌تواند روی سیستم SMTP را نصب و شبانه روز سیستم خود را برای دریافت نامه های خود روشن و فعال نگه دارد؟

جواب این سوال ساده است کاربر باید وظیفه دریافت نامه هایش را به یک کارگزار مطمئن که بصورت دائم فعال است و سیستم SMTP را فراهم کرده بسپارد. هر موقع که نامه‌ای برای او ارسال می‌شود کارگزارش آنرا دریافت و ذخیره می‌نماید. هنگامی که کاربر تمایل داشت سری به نامه های رسیده‌اش بزند و نامه‌ای را دریافت نماید بایستی با این کارگزار ارتباط برقرار نماید.

POP3^۲ پروتکلی ساده برای دریافت نامه‌های الکترونیکی از سرورس‌دهنده کارگزار شما است. این پروتکل مجموعه‌ای از فرامین برای برقراری اتصال، قطع اتصال، دریافت پیام‌ها و حذف آنها می‌باشد. این پروتکل نیز همانند SMTP فرامین متنی دارد.

بدون آنکه بخواهیم وارد نکات ریز این پروتکل بشویم امکانات کلی آنرا معرفی می‌کنیم.

- **نصب فیلتر:** شما از سیستم پست الکترونیکی می‌خواهید که نامه های دریافتی از یک آدرس خاص را اصلاً تحویل نگیرد یا نامه هائی که قسمت موضوع^۳ آن شامل کلمات کلیدی خاص می‌شود را حذف کند. یا مثلاً نامه هائی را که کلمه‌ای خاص در آدرس فرستنده‌اش یافت می‌شود حذف نماید (این امکان برای رهائی از شر مزاحمت شرکتهای تبلیغاتی که پیاپی نامه ارسال می‌کنند بسیار مفید است)
- **ارسال نامه های رسیده به آدرسی دیگر^۴:** فرض کنید که در شهر خود از کارگزاری صندوق پست الکترونیکی گرفته‌اید و برای مدتی به خارج از کشور سفر می‌کنید، می‌توانید از سیستم پستی خود بخواهید که نامه های شما به آدرس پستی جدید ارسال شود. همچنین این امکان وجود دارد که یک نامه را بدون دخل و تصرف به آدرسی دیگر ارسال کنید.

^۱ Final Delivery

^۲ Post Office Protocol

^۳ Subject

^۴ Forwarding

• **Vacation Daemon**: فرض کنید می‌خواهید برای چند روزی جایی بروید که دسترسی به اینترنت ندارید. می‌توانید سیستم پستی را وادار کنید که ضمن دریافت نامه‌ها یک پیغام برای ارسال کنندگان نامه بفرستد. مثلاً یک شرکت که هر روز نامه‌های زیادی را دریافت می‌کند و ممکن است پاسخ دستی به آنها طولانی مدت شود، می‌تواند از سیستم پستی بخواهد که بصورت خودکار برای فرستندگان نامه پیامی ارسال کرده و به پرسشهای متداول آنها پاسخ بدهد.

اطلاعات دقیق این پروتکل در RFC-1225 تشریح شده است.

۶) مراجع این فصل

مجموعه مراجع زیر می‌توانند برای دست آوردن جزئیات دقیق و تحقیق جامع در مورد مفاهیم معرفی شده در این فصل مفید واقع شوند.

RFC 1341	"MIME (Multipurpose Internet Mail Extensions) Mechanisms for Specifying and Describing the Format of Internet Message Bodies," Borenstein, N.; Freed, N.; 1992
RFC 1143	"Q Method of Implementing Telnet Option Negotiation," Bernstein, D.J.; 1990
RFC 1090	"SMTP on X.25," Ullmann, R.; 1989
RFC 1056	"PCMAIL: A Distributed Mail System for Personal Computers," Lambert, M.L.; 1988
RFC 974	"Mail Routing and the Domain System," Partridge, C.; 1986
RFC 822	"Standard for the Format of ARPA Internet Text Messages," Crocker, D.; 1982
RFC 821	"Simple Mail Transfer Protocol," Postel, J.B.; 1982