Introducción:

Correo Electrónico:

- Servicio de red que permite a los usuarios enviar y recibir mensajes mediante sistemas de comunicación electrónica.
- Principalmente se usa este nombre para denominar al sistema que provee este servicio en internet, mediante el protocolo SMTP, aunque por extensión también puede verse aplicado a sistemas análogos que usen otras tecnologías.

Cliente de Correo Electrónico:

- Es un software de computadora usado para leer y enviar mensajes de correo electrónico
- Originalmente fueron pensados para solo leer mensajes de correo de usuario, enviados por:
    Agente de Reparto de Correo (Mail Delivery Agent)
    Agente de Transferencia de Correo (Mail Transfer Agent)
- Dado que las diferentes versiones de Microsoft Windows para uso doméstico nunca han proporcionado un agente de transferencia de correo, los clientes de correo más modernos deben soportar protocolos como POP3 e Internet Message Access Protocol (IMAP) para comunicarse con un MTA remoto localizado en la máquina de proveedores de correo electrónico.
- La gran mayoría de clientes de correo electrónico emplean el Protocolo de Transferencia Simple de Correo (Simple Mail Transfer Protocol, SMTP) para enviar los mensajes de correo electrónico.
- Un importante estándar soportado por la mayoría de los clientes de correo electrónico es MIME, que se emplea para el envío de archivos binarios adjuntos al correo. Los adjuntos son ficheros que no forman parte del correo electrónico propiamente dicho, pero que se envían junto con éste.

Funcionamiento

Escritura del mensaje:

No se pueden mandar mensajes entre computadores personales o entre dos terminales de una computadora central. Los mensajes se archivan en un buzón (una manera rápida de mandar mensajes). Cuando una persona decide escribir un correo electrónico, su programa (o correo web) le pedirá como mínimo tres cosas:

- **Destinatario**: una o varias direcciones de correo a las que ha de llegar el mensaje
- **Asunto**: una descripción corta que verá la persona que lo reciba antes de abrir el correo
- El propio **mensaje**. Puede ser sólo texto, o incluir formato, y no hay límite de tamaño

Además, se suele dar la opción de incluir archivos *adjuntos* al mensaje. Esto permite traspasar datos informáticos de cualquier tipo mediante el correo electrónico.

Para especificar el **destinatario** del mensaje, se escribe su dirección de correo en el campo llamado **Para** dentro de la interfaz (ver imagen de arriba). Si el destino son varias personas, normalmente se puede usar una lista con todas las direcciones, separadas por comas o punto y coma.

Además del campo **Para** existen los campos *CC* y *CCO*, que son opcionales y sirven para hacer llegar copias del mensaje a otras personas:

- Campo **CC** (*Copia de Carbon*): quienes estén en esta lista recibirán también el mensaje, pero verán que no va dirigido a ellos, sino a quien esté puesto en el campo **Para**. Como el campo **CC** lo ven todos los que reciben el mensaje, tanto el destinatario principal como los del campo **CC**

pueden ver la lista completa.
- Campo **CCO** (*Copia de Carbon Oculta*): una variante del **CC**, que hace que los destinatarios reciban el mensaje sin aparecer en ninguna lista. Por tanto, el campo **CCO** nunca lo ve ningún destinatario.

Un ejemplo: *Ana* escribe un correo electrónico a *Beatriz* (su profesora), para enviarle un trabajo. Sus compañeros de grupo, *Carlos* y *David*, quieren recibir una copia del mensaje como comprobante de que se ha enviado correctamente, así que les incluye en el campo **CC**. Por último, sabe que a su hermano *Esteban* también le gustaría ver este trabajo aunque no forma parte del grupo, así que le incluye en el campo **CCO** para que reciba una copia sin que los demás se enteren.

Entonces:

- *Beatriz* recibe el mensaje dirigido a ella (sale en el campo **Para**), y ve que *Carlos* y *David* también lo han recibido
- *Carlos* recibe un mensaje que no va dirigido a él, pero ve que aparece en el campo **CC**, y por eso lo recibe. En el campo **Para** sigue viendo a *Beatriz*
- *David*, igual que *Carlos*, ya que estaban en la misma lista (**CC**)
- *Esteban* recibe el correo de *Ana*, que está dirigido a *Beatriz*. Ve que *Carlos* y *David* también lo han recibido (ya que salen en el **CC**), pero no se puede ver a él mismo en ninguna lista, cosa que le extraña. Al final, supone que es que *Ana* le incluyó en el campo **CCO**.
- Campo **Reply-To** (responder) Dirección dónde el emisor quiere que se le conteste. Muy útil si el emisor dispone de varias cuentas.
- Campo **Date** (fecha, y hora, del mensaje) Fecha y hora de cuando se envío del mensaje. Si el sistema que envía el mensaje tiene la fecha y/u hora equivocadas, puede generar confusión.

Otros campos, menos importantes son:

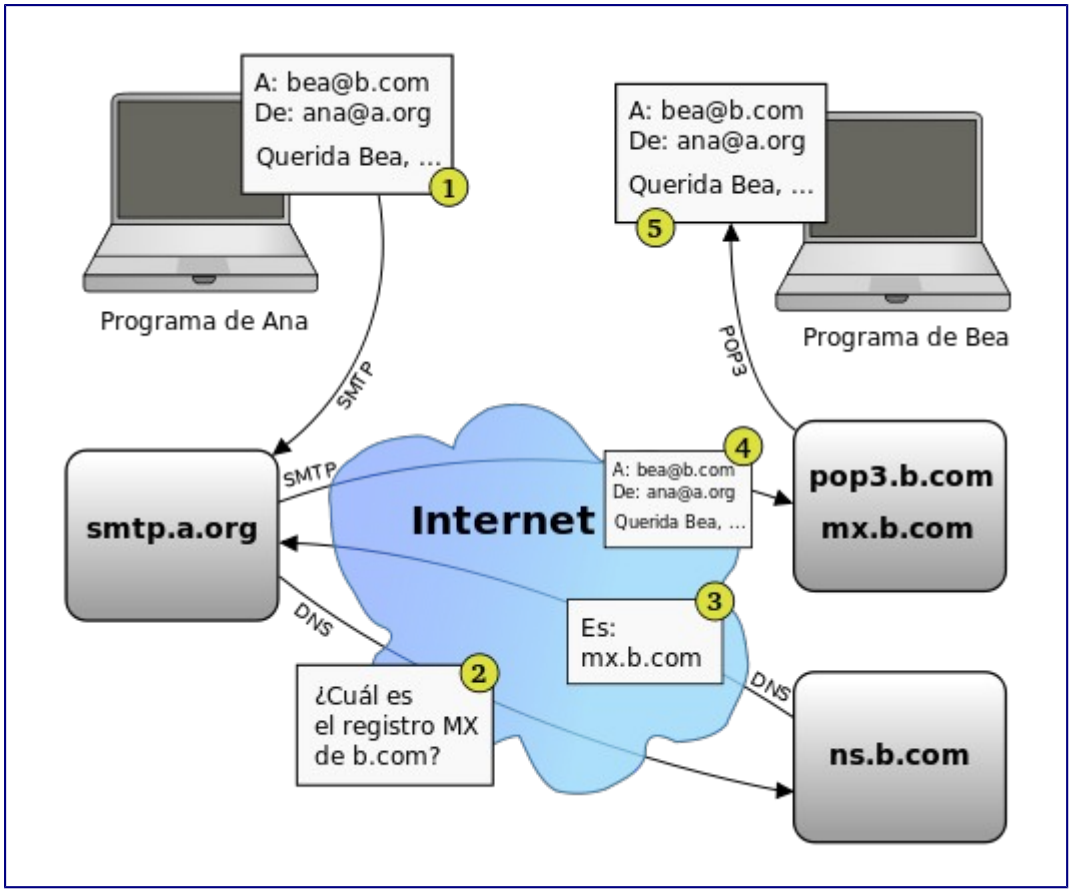
- Sender: Sistema o persona que lo envía
- Received: Lista de los MTA que lo transportaron
- Message-Id: Número único para referencia
- In-Reply-to: Id. del mensaje que se contesta
- References: Otros Id del mensaje
- Keywords: Palabras claves de usuario
- X-Usuario: Definibles por el usuario

Encabezado de un correo electrónico.

La cabecera del mensaje normalmente, se muestra resumida. Para ver todos los detalles bastará con expandir, mediante la opción oportuna, dicha cabecera.

Envío

El envío de un mensaje de correo es un proceso largo y complejo. Éste es un esquema de un caso típico:



En este ejemplo ficticio, *Ana* (**ana@a.org**) envía un correo a *Bea* (**bea@b.com**). Cada una de ellas tiene su cuenta de correo electrónico en un servidor distinto (una en `a.org`, otra en `b.com`), pero éstos se pondrán en contacto para transferir el mensaje.
Por pasos:

1. *Ana* escribe el correo en su programa cliente de correo electronico. Al darle a *Enviar*, el programa contacta con el servidor de correo usado por *Ana* (en este caso, `smtp.a.org`). Se comunica usando un lenguaje conocido como protocolo SMTP. Le transfiere el correo, y le da la orden de enviarlo.
2. El servidor SMTP ve que ha de entregar un correo a alguien del dominio `b.com`, pero no sabe con qué ordenador tiene que contactar. Por eso consulta a su servidor DNS (usando el protocolo DNS), y le pregunta quién es el encargado de gestionar el correo del dominio `b.com`. Técnicamente, le está preguntando el registro MX asociado a ese dominio.
3. Como respuesta a esta petición, el servidor DNS contesta con el nombre del dominio del servidor de correo de *Bea*. En este caso es `mx.b.com`; es un ordenador gestionado por el proveedor de internet de *Bea*.
4. El servidor SMTP (`smtp.a.org`) ya puede contactar con `mx.b.com` y transferirle el mensaje, que quedará guardado en este ordenador. Se usa otra vez el protocolo SMTP.
5. Más adelante (quizás días después), *Bea* aprieta el botón "*Recibir nuevo correo*" en su programa cliente de correo. Esto empieza una conexión, mediante el protocolo POP3 o IMAP, al ordenador que está guardando los correos nuevos que le han llegado. Este ordenador (`pop3.b.com`) es el mismo que el del paso anterior (`mx.b.com`), ya que se encarga tanto de recibir correos del exterior como de entregárselos a sus usuarios. En el esquema, *Bea* recibe el mensaje de *Ana* mediante el protocolo POP3.

Ésta es la secuencia básica, pero pueden darse varios casos especiales:

- Si ambas personas están en la misma red (una Intranet de una empresa, por ejemplo), entonces no se pasa por Internet. También es posible que el servidor de correo de *Ana* y el de *Bea* sean el mismo ordenador.
- *Ana* podría tener instalado un servidor SMTP en su ordenador, de forma que el paso 1 se haría en su mismo ordenador. De la misma forma, *Bea* podría tener su servidor de correo en el propio

ordenador.
- Una persona puede no usar un programa de correo electronico, sino un webmail. El proceso es casi el mismo, pero se usan conexiones HTTP para acceder al correo de cada usuario en vez de usar SMTP o IMAP/POP3.
- Normalmente existe más de un servidor de correo (MX) disponible, para que aunque uno falle, se siga pudiendo recibir correo.

Si el usuario quiere puede almacenar los mensajes que envía, bien de forma automática (con la opción correspondiente), bien sólo para los mensajes que así lo desee. Estos mensajes quedan guardados en la carpeta "Enviados".

Recepcion :

Cuando una persona recibe un mensaje de correo electrónico puede verse en la bandeja de entrada un resumen de él:

- Remitente (o De o De: o From o From: -en inglés-): esta casilla indica quién envía el mensaje. Puede aparecer el nombre de la persona o entidad que nos lo envía (o su apodo o lo que desee el remitente). Si quien envía el mensaje no ha configurado su programa o correo web al respecto aparecerá su dirección de correo electrónico.
- Asunto: en este campo se ve el tema que trata el mensaje (o lo que el remitente de él desee). Si quien envía el mensaje ha dejado esta casilla en blanco se lee [ninguno] o [sin asunto]
    - Si el mensaje es una respuesta el asunto suele empezar por RE: o Re: (abreviatura de responder o *reply* -en inglés-, seguida de dos puntos). Aunque según de dónde proceda el mensaje pueden aparecer An: (del alemán *antwort*), Sv: (del sueco *svar*), etc.
    - Cuando el mensaje procede de un reenvío el asunto suele comenzar por RV: (abreviatura de reenviar) o Fwd: (del inglés *forward*), aunque a veces empieza por Rm: (abreviatura de remitir)
- Fecha: esta casilla indica cuándo fue enviado el mensaje o cuándo ha llegado a la bandeja de entrada del receptor. Puede haber dos casillas que sustituyan a este campo, una para indicar la fecha y hora de expedición del mensaje y otra para expresar el momento de su recepción.

Además pueden aparecer otras casillas como:

- Tamaño: indica el espacio que ocupa el mensaje y, en su caso, fichero(s) adjunto(s)
- Destinatarios (o Para o Para: o To o To: -en inglés-): muestra a quiénes se envió el mensaje
- **Datos adjuntos**: si aparece una marca (habitualmente un clip) significa que el mensaje viene con uno o varios ficheros anexos
- Prioridad: expresa la importancia o urgencia del mensaje según el remitente (alta -se suele indicar con un signo de exclamación-, normal -no suele llevar marca alguna- o baja -suele indicarse con una flecha apuntando para abajo-)
- Marca (de seguimiento): si está activada (p.e. mostrando una bandera) indica que hay que tener en cuenta este mensaje (previamente lo ha marcado la persona que lo ha recibido)
- Inspeccionar u omitir: pinchando en esta casilla se puede marcar el mensaje para inspeccionarlo (suelen aparecer unas gafas en la casilla y ponerse de color llamativo -normalmente rojo- las letras de los demás campos). Pinchando otra vez se puede marcar para omitirlo (suele aparecer el símbolo de "prohibido el paso" en este campo y ponerse en un tono suave -normalmente gris- las letras de las demás casillas). Pinchando una vez más volvemos a dejar el mensaje sin ninguna de las dos marcas mencionadas

- Cuenta: Si utilizamos un cliente de correo electronico configurado con varias cuentas de correo esta casilla indica a cuál de ellas ha llegado el mensaje en cuestión
- Primeras palabras del (cuerpo del) mensaje

Los mensajes recibidos pero sin haber sido leídos aún suelen mostrar su resumen en negrita. Después de su lectura figuran con letra normal. A veces si seleccionamos estos mensajes sin abrirlos podemos ver abajo una previsualización de su contenido.

Si el destinatario desea leer el mensaje tiene que abrirlo (normalmente haciendo (doble) clic sobre el contenido de su asunto con el puntero del ratón). Entonces el receptor puede ver un encabezado arriba seguido por el cuerpo del mensaje. En la cabecera del mensaje aparecen varias o todas las casillas arriba mencionadas (salvo las primeras palabras del cuerpo del mensaje). Los ficheros adjuntos, si existen, pueden aparecer en el encabezado o debajo del cuerpo del mensaje.

Una vez que el destinatario ha recibido (y, normalmente, leído) el mensaje puede hacer varias cosas con él. Normalmente los sistemas de correo (tanto programas como correo web) ofrecen opciones como:

- **Responder**: escribir un mensaje a la persona que ha mandado el correo (que es sólo una). Existe la variante **Responder a todos**, que pone como destinatarios tanto al que lo envía como a quienes estaban en el campo *CC*
- **Reenviar** (o **remitir**): pasar este correo a una tercera persona, que verá quién era el origen y destinatario original, junto con el cuerpo del mensaje. De forma opcional se puede añadir más texto al mensaje o borrar los encabezados e incluso el cuerpo (o parte de él) de anteriores envíos del mensaje.
- **Marcar como correo no deseado** (***spam***): separar el correo y esconderlo para que no moleste, de paso instruyendo al programa para que intente detectar mejor mensajes parecidos a éste. Se usa para evitar la publicidad no solicitada.
- **Archivar**: guardar el mensaje en el ordenador, pero sin borrarlo, de forma que se pueda consultar más adelante. Esta opción no está en forma explícita, ya que estos programas guardan los mensajes automáticamente.
- **Borrar**: Se envía el mensaje a una carpeta *Elementos eliminados* que puede ser vaciada posteriormente.
- **Mover a carpeta** o **Añadir etiquetas**: algunos sistemas permiten catalogar los mensajes en distintos apartados según el tema del que traten. Otros permiten añadir marcas definidas por el usuario (ej: "trabajo", "casa", etc.).

Problemas

El principal problema actual es el ***correo no deseado***, que se refiere a la recepción de correos no solicitados, normalmente de publicidad engañosa, y en grandes cantidades, promoviendo *pornografía* y otros productos y servicios de calidad sospechosa.
Usualmente los mensajes indican como remitente del correo una dirección falsa. Por esta razón, es más difícil localizar a los verdaderos remitentes, y no sirve de nada contestar a los mensajes de correo no deseado: las respuestas serán recibidas por usuarios que nada tienen que ver con ellos. Por ahora, el servicio de correo electrónico no puede identificar los mensajes de forma que se pueda discriminar la verdadera dirección de correo electrónico del remitente, de una falsa. Esta situación que puede resultar chocante en un primer momento, es semejante por ejemplo a la que ocurre con el correo postal ordinario: nada impide poner en una carta o postal una dirección de remitente aleatoria: el correo llegará en cualquier caso. No obstante, hay tecnologías desarrolladas en esta dirección: por ejemplo el remitente

puede firmar sus mensajes mediante criptografía de clave pública.
Además del *correo no deseado*, existen otros problemas que afectan a la seguridad y veracidad de este medio de comunicación:

- los **virus informáticos**, que se propagan mediante ficheros adjuntos infectando el ordenador de quien los abre
- la **suplantación de identidad**, que es correo fraudulento que generalmente intenta conseguir información bancaria
- los **bulos** (bromas, burlas, o hoax), que difunden noticias falsas masívamente
- las **cadenas de correo electrónico**, que consisten en reenviar un mensaje a mucha gente; aunque parece inofensivo, la publicación de listas de direcciones de correo contribuye a la propagación a gran escala del *correo no deseado* y de mensajes con virus, *suplantadores de identidad* y *engaños*.

Pérdida progresiva de la privacidad
En 2014 los principales proveedores de correo web como Google, Hotmail o Yahoo exigen como requisito proveer datos personales como un **número de teléfono obligatorio** o una dirección de correo alternativa obligatoria para así impedir las altas anónimas o de personas que no puedan tener acceso a la compra de un teléfono móvil.

Servicios de correo electrónico

Principales proveedores de servicios de correo electrónico gratuito:

- Gmail: webmail, POP3 e IMAP
- Outlook.com: webmail y POP3
- Yahoo! Mail: webmail y POP3 con publicidad

Los servicios de correo de pago los suelen dar las compañías de acceso a Internet o los registradores de dominios. También hay servicios especiales, como Mailinator, que ofrece cuentas de correo temporales (caducan en poco tiempo) pero que no necesitan registro.

Programas para leer y organizar correo

- Windows Live Mail: Windows.
- Evolution: GNU/Linux.
- Mail: Mac OS X e iOS.
- Outlook Express: Windows.
- Thunderbird: Windows, GNU/Linux, Mac OS X.

Programas servidores de correo

Éstos son usados por el ordenador servidor de correo para proporcionar el servicio a los clientes, que podrán usarlo mediante un *programa de correo*.

- Principales programas servidores:
    - Mercury Mail Server: Windows, Unix, GNU/Linux.
    - Microsoft Exchange Server: Windows.
    - MailEnable: Windows.
    - MDaemon: Windows.
    - Exim: Unix.
    - Sendmail: Unix.
    - Qmail: Unix.
    - Postfix: Unix.
    - Zimbra: Unix, Windows.

- Lotus Domino: GNU/Linux, OS400, Windows.
- Windows Live Mail
- Windows Live Messenger

También existen otros programas para dar el servicio de *correo web*.

==========

MIME

MIME (Múltiples Internet Mail Extensions) es una extensión del protocolo de correo de Internet original que permite a las personas utilizan el protocolo para el intercambio de diferentes tipos de archivos de datos en Internet: audio, vídeo, imágenes, programas de aplicaciones y otros tipos, así como el texto ASCII manejado en el protocolo original, el *Simple Mail Transfer Protocol* (SMTP).

Los servidores insertan la cabecera MIME al comienzo de cualquier transmisión Web. Los clientes usan esta cabecera para seleccionar una aplicación "programa" apropiada para el tipo de datos que indica la cabecera. Algunos de estos programas están integradas en el cliente Web o el navegador (por ejemplo, todos los navegadores vienen con GIF y JPEG programas de imagen, así como la capacidad de manejar archivos HTML); otros jugadores pueden necesitar ser descargado.

En sentido general las extensiones de MIME van encaminadas a soportar:

- Texto en conjuntos de caracteres que no sean ASCII;
- Archivos adjuntos que no contenían texto;
- Cuerpos de los mensajes con varias partes (multi-part);
- Información de encabezados con conjuntos de caracteres distintos de ASCII.

Aunque MIME fue diseñado principalmente para SMTP, hoy en día su uso ha crecido más allá de describir el contenido del correo electrónico y ahora a menudo incluye descripciones de tipo de contenido en general, incluso para la web.

Prácticamente todos los mensajes de correo electrónico escritos por personas en Internet y una proporción considerable de los mensajes automático son transmitidos en formato MIME a través de SMTP. El Correo electrónico de Internet está tan estrechamente asociado con los estándares SMTP y MIME que a veces se llama correo electrónico **SMTP/MIME**.

Los tipos de contenido definidos por el estándar MIME tienen gran importancia también fuera del contexto de correo electrónico, como por ejemplo algunos protocolos de red como HTTP para la World Wide Web (WWW).

MIME está especificado en seis vínculos memorandos RFC: RFC 2045, RFC 2046, RFC 2047, RFC 4288, RFC 4289 y RFC 2049.

> Una RFC (**solicitud de comentarios**) es una publicación del Grupo de Trabajo de Ingeniería de Internet (IETF) y la Sociedad de Internet, los principales organismos de desarrollo técnico y que establecen normas para la Internet.

En la actualidad ningún programa de correo electrónico o navegador de Internet puede considerarse completo si no acepta MIME en sus diferentes facetas (texto y formatos de archivo).

Introducción

> El protocolo básico de transmisión de mensajes electrónicos de Internet soporta sólo caracteres ASCII de 7 bit. Esto limita los mensajes de correo electrónico, ya que incluyen sólo caracteres suficientes para escribir en un número reducido de lenguajes, principalmente Inglés. Otros lenguajes basados en el Alfabeto latino es adicionalmente un componente fundamental en protocolos de comunicación como HTTP, el que requiere que los datos sean transmitidos como un e-mail aunque los datos pueden no ser un e-mail propiamente dicho. Los clientes de correo y los servidores de correo convierten automáticamente desde y a formato MIME cuando envían o reciben e-mails (SMTP/MIME).

Cabeceras MIME

MIME-Version

La presencia de este encabezado indica que el mensaje utiliza el formato MIME. Su valor es típicamente igual a "1.0" por lo que este encabezado aparece como:

```
MIME-Version: 1.0
```

Según Nathaniel Borenstein, el co-creador de MIME, la intención era permitir cambios a MIME, avanzar a la versión 2.0 y así sucesivamente, pero esta decisión llevó al resultado opuesto, haciendo casi imposible la creación de una nueva versión del estándar.

"No se especificó adecuadamente cómo manejar una versión futura de MIME", dijo Borenstein. "Así que si escribes algo que el 1.0 conoce, ¿qué debería hacer si se encuentra con un 2.0 o 1.1? En cierto modo me pareció que era obvio, pero resultó que todo el mundo lo implementó en diferentes maneras. Y el resultado es que sería casi imposible para Internet definir alguna vez un 2.0 o un 1.1. "

En una reunión de IETF realizada en Julio 2007 se decidió mantener el número de versión en "1.0" aunque se han realizado muchas actualizaciones a la versión de MIME.

Content-Type

Este encabezado indica el tipo de medio que representa el contenido del mensaje, consiste en un tipo (*type)* y un subtipo (*subtype)*. Por ejemplo:

```
Content-Type: text / plain
```

A través del uso del tipo **multiparte** (*multipart*), MIME da la posibilidad de crear mensajes que tengan partes y subpartes organizadas en una **estructura de árbol** en la que los nodos *hoja* pueden ser cualquier tipo de contenido no multiparte y los nodos que no son *hojas* pueden ser de cualquiera de las variedades de tipos multiparte. Este mecanismo soporta:

- **Mensajes de texto simples usando *text / plain*** (el valor predeterminado de "Content-Type:")
- Texto y los archivos adjuntos (*multipart / mixtos* con una parte *text / plain* y en otras partes que no son de texto). Por ejemplo: application / pdf para documentos pdf, application / vnd.oasis.opendocument.text para OpenDocument text). Un mensaje MIME que incluye un archivo adjunto generalmente indica el nombre original del archivo con un encabezado "Content-disposition:" o por un atributo name de Content-Type, por lo que el tipo o formato del archivo se indica usando tanto el encabezado MIME content-type y la extensión del archivo (usualmente dependiente del SO).

```
Content-Type: application/vnd.oasis.opendocument.text;
    name="Carta.odt"
Content-Disposition: inline;
    filename="Carta.odt"
```

- Responder con el mensaje original adjunto (*multipart / mixed* con una parte *text / plain* y el mensaje original como una parte *mensaje / rfc822*)
- Contenido alternativo, un mensaje que contiene tanto un texto plano como en otro formato, usualmente HTML (*multiparte / alternativo* con el mismo contenido en *text / plain* y formato *text / html*)
- Imagen, audio, vídeo y aplicaciones (por ejemplo, *image / jpeg* , *audio / mp3* , *video / mp4* , y *application / msword* y así sucesivamente)
- Muchas otras construcciones de mensajes

Content-Disposition

Las especificaciones MIME originales sólo describen la estructura de los mensajes de correo. Además, no abordan la cuestión de estilos de presentación. El campo de la cabecera content-disposition se añadió en el RFC 2183 para especificar el estilo de presentación. Una parte MIME puede tener:

- Una *línea* content-disposition, lo que significa que se debe mostrar automáticamente cuando se muestra el mensaje, o
- Un *archivo adjunto* content-disposition, en cuyo caso no se muestra automáticamente y requiere algún tipo de acción por parte del usuario para abrirlo.

Además del estilo de presentación, el encabezado content-disposition también ofrece campos para especificar el nombre del archivo, la fecha de creación y fecha de modificación, que puede ser utilizado por el agente de usuario de correo del lector para almacenar el archivo adjunto.

El siguiente ejemplo está tomado de RFC 2183, donde se define la cabecera

```
Content-Disposition: attachment; filename = genome.jpeg;
  modificación-date = "Mié, 12 de febrero 1997 16:29:51 -0500";
```

El nombre de archivo puede ser codificado como se define en el RFC 2231.

A partir de 2010, una buena mayoría de los agentes de correo no siguen esta receta totalmente. El cliente de correo ampliamente utilizado, Mozilla Thunderbird, toma sus propias decisiones acerca de qué partes MIME se debe mostrar automáticamente, haciendo caso omiso de los encabezados content-disposition en los mensajes. Los Thunderbird anterior a la versión 3 también envían mensajes nuevos compuestos con inline content-disposition para todas las partes MIME. La mayoría de los usuarios no son conscientes de cómo configurar el archivo adjunto (attachment) content-disposition. Muchos agentes de usuario de correo también envían mensajes con el nombre de archivo en parámetro nombre (name) del encabezado content-type en lugar del nombre del archivo (filename) de parámetros de la cabecera content-disposition. Esta práctica se desaconseja - el nombre del archivo debe ser especificado ya sea a través de sólo el nombre del archivo de parámetros, o tanto a través del nombre de archivo y los parámetros nombres (name).

En HTTP, el encabezado de respuesta `Content-Disposition: attachment` se utiliza generalmente para hacer alusión al cliente al presentar el cuerpo de la respuesta como un archivo descargable. Normalmente, cuando se recibe una respuesta, el navegador Web le pedirá al usuario guardar su contenido como un archivo en lugar de mostrarla como una página en una ventana del navegador, con el parámetro *nombre del archivo (filename)* que sugiere el nombre de archivo por defecto (esto es útil para generar contenido dinámico, donde se deriva el nombre de archivo de la URL puede ser entendible o confuso para el usuario).

Content-Transfer-Encoding

En Junio de 1992, MIME (RFC 1341 queda obsoleta por la nueva RFC 2045) define un conjunto de métodos para representar datos binarios usando texto ASCII. El encabezado MIME content-transfer-encoding: indica el método que ha sido usado. La RFC y la lista de IANA definen los siguientes valores, que no son sensibles a mayúsculas ni minúsculas:

- Indica si es o no un esquema de **codificación de binario a texto**, se utiliza en la parte superior de la codificación original tal

como se especifica en la cabecera Content-Type:

1. Si un método de **codificación de binario a texto**, se ha utilizado, se declara uno.
2. Si no, se proporciona una etiqueta descriptiva para el formato del contenido, con respecto a la presencia de 8 bits o contenido binario.

- Adecuado para usar con SMTP:
  - **7bit** - soporta hasta 998 octetos por línea de código; los caracteres están en el rango entre 1..127 con CR y LF (códigos 13 y 10 respectivamente) que sólo pueden aparecer como parte de un fin de línea CRLF. Este es el valor predeterminado.
  - **Quoted printable (QP)** - usado para codificar secuencias arbitrarias de octetos de forma que satisfaga las reglas de 7bit. Fue diseñado para ser eficiente y en la mayoría de los casos legible para un humano cuando es usado con datos de texto que consisten primariamente en caracteres del conjunto US-ASCII y también contiene una pequeña proporción de bytes con valor fuera de ese rango.
  - **base64** - usado para codificar secuencias arbitrarias de octetos de forma que satisfaga las reglas de 7bit. Tiene una sobrecarga fija al ejecutar el algoritmo y tiene el propósito de ser usado con datos que no sean de texto o textos que contengan pocos valores dentro del rango de ASCII.
- Adecuado para su uso con servidores SMTP que apoyan el 8BITMIME **extensión SMTP** (RFC 6152):
  - **8bit** - soporta hasta 998 octetos por línea de código, los caracteres están en el rango entre 1..256 con CR y LF (códigos 13 y 10 respectivamente) que sólo pueden aparecer como parte de un fin de línea CRLF.
- Adecuado para su uso con servidores SMTP que apoyan la extensión BINARYMIME SMTP (RFC 3030):
  - **binario** - cualquier secuencia de octetos.

No hay codificación definida que está diseñado explícitamente para enviar datos binarios arbitrarios a través de un soporte SMTP con la extensión 8BITMIME. Por lo tanto, si BINARYMIME no es compatible, base64 o quoted-printable (con sus ineficiencias asociadas) tienen que ser usadas aun. Esta restricción no se aplica a otros usos de MIME como Servicios Web con archivos adjuntos MIME o MTOM (masa máxima de despegue).

Encoded-Word

Desde la RFC 2822, los nombres y valores de los encabezados MIME de mensajes son siempre caracteres ASCII; los valores que contengan otro tipo de caracteres tienen que usar la sintaxis de palabra codificada o encoded-word (RFC 2047) en lugar del texto literal. Esta sintaxis utiliza una cadena de caracteres ASCII que indica el conjunto de caracteres original (el "charset") y el content-transfer-encoding usado para mapear los bytes del conjunto original a caracteres ASCII.

Su forma general es:

```
=?charset?codificación?texto codificado?=
```

- *charset* puede ser cualquier conjunto de caracteres registrada con IANA. Típicamente coincidirá con el charset del cuerpo del mensaje.
- *Encoding* (codificación) puede ser " `Q` "denota Q-encoding que es similar al quoted-printable o " `B` "que denota la codificación base64.
- *encoded text* (texto codificado) es el texto codificado con Q-encoding o base64.
- Una *encoded-word* (palabra codificada) no puede contener más de 75 caracteres de largo, incluyendo *charset* , *encoding* , *encoged text*, y delimitadores. Si es conveniente codificar más texto del que cabe en una *encoged-word* de 75 caracteres, puede usarse múltiples *encoded-word* (separados por CRLF SPACE).

Diferencia entre Q-encoding y quoted-printable

Los códigos ASCII para el signo de interrogación ("?") Y signo igual ("=") no podrán ser representadas directamente, ya que se utilizan

como delimitadores del encoded-word. El código ASCII reservado para el espacio no puede ser representado directamente porque puede ocasionar que intérpretes antiguos dividan, de forma no deseada, el encoded-word. Para hacer la codificación más pequeña y fácil de leer, el símbolo *subrayado* (_) se utiliza en lugar del espacio, creando el efecto colateral que este símbolo no se pueda representar directamente. El uso de encoded-word en ciertas partes de los encabezados impone otras restricciones sobre cuáles caracteres pueden o no ser representados directamente.

Por ejemplo:

```
S Subject: =?iso-8859-1?Q?=A1Hola,_se=F1or!?=
```

Es interpretado como:

```
Subject: ¡Hola, señor!
```

El formato encoded-word no se utiliza para los nombres de los encabezados (por ejemplo Subject). Estos nombres de encabezados son siempre en Inglés. Cuando se lee el mensaje con un cliente de correo en otro idioma que no sea Inglés, los nombres de los encabezados son traducidos por el cliente.

Mensajes de varias

Un mensaje MIME multiparte contiene una frontera en el encabezado "Content-type:"; esta frontera, que no puede aparecer en ninguna de las partes, es ubicada entre cada una de ellas, y al inicio y al final del cuerpo del mensaje, como se muestra a continuación:

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

This is a multi-part message in MIME format.
--frontera
Content-type: text/plain

Este es el cuerpo del mensaje
--frontera
Content-type: application/octet-stream
Content-transfer-encoding: base64

PGh0bWw+CiAgPGhlYWQ+CiAgPC9oZWFkPgogIDxib2R5PgogICAgPHA+RXN0ZSBlcyBlbCBjdWVy
cG8gZGVsIG1lbnNhamU8L3A+CiAgPC9ib2R5Pgo8L2h0bWw+Cic=\
--frontera--
```

Cada parte consiste de su propio encabezado de contenido (cero o más campos de encabezados Content-) y un cuerpo. El contenido multiparte puede estar anidado. El encabezado content-transfer-encoding de un tipo multiparte tiene siempre que ser "7bit", "8bit" o "binary" para evitar las complicaciones impuestas con la presencia de múltiples niveles de decodificación. El bloque multiparte, como un todo, no tiene especificación acerca del conjunto de caracteres (charset); los caracteres no ASCII en los encabezados de la parte son manejados a través de Encoded-Word, y los cuerpos de las partes pueden tener conjuntos de caracteres especificados si aplicase para su tipo de contenido.

Notas:

- Antes de la primera frontera hay un área que es ignorada por clientes de correo electrónico que soportan MIME. Esta área es generalmente usada para poner un mensaje para usuarios de clientes viejos que no soporten MIME.
- No es hasta el momento de enviar el mensaje que el cliente de correo escoge una cadena de caracteres para usar en la frontera entre las partes, esto permite buscar una cadena de texto que no coincida con ninguna porción del cuerpo de ninguna de las partes. Esto típicamente es implementado usando una cadena larga generada aleatoriamente.
- La última frontera debe tener dos guiones al final.

Subtipos de multipartes

El estándar MIME define varios subtipos para mensajes multiparte, estos especifican la naturaleza de la parte del mensaje y su relación con otras partes. El subtipo es especificado en el encabezado "Content-type" para todo el mensaje. Por ejemplo, un mensaje MIME multiparte que usa el subtipo *digest* tendrá un "Content-Type": "multipart/digest".

La RFC inicialmente define 4 subtipos: mixed, digest, alternate y parallel. Una aplicación que cumpla mínimamente el estándar debe soportar al menos mixed y digest; el resto de los subtipos son opcionales. Otras RFCs definen subtipos adicionales como: signed y form-data.

Los subtipos más utilizados son:

Mixed

Multipart/mixed es usado para enviar mensajes o archivos con diferentes encabezados "Content-Type" ya sea en línea o como adjuntos. Si se envían imágenes u otros archivos fácilmente legibles, la mayoría de los clientes de correo electrónico las mostrarán como parte del mensaje (a menos que se especifique de manera diferente el encabezado "Content-disposition"). De otra manera serán ofrecidos como adjuntos. El content-type implícito para cada parte es "text/plain".

Definido en el RFC 2046, Sección 5.1.3

Digest

Multipart/digest es una forma simple de enviar múltiples mensajes de texto. El content-type implícito para cada parte es "message/rfc822".

Definido en el RFC 2046, Sección 5.1.5

Message

Una parte del mensaje / rfc822 contiene un mensaje de correo electrónico, incluyendo cualquier cabecera. Esto se utiliza para digest así como para el reenvío de correo electrónico.

Definido en el RFC 2046.

Alternative

El subtipo multipart/alternative indica que cada parte es una versión "alternativa" del mismo contenido (o similar), cada una en formatos diferentes denotados por su encabezado "Content-Type". Los formatos son ordenados atendiendo a cuan fieles son al original, con el menos fiel al inicio. Los sistemas pueden escoger la "mejor" representación que ellos son capaces de procesar; en general esta será la última parte que el sistema entiende, a menos que otros factores puedan afectar este comportamiento.

Dado que es poco probable que un cliente quiera enviar una versión que es poco fiel a la versión en texto plano, esta estructura ubica la versión en texto plano (si existe) primero. Esto facilita la tarea de leer los mensajes para los usuarios de clientes que no entienden mensajes multipartes.

Lo que ocurre más comúnmente es usar multipart/alternative para mensajes con dos partes, una como texto plano (text/plain) y una HTML (text/html). La parte en texto plano provee compatibilidad con clientes viejos que no son capaces de entender otros formatos, mientras que la parte HTML permite usar formato de texto y enlaces. Muchos clientes de correo ofrecen al usuario la posibilidad de preferir texto plano sobre HTML; esto es un ejemplo de como los factores locales pueden afectar en cómo una aplicación elige la "mejor" parte del mensaje para mostrar.

Aunque se pretende que cada parte represente el mismo contenido, esto no es requerido. Algunos filtros antispam examinan únicamente la parte text/plain de un mensaje porque es más fácil de analizar que las partes text/html. Pero los spammers al notar esto, comenzaron a crear mensajes con una parte text/plain que aparenta ser inocua e incluyen la propaganda en la parte text/html. Los mantenedores de programas anti-spam han modificado sus filtros, penalizando a los mensajes con textos muy diferentes en un mensaje multipart/alternative.

Definido en el RFC 2046, Sección 5.1.4

Related

El subtipo multipart/related es usado para indicar que las partes del mensaje no deben ser consideradas individualmente sino como agregados de un todo. El mensaje consiste de una parte raíz (implícitamente la primera) que hace referencia a otras partes, las que a su vez pueden hacer referencia a otras partes. Las partes son comúnmente referenciadas por el encabezado: "Content-ID". La sintaxis de la referencia no está especificada sino que está dictada por la codificación o el protocolo usado en la parte que contiene la referencia.

Un uso común de este subtipo es para enviar páginas web completas con imágenes en un único mensaje. La parte raíz contendría el documento HTML, que usaría etiquetas HTML para imágenes, para referirse a las imágenes almacenadas en partes subsiguientes.

Definido en el RFC 2387

Report

Multipart/report es un tipo de mensaje que contiene datos formateados para que un servidor de correo lo interprete. Está entre un text/plain (o algún otro tipo de contenido fácilmente legible) y un message/delivery-status, que contiene los datos formateados para el servidor de correo para leer.

Definido en el RFC 6522

Signed

El subtipo multipart/signed es usado para adjuntar una firma digital al mensaje. Esta tiene dos partes, una parte *cuerpo* y una parte *firma*. La parte del cuerpo completa, incluyendo los encabezados MIME, es usada para crear la parte de la firma. Existen muchos tipos de firmas, como “application / pgp-signature” (RFC 3156) y “application / x-pkcs7-signature” (S / MIME).

Definido en el RFC 1847, sección 2.1

Encrypted

Un mensaje multipart/encrypted tiene dos partes. La primera contiene información de control que es necesaria para descifrar la segunda parte: octeto corriente de aplicación. De manera similar a los mensajes firmados, hay diferentes implementaciones que se identifican por sus tipos de contenido independientes para la parte de control. Los tipos más comunes son "application / pgp-encriptado" (RFC 3156) y "application / pkcs7-mime" (S / MIME).

Definido en el RFC 1847, sección 2.2

Form-data

Como su nombre lo indica, multipart/form-data es usada para expresar valores enviados a través de un formulario. Originalmente definido como parte de HTML4.0, es mayormente utilizado para enviar archivos vía HTTP.

Definido en el RFC 2388

Mixed-Replace (Experimental)

El tipo de contenido multipart/x-mixed-replace fue desarrollado como parte de una tecnología para emular server push y streaming através de HTTP.

> **Server push:** describe un estilo de comunicaciones sobre Internet donde la petición de una transacción se origina en el servidor

Todas las partes de un mensaje mixed-replace poseen el mismo significado semántico. Sin embargo, cada parte invalida - "reemplaza" - a la parte previa tan pronto como es recibida completamente. Los clientes deben procesar la parte individual al momento de su llegada y no deben esperar a que termine el mensaje completo.

Desarrollado originalmente por Netscape, aún es soportado por Mozilla, Firefox, Safari (pero no en Safari para iPhone) y Opera, pero tradicionalmente ignorada por Microsoft. Es comúnmente usado en cámaras IP como el tipo MIME para MJPEG.

Byteranges

El multipart / ByteRange se utiliza para representar intervalos de bytes no contiguas de un único mensaje. Es utilizado por HTTP cuando un servidor devuelve varios intervalos de bytes y se define en el RFC 2616 .

==========

Simple Mail Transfer Protocol (SMTP)
(*transferencia simple de correo electrónico*)

Es un protocolo de red utilizado para el intercambio de mensajes de correo electrónico Fue definido en el estandar **RFC 2821.**
Opera en los servicios de correo electrónico. Sin embargo, este protocolo posee algunas limitaciones en cuanto a la recepción de mensajes en el servidor de destino (cola de mensajes recibidos). Como alternativa a esta limitación se asocia normalmente a este protocolo con otros, como el POP o IMAP, otorgando a SMTP la tarea específica de enviar correo, y recibirlos empleando (POP O IMAP).
En 1982 se diseñó el primer sistema para intercambiar correos electrónicos en ARPANET(red de computadora, creada para el departamento de defensas de estados unidos), en los estandares **RFC 821** y **RFC 822.** La comunicación entre el cliente y el servidor consiste en líneas de texto compuestas por caracteres ASCII(es código de caracteres basado en el alfabeto latino).

Modelo de procesamiento de correo

El correo electrónico es presentado por un cliente de correo (MUA, *agente de usuario de correo*) a un servidor de correo (MSA, *agente de sumisión de correo*). MSA entrega el correo a su agente de transferencia postal mejor conocido como el MTA (Mail Transfer Agent, *Agente de Transferencia de Correo*). En algunas ocasiones, estos dos agentes son casos diferentes aunque hay que destacar que provienen del mismo software de donde fueron lanzados sólo que presentan opciones diferentes dentro de la misma máquina.
El procesamiento local que se presenta puede ser realizado en una sola máquina o partido entre varias aplicaciones, aquí SMTP es usado para la transferencia de mensajes internamente, con cada uno de los hosts configurados para usar la siguiente aplicación como un anfitrión. Para lograr la localización del servidor objetivo, el MTA divisorio tiene que usar el sistema de nombre de dominio (DNS) para lograr la búsqueda del registro interno de cambiado de correo conocido como registro MX para la esfera del recipiente (la parte de la dirección a la derecha). Es en ese instante cuando el registro de MX devuelto contiene el nombre del anfitrión objetivo, el MTA se une al servidor de cambio. Una vez que MX acepta el mensaje entrante, este a su vez se lo da a un agente de entrega de correo (MDA) para luego ser llevado a la entrega de correo local. El MDA, además de entregar mensajes es también capaz de salvar mensajes en un buzón de formato, y la recepción de correo puede ser realizada usando muchas computadoras.

Puertos

Los administradores de servidor pueden elegir si los clientes utilizan TCP puerto 25 (SMTP) o el puerto

587 (Presentación) para retransmitir el correo saliente a una inicial del servidor de correo. Las especificaciones y muchos servidores soportan ambos. Aunque algunos servidores soportan el puerto 465 para el legado SMTP seguro en violación de las especificaciones, es preferible utilizar los puertos estándar y comandos ESMTP estándar de acuerdo con RFC 3207, si se debe utilizar una sesión segura entre el cliente y el servidor.
Algunos servidores están configurados para rechazar toda la retransmisión en el puerto 25, pero los usuarios válidos de autenticación en el puerto 587 pueden retransmitir correo a cualquier dirección válida.
Algunos proveedores de servicios de Internet interceptan el puerto 25, volviendo a dirigir el tráfico a su propio servidor SMTP, independientemente de la dirección de destino. Esto significa que no es posible para sus usuarios acceder a un servidor SMTP fuera de la red, a través del puerto 25.
Algunos servidores SMTP soportan el acceso autenticado en otro puerto que no sea 587 o 25 para permitir a los usuarios conectarse a ellos, incluso si el puerto 25 está bloqueado, pero 587 es el puerto estándar y ampliamente apoyada por los usuarios enviar correo nuevo. Los puertos 25 y 587 se utilizan para proporcionar la conectividad del cliente con el servicio de transporte en la parte delantera de la función de servidor de acceso de cliente (CAS). Los puertos 25, 465 y 475 son utilizados por el servicio de transporte de buzón de correo. Sin embargo, cuando la función de buzón se combina con la función de CAS en un único servidor, el puerto 2525 se utiliza por la función de buzón de SMTP desde el servicio de transporte de extremo delantero del CAS, CAS, mientras que continúa para utilizar el puerto 25. Puerto 465 es utilizado por el servicio de transporte de buzón de correo para recibir las conexiones de cliente proxy de la función CAS. Puerto 475 es utilizado por la función de buzón para comunicarse directamente con otras funciones de buzón, la transferencia de correo entre el servicio de envío de transporte de buzón de correo y el servicio de entrega de transporte buzón.

Descripción del Protocolo

SMTP es un protocolo orientado a la conexión basado en texto, en el que un remitente de correo se comunica con un receptor de correo electrónico mediante la emisión de secuencias de comandos y el suministro de los datos necesarios en un canal de flujo de datos ordenado fiable, normalmente un protocolo de control de transmisión de conexión (TCP). Una sesión SMTP consiste en comandos originados por un cliente SMTP (el agente de inicio, emisor o transmisor) y las respuestas correspondientes del SMTP del servidor (el agente de escucha, o receptor) para que la sesión se abra y se intercambian los parámetros de la sesión. Una sesión puede incluir cero o más transacciones SMTP. Una transacción de SMTP se compone de tres secuencias de comando / respuesta (véase el ejemplo a continuación).

Ellos son:

- **MAIL**: comando para establecer la dirección de retorno, también conocido como Return-Path, remitente o sobre. Esta es la dirección para mensajes de despedida.
- **RCPT**: comando, para establecer un destinatario de este mensaje. Este mandato puede emitirse varias veces, una para cada destinatario. Estas direcciones son también parte de la envolvente.
- **DATA**: para enviar el mensaje de texto. Este es el contenido del mensaje, en lugar de su envoltura. Se compone de una cabecera de mensaje y el cuerpo del mensaje separado por una línea en blanco. DATA es en realidad un grupo de comandos, y el servidor responde dos veces: una vez para el comando de datos adecuada, para reconocer que está listo para recibir el texto, y la segunda vez después de la secuencia final de los datos, para aceptar o rechazar todo el mensaje.

Resumen simple del funcionamiento del protocolo SMTP

- Cuando un cliente establece una conexión con el servidor SMTP, espera a que éste envíe un mensaje **“220 Service ready”** o **“421 Service non available”**

- Se envía un **HELO** desde el cliente. Con ello el servidor se identifica. Esto puede usarse para comprobar si se conectó con el servidor SMTP correcto.

- El cliente comienza la transacción del correo con la orden **MAIL FROM**. Como argumento de esta orden se puede pasar la dirección de correo al que el servidor notificará cualquier fallo en el envío del correo (Por ejemplo, **MAIL FROM:<fuente@host0>**). Luego si el servidor comprueba que el origen es válido, el servidor responde **“250 OK”**.

- Ya le hemos dicho al servidor que queremos mandar un correo, ahora hay que comunicarle a quien. La orden para esto es **RCPT TO:<destino@host>**. Se pueden mandar tantas órdenes RCPT como destinatarios del correo queramos. Por cada destinatario, el servidor contestará **“250 OK”** o bien **“550 No such user here”**, si no encuentra al destinatario.

- Una vez enviados todos los RCPT, el cliente envía una orden **DATA** para indicar que a continuación se envían los contenidos del mensaje. El servidor responde **“354 Start mail input, end with <CRLF>.<CRLF>”** Esto indica al cliente como ha de notificar el fin del mensaje.

- Ahora el cliente envía el cuerpo del mensaje, línea a línea. Una vez finalizado, se termina con un <CRLF>.<CRLF> (la última línea será un punto), a lo que el servidor contestará **“250 OK”**, o un mensaje de error apropiado.

- Tras el envío, el cliente, si no tiene que enviar más correos, con la orden **QUIT** corta la conexión. También puede usar la orden **TURN**, con lo que el cliente pasa a ser el servidor, y el servidor se convierte en cliente. Finalmente, si tiene más mensajes que enviar, repite el proceso hasta completarlos.

Puede que el servidor SMTP soporte las extensiones definidas en el RFC 1651, en este caso, la orden HELO puede ser sustituida por la orden EHLO, con lo que el servidor contestará con una lista de las extensiones admitidas. Si el servidor no soporta las extensiones, contestará con un mensaje "500 Syntax error, command unrecognized".

En el ejemplo pueden verse las órdenes básicas de SMTP:

- HELO, para abrir una sesión con el servidor
- MAIL FROM, para indicar quien envía el mensaje
- RCPT TO, para indicar el destinatario del mensaje
- DATA, para indicar el comienzo del mensaje, éste finalizará cuando haya una línea únicamente con un punto.
- QUIT, para cerrar la sesión
- RSET Aborta la transacción en curso y borra todos los registros.
- SEND Inicia una transacción en la cual el mensaje se entrega a una terminal.
- SOML El mensaje se entrega a un terminal o a un buzón.
- SAML El mensaje se entrega a un terminal y a un buzón.
- VRFY Solicita al servidor la verificación de todo un argumento.
- EXPN Solicita al servidor la confirmación del argumento.
- HELP Permite solicitar información sobre un comando.
- NOOP Se emplea para reiniciar los temporizadores.
- TURN Solicita al servidor que intercambien los papeles.

De los tres dígitos del código numérico, el primero indica la categoría de la respuesta, estando definidas las siguientes categorías:

- 2XX, la operación solicitada mediante el comando anterior ha sido concluida con éxito
- 3XX, la orden ha sido aceptada, pero el servidor esta pendiente de que el cliente le envíe nuevos datos para terminar la operación
- 4XX, para una respuesta de error, pero se espera a que se repita la instrucción
- 5XX, para indicar una condición de error permanente, por lo que no debe repetirse la orden

Una vez que el servidor recibe el mensaje finalizado con un punto puede almacenarlo si es para un destinatario que pertenece a su dominio, o bien retransmitirlo a otro servidor para que finalmente llegue a un servidor del dominio del receptor.

Ejemplo de una comunicación SMTP

En primer lugar se ha de establecer una conexión entre el emisor (cliente) y el receptor (servidor). Esto puede hacerse automáticamente con un programa cliente de correo o mediante un cliente telnet.

En el siguiente ejemplo se muestra una conexión típica. Se nombra con la letra C al cliente y con S al servidor.

```
S: 220 Servidor SMTP
C: HELO miequipo.midominio.com
S: 250 Hello, please to meet you
C: MAIL FROM: <yo@midominio.com>
S: 250 Ok
C: RCPT TO: <destinatario@sudominio.com>
S: 250 Ok
C: DATA
S: 354 End data with <CR><LF>.<CR><LF>
C: Subject: Campo de asunto
C: From: yo@midominio.com
C: To: destinatario@sudominio.com
C:
C: Hola,
C: Esto es una prueba.
C: Hasta luego.
C:
C: .
S: 250 Ok: queued as 12345
C: quit
S: 221 Bye
```

Formato del mensaje

Como se muestra en el ejemplo anterior, el mensaje es enviado por el cliente después de que éste manda la orden DATA al servidor. El mensaje está compuesto por dos partes:

- **Cabecera**: en el ejemplo las tres primeras líneas del mensaje son la cabecera. En ellas se usan unas palabras clave para definir los campos del mensaje. Estos campos ayudan a los clientes de correo a organizarlos y mostrarlos. Los más típicos son *subject* (asunto), *from* (emisor) y *to* (receptor). Éstos dos últimos campos no hay que confundirlos con las órdenes MAIL FROM y RCPT TO, que pertenecen al protocolo, pero no al formato del mensaje.
- **Cuerpo del mensaje**: es el mensaje propiamente dicho. En el SMTP básico está compuesto únicamente por texto, y finalizado con una línea en la que el único carácter es un punto.

SMTP vs Recuperación de correo

El protocolo de transferencia de correo simple (SMTP) solo se encarga de entregar el mensaje. En un ambiente común el mensaje es enviado a un servidor de correo de salto siguiente a medida que llega a su destino. El correo se enlaza basado en el servidor de destino. Otros protocolos como el protocolo de oficina de correos (POP) y el protocolo de acceso a mensaje de internet (IMAP) su estructura es para usuarios individuales, recuperación de mensajes, gestión de buzones de correo. SMTP usa una función, el procesamiento de colas de correo en un servidor remoto, permite que un servidor de correo de forma intermitente conectado a mandar mensajes desde un servidor remoto. El IMAP y el POP son protocolos inadecuados para la retransmisión de correo de máquinas de forma intermitente-conectados, sino que están diseñados para funcionar después de la entrega final.

Inicio remoto de mensaje en cola

Es una característica de SMTP que permite a un host remoto para iniciar el procesamiento de la cola de correo en el servidor por lo que puede recibir mensajes destinados a ella mediante el envío del comando TURN. Esta característica se considera insegura pero usando el comando ETRN en la extensión RFC 1985 funciona de forma más segura.

Petición de Reenvío de Correo Bajo Demanda (ODMR)

On-Demand Mail Relay (ODMR por sus siglas en Inglés) es una extensión de SMTP estandarizada en la RFC 2645 que permite que el correo electrónico sea transmitido al receptor después de que él ha sido aprobado. Usa la orden de SMTP ampliada ATRN, disponible para la direcciones de IP dinámicas. El cliente publica EHLO y órdenes de AUTH de servicios ODMR de correo, ODMR comienza a actuar como un cliente SMTP y comienza a enviar todos los mensajes dirigidos a un cliente usando el protocolo SMTP, al iniciar sesión el cortafuegos o el servidor pueden bloquear la sesión entrante debido a IP dinámicas. Sólo el servidor ODMR, el proveedor del servicio, debe escuchar las sesiones SMTP en una dirección de IP fija.

Internacionalización

Muchos usuarios cuyo lenguaje base no es el latín han tenido dificultades con el requisito de correo electrónico en América. RFC 6531 fue creado para resolver ese problema, proporcionando características de internacionalización de SMTP, la extensión SMTPUTF8. RFC 6531 proporciona soporte para caracteres de varios bytes y no para ASCII en las direcciones de correo electrónico. El soporte del internacionalización actualmente es limitada pero hay un gran interés en la ampliación de el RFC 6531. RFC en países como en china, que tiene una gran base de usuarios en América.

Correo saliente con SMTP

Un cliente de correo electrónico tiene que saber la dirección IP de su servidor SMTP inicial y esto tiene que ser dado como parte de su configuración (usualmente dada como un nombre DNS). Este servidor enviará mensajes salientes en nombre del usuario.

Restricción de acceso y salida al servidor de correo

En un ambiente de servidores, los administradores deben tomar medidas de control en donde los

servidores estén disponibles para los clientes. Esto permite implementar seguridad frente a posibles amenazas. Anteriormente, la mayoría de los sistemas imponían restricciones de uso de acuerdo a la ubicación del cliente, sólo estaba permitido su uso por aquellos clientes cuya dirección IP es una de las controladas por los administradores del servidor. Los servidores SMTP modernos se caracterizan por ofrecer un sistema alternativo, el cual requiere de una autenticación mediante credenciales por parte de los clientes antes de permitir el acceso.

Restringir el acceso por ubicación

Mediante este sistema, el servidor SMTP relativo al ISP no permitirá el acceso de los usuarios que están fuera de la red del ISP. Específicamente, el servidor solo puede permitir el acceso de aquellos usuarios cuya dirección IP fue proporcionada por el ISP, lo cual es equivalente a exigir que estén conectados a internet mediante el mismo ISP. Un usuario móvil suele estar a menudo en una red distinta a la normal de su ISP, y luego descubrir que el envío de correo electrónico falla porque la elección del servidor SMTP configurado ya no es accesible. Este sistema tiene distintas variaciones, por ejemplo, el servidor SMTP de la organización sólo puede proporcionar servicio a los usuarios en la misma red, esto se hace cumplir mediante cortafuegos para bloquear el acceso de los usuarios en general a través de Internet. O puede que el servicio realice comprobaciones de alcance en la dirección IP del cliente. Estos métodos son utilizados normalmente por empresas e instituciones, como las universidades que proporcionan un servidor SMTP para el correo saliente solo para su uso interno dentro de la organización. Sin embargo, la mayoría de estos organismos utilizan ahora métodos de autenticación de cliente, tal como se describe a continuación. Al restringir el acceso a determinadas direcciones IP, los administradores de servidores pueden reconocer fácilmente la dirección IP de cualquier agresor. Como esta representa una dirección significativa para ellos, los administradores pueden hacer frente a la máquina o usuario sospechoso. Cuando un usuario es móvil, y puede utilizar diferentes proveedores para conectarse a internet, este tipo de restricción de uso es costoso, y la alteración de la configuración perteneciente a la dirección de correo electrónico del servidor SMTP saliente resulta ser poco práctica. Es altamente deseable poder utilizar la información de configuración del cliente de correo electrónico que no necesita cambiar.

Seguridad y spam

Una de las limitaciones del SMTP original es que no facilita métodos de autenticación a los emisores, así que se definió la extensión SMTP-AUTH en RFC 2554.

A pesar de esto, el spam es aún el mayor problema. No se cree que las extensiones sean una forma práctica para prevenirlo. Internet Mail 2000 es una de las propuestas para reemplazarlo.

Diferentes metodologías han aparecido para combatir el spam. Entre ellas destacan DKIM, Sender Policy Framework (SPF) y desde el 2012 Domain-based Message Authentication, Reporting and Conformance (DMARC).

==========

Protocolos:

POP3 (Post Office Protocol):

- En informática se utiliza POP3 en clientes locales de correo para obtener los mensajes de correo electrónico almacenados en un servidor remoto.
- Protocolo de nivel de aplicación en el modelo OSI.
- POP3 esta diseñado para recibir correo, no para enviarlo, le permite a los usuarios con conexiones intermitentes o muy lentas, descargar su correo electrónico mientras tienen conexión y revisarlo posteriormente incluso estando desconectados.
- Al igual que otros viejos protocolos de Internet, POP3 utilizaba un mecanismo de firmado sin cifrado. La transmisión de contraseñas en texto plano aun se da. En la actualidad POP3 cuenta con diversos métodos de autenticación que ofrecen una diversa gama de niveles de protección contra los accesos ilegales al buzón de correo de los usuarios. Uno de estos es APOP, el cual utiliza MD5 para evitar los ataques de contraseñas. (Message-Digest Algorithm 5 es un algoritmo de reducción criptográfico de 128 bits).

Puertos:
Un servidor POP3 escucha en el conocido puerto 110 TCP. Comunicación encriptada puede ser bien pedida posterior a la iniciación de la misma usando STLS (STARTTLS) si es soportado o por POP3S el cual se conecta con el servidor usando TLS (Transport Layer Security) o SSL (Secure Sockets Layer) por el puerto 995 TCP.

Historia:
- POP1 fue especificado en 1984 (RFC 918) POP2 en el año 1985 (RFC 937).
- POP3 originalmente especificado en 1988 (RFC 1081). Su actual especificación en RFC 1939, actualizado con un mecanismo de extensión RFC 2449 y un mecanismo de autenticación en RFC 1734.
- La especificación original de POP3 solo soportaba el mecanismo usuario/contraseña de forma no encriptado. Actualmente existen diferentes métodos de autenticación para proveer distintos niveles de protección para los posibles accesos ilegales para un cliente de correo electrónico.
- Algunos de ellos son SASL para quienes posean la extensión AUTH, APOP es un protocolo del tipo challenge/response que usa la función de Hash MD5 para evitar ataques de terceros.
- Algunos clientes que utilizan APOP son Mozilla Thunderbird, Opera Mail, Eudora Mail, etc.
- POP4 existe de forma informal, agregando algunos detalles al actual POP3, pero no ha progresado desde 2003.

Extensión:
- STARTTLS: Permite el uso de TLS (Transport Layer Security) o SSL (Secure Sockets Layer) los cuales son protocolos de aplicación con la función de encriptado, diseñados para mantener seguridad en las comunicaciones, para comunicarse usando el comando STLS en el puerto standard de POP3 y no en la alternativa.
- SDPS: Demon Internet introduce la extensión de POP3 que permite múltiples cuentas por dominio el cual fue conocido como Standar Dial-up POP3 Service. Para acceder a cada cuenta, el usuario incluye el hostname.

Comunicación POP3:

- Para establecer una conexión a un servidor POP, el cliente de correo abre una conexión TCP en el puerto 110 del servidor.

- Cuando la conexión se ha establecido, el servidor POP enviá al cliente POP y después las dos maquinas se envían entre si ordenes y respuestas que se especifican en el protocolo.
- Como parte de esta comunicación, al cliente POP se le pide autenticación, donde el nombre de usuario y contraseña se enviá al servidor.
- Si la autenticación es correcta, el cliente pasa a estado de transacción, donde se pueden realizar ordenes LIST, RETR, DELE, para mostrar, descargar y eliminar mensajes del servidor respectivamente.
- Los mensajes definidos para su eliminación no se quitan realmente del servidor hasta que el cliente envía la orden QUIT para finalizar sesión.
- En ese momento el servidor pasa al estado de actualización, fase en la que se eliminan los mensajes marcados, y se limpian todos los recursos restantes de la sesión.

Comandos (Utilización):

USER <nombre>	Identificación de usuario (solo se realiza una vez)
PASS <password>	Envía la clave del servidor
STAT	Da el número de mensajes no borrados en el buzón y su longitud total
LIST	Muestra todos los mensajes no borrados con su longitud
RETR <número>	Solicita el envío del mensaje especificando el numero (no se borra del buzón)
TOP <número> <linea>	Muestra la cabecera y el número de lineas requerido del mensaje especificando el número
DELE <número>	Borra el mensaje especificando el numero
RSET	Recupera los mensajes borrados (en la conexión actual)
UIDL <número>	Devuelve una cadena identificatoria del mensaje persistente a través de las sesiones. Si no se especifica <número> se devuelve una lista con los números de mensajes y su cadena identificatoria de los mensajes no borrados
QUIT	Salir

Ejemplo De Diálogo:

Diálogo usando APOP

S: <wait for connection on TCP port 110>
C: <open connection>
S: +OK POP3 server ready <1896.697170952@dbc.mtview.ca.us>
C: APOP mrose c4c9334bac560ecc979e58001b3e22fb
S: +OK mrose's maildrop has 2 messages (320 octets)
C: STAT
S: +OK 2 320
C: LIST
S: +OK 2 messages (320 octets)
S: 1 120
S: 2 200
S: .
C: RETR 1
S: +OK 120 octets
S: <the POP3 server sends message 1>
S: .
C: DELE 1
S: +OK message 1 deleted
C: RETR 2

S: +OK 200 octets
S: <the POP3 server sends message 2>
S: .
C: DELE 2
S: +OK message 2 deleted
C: QUIT
S: +OK dewey POP3 server signing off (maildrop empty)
C: <close connection>
S: <wait for next connection>

Servidores POP3 sin el uso de APOP esperan que el cliente inicie sesión con USER y PASS

C: USER mrose
S: +OK User accepted
C: PASS tanstaaf
S: +OK Pass accepted

Ventajas:
- Si administras múltiples correos electrónicos desde un mismo cliente, podrás revisarlos desde un mismo lugar.
- Los mensajes son almacenados localmente por lo que siempre podrás acceder a ellos independientemente de si estás conectado o no.
- La apertura de archivos adjuntos funciona generalmente más rápido ya que éstos son descargados simultáneamente con el contenido del mensaje.
- Gracias a que los mensajes son descargados en tu PC, el espacio en disco esta limitado por el tamaño de tu disco rígido y no por la cuenta de hosting que poseas.

Desventajas:
- Si recibimos archivos o contenidos que puedan infectar nuestra PC, es mas probable que nos afecte debido a que el contenido es 100% local.
- Si tenemos un problema con nuestra computadora y perdemos toda la información, a menos que realicemos copias de seguridad, perderemos toda nuestra información.
- Si tenemos configurado nuestro cliente de correo para dejar una copia en el servidor, a medida que tu cuenta crezca en tamaño, tardaremos más en revisar nuevos correos ya que sera necesario revisar que correos ya fueron descargados y cuales no.

==========

IMAP (Internet Message Access Protocol) :

Internet Message Access Protocol (IMAP, Protocolo de acceso a mensajes de internet), es un protocolo de aplicación que permite el acceso a mensajes almacenados en un servidor de Internet. Mediante IMAP se puede tener acceso al correo electrónico desde cualquier equipo que tenga una conexión a Internet. IMAP tiene varias ventajas sobre POP (otro protocolo empleado para obtener correos desde un servidor). Por ejemplo, es posible especificar en IMAP carpetas del lado del servidor. Por otro lado, es más complejo que POP ya que permite visualizar los mensajes de manera remota y no descargando los mensajes como lo hace POP.
IMAP y POP3 (Post Office Protocol versión 3) son los dos protocolos que prevalecen en la obtención de correo electrónico. Todos los servidores y clientes de correo electrónico están virtualmente soportados por ambos, aunque en algunos casos hay algunas interfaces específicas del fabricante típicamente propietarias. Por ejemplo, los protocolos propietarios utilizados entre el cliente Microsoft Outlook y su servidor Microsoft Exchange Server o el cliente Lotus Notes de IBM y el servidor Domino. Sin embargo, estos productos también soportan interoperabilidad con IMAP y POP3 con otros clientes y servidores. La versión actual de IMAP, IMAP versión 4 revisión 1 (IMAP4rev1), está definida por el RFC 3501.

IMAP fue diseñado como una moderna alternativa a POP por Mark Crispin en el año 1986. Fundamentalmente, los dos protocolos les permiten a los clientes de correo acceder a los mensajes almacenados en un servidor de correo.
Ya sea empleando POP3 o IMAP4 para obtener los mensajes, los clientes utilizan SMTP para enviar mensajes. Los clientes de correo electrónico son comúnmente denominados clientes POP o IMAP, pero en ambos casos se utiliza SMTP.

Cuadro comparativo POP3 vs IMAP

Protocolo	Ventajas	Desventajas
IMAP4	• Trabaja en modo de conexión permanente, por lo que avisa inmediatamente de la llegada de nuevo correo • Transmite solo las cabeceras por lo que el usuario puede decidir su borrado inmediato • La bajada del mensaje se produce solo cuando el usuario quiere leerlo • El almacenamiento local del mensaje es opcional (una opción del cliente de correo) • Gestiona carpetas, plantillas y borradores en el servidor • El almacenamiento de mensajes y carpetas en el servidor permite su uso desde múltiples dispositivos y de forma simultánea • Permite la búsqueda de mensajes por medio de palabras claves • Los mensajes se pueden etiquetar. El marcado queda en el servidor • Se pueden crear carpetas compartidas con otros usuarios (depende del servidor)	• No todos los clientes de correo soporta la extensión IMAP IDLE (aviso de nuevos correos) • Necesita una transacción por cada correo que se quiera leer • Hay un retraso en la aparición del mensaje en la pantalla del usuario, mientras se descarga • Si se pierde la conexión, no se podrá ver el mensaje salvo si el cliente de correo lo haya almacenado en local • Las carpetas, plantillas y borradores no podrán ser leídos usando POP (excepto la Bandeja de entrada)
POP3	• Los correos aparecen inmediatamente porque quedan residentes en el dispositivo (una vez descargados)	• Sólo se conecta periódicamente cada X minutos para buscar por nuevo correo • La conexión periódica provoca un aumento del tráfico y un retraso en la respuesta del cliente (esperar la descarga completa) • En cada conexión, se baja todos los correos nuevos, vayan a ser después leídos o no • Los correos ocupan espacio local del dispositivo • Por defecto, elimina los mensajes del servidor, haciendo imposible el acceso a ellos desde otro dispositivo

Modos de operación conectado y desconectado

Al utilizar POP, los clientes se conectan brevemente al servidor de correo, solamente el tiempo que les tome descargar los nuevos mensajes. Al utilizar IMAP4, los clientes permanecen conectados el tiempo que su interfaz permanezca activa y descargan los mensajes bajo demanda. Esta manera de trabajar de IMAP puede dar tiempos de respuesta más rápidos para usuarios que tienen una gran cantidad de mensajes o mensajes grandes.

Conexión de múltiples clientes de forma simultánea a un mismo buzón

El protocolo POP requiere que el cliente conectado sea el único conectado al buzón. En contraste, el protocolo IMAP4 permite accesos simultáneos a múltiples clientes y proporciona ciertos mecanismos a los clientes para que se detecten los cambios hechos a un buzón por parte de otro cliente conectado. Vea, por ejemplo, el RFC3501, sección 5.2 que, específicamente, dice "accesos simultáneos al mismo buzón por múltiples agentes".

Acceso a partes MIME de los mensajes y obtención parcial

Casi todo el correo electrónico de Internet se transmite en formato MIME, permitiendo a los mensajes tener una estructura en árbol, donde las hojas son de una variedad de tipos de contenidos en una única parte, y los nodos que no son hojas son de una variedad de tipos multiparte. El protocolo IMAP4 permite a los clientes obtener separadamente cualquier parte MIME individual y también obtener porciones de las partes individuales o los mensajes completos. Estos mecanismos permiten a los clientes obtener la porción de texto de un mensaje sin tener que descargar los archivos adjuntos o en flujos.

Información del estado del mensaje

A través de la utilización de señales definidas en el protocolo IMAP4, los clientes pueden seguir el estado del mensaje: por ejemplo, si el mensaje ha sido leído o no, respondido o eliminado. Estas señales se almacenan en el servidor, de manera que varios clientes conectados al mismo buzón, en diferentes momentos, pueden detectar los cambios hechos por otros clientes. POP no ofrece ningún mecanismo para los clientes para que almacenen esta información de estado en el servidor, así que si un usuario accede a un buzón con dos clientes POP diferentes (en tiempos diferentes), la información de estado -como si el mensaje ha sido accedido- no puede sincronizarse entre los clientes. El protocolo IMAP4 soporta tanto el sistema predefinido de señales del sistema, como el de palabras claves definidas. Las señales del sistema indican la información de estado, tales como si el mensaje ha sido leído. Las palabras clave, que no se soporta en todos los servidores IMAP, permite que a los mensajes se les den una o más etiquetas cuyo significado depende del cliente. Las palabras clave IMAP no deben confundirse con etiquetas propietarias de los servicios de correo basados en web, que alguna web se traducen en carpetas IMAP por los correspondientes servidores propietarios.

Múltiples buzones en el servidor

Los clientes de IMAP4 pueden crear, renombrar y/o eliminar buzones (por lo general presentado como carpetas al usuario) en el servidor, y copiar mensajes entre buzones. El soporte para múltiples buzones también le permite a los servidores proporcionar acceso a carpetas públicas y compartidas. La *IMAP4 Access Control List (ACL) Extension* (RFC 4314) se puede usar para regular los derechos de acceso.

Búsquedas de parte del servidor

IMAP4 proporciona un mecanismo para que los clientes pidan al servidor que busque mensajes de acuerdo a una cierta variedad de criterios. Este mecanismo evita que los clientes descarguen todos los mensajes de su buzón de correo, agilizando de esta manera las búsquedas.

Mecanismo de extensión

Como reflejo de la experiencia en los protocolos anteriores de Internet, IMAP define un mecanismo explícito mediante el cual se puede ser extender. Se han propuesto muchas extensiones de IMAP4 y son de uso común. IMAP2bis no tiene ese mecanismo, y POP tiene definido uno en RFC 2449.
Un ejemplo de extensión es el IMAP IDLE, que sirve para que el servidor avise al cliente cuando ha llegado un nuevo mensaje de correo y éstos se sincronicen. Sin esta extensión, para realizar la misma tarea, el cliente debería contactar periódicamente al servidor para ver si hay mensajes nuevos.