

Correu electrònic i missatgeria

Eduard Canet i Ricart

Serveis de xarxa i Internet

Índex

Introducció	5
Resultats d'aprenentatge	7
1 Instal·lació i administració del servei de correu electrònic	9
1.1 Protocols de correu electrònic	9
1.1.1 Format dels missatges	12
1.1.2 Bústies de correu	15
1.1.3 Funcionament de l'SMTP	16
1.1.4 MIME	20
1.2 Instal·lació d'un servidor	24
1.2.1 Instal·lació de l'aplicació servidor	24
1.2.2 Usos indeguts del servidor de correu	26
1.3 Accés remot al correu	27
1.3.1 Servei POP	27
1.3.2 Servei IMAP	31
1.3.3 Clients de correu	37
1.4 Correu encriptat i signat	38
1.4.1 Seguretat en el correu	39
1.4.2 Propietats de seguretat	40
1.4.3 Implementació de seguretat	42
1.5 Servidor de correu segur	45
2 Instal·lació i administració de serveis de missatgeria instantània, notícies i llistes de distribució	47
2.1 Missatgeria instantània	47
2.1.1 Funcionament de la missatgeria	50
2.1.2 Clients de missatgeria	55
2.2 Llistes de distribució	61
2.2.1 Creació d'un gestor de llistes	62
2.2.2 Creació i utilització de llistes	69
2.3 Servei de notícies	78
2.3.1 Descripció general	79

Introducció

En aquesta unitat del mòdul *Serveis de xarxa i Internet* es presenta de manera molt exhaustiva el funcionament del correu electrònic o *e-mail*. Es mostren tots els elements que intervenen en aquest tipus de comunicacions; es detalla el funcionament dels protocols SMTP, POP i IMAP; s'estudia el format dels missatges, i també els missatges MIME. A part del correu electrònic, s'expliquen altres formes de comunicació com les llistes de correu i de notícies o la missatgeria instantània, que són molt populars.

En molts aspectes el correu electrònic imita el funcionament del correu postal. És un sistema distribuït que permet als usuaris enviar missatges a un destinatari final. El correu electrònic ha tingut una gran evolució des dels primers sistemes, que únicament permetien intercanviar missatges de text ASCII, fins als correus electrònics amb continguts multimèdia d'avui en dia.

L'apartat "Instal·lació i administració del servei de correu electrònic" mostra el funcionament i els elements que participen en l'enviament de missatges de correu. En el servei de correu es diferencia molt clarament entre el mecanisme de transport dels missatges i els missatges. S'analitzen els mecanismes dels protocols SMTP, POP i IMAP per tal de comprendre amb profunditat el seu funcionament, i s'aprendrà a instal·lar i configurar programari capaç d'implementar aquests protocols.

Un altre aspecte tractat és el de la seguretat dels missatges i de la comunicació. S'exposen els mecanismes per xifrar i autenticar missatges, proporcionant les prestacions de xifrat, autenticació, integritat i no-repudi. Per obtenir aquestes propietats cal familiaritzar-se amb els certificats digitals. Les comunicacions segures entre dos interlocutors es basen principalment en la utilització de capes de transport segur com SSL i TLS. S'exposarà com configurar servidors SMTP, POP i IMAP per permetre connexions segures a través dels certificats digitals.

En l'apartat "Instal·lació i administració de serveis de missatgeria instantània, notícies i llistes de distribució" es descriu el funcionament i la implementació i configuració de cada un d'aquests tipus de serveis. Es realitza una comparativa entre cada servei per entendre bé la seva funcionalitat, el seu propòsit. També s'explica com instal·lar i configurar el servei de missatgeria instantània i com utilitzar clients gràfics i de text. El servei de missatgeria es basa en el protocol XMPP, també anomenat JABBER, del qual se n'expliquen les característiques més rellevants.

Així mateix, es mostra com instal·lar i configurar el servei de llistes de distribució o llistes de correu i els mecanismes que s'han anat utilitzant al llarg del temps (des d'àlies fins als servidors actuals). Es classifiquen les llistes segons la seva funcionalitat o tipus i l'accés que permeten. També es donen exemples de funcionament i administració.

La immensa popularitat del WWW i la utilització de llistes de correu han fet perdre importància al servei de notícies. Basat en el protocol NNTP, el servei de notícies permetia publicar articles com qui publica anuncis en un tauler. És un mecanisme per publicar i compartir informació sense que calgui indicar els destinataris.

La majoria d'usuaris no han utilitzat mai el servei de notícies, però segurament no n'hi ha cap que no hagi xatejat en algun moment o altre. Els sistemes de missatgeria instantània, començant pel mòbil i acabant pels televisors actuals que permeten connexions amb Skype, són àmpliament utilitzats. Permeten una comunicació immediata entre usuaris identificats, fins i tot amb àudio i vídeo. Es mostra com realitzar una apropiada gestió dels comptes d'usuari i verificar l'accés als serveis de missatgeria instantània, notícies i llistes de distribució. També s'explica com elaborar documentació relativa als procediments tractats al llarg de l'apartat.

Per assolir els objectius d'aquesta unitat és molt recomanable que feu les activitats i els exercicis d'autoavaluació proposats en cadascun dels apartats de la unitat.

Resultats d'aprenentatge

En finalitzar aquesta unitat, l'estudiant:

1. Administra servidors de correu electrònic, aplicant criteris de configuració i garantint la seguretat del servei.

- Descriu els diferents protocols que intervenen en l'enviament i recollida del correu electrònic.
- Instal·la i configura un servidor de correu electrònic.
- Crea comptes d'usuari i en verifica l'accés.
- Estableix i aplica mètodes per impedir usos indeguts del servidor de correu electrònic.
- Instal·la serveis per permetre la recollida remota del correu en les bústies d'usuari.
- Usa clients de correu electrònic per enviar i rebre correu des dels comptes creats en el servidor.
- Utilitza la signatura digital i el correu xifrat.
- Configura el servidor de correu com a servei segur.
- Elabora documentació relativa a la instal·lació, configuració i recomanacions d'utilització del servei.

2. Administra serveis de missatgeria instantània, notícies i llistes de distribució, verificant i assegurant l'accés dels usuaris.

- Descriu els serveis de missatgeria instantània, notícies i llistes de distribució.
- Instal·la i configura el servei de missatgeria instantània.
- Utilitza clients gràfics i de text de missatgeria instantània.
- Instal·la i configura el servei de notícies.
- Instal·la i configura el servei de llistes de distribució.
- Determina el tipus de llista i els modes d'accés permesos.
- Crea comptes d'usuari i en verifica l'accés als serveis de missatgeria instantània, notícies i llistes de distribució.
- Elabora documentació relativa a la instal·lació, configuració i recomanacions d'ús dels serveis de missatgeria instantània, notícies i llistes de distribució.

1. Instal·lació i administració del servei de correu electrònic

En molts aspectes el correu electrònic imita el funcionament del correu postal. És un sistema distribuït que permet als usuaris enviar missatges a un destinatari final. El correu electrònic ha evolucionat molt dels primers sistemes que únicament permetien intercanviar missatges de text ASCII als correus electrònics amb continguts multimèdia d'avui en dia.

En el servei de correu es diferencia molt clarament entre el mecanisme de transport dels missatges i els missatges. El mecanisme de transport dels missatges és el protocol SMTP, i és independent del format i el contingut del missatge. Els missatges originals eren en text pla ASCII de 7 bits, però actualment en un missatge es permet tot tipus de contingut. Això és possible gràcies als tipus MIME, que descriuen i codifiquen els missatges.

El mateix disseny del sistema de correu ha evolucionat a mesura que ha avançat la tecnologia a internet. En el model bàsic de transport per SMTP s'exigeix que el receptor disposi de connexió permanent i que es connecti al servidor de correu localment per tal de consultar-lo. Quan els usuaris tenen accés a internet mitjançant un ISP (proveïdor de serveis d'internet) volen poder baixar tot el correu de cop i examinar-lo un cop tancada la connexió (per no pagar connexió telefònica). El protocol POP proporciona el mecanisme per descarregar del servidor de correu els missatges de l'usuari.

Amb la popularització d'internet s'abaixen els costos de connexió. L'usuari s'acostuma a baixar el correu des de llocs diferents, però usant el POP té l'inconvenient que el correu li queda repartit per diverses màquines. Cal un mecanisme que permeti accedir i gestionar el correu i les bústies directament en el servidor. El protocol IMAP ho fa possible.

Tot això ha canviat amb la popularització d'internet a tots els nivells i usant tota mena de dispositius. La major part del correu electrònic funciona actualment per web, gràcies a proveïdors *webmail* com els coneguts Gmail o Yahoo.

1.1 Protocols de correu electrònic

El correu electrònic és un dels primers serveis que es va utilitzar a les xarxes i un dels més populars a internet. Ha evolucionat molt des dels primers sistemes, que podien intercanviar únicament missatges de text ASCII (7 bits), fins als portals web usats avui dia per milions d'usuaris per enviar-se continguts multimèdia.

El 1982 es van desenvolupar els estàndards que defineixen el correu electrònic. Es descriuen en els documents RFC 821, que explica el protocol de transmissió,

ASCII

Acrònim amb el qual es coneix l'American Standard Code for Information Interchange, el codi de caràcters establert com a estàndard americà el 1963 i que va esdevenir *de facto* l'estàndard mundial.

Per obtenir més informació sobre l'especificació del protocol SMTP en els RFC 821, 822, 2821 i 2822, aneu a la secció "Adreces d'interès" del web del mòdul.

Per conèixer més detalls de l'estàndard MIME, consulteu la secció "Els tipus MIME".

i RFC 822, que descriu el format dels missatges. Aquests dos estàndards han evolucionat i actualment s'utilitzen els documents RFC 2821 i RFC 2822. A més, per permetre missatges multimèdia s'ha definit l'estàndard MIME en el document RFC 2231.

L'especificació original distingeix molt clarament entre el mecanisme de transport dels missatges i els missatges. El mecanisme de transport dels missatges és el **protocol SMTP**.

El protocol SMTP (*Simple Mail Transport Protocol* o **protocol simple de transport de correu**) és independent del format i el contingut del missatge. El missatge es compon del **sobre** (*envelope*) i el **contingut**, format per les capçaleres i el cos del missatge.

El correu electrònic és un sistema distribuït que permet als usuaris enviar missatges a un destinatari final. Hi intervenen diversos agents:

Ambigüitat dels agents

Els agents que intervenen en el sistema de correu electrònic fan sovint més d'un paper, fet que provoca ambigüitat en la seva definició.

Servidor SMTP

Sovint el programari de servidor SMTP (per exemple, Sendmail) fa tant la funció de client (emissor de missatges) com de servidor (receptor de missatges).

- **MUA** (*Mail User Agent* o **agent de correu d'usuari**). L'usuari utilitza un MUA per redactar, rebre i manipular correus electrònics. Un MUA és un programari que permet aquestes capacitats, que poden ser aplicacions en línia d'ordres (per exemple, l'ordre *mail* d'Unix), aplicacions de text (per exemple, Mutt o Pine), interfícies gràfiques (com Thunderbird) i portals de correu web (com Gmail o Yahoo). L'usuari interactua usualment amb un MUA. En el cas d'enviar un correu, el MUA lliura el missatge al sistema de transport (SMTP) per fer-lo arribar al destinatari. En el cas de recepció de correu, el MUA obté el missatge d'una bústia de correu (on l'ha dipositat l'SMTP) i el mostra a l'usuari.
- **MTA** (*Mail Transport Agent* o **agent de transport de correu**). L'agent de transport de correu és l'encarregat de transportar els missatges al destinatari indicat. Aquesta tasca la fa el protocol **SMTP**. L'MTA rep el missatge d'un MUA i s'encarrega del seu transport fins al destinatari final. Generalment realitzen la funció de client/servidor o emissor/receptor al mateix temps. La funció que es realitza en cada cas és:
 - **MTA client SMTP (emissor)**. S'anomena client de correu o emissor (segons l'arquitectura client/servidor) el servidor SMTP (fixeu-vos en l'ambigüitat) que envia el correu al destinatari. És qui envia el correu utilitzant el protocol SMTP. Estableix les connexions amb els servidors/receptors SMTP.
 - **MTA servidor SMTP (receptor)**. S'anomena servidor de correu o receptor el programari de servidor SMTP que rep els missatges de correu entrant i els lliura a la bústia del destinatari si es tracta d'un lliurament local, o els reenvia a un altre servidor SMTP si va destinat a un sistema remot. El fet que un receptor MTA rebí un correu no significa que el missatge hagi arribat al destinatari final.
- **MDA** (*Mail Delivery Agent* o **agent de lliurament de correu**). Un altre element en l'estructura de correu és el MDA. És l'encarregat de fer el

lliurament final del missatge a la bústia del destinatari. En el procés pot realitzar diverses accions segons un conjunt de regles *.forward* definibles per l'usuari. Un exemple d'MDA és el programa procmail, que permet filtrar els missatges entrants per posar-los en una bústia o una altra, esborrar-los, marcar-los com a correu brossa (*spam*), fer-ne còpies, reenviar-los a altres bústies i a altres destinataris... Usualment, en sistemes de correu que no disposen d'MDA és el mateix MTA el que diposita el missatge a la bústia del destinatari final.

@ (arrova)

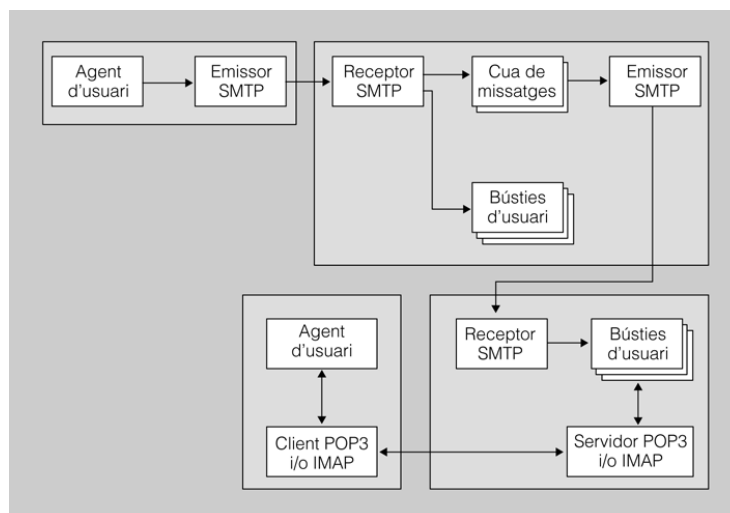
Significa "at" en anglès o "a" en català. Usualment la composició d'una adreça de correu electrònic es descriu com a *usuari@màquina* (usuari tal a la màquina qual), però el nom del domini no correspon necessàriament al nom de la màquina. És més correcte dir *usuari@domini*.

Per exemple, *pere@gmail.com* indica el compte de correu d'en Pere en la màquina *gmail.com*. Però, de fet, no hi ha cap màquina que es digui així, sinó que és el compte de correu d'en Pere en el domini *gmail.com*. En realitat, Gmail té diverses màquines que responen a aquest nom de domini.

També hi ha altres conceptes que intervenen en el sistema de correu electrònic:

- **Adreça de correu.** Els usuaris que volen utilitzar el sistema de correu electrònic han de disposar d'una bústia en un servidor de correu. Els missatges s'adrecen utilitzant la coneguda nomenclatura **usuari@domini-servidor-correu**, que es llegeix com a "compte de correu de l'usuari tal en el domini qual". Així, per exemple, a un usuari amb un compte de correu de nom pere en el domini *ioc.cat* se li poden adreçar missatges a l'adreça *pere@ioc.cat*.
- **Bústia de correu o mailbox.** Els usuaris tenen bústies en un servidor de correu. Quan el servidor de correu MTA rep un missatge destinat a un usuari amb compte de correu en el mateix servidor, el diposita a la bústia de correu corresponent (si no hi ha un MDA pel mig). Fixeu-vos que dipositar el missatge en la bústia de l'usuari no garanteix que l'usuari el llegeixi. Cal un altre pas, que és la recuperació del missatge de la bústia per part de l'usuari. Aquest pas es realitza des d'un MUA i sovint empra protocols com **POP** o **IMAP**, fora de l'abast de les explicacions del correu SMTP.
- **Llista de correu i àlies.** Els àlies i les llistes de correu es tradueixen en adreces de comptes de correu. Si la llista de correu es gestiona localment, el MUA local l'expandirà en el conjunt d'adreces d'usuari corresponents i enviarà el correu electrònic a cada una. Si la llista d'usuaris és remota, s'envia el correu electrònic al sistema remot i serà l'MTA remot el que l'expandirà i enviarà un correu electrònic a cada membre de la llista. Fixeu-vos que si la llista conté usuaris d'altres dominis de correu (on sigui del món), farà arribar una còpia a aquests usuaris.

El model funcional del protocol SMTP, que mostra els elements que intervenen en una comunicació d'aquest tipus, es pot veure a figura 1.1.

FIGURA 1.1. Model funcional del protocol SMTP**Exemple de funcionament del correu electrònic**

El correu web té un funcionament similar al correu electrònic.

Per exemple, un usuari de Gmail utilitza com a MUA el web de Gmail per enviar un missatge a un usuari de Yahoo. Gmail transfereix el missatge per SMTP al servidor de correu de Yahoo i el missatge es diposita a la bústia del destinatari. Aquest, quan li sembla, consulta el correu utilitzant el lloc web de Yahoo com a MUA.

Exemples de programes
que implementen SMTP:
Sendmail, Exim, Postfix, MS
Exchange Server.

Per tant, vist en conjunt, un MUA (Thunderbird, per exemple) permet a l'usuari crear un correu electrònic i lliurar-lo a l'MTA del sistema (per exemple, Sendmail) perquè el faci arribar al destinatari final. Usant el protocol SMTP, l'MTA s'encarrega de fer el lliurament al sistema final (pot ser amb una connexió directa o encaminant-se a través de diversos MTA) i el missatge es diposita en la bústia de correu del receptor. En aquest procés de deixar el missatge en la bústia del receptor hi pot haver un MDA que "postprocessi" el missatge o ho pot fer l'MTA directament. Quan ho considera oportú, el receptor recupera els missatges de la bústia utilitzant un MUA i un mecanisme d'accés adequat (per exemple, amb Thunderbird, usant el protocol POP o IMAP).

1.1.1 Format dels missatges

El protocol SMTP s'encarrega del transport de missatges de correu amb independència del format i del contingut. Els missatges es componen de diferents elements que es descriuen en l'especificació SMTP (corresponent al document RFC 821) i en l'especificació pròpia dels missatges d'internet (document RFC 822).

El missatge es desglossa en els elements següents:

- **Sobre** o *envelope*. Com passa en el correu postal, per fer arribar un missatge cal un sobre en el qual s'indiquin el destinatari i el remitent. L'especificació de l'SMTP (document RFC 821) descriu com a sobre el conjunt de dades necessàries per al transport del missatge (emissor, receptor, prioritat, nivell

Els camps FROM i RCPT del sobre no porten dos punts (:) mentre que les capçaleres From i To del contingut sí que en porten.

Alguns exemples de capçaleres: From:, To:, Date:, Subject:

de seguretat...). Generalment, el sobre consta tan sols dels camps FROM (emissor) i RCPT (receptor). L'MTA utilitza el sobre per encaminar el missatge. De fet, la separació entre sobre i contingut és confusa i l'MTA obté les dades del sobre a partir de les capçaleres del contingut.

- **Contingut.** El contingut d'un missatge és el que està descrit en el document RFC 822 *Standard for ARPA Internet Text Messages* (estàndard per als missatges de text d'internet). Tot contingut consta d'un conjunt de capçaleres (*headers*), una línia en blanc i un cos (*body*) del missatge:
 - **Capçaleres (*headers*).** El missatge conté capçaleres del tipus *clau: valor*, cada una en una línia independent.
 - **Línia en blanc.** Les capçaleres se separen del cos del missatge amb una línia en blanc.
 - **Cos del missatge.** El cos del missatge conté el missatge que es vol fer arribar al receptor. L'especificació inicial només permet text ASCII de 7 bits (sense símbols internacionals). El cos del missatge acaba amb una línia que conté a l'inici únicament un punt.

Les capçaleres descrites en l'especificació inicial tenen com a objectiu descriure clarament l'emissor i el receptor o receptors del missatge i la data, i permetre identificar el missatge (ID únic), entre d'altres.

En l'exemple següent podeu veure els elements que formen el contingut i les **capçaleres usals** d'un correu electrònic:

```
1 Received: by 10.100.195.12 with HTTP; Sun, 11 May 2008 10:11:38 -0700 (PDT)
2 Message-ID: <7b4e8fcc0805111011g83da6b0rdbb4f63409024720@mail.gmail.com>
3 Date: Sun, 11 May 2008 19:11:38 +0200
4 From: "Pere Puig" <puig@correu.fp-oberta.org>
5 To: ppuig@correu.fp-oberta.org
6 Subject: =?ISO-8859-1?Q?Exemple_de_missatge_de_correu_amb_capçaleres
7 Delivered-To: ppuig@correu.fp-oberta.org
8
9 Hola,
10 Això és un exemple de missatge de correu.
11 Conté les capçaleres usals.
12 S'ha generat des del web de Gmail i s'envia també a Gmail.
13
14 Pere
```

Aquestes són algunes de les capçaleres estàndard:

- **From:** indica l'adreça de correu de l'emissor del missatge.
- **Sender:** indica l'adreça de qui ha enviat el missatge. No s'utilitza si qui ha enviat el missatge és l'emissor del missatge. Serveix per diferenciar entre qui envia el missatge físicament i en nom de qui ho fa.
- **To i Cc:** serveixen per indicar els destinataris del missatge. La idea original era posar un destinatari en el *To* i la resta en el *Cc*, però amb la utilització dels MUA actuals i la utilització de llistes d'usuaris generalment es posen tots els destinataris en el *To*.

- **Bcc:** prové de l'anglès *blind carbon copy* o còpia oculta. S'indiquen els destinataris que han de rebre el missatge però que no han d'aparèixer a la llista de destinataris. Serveix per evitar que els altres destinataris sàpiguen qui n'ha rebut una còpia.
- **Reply-to:** indica l'adreça de retorn del missatge al remitent. L'emissor pot voler que si el missatge es retorna o es respon, l'adreça a la qual es dirigeix la resposta sigui diferent de la indicada en el camp *From*. És útil per concentrar les respostes en un compte de correu quan l'emissor en té més d'un.
- **Received:** cada MTA que processa un missatge afegeix una entrada de tipus *Received* en el missatge. És una manera de realitzar el seguiment o la traçabilitat dels MTA pels quals ha passat el missatge. La informació afegida descriu l'emissor (*From*) i el receptor (*By*), el mecanisme físic (*Via*), l'identificador del missatge (*ID*) i la data i hora (*Date*).
- **Date:** indica la data i hora en què s'ha generat el missatge. L'hi afegeix el primer MTA que rep el missatge del MUA.
- **Message-ID:** és l'identificador únic del missatge. Cada missatge s'ha de poder referenciar de manera única a tot el món. Això permet que les respostes indiquin a quin missatge es refereixen. S'utilitzen els noms de domini i un identificador numèric únic que genera l'MTA que rep el missatge per enviar.
- **Subject:** descriu el propòsit del missatge o assumpte. És un petit text explicatiu.
- **In-reply-to:** quan un missatge és una resposta a un missatge anterior, aquest camp indica a quin missatge original fa referència.
- **Keywords:** és la llista separada per comes de paraules clau descriptives del missatge.
- **Comments:** és el text de comentari del missatge que no interfereix en el contingut.
- **References:** quan un missatge fa referència a altres missatges anteriors, es pot indicar mitjançant aquesta capçalera.
- **Encrypted:** indica el tipus d'encriptació que s'ha utilitzat per al missatge. L'especificació del format dels missatges de correu (descrita en el document RFC 822) no indica cap tipus d'encriptació, simplement reserva una capçalera per indicar-ne el tipus.
- **Return-path:** identifica el camí de retorn cap a l'origen. Aquesta informació l'ha de posar l'MTA receptor. Actualment està en desús, de manera que normalment conté l'adreça de l'emissor.
- **X-*<userDefined>*:** els usuaris poden crear les pròpies capçaleres amb el nom que vulguin però començant per X-. D'aquesta manera s'assegura que si apareixen noves capçaleres oficials en el futur, no xocaran amb capçaleres definides pels usuaris.

1.1.2 Bústies de correu

Les **bústies de correu** són el sistema que permet l'emmagatzematge dels correus electrònics. Estan ubicades en l'espai de disc del servidor que allotja el servei de correu electrònic. Els dos principals formats són el mbox i el Maildir.

El format tradicional d'UNIX per a les bústies de correu és l'**mbox**. Les bústies dels usuaris s'emmagatzemen normalment a la carpeta `/var/mail` o `/var/spool/mail` (a vegades una és un enllaç simbòlic de l'altra). En aquesta carpeta hi ha un sol fitxer per usuari que conté tots els seus correus, concatenats un darrere l'altre. Aquest fet fa que sigui molt ràpid fer una cerca en la correspondència d'un usuari, tot i que aquest sistema no és gens escalable. Un dels grans problemes són els bloquejos, ja que per afegir un nou correu cal bloquejar el fitxer i això el fa inaccessible per a la cerca. El RFC 4155 dona informació sobre aquest tipus de bústia, però en cap cas es tracta d'unes especificacions.

La principal innovació del format **Maildir** és que té un fitxer per a cada correu i estan estructurats en carpetes, i això fa que no es produeixin bloquejos (només a nivell d'un sol correu). Els tres principals directoris són:

- *new*: és la carpeta on van a parar els correus nous. Un cop llegits passen a la carpeta *cur*.
- *cur*: és on es troben els correus que ja no són nous.
- *tmp*: és una carpeta temporal que, entre d'altres coses, serveix per rebre correctament els missatges abans de ser moguts a la carpeta *new*.

Aquest sistema és més estable, ràpid i escalable que el tradicional mbox. I el problema de corrupció de fitxer afecten notablement menys a aquest sistema ja que tots els correus estan separats.

No obstant, també hi ha altres formats de bústia, més minoritaris. Aquests són:

- **dbx**: format de bústia d'alt rendiment per a Dovecot. Té dues variants:
 - **sdbx** (*single dbx*): semblant al Maildir, un missatge per correu.
 - **mdbx** (*multidbx*): múltiples correus per fitxers, però no a l'estil de mbox.
- **mbx/mix**: format de bústia del programari UW-IMAP de la Universitat de Washington. L'anterior format era el mbx, que s'ha substituït pel mix, que permet un millor rendiment.
- **Mailstore**: format de bústia originari del programari Exim. Consta de dos fitxers per correu amb les extensions `.env` i `.msg`, un per al sobre (*envelope*) i l'altre per al missatge.
- **Pst** (*Personal Storage Table*): format obert propietari de Microsoft. És utilitzat per Microsoft Exchange Server i Microsoft Outlook.

1.1.3 Funcionament de l'SMTP

'Push'

Es diu que l'SMTP és un protocol que fa *push* (lliura), però no *pull* (agafa). Els usuaris finals han d'usar altres mecanismes per accedir remotament als seus comptes de correu.

El funcionament del protocol SMTP imita el correu postal en molts aspectes. L'SMTP és un protocol d'emmagatzemament i enviament que funciona igual que es fa amb les cartes de correu, que es lliuren en una oficina postal, d'allà a una altra, i així successivament fins arribar al destinatari final. De fet, les cartes es lliuren a la bústia del destinatari final i aquest les ha de recollir.

El **servidor SMTP** és una aplicació distribuïda que permet enviar missatges electrònics. Utilitza el protocol de transport TCP i el port 25.

MX

En la base de dades d'un servidor DNS, els equips que fan de servidors de correu d'un domini s'identifiquen per les entrades tipus MX. Si un domini no disposa d'entrades MX, s'utilitza l'amfitrió que defineix el domini.

En l'esquema original en què es va desenvolupar l'SMTP, una organització disposa d'un servidor SMTP (un MTA) que rep correu electrònic de fora de l'organització i el diposita en les bústies de correu locals del servidor. També recull el correu intern de l'organització i l'envia fora.

Cada organització disposa d'una o més màquines encarregades de gestionar el correu. Així, quan s'envia un correu electrònic a l'usuari `pere@ioc.cat`, cal que l'organització o domini `ioc.cat` disposi de màquines que fan la funció de servidors de correu. ¿Com trobarà l'SMTP a quin servidor de correu ha de lliurar els correus electrònics destinats a un domini? Utilitzant el protocol DNS (*domain name system*, sistema de noms de domini) i fent una consulta de tipus MX obtindrà la màquina o màquines que fan la funció de servidors de correu del domini consultat.

El client SMTP o emissor estableix una connexió TCP amb el port 25 del servidor SMTP o receptor. En una mateixa connexió l'emissor pot enviar un o més missatges al receptor. Si el mateix missatge va destinat a diversos receptors del sistema final, el missatge s'envia un sol cop i l'MTA receptor el replica a cada destinatari.

El client SMTP o emissor disposa d'una cua de missatges per enviar i una llista de destinataris per a cada missatge. Els destinataris poden ser en destinacions diferents (evidentment) i, per tant, li caldrà connectar-se als diferents servidors de destinació per fer-los arribar els missatges.

Llistes negres de servidors de correu

Els servidors de correu que accepten correus electrònics de tots els clients a totes les destinacions són inclosos en llistes negres perquè poden ser generadors de correu brossa.

Correu brossa o 'spam'

Correu brossa, correu no desitjat o no sol·licitat. És un correu que es rep insistentment i que bombardeja les bústies dels usuaris de manera mecànica.

Quan un destinatari no és accessible, el missatge es pot tornar a posar a la cua de missatges pendents d'enviar o es pot descartar (segurament després de diversos intents infructuosos) tot notificant-ne l'emissor.

Avui en dia el servidor de correu pot ser a qualsevol lloc del món i no cal que cada organització en tingui un. Es pot utilitzar el del proveïdor ISP o el de qualsevol servei extern de correu (per exemple, Google permet externalitzar el correu a empreses tot mantenint el domini propi de l'empresa). Això significa que el servidor SMTP ha de verificar si accepta o no peticions d'enviar correu d'un client. Es pot verificar el client mitjançant l'adreça IP o mitjançant altres mecanismes d'autenticació i seguretat. Evidentment, disposar d'un servidor SMTP que accepta peticions de clients sense verificar qui són és una porta oberta a permetre correu

brossa. Normalment els servidors SMTP restringeixen qui pot fer ús del servei (quins clients) i a quines destinacions.

Un cop un servidor SMTP accepta un correu electrònic per fer-ne el lliurament (d'un MUA com Thunderbird, per exemple) tot validant que accepta rebre correus electrònics d'aquest client, estableix una connexió TCP al port 25 del servidor SMTP destinatari (ha obtingut l'adreça IP fent la resolució DNS de la part del domini de l'adreça de correu).

Ordres/respostes SMTP

L'emissor sempre porta el control de la comunicació i inicia la connexió amb el receptor. El diàleg consisteix en un intercanvi d'ordres i respostes que segueixen les especificacions de Telnet:

S'entén per CRLF una línia en blanc. Ve de l'anglès *Carriage Return - Line Feed* (retorn de carro - salt de línia).

- **Ordres.** Són codis de quatre caràcters (HELO, MAIL, DATA...) i arguments opcionals separats per espais i acabats amb <CRLF>. Per a cada ordre es rep una resposta del receptor.
- **Respostes.** Són codis numèrics de tres dígit, un espai i un missatge descriptiu que pot variar segons la implementació.

Un diàleg bàsic entre emissor i receptor SMTP podria ser el següent:

- **HELO <domini> / EHLO <domini>.** Un cop connectat, l'emissor s'ha d'identificar amb l'ordre HELO i indicar el domini al qual es connecta. Actualment, els servidors SMTP utilitzen extensions i l'ordre preferida per identificar-se és EHLO (significa extended HELO).
- **MAIL FROM: <emissor>.** Identifica l'emissor del missatge i genera la capçalera *From* del missatge. El receptor comprova que l'emissor sigui un usuari vàlid, és a dir, que accepti missatges d'aquest origen. Si no el pot validar, envia una resposta denegant-li la comunicació. Els equips amb el *relay* configurat per permetre enviar missatges de tothom són els principals generadors de correu brossa.
- **RCPT TO: <destinatari>.** Indica el destinatari del missatge. Aquesta ordre es pot repetir tantes vegades com destinataris tingui el missatge. També cal que el receptor accepti el destinatari, que pot ser un destinatari local, o que accepti fer el reenviament si és un destinatari remot. Aquesta ordre genera la capçalera *To* en el missatge.
- **DATA.** Indica que a continuació s'enviarà el missatge. Tot el que es transmet a continuació és el contingut del missatge, que finalitzarà en trobar una línia que només inclou un punt (<CRLF>). El contingut segueix les especificacions del document RFC 822; per tant, pot contenir capçaleres a l'inici, una línia en blanc a manera de separador i el cos. No es pot enviar un missatge (l'ordre DATA) fins que el receptor no ha confirmat que accepta almenys un destinatari. Això evita transmetre missatges que es descartarien en la destinació.

- **QUIT.** L'emissor envia l'ordre per indicar al receptor que vol finalitzar la comunicació. El receptor confirma la recepció i llavors tots dos poden finalitzar la transmissió.

En la secció "Annexos" del web d'aquest mòdul teniu captures dels diferents diàlegs SMTP, POP i IMAP.

En l'exemple següent podeu veure un diàleg client/servidor SMTP mitjançant ordres i respostes Telnet:

```

1 [root@host ~]# telnet www.escola.org 25
2 Trying 22.170.21.168...
3 Connected to www.escola.org.
4 Escape character is '^]'.
5 220 escola.org ESMTP Sendmail 8.13.8/8.13.8; Sat, 26 Apr 2008
6 19:56:05 +0200
7 EHLO escola.org
8 250-escola.org Hello 106.Red-71-92-14.dynamicIP.rima-tde.net
9 71.92.14.106], pleased to meet you
10 250-ENHANCEDSTATUSCODES
11 250-PIPELINING
12 250-8BITMIME
13 250-SIZE 10000000
14 250-DSN
15 250-ETRN
16 250-AUTH DIGEST-MD5 CRAM-MD5 LOGIN PLAIN
17 250-DELIVERBY
18 250 HELP
19
20 MAIL FROM: pere@xtec.cat
21 250 2.1.0 pere@xtec.cat... Sender ok
22 RCPT TO: pere@correu.escola.org
23 250 2.1.5 pere@correu.escola.org... Recipient ok
24
25 DATA
26 354 Enter mail, end with "." on a line by itself
27 Hola,
28 Aquest és un missatge de prova per enviar un
29 correu usant Telnet al servidor SMTP de l'escola.
30 S'envia una còpia a dos usuaris locals al servidor.
31 S'ha denegat fer relaying i enviar una còpia a
32 l'exterior.
33 Pere
34 .
35 250 2.0.0 m3QH5B3012660 Message accepted for delivery
36
37 QUIT
38 221 2.0.0 escola.org closing connection
39 Connection closed by foreign host.
```

Ordres SMTP

Per obtenir la llista d'ordres del protocol podeu consultar el document RFC 2821. Atès que és un servidor concret, podeu consultar les ordres que implementa amb l'ordre HELP.

Amb els camps MAIL FROM i RCPT TO, el protocol SMTP obté les dades necessàries per generar el sobre o *envelope*.

A part de les ordres bàsiques mostrades anteriorment, hi ha altres ordres en el protocol SMTP, com ara les següents:

- **RSET.** L'emissor pot interrompre l'enviament de missatges.
- **NOOP.** Aquesta ordre no fa res ('no operate'), però força el receptor a enviar una resposta afirmativa. Serveix per confirmar que la connexió encara és oberta.
- **HELP.** Fa una llista de les ordres que implementa el servidor. Els servidors SMTP no implementen necessàriament totes les ordres descrites pel protocol.

- **VERFY <destinatari>**. L'emissor pot verificar l'existència del destinatari.
- **EXPN <destinatari>**. Permet a l'emissor verificar l'existència d'una llista de correu i obtenir-ne els noms dels membres.
- **SEND, SOML, SAML**. Permeten enviar els missatges tant a les bústies de correu com als terminals.
- **TURN**. Permet intercanviar els papers entre emissor i receptor. El receptor hi ha d'estar d'acord.

Els servidors SMTP no implementen necessàriament totes les ordres, però hi ha un conjunt d'ordres mínim definit pel protocol que tot servidor SMTP ha d'implementar.

El **conjunt d'ordres mínim** que tot servidor SMTP ha d'implementar és el següent:

```
HELO <domini>
MAIL FROM: <emissor>
RCPT TO: <destinatari>
DATA
RSET
NOOP QUIT
```

El protocol SMTP permet treballar amb missatges ASCII de 8 bits i amb extensions del protocol, és a dir, afegir als servidors SMTP funcionalitats extres segons el programari de servidor utilitzat. El client pot sol·licitar al receptor la llista de les extensions que implementa i fer-li saber que les vol utilitzar. El mecanisme consisteix a fer que el client envii un **EHLO** en lloc del HELO estàndard. Si el receptor implementa extensions, respondrà afirmativament i en farà una llista; si no les implementa, respondrà negativament. Llavors l'emissor pot fer un HELO estàndard.

Les respostes es poden classificar en quatre grans grups. El primer dígit del codi numèric de tres dígits de la resposta indica el grup al qual pertany:

- **Positiva (2xx)**. L'acció que ha sol·licitat l'emissor és acceptada pel receptor. L'emissor pot fer una nova sol·licitud. Les respostes d'aquest grup comencen totes pel dígit 2. En els llistats de codi es pot observar que, per exemple, el valor 250 correspon a OK o acció realitzada correctament.
- **Intermèdia positiva (3xx)**. L'acció sol·licitada s'ha acceptat, però està suspesa pendent de rebre informació addicional que l'emissor haurà de proporcionar.
- **Negativa transitòria (4xx)**. La sol·licitud no s'ha acceptat i l'acció no s'ha realitzat, però es tracta d'un error temporal i es pot tornar a intentar més tard. L'emissor pot tornar a fer la sol·licitud més endavant.
- **Negativa pertinent (5xx)**. L'ordre no s'ha realitzat i, per tant, la sol·licitud no ha estat acceptada.

1.1.4 MIME

Els missatges de correu tenen el format definit en l’RFC 822 (actualment, RFC 2822), que únicament permet missatges de text net ASCII de 7 bits. No es permeten els caràcters accentuats, caràcters internacionals (ASCII de 8 bits) i molt menys la transferència de dades binàries com imatges, àudio, aplicacions, PDF o altres. Però tot això i molt més s’envia avui en dia per correu electrònic.

El juny del 1992 es va definir el que es coneix com a **MIME** (*Multipurpose Internet Mail Extension* o **extensió de correu d’internet per a ús múltiple**) en l’RFC 1341, que actualment ha evolucionat en els RFC 2045 i RFC 2049. El MIME utilitza missatges RFC 822, però afegint una estructura al cos del missatge i regles de codificació per a missatges no ASCII. El gran avantatge del MIME és que permet seguir utilitzant les mateixes eines de l’SMTTP que fins ara i només cal modificar els MUA perquè apliquin MIME. A l’MTA, el cos del missatge li és absolutament indiferent (per tant, pot estar codificat), ja que només utilitza el sobre per enviar el missatge i el contingut s’envia com un tot.

El MIME es basa en tres elements per permetre qualsevol tipus de contingut en un missatge de correu:

- **Capçaleres MIME.** Es creen cinc noves capçaleres de correu per definir informació del cos del missatge. No totes són obligatòries.
- **Formats de contingut.** Es defineixen diferents formats de contingut que permeten als MUA receptors interpretar el contingut de manera adequada i saber si reben un full de càlcul, un vídeo...
- **Esquemes de codificació de transferència.** Es realitza una transformació de les dades a un format manipulable per al transport SMTP (que només permet caràcters ASCII de 7 bits).

Podeu veure els components d’un missatge amb contingut MIME en l’exemple següent:

```
1 From root@tftp.server.cat Fri Jun 13 17:26:31 2012
2 Return-Path: <root@tftp.server.cat>
3 Received: from tftp.server.cat (localhost [127.0.0.1])
4   by tftp.server.cat (8.14.1/8.14.1) with ESMTP id m5DFQTH7003922
5   for <pere@tftp.server.cat>; Fri, 13 Jun 2012 17:26:30 +0200
6 Received: (from root@localhost)
7   by tftp.server.cat (8.14.1/8.14.1/Submit) id m5DFQSIq003918
8   for pere@tftp.server.cat; Fri, 13 Jun 2012 17:26:28 +0200
9 Date: Fri, 13 Jun 2012 17:26:27 +0200
10 From: root <root@tftp.server.cat>
11 To: pere@tftp.server.cat
12 Subject: missatge amb atachment
13 Message-ID: <20080613152627.GA3909@portatil.local.lan>
14 MIME-Version: 1.0
15 Content-Type: multipart/mixed; boundary="zYM0uCDKw75PZbzx"
16 Content-Disposition: inline
17 User-Agent: Mutt/1.5.17 (2007-11-01)
18 Status: 0
19
```

```
20 —zYM0uCDKw75PZbzx
21 Content-Type: text/plain; charset=us-ascii
22 Content-Disposition: inline
23
24 missatge de root a l'usuari pere
25 conté adjunt un pdf i jpeg
26 adéu!
27
28 —zYM0uCDKw75PZbzx
29 Content-Type: application/pdf
30 Content-Disposition: attachment;
31 filename="informatica_AX_ud2.pdf"
32 Content-Transfer-Encoding: base64
33 ... output suprimir (contingut del pdf codificat en base64) ...
34 —zYM0uCDKw75PZbzx
35 Content-Type: image/jpeg
36 Content-Disposition: attachment; filename="cd15_11_puerto-
37 madrin.jpg"
38 Content-Transfer-Encoding: base64
39 ... output suprimir (contingut del jpeg codificat en base64) ...
40 —zYM0uCDKw75PZbzx—
```

Capçaleres MIME

Les cinc capçaleres que defineix l'especificació MIME aporten informació del contingut del missatge.

Aquestes capçaleres són les següents:

- **MIME-version.** Identifica el tipus MIME del missatge. Si indica 1.0, es tracta d'un missatge MIME; si no, es tracta d'un missatge ASCII.
- **Content-description.** És un text que descriu el tipus de contingut. No és obligatori i no té cap funcionalitat més enllà de la merament descriptiva.
- **Content-ID.** Identifica el contingut de manera única, igual que ho fa el camp *Message-ID*.
- **Content-transfer-encoding.** És el mecanisme de codificació utilitzat en el missatge per poder-lo transmetre. El contingut que no és ASCII de 7 bits es codifica per poder ser transmès.
- **Content-type.** Descriu el tipus de contingut segons la taula de tipus MIME. Això permet a un MUA obrir l'aplicació pertinent per gestionar el contingut. Si, per exemple, el tipus és *image/jpeg*, permet al MUA saber que en pot manipular el contingut amb una aplicació de gestió d'imatges.

Tipus MIME

Es defineix un conjunt de tipus i subtipus MIME amb un esquema tipus/subtipus. Originàriament es van definir els set tipus que es descriuen a continuació, però en l'actualitat n'hi ha molts més.

- **text/native.** Text net en format ASCII de 7 bits.

- **multipart/<subtipus>**. El missatge conté múltiples parts independents. Un delimitador (o *boundary*) indica la separació de cada part. El delimitador és únic i no apareix en el cos de les parts. El delimitador es troba a l'inici i al final de cada part i comença amb dos guions. L'última part acaba amb un delimitador que comença i acaba amb dos guions. Cada part pot ser qualsevol cosa!
- **multipart/parallel**. Múltiples parts, en ordre. És a dir, les parts s'han de mostrar en el receptor en l'ordre indicat.
- **multipart/mixed**. Múltiples parts. No es defineix cap ordre.
- **multipart/alternative**. Les parts són versions alternatives del mateix contingut en ordre creixent de fidelitat. El receptor escull la més apropiada. Per exemple, un text s'envia com a text pla en una primera part i com a PDF en una segona; si el receptor no disposa de PDF podrà usar la part en text net.
- **multipart/digest**. Cada part és un missatge de correu individual. S'utilitza quan un correu electrònic conté diversos correus electrònics en el seu interior (per exemple, reenviaments).
- **message/rfc822**. El cos és un missatge de correu complet, amb capçaleres i cos. Pot ser un missatge MIME tot i que al nom hi digui "rfc822".
- **message/partial**. Permet fragmentar un missatge llarg en diferents missatges. Cada fragment ha de disposar d'un identificador, número de fragment i nombre total de fragments.
- **message/external body**. Les dades del cos del missatge no estan en el missatge sinó que cal baixar-les a part. En la capçalera Content-type es descriu el tipus de contingut i el tipus d'accés, que pot ser FTP, TFTP, anon-FTP (FTP anònim), *local-file*, AFI i *mail-server*. Per exemple, el contingut pot ser una imatge no inclosa en el missatge sinó que calgui baixar d'un servidor FTP.
- **image/jpeg**. Imatge codificada JPEG
- **image/gif**. Imatge GIF
- **video/mpeg**. Vídeo en format MPEG (*Moving Picture Experts Group*, grup d'experts d'imatges en moviment)
- **audio/basic**. Àudio en format estàndard
- **application/postscript**. Dades binàries en format PostScript. Per exemple, PDF.
- **application/octet-stream**. Dades binàries

Codificació de transferència

Les dades binàries i els caràcters internacionals (que no pertanyen al conjunt ASCII de 7 bits) no es poden enviar per correu electrònic. Per poder-ho fer, cal codificar-los en un altre format.

L'especificació MIME defineix els tipus de codificacions següents:

- **7bit.** Indica que les dades es transfereixen en ASCII de 7 bits. No es realitza cap codificació.
- **8bit.** No es realitza cap codificació i les dades es transmeten en ASCII de 8 bits. Evidentment, cal que receptor i emissor permetin la transferència a 8 bits (una extensió d'SMTP).
- **Binary.** Es transmeten les dades en binari tal com són, sense cap codificació ni control de la longitud de les línies. Si s'envien dades en binari (en cru), no es garanteix que la transmissió sigui correcta.
- **X-token.** Indica la utilització d'un esquema de codificació de transport no estàndard, un esquema propi. Emissor i receptor han de compartir aquest esquema de codificació.
- **Quoted-printable.** Quan la majoria de caràcters del missatge són imprimibles excepte una petita part, és més eficient utilitzar aquesta codificació que Base64. Aquest esquema codifica els caràcters no imprimibles amb un signe igual (=) i el codi hexadecimal del caràcter. Es garanteix que les línies tenen una longitud no superior a 36 caràcters mitjançant salts de línia reversibles.
- **Base64.** És l'esquema de codificació més usat per a la transferència d'informació binària. Converteix l'entrada en un conjunt de caràcters imprimibles i, per tant, immunes al transport per SMTP. Consta d'un conjunt de 63 caràcters imprimibles i un més de farciment ($2^6 = 64$ caràcters). Cada 24 bits de l'entrada binària (3 bytes) es codifica en quatre blocs de 6 bits ($4 * 6 = 24$ bits). A cada bloc de 6 bits li correspon un caràcter imprimible que es posa en 1 byte. Per tant, per cada 24 bits d'entrada binària, s'utilitzen 32 bits de transmissió ($4 * 1$ byte).

Base64

Per aprendre el funcionament de Base64 podeu consultar la Viquipèdia:

en.wikipedia.org/wiki/Base64

Exemple de codificació en Base64

Aquest és un petit exemple extret de la Viquipèdia on s'observa que el text "Man" original (3 bytes = 24 bits) acaba codificat en Base64 com a "TWFu" (4 bytes).

Text content M a n

ASCII 77 97 110

Bit pattern 01001101 01100001 01101110 (8 bits * byte)

Bit pattern 010011 010110 000101 101110 (divisió en blocs de 6 bits)

Index 19 22 5 46

Base64-encoded T W F u

1.2 Instal·lació d'un servidor

Per obtenir més informació sobre els servidors de correu actuals, consulteu l'activitat titulada: Quota de mercat dels servidors de correu.

Sendmail

Per conèixer els orígens i l'evolució de Sendmail us recomanem consultar l'article sobre el servei a la Viquipèdia:

en.wikipedia.org/wiki/Sendmail

Hi ha diverses aplicacions de servidor de correu en el mercat i moltíssims clients de correu de tota mena, tant en versió gràfica com d'entorn de text. Algunes d'aquestes aplicacions són de font pública i es poden baixar gratuïtament d'internet.

La majoria de sistemes GNU/Linux proporcionen l'aplicació client Mail i sovint també Mutt, que és una versió de Mail amb pantalles en mode text. Els sistemes GNU/Linux i Unix també disposen d'una aplicació servidor omnipresent anomenada Sendmail.

Quan es parla d'instal·lar el servei de correu es fa referència al procés d'instal·lació i configuració del programari del servidor. Això es fa de manera molt similar a la d'altres serveis de xarxa (com els serveis DHCP, DNS, HTTP o FTP): es tracta d'instal·lar els paquets o *tarballs* de l'aplicació servidor i fer-ne la configuració apropiada.

Per fer això cal plantejar-se els passos i reflexions següents:

- Preguntar-se i buscar l'aplicació més adient: *Quin programari proporciona aquest servei? Quines característiques té? Com es pot adquirir?*
- Observar l'estat de la xarxa actual: *El servei ja està en funcionament? Existeix ja un servidor de correu instal·lat i actiu?*
- Obtener l'aplicació que proporciona el servei de correu.
- Instal·lar l'aplicació servidor.
- Comprovar que la instal·lació s'ha fet correctament.
- Configurar el servei en el servidor i comprovar que els clients hi poden accedir.
- Comprovar que el servei funciona correctament.

L'eina utilitzada en aquest mòdul per estudiar els serveis de servidor de correu és **Sendmail**. Podeu trobar tota la informació sobre aquest servidor a www.sendmail.org.

Usualment, l'administrador acaba utilitzant l'aplicació servidor que li proporciona el seu sistema operatiu. Si utilitzeu Windows, l'empresa Microsoft disposa d'una aplicació pròpia, però en podeu trobar d'altres a internet. Igualment, si utilitzeu GNU/Linux, segurament la mateixa distribució proporciona un servidor de correu o bé n'existeix algun de clàssic provinent d'Unix. De totes maneres, en podeu obtenir d'altres a internet.

1.2.1 Instal·lació de l'aplicació servidor

Els usuaris de GNU/Linux poden buscar fàcilment per internet paquets de servidor de correu Sendmail usant eines com yum o apt-get i els repositoris de paquets

apropiats segons la distribució que utilitzin. A més, sempre es pot recórrer a Google per localitzar tot allò que faci falta.

Un cop instal·lat el programari, cal identificar què s'ha instal·lat, els paquets i el contingut. A vegades no s'instal·len paquets sinó fitxers *tarball*, el contingut dels quals també cal saber examinar. És important identificar els components instal·lats que corresponen a fitxers executables, els que corresponen a fitxers de configuració i els que corresponen a fitxers de documentació.

Tot servei instal·lat s'ha de configurar apropiadament i s'ha posar en marxa. Per tant, cal saber gestionar l'estat del servei (engegar, aturar, recarregar...) i definir l'estat que ha de tenir en els diferents *runlevels* del sistema.

En definitiva, el procediment d'instal·lació inclou usualment:

- Buscar el programari del servei (sigui en format de paquets *.deb*, *.rpm* o *.tar*) i descarregar-lo amb l'eina apropiada segons la distribució que s'utilitzi.
- Examinar el sistema per identificar el programari, els paquets, instal·lat relacionats amb el servei.
- Identificar els components del servei: fitxers executables, fitxers de configuració i fitxers de documentació.
- Consultar i establir l'estat del servei (engegar i aturar) i saber establir l'estat per defecte per a cada *runlevel*.

Verificació de l'accés als comptes de correu

Un servidor de correu realitza la funcionalitat de client i de servidor SMTP. Com a client s'encarrega d'enviar els missatges que hi ha a la cua de missatges al servidor apropiat. És a dir, si un missatge està dirigit a un usuari del domini gmail.com, s'encarrega de fer arribar el missatge a algun dels servidors de correu d'aquest domini.

Com a servidor SMTP té la funció d'escollir les peticions entrants que li fan els clients SMTP i atendre-les. Això significa “rebre” els missatges i fer els passos necessaris per fer-los arribar a la bústia del client. Si el sistema de correu no utilitza un MDA propi (ho són programes com procmail o SpamAssassin), el mateix Sendmail farà aquesta funció.

Quan actua com a **servidor SMTP** el servidor de correu també pot realitzar la funcionalitat de MDA (*Mail Delivery Agent*) i encarregar-se de deixar els missatges a la bústia de cada usuari local.

De fet, si la configuració de correu actual utilitzada és la que es crea per defecte en instal·lar Sendmail, no hi ha cap MDA específic sinó que és el mateix Sendmail el que fa aquesta funció. Això significa que també s'ha d'encarregar de crear les bústies de correu dels usuaris i de gestionar-les (si no és que ja ho fa el mateix sistema operatiu).

Si no hi ha altres agents tipus MDA en funcionament, la creació i la gestió de les **bústies d'usuari** és missió del servidor de correu.

Un **compte de correu** no és altra cosa que disposar d'una bústia de correu en el sistema.

Usuaris locals

Si es vol que un usuari local no pugui iniciar una sessió d'usuari en el sistema, se li pot assignar com a *shell* /sbin/nologin.

Tot usuari de sistemes GNU/Linux disposa d'un compte local en la màquina on té el compte d'usuari. Així, un usuari de nom pere en un amfitrió anomenat pc-jocs/ pot rebre correu a l'adreça pere@pc-jocs.

És evident que cada usuari que vol tenir un compte de correu ha de disposar d'una **bústia** pròpia en la qual el servidor ha de poder desar el correu destinat a l'usuari. L'usuari accedeix a la seva bústia per consultar el seu correu.

Des del punt de vista de la creació de bústies es fa la classificació següent:

- **Usuaris locals del sistema.** En sistemes GNU/Linux tots els usuaris del sistema disposen d'una bústia de correu local. Què cal fer perquè els usuaris d'un servidor tinguin correu local? Res. Tots els usuaris locals d'un *host* tenen una bústia pròpia i tothom s'hi pot adreçar indicant **usuari@host**. Aquest mecanisme obliga a generar comptes d'usuaris locals en el sistema per tal de poder disposar dels comptes de correu.
- **Usuaris del servei.** Hi ha servidors de correu que permeten crear comptes de correu sense necessitat de crear comptes d'usuari locals en el sistema. És a dir, es tracta d'usuaris que existeixen només per al servidor de correu però no per al sistema operatiu.

1.2.2 Usos indeguts del servidor de correu

El problema principal del correu electrònic és el correu brossa o *spam*, és a dir, el correu no desitjat. Des del punt de vista del client, convé saber filtrar el correu per detectar el correu brossa i informar-ne al servidor (generalment és un *webmail*). Des del punt de vista del servidor, cal saber filtrar el correu brossa i cal establir mecanismes per no participar en la seva difusió.

Actualment la majoria de serveis de correu del tipus *webmail* incorporen les dues prestacions següents:

- **Filtrat automàtic de correu brossa.** Gmail, per exemple, filtra els correus i intenta detectar quins són brossa i els posa directament en una carpeta amb aquest nom. Gmail aplica regles complexes de filtrat per detectar correus brossa segons el seu criteri. Els usuaris el poden ajudar informant-lo dels missatges que consideren brossa.
- **Cerca automàtica de virus.** S'aplica un antivirus als continguts que s'adjunten als fitxers. D'aquesta manera s'evita la propagació indiscriminada de continguts maliciosos.

1.3 Accés remot al correu

La manera com els usuaris accedeixen al seu correu ha anat evolucionant al mateix pas que ho ha anat fent la tecnologia. Originalment s'utilitzava simplement l'SMTP (Sendmail, per exemple) i els usuaris havien d'accedir al servidor iniciant una sessió d'usuari per consultar el correu amb eines com Mail. És a dir, els usuaris havien d'anar físicament on hi havia la màquina servidor i iniciar-hi una sessió o bé connectar-s'hi via Telnet.

Però els usuaris volien poder descarregar i enviar des de casa els missatges de correu. El protocol POP d'accés remot a les bústies de correu va proporcionar aquest servei. Els usuaris es connectaven per mòdem, es connectaven al servidor de correu i es descarregaven tot el correu de cop i aprofitaven també per enviar missatges. En aquest model, la gestió dels missatges es feia a casa, el servidor simplement els acumulava per permetre'n la descàrrega. La tecnologia d'accés a internet per mòdem implicava pagar per les trucades. Per tant, l'usuari tenia interès en baixar tot el correu, finalitzar la trucada (per no seguir pagant) i examinar tranquil·lament els missatges sense connexió a internet.

Amb l'aparició de les tarifes planes, els usuaris ja no s'han de preocupar de fer una connexió curta al servidor i poden estar connectats permanentment. El protocol IMAP d'accés a bústies remotes permet l'accés dels clients a les seves bústies realitzant totes les gestions (carpetes, etiquetat, filtrat...) en el mateix servidor. Això resol un dels problemes típics del POP, que és que baixant els missatges en màquines diferents el correu quedava repartit per diferents llocs.

Sempre que es configura un client de correu cal indicar:

- Servidor de **correu entrant**: un servidor POP o IMAP des d'on es descarreguen els missatges de l'usuari.
- Servidor de **correu sortint**: el servidor SMTP a qui cal lliurar el correu que genera l'usuari per tal que sigui enviat al destinatari.

Actualment, la majoria de clients de correu utilitzen serveis *webmail* com Gmail, Yahoo o altres. Els clients es connecten al servidor i accedeixen a la seva bústia utilitzant una interfície web. Tota la gestió del correu es fa des d'un navegador.

1.3.1 Servei POP

En el model de transport de correu SMTP s'exigeix que el receptor disposi de connexió permanent a internet. Està pensat per a correu entre organitzacions connectades a la xarxa i que disposen d'un servidor de correu que conté les bústies dels usuaris locals de l'organització. Això obliga els usuaris a treballar localment

POP3 és la versió actual del protocol POP. Aquí usem tots dos noms indistintament.

en el servidor per accedir a les seves bústies. Amb la popularització d'internet sorgeix el problema dels usuaris que hi accedeixen per ISP i que no tenen connexió permanent (per exemple, amb mòdem).

POP3 i el correu postal

El POP3 és un mecanisme similar al correu postal. El carter deixa les cartes a la nostra bústia i les recollim quan ens sembla.

S'ideja un mecanisme per a l'accés remot als comptes de correu, de manera que l'usuari es connecta quan vol, accedeix a la bústia de correu per recuperar els missatges i finalitza la connexió. POP3 i IMAP són protocols que permeten l'accés remot de clients a les bústies de correu.

POP3 (*Post Office Protocol* o **protocol d'accés simple a les bústies de correu**) és un protocol de capa d'aplicació de la pila de protocols TCP/IP (port 110) definit en l'RFC 1939. Permet a un client de correu (MUA) obtenir remotament el correu dipositat en la bústia de l'usuari en un servidor POP3.

Dispersió del correu

Si l'usuari baixa correu POP des de màquines diferents, li queda dispersat. En cada màquina queda desat localment el que s'hi ha baixat.

Normalment, l'usuari utilitza una aplicació client de POP3 (per exemple, Thunderbird) i baixa el correu del servidor POP. Els missatges que es baixen es desen a la màquina de l'usuari (localment) i s'esborren del servidor (es pot configurar si s'esborren o no). Finalment es tanca la connexió.

Funcionament del POP3

Amb el protocol POP3 el client fa una connexió TCP/IP al port 110 del servidor, baixa el correu i tanca la connexió. En aquest procés, client i servidor passen per tres estats (autorització, transacció i actualització) i s'intercanvien ordres i respostes seguint el model de diàleg de Telnet:

- **Ordres.** Són ordres de text de quatre caràcters seguides d'espais i els arguments que requereixin. Finalitzen amb un <CRLF>.
- **Respostes.** Són una cadena de caràcters que comença per **+OK** o **-ERR** més una descripció. Les respostes afirmatives comencen per **+OK**, i les d'error per **-ERR**.

Baixar missatges del servidor POP3

Alguns MUA bàsics baixen tots els missatges del servidor de cop. Si es deixen els missatges ja llegits en el servidor, es tornaran a baixar cada cop que s'hi accedeix.

Vegeu una llista de les ordres utilitzables en el protocol POP3 agrupades segons l'estat:

1. Autorització

- **USER <nomUsuari>.** El client s'identifica en el servidor POP indicant el nom d'usuari, que ha de correspondre a una bústia de correu del servidor.
- **PASS <password>.** El client s'ha d'autenticar indicant un nom d'usuari i una contrasenya vàlids. L'ordre PASS permet indicar aquesta contrasenya en text net.
- **APOP <nomUsuari> password-md5.** Per proporcionar més seguretat en el procés d'autorització, l'usuari pot fer servir l'ordre APOP, que té com a arguments el nom d'usuari i la contrasenya encriptada usant una funció resum o *hash*, com per exemple *MD5*.

2. Transacció

- **STAT**. Demana l'estat de la bústia. El servidor retorna el nombre de missatges que conté i el total de *bytes* que ocupa.
- **LIST [msg]**. Llista els missatges o un missatge concret. No en llista el contingut sinó que llista el número de missatge i el nombre de *bytes* que ocupa cada missatge.
- **RETR msg**. Baixa un missatge concret del servidor. Els missatges es poden indicar pel número de missatge.
- **DELE msg**. Marca el missatge indicat per ser esborrat. No l'esborra immediatament, només el marca; l'esborra en l'estat final d'actualització. En els MUA que actuen de clients POP és típic permetre configurar si es deixen els missatges en el servidor o s'eliminen. Usualment s'eliminen perquè només hi quedin els nous.
- **NOOP**. Aquesta ordre força el servidor a emetre una resposta positiva. Serveix per comprovar que la connexió encara és oberta.
- **RSET**. Si s'executa aquesta ordre abans de passar a l'estat d'actualització, RSET desmarca tots els missatges que estaven marcats per esborrar.
- **TOP msg nLin**. En lloc de baixar el missatge sencer com fa l'ordre RETR, l'ordre TOP permet baixar-ne les línies inicials. Baixa les capçaleres i les línies inicials (*nLin*). Aquesta ordre és útil per baixar només les capçaleres (remittents, assumptes...) i per filtrar els missatges a baixar i marcar-los per esborrar-los directament sense baixar-los.
- **UIDL [msg]**. Els missatges s'identifiquen pel seu número d'ordre (com fa l'ordre LIST), però aquest número pot variar entre connexions si s'esborren els missatges que el precedeixen. Per identificar de manera única un missatge independentment de la posició que ocupa es pot usar el UID. El UID és únic per a cada missatge d'una bústia. L'ordre UIDL pot llistar els UID i els números d'ordre de tots els missatges o els d'un missatge concret.
- **QUIT**. Indica al servidor que el client vol finalitzar la connexió. El servidor passa de l'estat de transacció al d'actualització.

3. Actualització

- En aquest estat no hi ha ordres. El servidor elimina els missatges marcats per ser esborrats, emet una resposta positiva al client i tots dos tanquen la connexió.

L'exemple següent correspon a un diàleg mitjançant Telnet entre client i servidor usant el protocol POP. S'hi poden veure les ordres i respostes del protocol:

```
1 [root@portatil ~]# telnet localhost 110
2 Trying 127.0.0.1...
3 Connected to localhost.
4 Escape character is '^'.
5 +OK POP3 localhost 2007a.104 server ready
```

```
6 USER pere
7 +OK User name accepted, password please
8 PASS pere
9 +OK Mailbox open, 1 messages
10
11 STAT
12 +OK 1 480
13 NOOP
14 +OK No-op to you too!
15 LIST
16 +OK Mailbox scan listing follows
17 1 480
18 .
19 RETR 1
20 +OK 480 octets
21 Return-Path: <root@localhost.localdomain>
22 Received: from tftp.server.cat (localhost [127.0.0.1])
23   by tftp.server.cat (8.14.1/8.14.1) with ESMTP id m5DI4ig8005681
24   for pere@tftp.server.cat; Fri, 13 Jun 2008 20:06:12 +0200
25 Date: Fri, 13 Jun 2008 20:04:44 +0200
26 From: root <root@localhost.localdomain>
27 Message-Id: <200806131806.m5DI4ig8005681@tftp.server.cat>
28 Status:
29
30 Aquest és un e-mail qualsevol,
31 el text s'escriu fins acabar
32 amb una línia que només conté un punt
33 .
34
35 QUIT
36 +OK Sayonara
37 Connection closed by foreign host.
```

Hi ha diverses implementacions de servidors POP i cada una compta amb un conjunt d'ordres i extensions pròpies. L'especificació POP3 requereix que s'implementin almenys les ordres STAT, LIST, RETR, DELE, NOOP i REST.

El model POP3

El POP3 és un protocol que permet l'accés remot a les bústies de correu dels usuaris, com l'IMAP. Ni el POP3 ni l'IMAP transporten el correu, aquesta funció la fa l'SMTP, sinó que ofereixen els mecanismes per a que un usuari amb el seu MUA pugui accedir a la seva bústia.

Com la majoria de serveis de xarxa a internet, el protocol POP3 s'estructura seguint l'esquema client/servidor. Aquests són els agents que intervenen en una comunicació POP3:

- **MUA** (*Mail User Agent* o **agent d'usuari de correu**). L'usuari interactua amb un agent d'usuari per accedir al correu mitjançant POP3. Són agents d'usuari programes com Thunderbird, GetMail, Fetchmail, MS Outlook Express, Eudora, Gmail... Les aplicacions MUA poden ser de text, gràfiques i fins i tot interfícies web. Aquestes aplicacions incorporen el programari necessari per actuar de clients POP3.
- **Client POP3**. La part pròpiament encarregada de comunicar amb el servidor POP3 per obtenir els missatges de la bústia de correu de l'usuari

és el client POP3. Client i servidor POP3 parlen un llenguatge comú, que és el protocol POP3.

- **Servidor POP3.** Per poder implementar l'accés remot al correu cal disposar d'un servidor POP3 en funcionament. Aquest servidor POP3 conté les bústies dels usuaris o hi accedeix. Els missatges es reben mitjançant SMTP, i és un MTA o un MDA el que els diposita a la bústia. El servidor POP3 atén les peticions dels clients POP3 per baixar el correu.

Un concepte que ajuda a diferenciar el funcionament de POP3 i IMAP és que en l'esquema de POP3 es considera que l'emmagatzematge del correu es realitza en la màquina de l'usuari. El servidor acumula els missatges nous i aquests es baixen tots a l'amfitrió (*host*) de l'usuari i s'eliminen del servidor. Per tant, gestionar-los és responsabilitat de l'usuari. Si bé és cert que els missatges es poden desar en el servidor (sense esborrar), no hi ha eines per gestionar-los, tots es troben en una mateixa carpeta. El servidor POP3 ofereix poques funcionalitats: baixar missatges, baixar les capçaleres i esborrar els missatges.

Una sessió POP3 passa per tres estats clarament diferenciats:

1. **Autorització.** Un cop feta la connexió TCP/IP pel port 110 entre el client i el servidor POP3, s'entra en l'estat d'autorització. Cal que el client s'identifiqui davant del servidor POP3 indicant el nom d'usuari i la contrasenya.
2. **Transacció.** Un cop el client ha estat autoritzat pel servidor, s'entra en l'estat de transacció. En aquest estat el client demana accions (dona ordres) al servidor i aquest les atén. És a dir, en aquest estat el client descarrega el correu, marca missatges per esborrar, demana les capçaleres dels missatges, en fa una llista per ordre... El client finalitza l'estat de transacció utilitzant l'ordre QUIT.
3. **Actualització.** El servidor entra en l'estat d'actualització en rebre l'ordre QUIT del client. Elimina els missatges marcats per esborrar (que fins ara no s'havien eliminat) i envia un OK al receptor. Ara tots dos poden finalitzar la comunicació.

Accés POP3 al correu web

Molts serveis de correu web o *webmail* permeten baixar correu d'altres serveis usant el protocol POP3 o IMAP. Per exemple, des de Gmail es poden baixar missatges de Yahoo, i viceversa.

1.3.2 Servei IMAP

IMAP (*Internet Message Access Protocol* o **protocol d'accés a missatges d'Internet**) és un protocol de capa d'aplicació del model TCP/IP que proporciona a l'usuari accés remot a la seva bústia de correu. L'IMAP sorgeix com a resposta al problema d'accés al correu des de diferents ordinadors utilitzant POP.

El POP és un protocol pensat per baixar el correu del servidor al PC local de l'usuari i poder-lo manipular després sense connexió a internet. Usant POP es considera que el correu resideix en l'equip de l'usuari, que baixa tot el correu de cop cada vegada que es connecta al servidor.

Quan els usuaris es van acostumar a consultar el correu remotament, ho van començar a fer des d'equips diferents: a casa, a la feina, de vacances... Cada cop que ho feien deixaven part del seu correu en llocs diferents. El que s'havien baixat a casa no es podia consultar a la feina, i viceversa. Un cop els usuaris es van acostumar a disposar de connexió d'internet més assíduament, calia un mecanisme més evolucionat d'accés remot al correu.

Ni l'IMAP ni el POP són protocols de transmissió de correu. Usualment és el protocol SMTP qui fa aquesta funció.

Per obtenir més informació sobre l'especificació del protocol IMAP en els RFC 1064 i 3501, aneu a la secció "Adreces d'interès" del web d'aquest crèdit.

L'IMAP presenta un enfocament diferent. Els missatges de correu es dipositen en el servidor i allà s'emmagatzemen en carpetes (o *folders* o *mailboxes*) i on es manipulen. L'usuari els pot baixar localment, però com a còpia temporal. Per tant, tota la gestió dels missatges de correu té lloc en el servidor. Això fa de l'IMAP un protocol més complex que el POP.

L'IMAP és un protocol de capa d'aplicació de la pila de protocols TCP/IP (port 143) definit en el document RFC 1064. Permet a un client de correu (MUA) obtenir remotament el correu dipositat en la bústia de l'usuari en un servidor IMAP.

L'IMAP sorgeix el 1986 amb el nom d'*Interim Mail Access Protocol*, que en la versió següent es canvia per *Interactive Mail Access Protocol* (document RFC 1064), i que finalment serà *Internet Mail Access Protocol*. L'evolució actual és IMAP versió 4 revisió 1 (març de 2003), corresponent al document RFC 3501 (del qual també s'han fet actualitzacions i extensions) que s'ha creat sota els auspicis de l'IETF.

Difusió del servei IMAP

Avui en dia l'IMAP està molt estès, però no és estrany trobar ISP i portals de correu web que permeten baixar el correu únicament mitjançant el POP.

El protocol IMAP està pensat per tenir en el servidor tot el correu de l'usuari organitzat en carpetes jeràrquiques de manera indefinida. Es permet la manipulació remota de les carpetes i els missatges. Tant les unes com els altres es poden crear, modificar i suprimir. Els missatges no s'esborren si no ho indica explícitament l'usuari. A més, aporta la funcionalitat de cerca i filtratge de missatges directament en el servidor. És a dir, no cal baixar els missatges per buscar els que compleixen unes condicions determinades. El protocol permet l'accés concurrent de diversos usuaris a la mateixa bústia, i el servidor pot notificar l'arribada de correu nou. Els missatges multipart es poden baixar parcialment, es poden buscar parts i baixar només les que interessin. Tots els missatges i les bústies tenen indicadors d'estat que descriuen, per exemple, si el missatge s'ha llegit, si s'ha contestat, si és nou...

El model IMAP

Com la majoria de serveis de xarxa a internet, el protocol IMAP s'estructura seguint l'esquema client/servidor. Aquests són els agents que intervenen en una comunicació IMAP:

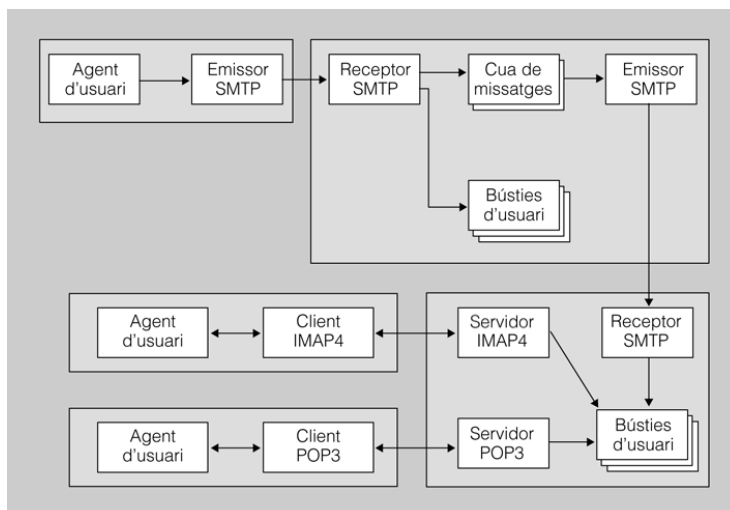
- **MUA** (*Mail User Agent* o **agent d'usuari de correu**). L'usuari interactua amb un agent d'usuari per accedir al correu mitjançant l'IMAP. Són agents d'usuari programes com Thunderbird, GetMail, Fetchmail, MS Outlook Express, Eudora o Gmail. Les aplicacions MUA poden ser de text, gràfiques

i fins i tot d'interfície web. Aquestes aplicacions disposen del programari necessari per actuar com a clients IMAP.

- **Client IMAP.** La part pròpiament encarregada de comunicar amb el servidor IMAP per obtenir els missatges de la bústia de correu de l'usuari és el client IMAP. Client i servidor IMAP parlen un llenguatge comú: el protocol IMAP.
- **Servidor IMAP.** Per implementar l'accés remot al correu cal disposar d'un servidor IMAP en funcionament. Aquest servidor IMAP conté les bústies dels usuaris o hi accedeix. Els missatges es reben per SMTP i és un MTA o un MDA els que els diposita a la bústia. El servidor IMAP atén les peticions dels clients IMAP per gestionar el correu.

Observeu el model funcional del protocol IMAP en la figura 1.2, on mitjançant el seu MUA un usuari descarrega el correu del servidor amb el POP o IMAP.

FIGURA 1.2. Model funcional del protocol IMAP



En el protocol IMAP s'observen quatre estats:

1. **No autenticat.** Quan s'estableix la connexió TCP/IP entre el client i el servidor s'entra en aquest estat. El client s'ha d'autenticar en el servidor, ha d'acreditar ser un usuari vàlid. Per fer-ho ha d'indicar el nom d'usuari i la contrasenya.
2. **Autenticat.** Un cop autenticat i abans de poder manipular missatges, ha de seleccionar la bústia (carpeta, *folder* o *mailbox*) amb la qual operarà. En aquest estat pot manipular les carpetes (crear-ne, esborrar-les, modificar-les i veure'n l'estat), però no els missatges fins que no se n'ha seleccionada una.
3. **Seleccionat.** Un cop s'ha seleccionat una carpeta, s'entra en aquest estat, que permet la manipulació dels continguts de la carpeta.
4. **Logout.** En aquest estat es tanca la connexió. S'hi pot arribar tant per petició del client com per decisió unilateral del servidor.

El servidor IMAP emmagatzema permanentment els missatges de l'usuari. Per fer-ho utilitza un sistema de bústies o carpetes jeràrquiques i atributs que descriuen tant l'estat de les bústies com dels missatges:

Carpetes

Hi ha una bústia o *mailbox* que és la de l'usuari. Dins d'aquesta bústia, s'hi poden crear carpetes indicades de manera relativa, com els directoris d'una estructura de fitxers. Les carpetes disposen almenys de dos atributs:

- **Next UID** (UID següent). Indica l'UID que s'assignarà al missatge següent.
- **UID Validity Value** (UIDVALIDITY). És un valor d'identificador únic assignat a la carpeta seleccionada. La combinació de nom de carpeta UIDVALIDITY i UID identifica de manera perpètua un missatge en el servidor.

Atributs de missatge

Els missatges tenen atributs que s'emmagatzemen en les pròpies bústies que en faciliten la gestió:

- **UID**. Identificador únic del missatge. És un número de 32 bits que s'assigna ascendentment a mesura que arriben missatges (no és necessàriament correlatiu). Això permet al servidor saber, en una bústia, a partir de quin número de missatge hi ha els missatges nous (en POP això no és possible).
- **Número de seqüència**. Número relatiu del missatge dins de la bústia (de 1 a n per a n missatges). Els números de seqüència s'assignen correlativament segons el UID en ordre ascendent i varien en esborrar-se i afegir-se nous missatges.
- **Indicadors**. Els indicadors, *flags* o banderes informen de l'estat del missatge. Per exemple, si s'ha llegit o esborrat. Els indicadors són: *Seen* (llegit), *Answered* (respost), *Flagged* (marcat), *Deleted* (esborrat), *Draft* (esborrany) i *Recent* (nou).
- **Data interna**. Data i hora d'arribada del missatge al servidor IMAP (no és la data i hora de l'emissió del missatge que hi ha en la capçalera *Date*).
- **Longitud**. Nombre de *bytes* del missatge.
- **Estructura del sobre**. Representació analitzada de les capçaleres del missatge.
- **Estructura del cos**. Representació analitzada de l'estructura MIME del cos del missatge.
- **Parts de text del missatge**. Per permetre la cerca de les diferents parts de text del missatge. Es pot fer l'accés segons la part de capçaleres, cos, part del cos MIME i capçalera MIME.

Funcionament de l'IMAP

Amb el protocol IMAP el client fa una connexió TCP/IP al port 143 del servidor i s'inicia un diàleg entre el client i el servidor en què tots dos poden prendre la iniciativa. En aquest procés se succeeixen els quatre estats del protocol IMAP (no autenticat, autenticat, seleccionat i *logout*) i s'intercanvien ordres i respostes seguint el model de diàleg de Telnet:

- **Ordres.** Són ordres de text que inclouen un *tag* (o etiqueta curta) inicial, l'ordre i els seus arguments. Cada ordre comença amb una etiqueta inicial diferent per diferenciar-la de les altres ordres i aquesta és obligatòria. Quan el servidor emeti la resposta final de l'ordre i indiqui si s'ha realitzat correctament o no, ho farà mostrant l'etiqueta de l'ordre a la qual respon. Per exemple, es pot usar *a001* per a la primera ordre, *a002* per a la segona... El client pot enviar ordres sense esperar que finalitzin les anteriors.
- **Respostes.** El servidor pot enviar dades al client tant com a resposta a una ordre com de manera unilateral (per exemple, per informar que hi ha correu nou). El client ha d'estar en tot moment a punt per rebre aquestes dades. Que el servidor envii dades al client no significa que l'execució de l'ordre del client hagi finalitzat. Només es dona per finalitzada quan el servidor emet una resposta amb la mateixa etiqueta que l'ordre del client. El servidor pot processar una ordre abans d'acabar de processar l'ordre anterior.

Vegeu les ordres utilitzables en el protocol IMAP agrupades segons l'estat:

1. Ordres generals (qualsevol estat)

- **Capability.** Llista les capacitats del servidor. Permet al client saber quines són les prestacions del servidor.
- **Noop.** Exigeix una resposta afirmativa del servidor. Permet al client saber si encara es manté la connexió.
- **Logout.** És la notificació del client al servidor per fer-li saber que vol finalitzar la connexió.

2. No autenticat

- **Authenticate <tipus>.** Indica al servidor el mecanisme d'autenticació a utilitzar.
- **Login <user> <password>.** El client s'identifica en el servidor indicant el nom d'usuari i la contrasenya. El format varia (text net, hash MD5...) segons el tipus d'autenticació utilitzat.

3. Autenticat

- **Select <bústia>.** Selecciona la bústia amb què ha d'operar. En fer-ho, el servidor emet una resposta no etiquetada en què informa dels atributs (*flags*) de la bústia, del nombre de missatges que conté (*exists*) i del nombre de missatges recents (*recent*). També pot indicar el

Diferència entre POP i IMAP

En el protocol POP el servidor només pot respondre a peticions del client, però no pot prendre la iniciativa. En el protocol IMAP sí.

Avantatges de l'IMAP respecte al POP

Un dels avantatges de l'IMAP respecte al POP és que el servidor sap a partir de quin número de missatge hi ha els missatges nous (*RECENT*).

número del primer missatge no llegit (*nseen*) i la llista d'atributs que es poden modificar (*permanent flags*).

- **Examine <bústia>**. Realitza la mateixa funció que l'ordre *Select* però només de lectura.
- **Create <bústia>**. Crea la bústia amb el nom indicat.
- **Delete <bústia>**. Esborra la bústia indicada.
- **Rename <bústia> <nomNou>**. Permet assignar un nom nou a la bústia.
- **Subscribe <bústia>**. Les bústies poden estar actives o no. Aquesta ordre les activa.
- **Unsubscribe <bústia>**. Permet desactivar una bústia.
- **List <bústia> <criteri>**. Llista les bústies que compleixen el criteri indicat dins de la bústia seleccionada.
- **Lsub <bústia> <criteri>**. Realitza la mateixa funció que l'ordre anterior però només per a les bústies actives.
- **Status <bústia> <atributs>**. Permet conèixer l'estat d'una bústia per mitjà dels seus atributs. Els atributs són els següents:
 - *MESSAGE*: nombre de missatges dins la bústia
 - *RECENT*: nombre de missatges recents (nous)
 - *UIDNEXT*: UID assignat al següent missatge que arribi a la bústia
 - *UIDVALIDITY*: valor UID de la bústia
 - *UNSEEN*: nombre de missatges no llegits
- **Append bústia [atributs] [data-hora] literal**. Permet afegir un text al final de la bústia com si es tractés d'un missatge nou. El missatge s'afegeix amb la data, hora i atributs indicats.

4. Seleccionat

- **Check**. El client sol·licita al servidor que es faci un control de la bústia en un moment determinat.
- **Close**. Tanca la bústia i elimina tots els missatges que conté que tenen l'indicador d'esborrament (*deleted*) activat.
- **Expugne**. Permet esborrar els missatges que tenen l'atribut d'esborrament (*deleted*) activat sense que calgui tancar la bústia.
- **Search [charset] criteri**. Permet buscar missatges dins de la bústia que compleixen el criteri de cerca indicat.
- **Fetch dades atributsRecuperació**. Permet recuperar un conjunt de missatges totalment o parcialment segons els atributs de recuperació indicats.
- **Store conjuntMissatges atributs**. Permet alterar les dades d'atributs associats a un conjunt de missatges en una bústia.
- **Copy conjuntMissatges bústia**. Copia un conjunt de missatges al final de la bústia indicada.

- **UID ordre.** Retorna l'UID d'un missatge. S'utilitza conjuntament amb les ordres COPY, FETCH, STORE o SEARCH per retornar els UID manipulats.

5. Experimental

- **X<ordre>.** Es poden desenvolupar ordres experimentals fora de l'especificació IMAP. Per fer-ho cal que les ordres comencin amb *XnomOrdre*. D'aquesta manera s'evita que es produeixin conflictes amb ordres futures.

En l'exemple següent podeu veure tot el diàleg client/servidor d'una sessió IMAP utilitzant Telnet:

```
1 [root@portatil ~]# telnet localhost 143
2 Trying 127.0.0.1...
3 Connected to localhost.
4 Escape character is '^'.
5 * OK [CAPABILITY IMAP4REV1 LITERAL+ SASL-IR LOGIN-REFERRALS
6 STARTTLS] localhost IMAP4rev1 2007a.403 at Sat, 14 Jun 2008
7 13:16:47 +0200 (CEST)
8
9 a003 LOGIN pere pere
10 a003 OK [CAPABILITY IMAP4REV1 LITERAL+ IDLE UIDPLUS NAMESPACE
11 CHILDREN MAILBOX-REFERRALS BINARY UNSELECT ESEARCH WITHIN SCAN
12 SORT THREAD=REFERENCES THREAD=ORDEREDSUBJECT MULTIAPPEND] User
13 pere authenticated
14
15 a004 SELECT inbox
16 * 4 EXISTS
17 * NO Mailbox vulnerable – directory /var/spool/mail must have
18 1777 protection
19 * 4 RECENT
20 * OK [UIDVALIDITY 1213385060] UID validity status
21 * OK [UIDNEXT 6] Predicted next UID
22 * FLAGS (\Answered \Flagged \Deleted \Draft \Seen)
23 * OK [PERMANENTFLAGS (\* \Answered \Flagged \Deleted \Draft
24 \Seen)] Permanent flags
25 * OK [UNSEEN 1] first unseen message in /var/spool/mail/pere
26 * NO Mailbox vulnerable – directory /var/spool/mail must have
27 1777 protection
28 a004 OK [READ-WRITE] SELECT completed
29
30 a005 FETCH 1 RFC822.TEXT
31 * 1 FETCH (RFC822.TEXT {63}
32 exemple de missatge de la usuària anna
33 a l'usuari pere
34 adéu
35 )
36 a005 OK FETCH completed
37
38 a006 LOGOUT
39 * BYE portatil IMAP4rev1 server terminating connection
40 a006 OK LOGOUT completed
```

1.3.3 Clients de correu

L'accés a les bústies de correu electrònic a través de les ordres i comandes que ofereixen els protocols IMAP i POP és complex i per a un usuari normal, impracticable. No obstant, existeix tot un conjunt de programari que automatitza totes

aquestes tasques i les presenta en un entorn més amigable. Aquest programari s'anomena **client de correu**. Existeixen clients de correu de característiques molt diverses.

En general es poden classificar en:

- **Clients de text.** Utilitats Unix i GNU/Linux de tota la vida que permeten generar missatges de correu i accedir a la bústia de correu pròpia. Són utilitats de consola, és a dir, de text. N'hi ha de format ben simple, com l'ordre *mail*, i evolucions que permeten treballar amb programes de menús en color també en entorn de consola, com Mutt, Alpine...
- **Clients gràfics.** Amb la popularització dels entorns gràfics van sorgir programes client de correu com Eudora, Evince, Outlook o Thunderbird. Durant un temps aquests programes eren l'eina més usada pels usuaris per accedir a les seves bústies de correu.
- **Client web.** Actualment la majoria de comptes de correus dels usuaris són de tipus correu web, principalment en serveis gratuïts com Gmail, Yahoo o Hotmail. Fins i tot els entorns corporatius utilitzen correu web hostatjat en serveis externalitzats.

1.4 Correu encriptat i signat

Les comunicacions per correu electrònic s'han tornat cada vegada més importants en la nostra vida diària, no només pels correus amb acudits, presentacions gracioses o crítiques al govern. També són imprescindibles per al funcionament de moltíssimes empreses i organitzacions. De fet, molta documentació que abans es feia per escrit ara es fa telemàticament per correu electrònic.

Això fa necessari poder estar segurs de la identitat de l'emissor i del receptor dels correus. En entorns de seguretat més exigents fins i tot es pot requerir l'encriptació del contingut per evitar que sigui accessible per tercers.

El concepte clau per a la confiança en les comunicacions per correu electrònic és la **signatura digital**, que permet assegurar que un emissor és qui diu ser i garanteix també que el contingut del missatge no s'ha alterat per tercers. Per implementar la signatura digital i l'encriptació cal utilitzar **certificats digitals**.

Consulteu a "Annexos" l'explicació més àmplia de conceptes globals de seguretat i documents específics per a PGP, S/MIME i certificats digitals.

És important tenir clars els conceptes generals de seguretat per poder diferenciar els termes relacionats amb la seguretat dels quals no es coneix el significat amb exactitud.

Els puntals de la seguretat en el correu són: autenticació, integritat, no repudi i encriptació. Es poden crear parelles de claus públiques i privades (certificats digitals) i hi ha diversos formats de claus existents.

L'MTA i els servidors de correu transporten missatges independentment del fet de que estiguin encriptats o signats. Són els **clients de correu** els que han de proporcionar a l'usuari la capacitat d'**encriptar** i **signar** missatges .

Per poder gestionar correu signat i encriptat cal utilitzar clients de correu que permetin fer aquestes funcions. Cada usuari ha de disposar dels certificats digitals apropiats. Per al protocol de transport de correu SMTP i al servidor de correu (per exemple, Sendmail) que el missatge que processen sigui encriptat i signat és indiferent, igual que al carter no li afecta que el contingut d'una carta estigui encriptat amb una clau secreta o que la carta porti imprès el segell d'una entitat. Són els clients de correu, com Thunderbird, els que han de tenir la capacitat de gestionar aquest tipus de correu. Per exemple, a Thunderbird cal afegir-li programari addicional (Open PGP o S/MIME, per exemple) per permetre-li signar i encriptar missatges.

Les dues tecnologies de signat i encriptat de correu tractades aquí són:

- **Open PGP.** Aquesta tecnologia permet generar missatges de correu encriptats i signats i, lògicament, desxifrar-ne el contingut i verificar les signatures. Es basa en el programa PGP (*Pretty Good Privacy*), desenvolupat per Phil Zimmermann. Utilitza el model de seguretat anomenat *web of trust*, en què cada usuari és el responsable de la gestió de la confiança en els certificats dels altres usuaris.
- **S/MIME.** Aquest estàndard proporciona les mateixes característiques de signatura digital i encriptació (i a la inversa) que Open PGP, però requereix un altre format per als certificats digitals. El model de seguretat que utilitza és el PKI (*Public Key Infrastructure*), en què existeix una estructura piramidal d'entitats de certificació (*Certification Authority*, CA) que determinen la confiança en els certificats digitals. Aquest PKI és el model en què es basen la majoria de protocols de seguretat d'internet, com HTTPS o SMTPS, que utilitzen SSL o TLS. Usualment els navegadors web incorporen per defecte els certificats de les CA més destacades del món.

1.4.1 Seguretat en el correu

L'intercanvi de missatges de correu es produeix utilitzant protocols SMTP, POP i IMAP, que són insegurs, és a dir, que transporten el contingut en forma de text pla. Per tant, qualsevol intermediari en la xarxa pot monitorar (usant un *sniffer* o rastrejador) el contingut dels missatges. Qualsevol eina tipus Wireshark permet fer un seguiment dels continguts de qualsevol protocol TCP en text pla.

Monitoratge de contingut

Un exemple típic de monitoratge és monitorar el diàleg entre dos amfitrions usant Wireshark. Permet fer un seguiment de les trames de xarxa i utilitzant l'opció de seguiment del flux TCP anomenada *Follow TCP Stream* es pot observar clarament el text de tot el diàleg efectuat.

Això significa que quan dos interlocutors intercanvien missatges per qualsevol xarxa el contingut del seu diàleg pot ser monitorat per tercers. I no només això, sinó que la conversa pot ser falsejada per aquests tercers (el conegut com a *man-in-the-middle*). És a dir, un usuari pot creure que està dialogant amb la seva entitat bancària quan en realitat ho està fent amb uns impostors.

1.4.2 Propietats de seguretat

Els aspectes de seguretat desitjables en la comunicació entre dos interlocutors són:

- Confidencialitat (encriptació)
- Autenticitat
- No-repudi
- Integritat

Un error comú es barrejar aquests conceptes i pensar que tots quatre van junts, i no sempre és així. Cal aplicar a cada cas el tipus de seguretat que faci falta.

Confidencialitat

Si un emissor vol enviar un missatge secret a un receptor de manera que únicament aquest el pugui entendre, ha d'encriptar el missatge. El destinatari ha de conèixer la clau o ha de disposar d'un mecanisme per desencriptar el missatge i obtenir-ne el contingut original.

Encriptar és codificar el missatge utilitzant algun tipus de clau o mecanisme per fer inaccessible el missatge a qualsevol usuari que no sigui el destinatari.

Que el missatge estigui encriptat garanteix que només l'emissor és capaç d'entendre'n el contingut (se suposa que han acordat el mecanisme per desencriptar). Ara bé, pot estar segur el destinatari que el missatge prové realment de qui creu que prové? Pot estar segur que el missatge no ha estat alterat?

Encriptar un missatge proporciona **confidencialitat**, però no és un mecanisme que ofereixi la seguretat que l'emissor és qui diu ser ni que el missatge és tal com era en l'origen (sense modificacions posteriors).

Autenticitat

Quan un receptor rep un missatge del banc informant-lo que ha de fer un pagament o que ha d'enviar el número de la seva targeta de crèdit per una raó determinada, però no pot estar segur el receptor que el missatge procedeix realment del banc.

L'**autenticació** és la característica de seguretat que permet a un receptor estar segur que el missatge prové de qui diu provenir. Alhora, proporciona a l'emissor la garantia que el receptor sap del cert que el missatge li ha enviat ell.

No-repudi

Si un emissor envia un missatge autenticat a un emissor no té manera legal de desdir-se d'haver-lo enviat. Imagineu que un directiu d'una empresa envia un missatge inapropiat a un treballador. Si el missatge és autenticat, el directiu no pot negar legalment que l'ha enviat. El mateix pot passar si l'empresa A envia un correu electrònic a un client oferint-li els productes a meitat de preu, i després intenta retractar-se'n dient que el correu no era seu. Si el missatge és autenticat, queda legalment demostrat que l'empresa A n'és l'emissora.

Aquesta característica que impedeix que l'emissor pugui negar haver enviat el missatge s'anomena **no-repudi**.

Cal un mecanisme de seguretat que permeti a un emissor enviar missatges de manera que els receptors tinguin la certesa absoluta que l'emissor és qui diu ser.

Una confusió habitual és creure que per establir seguretat també cal encriptar. Això no és cert. Una administració pública, per exemple, pot enviar missatges als ciutadans garantint l'autenticitat del missatge, però no li cal encriptar els missatges, no li cal fer-los secrets.

Integritat

No n'hi ha prou amb l'encriptació i l'autenticació per establir una comunicació cent per cent segura. S'ha d'assegurar que el missatge no ha estat alterat, és a dir que no s'han eliminat o afegit parts, ni tampoc que se n'hagin modificat. S'ha d'assegurar que el missatge està íntegre. L'encriptació i l'autenticació asseguren que el missatge procedeix de qui diu procedir i que no ho veu ningú més, però no s'assegura que no hagi estat modificat.

La **integritat** és la propietat de seguretat que garanteix que el missatge no ha estat alterat i que arriba al destinatari tal com s'ha generat en l'origen.

Per implementar integritat no cal encriptar els missatges, són coses independents. En canvi, la integritat i l'autenticitat van juntes, és a dir, tenint l'una també s'obté l'altra (i viceversa).

1.4.3 Implementació de seguretat

Per implementar seguretat en la comunicació avui en dia s'utilitzen habitualment els mecanismes de certificats digitals basats en la criptografia de clau asimètrica. És a dir, basats en el fet que cada interlocutor disposa d'una clau privada (coneguda i accessible només per ell) i d'una clau pública o **certificat** (coneguda per tots els interlocutors).

Els certificats són les claus públiques **signades**, avalades, per una entitat de certificació o CA (*Certification Authority*).

Per implementar seguretat els interlocutors disposen de: una **clau privada** i un **certificat**, o clau pública signada per una CA.

Vegeu com s'implementa la seguretat de clau pública/privada:

- Encriptació
- Signatura o certificat digital

En realitat, la qüestió de la seguretat és més complexa si s'hi afegeixen les CA, els anells de clau, el sistema PKI... En aquesta unitat es tracta la versió més simple d'aquest model i s'estudia només allò estrictament necessari per a la comunicació segura entre dos interlocutors.

Cal tenir molt present que la clau privada és això, privada: només la coneix i només hi pot accedir el seu propietari. Mai es comunica a un tercer. En canvi, la clau pública és coneguda per totes les parts que intervenen en la comunicació. De fet, si els interlocutors no la coneixen cal enviar-los-la per tal que la tinguin. Si no fos així, el sistema de clau pública/privada no funcionaria.

La parella clau pública/privada permet encriptar i signar. Quan se n'utilitza una per fer una acció, cal l'altra per desfer-la. Funcionen conjuntament: si una fa, l'altra desfà, i a l'inrevés.

Mala interpretació del funcionament de les claus

Un error típic és creure que cada clau només pot fer una cosa i dir que "la clau privada sempre encripta i signa i la pública desencripta i verifica". No va així. Segons l'acció a fer se n'utilitza una o l'altra (i per desfer l'acció sempre cal aplicar la contrària).

Encriptació

Un **emissor A** vol enviar un missatge encriptat a un **destinatari B**, de manera que únicament aquest tingui la capacitat de conèixer el contingut real del missatge.

Per fer-ho, l'emissor encripta el missatge amb la clau pública del destinatari. De fet, qualsevol emissor del món ho pot fer, precisament perquè la clau pública del destinatari és pública.

Un cop encriptat el missatge, únicament el pot desencriptar qui tingui la clau privada associada a la clau pública usada per encriptar-lo. És a dir, només podrà desencriptar el missatge el destinatari.

Publicació de la clau pública

La clau pública ha de ser coneguda per tots els participants en la comunicació. Els mecanismes per donar-la a conèixer són:

- Publicar-la en un servidor públic de claus.
- Enviar la clau als destinataris als quals s'adreça l'emissor (amics, coneguts i saludats).
- Adjuntar la clau pública als missatges.

Què coneix l'emissor del destinatari? La seva clau pública. De manera que l'emissor encripta el missatge amb la clau pública del destinatari. De fet, qualsevol emissor del món ho pot fer, precisament perquè la clau pública del destinatari és pública.

Un cop encriptat el missatge, la resta (inclòs l'emissor) no poden desencriptar-lo. Únicament el pot desencriptar qui tingui la clau privada associada a la clau pública usada. És a dir, només podrà desencriptar el missatge qui disposi de la clau privada del destinatari B, és a dir que només el destinatari podrà fer-ho.

Qualsevol pot **encriptar** un missatge usant la **clau pública del destinatari**, que precisament és pública. Únicament el destinatari pot **desencriptar** el missatge usant la seva **clau privada**.

Signatura

Signar un missatge proporciona **integritat, autenticació i no-repudi** simultàniament. Quan un missatge està signat per l'emissor és irrefutable que el missatge procedeix d'ell i, a més, garanteix que no s'ha modificat per tercers. El receptor pot **verificar** així que el missatge és autèntic.

El procés físic de signar el missatge consisteix a afegir al missatge el certificat digital de l'emissor. De vegades el fet de signar un missatge s'anomena certificar-lo o es parla de missatge amb certificat digital.

El funcionament és força similar al de l'encriptació, però en aquest cas l'emissor **signar** un missatge utilitzant la seva pròpia **clau privada**, i el receptor utilitza la **clau pública** de l'emissor per **verificar** el missatge.

Com que les parelles de claus pública/privada només funcionen conjuntament, la verificació només serà correcta si el missatge s'ha encriptat amb la clau privada corresponent. És a dir, la verificació amb la clau pública de l'emissor A només funcionarà si el missatge s'ha signat amb la clau privada de l'emissor A.

Qualsevol pot **verificar** la signatura d'un missatge usant la **clau pública de l'emissor**. Únicament l'emissor pot **signar** el missatge usant la seva **clau privada**.

El procés de signatura és complementari al de l'enciptació. El primer proporciona integritat, autenticació i no-repudi, mentre que el segon proporciona confidencialitat.

Que un missatge estigui **signat** no significa que sigui secret. Perquè sigui **secret** també ha d'estar **enciptat**.

El procés mecànic de signar un missatge consisteix a aplicar la clau privada al missatge. Això comporta els processos següents:

- **Integritat:** no es codifica tot el missatge amb la clau privada, ja que això implica un sobrecost de temps i esforç de càlcul. El procés tècnic que es fa és generar un *hash* o resum del missatge i signar-lo. Si el missatge es modifiqués, el resum no coincidiria amb el missatge.
- **Autenticitat:** per autenticar el missatge s'adjunta el certificat o clau pública de l'emissor avalat per una autoritat de certificació o CA. Aquest certificat i el resum van codificats amb la clau privada. No tot el missatge, només aquesta part. Així el receptor verifica amb la clau pública de l'emissor que el certificat de l'emissor és vàlid i que el *hash* també.

Tècniques de 'hash' o resum

Una funció resum (*hash*) és una funció que permet reduir qualsevol informació de qualsevol mida a un valor de mida fixa. Entre les seves utilitats hi ha la validació de la integritat de fitxers (tant per temes d'autenticitat com de comprovació d'errors en la transmissió), autenticació amb signatura digital, i en programació i base de dades per a la indexació de les dades.

Són funcions exhaustives, i moltes vegades s'anomenen de direcció única, ja que la funció inversa no dona un únic resultat. A més, el càlcul de la funció inversa és costós. Per exemple, per calcular el valor d'origen normalment es recorre a atacs de força bruta.

Algunes funcions resum són els CRC (*Cyclic Redundancy Checks*), els MD (*Message Digest*) i els SHA (*Secure Hash Algorithm*).

En definitiva, el procés de signar un missatge o incorporar al missatge una signatura digital consta dels passos següents:

1. Es genera un *hash*, *message digest* o *fingerprint* del missatge.
2. S'encipta el resum amb la clau privada de l'emissor. Això és la signatura digital.
3. El destinatari rep el missatge i fa un nou resum, és a dir, torna a fer el *hash* basant-se en el contingut del missatge rebut.
4. La signatura digital rebuda (resum enciptat) es desencipta amb la clau pública de l'emissor.

Per tant:

- Els dos resums han de ser iguals: això garanteix l'autenticació i la integritat.

- Es garanteix l'autenticació: només l'emissor pot haver generat el missatge si es pot descriptar per la clau pública de l'emissor.
- Es garanteix la integritat: ningú més pot haver modificat el missatge perquè això hauria modificat el resum i no hi ha ningú més que tingui la clau privada de l'emissor per tornar-lo a signar.
- No cal encriptar tot el missatge, només el resum, que és la part que garanteix que no s'ha modificat.

1.5 Servidor de correu segur

Els protocols de correu analitzats en aquesta documentació són tots protocols de transport d'informació en text pla. Qualsevol comunicació SMTP, POP o IMAP es fa en text pla i pot ser monitorada per tercers que tinguin accés als nodes de xarxa per on passen aquestes comunicacions. De fet, s'han vist exemples de monitoratge de les converses TCP utilitzant Wireshark. Aquesta feblesa no és exclusiva dels protocols de correu, sinó que és comuna a la majoria de protocols d'Internet, com HTTP, FTP o TFTP.

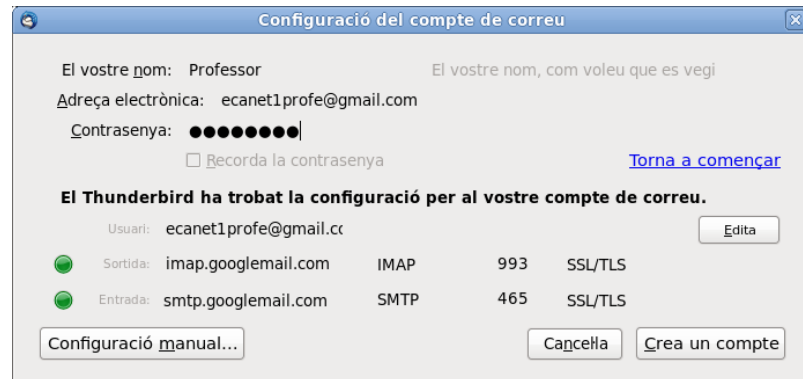
Els protocols de correu tenen mecanismes similars als usats en HTML per establir canals de comunicació segurs:

- **SMTPS**: no és un protocol diferent de l'SMTP ni una extensió seva, és simplement una manera d'anomenar l'ús d'SMTP a través d'SSL o TLS. Així com l'usuari estableix comunicacions HTTP segures usant HTTPS, pot establir un transport de correu segur amb SMTPS. El protocol és el mateix, però viatja per SSL, que li proporciona seguretat. Per usar SMTPS cal comunicar-se per un port diferent del 25, en concret el 465. Això representava un problema, ja que el sistema de correu a Internet es basa majoritàriament en la utilització del port 25, però aquest problema es va resoldre amb la utilització d'STARTTLS
- **POPS**: com passa amb HTTPS i SMTPS, POPS no és un protocol pròpiament dit, sinó tan sols la utilització de l'accés remot a bústies POP amb transport segur SSL o TLS. Utilitza el port 995 en lloc del port 110 clàssic de POP
- **IMAPS**: és també la manera d'anomenar el protocol IMAP quan viatja per una capa de transport segur com SSL o TLS. El port utilitzat per IMAP sobre SSL és el 993

Els acrònims SMTPS, POPS i IMAPS indiquen la utilització del protocol usant una **capa de transport segura** SSL o TLS, cosa que permet una comunicació encriptada que no pot ser monitorada per tercers (com passa amb HTTP sobre SSL, que s'anomena HTTPS).

En la figura 2.2 es pot veure la pantalla de Thunderbird de creació d'un compte de correu de Gmail. Observeu que el correu entrant utilitza IMAP sobre SSL/TLS al port 993 del servidor `imap.googlemail.com`, que és el servidor de correu de Gmail. També es configura com a correu sortint el servidor SMTP sobre SSL/TLS de Google. S'utilitza el port 465 del servidor `smtp.googlemail.com`, que és el nom de l'amfitrió de Gmail que fa la funció de servidor SMTP.

FIGURA 1.3. Configuració de compte de correu a Gmail



L'exemple següent mostra com s'inicia un diàleg en mode consola amb un servidor SSL. S'utilitza una de les utilitats de l'ordre *openssl*, que permet actuar com a client SSL. En aquest exemple es contacta el servidor IMAP de Gmail:

```
1 [root@host ~]# openssl s_client -crlf -connect imap.gmail.com:993
2 [root@host ~]# openssl s_client -crlf -connect smtp.googlemail.com:465
```

2. Instal·lació i administració de serveis de missatgeria instantània, notícies i llistes de distribució

Els serveis de missatgeria instantània, notícies i llistes de distribució complementen el sistema de missatgeria tradicional basat en el correu electrònic, cadascun amb les seves pròpies característiques.

La missatgeria instantània permet la comunicació en temps real, i aquesta és una de les principals diferències amb el servei de correu electrònic. Aquest tipus de comunicació permet incloure més d'un interlocutor i en diferents formats multimèdia com el text, l'àudio i/o el vídeo.

Les llistes de distribució permeten rebre missatges de correu electrònic sobre un tema determinat. La diferència entre el correu electrònic i les llistes de distribució rau en el fet que no coneixem el destinatari, és a dir, quan enviem un missatge a la llista, aquesta la reenvia als seus subscriptors en el cas de llistes de discussió. Un altre cas són les llistes de publicació on els missatges són unidireccionals (només rebem missatges).

El servei de notícies funciona de manera molt similar a un tauler d'anuncis on es poden penjar i consultar notícies. En aquest cas tampoc cal especificar cap destinatari, ja que són els mateixos usuaris els que van a aquest tauler. Aquest servei es realitza a través d'un protocol específic, el NNTP (*Network News Transfer Protocol*).

2.1 Missatgeria instantània

L'evolució de les tecnologies, l'abaratiment de costos i la popularització d'internet han provocat que avui en dia una gran quantitat de dispositius permetin navegar i comunicar-se per internet. No només mitjançant els ordinadors, sinó també amb els mòbils, les tauletes tàctils i fins i tot els televisors. Una forma popular de comunicar-se entre usuaris és per xats i missatgeria instantània, que permeten la transmissió en temps real de text, àudio i vídeo.

La **missatgeria instantània** o **IM** (*Instant Messaging*) proporciona comunicació entre dos o més usuaris en temps real. Aquesta comunicació pot ser en format de text, àudio i vídeo.

Per entendre millor el concepte d'IM, cal contrastar-lo amb altres formes de comunicació i la seva evolució.

Una de les formes clàssiques de comunicació entre dos interlocutors és la **correspondència**: enviar-se cartes. En les comunicacions informàtiques aquest

mecanisme s'implementa amb el correu SMTP. La diferència del correu electrònic amb l'IM és que el primer és una comunicació **asíncrona**, mentre que en els diàlegs IM (comunicacions de xat en línia) la comunicació és **síncrona**. Els missatges s'envien en temps real i els interlocutors emeten i reben missatges amb immediatesa, construint una comunicació fluida.

Seguint l'evolució històrica de la comunicació social, després del correu postal una altra forma popular de comunicació va ser el **telèfon**. Aquest proporciona una comunicació **síncrona** entre dos interlocutors que dialoguen en temps real. Un primer mecanisme per implantar en les xarxes informàtiques un equivalent al telèfon va ser l'aplicació **talk**, que permetia el diàleg de text en temps real entre dos interlocutors. De fet, talk és un predecessor dels sistemes d'IM actuals. Tant el telèfon com talk permeten la comunicació entre dos interlocutors que coneixen d'antuvi la identitat l'un de l'altre (el número de telèfon o l'adreça usuari@host del destinatari).

Amb l'expansió d'internet va sorgir un tipus de comunicació anomenat **xat**, en el qual podem incloure la missatgeria instantània, tot i algunes particularitats. Els xats permeten la comunicació entre dos o més clients en "sales" d'un servei de xat. Els usuaris s'identifiquen per un sobrenom (*nickname*) que generalment no té perquè ser un nom real (permet un cert anonimat). Molts xats funcionen per interfície web i també amb programes clients específics.

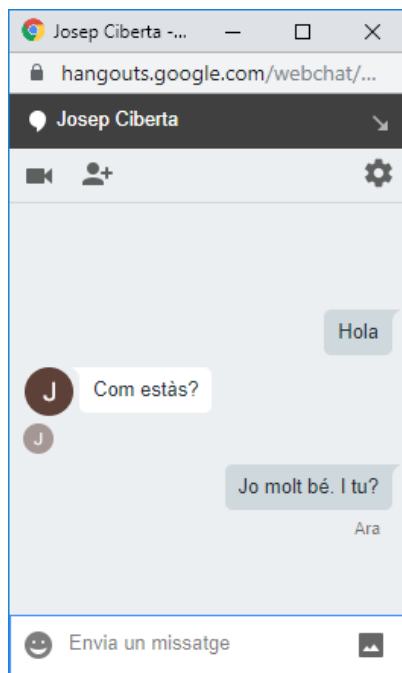
El sorgiment del telèfon mòbil i els **missatges SMS** van ser el següent pas evolutiu en aquest camp. Els missatges SMS són una forma de missatgeria instantània per text, on temps real i **síncron** és una mica més dubtós.

Actualment, els telèfons mòbils presten la majoria dels serveis de comunicació (navegació web, xats i missatgeria instantània, entre d'altres) a través d'internet. També els televisors incorporen prestacions d'Internet com l'accés a continguts a la carta, la navegació web i la missatgeria instantània. Sí, no és estrany trobar televisors que incorporen una càmera i un micròfon i que permeten utilitzar eines de comunicació com Skype.

La **comunicació per missatgeria instantània (IM)** es diferencia del correu i de les llistes de distribució pel fet que és comunicació **en temps real** o **síncrona**. És una comunicació de tipus xat, però es diferencia en el fet que és una comunicació no anònima, sinó que es produeix entre dos o més **interlocutors coneguts**.

En la figura 2.1 es pot veure un diàleg xat entre dos usuaris que usen el client de missatgeria instantània Google Hangouts. Observeu que les prestacions d'aquest servei de missatgeria inclouen el xat en text i vídeo i la realització de trucades telefòniques.

FIGURA 2.1. Exemple de diàleg de missatgeria instantània



En resum, la missatgeria instantània té com a finalitat comunicar dos o més interlocutors. Es pot dir que “xategen”, ja sigui en format de text, àudio o vídeo. Aquests usuaris es volen comunicar específicament entre ells. Per exemple, en Pere vol parlar amb l’Anna per acordar qui farà el sopar avui o a qui li toca recollir els nens. Sovint la comunicació es dona en una finestra de diàleg que s’obre en rebre la notificació que algú altre vol parlar amb nosaltres o quan establim una connexió seleccionant algun dels programes client d’IM existents. Tot i que aquestes comunicacions s’anomenen xat, cal diferenciar-les del xat basat en l’**IRC** (*Internet Relay Chat* o conversa interactiva per internet), en què usuaris (anònims o no, desconeguts o no) s’agrupen en sales per debatre temes i aficions o simplement per xerrar.

Funcionalitats: els sistemes de missatgeria instantània actuals permeten les comunicacions no només en format de text (el xat clàssic) sinó també usant àudio i vídeo. Depenent de les aplicacions utilitzades es proporcionen més funcionalitats i tot. Algunes d’elles són:

- **Text:** les comunicacions entre els interlocutors es produeixen amb missatges de text que s’escriuen usualment en finestres de diàleg. A mesura que un interlocutor va escrivint les frases apareixen en la finestra de tots els interlocutors.
- **Àudio:** una primera evolució del xat de text va ser incorporar àudio. D’aquesta manera els interlocutors podien parlar entre ells utilitzant la comunicació IM. Evidentment, cal disposar de micròfon i d’altaveus per poder establir aquest tipus de comunicació.
- **Vídeo:** actualment la majoria d’equips informàtics disposen de càmera, micròfon i sistema de so, cosa que permet establir comunicacions audiovisuals

en temps real. En aquestes comunicacions els interlocutors no només parlen sinó que es veuen com si fossin l'un davant de l'altre.

- **Transferència de fitxers:** una característica afegida que proporcionen sovint els clients IM és la transferència de fitxers. La transferència no es realitza a través del servidor ni és un protocol especialment dissenyat per a aquesta finalitat. Simplement, les aplicacions client permeten que en una comunicació directa entre dos interlocutors es puguin transferir fitxers usant el canal de comunicació establert.
- **Compartir escriptori:** de la mateixa manera que es comparteixen fitxers entre dos clients es poden realitzar transferències de tota mena de dades directes. Un cas concret és compartir un escriptori. Això significa que un usuari pot atorgar el permís perquè el seu escriptori sigui visible i també governable, si vol, per l'altre interlocutor. Tot i els perills de seguretat que això comporta, és un mecanisme molt pràctic per solucionar problemes a distància i per ensenyar als usuaris coses que poden fer amb el seu ordinador.
- **Ubicació en el mapa:** un servei més que es pot oferir als usuaris amb la missatgeria instantània és indicar la posició de l'usuari i localitzar la dels altres usuaris via GPS. L'usuari pot fins i tot satisfer la curiositat de veure “qui hi ha per aquí”, és a dir, detectar usuaris geogràficament propers.
- **Trucades telefòniques:** ha esdevingut molt popular fer trucades telefòniques per Internet mitjançant serveis com Skype o usant clients IM que disposen d'aquesta prestació. Es pot trucar a telèfons fixos i mòbils. Usualment aquest servei és de pagament.

2.1.1 Funcionament de la missatgeria

Per entendre el funcionament dels serveis de missatgeria instantània, cal tenir clar el funcionament dels xats i les seves diferències respecte de la missatgeria instantània.

Xat

Les comunicacions de xat permeten converses de text i àudio entre usuaris i grups d'usuaris en les clàssiques sales de xat que els servidors posen a disposició dels clients. Una sala de xat és un espai virtual on els usuaris poden dialogar entre ells en temps real. En aquestes comunicacions, el protocol usat és l'**IRC** o conversa interactiva per Internet (Internet Relay Chat). Els servidors IRC proporcionen sales virtuals on els usuaris es poden trobar i on poden xatejar usant aplicacions client de xat o el mateix navegador web. Els clients es connecten a un servidor indicant un sobrenom o *nickname*. L'accés pot ser obert a tothom (públic) o reservat a usuaris registrats. Un cop connectats, els clients poden veure quines sales hi ha i entrar en una o crear-ne de noves. Tots els usuaris d'una sala formen un xat, de manera que els missatges que generen es mostren a tots els participants.

Sovint es permet també la possibilitat d'establir una comunicació privada entre dos usuaris, en aquest cas es parla d'un *xat privat*.

En els xats IRC els usuaris s'identifiquen per sobrenoms i es troben en sales on xategen plegats. Tot i que es permet la comunicació vis-a-vis, l'objectiu principal és la comunicació col·lectiva amb els altres usuaris de la sala, “fer-la petar”, “xafardejar”. Un client es connecta a un servidor i pot comunicar-se amb els altres usuaris identificant-se pel seu sobrenom en les sales on entri. Si uns amics volen fer un xat un divendres a la tarda, han d'acordar prèviament en quin servidor i en quina sala es trobaran. Si volen establir una conversa privada cal que coneguin els sobrenoms dels seus interlocutors.

En els **xats IRC**, els usuaris s'identifiquen per **sobrenoms** i es troben en sales on xategen plegats o en diàlegs a dues bandes. Les converses permeten un cert grau d'anonimat en tant que es realitzen entre usuaris identificats per un sobrenom i no pel nom real.

Un dels problemes d'aquest model és la falta d'identificació dels usuaris, la repetició de *nicknames* i la impossibilitat de saber qui hi ha contactat. Metafòricament, es pot entendre el xat IRC com un mecanisme que permet a un usuari passejar per un edifici i parlar-se a parlar amb gent que troba en les diferents sales. El seu propòsit no és la comunicació concreta amb un destinatari concret.

Missatgeria instantània

La missatgeria instantània (IM) permet als usuaris identificar-se com a tals i mantenir converses amb un o més destinataris concrets. A diferència del model de xat IRC, no es tracta d'entrar, per exemple, a la sala “Amics de la sardana” per fer-la petar amb qui hi hagi allà, sinó que es tracta d'establir una conversa en temps real amb un destinatari o destinataris desitjats.

Per poder usar l'IM, els usuaris s'han d'identificar en el servidor; d'aquesta manera es pot consultar quins usuaris hi ha connectats al servei i establir contacte amb un d'ells o més. El servidor proporciona el servei de missatgeria instantània als seus clients, als membres registrats del seu servei, però no a altres clients, si bé es poden establir passarel·les (*gateways*) entre serveis.

El servei de **missatgeria instantània** permet als usuaris registrar-se en el servidor facilitant la identificació dels usuaris connectats i permetent el diàleg directe entre un o més usuaris escollits.

En una primera fase d'implantació de la missatgeria instantània, els proveïdors d'aquest servei utilitzaven protocols propis i privats implantats als seus servidors. Els usuaris estaven obligats a usar les seves aplicacions clients i no es podien connectar amb usuaris d'altres serveis. De fet, no existia un protocol IM com a tal, sinó que cada un anava pel seu cantó. Això va esdevenir un problema evident a mesura que les comunitats de clients van anar creixent i formaven bosses d'usuaris aïllats els uns dels altres. Molts usuaris optaven per disposar de més d'un

compte en proveïdors de missatgeria diferents per tal d'abastar el màxim possible d'interlocutors. L'inconvenient és que per a cada compte calia usar una aplicació client pròpia i, per exemple, s'havien de tenir oberts alhora els diferents clients.

Tres canvis significatius han facilitat la comunicació dels usuaris independitzant-la del servei de missatgeria usat:

- **Clients multiplataforma:** permeten als usuaris connectar-se a diferents serveis sense necessitat d'usar un client específic per a cada servei. L'aplicació client "parla" el llenguatge particular usat com a protocol per cada proveïdor de servei propietari.
- **Passarel·les:** alguns proveïdors de serveis, davant de la necessitat dels seus clients de contactar amb clients d'altres proveïdors, han establert acords de comunicació de manera que els clients d'uns i altres es poden comunicar usant passarel·les (*gateways*) que són transparents (imperceptibles) per a l'usuari. Evidentment, hi ha proveïdors que ofereixen aquest servei als seus usuaris i d'altres més preocupats a mantenir-los captius i no facilitar-los la comunicació amb usuaris d'altres serveis.
- **Jabber o XMPP:** és un protocol obert de missatgeria instantània àmpliament utilitzat i s'ha convertit en un estàndard de compatibilitat entre serveis. Quan un proveïdor de servei de missatgeria instantània diu que és Jabber o XMPP significa que permet comunicacions obertes.

Els mecanismes per comunicar usuaris clients de diferents proveïdors de serveis de missatgeria instantània poden residir en l'aplicació client o l'aplicació servidor.

- **Aplicació client:** té la capacitat de "parlar" protocols diferents. S'hi configuren els comptes que connecten amb els comptes a usar.
- **Aplicació servidor:** el proveïdor d'internet proporciona la capacitat de comunicar amb altres servidors de manera transparent per l'usuari.

Protocol XMPP o Jabber

En la secció "Annexos" del web d'aquest mòdul teniu captures dels diferents diàlegs XMPP.

Sovint al protocol de missatgeria instantània, i fins i tot al servei, se l'anomena Jabber. Aquest és el nom que es va donar al protocol desenvolupat el 1999 per proporcionar serveis de missatgeria instantània, informació de presència i manteniment de llistes de contacte. Es tractava d'un protocol obert desenvolupat per la comunitat **Jabber Open Source**. Aquest protocol va esdevenir molt popular en tractar-se d'un protocol obert, i finalment l'IETF el va convertir en un dels seus estàndards amb el nom d'**XMPP**.

El protocol **XMPP** o **Extensible Messaging and Presence Protocol** (protocol de presència i missatgeria extensible) és l'estandardització de l'IETF del protocol Jabber. Es tracta d'un protocol obert basat en XML (*eXtensible Markup Language* o llenguatge de marques extensible) que proporciona serveis de missatgeria instantània i de presència.

L'organització que va desenvolupar originalment Jabber s'ha reconvertit amb el nom XMPP Standards Foundation i continua desenvolupant versions del protocol. Per conèixer a fons la seva especificació es poden consultar els RFC 3920 a 3923, emesos inicialment per l'IETF. Actualment s'han refet amb les especificacions RFC 6120, 6121 i 6122; i aquesta darrera ha estat actualitzada per la RFC 7622.

El primer servidor que va implementar el protocol XMPP va ser Jabber.org. En tractar-se d'un protocol obert, altres proveïdors l'han anat implementant i s'ha convertit *de facto* en el protocol estàndard de passarel·la entre proveïdors que utilitzen protocols particulars.

Les seves característiques principals són:

- **Estàndard obert:** és un protocol obert, qualsevol pot utilitzar-lo sense infringir normes de propietat intel·lectual. Això ha permès que s'utilitzi àmpliament.
- **Distribuït:** es tracta d'un protocol descentralitzat. No té un servidor central, sinó que cada organització pot tenir en funcionament el seu propi servidor. Els usuaris d'un servidor es poden comunicar directament entre ells, però també amb usuaris d'altres servidors.
- **Extensible:** el protocol permet definir extensions de manera que es puguin afegir noves funcionalitats al servei. Per exemple, Jingle, que és una extensió compatible amb SIP (*Session Initiation Protocol*) per a veu, vídeo, la transferència de fitxers i altres aplicacions
- **Seguretat:** permet l'establiment de connexions segures usant SSL/TLS, STARTTLS i autenticació segura usant SASL.

El model funcional

El protocol XMPP utilitza una estructura client/servidor descentralitzada. No existeix un servidor únic per a tota la comunitat Jabber, sinó que cada domini pot tenir en funcionament el seu propi servidor. Això contrasta amb altres models d'IM, com per exemple AOL, que té un servidor centralitzat per a tots els usuaris.

La diferència amb l'XMPP és que els usuaris de Jabber d'un servei es poden comunicar amb usuaris de Jabber d'un altre servei sense que existeixi un servidor centralitzat que aglutini tota la informació. Una analogia és dir que els clients d'una companyia telefònica poden trucar sense cap problema a clients d'una altra companyia (no estan limitats a trucar només als clients de la seva mateixa companyia; el preu és un altre tema...), tan sols cal conèixer l'identificador

Una confusió important en el món de la missatgeria instantània és el terme Jabber, que s'usa tant per descriure el protocol XMPP com el programari de servidor *jabberd* (el dimoni), com una de les organitzacions que presten el servei d'IM a clients de manera gratuïta, Jabber.org.

Quan es parla d'un servidor en realitat poden ser diverses les màquines que presten el servei dins d'un nom de domini.

(número de telèfon) del destinatari. Només cal tenir un compte per poder accedir a tota la comunitat Jabber a internet. Evidentment, cada usuari es pot crear tants comptes com desitgi, igual com es fa amb els comptes de correu. En aquest model, diferents organitzacions no relacionades entre elles poden oferir el servei de Jabber als seus usuaris.

Cada usuari té un compte únic anomenat **JID** o **Jabber ID**, que l'identifica com a usuari en un servidor concret. Aquest JID es vàlid per comunicar-se amb qualsevol altre usuari Jabber, sigui quin sigui el seu servidor. El format de compte JID és `user@server.cat`.

Exemple d'ús de Jabber

L'empresa X ofereix missatgeria instantània als seus treballadors amb un servidor propi. També l'ofereixen la universitat Y i l'escola Z.

Els usuaris d'aquestes tres entitats es poden comunicar no només entre si, sinó amb tots els altres usuaris de la comunitat Jabber, simplement coneixent l'identificador JID dels destinataris.

La **comunitat Jabber** està formada per tots aquells usuaris que tenen un compte (JID) en servidors compatibles i oberts al protocol XMPP. Això permet a tots aquests usuaris comunicar-se independentment del servidor en el qual tenen el compte.

Les passarel·les o *gateways* permeten comunicar amb usuaris d'altres xarxes d'IM com si també fossin usuaris XMPP.

És a nivell de servidor que s'implementa el transport o *gateway* entre xarxes de missatgeria instantània diferents i amb protocols diferents. En el model Jabber no és l'aplicació client de missatgeria instantània la que "parla" diversos llenguatges o protocols, sinó que és el servidor el que estableix comunicacions amb servidors que implementen protocols diferents. Aquests servidors executen serveis de passarel·la o *gateway* que permeten la comunicació amb els servidors Jabber. És a dir, XMPP proporciona una interfície comuna amb la qual comunicar-se. Això permet que els usuaris es puguin comunicar amb els usuaris d'aquestes altres xarxes com si fossin també usuaris XMPP.

En resum, la comunicació entre usuaris XMPP pot ser:

- **Local:** els usuaris registrats d'un mateix servei Jabber es comuniquen entre ells a través del servidor. Aquests usuaris s'han registrat en aquest servidor. Si, per exemple, es diuen Pere i Anna i el servidor s'anomena `ioc.cat`, els seus JID són `pere@ioc.cat` i `anna@ioc.cat`. Cadascú es connecta al seu servidor i la comunicació es realitza a través del servidor o es pot generar una connexió individual entre ells (per exemple, per a les transferències de fitxers).
- **Altres servidors Jabber:** si, per exemple, `pere@ioc.cat` vol establir una sessió IM amb l'usuari `jordi@edt.cat`, el servidor del qual (`edt.cat`) també

proporciona servei XMPP, la comunicació l'estableix cada usuari amb el seu servidor. Els servidors es comuniquen entre ells per transferir la comunicació, sempre que l'administrador permeti aquests tipus de comunicació.

- **Serveis externs:** per comunicar usuaris que pertanyen a xarxes de missatgeria instantània diferents i que utilitzen protocols de comunicació diferents calen mecanismes de *gateway* entre els servidors. Com sempre, els usuaris inicien sessions en el seu servidor (cada un es connecta al seu) i són els servidors els que es comuniquen entre si realitzant una conversió dels seus protocols al format obert XMPP.

Suport XMPP

Quan va aparèixer el protocol XMPP molts dels serveis de missatgeria instantània del moment van veure una oportunitat pels avantatges que oferia un protocol obert i van començar a donar suport a XMPP, és a dir, els seus clients de missatgeria instantània van començar a ser compatibles amb aquest protocol.

Destaquen els següents:

- Google Talk
- AIM (AOL Instant Messenger)
- Yahoo! Messenger
- Skype
- Facebook

Fins i tot, Whatsapp utilitza una versió modificada del protocol XMPP, almenys quan va ser comprada per Facebook.

No obstant, per raons diverses, totes aquestes grans empreses han deixat de donar servei amb aquest protocol. Google ho va fer quan va passar de Google Talk a Google Hangouts; AOL (America Online) ha donat suport limitat a XMPP fins al 2017; Yahoo! Messenger es va substituir al 2018 per un nou servei anomenat Yahoo Together, que no va arribar a l'any de vida; Microsoft manté un suport molt limitat, i Facebook va deixar de donar-hi suport al 2014.

A part de raons comercials i estratègiques, també cal comentar alguna raó tècnica: XMPP està molt centrat en la missatgeria instantània de text, tot i que a través d'extensions permet afegir serveis tals com trucades de veu i videotrucades.

2.1.2 Clients de missatgeria

Existeixen clients de missatgeria instantània per a tots els gustos i de tots els colors. Inicialment, el proveïdors de serveis requerien la seva pròpia aplicació client,

per tant, cada proveïdor disposava d'un client diferent. Posteriorment han anat apareixent clients multiplataforma que permeten connectar amb serveis diferents “parlant” el protocol apropiat en cada cas.

Clients XMPP

Llista actualitzada de clients
XMPP:

xmpp.org/software/clients.html

Les distribucions de GNU/Linux acostumen a incorporar clients gràfics multiplataforma. Els més coneguts són:

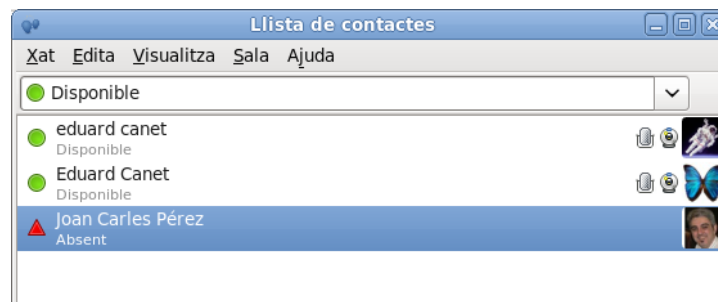
- Empathy
- Pidgin
- Gossip
- Psi
- Gajim

Empathy

Un client de missatgeria instantània actualment popular en sistemes GNU/Linux és Empathy, que permet a un mateix usuari crear múltiples comptes de missatgeria de diferents tipus. Empathy és un exemple de la capacitat multiplataforma que tenen la majoria de clients IM, ja que és capaç d'establir connexions de missatgeria instantània amb multitud de serveis que utilitzen protocols diferents.

La figura 2.2 mostra l'estat i la llista de contactes d'un usuari determinat.

FIGURA 2.2. Client de missatgeria Empathy



Les funcionalitats principals d'aquest client són:

- Veure l'estat del compte.
- Configurar un compte.
- Llistar de contactes.
- Crear grups de contactes.
- Unir-se a sales.
- Gestionar les sales preferides.
- Realitzar xats.

Estat del compte

L'usuari pot indicar quin és el seu estat o *status* en el menú desplegable. D'aquesta manera determina quina és la visibilitat que vol tenir. Pot escollir entre estar visible i disponible, no disponible o fins i tot no visible.

Els estats possibles són:

- **Disponible:** els altres usuaris poden contactar amb aquest usuari.
- **Ocupat:** l'usuari és visible per als altres usuaris, però no hi poden contactar perquè està ocupat.
- **Amagat:** els altres usuaris no poden saber quin és el seu estat, no saben si està connectat o no.
- **Absent:** l'usuari disponible amb un temps d'inactivitat en el sistema passa a estar absent (igual que passa amb l'ordinador quan entra en mode d'estalvi d'energia).
- **Desconnectat:** l'usuari no està connectat al sistema de missatgeria.
- **Missatge personalitzat:** es pot personalitzar el missatge que es vol mostrar en cada un dels estats, de manera que en lloc de veure paraula predeterminada els altres usuaris veuen un missatge personalitzat. Així, per exemple, algú molt optimista pot personalitzar l'estat *Disponible* per *A punt per a tot!*, algú altre molt enfeinat pot personalitzar l'estat *Ocupat* per un missatge com *De bòlit...*

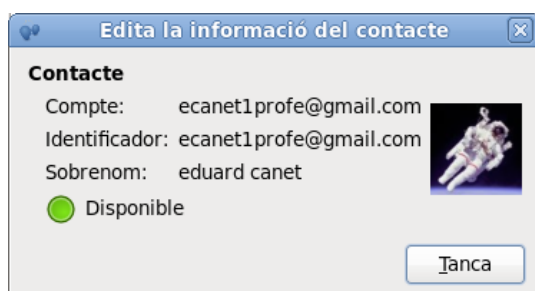
Llista de contactes

La finestra principal que es mostra en obrir Empathy és la llista de contactes de l'usuari, com es pot observar en la figura 2.2. Aquesta llista es pot editar afegint-hi i eliminant-ne contactes. Per eliminar-ne simplement cal seleccionar *suprimir*. Per afegir nous contactes cal indicar l'identificador, el JID del contacte a afegir. Es poden observar les dades de cada un dels contactes configurats seleccionant l'opció *informació*.

L'aspecte del llistat dels contactes es pot configurar amb el menú *Visualitza*, que permet indicar si el llistat és de tipus compacte, normal o amb icones, si està ordenat per nom o estat i si es mostren o no els contactes desconnectats.

La figura 2.3 mostra les dades del contacte d'un usuari.

FIGURA 2.3. Informació d'un contacte



Configuració del compte

L'element més important en la utilització d'un client de missatgeria instantània és configurar correctament el compte de l'usuari. Antigament, els usuaris utilitzaven un client propi per a cada servidor amb el qual volien connectar. Un pas posterior va ser usar clients multiprotocol (com Empathy) per poder connectar amb servidors i protocols diferents. Actualment, molts dels serveis són compatibles amb Jabber, de manera que amb un sol compte Jabber es pot accedir als contactes, sigui quin sigui el seu proveïdor de servei.

L'usuari pot crear tants **comptes de missatgeria instantània** associats al seu usuari com desitgi. Per a cada compte que crea es connecta amb el servidor pertinent i n'obté la **llista de contactes**. Un compte permet accedir a tots els contactes d'aquell servidor i de tots els altres servidors amb acords o compatibles amb **XMPP**.

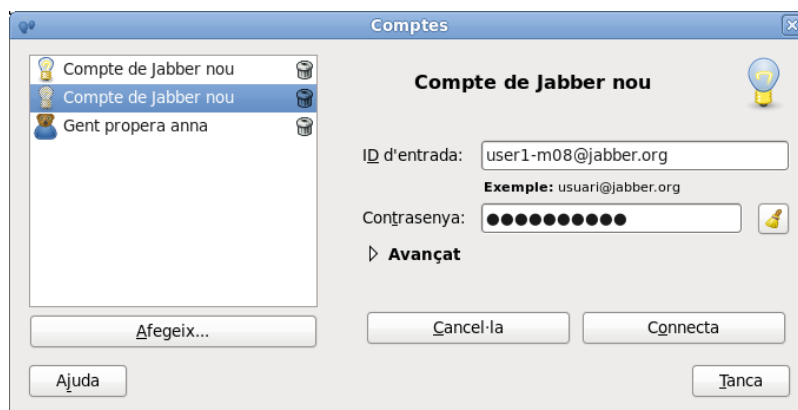
Us recordem que no cal crear un compte per a cada servidor si aquests serveis tenen entre ells un acord de connectivitat o accepten l'accés extern via XMPP.

La configuració de comptes permet:

- **Usar un compte ja existent:** si l'usuari disposa ja de comptes de missatgeria en proveïdors públics pot configurar directament el compte indicant el seu JID i les dades que siguin necessàries per accedir al servidor. Sovint n'hi ha prou amb el JID, però es pot afegir informació addicional com qüestions de seguretat, un port o nom de servidor diferent...
- **Crear un compte nou:** cal seleccionar en el menú desplegable quin tipus de compte es vol crear. Empathy permet crear comptes nous en determinats servidors de missatgeria instantània.

La figura 2.4 mostra el procés de creació d'un compte de Jabber al servidor Jabber.org. És tan senzill com indicar un JID vàlid i la contrasenya que es vol usar. Recordeu que cal fer clic a *Connecta* per crear el compte. Assegureu-vos també de seleccionar l'opció *habilitat* per poder-ne fer ús.

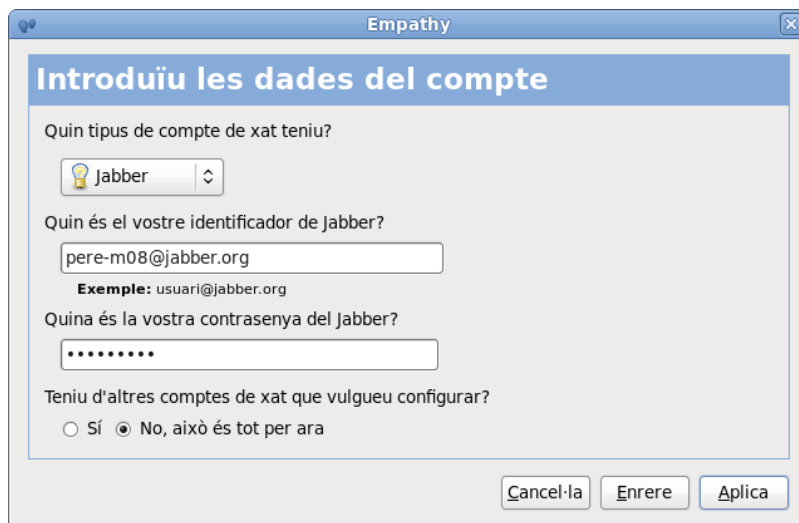
FIGURA 2.4. Creació d'un compte a Jabber.org



Quan un usuari posa en marxa per primera vegada el client Empathy de missatgeria instantània, se li permet configurar un o més comptes, ja siguin nous o preexistents

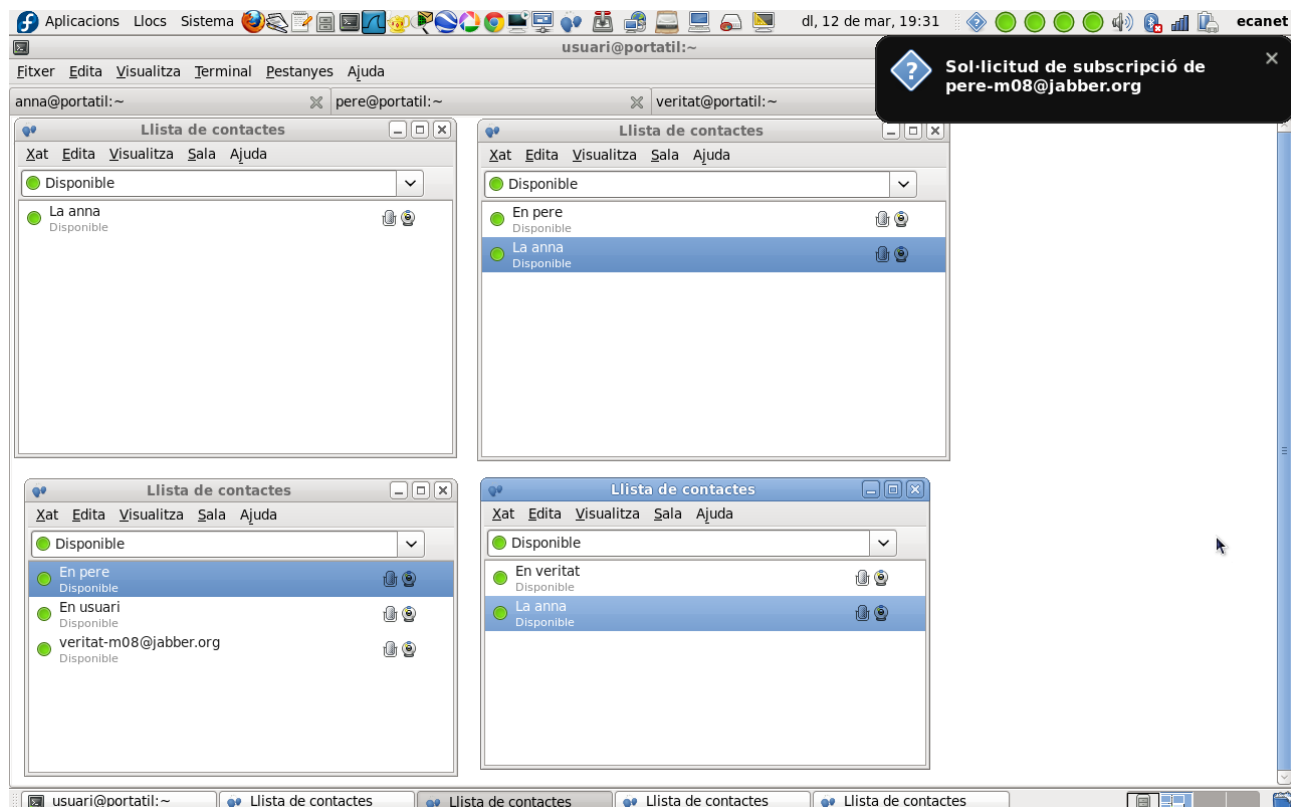
en proveïdors d'IM. La figura 2.5 mostra el procés d'utilització d'un compte Jabber ja existent iniciat quan l'usuari entra al programa per primer cop.

FIGURA 2.5. Assistent de configuració inicial



La figura 2.6 mostra quatre aplicacions client Empathy obertes des de sessions de *bash* per als usuaris “anna”, “pere”, “veritat” i “usuari”. Es pot observar a la part superior dreta de notificació dels quatre indicadors verds d'Empathy, un per a cada client. També s'observa el missatge de notificació que rep l'usuari “usuari” informant-lo que pere l'ha posat a la seva llista de contactes i demanant-li si vol fer el mateix (incorporar “pere” a la seva llista).

FIGURA 2.6. Pantalla amb quatre usuaris



Xat

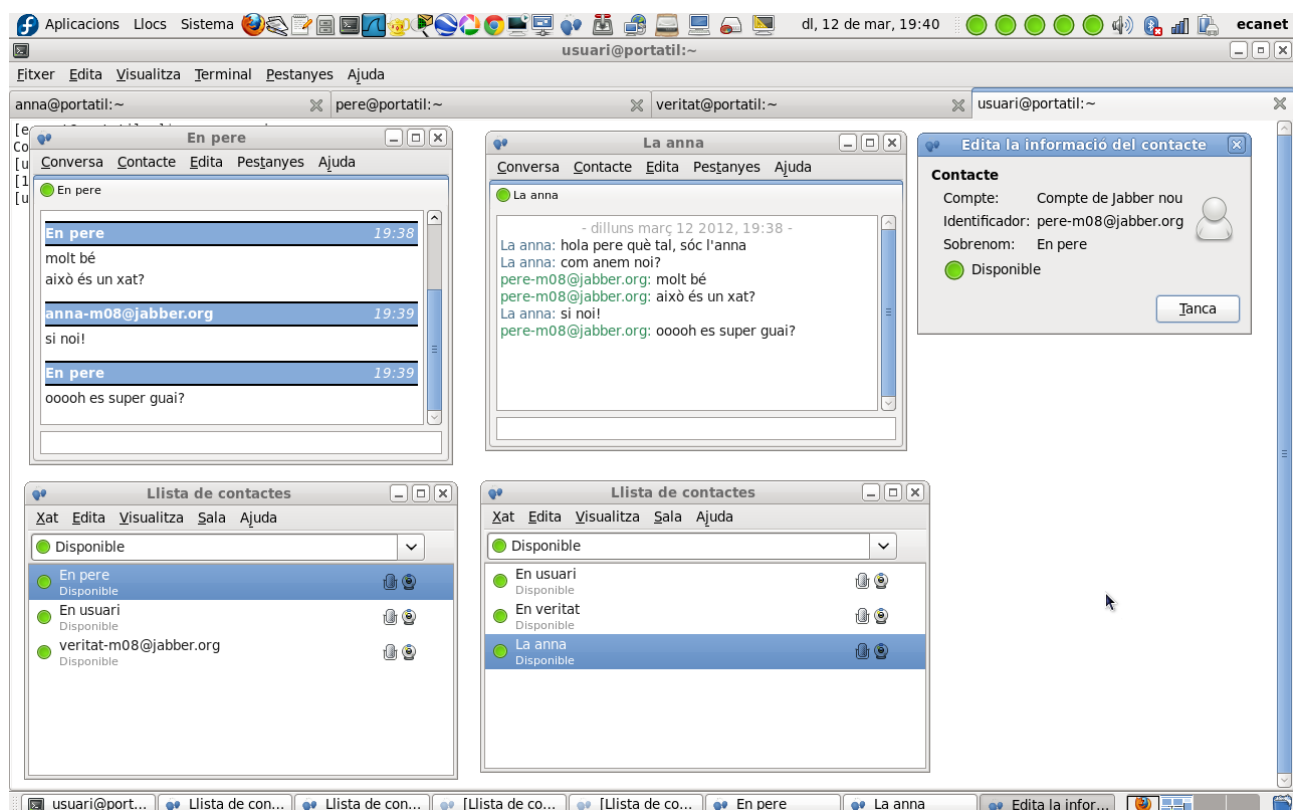
Evidentment, la principal funció d'un client de missatgeria instantània és permetre l'usuari parlar amb els altres contactes, xatejar. Els xats es poden fer seleccionant un per un els contactes amb qui parlar o seleccionant directament un grup de contactes.

Els xats poden ser de:

- Text
- Àudio
- Vídeo
- Transferència de fitxers
- Compartir l'escriptori

La figura 2.7 mostra el diàleg de text entre dos usuaris. S'ha seleccionat també l'opció que mostra la informació del contacte "pere".

FIGURA 2.7. Xat entre anna i pere



Un usuari pot obrir múltiples xats. Per a cada xat s'obre una pestanya nova en la qual es produeix la conversa. La figura 2.8 mostra com la finestra de xat de la usuària "anna" té dues pestanyes: en una dialoga amb "veritat-m08" i en l'altra amb "usuari".

FIGURA 2.8. L'Anna manté dues converses

2.2 Llistes de distribució

Una llista de distribució és una llista de membres (usuaris, clients, empleats, simpatitzants, socis, adeptes, col·legues, amics de classe, de la “mili”, jugadors de futbol, afiliats a Metges Sense Fronteres...) que reben missatges sobre un tema determinat. Algunes llistes només permeten als inscrits rebre missatges i d’altres els permeten també participar-hi enviant missatges.

Així, per exemple, quan els clients d’un supermercat es fan la targeta client esdevenen membres d’una llista del supermercat, que els envia periòdicament publicitat a casa o per correu electrònic. Si et fas soci del zoo, també reps periòdicament a casa informació d’aquesta institució, possiblement una revista trimestral i correus electrònics amb novetats del zoo i notícies sobre fauna.

Hi ha altres possibles llistes de distribució, com per exemple “Alumnes d’M08”, “Kernel Linux” o “El festeig de l’elefant marí en l’època de zel”, els membres de les quals poden participar-hi generant missatges que s’envien a la resta de destinataris de la llista.

Una **llista de correu** (o *email mailing list*) és una llista d’adreces de correu dels **subscriptors** que reben els mateixos missatges. Les llistes poden ser només de **publicació** (*announcement list*) o de **discussió** (*discussion list*).

Un mecanisme per crear llistes electròniques o *elists* són els àlies de correus o àlies de llistes d’adreces, que permeten els clients de correu com, per exemple, Thunderbird. Un àlies és un nom identificatiu assignat a un compte de correu o a un conjunt com a nom de grup. El pas següent és organitzar aquesta llista. Per fer-ho un dels membres fa la tasca de gestor o administrador, decideix qui en forma part i qui no, quins missatges s’hi poden enviar i quins no. Per facilitar la gestió de les llistes es van dissenyar programes que n’automatitzaven el funcionament. Aquests permeten gestionar la publicació i la distribució de

missatges, la subscripció i baixa i altres aspectes del seu funcionament. Això es fa amb missatges a la llista que contenen ordres adreçades al programari.

És típic que en el peu dels missatges d'una llista de distribució s'inclouï un text que expliqui com donar-se d'alta o de baixa. Són els típics missatges com “Escriu un missatge a aquesta adreça i passaràs a ser membre de la llista” o “Respon a aquesta adreça per deixar de formar part de la llista”. Actualment, les aplicacions que gestionen llistes de correu acostumen a incorporar un frontal web per a l'administració de les llistes.

Els grups de debat, discussions o fòrums poden realitzar-se en formats diferents (llistes de distribució, *news*, web...) i permeten als subscriptors la publicació de missatges. Usualment, les discussions s'emmagatzemen en servidors, s'**arxiven**. Els servidors n'indexen el contingut i en permeten la recerca posterior.

Els debats que es produeixen en les llistes de discussió normalment s'emmagatzemen o **arxiven** permanentment a internet en servidors que n'organitzen i indexen els continguts. És a dir, la majoria de les discussions, debats i missatges que publiquen els subscriptors de les llistes es conserven. És per això que quan busquem al Google “problema instal·lar sendmail en Debian” ens apareixen múltiples converses d'usuaris en fòrums de debat.

Es pot configurar un grup de discussió per rebre les **publicacions individuals**, una a una, o agrupades per períodes de temps o quantitat. Per exemple, es pot demanar un resum diari o un resum de cada 10 publicacions. En aquest cas es diu que l'usuari rep un **resum o digest**.

2.2.1 Creació d'un gestor de llistes

Existeixen diversos mecanismes per crear llistes de distribució, des de programari especialitzat fins als àlies de correu.

Els principals mecanismes per implementar llistes són:

- Àlies
- Grups de correu de serveis d'Internet: Google Groups
- Àlies de Sendmail
- Servidor de llistes

Si una entitat vol gestionar una o més llistes de distribució sense externalitzar-les i d'una manera més avançada i eficient que els simples àlies, ha d'usar algun dels

paquets de programari que fan de servidor de llistes de distribució. Existeixen diverses aplicacions que fan aquesta tasca:

- GNU Mailman: és un programari de codi lliure realitzat bàsicament amb el llenguatge de programació Python. Ha estat elaborat per programadors de GNU.
- Majordomo: és un programari que va ser àmpliament utilitzat, però que actualment no té tant seguiment.
- Listserv: és el pare de tots els programes de llistes de distribució. El seu creador va ser pioner en el desenvolupament d'aquest tipus de programes. Actualment no és de llicència pública.

El procés per instal·lar el programa que gestionarà el servidor de llistes de distribució és similar al procés seguit per a la instal·lació de la majoria de serveis del sistema. En resum, cal:

- Identificar els paquets de programari que contenen l'aplicació, descarregar-los i instal·lar-los. Es poden localitzar en els repositoris de paquets apropiats segons la distribució de GNU/Linux que s'estigui utilitzant o es pot descarregar el fitxer .tar original.
- Un cop instal·lat, cal determinar l'estat que ha de tenir el servei en cada *runlevel* (nivell d'execució) del sistema. És a dir, cal automatitzar si ha d'estar engegat o parat per defecte en cada un d'ells. Si el servei es deixa apagat per defecte, evidentment caldrà posar-lo en funcionament manualment. En el cas de GNU Mailman es tracta d'un servei autònom (*stand-alone*).
- Identificar els components de l'aplicació instal·lada. S'ha de saber determinar quins són els fitxers executables, quins els de documentació i quins els de configuració.
- Identificar el PID del servei.
- Identificar i monitorar els fitxers de registre (*log*) del servei.

El programa GNU Mailman s'acompanya d'una extensa documentació en format PDF i web disponible a `/usr/share/doc/mailman`. En concret, convé que examineu el contingut de:

- Guia d'instal·lació
- Guia d'administració de llistes
- Guia d'usuari membre d'una llista

Per a la instal·lació de Mailman, primer cal haver instal·lat com a requisits un servidor de correu i un servidor web. En aquest cas s'han fet servir **Postfix** i

Apache. S'ha d'habilitar també el mòdul **CGI** (Common Gateway Interface) per a Apache. Aquest mòdul permet al servidor web executar programes del servidor com si fossin aplicacions web i obtenir pàgines de manera dinàmica.

Per habilitar el mòdul i reiniciar el servei:

```
1 root@server:~# a2enmod cgi
2 Your MPM seems to be threaded. Selecting cgid instead of cgi.
3 Enabling module cgid.
4 To activate the new configuration, you need to run:
5     systemctl restart apache2
6
7 root@server:~# service apache2 restart
8 root@server:~# service apache2 status•
9 apache2.service – The Apache HTTP Server
10    Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset:
           enabled)
11    Active: active (running) since Mon 2019-12-02 16:30:09 CET; 4s ago
12    Process: 4663 ExecStop=/usr/sbin/apachectl stop (code=exited, status=0/
           SUCCESS)
13    Process: 4669 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/
           SUCCESS)
14    Main PID: 4674 (apache2)
15    Tasks: 56 (limit: 4915)
16    CGroup: /system.slice/apache2.service
           4674 /usr/sbin/apache2 -k start
           4675 /usr/sbin/apache2 -k start
           4676 /usr/sbin/apache2 -k start
           4677 /usr/sbin/apache2 -k start
21
22 des 02 16:30:09 server.ioc.cat systemd[1]: Stopped The Apache HTTP Server.
23 des 02 16:30:09 server.ioc.cat systemd[1]: Starting The Apache HTTP Server...
24 des 02 16:30:09 server.ioc.cat systemd[1]: Started The Apache HTTP Server.
```

Després s'instal·la el programa de gestió de llistes de distribució GNU Mailman:

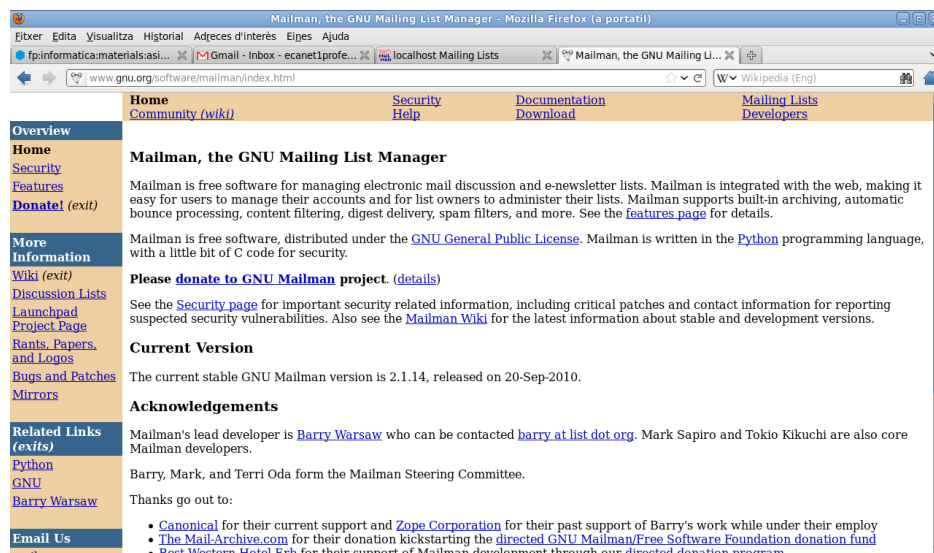
```
1 root@server:~# apt-get install mailman
2 S'està llegint la llista de ...paquets Fet
3 S'està construint l'arbre de dependències
4 S'està llegint la informació de l'...estat Fet
5 S'instal·laran els següents paquets extres:
6     python-dnspython
7 Paquets suggerits:
8     spamassassin lynx listadmin
9 S'instal·laran els paquets NOUS següents:
10    mailman python-dnspython
11 0 actualitzats, 2 nous a instal·lar, 0 a suprimir i 337 no actualitzats.
12 S'ha d'obtenir 4568 kB d'arxius.
13 Després d'aquesta operació s'empraran 39,8 MB d'espai en disc addicional.
14 Voleu continuar? [S/n]
```

Observeu que el servei no arranca bé, ja que falta configurar-lo:

```
1 root@server:~# service mailman status•
2 mailman.service – LSB: Mailman Master Queue Runner
3    Loaded: loaded (/etc/init.d/mailman; generated; vendor preset: enabled)
4    Active: active (exited) since Mon 2019-12-02 16:33:51 CET; 25s ago
5    Docs: man:systemd-sysv-generator(8)
6
7 des 02 16:33:51 server.ioc.cat systemd[1]: Starting LSB: Mailman Master Queue
           Runner...
8 des 02 16:33:51 server.ioc.cat mailman[7555]: Site list for mailman missing (
           looking for list named 'mailman'). ... (warning).
9 des 02 16:33:51 server.ioc.cat mailman[7555]: Please create it; until then,
           mailman will refuse to start. ... (warning).
10 des 02 16:33:51 server.ioc.cat systemd[1]: Started LSB: Mailman Master Queue
           Runner.
```


La figura 2.9 mostra la pàgina principal del web de GNU Mailman, on es pot trobar tota la documentació i versions de l'aplicació.

FIGURA 2.9. Pàgina principal del web de GNU Mailman



Arrancada del servidor Mailman

Un cop realitzat el procés d'instal·lació cal assegurar-se que tots els components estan configurats apropiadament i realitzar petites tasques de configuració abans de poder posar en marxa el servei i utilitzar les llistes de distribució.

S'aconsella realitzar els passos següents:

1. Comprovar usuaris i grups creats.
2. Identificar el directori base.
3. Comprovar els permisos de fitxers i directoris.
4. Comprovar la configuració del Mailman en el servidor web (en aquest cas, Apache).
5. Reiniciar el servei d'Apache.
6. Crear la llista principal o per defecte.
7. Assignar la contrasenya d'administració del servei.
8. Engegar el servei.
9. Verificar l'accés via web a l'administració de les llistes.

1. Es crea un usuari i un grup anomenats *list* amb els quals s'executa el servei.

```

1 root@server:~# grep "list" /etc/passwd
2 mailman:x:38:38:GNU Mailing List Manager:/usr/lib/mailman:/sbin/nologin
3
4 root@server:~# grep "list" /etc/group
5 mailman:x:38:

```

2. El directori base que conté l'aplicació és:

```

1 /usr/lib/mailman

```

Dins d'aquest directori hi ha una estructura de subdirectoris amb tots els elements necessaris per al seu funcionament, tant la part d'aplicació web com la configuració o els executables en la línia d'ordres.

Aquesta és l'estructura de directoris que neix del directori base (es mostra només un nivell de profunditat):

```

1 root@server:~# tree -L 1 /usr/lib/mailman/
2 /usr/lib/mailman/
3  — bin
4  — cron
5  — mail
6  — Mailman
7  — scripts
8
9 5 directories, 0 files

```

3. El següent pas és comprovar els permisos dels fitxers i directoris:

```

1 root@server:~# /usr/lib/mailman/bin/check_perms

```

Si sorgeixen problemes es pot seguir la indicació i usar l'opció *-f* per intentar solucionar-los.

```

1 root@server:~# /usr/lib/mailman/bin/check_perms -f
2 Problemes trobats: 14
3 Re-executa com mailman (o root) amb la senyal -f per a fixar

```

setgid (set group ID)

Flag d'accés d'un fitxer que dona permís a l'usuari a executar un executable amb els permisos d'execució del grup al qual pertany el fitxer. L'equivalent per a l'usuari és el **setuid** (set user ID).

No obstant, aquesta opció no sempre acaba resolent tots els problemes. En aquest cas hi ha problemes de fitxers que no pertanyen al grup corresponent i no tenen activat el setgid:

```

1 root@server:~# chgrp list /usr/lib/cgi-bin/* -R
2 root@server:~# chgrp list /var/lib/mailman/* -R
3 root@server:~# chmod g+s /var/lib/mailman/cgi-bin/* -R
4 root@server:~# /usr/lib/mailman/bin/check_perms
5 No s'han trobat problemes

```

4. Cal enllaçar GNU Mailman amb Apache. A /etc/mailman/apache.conf hi ha un exemple de configuració. La part que no està comentada s'afegeix a /etc/apache2/apache2.conf just després del darrer <Directory>.

```

1 ...
2 ScriptAlias /cgi-bin/mailman/ /usr/lib/cgi-bin/mailman/
3 Alias /pipermail/ /var/lib/mailman/archives/public/
4 Alias /images/mailman/ /usr/share/images/mailman/
5

```

```

6 <Directory /usr/lib/cgi-bin/mailman/>
7     AllowOverride None
8     Options ExecCGI
9     AddHandler cgi-script .cgi
10    Require all granted
11 </Directory>
12 <Directory /var/lib/mailman/archives/public/>
13     Options FollowSymLinks
14     AllowOverride None
15     Require all granted
16 </Directory>
17 <Directory /usr/share/images/mailman/>
18     AllowOverride None
19     Require all granted
20 </Directory>
21 ...

```

5. Cal reiniciar el servei web perquè tinguin efecte els canvis:

```

1 root@server:~# service apache2 restart
2 root@server:~# service apache2 status•
3 apache2.service – The Apache HTTP Server
4     Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset:
5           enabled)
6     Active: active (running) since Mon 2019-12-02 16:43:25 CET; 21ms ago
7     Process: 7834 ExecStop=/usr/sbin/apachectl stop (code=exited, status=0/
8           SUCCESS)
9     Process: 7840 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/
10           SUCCESS)
11    Main PID: 7846 (apache2)
12     Tasks: 56 (limit: 4915)
13    CGroup: /system.slice/apache2.service
14            7846 /usr/sbin/apache2 -k start
15            7849 /usr/sbin/apache2 -k start
16            7850 /usr/sbin/apache2 -k start
17            7852 /usr/sbin/apache2 -k start
18
19 des 02 16:43:25 server.ioc.cat systemd[1]: Stopped The Apache HTTP Server.
20 des 02 16:43:25 server.ioc.cat systemd[1]: Starting The Apache HTTP Server...
21 des 02 16:43:25 server.ioc.cat systemd[1]: Started The Apache HTTP Server.

```

6. Cal crear una llista principal per al funcionament del servei. La llista s'anomena usualment *mailman*.

7. Cal crear la contrasenya de l'administrador del servei (el *root* del servei).

```

1 root@server:~# newlist mailman
2 Introduïu l'adreça electrònica de l'encarregat de la llista: admin@server.ioc.
3   cat
4 Contrasenya inicial de mailman: peremailman
5 Haureu d'editar el fitxer /etc/aliases (o equivalent) per a finalitzar la
6 creació de la vostra llista de correu. Hi haureu d'afegir les línies
7 següents i possiblement executar el programa «»newaliases:
8
9 ## Llistes de correu mailman
10 mailman:                "|/usr/lib/mailman/mail/mailman post mailman"
11 mailman-admin:          "|/usr/lib/mailman/mail/mailman admin mailman"
12 mailman-bounces:        "|/usr/lib/mailman/mail/mailman bounces mailman"
13 mailman-confirm:        "|/usr/lib/mailman/mail/mailman confirm mailman"
14 mailman-join:           "|/usr/lib/mailman/mail/mailman join mailman"
15 mailman-leave:          "|/usr/lib/mailman/mail/mailman leave mailman"
16 mailman-owner:          "|/usr/lib/mailman/mail/mailman owner mailman"
17 mailman-request:        "|/usr/lib/mailman/mail/mailman request mailman"
18 mailman-subscribe:      "|/usr/lib/mailman/mail/mailman subscribe mailman"
19 mailman-unsubscribe:    "|/usr/lib/mailman/mail/mailman unsubscribe mailman"
20
21 Premeu la tecla de retorn per a notificar el propietari de mailman...

```

S'ha creat automàticament un conjunt de llistes. La finalitat de cada una d'elles es descriurà posteriorment. L'administrador del servei serà l'usuari admin i la contrasenya d'administració que s'ha establert és admin (caldrà posar una contrasenya més segura).

8. Arribats a aquest punt, el servei ja es pot posar en funcionament:

```

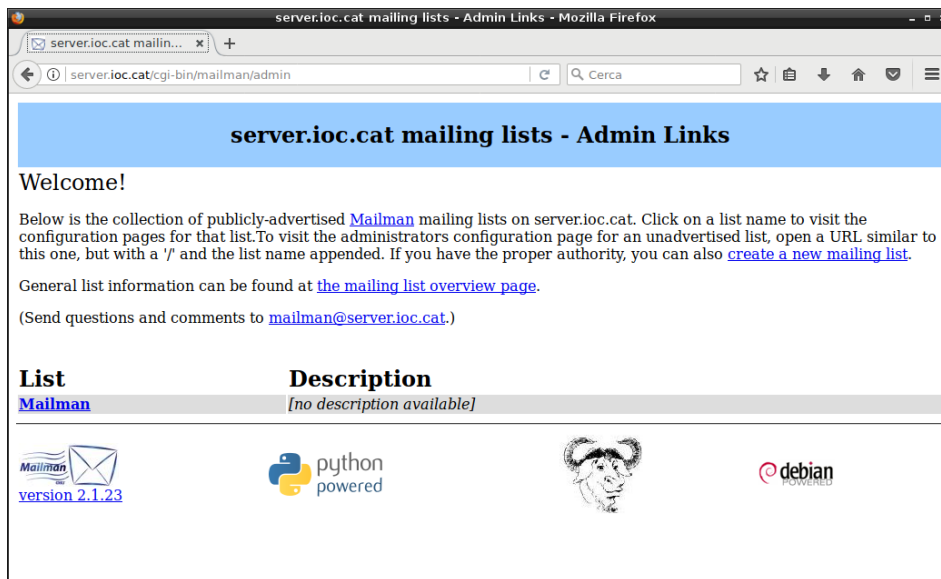
1 root@server:~# service mailman restart
2 root@server:~# service mailman status•
3 mailman.service – LSB: Mailman Master Queue Runner
4   Loaded: loaded (/etc/init.d/mailman; generated; vendor preset: enabled)
5   Active: active (running) since Mon 2019-12-02 16:43:40 CET; 72ms ago
6     Docs: man:systemd-sysv-generator(8)
7   Process: 7917 ExecStop=/etc/init.d/mailman stop (code=exited, status=0/
        SUCCESS)
8   Process: 7922 ExecStart=/etc/init.d/mailman start (code=exited, status=0/
        SUCCESS)
9     Tasks: 9 (limit: 4915)
10    CGroup: /system.slice/mailman.service
11            7930 /usr/bin/python /usr/lib/mailman/bin/mailmanctl -s -q start
12            7931 /usr/bin/python /var/lib/mailman/bin/grunner --runner=
        ArchRunner:0:1 -s
13            7932 /usr/bin/python /var/lib/mailman/bin/grunner --runner=
        BounceRunner:0:1 -s
14            7933 /usr/bin/python /var/lib/mailman/bin/grunner --runner=
        CommandRunner:0:1 -s
15            7934 /usr/bin/python /var/lib/mailman/bin/grunner --runner=
        IncomingRunner:0:1 -s
16            7935 /usr/bin/python /var/lib/mailman/bin/grunner --runner=
        NewsRunner:0:1 -s
17            7936 /usr/bin/python /var/lib/mailman/bin/grunner --runner=
        OutgoingRunner:0:1 -s
18            7937 /usr/bin/python /var/lib/mailman/bin/grunner --runner=
        VirginRunner:0:1 -s
19            7938 /usr/bin/python /var/lib/mailman/bin/grunner --runner=
        RetryRunner:0:1 -s
20
21 des 02 16:43:40 server.ioc.cat systemd[1]: Stopped LSB: Mailman Master Queue
        Runner.
22 des 02 16:43:40 server.ioc.cat systemd[1]: Starting LSB: Mailman Master Queue
        Runner...
23 des 02 16:43:40 server.ioc.cat mailman[7922]: Starting Mailman master grunner:
        mailmanctl.
24 des 02 16:43:40 server.ioc.cat systemd[1]: Started LSB: Mailman Master Queue
        Runner.
```

9. Vegeu a la figura 2.10 l'accés al lloc web local d'administració del servei Mailman. La ruta és:

```

1 http://server.ioc.cat/cgi-bin/mailman/admin
```

En aquesta pàgina es poden crear noves llistes i administrar el servei.

FIGURA 2.10. Web d'administració de Mailman

2.2.2 Creació i utilització de llistes

Hi ha diversos passos necessaris per a la creació i administració de llistes de distribució amb el programa d'entorn web de gestió llistes Mailman. Es poden dur a terme els processos següents:

- Creació d'una llista
- Subscripció a la llista
- Utilització de la llista

Exemple de creació d'una llista

En aquest primer exemple es crearà una llista anomenada “alumnes-m08”:

```

1 root@server:~# newlist alumnes-m08
2 Introduïu l'adreça electrònica de l'encarregat de la llista: jciberta@server.
   ioc.cat
3 Contrasenya inicial de alumnes-m08:
4 Haureu d'editar el fitxer /etc/aliases (o equivalent) per finalitzar la
5 creació de la vostra llista de correu. Hi haureu d'afegir les línies
6 següents i possiblement executar el programa «»newaliases»:
7
8 ## Llista de correu alumnes-m08
9 alumnes-m08:                "|/var/lib/mailman/mail/mailman post alumnes-m08"
10 alumnes-m08-admin:          "|/var/lib/mailman/mail/mailman admin alumnes-m08"
11 alumnes-m08-bounces:        "|/var/lib/mailman/mail/mailman bounces alumnes-m08"
12 alumnes-m08-confirm:        "|/var/lib/mailman/mail/mailman confirm alumnes-m08"
13 alumnes-m08-join:           "|/var/lib/mailman/mail/mailman join alumnes-m08"
14 alumnes-m08-leave:          "|/var/lib/mailman/mail/mailman leave alumnes-m08"
15 alumnes-m08-owner:          "|/var/lib/mailman/mail/mailman owner alumnes-m08"
16 alumnes-m08-request:        "|/var/lib/mailman/mail/mailman request alumnes-m08"
17 alumnes-m08-subscribe:      "|/var/lib/mailman/mail/mailman subscribe alumnes-m08"
18
19 alumnes-m08-unsubscribe:     "|/var/lib/mailman/mail/mailman unsubscribe alumnes-
   m08"

```

20 Premeu la tecla de retorn per notificar el propietari de alumnes-m08...

Un cop creada la llista es pot accedir via web a la informació de la pàgina. La figura 2.11 mostra part de la pàgina d'informació de la llista "alumnes-m08". La URL és <http://server.ioc.cat/cgi-bin/mailman/listinfo/alumnes-m08>.

FIGURA 2.11. Pàgina d'informació de la llista alumnes-m08

Alumnes-m08 --

About Alumnes-m08 English (USA)

To see the collection of prior postings to the list, visit the [Alumnes-m08 Archives](#).

Using Alumnes-m08

To post a message to all the list members, send email to alumnes-m08@server.ioc.cat.

You can subscribe to the list, or change your existing subscription, in the sections below.

Subscribing to Alumnes-m08

Subscribe to Alumnes-m08 by filling out the following form. You will be sent email requesting confirmation, to prevent others from gratuitously subscribing you. This is a private list, which means that the list of members is not available to non-members.

Your email address:

Your name (optional):

You may enter a privacy password below. This provides only mild security, but should prevent others from messing with your subscription. **Do not use a valuable password** as it will occasionally be emailed back to you in cleartext.

If you choose not to enter a password, one will be automatically generated for you, and it will be sent to you once you've confirmed your subscription. You can always request a mail-back of your password when you edit your personal options.

Pick a password:

Reenter password to confirm:

Which language do you prefer to display your messages? English (USA)

Would you like to receive list mail batched in a daily digest? ☒ No ☐ Yes

Aquesta pàgina està dividida en quatre parts:

- *About* permet accedir a la llista de missatges (*posts*) publicats en la llista.
- *Using* mostra l'adreça de correu que cal per publicar. És a dir, els subscriptors de la llista han d'enviar els seus missatges a l'adreça indicada, que en aquest cas és `alumnes-m08@localhost.localdomain`.
- *Subscribing* permet fer-se subscriptor de la llista. Per fer-ho cal indicar una adreça de correu i una contrasenya. Als subscriptors se'ls permet seleccionar l'idioma i decidir si volen rebre els missatges un a un o en forma de *digest* (agrupats per períodes de temps o nombre de missatges).
- *Subscribers* són opcions només vàlides per a subscriptors. Permeten visualitzar la llista de subscriptors, anul·lar la subscripció o modificar-ne les propietats.

L'administrador de la llista i l'administrador global poden accedir a la pàgina d'administració de la llista i configurar-ne nombrosos aspectes de funcionament. La figura 2.12 mostra algunes de les opcions de configuració.

FIGURA 2.12. Administració de la llista alumnes-m08

Alumnes-m08 mailing list administration
General Options Section

Configuration Categories

- [\[General Options\]](#)
- [Passwords](#)
- [Language options](#)
- [Membership Management...](#)
- [Non-digest options](#)
- [Digest options](#)

Other Administrative Activities

- [Privacy options...](#)
- [Bounce processing](#)
- [Archiving Options](#)
- [Mail<->News gateways](#)
- [Auto-responder](#)
- [Content filtering](#)
- [Topics](#)
- [Tend to pending moderator requests](#)
- [Go to the general list information page](#)
- [Edit the public HTML pages and text files](#)
- [Go to list archives](#)
- [Logout](#)

Make your changes in the following section, then submit them using the **Submit Your Changes** button below.

General Options

Fundamental list characteristics, including descriptive info and basic behaviors.

Description	Value
<i>General list personality</i>	
The public name of this list (make case-changes only). (Details for real_name)	Alumnes-m08
The list administrator email addresses. Multiple administrator addresses, each on separate line is okay. (Details for owner)	jciberta@server.ioc.cat
The list moderator email addresses. Multiple moderator addresses, each on separate line is okay. (Details for moderator)	
A terse phrase identifying this list. (Details for description)	
An introductory description - a few paragraphs - about the list. It will be included as html at the top of the listinfo page. Carriage returns will	

Observeu com els respectius administradors de les diferents llistes han rebut un correu indicant la creació de la llista:

```

1 root@server:~# su admin
2 admin@server:/root$ mail
3 "/var/mail/admin": 1 message 1 new
4 >N 1 mailman-owner@serv dl des 2 16:43 46/2633 Your new mailing list:
   mailman
5 ? q
6 Held 1 message in /var/mail/admin
7 admin@server:/root$ su jciberta
8 Contrasenya:
9 jciberta@server:/root$ mail
10 "/var/mail/jciberta": 1 message 1 new
11 >N 1 mailman-owner@serv dt des 3 15:28 46/2702 Your new mailing list:
   alumnes-m08
12 ? q
13 Held 1 message in /var/mail/jciberta
14 jciberta@server:/root$

```

Subscripció a la llista

Els usuaris es poden subscriure a la llista accedint al web de cada llista, per correu electrònic (a l'adreça de subscripció) o si l'administrador de la llista els inscriu.

El procés consta dels passos següents:

1. Consulta les llistes del lloc.
2. Fa la petició de subscripció.

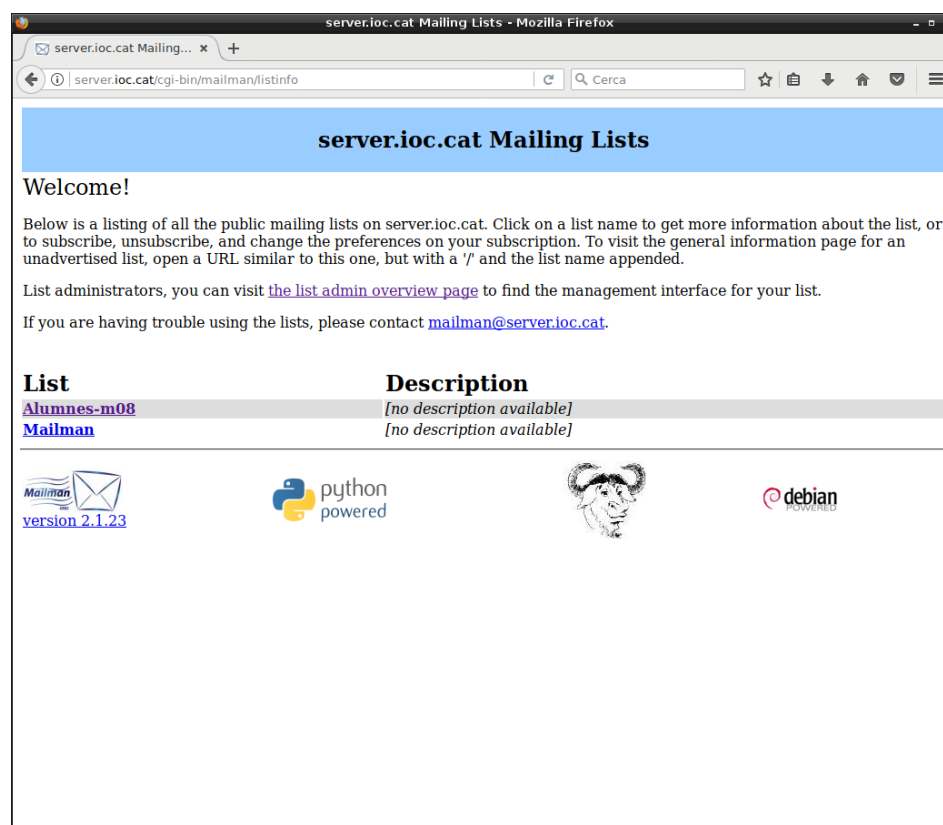
3. Rep un correu demanant la confirmació de la subscripció.
4. Confirma la subscripció per una de les tres vies proposades.
5. Estableix o modifica les condicions de subscripció.
6. Un cop confirmat, l'usuari pot començar a utilitzar la llista i a enviar missatges.

1. Es pot accedir via web a l'índex de les llistes de distribució d'un lloc web:

1 `http://server.ioc.cat/cgi-bin/mailman/listinfo`

La figura 2.13 mostra que s'han creat dues llistes. Els usuaris subscrits poden utilitzar-les per intercanviar informació realitzant els seus *posts*. Els usuaris que encara no en són subscriptors poden accedir al web per subscriure-s'hi o poden enviar un correu electrònic amb una petició de subscripció.

FIGURA 2.13. Índex de llistes públiques d'un lloc web



2. L'usuari "alumne" ha omplert una petició de subscripció via web (vegeu figura 2.14).

FIGURA 2.14. Subscripció a alumnes-m08

Alumnes-m08 --

About Alumnes-m08 English (USA)

To see the collection of prior postings to the list, visit the [Alumnes-m08 Archives](#).

Using Alumnes-m08

To post a message to all the list members, send email to alumnes-m08@server.ioc.cat.

You can subscribe to the list, or change your existing subscription, in the sections below.

Subscribing to Alumnes-m08

Subscribe to Alumnes-m08 by filling out the following form. You will be sent email requesting confirmation, to prevent others from gratuitously subscribing you. This is a private list, which means that the list of members is not available to non-members.

Your email address:

Your name (optional):

You may enter a privacy password below. This provides only mild security, but should prevent others from messing with your subscription. **Do not use a valuable password** as it will occasionally be emailed back to you in cleartext.

If you choose not to enter a password, one will be automatically generated for you, and it will be sent to you once you've confirmed your subscription. You can always request a mail-back of your password when you edit your personal options.

Pick a password:

Reenter password to confirm:

Which language do you prefer to display your messages? English (USA)

Would you like to receive list mail batched in a daily digest? ☒ No ☐ Yes

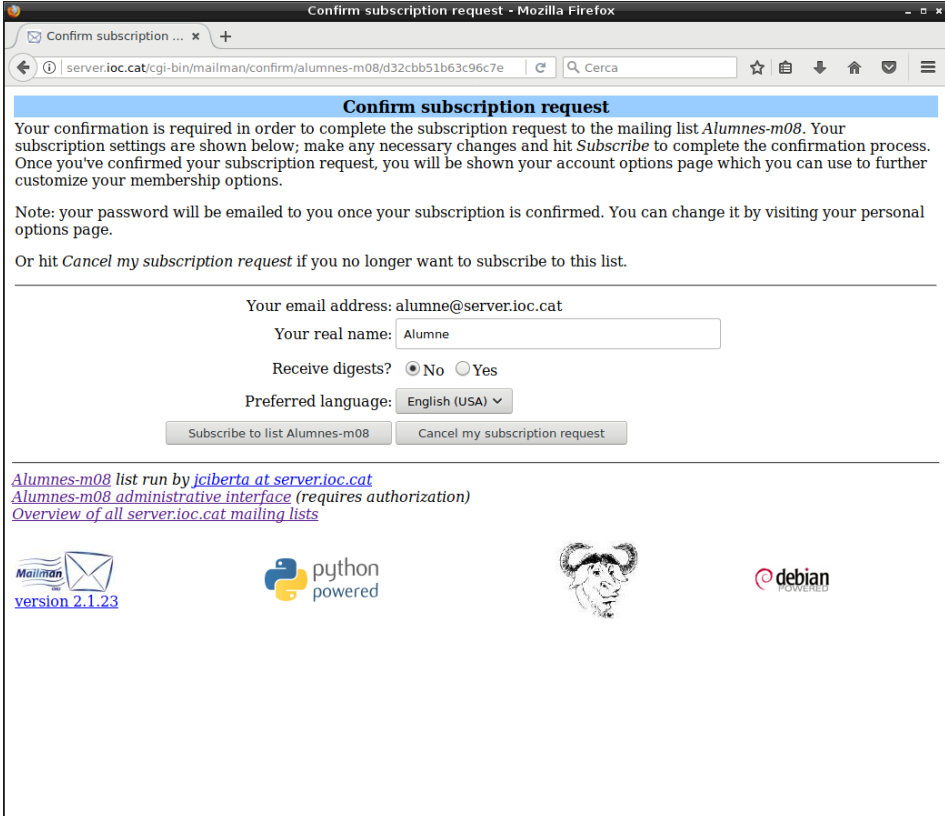
3. L'usuari "alumne" ha rebut a la seva bústia el correu electrònic de confirmació. De fet, el procés de confirmació depèn de com el configuri l'administrador de la llista. Demanar una confirmació a l'usuari és una prevenció contra la utilització fraudulenta de comptes d'altres usuaris. Quan Mailman envia un correu electrònic a l'usuari i exigeix una resposta, es verifica realment que l'usuari és qui ha fet la petició de subscripció.

```

1 root@server:~# su alumne
2 Contrasenya:
3 alumne@server:/root$ mail
4 "/var/mail/alumne": 1 message 1 unread
5 >U 1 alumnes-m08-reques dt des 3 15:52 48/2470 confirm
6   d32cbb51b63c96c7e60ed4a4b48d783d0efb074f
7 ? 1

```

4. L'alumne confirma la seva subscripció utilitzant algun dels procediments indicats en el correu electrònic del llistat anterior. La figura 2.15 mostra la pantalla de confirmació usada per l'alumne.

FIGURA 2.15. Confirmació de la subscripció

The screenshot shows a web browser window titled "Confirm subscription request - Mozilla Firefox". The address bar displays the URL: `server.ioc.cat/cgi-bin/mailman/confirm/alumnes-m08/d32cbb51b63c96c7e`. The page content is as follows:

Confirm subscription request

Your confirmation is required in order to complete the subscription request to the mailing list *Alumnes-m08*. Your subscription settings are shown below; make any necessary changes and hit *Subscribe* to complete the confirmation process. Once you've confirmed your subscription request, you will be shown your account options page which you can use to further customize your membership options.

Note: your password will be emailed to you once your subscription is confirmed. You can change it by visiting your personal options page.

Or hit *Cancel my subscription request* if you no longer want to subscribe to this list.




Your email address: `alumne@server.ioc.cat`

Your real name:

Receive digests? ☒ No ☐ Yes

Preferred language: English (USA) ▾

[Alumnes-m08 list run by `jciberta at server.ioc.cat`](#)
[Alumnes-m08 administrative interface \(requires authorization\)](#)
[Overview of all server.ioc.cat mailing lists](#)

version 2.1.23

5. Tot usuari se subscriu a una llista amb unes condicions o configuració determinada. Inicialment s'aplica la configuració de subscripció per defecte, però l'usuari pot canviar aquesta configuració per a cada una de les llistes a les quals està subscrit. La figura 2.16 mostra part d'aquestes opcions.

FIGURA 2.16. Configuració de subscripció de l'alumne a alumnes-m08

alumne at server.ioc.cat, Alumne membership configuration for Alumnes-m08 - Mozilla Firefox

alumne at server.ioc.cat, Alumne's subscription status, password, and options for the Alumnes-m08 mailing list. [Log out](#)

Changing your Alumnes-m08 membership information

You can change the address that you are subscribed to the mailing list with by entering the new address in the fields below. Note that a confirmation email will be sent to the new address, and the change must be confirmed before it is processed.

Confirmations time out after about 3 days.

You can also optionally set or change your real name (i.e. *John Smith*).

If you want to make the membership changes for all the lists that you are subscribed to at server.ioc.cat, turn on the *Change globally* check box.

New address:

Again to confirm:

Your name (optional):

[Change My Address and Name](#)

☐ Change globally

Unsubscribing from Alumnes-m08

Turn on the confirmation checkbox and hit this button to unsubscribe from this mailing list. **Warning:** This action will be taken immediately!

[Unsubscribe](#)

☐ Yes, I really want to unsubscribe

Your other server.ioc.cat subscriptions

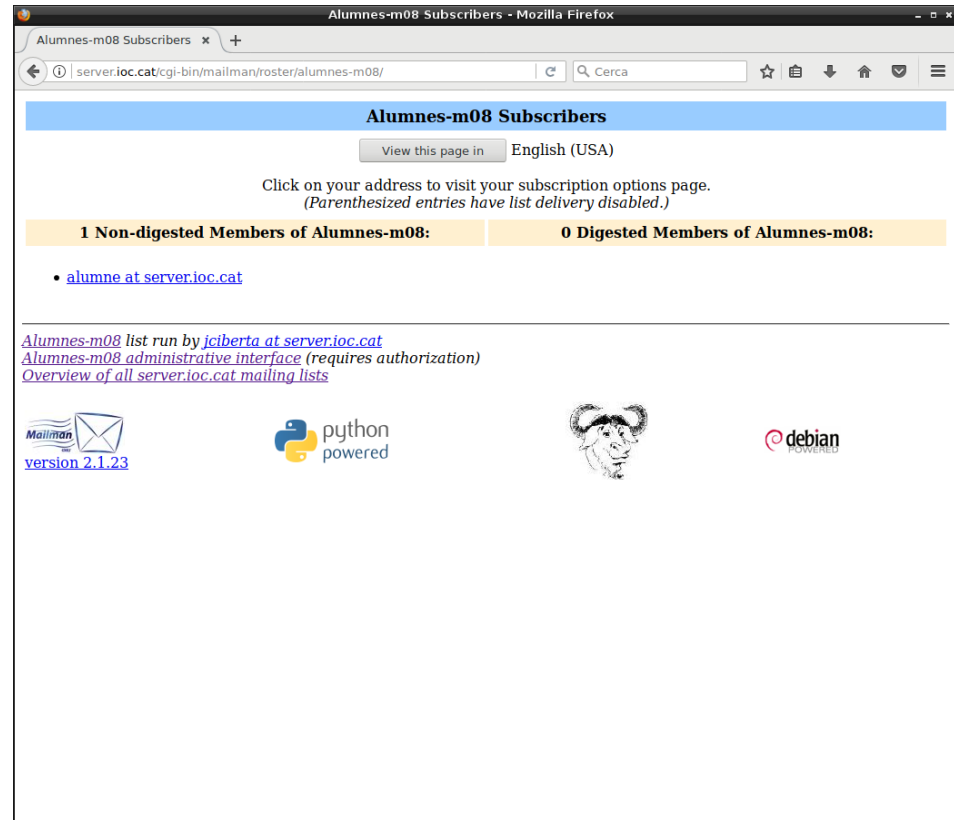
You can view a list of all the other mailing lists at server.ioc.cat for which you are a member. Use this if you want to make the same membership option changes to this other subscriptions.

[List my other subscriptions](#)

Your Alumnes-m08 Password

[Forgotten Your Password?](#) [Change Your Password](#)

6. L'alumne ja és subscriptor de la llista i pot usar-la al seu gust. Pot publicar missatges, examinar els *posts*, consultar les llistes del lloc i modificar la configuració de cada una de les seves subscripcions. La figura 2.17 mostra que l'usuari "alumne" és subscriptor de la llista i rep els missatges a mesura que es creen. També s'observa que no hi ha cap usuari que rebi els missatges en format de resum diari.

FIGURA 2.17. Llistat dels subscriptors de la llista alumnes-m08

En completar el procés de subscripció a una llista, l'usuari rep un correu electrònic del servei Mailman donant-li la benvinguda. A continuació podeu observar el missatge rebut per l'usuària "anna":

```

1 Date: Tue, 03 Dec 2019 16:01:09 +0100
2 From: alumnes-m08-request@server.ioc.cat
3 To: alumne@server.ioc.cat
4 Subject: Welcome to the "Alumnes-m08" mailing list
5
6 Welcome to the Alumnes-m08@server.ioc.cat mailing list!
7
8 To post to this list, send your message to:
9
10     alumnes-m08@server.ioc.cat
11
12 General information about the mailing list is at:
13
14     http://server.ioc.cat/cgi-bin/mailman/listinfo/alumnes-m08
15
16 If you ever want to unsubscribe or change your options (eg, switch to
17 or from digest mode, change your password, etc.), visit your
18 subscription page at:
19
20     http://server.ioc.cat/cgi-bin/mailman/options/alumnes-m08/alumne%40server.ioc
21     .cat
22
23 You can also make such adjustments via email by sending a message to:
24
25     Alumnes-m08-request@server.ioc.cat
26
27 with the word 'help' in the subject or body (don't include the
28 quotes), and you will get back a message with instructions.
29
30 You must know your password to change your options (including changing

```

31 the password, itself) or to unsubscribe without confirmation. It is:
32 ...

D'aquest missatge es pot extreure la informació següent:

- Correu electrònic per fer apunts (*posts*): `alumnes-m08@server.ioc.cat`.
- Informació de la llista: `http://server.ioc.cat/cgi-bin/mailman/listinfo/alumnes-m08`
- Cancel·lació de la subscripció: `http://server.ioc.cat/cgi-bin/mailman/options/alumnes-m08/alumne%40server.ioc.cat`
- Adreça d'ordres: `Alumnes-m08-request@server.ioc.cat`.

Utilització de la llista

Els subscriptors poden enviar missatges a la llista de distribució, poden modificar la configuració de la seva subscripció (per cada llista on estan subscrits) i poden anul·lar la subscripció. Depenent de la configuració establerta pel propietari de la llista, els usuaris no subscriptors també podran escriure en una llista. Si la llista és de tipus *anunncement only*, només el propietari o els membres autoritzats poden escriure-hi, mentre que els subscriptors poden veure els apunts però no escriure'n.

- Fer un apunt

En el cas de la llista `alumnes-m08`, els subscriptors poden fer *posts* enviant missatges a:

1 `alumnes-m08@server.ioc.cat`

- Consultar la configuració de l'usuari "alumne" a la llista `alumnes-m08`:

1 `http://server.ioc.cat/cgi-bin/mailman/options/alumnes-m08/alumne—at-server.ioc.cat`

- Llegir els apunts:

1 `http://server.ioc.cat/pipermail/alumnes-m08/`

- Anul·lar una subscripció:

```
1 http://server.ioc.cat/cgi-bin/mailman/options/alumnes-m08/alumne@server.ioc.cat
```

- Enviar ordres al motor d'ordres de Mailman:

```
1 alumne@server:~$ mail -s "comanda" Alumnes-m08-request@server.ioc.cat
2 unsubscribe alumne
3 .
```

2.3 Servei de notícies

El **servei de notícies** o **NNTP** (*Network News Transfer Protocol* o **protocol de transferència d'articles**, o més senzillament *news*) està pensat per proporcionar una funcionalitat similar als taulers d'anuncis, on tothom pot publicar i llegir els missatges que hi ha penjats. En molts aspectes s'assembla al servei de correu electrònic, però el diferencia que aquí no cal especificar un destinatari.

L'objectiu que persegueix l'NNTP és difondre articles arreu del món sense que qui els publica n'hagi d'enviar una còpia als destinataris. Els articles es publiquen en servidors que els propaguen a altres servidors. Els usuaris que volen accedir als articles utilitzen un client *news* per connectar per NNTP amb els servidors locals o remots. Els articles s'organitzen en grups segons la temàtica o àmbit territorial, per exemple, per facilitar-ne la recerca.

L'NNTP és un protocol pensat per a la distribució, consulta, cerca i publicació d'articles mitjançant TCP (port 119). Està basat en el model client/servidor.

El servei de notícies es coneix indistintament com a servei *news*, servei NNTP o servei **Usenet**. Usenet és el nom de la xarxa original d'Unix en què es va basar el primer servei de notícies. Utilitzava l'**UUCP** (*Unix to Unix Copy Protocol* o **protocol de còpia d'Unix a Unix**) per copiar els articles d'una màquina Unix a una altra. Els usuaris feien connexions locals (mitjançant trucada telefònica) als servidors locals per accedir als articles existents i deixar-hi els seus.

El protocol NNTP es descriu originàriament en el document RFC 977 de l'any 1986. La versió actual correspon a l'RFC 3977 de l'any 2006. El format dels articles es descriu en l'RFC 1036 de l'any 1987 i es basa principalment en el mateix format que els missatges de correu (document RFC 822). També es pot fer servir l'IMAP per gestionar el servei de notícies.

Els serveis de notícies NNTP estan en retrocés a causa del gran èxit del servei web (HTTP) a internet. Cada cop s'utilitzen menys els servidors de notícies i més els fòrums web, que proporcionen una funcionalitat equivalent.

Per obtenir més informació sobre l'especificació del protocol NNTP en els RFC 977, 1036 i 3977 consulteu la secció "Adreces d'interès" del web d'aquest mòdul.

2.3.1 Descripció general

Hi ha diversos mecanismes per distribuir articles a usuaris repartits per internet. Potser el més evident és el correu electrònic, mitjançant les **l·listes de distribució** o *Internet mailing lists*. L'usuari envia l'article per correu electrònic a una llista d'usuaris que creu que hi estaran interessats. L'inconvenient d'aquest model és l'ús ineficient de l'amplada de banda de la xarxa. Requereix enviar una còpia a cada destinatari. A més, es poden produir duplicacions, els usuaris pertanyen a diverses l·listes i el reben diverses vegades, canvien d'ubicació o de correu electrònic... No hi ha un mecanisme de selecció de què es propaga i de què es vol rebre.

El servei de notícies és una evolució millorada del servei Usenet original que funcionava sobre la xarxa UUCP. Aquest mecanisme obligava els usuaris a realitzar una connexió als servidors Unix amb servei de notícies per iniciar-hi una sessió local, i així poder accedir als articles. Normalment es tractava de connexions per mòdem amb trucada telefònica al servidor i exigia a l'usuari disposar d'un compte en la màquina.

El servei de notícies NNTP utilitza un repositori central d'emmagatzemament d'articles que es distribueix de manera descentralitzada a altres servidors. L'arquitectura client/servidor permet als usuaris connectar-se als servidors per gestionar els articles.

L'NNTP utilitza el model client/servidor:

- **Client.** El client demana al servidor l'article que vol veure (en lloc de baixar-los tots). Client i servidor parlen en llenguatge NNTP. L'aplicació client normalment és un programari lector/generador d'articles.
- **Servidor.** El servidor rep i envia notícies als subscriptors i dels subscriptors, i en propaga a altres servidors. En el servidor hi ha programari que permet als subscriptors seleccionar els ítems que volen. Hi ha mecanismes d'indexació, selecció, referències creuades i expiració. El servidor ofereix servei a una àrea d'influència com una LAN, un campus, una ciutat, un país... Normalment, el servidor és un programari que treballa en segon pla (*background*) en forma de dimoni. Els articles sovint s'emmagatzemen en una cua (*spool*) a la qual els subscriptors accedeixen per obtenir-los i dipositar-los. Hi ha la figura del servidor esclau (*slave server*), que manté una memòria cau de notícies per donar servei a la seva àrea.

En el **protocol NNTP** només els articles no duplicats i desitjats són transferits.

Els elements i les funcions que intervenen en el servei de notícies són:

- **Distribució.** Usenet UUCP (el model antic) utilitzava la distribució d'articles per inundació d'amfitrió *host* a amfitrió. Es feien còpies de tot a tots els

amfitrions, evidentment amb una utilització ineficient dels recursos. Amb l'NNTP es fa la selecció de què es vol rebre i enviar i a qui. El subscriptor demana la llista de novetats i baixa el que li interessa (no tot). El client també informa el servidor dels articles que vol publicar, i el servidor els accepta si no són duplicats (pot filtrar). No es garanteix que un article arribi a tots els servidors del món ni a un de concret. S'eviten els bucles controlant el camí per on es distribueixen els articles. No hi ha un únic repositori mundial dels articles, sinó que formen una base de dades distribuïda (com la informació DNS).

- **Articles.** Un article o *news* és un text que un subscriptor publica en un servidor per tal que altres usuaris el puguin llegir (tipus tauler d'anuncis). Els articles porten associat un temps d'expiració (publicació per un període de temps limitat) que pot ser establert pel redactor, el servidor o el moderador. Per facilitar la cerca d'informació, els articles s'organitzen en grups segons el tema de manera jeràrquica. Un article pot pertànyer a diversos temes. La nomenclatura dels grups va de nivell més ampli a més concret separats per un punt. Per exemple: comp.os.linux (ordinadors, sistemes operatius, Linux).
- **Administració d'articles i grups.** Els articles i els grups que hi ha a Usenet poden ser administrats per un o més **moderadors**. Hi ha grups sense administració, grups en què les decisions es prenen de manera assembleària i grups moderats. El moderador decideix si permet la publicació de l'article o no, i en pot decidir la data d'expiració. A Usenet hi ha regles i votacions per decidir la gestió (creació, eliminació, modificació) de grups de notícies a nivell global.
- **Grups.** Els articles publicats en el sistema de notícies es van organitzar jeràrquicament en grups. Es van definir set grups en el nivell principal que posteriorment es van convertir en vuit. Són els següents:
 - *Comp*: temes relacionats amb la informàtica
 - *Misc*: temes no classificables en els altres apartats
 - *News*: articles sobre el mateix sistema de notícies
 - *Rec*: activitats recreatives, aficions, jocs...
 - *Sci*: temes científics
 - *Soc*: temes socials, culturals i humanístics
 - *Talk*: debats, opinions, discussions
 - *Humanities*: discussions de temes d'humanitats, literatura, filosofia

Hi ha una jerarquia alternativa a l'estàndard en què hi ha grups que no es podien crear en la jerarquia oficial. Són una mena de grup sense normes:

- *Alt*: jerarquia alternativa a l'oficial, que conté de tot i sense normes.

L'NNTP és un protocol basat en TCP que utilitza el port 119. Com tots els protocols “vells” d'internet, no ofereix cap mena de xifratge ni privacitat en

la informació que transporta. Es pot utilitzar NNTP per SSL (modalitat que anomenem NNTPS), la qual cosa permet connexions segures. Utilitza el port 563.

El format dels articles NNTP es basa en el document RFC 1036, dedicat a Usenet. Al seu torn, aquest format es basa en el format dels missatges de correu (document RFC 822). És a dir, els articles de Usenet són en concepte similars als correus electrònics. Els articles tenen una estructura basada en un conjunt de capçaleres, una línia en blanc de separació (CRLF) i el cos o text de l'article, igual que els correus electrònics.