IMAP: ovvero Internet Message Access Protocol

POP3 è estremamente diffuso ma presenta numerose limitazioni. Il protocollo IMAP, di cui parliamo in questo articolo, è teso a fornire moderne e più ampie funzionalità rispetto al suo progenitore

Matteo Garofano

m.garofano@oltrelinux.com
Amministratore di sistema presso un Internet Service Provider, si occupa di networking e sicurezza.
Utilizza volentieri Slackware, NetBSD e Python.

La logica che sottende al protocollo POP3 è quella dell'elaborazione off-line: la posta risiede in un server che viene contattato dall'utente il quale, attraverso opportuni programmi, preleva i messaggi giacenti per poi elaborarli localmente con le operazioni di apertura, lettura, smistamento in archivi specifici e cancellazione.

In tal modo si riesce a gestire comodamente un archivio di un singolo PC, ma se l'utente si sposta ed utilizza diverse macchine per accedere alla propria posta la cosa si può complicare. E' anche possibile che una mailbox non venga usata da un singolo utente. In alcuni casi, con il protocollo POP, si può procedere scaricando i messaggi da tutte le postazioni che ne hanno bisogno lasciandone una copia sul server.

Poiché lo spazio disponibile sul server per archiviare i propri messaggi, per quanto grande sia, è limitato, non sarà possibile lasciare una copia dei messaggi per un tempo indeterminato ma occorrerà, almeno periodicamente, cancellarli. Nel caso di accesso multiplo ad una mailbox potrebbe nascere confusione circa le operazioni di cancellazione che i diversi client devono svolgere.

L'organizzazione dei propri messaggi in archivi distinti suddivisi in cartelle è poi utile purché, ovviamente, i messaggi si trovino sempre nelle stesse directory ogni volta che usiamo la nostra posta elettronica. Questa fase di smistamento dei messaggi, nel caso di un utente che accede alla mailbox tramite POP da numerosi PC, dovrà essere ripetuta da ogni postazione con grande spreco di tempo.

Per questi motivi è stato sviluppato il protocollo IMAP che attualmente è giunto alla sua versione 4. L'idea che sta alla base è quella di delegare un certo numero di operazioni al server, il quale si occuperà della gestione e dell'organizzazione dei messaggi in cartelle e sottocartelle per la posta in entrata, in uscita e in directory create dall'utente. Tale modalità operativa presuppone un continuo scambio di informazioni tra client e server e, in questo, si discosta molto dal POP. Tale ripensamento circa la logica di funzionamento è dovuta, rispetto al passato, alle mutate disponibilità di connettività di molti paesi oltre al fatto che, attualmente, moltissimi sistemi di messaggistica sono usati all'interno di aziende attraverso reti locali.

Il protocollo POP3, nonostante sia più vecchio e meno ricco di funzionalità, non è stato soppiantato dal protocollo IMAP e le ragioni sono varie, non ultima la ritrosia degli utenti verso le novità e i cambiamenti. Attualmente è possibile fornire agli utenti l'accesso ad una casella postale attraverso entrambi i servizi POP e IMAP. Nonostante il maggiore sforzo sistemistico questa soluzione senz'altro soddisferà la maggior parte di persone e ci solleverà da difficili scelte. Nel caso decideste di utilizzare uno solo dei due servizi per dare accesso alla posta elettronica, dovrete operare la scelta tenendo in considerazione diversi fattori tra i quali: varietà e tipologia degli utenti finali, client che si desidera supportare, tipologia del servizio offerto. Per meglio operare questa valutazione si possono schematicamente riassumere i vantaggi e gli svantaggi dei due sistemi come nel *riquadro 3*.

Nonostante le due tipologie di protocolli POP3 e IMAP siano pensate per due modalità d'uso diverse, numerose sono le caratteristiche che li accomunano.

Si può infatti ricordare che entrambe i protocolli supportano le operazioni off-line, non consentono l'invio dei messaggi che deve avvenire

attraverso un server (MTA, Mail Transfer Agent) e sono disponibili per un grande numero di piattaforme; inoltre, molti client commerciali ed opensource possono utilizzare entrambi i protocolli. I protocolli sono aperti e definiti attraverso i numerosi documenti RFC ed inoltre sono disponibili molti progetti opensource che consentono di accedere al codice sorgente per conoscere quali possono essere le implementazioni.

Architettura e stati di una sessione IMAP

Quando un demone IMAP viene avviato, in genere, si mette all'ascolto, in attesa di connessioni TCP alla porta 143. Alla connessione verso tale porta si inizia una sessione tra client e server che passa attraverso una sequenza di stati che possono essere descritti dai termini inglesi: *non-authenticated*, *authenticated*, *selected*, *logout*.
Vediamo di capire cosa accade nei diversi stati. Lo stato *non-authenticated* consente agli utenti che vogliono accedere al servizio di scegliere il metodo di autenticazione, usando il comando `authenticate` e di passare poi le proprie credenziali con la stringa:

```
LOGIN utente password
```

Se il processo di login ha successo, si passa allo stato *authenticated* che permette numerose operazioni nella propria mailbox. Tra queste ovviamente c'è l'accesso e la manipolazione delle cartelle. Si possono cancellare, creare, rinominare cartelle dove sono archiviati i nostri messaggi direttamente sul server.
Impartendo il comando `SELECT` si passerà allo stato *selected* e ci si troverà ad operare all'interno della cartella scelta. Con il comando `LOGOUT` si passa allo stato *logout* dove vengono liberate le risorse allocate dal server e chiusa la connessione.

Sessioni, operazioni server e client: un esempio

Vogliamo ora operare una sessione attraverso una connessione "manuale" utilizzando il protocollo IMAP, certo non è il modo più comodo, ma è senz'altro il modo migliore per capire l'interazione client-server.
In primo luogo dovremo disporre di un server IMAP a cui collegarci, magari sulla nostra macchina locale oppure al nostro provider che ci fornisce tale servizio. Se non sapete come fare, nel capitolo successivo apprenderemo come installare un server IMAP sulla propria Linux-box.
Per connetterci al server IMAP dalla linea di comando, useremo il semplice client telnet, apriremo una connessione TCP alla porta 143 e impartiremo i comandi che desideriamo eseguire.
I comandi devono essere preceduti da un identificatore, cioè un prefisso alfanumerico chiamato *tag*, necessario a far corrispondere le risposte del server alle domande inviate dal client, nel nostro esempio useremo `a1`, `a2`, `a3`, ecc.
Addentriamoci nella nostra sessione IMAP aprendo un terminale ed attivando la connessione:

RIQUADRO 1

Struttura del formato maildir

```
/var/spool/mail/m.garofano@oltrelinux.com# tree -R
.
|-- cur
|   |--  1159285485.V306I4c626.mail.oltrelinux.com:2,
|   `-- 1159284653.V306I50550.mail.oltrelinux.com:2,ST
|-- dovecot-uidlist
|-- dovecot.index
|-- dovecot.index.cache
|-- dovecot.index.log
|-- maildirsize
|-- new
|   |-- 1159285485.V306I4b678.mail.oltrelinux.com
|   |-- 1159285040.V306I502bf.mail.oltrelinux.com
|   `-- 1159285485.V306I4c626.mail.oltrelinux.com
|-- subscriptions
`-- tmp
```

La mailbox, in formato maildir, contiene 3 messaggi nuovi (cartella `new`) e due già esistenti (cartella `cur`), uno dei quali marcato con il flag "già letto" `(S)` e da cancellare "delete" `(T)`.

RIQUADRO 2

Attributi dei messaggi IMAP

Una delle forze del protocollo IMAP è quello di avere una gestione evoluta dello stato dei messaggi, attraverso l'assegnazione di diversi flag si riesce ad avere una gestione centralizzata della posta e non più client-centrica.
I flag, caratterizzati dal carattere "`\`"anteposto al codice che identifica lo stato, possono essere:

`\Answered`: indica che al messaggio è stata inviata una risposta (reply);

`\Deleted`: il messaggio è stato segnato affinché venga cancellato;

`\Draft`: il messaggio è in fase di stesura è quindi considerato come una bozza;

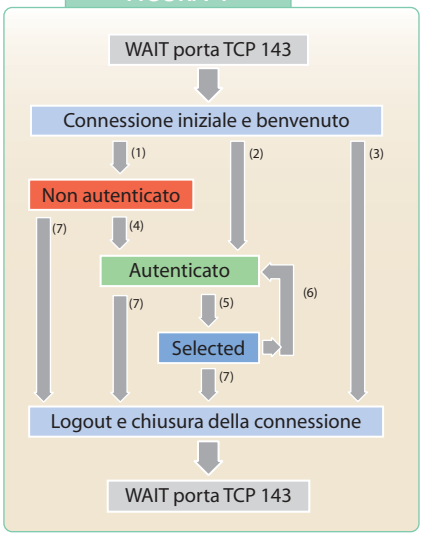
`\Flagged`: il messaggio è impostato come "urgente";

`\Recent`: indica che il messaggio è nuovo pertanto ancora da leggere;

`\Seen:` questo flag significa che il messaggio è stato già letto.

FIGURA 1

Stati di una sessione IMAP



(1) connessione senza autenticazione (OK *messaggio di benvenuto*);
(2) connessione pre-autenticata (PREAUTH *benvenuto*);
(3) connessione rifiutata (BYE *messagggio*);
(4) LOGIN avvenuta o comando di autenticazione;
(5) operazione di SELECT avvenuta o comando di EXAMINE;
(6) comando di chiusura CLOSE o errore nell'operazione di SELECT o EXAMINE;
(7) comando di LOGOUT, shutdown del server o chiusura della connessione per interruzione o timeout;

RIQUADRO 3

Vantaggi e svantaggi

Per riassumere i vantaggi e gli svantaggi delle modalità off-line, tipica del protocollo POP, e quella on-line a cui si riferisce il protocollo IMAP si può dire:

Vantaggi dell'*off-line*:

- riduzione al minimo dei tempi di uso on-line;
- minimo utilizzo di risorse da parte del server.

Vantaggi dell'*on-line*:

- possibilità di usare più postazioni diverse in tempi diversi;
- possibilità di accesso concorrenziale a mailbox condivise;
- possibilità di usare macchine senza storage di dati (ad esempio in laboratori) o postazioni diskless.

```
$ telnet localhost 143
```

ci viene data la possibilità di interagire con il server IMAP, se volessimo conoscere di cosa "è capace" il nostro server possiamo lanciare il comando:

```
a1 CAPABILITY
* CAPABILITY IMAP4rev1 SORT MULTIAPPEND UNSELECT
 LITERAL+ IDLE NAMESPACE QUOTA AUTH=PLAIN AUTH=CRAM-MD5
a1 OK Capability completed.
```

Alcune delle capabilities sono facilmente comprensibili (SORT, AUTH, IDLE, QUOTA) altre sono estensioni recenti che consentono di compiere operazioni di sincronizzazione diverse (LITERAL+, MULTIAPPEND) o indicazioni su quali cartelle, dell'utente, di altri o condivise, sono accessibili e attraverso quali prefissi (NAMESPACE) o che consentono di ritornare dallo stato *Selected* allo stato *Authenticated* (UNSELECT). Procediamo poi ad autenticarci:

```
a2 LOGIN matteo ilmiosegreto
```

A seguito di una corretta procedura di login il nostro server ci segnala che siamo autenticati:

```
a2 OK Logged in
```

Elenchiamo le cartelle presenti nella nostra mailbox:

```
a3 list "" "*"
* LIST (\HasChildren) "." "Posta inviata"
* LIST (\HasNoChildren) "." "Bozze"
* LIST (\HasNoChildren) "." "Cestino"
* LIST (\HasNoChildren) "." "Posta inviata.prove"
* LIST (\Unmarked) "." "INBOX"
a3 OK List completed.
```

Qui il server ci evidenzia che quattro nostre cartelle non hanno sottocartelle (HasNoChildren) mentre una le ha (HasChildre) nel nostro caso la cartella prove è contenuta in "Posta inviata". Scegliamo di operare nella cartella predefinita per i nuovi messaggi:

```
a4 select INBOX
```

Una lunga risposta ci viene fornita:

```
* FLAGS (\Answered \Flagged \Deleted \Seen \Draft)
* OK [PERMANENTFLAGS (\Answered \Flagged \Deleted
                      \Seen \Draft \*)] Flags permitted.
* 14 EXISTS
* 11 RECENT
* OK [UNSEEN 3] First unseen.
* OK [UIDVALIDITY 1148397735] UIDs valid
* OK [UIDNEXT 15] Predicted next UID
a4 OK [READ-WRITE] Select completed.
```

Il parametro FLAGS ci indica quali sono i flag che possono essere usati nella cartella (Answered, Flagged, Deleted, Seen, Draft) e questi non possono essere modificati dal client (PERMANENTFLAGS). Visualizziamo alcune informazioni del messaggio 5:

```
a5 fetch 5 fast
```

```
* 1 FETCH (FLAGS (\Seen) INTERNALDATE "24-Jul-2006
                        17:48:01 +0200" RFC822.SIZE 1888)
a5 OK Fetch completed.
```

Tra le altre informazioni relative al messaggio, il flag `Seen` ci indica che il messaggio è già stato letto. Ora vogliamo leggere il soggetto dei messaggi 3 e 4:

```
a6 fetch 3:4 (flags body[header.fields (subject)])

Subject: test
)
* 3 FETCH (FLAGS (\Seen) BODY[HEADER.FIELDS
                                        (SUBJECT)] {22}
Subject: PROVA Oltrelinux
)
* 4 FETCH (FLAGS (\Seen) BODY[HEADER.FIELDS
                                        (SUBJECT)] {48}
Subject: Caro amico ti scrivo
)
a6 OK Fetch completed.
```

Segniamo come da cancellare il messaggio 3 utilizzando il comando `STORE` e il flag `Deleted`:

```
a7 STORE 3 +FLAGS (\Deleted)
* 2 FETCH (FLAGS (\Deleted))
a7 OK Store completed.
```

E procediamo poi alla cancellazione dei messaggi precedentemente marcati come "da cancellare":

```
a8 EXPUNGE
* 2 EXPUNGE
a8 OK Expunge completed.
```

L'operazione è andata a buon fine, terminiamo "educatamente" la nostra sessione:

```
a9 LOGOUT
* BYE Logging out
a9 OK Logout completed.
```

Se qualcuno volesse utilizzare un canale criptato per svolgere queste prove o per accedere alla propria posta in condizioni di sicurezza non dovrà far altro che utilizzare `openssl` e connettersi alla porta 993 del server di suo interesse:

```
$ openssl s_client -connect localhost:993
```

RIQUADRO 4

Comandi in stato Authenticated

Appena autenticati viene fornita la possibilità di svolgere numerose operazioni sulle cartelle disponibili.

SELECT `cartella`: seleziona la mailbox con cui lavorare

EXAMINE `cartella`: seleziona la mailbox ma con accesso in sola lettura

CREATE `cartella`: crea una nuova cartella

DELETE `cartella`: cancella una cartella

RENAME `vecchio-nome-cartella nuovo-nome-cartella`: rinomina una cartella

SUBSCRIBE `cartella`: aggiunge la cartella alla lista di quelle attive da visualizzare in un client IMAP

UNSUBSCRIBE `cartella`: elimina la cartella dalla lista delle caselle attive da visualizzare in un client IMAP

LIST `riferimento cartella`: restituisce un sottoinsieme dei nomi disponibili al client

LSUB `riferimento cartella`: restituisce un sottoinsieme delle casella sottoscritte dall'utente

STATUS `cartella stato`: indica lo stato della cartella

APPEND `cartella messaggio`: aggiunge un messaggio alla cartella selezionata

RIQUADRO 5

Comandi in stato Selected

I messaggi gestiti dal protocollo IMAP vengono organizzati in cartelle e possono essere manipolati secondo numerose operazioni descritte qui di seguito:

CHECK: consente la manutenzione del server

EXPUNGE: rimuove i messaggi con il flag `delete` attivo, rimane poi nello stato SELECTED;

SEARCH `pattern parametri`: permette una ricerca in base ai parametri indicati;

FETCH `messaggi dati`: visualizza tutto o parte (soggetto, data, contenuto) dei messaggi scelti;

STORE `messaggi flags`: in genere usato per modificare gli attributi (flag) dei messaggi;

COPY `messaggi cartella`: non esiste una operazione di spostamento (move) ma bisogna operare una copia e una rimozione dell'originale. La copia si esegue con il comando `COPY`;

UID `comando argomenti`: usato per operare sui messaggi basandosi sull'identificativo UID piuttosto che sul suo numero;

CLOSE: fa entrare in AUTHENTICATED state e cancella i messaggi con flag DELETE della cartella corrente.

Alcuni server IMAP

La disponibilità di server IMAP di tipo opensource non è altrettanto ampia rispetto a quella dei server POP3. Nonostante ciò, esistono alcuni progetti ormai consolidati e molto utilizzati, mentre altri stanno crescendo negli ultimi anni e, pur soffrendo di una scarsa maturità, offrono funzionalità utili.
Tra i progetti più interessanti e diffusi figurano sicuramente **courier-imap**, **cyrus-imap**, **uw-imap** e **dovecot**.
Courier-imap è uno dei componenti che fanno parte della suite **courier-mta** il ben noto sistema completo di posta elettronica sviluppato dalla Inter7, offre, oltre al servizio IMAP, anche quello POP3.
Il progetto **cyrus-imap,** da lungo tempo seguito dalla Carnagie Mellon University, offre un server IMAP molto maturo e completo di numerose caratteristiche ed opzioni.
Il server **uw-imap** è sviluppato dall'università di Washington e viene fornito insieme a numerosi altri tool tra cui una libreria per costruire client IMAP, un **MDA** (Mail Delivery Agent), un server POP2 e POP3 ed altro ancora.
Dovecot è un server IMAP di moderna architettura, scalabile, flessibile, sicuro e facilmente amministrabile. Lo abbiamo già utilizzato e cominciato a conoscere nella puntata dedicata ai server POP e lo utilizzeremo nuovamente per fornire il servizio IMAP.
Attualmente sono pochi i server IMAP liberamente disponibili in rete che incorporano tutte le caratteristiche messe a disposizione da Dovecot. Dovecot è scritto in C e sviluppato tenendo in primo piano gli aspetti della sicurezza. Ma oltre a ciò presenta la possibilità di usare sistemi di autenticazione multipli schierando tra questi i sistemi PAM e LDAP e interfacciandosi nativamente a database SQL come MySQL, PostgreSQL e altri ancora oltre ovviamente ai classici file `passwd` e `shadow` di sistema anche in mix tra di loro.
Dovecot dispone già della possibilità di comunicazioni crittografate (SSL/TSL) ed è pronto all'uso su reti che utilizzano il protocollo IPV6; inoltre, tra le caratteristiche più importanti che possono aiutare nella migrazione "morbida" da un sistema di posta elettronica ad un altro, sono supportati entrambi i formati Mailbox e Maildir.
Anche per i server IMAP conviene verificare quali siano le esigenze da soddisfare in termini di servizi da offrire, di integrazione con sistemi già presenti e futuri, di scalabilità e di flessibilità. Sarà opportuno prevedere se si intenderà fornire anche il servizio POP3, di quale formato di storage delle E-Mail si dispone o si vorrà disporre, quale tipo di autenticazione si vuole supportare e dove si desidera archiviare le credenziali: ad esempio utilizzando Kerberos, LDAP, MySQL, file `passwd`, PAM.
Vista la minor disponibilità di server IMAP rispetto ai server POP3 talvolta la scelta è obbligata, in particolare quando si deve integrare il servizio in sistemi già esistenti, più facile è la scelta per coloro che devono implementare un servizio "da zero".

Una semplice configurazione di IMAP con canale SSL

Ci apprestiamo ora a passare alla fase operativa, dove mettiamo in funzione un server IMAP per consentire il prelievo della posta secondo la modalità che abbiamo visto. Supponiamo uno scenario piuttosto semplice, ma tutt'altro che inverosimile, che ci consentirà in maniera rapida di fornire l'accesso alle mailbox attraverso il servizio IMAP.
Utilizzeremo il demone Dovecot e rimandiamo alle istruzioni date nel numero scorso di Linux&C (L&C54) per una semplice installazione o consigliamo di utilizzare i binari precompilati per le propria distribuzione.
Trattandosi di una installazione basilare, per l'autenticazione ci appoggeremo ai tradizionali file `passwd` e `shadow` mentre i messaggi saranno archiviati nel nostro filesystem nel formato maildir e le mailbox saranno collocate all'interno di una directory piuttosto tipica (`/var/spool/mail/`).
Ci preoccuperemo inoltre, di rendere disponibile un canale crittografato che consentirà agli utenti di far transitare in sicurezza sulla rete le proprie credenziali e i propri messaggi, attiveremo infatti il protocollo IMAPS.
Con una tale configurazione un "normale" PC potrà fornire il servizio IMAP ad alcune migliaia di utenti, magari fornendo anche il servizio SMTP su un'unica macchina.
Interessante notare che il tipo di formato di immagazzinamento mailbox o maildir, in caso di una configurazione "standard", così come viene impostato dagli MTA più diffusi (`sendmail`, `exim`, `postfix`, `qmail`), viene rilevata automaticamente altrimenti potremo impartire le nostre preferenze.
Terminate le configurazioni, non ci resta che far ripartire il servizio con:

```
$ sudo /etc/init.d/dovecot restart
```

Attivazione canale sicuro IMAPS

Una interessante caratteristica di alcuni server IMAP, tra cui Dovecot, è quella di consentire di instaurare delle sessioni protette attraverso canali criptati.
Tale modalità, chiamata **TLS** dall'inglese **T**ransport **L**ayer **S**ecurity, ci fornisce un notevole incremento nella sicurezza delle comunicazioni tra client e server poiché andremo a

RIQUADRO 6

Formati di storage dei messaggi: Mailbox vs Maildir

Tra gli amministratori di mailserver talvolta si discute su quale sia il miglior formato di archiviazione dei messaggi di posta elettronica, questo poiché ne esistono diversi quali ad esempio mailbox, maildir, mbx, mailstore, dbox. Tra i più affermati si individuano il formato Mailbox, noto anche come mbox, e quello Maildir: il primo famoso per essere un formato "storico" dei sistemi *nix, da lungo tempo utilizzato, il secondo per essere stato introdotto insieme allo sviluppo del famoso MTA Qmail, noto per la sua solidità. Come spesso succede nel confrontare due diversi oggetti o software è difficile, se non impossibile, dichiarare un "vincitore assoluto" cioè capire quale dei due è sempre superiore all'altro. Ciò avviene poiché i risultati variano in funzione delle condizioni di utilizzo. Queste ultime spesso non sono conosciute a priori o addirittura variano nel tempo. Nel caso dei sistemi di posta elettronica e di storage dei messaggi le variabili in gioco sono molte e difficilmente prevedibili a priori. Ad esempio per ciascuna mailbox possono variare la dimensione dei messaggi, il numero di messaggi, il numero di messaggi che vengono cancellati o lasciati sul server, la frequenza degli accessi. Senza avere la pretesa di individuare "il miglior" formato di archiviazione cerchiamo di capire quali siano le loro caratteristiche.

Formato mailbox (caratteristiche valide anche per il formato mbx):

tutti i messaggi sono archiviati in un unico file, ciascun messaggio inizia con un testo formattato dalla stringa "`From: mittente@dominio.com`" e termina con una linea vuota.

Pregi: formato compatto, molto testato, supportato da quasi tutti gli MTA e da molti server IMAP e POP3, facilità di manipolazione dell'intera mailbox da parte dell'amministratore di sistema.

Difetti: fragile, in caso di crash può lasciare la mailbox in uno stato di incoerenza, non è adatto a filesystem di rete o distribuiti in quanto è necessario bloccare il file durante la lettura dei messaggi.

Performance: buone se tutti messaggi vengono prelevati e cancellati dalla mailbox, avviene una sola lettura di un singolo file; **scarse** in caso di modifica del contenuto di grandi mailbox, avviene la riscrittura di un grande file a seguito anche di piccole modifiche come ad esempio la cancellazione di un singolo messaggio.

Formato maildir:

ciascun messaggio è salvato in un singolo file, i file sono organizzati all'interno di una cartella che contiene le sottocartelle `new/` `cur/` e `tmp/`. La cartella `new/` contiene i messaggi appena arrivati, la cartella `cur/` quelli già letti, la cartella `tmp/` viene usata durante la ricezione.

Pregi: resistenza a crash e sistema di lettura/scrittura non bloccante pertanto adatto a filesystem di rete o distribuiti.

Difetti: enome numero di file e quindi di inode impiegati.

Performance: buone - buone in fase di cancellazione dei messaggi che avviene con una semplice operazione di unlink, **scarse** in fase di "listing", dove per conoscere le intestazioni (From e Subject) dei messaggi bisogna aprire e leggere numerosi file.

Alcuni dei difetti prestazionali dei sistemi di storage mailbox e maildir vengono, in parte, ovviati da sistemi di cache implementati da alcuni demoni POP3 e IMAP.
Risulta importante dal punto di vista delle performance sottolineare che il filesystem utilizzato è un fattore rilevante e quindi è opportuno, a meno di altri vincoli, scegliere la soluzione ideale filesystem/mailbox-storage tenendo in considerazione che i due fattori interagiscono.
L'accoppiata di ReiserFS o EXT3 con maildir può fornire una buona soluzione presupponendo che saranno trattati numerosi "piccoli" messaggi, dell'ordine di alcuni MB.
Buono sarà il risultato offerto dai filesystem XFS o JFS nel caso di uso di mailbox in formato mbox, dove le dimensioni di un singolo file, che rappresenta l'intera mailbox, possono diventare ragguardevoli. Una tipica esigenza è quella di convertire le mailbox dal formato mbox a maildir, per passare da un formato all'altro. Per fare ciò sono stati scritti programmi che svolgono il compito in modo automatico. Tra questi alcuni sono scritti in Perl e possono essere trovati sulla rete (vedi *riquadro 8*).

Altri formati mailbox

mbx: è il formato proprietario del server UW-IMAP. E' pensato per ottimizzare le performance e contiene in un unico file tutti i messaggi presenta pertanto una analogia con il formato mbox ed è scarsamente performante nelle operazioni di *expunge*. Le informazioni necessarie al protocollo IMAP (metadata) sono scritte in un record che precede il messaggio.

mailstore: è il formato creato dal progetto Exim, utilizza due file per ciascun messaggio: uno con estensione .env contiene gli header mentre il file con estensione .msg immagazzina contenuto del messaggio. Quando sopraggiunge un messaggio viene creato un file temporaneo con gli header che poi sarà rinominato in .env alla fine della consegna.

dbox: è il formato proprietario del server Dovecot pensato per avere buone prestazioni. I messaggi sono archiviati in uno o più file. Ciascun file può contenere uno o più messaggi.

Format\Software	dovecot	UW-IMAP	Courier-IMAP	Exim	Postfix	Sendmail	PINE	mutt	procmail	maildrop
mbox	Si	Si	No	Si	Si	Si	Si	Si	Si	Si
maildir	Si	No	Si	Si	Si	No	No	Si	Si	Si
mbx	No	Si	No	Si	No	No	Si	No	No	No
mailstore	No	No	No	Si	No	No	No	No	No	No
dbox	Si	No	No	No	No	No	No	No	No	No

prevenire la cattura tramite sniffer delle nostre credenziali di accesso al sistema e dei messaggi di posta che transitano fino a noi. Non sarà vano ricordare che la cattura delle credenziali di un utente consente l'accesso alla sua casella di posta. Una possibile aggravante si verifica se gli utenti POP sono "utenti reali" del server ed essi hanno accesso ai servizi Telnet o SSH, allora la sicurezza è a repentaglio per l'intero sistema e per i suoi utenti.

Per configurare un server IMAP, affinché si metta in attesa di connessioni da instradare su un canale crittografato, i passi da compiere sono piuttosto semplici.

In primo luogo sarà necessario generare un certificato.

Successivamente, agendo sul file di configurazione `dovecot.conf`, dovremo indicare al nostro server dove abbiamo collocato i certificati e le chiavi, indicare che vogliamo attivare il canale protetto e il protocollo IMAPS:

```
# se vogliamo soltanto la modalità protetta possiamo
# disabilitate il protocollo IMAP in chiaro
protocols = imaps
ssl_disable = no
ssl_cert_file = /etc/ssl/certs/ssl-cert-snakeoil.pem
ssl_key_file = /etc/ssl/private/ssl-cert-snakeoil.key
```

Fatto ciò non ci resta che far ripartire il servizio e sfruttare il collegamento al nostro server IMAP attraverso un canale protetto configurando opportunamente il nostro client di posta preferito.

Attualmente moltissimi client di posta supportano tale modalità di comunicazione e nella nostra configurazione dovremo connetterci alla porta 993 su cui è in ascolto il nostro server. Ricordate che se state usando un certificato autofirmato il client segnalerà l'impossibilità di attestare l'autorità certificante, indicando quindi l'impossibilità di garantire la protezione della comunicazione: ricordate che esistono gli attacchi MITM con generazione di certificati "al volo"! Per ovviare a ciò potrete attuare due strade, la prima è realizzabile nel caso di un piccolo numero predefinito di client e prevede di installare su ciascun client il certificato della "nostra" CA (Certification Autority). L'altra soluzione rende disponibile il servizio a chiunque e quindi è adatto ad un uso pubblico e prevede l'uso sul server di un certificato firmato da una CA riconosciuta.

Conclusioni

Nella prossima puntata vedremo un uso evoluto dei server POP e IMAP che si appoggiano ad un database per gestire le credenziali di accesso e dati utili degli account dei nostri utenti che diventeranno "virtuali".

RIQUADRO 7

Configurazione Dovecot V 1.0 IMAP e crittografia

```
protocols = imap
disable_plaintext_auth = no
log_path = /var/log/dovecot.log
info_log_path = /var/log/dovecot.log
first_valid_uid = 1000
default_mail_env = maildir:/var/spool/mail/%u

protocol imap {
   # parametri specifici al protocollo imap.
   # Vedere file esempio dovecot.conf
}

auth default {
   mechanisms = md5
   userdb passwd {
        /etc/passwd
   }
   passdb shadow {
        /etc/shadow
   }
   user = root
}
```

RIQUADRO 8

Webografia

Dovecot
http://dovecot.org
Courier-imap
http://www.courier-mta.org/imap/
Cyrus-imap
http://asg.web.cmu.edu/cyrus/imapd/
UW-imap
http://www.washington.edu/imap/
Formato maildir:
http://www.qmail.org/man/man5/maildir.html
Formato mbox:
http://www.qmail.org/man/man5/mbox.html
Altri formati mailbox:
http://wiki.dovecot.org/MailboxFormat
Conversione da Mailbox a Maildir:
http://www.qmail.org/yammc.pl
Riferimenti tra RFC ed estensioni:
http://www.networksorcery.com/enp/protocol/imap.htm