

# Webmail

## Articolo pubblicato su: [www.sicurezzaite.com](http://www.sicurezzaite.com) Assistenza linux e mail server



**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.



Ci troviamo ancora a parlare di posta elettronica, croce e delizia degli utenti Internet. Nelle precedenti puntate ci siamo curati di come fosse possibile prelevare, attraverso protocolli dedicati, i messaggi email giacenti nella nostra casella postale virtuale.

I protocolli deputati, POP3 e IMAP, sono però stati, almeno parzialmente, spodestati da una fantasiosa interpretazione dell'uso del protocollo http. Alcuni sviluppatori di applicazioni web hanno infatti pensato di far pervenire i messaggi di posta elettronica attraverso tale protocollo: il successo è stato notevole.

### Le webmail

Un modo diverso di dare accesso alla posta elettronica giacente sul mailserver e quello di consentire la lettura attraverso un browser: il protocollo HTTP viene utilizzato sia per passare i contenuti di una casella di posta elettronica in formato html che per la creazione di messaggi on-line, aggiungendo allegati e spedendo messaggi verso l'esterno. Il tutto è possibile, in genere, dopo aver fornito le proprie credenziali sotto forma di username e password.

Descrivere come si usi una webmail forse non è necessario visto che con ogni probabilità molti di voi, se non tutti, hanno avuto accesso a uno di tali servizi.

Sono molti i provider a fornire servizi del genere: anzi, per la verità alcuni provider recentemente hanno limitato l'accesso alla posta elettronica fornita gratuitamente al solo accesso web, optando un compromesso tra introiti da pubblicità sul web, di solito sempre o quasi presente, e l'utilizzo di risorse a carico del gestore che offre il servizio.

### Vantaggi e svantaggi

I vantaggi che l'utente trova nell'uso di una webmail per gestire la propria posta elettronica piuttosto che un programma client specifico sono, per alcune tipologie di utenti, molto rilevanti. Attraverso la fruizione dei browser non occorre installare alcun software aggiuntivo sul proprio PC, molto spesso non serve neppure configurare alcunché, la sola username (spesso l'indirizzo di posta elettronica) e la password sono sufficienti a permettere di accedere ai propri messaggi.

Tale servizio ha il grande vantaggio di essere disponibile praticamente ovunque si trovi dispositivo connesso ad internet, dal più classico PC ai cellulari di nuova generazione (display permettendo!): in pratica, ovunque noi siamo si riesce a disporre dell'intero nostro archivio mail. La propria natura "on-line" è anche il principale svantaggio: senza connessione, l'accesso alla posta, anche offline, ci è completamente precluso. Inoltre, la composizione di una mail, dovendo necessariamente passare tramite il server web, può risentire di una certa macchiosità. Talvolta è penalizzante la mancanza di alcune funzionalità che sui normali client sono presenti di default mentre latitano sulle webmail più diffuse, anche se ultimamente, complice Google con il servizio Gmail, le cose stanno cambiando assai velocemente.

### Architettura

Per la webmail non è mai stato codificato un RFC che descriva come tale servizio debba funzionare, e quindi c'è stato spazio per gli sviluppatori per scatenare la loro fantasia. I differenti



progetti hanno architetture ed implementazioni molto diverse gli uni dagli altri.

Tanto per cominciare l'accesso ai messaggi può avvenire in almeno due modalità. Una soluzione è quella che svolge un accesso diretto attraverso i programmi (in genere CGI) ai file contenuti nella cartella di archiviazione dei messaggi: ad esempio la tradizionale `/var/spool/mail` per un sistema in stile *sendmail*. Questo approccio, piuttosto rudimentale, è ormai quasi del tutto in disuso per varie ragioni, tra queste la scarsa sicurezza, scalabilità e flessibilità.

Un diverso approccio è quello basato sul passaggio attraverso un servizio POP3 o IMAP. L'applicativo web interpella un servizio POP3 o IMAP locale o remoto e con esso interagisce prelevando, manipolando e cancellando i messaggi. Questa seconda modalità è più vicina alla filosofia \*NIX dove le diverse componenti vengono sviluppate indipendentemente secondo specifiche di pubblico dominio e successivamente interfacciate tra di loro, utilizzando uno schema a *strati applicativi*: così procedendo si ottiene una intercambiabilità di componenti, una maggiore flessibilità e una notevole scalabilità. Nella fase di tuning e debugging è possibile, inoltre, isolare i diversi componenti ed individuare singolarmente quello su cui operare.

In questa direzione si sono orientati la maggior parte dei progetti di webmail sviluppati negli ultimi anni.

## Groupware, evoluzioni integrate, gigawebmail & C.

Negli ultimi anni si stanno diffondendo alcuni progetti software, su cui non ci soffermeremo, che rappresentano l'ampiamiento del concetto di webmail: si tratta di software conosciuti come sistemi groupware o software collaborativi, programmi che operano per lo più lato server e che consentono, oltre alla gestione della posta elettronica, di manipolare informazioni private e condivise come calendari di appuntamenti, rubriche e archivi di dati.

Questi progetti sono pensati in genere per ambienti aziendali e mirano a rendere più efficiente lo scambio di informazioni tra individui che cooperano: alcuni esempi di buoni progetti di groupware opensource sono rappresentati da Phpgroupware, E-groupware e Opengroupware (vedere webografia).

Per quanto riguarda le webmail di "grosso calibro", di recente alcuni grandi provider, ad esempio Google Inc. con la famosa Gmail, hanno sviluppato ottimi sistemi di posta elettronica gestibile tramite web.

Spinte da tale consistente novità, altri provider che forniscono posta elettronica web-based hanno dovuto innovare e rilanciare l'offerta, come ad esempio Lycos Inc. che offre 3 GB ed ha

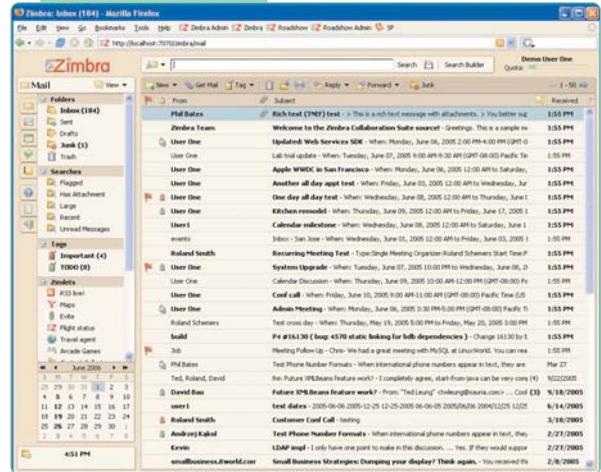
rinnovato integralmente la sua interfaccia web.

Tra le tecnologie efficaci ed in crescita si vedono quelle che fanno ricorso ad un massiccio uso delle tecnologie AJAX demandando così parte dei carichi computazionali ai client e permettendo di velocizzare le operazioni per l'utente.

Soluzioni integrate solo per grosse società? Non proprio, basti considerare Zimbra: si tratta di un software di collaborazione rilasciato sotto licenza Mozilla Public License, chiamato per la precisione "Zimbra Collaboration Suite" e sta conoscendo un'ampia diffusione. È basato sulla tecnologia AJAX e disponibile in diverse versioni, una opensource ("opensource edition"), che non offre funzionalità di amministrazione grafica e per la quale non viene fornito supporto, e due a pagamento con diverse livelli di servizio da parte della società produttrice (comprensiva di servizi di backup/restore, clustering/HA, gestione storage management, personalizzazione, gestione pubblicità, etc.).

Zimbra dispone di un grande numero di funzionalità comprese ovviamente quelle più strettamente tipiche della webmail, rubrica, ricerca, gestione allegati, amministrazione, gestione

FIGURA 1



Zimbra al lavoro: l'impatto grafico è impressionante, non trovate?

FIGURA 2



SqWebMail non ha certo lo stesso impatto grafico, ma in fatto a velocità è impareggiabile!!!

## RIQUADRO 1

**Acceleratori per PHP****APC Alternative PHP Cache**

<http://pecl.php.net/package/APC>

Svolge la funzionalità di cache, è free ed opensource e supporta le serie 4.3.x e 5.x del PHP.

**EAccelerator**

<http://eaccelerator.sourceforge.net/>

Derivato dal vecchio progetto Turck MMCache, ha aggiunto stabilità ed il supporto alla versione 5.x del PHP. Molto efficiente può portare a notevoli diminuzioni sui tempi di esecuzione degli script. Il codice è rilasciato sotto licenza GPL.

**PHP Accelerator**

<http://www.php-accelerator.co.uk/>

Si tratta di un progetto a sorgenti chiusi e di cui sono disponibili solo i binari, il codice inoltre da un certo tempo non viene aggiornato. Attualmente l'acceleratore funziona senza problemi con versioni recenti del PHP.

**Zend Optimizer**

<http://www.zend.com/store/products/zend-optimizer.php>

Ancora un'ottimizzatore che permette di diminuire i tempi di esecuzione degli script PHP, il software è free ma closed-source in quanto proprietà della Zend Technologies Ltd. È attivamente supportato e utilizzabile per tutte le versioni del PHP successive alla 4.2.x.

multi-dominio. Non è propriamente una webmail, ma un sistema completo di mailserver contenente tutti i componenti per svolgere le funzionalità che offre: server SMTP (Postfix), WEB (Apache Tomcat), antivirus (ClamAV), antispam (Spamassassin), LDAP (OpenLDAP), spell checker (Aspell), .

Il software è reso disponibile già pacchettizzato per numerose distribuzioni GNU/Linux e per MacOSX insieme ad un'ampia e curata documentazione.

I requisiti di sistema, trattandosi di un server con numerose funzionalità, non sono proprio quelle di un vecchio PC da riutilizzare: occorre infatti una buona dose di RAM (minimo 2 GB) e una buona potenza computazionale.

## Progetti webmail opensource

Facciamo una rapida rassegna di quello che la comunità di sviluppatori ha creato e rilasciato come codice aperto e con licenze opensource a riguardo dei progetti "webmail": alcuni di essi sono molto evoluti e ricchi di feature, una parte di queste funzionalità sono ritenute indispensabili dagli utenti mentre altre

sono da considerare poco più che gadget software.

I linguaggi di programmazione con cui le webmail sono realizzate sono i più disparati, dal classico C, che fornisce ottime performance, al Perl ed il PHP. Negli ultimi tempi stanno prendendo campo sistemi sviluppati anche con linguaggi forse meno diffusi ma comunque molto validi come Python e Ruby. Inoltre una parte delle funzionalità sono spesso fornite lato client da procedure scritte in linguaggio Javascript.

Nella scelta di un sistema di webmail sarà opportuno tenere conto di quale sarà l'architettura in cui vorremo impiegarla e le esigenze attuali e future del sistema, in particolare della scalabilità e affidabilità richieste dal servizio.

Se il numero di utenti cresce di un fattore 10 in breve tempo e non avremo tenuto in conto tale ipotesi preventivamente, le cose potrebbero complicarsi molto, senza considerare che i sistemi di mailserver sono alcuni dei più critici da arrestare per apportare modifiche e/o aggiornamenti o, nel caso peggiore, sostituzioni di componenti.

Come già detto nelle precedenti puntate, vale anche in questo caso, la raccomandazione di tentare già in fase di progettazione di stimare il livello di utilizzo di un sistema e di valutare perciò le risorse necessarie occorrenti per soddisfare tali richieste. Per questo può essere utile, se il nostro sistema è, ad esempio un sistema LAMP (Linux, Apache, MySQL, PHP) studiare le esigenze di ciascun componente e vedere quali risorse esso attinge dal sistema, ad esempio memoria, connessioni concorrenti, performance del back-end del database etc.

Per un dettaglio relativo ad alcuni dei più diffusi progetti opensource di webmail consultare la tabella 2

## Performance, scalabilità?

Vogliamo confrontare, almeno a livello concettuale, le diverse impostazioni date da due diffusi progetti di webmail che adottano un approccio e seguono obiettivi diversi: Squirrelmail e SqWebMail. Questi vogliono essere rappresentativi di due diverse modalità operative delle webmail che presentano ciascuna vantaggi e svantaggi.

SqWebMail è un progetto che mira ad essere veloce e snello nell'ingombro di risorse e pertanto adatto anche a sistemi non proprio super-performanti. Per ottenere i risultati SqWebMail si affida al linguaggio C e accede in modo diretto alle directory che contengono i file, solo nel formato maildir.

L'architettura è semplice, l'eseguibile `sqwebmaild` viene lanciato come demone e sotto il controllo dell'utente root (in alcuni pacchetti viene preferito usare il `suid` bit dell'eseguibile), di tale eseguibile si avviano, sotto forma di processi, un certo numero di istanze. L'eseguibile `sqwebmail` che viene usato come script



CGI dal server web non fa altro che raccogliere le informazioni dai moduli web e passarle, attraverso una UNIX domain socket, ad una delle istanze di `sqwebmaild` attive.

A questo punto, i dati verranno prima validati e poi elaborati. Sono state messe in atto nel software SqWebMail alcuni accorgimenti per tentare di minimizzare i problemi di sicurezza, in particolare le autenticazioni sono svolte da un modulo che fa uso, per default, delle librerie PAM se disponibili, o dei classici file `/etc/passwd` ed `/etc/shadow`.

Gli accessi vengono autorizzati per un tempo limitato in modo da minimizzare i potenziali rischi dovuti al caching dei browser o ad eventuali sottrazioni di credenziali attraverso sniffing.

La webmail SqWebMail dispone anche di un semplice programma, `sqwebpasswd`, che gira `guid` e che consente il cambio delle password.

D'altra parte i vantaggi di poter accedere direttamente ai file sono sostanziali e le performance di un programma compilato sono difficilmente pareggiabili.

Per la sua caratteristica di accesso diretto ai file è ovvio che la webmail deve risiedere sulla stessa macchina che contiene tali file rendendo difficile, se non impossibile, la possibilità di suddividere i servizi e di scalare in caso di aumenti di carichi.

Il linguaggio con cui è realizzata SqWebMail non agevola l'operazione di modifica del codice e, inoltre, non si può certo dire che questa webmail si distingua per la cura grafica dell'interfaccia utente.

Squirrelmail intende offrire le funzionalità classiche dei client mail da desktop e desidera essere modulare con una struttura a plugin. Dal canto suo si affida ad un linguaggio di scripting web molto diffuso e nato per lo sviluppo rapido di applicazioni. Si tratta però di un linguaggio interpretato e questo penalizza le prestazioni.

A peggiorare la situazione contribuisce il fatto che squirrelmail deve connettersi ad un server IMAP per l'operazione di autenticazione e per svolgere tutte le interazioni con la mailbox. A suo vantaggio conta la possibilità di essere facilmente esteso attraverso contributi scritti da diversi autori che hanno realizzato i loro plugin. Il linguaggio usato consente a tutti rapide modifiche e personalizzazioni sia grafiche sia dei programmi. Tra i vantaggi c'è una grandissima facilità di installazione. Nel codice di squirrelmail sono state sviluppate librerie per l'accesso ai server IMAP per non dover ricorrere alle funzionalità IMAP del PHP che, molto spesso, non sono presenti nei pacchetti precompilati di tale linguaggio.

## Ottimizzazioni delle webmail

Le webmail che si interfacciano ad un server IMAP hanno,

### RIQUADRO 2



## Configurazione SquirrelMail

Nell'ordine fornito dai menu di configurazione vengono chieste informazioni su:

**Organization Preferences:** impostazioni relative al sito (intestazioni, immagine logo e sue dimensioni, titolo delle pagine e altre informazioni relative al gestore del sito)

**Server Settings:** informazioni sul server IMAP e su quello SMTP remoto o locale, individuato con un hostname o con un IP, e su quale porta esso debba essere contattato.

**Folder Defaults:** impostazione delle cartelle IMAP, la cartella "cestino", quella della posta inviata, quella delle Bozze, alcuni comportamenti delle operazioni relativi alla cancellazione, al salvataggio e all'archiviazione della posta inviata.

**General Options:** impostazioni relative alle cartelle di gestione dei file di squirrelmail come gli allegati e i file dell'utente (rubrica, calendario, etc.), opzioni relative alla gestione delle identità e dei dati da parte dell'utente, opzioni sulle sessioni PHP e sulle modalità di ordinamento.

**Thems:** con questo menù si può impostare il tema di default tra quelli disponibili o attraverso un tema personalizzato costruito con un foglio di stile CSS.

**Address Book:** le opzioni relative alla rubrica, interessante il supporto ad LDAP.

**Plugins:** da qui si possono gestire i plugin installati attivandoli e disattivandoli, alcuni sono forniti nel pacchetto base.

**Database:** sebbene non necessario al funzionamento, Squirrelmail supporta l'uso di DataBase per la gestione dei dati relativi agli utenti. Sono supportati MySQL, PostgreSQL, msql. Si possono gestire separatamente rubriche globali ed individuali. Si potranno definire le tabelle contenenti i dati delle preferenze degli utenti e delle rubriche.

**Languages:** permette di scegliere la lingua e il set di caratteri predefinito.

come già detto, il grande pregio di poter essere scalabili e permettere facilmente la suddivisione dei servizi su macchine diverse. L'aspetto negativo relativo alle scarse performance causati da uno "strato" applicativo aggiuntivo, potrebbe scoraggiare qualcuno dallo scegliere questa soluzione. In realtà un certo numero di accorgimenti possono migliorare in modo drastico l'usabilità di questa soluzione.

Tra questi accorgimenti possiamo menzionare il *proxing* del servizio IMAP: il problema manifestato da alcune webmail è quello di non permettere connessioni persistenti con il server IMAP, facendo operare una operazione di login per quasi ogni operazione compiuta. In questi casi il caching delle connessioni da parte di un servizio di proxy potrà dare forti benefici poi-

ché la stessa connessione IMAP viene riutilizzata numerose volte.

Per tale specifico scopo è nato il progetto `imapproxy` (vedere webografia): chiaramente le webmail che utilizzano connessioni persistenti al server IMAP non beneficeranno da questa soluzione e sono pertanto già intrinsecamente "migliori", almeno dal punto di vista delle performance.

Se la nostra webmail è, ad esempio, sviluppata in PHP, possiamo installare e mettere in funzione degli acceleratori dell'interprete, essendo ormai diversi i progetti che mirano ad aumentare le prestazioni di questo famoso linguaggio (tanto per citare alcuni esempi: APC, EAccelerator, PHP Accelerator, Zend Optimizer).

Una accortezza che può migliorare le performance è quella di far girare il proprio server IMAP standalone e non attraverso il superdemone `inetd` o `xinetd`. Sarà poi opportuno notare che alcuni server IMAP precompilati sono configurati di default con

### RIQUADRO 3



## Configurazione Apache per Squirrelmail

```
Alias /webmail /var/www/htdocs/webmail/squirrelmail

<Directory /var/www/htdocs/webmail/squirrelmail>
  Options Indexes FollowSymLinks
  <IfModule mod_php4.c>
    php_flag register_globals off
  </IfModule>
  <IfModule mod_php5.c>
    php_flag register_globals off
  </IfModule>
  <IfModule mod_dir.c>
    DirectoryIndex index.php
  </IfModule>

  # accesso al file configtest solo al PC di amministrazione
  <Files configtest.php>

    order deny,allow
    deny from all
    allow from 192.168.0.100
  </Files>
</Directory>

# Se vi desidera dedicare una url semplice tipo
# http://webmail.oltrelinux.com
# dovrà' essere mappato da un server DNS
# <VirtualHost 1.2.3.4:80>
#   DocumentRoot /var/www/htdocs/webmail/squirrelmail
#   ServerName webmail.oltrelinux.com
#   se disponiamo di canale ssl possiamo reindirizzare le
#   richieste per per questo server verso il canale criptato
#   decommentando la linea sottostante
#   Redirect / https://webmail.oltrelinux.com
# </VirtualHost>
```

un esiguo numero di connessioni che possono ricevere contemporaneamente, se tale numero è troppo basso il servizio di webmail potrebbe operare in modo scarsamente efficace.

Alcune webmail, tra queste squirrelmail, utilizzano funzioni per l'elencazione dei messaggi: se il server IMAP supporta nativamente l'estensione SORT può essere conveniente fare in modo che questa venga sfruttata dalla webmail con incrementi che possono arrivare, in alcuni casi, da 30 secondi per elencare oltre 10.000 messaggi a 2-3 secondi.

Anche se può sembrare ovvio le installazioni che talvolta si trovano su alcuni server dimostrano che ovvio non è: le performance di una webmail che si appoggia ad un server IMAP dipendono in larga parte dalle performance del server IMAP stesso, e passare da un server UW-IMAP a Dovecot può far incrementare la velocità anche di un fattore 10. In alcuni casi, la sostituzione di un server Courier-Imap con Dovecot, in presenza di mailbox molto "affollate", ha portato il tempo necessario per il listing dei messaggi da oltre un minuto a circa un secondo.

## Sicurezza

Le webmail che si appoggiano ad IMAP offrono la possibilità di suddividere i servizi di accesso alle mailbox e quelli web/PHP permettendo facilmente di distribuire i carichi di lavoro. Tale possibilità ovviamente può essere resa possibile anche tra macchine residenti su reti diverse. In questo caso è consigliabile attivare la comunicazione cifrata attraverso il servizio IMAPS tra i server.

Altri accorgimenti possono servire ad impedire lo sfruttamento di bug nella webmail e migliorare la sicurezza del sistema, tra questi il modulo `apache mod_security` e l'uso delle restrizioni `safe_mode`, per quelle basate su PHP.

## Una webmail tutta nostra

Come già anticipato nella scelta della webmail che dovremo installare sarà necessario considerare le esigenze degli utenti, le funzionalità che si vorranno offrire e le risorse a disposizione, siano esse computazionali sia di amministrazione e gestione (da non sottovalutare queste ultime). Sarà poi conveniente in ogni caso affidarsi a progetti solidi e attivamente supportati così da poter contare su aggiornamenti rapidi in caso si dovessero scoprire dei problemi di sicurezza.

Per gli ambienti di produzione, sarà forse utile ricordare, è conveniente mettere in test i nuovi pacchetti webmail su macchine di prova per verificare che l'integrazione con il restante ambiente non dia problemi.



Per le nostre prove abbiamo deciso di scegliere una webmail di facile installazione e gestione che potesse accedere alla posta attraverso i protocolli IMAP e POP3 per permettere di integrarsi con sistemi SMTP/IMAP/POP3 già esistenti e una buona scalabilità in caso di crescite future.

Una delle più diffuse webmail che presenta le caratteristiche che cerchiamo è sicuramente Squirrelmail: a dimostrazione che tale progetto sia ad un buon stato di sviluppo ed affidabilità alcune ben note distribuzioni hanno incluso una versione pacchettizzata pronto all'uso per i propri utenti.

## Squirrelmail in pochi passi

Tra i pregi di Squirrelmail figura certamente la presenza nel pacchetto di uno script di installazione e amministrazione che semplifica enormemente la vita dell'amministratore di sistema. Una volta scaricato il pacchetto dal progetto, e dopo aver predisposto il nostro ambiente: una linux/BSD box con installato un webserver che supporti il PHP (ovviamente Apache è un'ottima scelta), il PHP deve essere almeno nella sua versione 4.1.2. Serve inoltre un server IMAP 4 rev.1, per esempio Dovecot, già visto nei numeri precedenti.

Perl è opzionale, utilizzato per eseguire lo script di installazione ed amministrazione: se non è presente la webmail funzionerà ugualmente ma occorrerà editare manualmente i file di configurazione.

Passiamo ora all'installazione: supponiamo che la cartella contenente i documenti gestiti dal webserver, la "document root" di apache, sia `/var/www/htdocs/`, creiamo una cartella dedicata alla webmail. Spostiamoci in questa directory e scompattiamo il pacchetto appena scaricato:

```
# mkdir /var/www/htdocs/webmail
# cd /var/www/htdocs/webmail
# tar xzvf /home/admin/squirrelmail-1.4.9a.tar.gz
```

rinominiamo la cartella

```
# mv squirrelmail-1.4.9a/ squirrelmail
```

creiamo la cartella degli allegati e quella di archiviazione degli allegati e quella dei dati utente

```
# mkdir /var/spool/squirrelmail/data/
# mkdir /var/spool/squirrelmail/attachments/
```

possiamo ora passare alla configurazione della webmail: spostiamoci nella directory `config` e lanciamo lo script perl che ci

faciliterà nella configurazione dei parametri del nostro server.

```
$ cd squirrelmail/config && ./config.pl
```

Fondamentale che nelle "general options" siano impostate le due cartelle appena create come spool per i messaggi e per gli allegati. Dovremo accertarci che le cartelle di spooling abbiano permessi e proprietari opportuni, per esempio con:

```
# chown -R www.nobody /var/spool/squirrelmail/data/
# chmod -R 750 /var/spool/squirrelmail/data/
```

idem per la cartella `/var/spool/squirrelmail/attachments/`. Dovremo a questo punto configurare il server apache affinché ci permetta l'accesso alla webmail impostando un alias o un server virtuale dedicato allo scopo e, in questo secondo caso, dovremo anche agire a livello di DNS per fare in modo che le richieste vengano correttamente indirizzate (vedi riquadro 3). Riavviato apache non ci resta che verificare la correttezza della configurazione utilizzando lo script PHP che nella nostra configurazione troviamo all'url:

<http://192.168.0.10/webmail/src/configtest.php>

che ci da utili informazioni di debug fino al raggiungimento della piena operatività.

## Nuove funzionalità con i plugins

Oltre alle funzionalità di base offerte da squirrelmail è possibile attivare alcuni plugins resi disponibili già nel pacchetto di base e personalizzare per le proprie esigenze le funzioni messe a disposizione della webmail.

### RIQUADRO 4



#### Creazione tabella di auth per MySQL / SqWebMail

```
CREATE TABLE passwd (
  id char(128) DEFAULT '' NOT NULL,
  crypt char(128) DEFAULT '' NOT NULL,
  clear char(128) DEFAULT '' NOT NULL,
  name char(128) DEFAULT '' NOT NULL,
  uid int(10) unsigned DEFAULT '65534' NOT NULL,
  gid int(10) unsigned DEFAULT '65534' NOT NULL,
  home char(255) DEFAULT '' NOT NULL,
  maildir char(255) DEFAULT '' NOT NULL,
  defaultdelivery char(255) DEFAULT '' NOT NULL,
  quota char(255) DEFAULT '' NOT NULL,
  options char(255) DEFAULT '' NOT NULL,
  KEY id (id(128))
);
```

## RIQUADRO 5



## File di configurazione /etc/courier/authmysqlrc

```
# Il nome del server, la username e la password di accesso
# (PARAMETRI OBBLIGATORI)
MYSQL_SERVER      localhost
MYSQL_USERNAME    mailuser
MYSQL_PASSWORD    ilmiosegreto

# Il nome del database MySql da usare          (PARAMETRO OBBLIGATORIO)
MYSQL_DATABASE    mailserver

# The name of the table containing your user data. See README.authmysqlrc
# for the required fields in this table.
MYSQL_USER_TABLE  passwd          (PARAMETRO OBBLIGATORIO)

# Si deve indicare almeno uno dei due metodi "in chiaro" o "criptato"
# Da notare che la password criptata con la funzione crypt del sistema
# e non con la quella di MySql !
MYSQL_CRYPT_PWFIELD  crypt
# MYSQL_CLEAR_PWFIELD  clear

# Se viene usato DEFAULT_DOMAIN quando un utente entra utilizzando
# la username 'utente', sarà tradotto in 'utente@DEFAULT_DOMAIN'
# DEFAULT_DOMAIN      example.com

# contiene il numero dello userid dell'account
# MYSQL_UID_FIELD
```

```
MYSQL_UID_FIELD    uid

# contiene il numero dello groupid dell'account
MYSQL_GID_FIELD    gid

# E' l' id della login in sostanza risulta una query tipo:
# SELECT MYSQL_UID_FIELD, MYSQL_GID_FIELD, ... WHERE id='loginid'
MYSQL_LOGIN_FIELD  id

# Permette di specificare una collocazione particolare della cartella
# maildir di default e' in $HOME/Maildir imposta il valore $HOME
# MYSQL_HOME_FIELD  home

# Permette di specificare una collocazione particolare della cartella
# maildir di default e' in $HOME/Maildir
# ($HOME e' impostato dalla variabile MYSQL_HOME_FIELD)
# MYSQL_MAILDIR_FIELD  maildir

# Parametro che definisce il campo da cui prendere il valore di quota
# MYSQL_QUOTA_FIELD    quota

# E' possibile aggiungere ulteriori condizioni da verificare
# attraverso una opportuna condizione WHERE ad esempio
# MYSQL_WHERE_CLAUSE  active='y'
```

Se questo non bastasse, la modularità del progetto, l'ampia base di installato e la diffusione del linguaggio con cui è scritto a fatto in modo che si sviluppasse una buona disponibilità di plugins non realizzati dagli sviluppatori del progetto squirrel-mail ma disponibili alla comunità.

C'è l'imbarazzo della scelta e si può scorrere la gamma dal sito del progetto. L'installazione rasenta la banalità, cosa assai gradita però in un ambiente che ci ha abituato a contorcimenti notevoli per avere risultati.

È sufficiente scompattare il pacchetto relativo al plugin di interesse nella cartella `plugins/` della nostra webmail, l'importante è non fare confusione con le versioni. Una volta fatto ciò non resta che rilanciare lo script di configurazione e con i menù andare a modificare l'elenco dei plugins attivi dove dovremmo trovare, disattivi, quelli appena installati.

## Installiamo Sqwebmail

Abbiamo discusso delle funzionalità di SqWebMail e qualcuno avrà trovato tale progetto interessante per le sue caratteristiche, magari perché desidera apportare delle modifiche su dei programmi CGI scritti in C.

È utile sapere che l'installazione di SqWebMail offre numerose opzioni tra cui scegliere in fase di compilazione, minore è la libertà di configurazione offerta da tale software in run-time. Tra i requisiti, oltre ai necessari compilatori C++, GNU make, le PCRE e un webserver che supporti le CGI, occorre aver installato anche la libreria di autenticazione di Courier

<http://www.courier-mta.org/authlib/>

La sequenza per installare SqWebMail è, secondo i suggerimenti degli autori del software, la seguente:

```
# ./configure [opzioni]
# make configure-check
# make
# make check
# make install-strip
# o make install se il precedente non funziona
# make install-configure
# installa alcuni file di configurazione
```

Alcune delle opzioni che possono essere ritenute più interessanti da usare con lo script `configure` sono:

`--with-cachedir=dir, --with-cacheowner=userid:`

SqWebMail usa una cache per tenere traccia delle login attive. Questo migliora sensibilmente le prestazioni del sistema. Di default viene usata la directory `/var/cache/sqwebmail` o `/var/run/sqwebmail` con questa opzione si può modificare tale impostazione.

`--with-maxmsgsize=n:` la dimensione massima del messaggio comprensiva degli allegati, il default è 2 MB, forse un po' poco per i tempi attuali.

A questo punto vi troverete con un script che permette di



avviare la webmail, ricordarvi di modificare la vostra script di boot in modo che il servizio parta automaticamente

```
/usr/lib/sqwebmail/libexec/sqwebmaild.rc start
```

## Configuriamo SqWebMail per l'uso di MySql

Per permettere l'accesso agli utenti, SqWebMail si appoggia al demone di autenticazione di courier, che fa uso delle librerie courier-authlib. Il demone di autenticazione utilizza dei moduli che si interfacciano con varie fonti, tra queste quella solitamente usata di default è PAM

Dopo aver installato appunto le librerie courier-authlib ci troveremo con un servizio di autenticazione altamente configurabile. Il file di configurazione principale è:

```
/etc/courier/authdaemonrc
```

Possiamo editare tale file ed aggiungere i moduli di autenticazione che preferiamo, la scelta è ampia: authuserdb, authpam, authpgsql, authldap, authmysql, authcustom, authpipe. Nel nostro caso vogliamo appoggiarci al DB MySql e modifi-

cheremo come segue, dando come prima scelta il database e come seconda PAM:

```
authmodulelist="authmysql authpam"
```

Ovviamente dovremo avere un server MySql in funzione con un database e una tabella dedicati allo scopo. Il modulo authmysql si aspetta una struttura delle tabelle che può essere costruita attraverso query riportata nel riquadro XXX.

Ora non ci resta che configurare il modulo appena aggiunto modificando il file /etc/courier/authmysqlrc seguendo le indicazioni del riquadro XXX. Come già visto in altre puntate si prelevano alcuni dati essenziali dai campi della tabella attraverso una query.

I campi da usare possono essere scelti da noi, anche la query finale potrà avere una forma molto varia in funzione di come andremo ad agire su alcuni parametri opzionali del file di configurazione. Non ci resta che un ritocco, se già non fosse sistemato, al file di configurazione di apache con alcuni parametri (esempio per apache2):

```
ScriptAlias /cgi-bin/ /usr/lib/cgi-bin/
<Directory "/usr/lib/cgi-bin">
    AllowOverride None
    Options ExecCGI -MultiViews
                                +SymLinksIfOwnerMatch
    Order allow,deny
    Allow from all
</Directory>
```

Fatto ciò non ci resta che avviare il demone di autenticazione, poi la webmail e testarla aprendo il nostro browser all'indirizzo [http://indirizzo\\_server\\_ip/cgi-bin/sqwebmail](http://indirizzo_server_ip/cgi-bin/sqwebmail)

## Conclusioni

Se non è possibile sviluppare in tempi brevi progetti analoghi a quelli dei grandi provider, il panorama di progetti opensource offre una vasta gamma di software per dotarsi di una efficace webmail. In questo modo la nostra posta sarà resa disponibile ovunque. Si consiglia però di riporre attenzione anche agli aspetti relativi alla sicurezza magari facendo transitare obbligatoriamente i dati attraverso un canale cifrato utilizzando il protocollo https del vostro webserver di fiducia.

Con questa modalità di accesso alla posta abbiamo chiuso la carrellata sul tema "accedere alla posta elettronica": l'argomento però non si esaurisce e presto altri articoli sull'argomento faranno capolino in queste pagine!



### RIQUADRO 6



### Webografia

- Phpgroupware, software groupware scritto in PHP  
<http://www.phpgroupware.org/>
- Opengroupware, software groupware licenza GPL/LGPL  
<http://www.opengroupware.org/>
- E-groupware, software groupware licenza GPL/LGPL  
<http://www.egroupware.org>
- Zimbra, Software collaborativo  
<http://www.zimbra.com/>
- Proxy IMAP  
<http://www.imaproxy.org/>
- SqWebMail (Courier MTA) - Linguaggi: C, C++, Perl  
<http://www.courier-mta.org/>
- SquirrelMail - Linguaggio: PHP  
<http://www.squirrelmail.org/>
- IMP - Linguaggio: PHP  
<http://www.horde.org/imp/>
- Open WebMail - Linguaggio: Perl  
<http://openwebmail.org/>
- IlohaMail - Linguaggio: PHP  
<http://ilohamail.org>