

01/08/2018



# Web Server - CentOS

*Guida installazione Web Server - LEMP / FTP / Mail*



Giulio Bosco - SAMT I2AA

# Web Server - CentOS

## *Guida installazione Web Server - LEMP / FTP / Mail*

Guida per installare un server basato su sistema operativo Linux, distribuzione CentOS (Versione meno stabile di RedHat).

## Sommario

|   |    |
|---|----|
| Requisiti del web server .....                          | 3  |
| Installazione Del WebServer .....                       | 4  |
| Installazione SO (Linux - CentOS) .....                 | 4  |
| Introduzione .....                                      | 4  |
| Macchina .....  | 4  |
| Installazione .....                                     | 4  |
| Configurazione base .....                               | 5  |
| Aggiornare il sistema operativo .....                   | 7  |
| Installazione NGINX (servizio http) .....               | 8  |
| Introduzione .....                                      | 8  |
| Installazione .....                                     | 8  |
| Configurazione Virtual Host .....                       | 9  |
| Installazione DB (MySQL - MariaDB) .....                | 11 |
| Introduzione .....                                      | 11 |
| Installazione .....                                     | 11 |
| Installazione Linguaggio Script Lato Server (PHP) ..... | 12 |
| Introduzione .....                                      | 12 |
| Installazione .....                                     | 12 |
| Configurazione .....                                    | 12 |
| Configurazione PHP su NGINX .....                       | 13 |
| Test funzionamento PHP .....                            | 13 |
| Installare phpMyAdmin .....                             | 14 |
| Introduzione .....                                      | 14 |
| Installazione .....                                     | 14 |

|                                    |    |
|------------------------------------|----|
| Installazione Server FTP .....     | 17 |
| Introduzione .....                 | 17 |
| Installazione .....                | 17 |
| Creazione utenti per FTP .....     | 18 |
| Installazione Server Mail .....    | 19 |
| Introduzione .....                 | 19 |
| Installazione Postfix .....        | 19 |
| Installazione Dovecot .....        | 21 |
| Installazione WebMail Client ..... | 24 |
| Introduzione .....                 | 24 |
| Installazione .....                | 24 |

## Requisiti del web server

---

- Sistema operativo: Linux (Distribuzione a scelta)
- Servizio http: Nginx
  - o Contenente 3 virtual-host (nome.com, sql.nome.com, cliente.com)
- Linguaggio script lato server: PHP
  - o Creare una pagina con le info su php per ogni virtual-host
- Server DB: MySQL
  - o 2 utenti di amministrazione
  - o 2 utenti per il cliente con un database
- Server FTP
  - o 2 utenti di amministrazione
  - o 2 utenti per il cliente
- Server Mail
  - o 2 utenti di amministrazione
  - o 2 utenti per il cliente
- Client web DB: phpMyAdmin
- Client web FTP
- Client web Mail
- Utenti per il cliente
  - o 1 amministratore del dominio
  - o 1 normale

## Installazione Del WebServer

---

### Installazione SO (Linux - CentOS)

#### Introduzione

Linux CentOS è un sistema operativo basato su Linux / UNIX sviluppato da RedHat, è molto usato per i server, dal momento che è un sistema operativo molto leggero.

È stato scelto CentOS come distribuzione di Linux, perché è stabile, ci sono moltissimi software compatibili e molte guide in caso si incontrassero problemi durante l'installazione del server.

Quando ci saranno indicazioni su comandi da eseguire prima del comando ci sarà un cancelletto che indicherà che il comando va eseguito come amministratore, mentre se un comando sarà preceduto da un dollaro potrà essere eseguito anche come utente normale.

```
# <comandoAmministratore>
$ <comandoStandard>
```

#### Macchina

Come macchina per il server è consigliato usare una virtual machine, per esempio si può usare Virtual Box oppure VM Ware (il secondo è un software a pagamento).

Si consiglia di usare una virtual machine per più motivi:

- Una virtual machine può essere clonata.
- Si può “tornare in dietro nel tempo” tramite gli snapshot (spiegato più avanti).
- Si possono creare e trasferire su qualunque macchina fisica, anche a caldo (che vuol dire si possono trasferire su un altro computer anche se sono accese).

Se si utilizza una virtual machine si consiglia creare una i seguenti requisiti:

- RAM: 2GB
- Processori: 4
- Disco: 25GB
- Scheda di rete: bridget

Nel caso in cui si usa un server fisico si consigliano gli stessi requisiti, con l'unica differenza che la scheda di rete è fisica e non virtuale, quindi non c'è bisogno che sia bridget, cioè usare la stessa scheda del computer che ospita la vm (termine usato al posto di virtual machine) assegnandole due indirizzi IP; uno verrà usato per l'hoster mentre il secondo per la VM.

#### Installazione

Per installare CentOS si usa la il file ISO (scaricabile dal sito ufficiale: [www.centos.org](http://www.centos.org)).

Inserire il file ISO nella VM come cd.

Per installare CentOS c'è una procedura guidata. Per un server, non è consigliato installare l'interfaccia grafica, perché è molto pesante e ne riduce moltissimo le prestazioni. In ambiente di test si consiglia di impostare delle password facili, mentre in ambiente di produzione si consigliano password molto complesse ed avere due o più utenti con diritti di amministratore.

### Configurazione base

CentOS non richiede una configurazione specifica se non per il secondo utente da creare con permessi root (super user), inoltre per lavorare con più comodità spesso si preferisce settare l'IP fisso ed abilitare il server SSH (Secure Shell). Dato che questa guida è stata sviluppata anche all'interno di un ambiente scolastico dove è presente un proxy, bisogna configurare anche quest'ultimo per poter accedere ad internet.

#### Creazione nuovo utente amministratore

Entrare come amministratore sulla VM CentOS.

Creare un nuovo utente chiamato "toor" con password "123toor":

```
# useradd toor
```

Assegnare una password al nuovo utente (altrimenti sarà impossibile eseguire un login su quell'utente):

```
# passwd toor
```

Dopo aver inserito questo comando chiederà di inserire due volte la password.

Al momento l'utente "toor" è di livello normale, senza permessi di amministrativi. Per assegnare i permessi amministrativi, eseguire questo comando.

```
# gpasswd -a demo wheel
```

Adesso l'utente ha lo stesso livello di "root".

#### Configurazione IP Statico

L'IP è numero composto da 4 byte che rappresenta un computer in una rete. Sui server si utilizza la versione statica, in maniera da saper individuare sempre la macchina virtuale. Per configurarlo bisogna loggarsi come amministratore, quindi con l'utente root o toor.

Se siete in una rete molto grande come quella di un'azienda o una scuola ed usate il server per dei test quindi non c'è un contesto di produzione, non utilizzate gli IP statici, perché non si conosce la struttura di rete.

Per configurare l'IP statico bisogna modificare un file di CentOS, per sicurezza prima di modificare un file di sistema conviene crearne una versione di backup.

```
# cp /etc/sysconfig/network-scripts/ifcfg-eth0  
/etc/sysconfig/network-scripts/ifcfg-eth0.back
```

Il file si potrebbe chiamare in maniera differente da quella mostrata, a dipendenza del tipo di scheda di rete.

Poi aprire il file con un editor di testo, per esempio “vi” oppure “nano”.

```
# vi /etc/sysconfig/network-scripts/ifcfg-eth0
```

All'interno del file ci sarà una struttura di questo tipo:

```
DEVICE=<nomeScheda>
HWADDR=<macAddress>
TYPE=<TipoDiScheda>
UUID=<identificativoScheda>
ONBOOT=no
NM_CONTROLLED=yes
BOOTPROTO=dhcp
```

Modificarlo come segue:

```
DEVICE=<nomeScheda>
HWADDR=<macAddress>
TYPE=<TipoDiScheda>
UUID=<identificativoScheda>
ONBOOT=yes
NM_CONTROLLED=no
BOOTPROTO=static
IPADDR=<ipAddr>
NETMASK=<netmask>
GATEWAY=<gateway>
```

i campi che sono uguali sia sopra che sotto non modificarli, mentre gli altri aggiungeteli e modificarli, nel campo <ipAddr> inserire l'IP desiderato per il server. Si consiglia di assegnare un indirizzo al di fuori del range DHCP, per evitare problemi (più macchine con un indirizzo uguale), la <netmask> bisogna inserire la netmask della rete alla quale ci si collega. Mentre il <gateway> si riferisce al punto di accesso ad internet.

### *Abilitare il Server SSH (Secure Shell)*

Il server SSH serve per poter connettersi in remoto alla macchina conoscendone l'indirizzo IP e un nome utente abilitato alla connessione remota tramite questo protocollo e la password. Può essere usato anche per trasferimento sicuro di file.

Su CentOS normalmente è già abilitato, bisogna solo abilitarlo per l'utente amministratore, che per questioni di sicurezza viene disabilitato. Aprire il file di configurazione del protocollo SSH. Come per l'indirizzo IP conviene creare una copia di backup per il file di configurazione.

```
# cp /etc/ssh/sshd_config /etc/ssh/sshd_config.back  
# vi /etc/ssh/sshd_config
```

Cercare la riga contenente “PermitRootLogin”, se è preceduto da un cancelletto rimuovete il cancelletto e assegnateli il valore yes.

### Configurare il Proxy

Per configurare il server Proxy sul comando “yum” che server per la gestione dei pacchetti. Bisogna aggiungere al file di configurazione /etc/yum.conf una stringa.

```
# vi /etc/yum.conf
```

Aggiungere la stringa in fondo al file:

```
proxy=http://<nomeUtente>:<password>@<indirizzoProxy>:<Porta>
```

Salvare ed uscire dal editor di testo.

### Aggiornamento sistema operativo

Per aggiornare il sistema operativo CentOS si può utilizzare il comando per la gestione dei pacchetti, YUM, il comando per l’aggiornamento dei pacchetti va eseguito come amministratore (quindi nel nostro caso root o toor)

```
# yum update
```

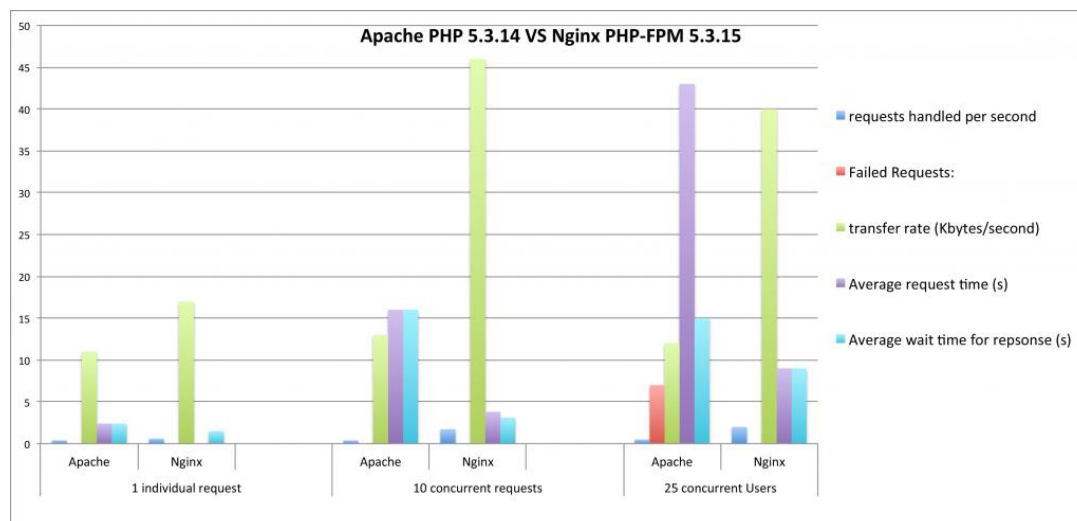


## Installazione NGINX (servizio http)

### Introduzione

NGINX è un servizio http, il servizio utilizzato dai browser per scaricare i file per la costruzione delle pagine web.

NGINX per siti di produzione è molto più performante rispetto ad Apache, software per servizio http molto diffuso e comune. Come dimostra il grafico sottostante.



<https://theorganicagency.com/blog/apache-vs-nginx-performance-comparison/>

### Installazione

Per installare NGINX utilizzeremo il gestore di pacchetti di CentOS, YUM. Prima di installare il servizio, bisogna installare per prima cosa un pacchetto extra, EPEL (Extra Packages for Enterprise Linux) con il comando

```
# yum install epel-release
```

Poi bisogna installare NGINX con il comando

```
# yum install nginx
```

Dopo aver installato il servizio bisogna avviarlo, con il comando dedicato hai servizi avviare NGINX.

```
# systemctl start nginx
```

Dopo aver avviato il servizio provare a connettersi all'indirizzo del server con un normale browser, come Chrome o Firefox. Nel caso in cui è andato tutto per il verso giusto si dovrebbe visualizzare pagina come quella sottostante.



Nel caso in cui si visualizzasse un messaggio di errore, probabilmente bisogna disabilitare il firewall interno di CentOS sulle porte destinate al servizio HTTP e HTTPS.

Per disabilitare il firewall sul servizio HTTP bisogna usare il comando:

```
# firewall-cmd --permanent --zone=public --add-service=http
```

Mentre per il servizio HTTPS basta scrivere https al posto di http:

```
# firewall-cmd --permanent --zone=public --add-service=https
```

Per effettuare le modifiche bisogna riavviare il firewall. Con il comando:

```
# firewall-cmd --reload
```

Adesso si dovrebbe poter visualizzare la pagina, come nell'immagine sopra.

## Configurazione Virtual Host

Per configurare i virtual host bisogna modificare i file di NGINX, si trovano nella cartella /etc/nginx/ il file principale è il file nginx.conf.

In particolare bisogna creare un nuovo file con il percorso /etc/nginx/conf.d/<nomesito>.conf inserendo nel file ciò che segue:

```
server {
    listen      80;
    server_name <virtual-host-name>;
```

```
# note that these lines are originally from the "location /" block
root    /usr/share/nginx/<virtual-host-name>;
index   index.html index.htm;

location / {
    try_files $uri $uri/ =404;
}
error_page 404 /404.html;
error_page 500 502 503 504 /50x.html;
location = /50x.html {
    root /usr/share/nginx/html;
}
}
```

Per ogni sito creare un file, per mantenere un ordine logico si consiglia di nominare tutti i file di tutti i virtual-host con lo stesso senso, per esempio nome.com.conf, e per le cartelle contenenti i siti conviene nominare alla stessa maniera del file di configurazione, con l'unica differenza non si mette l'estensione del file .conf.

## Installazione DB (MySQL - MariaDB)

### Introduzione

Software per database ce ne sono una marea, noi useremo una versione di MySQL gratuita, MariaDB. Questa versione è compatibile con il linguaggio di scripting che installeremo nel prossimo capitolo della guida.

### Installazione

Per installare MariaDB utilizzare il comando:

```
# yum install mariadb-server mariadb
```

Dopo aver installato il servizio avviare il servizio, con il comando:

```
# systemctl start mariadb
```

poi dopo aver avviato il servizio eseguire l'installazione dei mysql, la configurazione guidata si avvia con il comando:

```
# mysql_secure_installation
```

Questo richiederà di inserire la password di root per mysql, dal momento che non ne abbiamo ancora inserita nessuna premere enter e continuare, chiederà se si vuole cambiare la password, cambiatela. A dipendenza del contesto inseritene una più o meno facile, per esempio in un ambiente produttivo si consiglia una password molto complicata, mentre in ambiente di test una facile va benissimo comunque. Poi chiederà se l'accesso da remoto per l'utente root, in ambiente di test è consigliato lasciarlo attivo, anche in ambiente produttivo, basta ricordarsi di impostare una password molto complessa. Ora dobbiamo abilitare il servizio. Utilizzate il seguente comando:

```
# yum enable mariadb
```

Prima di passare al prossimo capitolo conviene fare ancora uno snapshot.

## Installazione Linguaggio Script Lato Server (PHP)

### Introduzione

Per creare dei siti dinamici spesso si utilizzano dei linguaggi di scripting lato server. Quindi interpretati dal Server, che creerà un file interpretabile dai browser comuni e di conseguenza glielo invierà. In questa guida utilizzeremo PHP, con la versione 5. Per testare il funzionamento di PHP utilizzeremo la funzione `phpinfo()`; che crea una pagina con tutte le informazioni relative a PHP.

### Installazione

Per PHP abbiamo bisogno di tre pacchetti, uno relativo al interprete PHP, uno ai tools, ed uno per l'utilizzo di MySQL:

- php
- php-mysql
- php-fpm

Installare tutti i pacchetti necessari.

```
# yum install php php-mysql php-fpm
```

### Configurazione

Per configurare PHP bisogna modificare il file suo file di configurazione. `/etc/php.ini` prima di eseguire qualunque modifica conviene creare una copia del file, così in caso di problemi si può sempre ritornare al punto di partenza.

```
# cp /etc/php.ini /etc/php.ini.back
```

Dopo aver copiato il file bisogna aprirlo per poi modificare alcuni paramentri.

```
# vi /etc/php.ini
```

Cercare il seguente parametro ed impostarlo a 0

```
cgi.fix_pathinfo=0
```

Il prossimo step è configurare php-fpm, modificare il file suo file di configurazione, ma prima bisogna creare una copia del file originale. File di configurazione: `/etc/php-fpm.d/www.conf` dopo copiato aprirlo per eseguire le modifiche.

```
# cp /etc/php-fpm.d/www.conf /etc/php-fpm.d/www.conf
# vi /etc/php-fpm.d/www.conf
```

Poi modificare le linee che iniziano come quelle qui sotto e modificarle come segue.

```
listen = /var/run/php-fpm/php-fpm.sock
listen.owner = nobody
listen.group = nobody
user = nginx
group = nginx
```

Dopo aver finito la configurazione avviare php-fpm e subito dopo abilitarlo.

```
# systemctl start php-fpm
# systemctl enable php-fpm
```

## Configurazione PHP su NGINX

Bisogna inserire l'interprete PHP su NGINX, bisogna abilitarlo su tutti i virtual host. Quindi aprire un file di configurazione ed aggiungere la configurazione seguente:

```
server {
    listen 80
    ...
    index index.php index.html index.htm
    location ~ /\.php$ {
        try_files $uri =404;
        fastcgi_pass unix:/var/run/php-fpm/php-fpm.sock;
        fastcgi_index index.php;
        fastcgi_param SCRIPT_FILENAME
        $document_root$fastcgi_script_name;
        include fastcgi_params;
    }
}
```

Eseguire tutti questi passaggi per tutti i virtual host su cui si desidera utilizzare PHP. Ora riavviare NGINX.

```
# systemctl restart nginx
```

## Test funzionamento PHP


Per testare il funzionamento di PHP utilizzare la funzione phpinfo(); quindi spostarsi nella cartella di un virtual host ed creare il file info.php poi modificarlo come segue. (Si può creare e modificare con lo stesso comando)

```
# vi /usr/share/nginx/<virtual-host-name>/info.php
```

Adesso inserire nel browser il seguente link.

```
http://<virtual-host-name>/info.php
```

Si dovrebbe aprire una pagina quella sottostante.

| PHP Version 5.4.16  |   |
|---|---|
|  |   |
| System  | Linux localhost.localdomain 3.10.0-693.5.2.el7.x86_64 #1 SMP Fri Oct 20 20:32:50 UTC 2017 x86_64  |
| Build Date  | Nov 15 2017 16:35:24  |
| Server API  | PHP/FastCGI   |
| Virtual Directory Support   | disabled  |
| Configuration File (php.ini) Path   | /etc  |
| Loaded Configuration File   | /etc/php.ini  |
| Scan this dir for additional .ini files   | /etc/php.d  |
| Additional .ini files parsed  | /etc/php.d/comail.ini, /etc/php.d/curl.ini, /etc/php.d/dom.ini, /etc/php.d/fileinfo.ini, /etc/php.d/gd.ini, /etc/php.d/gmp.ini, /etc/php.d/imap.ini, /etc/php.d/mcrypt.ini, /etc/php.d/mysqli.ini, /etc/php.d/mysql.ini, /etc/php.d/openssl.ini, /etc/php.d/pdo.ini, /etc/php.d/pdo_mysql.ini, /etc/php.d/pdo_sqlite.ini, /etc/php.d/phar.ini, /etc/php.d/posix.ini, /etc/php.d/sockets.ini, /etc/php.d/streams.ini, /etc/php.d/sysvshm.ini, /etc/php.d/sysvsem.ini, /etc/php.d/xdiff.ini, /etc/php.d/xmlreader.ini, /etc/php.d/xmlwriter.ini, /etc/php.d/xsl.ini, /etc/php.d/zip.ini |
| PHP API   | 20100412  |
| PHP Extension   | 20100525  |
| Zend Extension  | 220100525   |
| Zend Extension Build  | API220100525.NTS  |
| PHP Extension Build   | API20100525.NTS   |
| Debug Build   | no  |
| Thread Safety   | disabled  |
| Zend Signal Handling  | disabled  |
| Zend Memory Manager   | enabled   |
| Zend Multibyte Support  | provided by mbstring  |
| IPv6 Support  | enabled   |
| DTrace Support  | disabled  |
| Registered PHP Streams  | https, ftps, compress.zlib, compress.bzip2, php, file, glob, data, http, ftp, phar, zip   |
| Registered Stream   | tcp, udp, unix, udg, ssl, sshv3, tls  |

Ora se la pagina di test non interessa più si può eliminare con il seguente comando.

```
# rm /usr/share/nginx/html/info.php
```

## Installare phpMyAdmin

### Introduzione

phpMyAdmin è una web app per la gestione dei database MySQL.

### Installazione

Per installare phpMyAdmin c'è bisogno di due pacchetti:

- epel-release
- phpmyadmin

Il primo è già stato installato assieme a NGINX, quindi installare solo il secondo.

```
# yum install phpmyadmin
```

Ora creare un collegamento fra la cartella di PHP e la del virtual host al quale si vuole collegare phpMyAdmin, ci sono due opzioni per la pagina di phpMyAdmin, la differenza fra l'una e l'altra il nome con il quale ci si accede, per esempio con il primo si può fare <virtual-host-name>/phpMyAdmin oppure si può fare la richiesta della pagina con sql.<virtual-host-name> usando il secondo metodo bisogna modificare le tabelle DNS del

proprio dominio nel caso in cui il sito è pubblico. Il primo viene inserito in un virtual host già esistente, mentre il secondo bisogna creare un nuovo virtual host con nome `sql.<virtual-host-name>`

### Primo Metodo

Creare il collegamento della cartella di phpMyAdmin nella cartella `/usr/share/nginx/<virtual-host-name>`

```
# ln -s /usr/share/phpMyAdmin /usr/share/nginx/<vh-name>
```

(`<vh-name>` sta per `<virtual-host-name>`)

### Secondo Metodo

Creare il collegamento della cartella di phpMyAdmin come nel primo metodo, l'unica differenza è che il bisogna creare il collegamento in una posizione diversa.

```
# ln -s /usr/share/phpMyAdmin /usr/share/nginx/
```

Dopo aver creato il collegamento bisogna creare il nuovo virtual-host. Per creare un virtual host, utilizzeremo il file di configurazione di un altro virtual host, quindi copiare il file del virtual-host del nome del dominio e poi aprire il nuovo file per modificarlo.

```
# cp /etc/nginx/conf.d/<vh-name>.conf /etc/nginx/conf.d/sql.<vh-name>.conf
# vi /etc/nginx/conf.d/sql.<vh-name>.conf
```

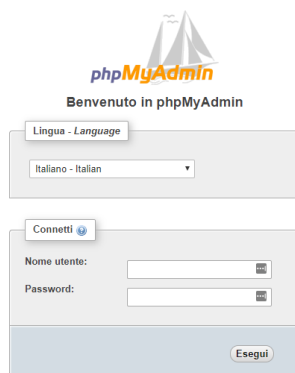
Poi modificare la riga contenente il `server_name` aggiungendo `"sql."` prima del nome del virtual host. Poi modificare la linea con scritto `root` con `"/usr/share/nginx/phpMyAdmin"`

Dopo aver installato phpMyAdmin bisogna riavviare l'interprete php.

```
# systemctl restart php-fpm
```

Ora digitare nel browser digitare l'indirizzo per arrivare a phpMyAdmin, può essere <http://<virtual-host-name>/phpMyAdmin> oppure <http://sql.<virtual-host-name>> a dipendenza di come avete eseguito il passaggio prima. Dovrebbe mostrare una pagina di login come la seguente:

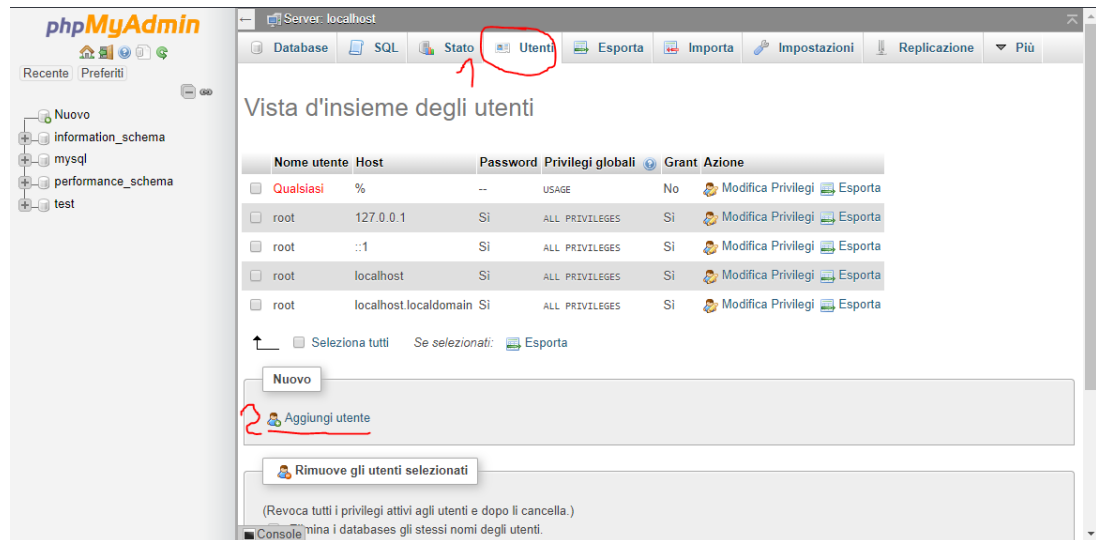
Se ci sono errori di lettura e scrittura controllare che la cartella di phpMyAdmin abbia come proprietario `nginx` e non `apache`, se gli errori non si risolvono controllare anche che la cartella `/var/lib/php` appartenga al gruppo `nginx` e all'utente `root`.



Ora eseguire il login in phpMyAdmin con l'utente `root` e la password che è stata inserita in MySQL. Ora possiamo creare i vari utenti specificati all'inizio della guida.



Dopo essere entrato in phpMyAdmin andare nella sezione utenti. Poi andare su “Aggiungi utente”.



Ora creare il secondo utente amministratore. Selezionare tutte le spunte possibili relative ai permessi sui database e poi cliccare su esegui, ora eseguire tutte operazioni per tutti gli utenti, in questo caso 2 per il cliente, uno amministratore del database e uno che può solo modificarne il contenuto. Quindi creare un nuovo utente con nome cliente\_amm, selezionare le prime due colonne dei privilegi globali e selezionare l’opzione “Crea un database con lo stesso nome e concedi tutti i privilegi”. Infine cliccare su esegui.

Poi creare un utente cliente\_user con solo i permessi di per modificare il contenuto dei db.

## Installazione Server FTP

### Introduzione

Il server FTP (File Transfer Protocol) server per poter aprire delle cartelle in remoto e trasferire file. Noi utilizzeremo il pacchetto ftp vsftpd, che sta per Very Secure FTP Deamon.

### Installazione

Per Installare il pacchetto utilizzare il comando:

```
# yum install vsftpd ftp -y
```

Ora modificare il file di installazione del servizio ftp, che è: /etc/vsftpd/vsftpd.conf

Modificare le seguenti stringhe come segue.

```
## Disable anonymous login ##
anonymous_enable=NO

## Uncomment ##
ascii_upload_enable=YES
ascii_download_enable=YES

## Uncomment - Enter your Welcome message - This is optional ##
ftpd_banner=Welcome to UNIXMEN FTP service.
```

Poi in fondo al file aggiungere la seguente linea di codice.

```
use_localtime=YES
```

Ora bisogna abilitare e avviare il servizio ftp (vsftpd), con i comandi:

```
# systemctl enable vsftpd
# systemctl start vsftpd
```

Per completare l'installazione, abilitare l'installazione aprire la porta 21 del firewall di sistema, dopo averle aperte riavviare il firewall

```
# firewall-cmd --permanent --port=21/tcp
# firewall-cmd --permanent --service=ftp
# firewall-cmd --reload
```

Per completare l'installazione bisogna cambiare il valore di una variabile del sistema operativo.

```
# setsebool -P tftp_home_dir on
```

## Creazione utenti per FTP

Per ogni utente FTP che si vuole creare sul servizio bisogna creare un vero utente sulla macchina, quindi per gli utenti non amministratori bisogna creare un utente senza la shell.

Creare utente amministratore del dominio, quindi creare un gruppo “cliente”.

```
# useradd -G cliente -d /usr/share/nginx/cliente.com/ -s /bin/false  
cliente_admin
```

Ora creare l'utente cliente\_user, che può solo leggere la cartella cliente.com

```
# useradd -G cliente -d /usr/share/nginx/cliente.com/ -s /bin/false  
cliente_user
```

Dopo aver creato gli utenti bisogna modificare i permessi di accesso alla cartella cliente.com, mettere come proprietario cliente\_admin, ed il gruppo cliente, poi modificare i permessi di accesso con il valore 750.

```
# chown cliente_admin:cliente /usr/share/nginx/cliente.com  
# chmod 750 /usr/share/nginx/cliente.com
```

Ora il server FTP dovrebbe funzionare correttamente.

## Installazione Server Mail

### Introduzione

Come servizi di mail utilizzare Postfix e Dovecot perché sono facili da configurare e gestire, sono anche sicuri. Inoltre consentono di avere delle mailbox virtuali, che significa più domini su un solo server, stesso concetto di NGINX con la differenza che qui si parla di e-mail.

### Installazione Postfix

Per poter installare postfix bisogna installare prima un pacchetto chiamato bind-utils

```
# yum install bind-utils -y
```

Poi bisogna configurare il dominio di bind-utils

```
# DOMAIN=<dominio.com>
# NSHOSTS=( "$(dig @4.2.2.2 +short MX ${DOMAIN}|sort -n|cut -d' ' -f2)" )
# for NS in ${NSHOSTS[@]}; do printf "%-15s => %-s\n" "$(dig @4.2.2.2 +short A ${NS})" "${NS}"; done
# unset DOMAIN NSHOSTS
```

Aggiornare il sistema operativo se non lo si ha fatto recentemente (1-2 giorni), conviene farlo comunque.

```
# yum update
```

Creare un gruppo destinato alle virtual mailbox. Assegnandogli l' id 2222. Poi creare un utente sempre per le virtual mailbox appartenente al gruppo vmail, con id 2222 e con directory home /var/vmail.

```
# groupadd vmail -g 2222
# useradd vmail -r -g 2222 -u 2222 -d /var/vmail -m -c "mail user"
```

Ora installare postfix, ma prima bisogna rimuovere sendmail.

```
# yum remove exim sendmail
# yum install postfix cronie
```

Ora configurare postfix, come in tutte le altre configurazioni conviene creare una copia di backup. Dopo aver creato la copia aprite con un editore di testi il file di configurazione.

```
# mv /etc/postfix/main.cf /etc/postfix/main.cf.bak
# vi /etc/postfix/main.cf
```

Ora sostituire il contenuto del file con ciò che segue.

```
queue_directory = /var/spool/postfix
```

```
command_directory = /usr/sbin
daemon_directory = /usr/libexec/postfix
data_directory = /var/lib/postfix
mail_owner = postfix
unknown_local_recipient_reject_code = 550
alias_maps = hash:/etc/postfix/aliases
alias_database = $alias_maps

inet_interfaces = all
inet_protocols = ipv4
mydestination = $myhostname, localhost.$mydomain, localhost

debug_peer_level = 2
debugger_command =
    PATH=/bin:/usr/bin:/usr/local/bin:/usr/X11R6/bin
    ddd $daemon_directory/$process_name $process_id & sleep 5

sendmail_path = /usr/sbin/sendmail.postfix
newaliases_path = /usr/bin/newaliases.postfix
mailq_path = /usr/bin/mailq.postfix
setgid_group = postdrop
html_directory = no
manpage_directory = /usr/share/man
sample_directory = /usr/share/doc/postfix-2.6.6/samples
readme_directory = /usr/share/doc/postfix-2.6.6/README_FILES

relay_domains = *
virtual_alias_maps=hash:/etc/postfix/vmail_aliases
virtual_mailbox_domains=hash:/etc/postfix/vmail_domains
virtual_mailbox_maps=hash:/etc/postfix/vmail_mailbox

virtual_mailbox_base = /var/vmail
virtual_minimum_uid = 2222
virtual_transport = virtual
virtual_uid_maps = static:2222
virtual_gid_maps = static:2222

smtpd_sasl_auth_enable = yes
smtpd_sasl_type = dovecot
smtpd_sasl_path = /var/run/dovecot/auth-client
smtpd_sasl_security_options = noanonymous
smtpd_sasl_tls_security_options = $smtpd_sasl_security_options
smtpd_sasl_local_domain = $mydomain
broken_sasl_auth_clients = yes

smtpd_recipient_restrictions = permit_mynetworks, permit_sasl_authenticated,
reject_unauth_destination
smtpd_relay_restrictions = permit_mynetworks, permit_sasl_authenticated,
reject_unauth_destination
```

Ora bisogna configurare i domini che sfrutteranno questo web-server

```
# vi /etc/postfix_domains
```

Inserire nel file appena aperto i domini come nell'esempio seguente

```
<dominio.com>          OK
<cliente.com>         OK
```

Dopo aver inserito i virtual domain bisogna inserire le mailbox e le loro cartelle di riferimento per il salvataggio delle mail.

```
# vi /etc/postfix/vmail_mailbox
<nome_utente>@<dominio.com>      <dominio.com>/<nome_utente>
<nome_utente>@<cliente.com>      <cliente.com>/<nome_utente>
```

infine bisogna creare il file per gli alias delle caselle di posta.

```
# vi /etc/postfix/vmail-aliases
<nome_utente>@<dominio.com> <nome_utente>@<dominio.com>
<nome_utente>@<cliente.com> <info>@<cliente.com>
```

Ora bisogna collegare postfix ai 3 file che abbiamo appena creato.

```
# postmap /etc/postfix/vmail_domains
# postmap /etc/postfix/vmail_mailbox
# postmap /etc/postfix/vmail_aliases
```

Ora bisogna creare un file che utilizzerà postfix per gli aliases durante la sua attività.

```
# touch /etc/postfix/aliases
```

Dopo aver creato questo file bisogna scommentare una linea di codice in un file di configurazione di postfix.

```
# vi +/submission /etc/postfix/master.cf
```

Ora scommentate la linea come quella seguente

```
submission inet n      -      n      -      -      smtpd
```

## Installazione Dovecot

Installazione dei pacchetti Dovecot.

```
# yum install dovecot
```

Copiare il file di configurazione principale ed aprirlo con un editore di testi.

```
# mv /etc/dovecot/dovecot.conf /etc/dovecot/dovecot.conf.bak
# vi /etc/dovecot/dovecot.conf
```

Inserire le seguenti linee di codice nel file di configurazione di dovecot.

```
listen = *
ssl = no
```

```
protocols = imap lmtp
disable_plaintext_auth = no
auth_mechanisms = plain login
mail_access_groups = vmail
default_login_user = vmail
first_valid_uid = 2222
first_valid_gid = 2222
#mail_location = maildir:~/Maildir
mail_location = maildir:/var/vmail/%d/%n
passdb {
  driver = passwd-file
  args = scheme=SHA1 /etc/dovecot/passwd
}
userdb {
  driver = static
  args = uid=2222 gid=2222 home=/var/vmail/%d/%n allow_all_users=yes
}
service auth {
  unix_listener auth-client {
    group = postfix
    mode = 0660
    user = postfix
  }
  user = root
}
service imap-login {
  process_min_avail = 1
  user = vmail
}
```

Creare un file per i gli utenti di dovecot.

```
# touch /etc/dovecot/passwd
```

Ora bisogna generare i codici hash delle password degli utenti.

```
# doveadm pw -s sha1 | cut -d '}' -f2
```

Questo comando richiede di inserire una password, inserirla e confermarla riscrivendola un'altra volta. Poi verrà generata come questa: “Bf50YcYHwzIpdY1AJQVgEBan00o=”, copiatela ed inseritela insieme alla mail nel file degli utenti che abbiamo creato prima, come segue.

```
<nome_utente>@<domain.com>:Bf50YcYHwzIpdY1AJQVgEBan00o=
```

Poi modificare i permessi del file, impostare come proprietario root, con permessi di lettura e scrittura, mentre gli altri nessun permesso.

```
# chown root: /etc/dovecot/passwd  
# chmod 600 /etc/dovecot/passwd
```

Ora bisogna abilitare i servizi ed avviarli.

```
# chkconfig postfix on  
# chkconfig dovecot on  
# service postfix restart  
# service dovecot restart
```

Per testare il funzionamento del servizio conviene utilizzare un tool chiamato swaks.

Installare swaks

```
# yum install swaks
```

Poi testare il funzionamento del smtp.

```
# swaks --to support@mydomain.com --from email@address.net
```



## Installazione WebMail Client

### Introduzione

Per poter dare al cliente una maggior libertà conviene creare un servizio di WebMail.

### Installazione

Come per tutti software che si installano prima di installare conviene aggiornare il sistema. Quindi eseguire il comando di aggiornamento.

```
# yum update
```

Dopo di che bisogna creare i database mysql.

```
# mysql -u root -p
mysql> CREATE DATABASE IF NOT EXISTS `roundcube`;
mysql> GRANT ALL PRIVILEGES ON `roundcube` . * TO
'roundcube'@'localhost' IDENTIFIED BY 'mySecretPassword';
mysql> FLUSH PRIVILEGES;
mysql> quit
```

Ora bisogna creare il virtual host per roundcube. lo creero webmail.<dominio.com>

```
Vi /etc/nginx/conf.d/webmail.<dominio.com>.conf
```

Ora inserire dentro il file appena creato ciò che segue.

```
server {
    listen      80;
    server_name webmail.<dominio.com>;

    # note that these lines are originally from the "location /"
    block
    root    /var/www/html/roundcube;
    index index.php index.html index.htm;

    location / {
        try_files $uri $uri/ =404;
    }
    error_page 404 /404.html;
    error_page 500 502 503 504 /50x.html;
    location = /50x.html {
        root /usr/share/nginx/html;
    }

    location ~ /\.php$ {
        try_files $uri =404;
```

```

        fastcgi_pass unix:/var/run/php-fpm/php-fpm.sock;
        fastcgi_index index.php;
        fastcgi_param SCRIPT_FILENAME
$document_root$fastcgi_script_name;
        include fastcgi_params;
    }
}

```

Scaricare l'ultima versione di roundcube.

```

# curl -L
"http://sourceforge.net/projects/roundcubemail/files/latest/download?source=files" > /tmp/roundcube-latest.tar.gz
# tar -zxvf /tmp/roundcube-latest.tar.gz -C /var/www/html
# rm -f /tmp/roundcube-latest.tar.gz
# cd /var/www/html
# mv roundcubemail-* roundcube
# chown root: -R roundcube/
# chown apache: -R roundcube/temp/
# chown apache: -R roundcube/logs/

```

Ora per configurare i database di roundcube utilizzeremo uno script SQL già scritto dagli sviluppatori di round cube.

Quindi digitare il comando seguente.

```

# mysql -u roundcube -p"<roundcube_mysql_pass>" roundcube <
roundcube/SQL/mysql.initial.sql

```

Anche di roundcube creare un file di configurazione di backup.

```

# cp roundcube/config/default.inc.php
roundcube/config/default.inc.php.old

```

Poi aprirlo per fare delle modifiche

```

# vim roundcube/config/main.inc.php

```

Alcune variabili vanno modificate, impostarle come segue.

```

$rcmail_config['default_host'] = 'localhost';
$rcmail_config['smtp_server'] = 'localhost';
$rcmail_config['smtp_user'] = '%u';
$rcmail_config['smtp_pass'] = '%p';
$rcmail_config['quota_zero_as_unlimited'] = true;

```

```
$rcmail_config['preview_pane'] = true;  
$rcmail_config['read_when_deleted'] = false;  
$rcmail_config['check_all_folders'] = true;  
$rcmail_config['db_dsnw'] =  
'mysqli://roundcube:<roundcube_mysql_pass>@localhost/roundcube';
```

Ora rimuovere la cartella di installazione. Per evitare che si possano cambiare dei parametri di roundcube.

```
# rm -rf roundcube/installer/
```

Per finire l'installazione riavviare NGINX con il comando:

```
# systemctl restart nginx
```

Ora per visualizzare il webmail aprire il browser ed andare sulla pagina `webmail.<dominio.com>`

## Conclusione

---

L'installazione è conclusa, ora il vostro server dovrebbe funzionare correttamente.