

Il Raspberry PI diventa il tuo cloud “personale”



x864garage.com

Comandi terminale in verde sfondo nero, return/invio in rosso

1: Scarichiamo Raspbian Lite o Standard.

In questo caso ho scaricato la versione con interfaccia grafica, visto che lo utilizzerò anche come postazione per la navigazione internet. (Utilizzando la versione Lite le risorse dedicate al desktop grafico saranno sfruttabili dal server, con ovvio incremento di prestazioni)

<https://www.raspberrypi.org/downloads/raspbian/>

The screenshot shows the Raspbian download page with three options:

- Raspbian Stretch with desktop and recommended software**
Image with desktop and recommended software based on Debian Stretch
Version: April 2019
Release date: 2019-04-08
Kernel version: 4.14
Release notes: [Link](#)
[Download Torrent](#) [Download ZIP](#)
SHA-256: a3ced697ca0481bb0ab3b1bd42c93eb24de6264f4b70ea0f7b6ecd74b33d83eb
- Raspbian Stretch with desktop**
Image with desktop based on Debian Stretch
Version: April 2019
Release date: 2019-04-08
Kernel version: 4.14
Release notes: [Link](#)
[Download Torrent](#) [Download ZIP](#)
SHA-256: 7e10a446f8e57210d0e9ad02f0c833aabb86e58187b4dc02431aff5a3f1ccb83
- Raspbian Stretch Lite**
Minimal image based on Debian Stretch
Version: April 2019
Release date: 2019-04-08
Kernel version: 4.14
Release notes: [Link](#)
[Download Torrent](#) [Download ZIP](#)
SHA-256: 03ec326d45c6eb6cef848cf9a1d6c7315a9410b49a276a6b28e67a40b11fdcf

2: Scriviamo l'immagine sulla SD card



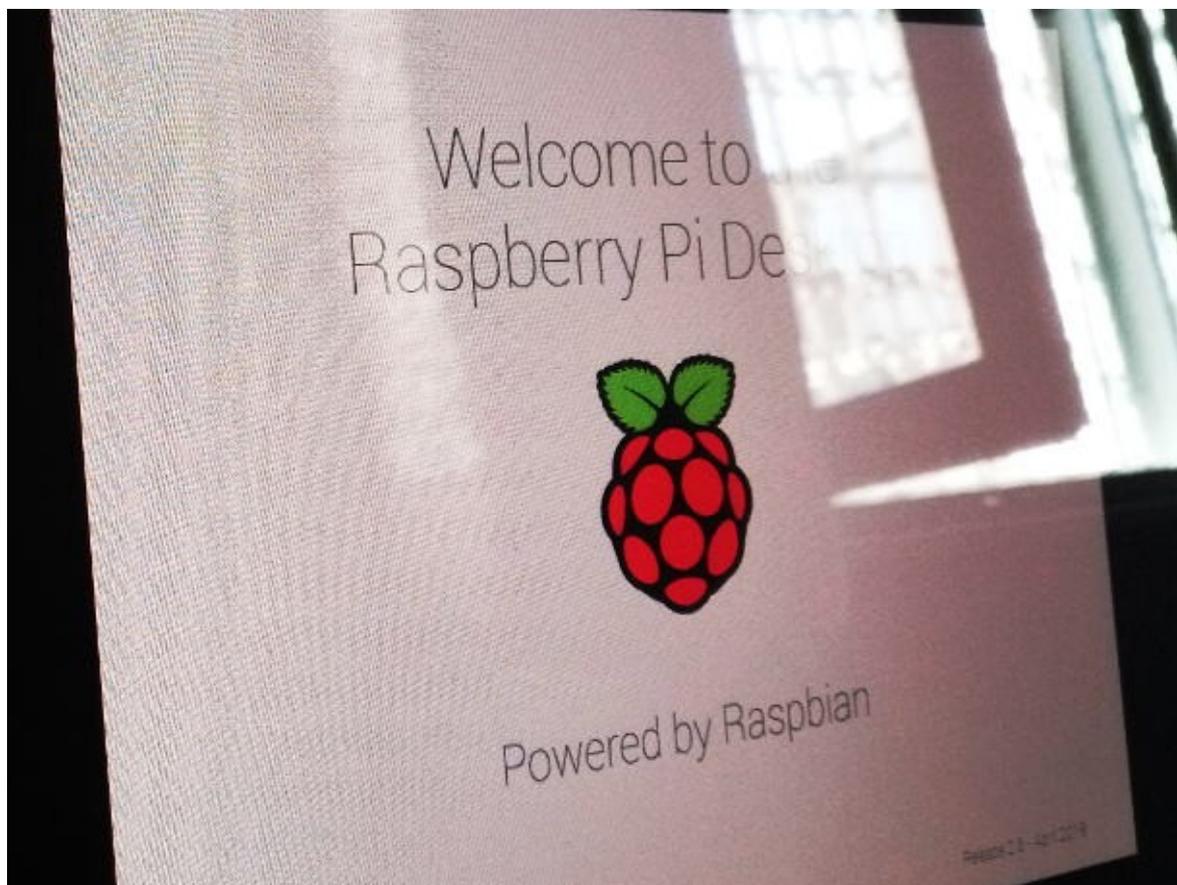
Possiamo scegliere diversi modi per trasferire l'immagine di Raspbian sulla SD (32GB). In Ubuntu creatore di dischi è il più utilizzato, mentre in OpenSuSE la scelta migliore a mio avviso è ImageWriter.

Scompattiamo lo zip e lanciamo la scrittura.

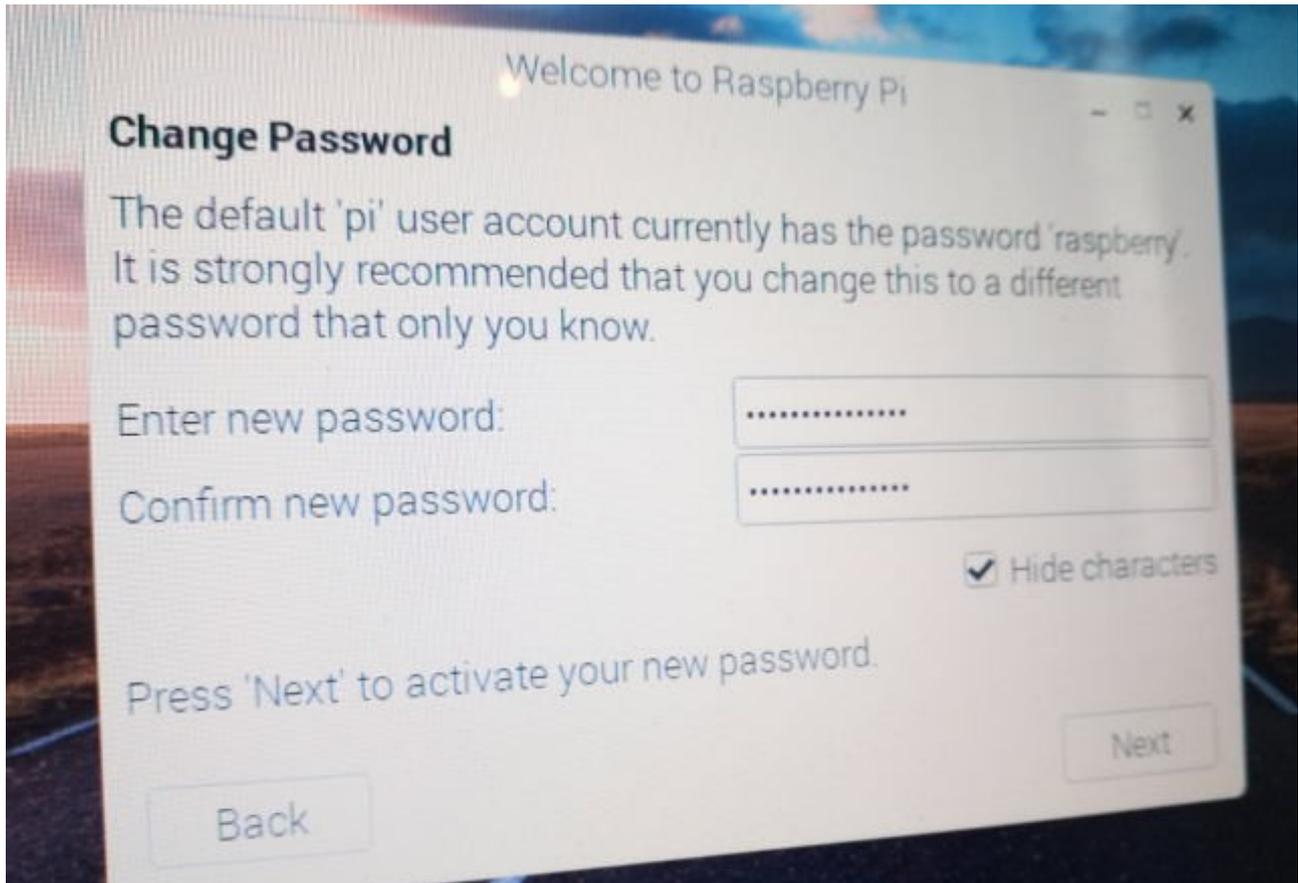




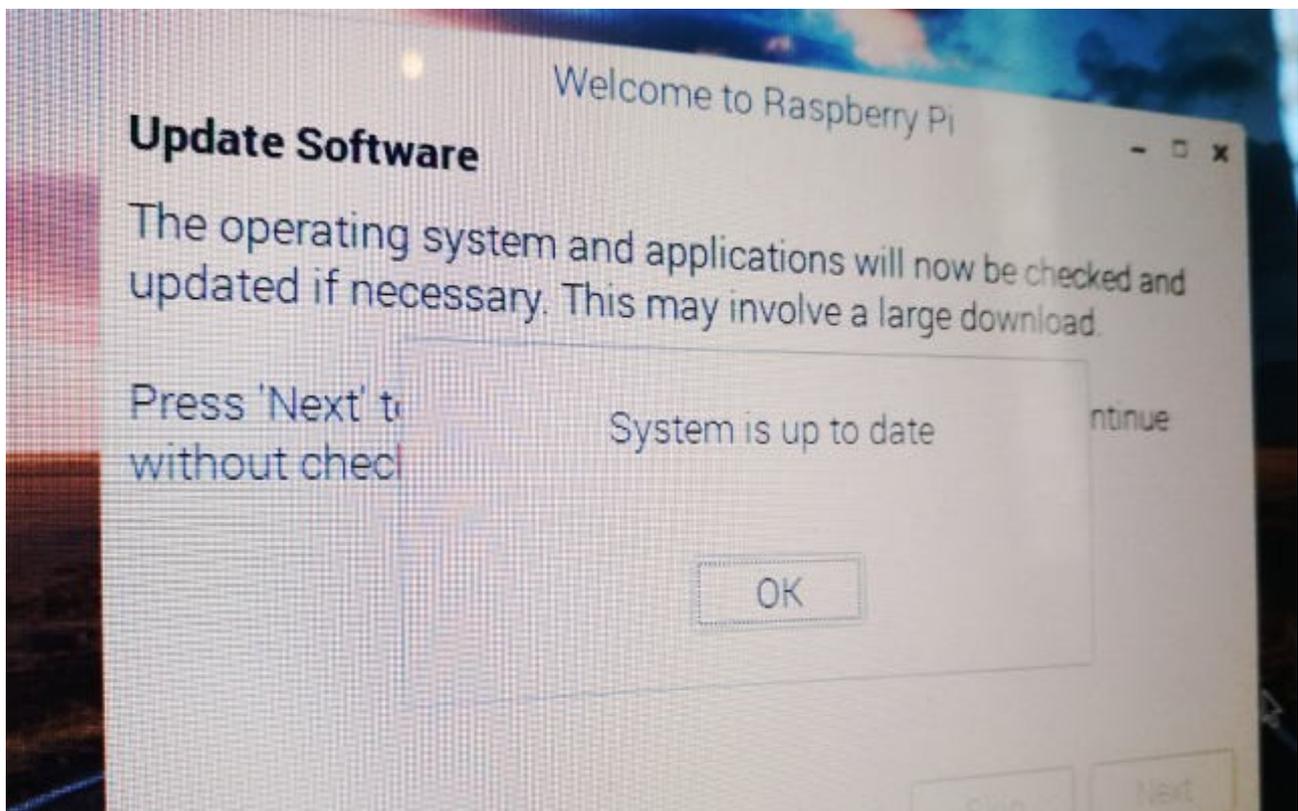
Avviamo il Raspberry!



Prima di tutto cambiamo la password di root.

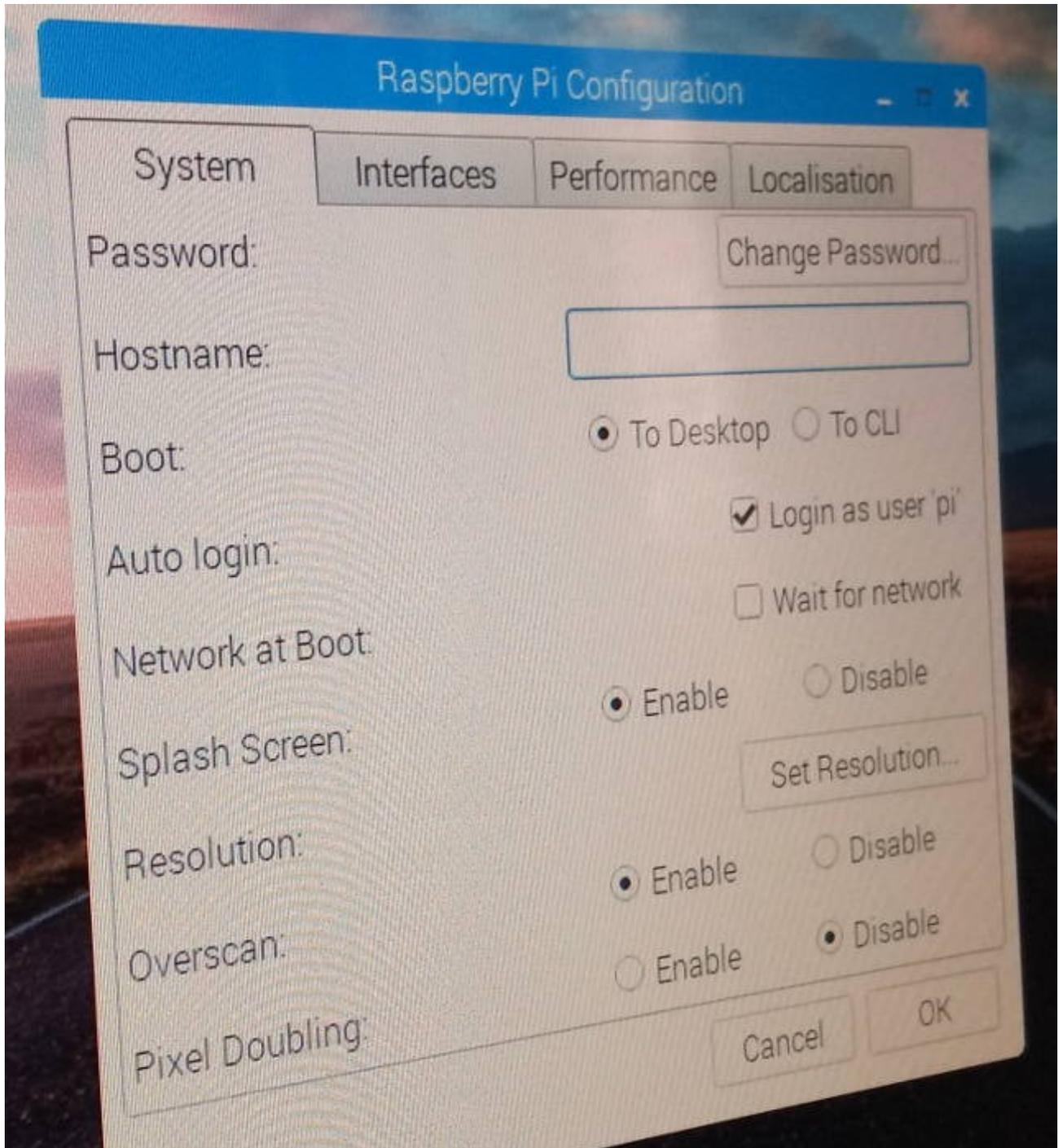


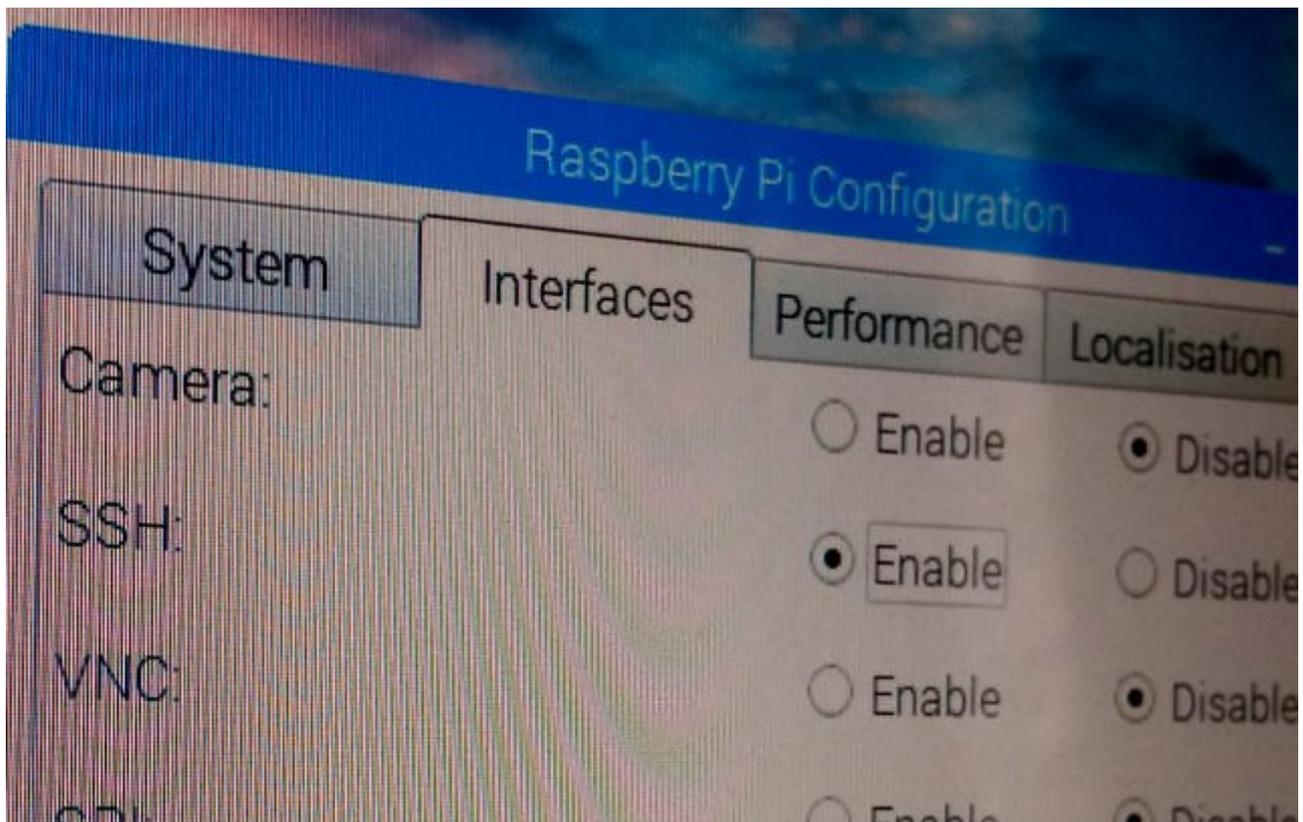
Successivamente facciamo l'update.



Abilitiamo SSH

Dal pannello Preferenze/RaspberryPiConfiguration inseriamo il nome che avrà il dispositivo "Hostname" (pippo/pluto/topolino... ecc).
Disabilitiamo Autologin.





Nel pannello Interface abilita SSH per controllare e comandare il Raspy da remoto.
In Performance se vuoi puoi ridurre la memoria dedicata alla GPU.

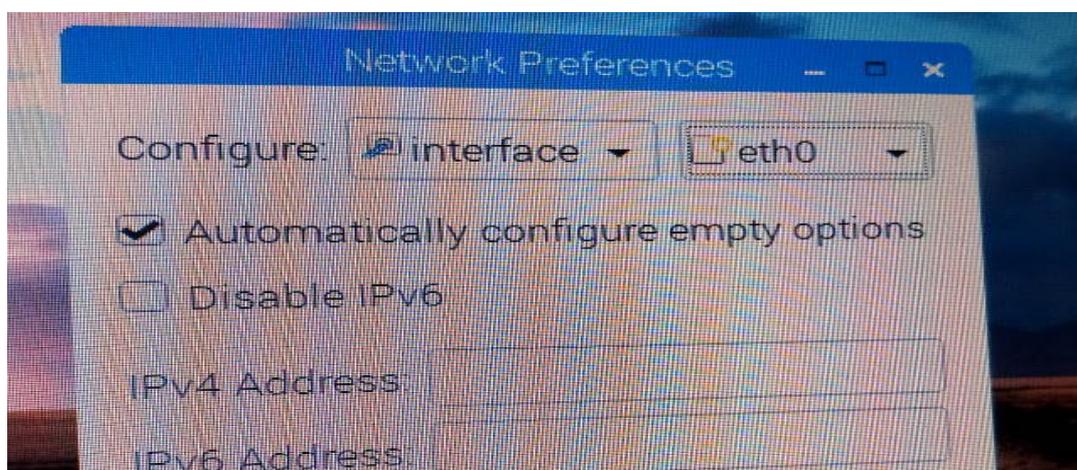
Impostiamo IP Statico

Dobbiamo ora impostare l'interfaccia eth0 in modalità statica.

Esempio:

192.168.1.90/24 (il 24 è la sottomaschera 255.255.255.0)

Maschera 255.255.255.0



Ora riavviamo Apache2 con: `systemctl restart apache2`

Il certificato SSL

Per poter utilizzare SSL prima bisogna creare il proprio certificato (a meno che tu non lo voglia comprare)

Da terminale digitare:

```
cd /etc/ssl/private (invio)
```

Entrato nella directory digita:

```
openssl genrsa -aes128 -out server.key 2048 (invio)
```

```
File Modifica Visualizza Terminale Schede Aiuto
Enabling conf charset.
Enabling conf localized-error-pages.
Enabling conf other-vhosts-access-log.
Enabling conf security.
Enabling conf serve-cgi-bin.
Enabling site 000-default.
Created symlink /etc/systemd/system/multi-user.target.wants/apache2.service → /lib/systemd/system/apache2.service.
Created symlink /etc/systemd/system/multi-user.target.wants/apache-htcacheclean.service → /lib/systemd/system/apache-htcacheclean.service.
Elaborazione dei trigger per libc-bin (2.24-11+deb9u4)...
Elaborazione dei trigger per systemd (232-25+deb9u11)...
root@X864SERVER:/home/pi# nano /etc/apache2/conf-enabled/security.conf
root@X864SERVER:/home/pi# nano /etc/apache2/apache2.conf
root@X864SERVER:/home/pi# nano /etc/apache2/apache2.conf
root@X864SERVER:/home/pi# nano /etc/apache2/sites-enabled/000-default.conf
root@X864SERVER:/home/pi# systemctl restart apache2
root@X864SERVER:/home/pi# cd /etc/ssl/private
root@X864SERVER:/etc/ssl/private# openssl genrsa -aes128 -out server.key 2048
Generating RSA private key, 2048 bit long modulus
.....+++++
.....+++++
e is 65537 (0x010001)
Enter pass phrase for server.key:
Verifying - Enter pass phrase for server.key:
root@X864SERVER:/etc/ssl/private#
```

Scegliere una passphrase e ripetere per controllo.

Ora dobbiamo rimuovere la passphrase dalla private key con:

```
openssl rsa -in server.key -out server.key (invio)
```

ripetere la passphrase per concludere.

```
.....+++++
.....+++++
e is 65537 (0x010001)
Enter pass phrase for server.key:
Verifying - Enter pass phrase for server.key:
root@X864SERVER:/etc/ssl/private# openssl rsa -in server.key -out server.key
Enter pass phrase for server.key:
writing RSA key
root@X864SERVER:/etc/ssl/private#
```

Completiamo i campi per la generazione del certificato con:

```
openssl req -new -days 3650 -key server.key -out server.csr (invio)
```

```
e is 65537 (0x010001)
Enter pass phrase for server.key:
Verifying - Enter pass phrase for server.key:
root@X864SERVER:/etc/ssl/private# openssl rsa -in server.key -out server.key
Enter pass phrase for server.key:
writing RSA key
root@X864SERVER:/etc/ssl/private# openssl req -new -days 3650 -key server.key -out server.csr
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:
```

Infine digitiamo:

```
openssl x509 -in server.csr -out server.crt -req -signkey server.key -days 3650 (invio)
```

```
File Modifica Visualizza Terminale Schede Aiuto
root@X864SERVER:/etc/ssl/private# openssl x509 -in server.csr -out server.crt -req -signkey server.key -days 3650
Signature ok
subject=C = IT, ST = Nations, L = RA, O = home, OU = nocomment, CN = mastabo, emailAddress = noname@mastabo.local
Getting Private key
root@X864SERVER:/etc/ssl/private#
```

Configuriamo SSL in Apache2

Dobbiamo dire ad Apache dove trovare il certificato e dobbiamo settare un'altra volta l'indirizzo mail.

Da terminale diventiamo superuser e digitiamo:

```
nano etc/apache2/sites-available/default-ssl.conf (invio)
```

Sostituiamo la mail alla linea 3 con la nostra (la vostra)

```
ServerAdmin webmaster@latua.tua
```

File: /etc/apache2/sites-available/default-ssl.conf

```
<IfModule mod_ssl.c>
  <VirtualHost _default_:443>
    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/html
```

Cambiamo il percorso del certificato alla linea 32 e 33 con:

```
/etc/ssl/private/server.crt (linea 32)  
/etc/ssl/private/server.key (linea 33)
```

```
GNU nano 2.7.4 File: /etc/apache2/sites-available/default-ssl.conf Modificato  
  
#Include conf-available/serve-cgi-bin.conf  
  
# SSL Engine Switch:  
# Enable/Disable SSL for this virtual host.  
SSLEngine on  
  
# A self-signed (snakeoil) certificate can be created by installing  
# the ssl-cert package. See  
# /usr/share/doc/apache2/README.Debian.gz for more info.  
# If both key and certificate are stored in the same file, only the  
# SSLCertificateFile directive is needed.  
SSLCertificateFile /etc/ssl/private/server.crt  
SSLCertificateKeyFile /etc/ssl/private/server.key
```

Ora digitiamo:

```
a2ensite default-ssl (invio)
```

```
service apache2 reload (invio)
```

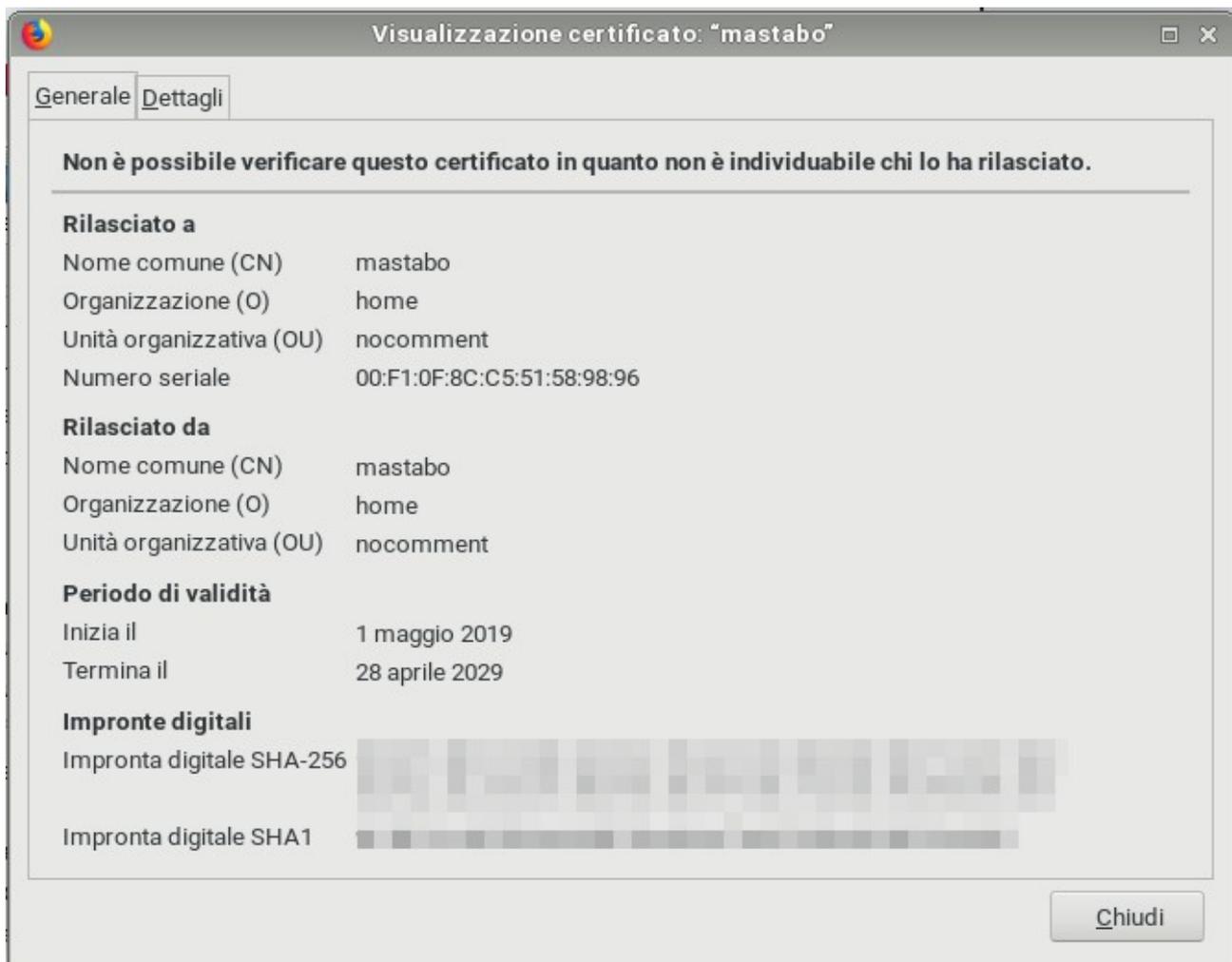
```
a2enmod ssl (invio)
```

```
pi@X864SERVER:~ $ sudo su  
root@X864SERVER:/home/pi# nano /etc/apache2/sites-available/default-ssl.conf  
root@X864SERVER:/home/pi# a2ensite default-ssl  
Enabling site default-ssl.  
To activate the new configuration, you need to run:  
systemctl reload apache2  
root@X864SERVER:/home/pi# service apache2 reload  
root@X864SERVER:/home/pi# a2enmod ssl  
Considering dependency setenvif for ssl:  
Module setenvif already enabled  
Considering dependency mime for ssl:  
Module mime already enabled  
Considering dependency socache_shmcb for ssl:  
Enabling module socache_shmcb.  
Enabling module ssl.  
See /usr/share/doc/apache2/README.Debian.gz on how to configure SSL and create self-sig  
ned certificates.  
To activate the new configuration, you need to run:  
systemctl restart apache2  
root@X864SERVER:/home/pi#
```

Digitiamo ora:

```
systemctl restart apache2 (invio)
```

Ora la nostra istanza di Apache dovrebbe essere accessibile anche in modalità SSL. Ovviamente il browser ci restituirà un messaggio terrificante! “Questo sito non è sicuro” Certo, ce lo siamo generato noi e non abbiamo pagato :-) (ci sono anche altri modi per renderlo trusted ma non lo si vedrà oggi)



Installiamo il database MariaDB

Da terminale digitiamo:

`apt -y install mariadb-server` (invio)

```
File Modifica Visualizza Terminale Schede Aiuto
Configurazione di libreadline5:armhf (5.2+dfsg-3)...
Configurazione di libfcgi-perl (0.78-2)...
Configurazione di libdbi-perl (1.636-1+b1)...
Configurazione di libhttp-date-perl (6.02-1)...
Configurazione di mariadb-server-core-10.1 (10.1.38-0+deb9u1)...
Configurazione di libhtml-template-perl (2.95-2)...
Configurazione di mariadb-client-core-10.1 (10.1.38-0+deb9u1)...
Configurazione di libcgi-fast-perl (1:2.12-1)...
Configurazione di libhttp-message-perl (6.11-1)...
Configurazione di libdbd-mysql-perl (4.041-2)...
Configurazione di mariadb-client-10.1 (10.1.38-0+deb9u1)...
Configurazione di mariadb-server-10.1 (10.1.38-0+deb9u1)...
Created symlink /etc/systemd/system/mysql.service → /lib/systemd/system/mariadb.service.
Created symlink /etc/systemd/system/mysqld.service → /lib/systemd/system/mariadb.service.
Created symlink /etc/systemd/system/multi-user.target.wants/mariadb.service → /lib/systemd/system/mariadb.service.
Configurazione di mariadb-server (10.1.38-0+deb9u1)...
Elaborazione dei trigger per libc-bin (2.24-11+deb9u4)...
Elaborazione dei trigger per systemd (232-25+deb9u11)...
root@X864SERVER: /home/pi#
```

Cambiamo il set dei caratteri da utf8mb4_general_ci a “utf8mb4”

Da terminale digitiamo:

```
nano /etc/mysql/mariadb.conf.d/50-server.cnf (invio)
```

e commentiamo con # la linea con collation-server.

```
# MySQL/MariaDB default is Latin1, but in Debian we rather default to the full
# utf8 4-byte character set. See also client.cnf
#
character-set-server = utf8mb4
#collation-server    = utf8mb4_general_ci
```

Salviamo e usciamo.

Digitiamo ora:

```
systemctl restart mariadb (invio)
```

Rendiamo MySQL più sicura

Da terminale digitiamo:

```
mysql_secure_installation (invio)
```

Se vuoi cambiare la password segui le istruzioni a terminale.

```
root@X864SERVER:/home/pi# mysql_secure_installation

NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB
      SERVERS IN PRODUCTION USE!  PLEASE READ EACH STEP CAREFULLY!

In order to log into MariaDB to secure it, we'll need the current
password for the root user.  If you've just installed MariaDB, and
you haven't set the root password yet, the password will be blank,
so you should just press enter here.

Enter current password for root (enter for none):
OK, successfully used password, moving on...

Setting the root password ensures that nobody can log into the MariaDB
root user without the proper authorisation.

You already have a root password set, so you can safely answer 'n'.

Change the root password? [Y/n] y
New password:
Re-enter new password:
Password updated successfully!
Reloading privilege tables..
... Success!
```

Alle prossime richieste rispondi sempre “Y” per acconsentire.

```
By default, a MariaDB installation has an anonymous user, allowing anyone to log into MariaDB without having to have a user account created for them. This is intended only for testing, and to make the installation go a bit smoother. You should remove them before moving into a production environment.
```

```
Remove anonymous users? [Y/n] y
... Success!
```

```
Normally, root should only be allowed to connect from 'localhost'. This ensures that someone cannot guess at the root password from the network.
```

```
Disallow root login remotely? [Y/n] y
... Success!
```

```
By default, MariaDB comes with a database named 'test' that anyone can access. This is also intended only for testing, and should be removed before moving into a production environment.
```

```
Remove test database and access to it? [Y/n] y
- Dropping test database...
... Success!
- Removing privileges on test database...
... Success!
```

```
Reloading the privilege tables will ensure that all changes made so far will take effect immediately.
```

```
Reload privilege tables now? [Y/n] y
... Success!
```

```
Cleaning up...
```

Installiamo PHPMyAdmin

Con questo installiamo PHPMyAdmin con php7.0

Da terminale digitiamo:

```
apt -y install phpmyadmin php-mbstring php-gettext (invio)
```



Ovviamente selezioniamo Apache2...

Ora il programma ci chiede di configurare automaticamente un database per poter utilizzare PHPMyAdmin scegli si, oppure no se vuoi farne uno a mano.

```
Configurazione in corso di phpmyadmin

Prima di poter essere utilizzato, il pacchetto phpmyadmin deve avere installato
e configurato un database. È possibile farlo anche tramite dbconfig-common.

Nel caso si sia amministratori esperti di database e si voglia fare questa
configurazione in maniera manuale, oppure se il database è preesistente,
rifiutare questa opzione. I dettagli sulle operazioni da svolgere sono molto
probabilmente descritti in /usr/share/doc/phpmyadmin.

Negli altri casi è meglio scegliere questa opzione.

Configurare il database di phpmyadmin con dbconfig-common?

<Si> <No>
```

Immetti ora una password per PHPMyAdmin. Ancora qualche passo e PHPMyAdmin è a posto...

Da terminale digitiamo:

```
mysql -u root -p mysql (invio)
```

(inseriamo la password creata poco fa)

```
update user set plugin="" where user='root'; (invio)
```

```
flush privileges; (invio)
```

```
exit (invio)
```

```
root@X864SERVER:/home/pi# mysql -u root -p mysql
Enter password:
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 21
Server version: 10.1.38-MariaDB-0+deb9u1 Raspbian 9.0

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [mysql]> update user set plugin='' where user='root';
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0

MariaDB [mysql]> flush privileges;
Query OK, 0 rows affected (0.00 sec)

MariaDB [mysql]> exit
Bye
root@X864SERVER:/home/pi#
```

Permettere solo ad alcuni IP l'accesso

In realtà non sempre serve questo, dipende cosa ci devi fare col tuo serverino, comunque puoi impostare da quali IP si può accedere.

Digita da terminale:

```
nano /etc/phpmyadmin/apache.conf (invio)
```

alla linea 8

```
Require ip 127.0.0.1 10.0.0.0/24 192.168.1.82/24 192.168.1.45/24
```

(192.168.1.82/24 192.168.1.45/24 sono nella mia rete in questo caso)

Salvare.

```
GNU nano 2.7.4 File: /etc/phpmyadmin/apache.conf Modificato
# phpMyAdmin default Apache configuration

Alias /phpmyadmin /usr/share/phpmyadmin

<Directory /usr/share/phpmyadmin>
  Options SymLinksIfOwnerMatch
  DirectoryIndex index.php
  Require ip 127.0.0.1 10.0.0.0/24 192.168.1.82/24 192.168.1.45/24
  <IfModule mod_php5.c>
    <IfModule mod_mime.c>
      AddType application/x-httpd-php .php
    </IfModule>
    <FilesMatch ".+\.php$">
      SetHandler application/x-httpd-php
    </FilesMatch>

    php_value include_path .
    php_admin_value upload_tmp_dir /var/lib/phpmyadmin/tmp
    php_admin_value open_basedir /usr/share/phpmyadmin:/etc/phpmyadmin:/var/lib/$
    php_admin_value mbstring.func_overload 0
  </IfModule>
  <IfModule mod_php.c>
    <IfModule mod_mime.c>
```

^G Guida **^O** Salva **^W** Cerca **^K** Taglia **^J** Giustifica **^C** Posizione
^X Esci **^R** Inserisci **^_** Sostituisci **^U** Incolla **^T** Ortografia **^_** Vai a riga

Ora riavviamo Apache2

```
systemctl restart apache2 (invio)
```

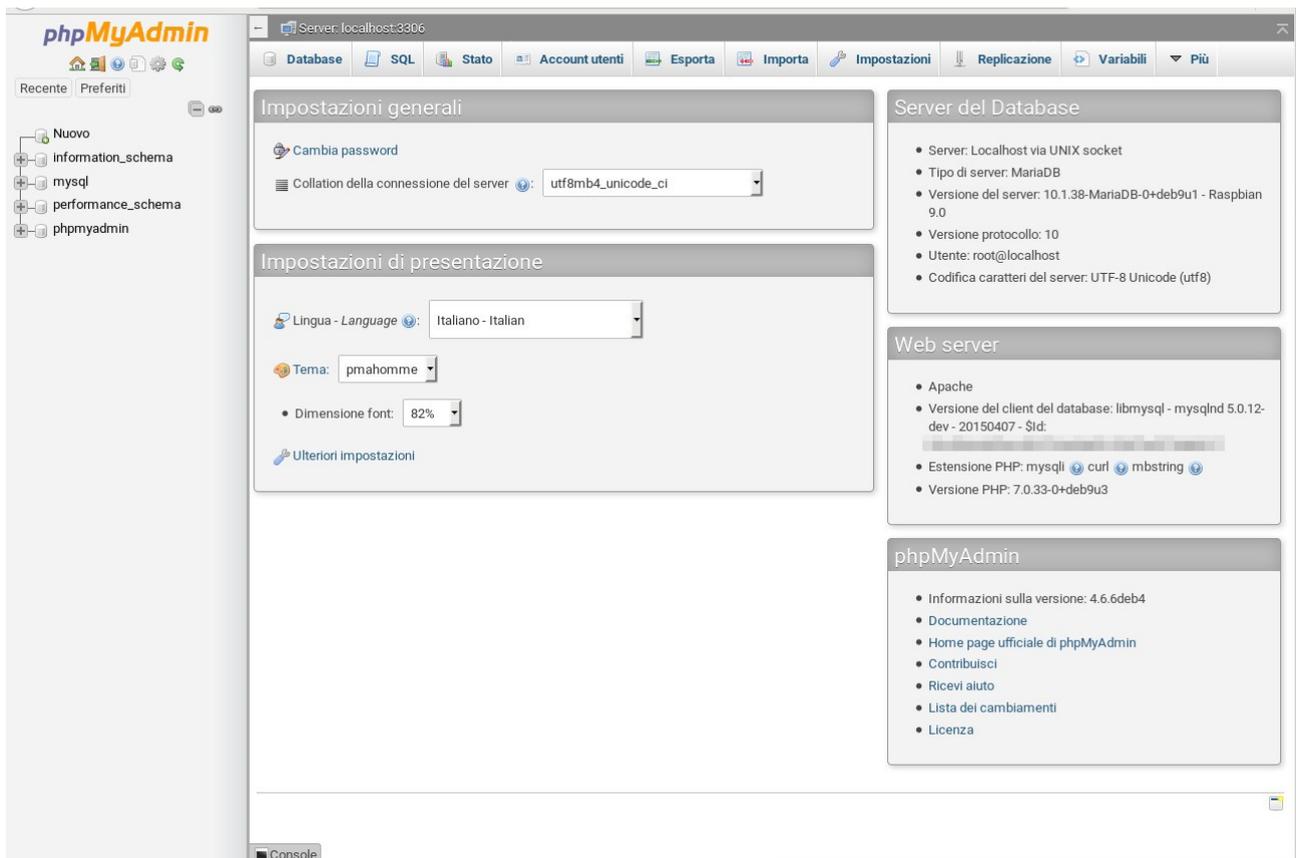
Visitiamo il portale di PHPMyAdmin

Nel browser digitare (in questo caso)

```
https://192.168.1.90/phpmyadmin (invio)
```

User: root

Password: (quella che hai scelto per il tuo sistema)



Scarichiamo Nextcloud “Server”

Archive File Web Installer Appliances

The archive should be extracted in a folder your web server has access to. Latest stable version: 16.0.0 ([Changelog](#))

[Download Nextcloud](#)

Follow the [Nextcloud Admin Manuals installation chapter](#).
If you already run Nextcloud, refer to the [upgrade manual](#).
[Need an enterprise solution?](#)

<https://nextcloud.com/install/#instructions-server>

Apriamo il terminale e scarichiamolo nella cartella Download.

```
wget https://download.nextcloud.com/server/releases/nextcloud-16.0.0.zip
```

(invio)

(Questa versione di Nextcloud è del 2Maggio2019)

```

root@X864SERVER:/home/pi# cd Downloads/
root@X864SERVER:/home/pi/Downloads# wget https://download.nextcloud.com/server/releases/nextcloud-16.0.0.zip
--2019-05-02 17:17:57-- https://download.nextcloud.com/server/releases/nextcloud-16.0.0.zip
Risoluzione di download.nextcloud.com (download.nextcloud.com)... 176.9.217.52, 2a01:4f8:130:32f1::52
Connessione a download.nextcloud.com (download.nextcloud.com)|176.9.217.52|:443... connesso.
Richiesta HTTP inviata, in attesa di risposta... 200 OK
Lunghezza: 79477367 (76M) [application/zip]
Salvataggio in: "nextcloud-16.0.0.zip"

nextcloud-16.0.0.zip 100%[=====>] 75,79M 1,04MB/s in 62s
2019-05-02 17:19:00 (1,23 MB/s) - "nextcloud-16.0.0.zip" salvato [79477367/79477367]
root@X864SERVER:/home/pi/Downloads# █

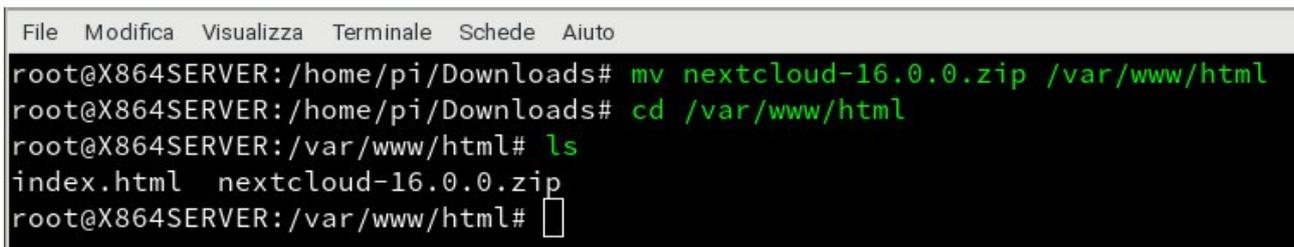
```

Spostiamo il file .zip nella cartella del server:

```
mv nextcloud-16.0.0.zip /var/www/html (invio)
```

```
cd /var/www/html (invio)
```

```
ls (invio)
```



```

File Modifica Visualizza Terminale Schede Aiuto
root@X864SERVER:/home/pi/Downloads# mv nextcloud-16.0.0.zip /var/www/html
root@X864SERVER:/home/pi/Downloads# cd /var/www/html
root@X864SERVER:/var/www/html# ls
index.html nextcloud-16.0.0.zip
root@X864SERVER:/var/www/html# █

```

Rimuoviamo il file html “index.html”

```
rm index.html (invio)
```

Scompattiamo il .zip

```
unzip nextcloud-16.0.0.zip (invio)
```

Rimuoviamo il file zip:

```
rm nextcloud-16.0.0.zip (invio)
```

Se vogliamo, possiamo spostare il contenuto della cartella “nextcloud” nella directory primaria “html”

Portiamoci dentro la cartella Nextcloud e digitiamo:

```
cp -R * /var/www/html (invio)
```

Ora abbiamo il tutto copiato nella directory “html”

```
File Modifica Visualizza Terminale Schede Aiuto
inflating: nextcloud/resources/config/mimetyemapping.dist.json
inflating: nextcloud/resources/config/ca-bundle.crt
inflating: nextcloud/resources/config/mimetyealiases.dist.json
creating: nextcloud/resources/codesigning/
inflating: nextcloud/resources/codesigning/root.crt
inflating: nextcloud/resources/codesigning/root.crl
inflating: nextcloud/resources/codesigning/core.crt
creating: nextcloud/ocm-provider/
inflating: nextcloud/ocm-provider/index.php
root@X864SERVER:/var/www/html# ls
nextcloud nextcloud-16.0.0.zip
root@X864SERVER:/var/www/html# rm nextcloud-16.0.0.zip
root@X864SERVER:/var/www/html# ls
nextcloud
root@X864SERVER:/var/www/html# cd nextcloud/
root@X864SERVER:/var/www/html/nextcloud# cp -R * /var/www/html
root@X864SERVER:/var/www/html/nextcloud# ls
3rdparty  console.php  index.html  ocm-provider  remote.php  status.php
apps      COPYING     index.php   ocs           resources   themes
AUTHORS   core        lib         ocs-provider  robots.txt  updater
config    cron.php    occ         public.php    settings    version.php
root@X864SERVER:/var/www/html/nextcloud# cd ..
root@X864SERVER:/var/www/html# ls
3rdparty  console.php  index.html  occ           public.php  settings    version.php
apps      COPYING     index.php   ocm-provider  remote.php  status.php
AUTHORS   core        lib         ocs           resources   themes
config    cron.php    nextcloud   ocs-provider  robots.txt  updater
root@X864SERVER:/var/www/html#
```

Rimuoviamo la cartella “nextcloud”

```
rm -R nextcloud (invio)
```

Ok, ti sei divertito a portare i file nella cartella html? Bene ora riportali nella cartella Nextcloud :-)
(lo so sono malefico) ricorda che ora la cartella nexcloud non c'è più, se provi con cp ti dirà che non esiste... fai un bel:

```
mkdir nextcloud (invio)
```

rimuovi tutti i file tranne la directory appena creata in var/www/html/

Diamo la possibilità ad Apache di modificare “Nextcloud” digitando:

```
chown -R www-data:www-data /var/www/html/nextcloud/ (invio)
```

Creiamo il database per Nextcloud

In PHPMyAdmin nella sezione utenti creiamo un nuovo utente:

```
User: matrix
Nome Host: localhost
Password: xxxxxx (quella che vuoi)
Password ripeti: xxxxxx
```

Seleziona i due flag per “Database per account utente” e “Privilegi globali” seleziona tutto.

Server: localhost:3306

Database SQL Stato Account utenti Esporta Importa Impostazioni

Aggiungi account utente

Informazioni di Login

Nome utente: Utilizza campo text:

Nome host: Locale localhost

Password: Utilizza campo text:

Re-inserisci:

Plugin di Autenticazione: Autenticazione MySQL nativa

Genera password:

Database per account utente

- Crea un database con lo stesso nome e concedi tutti i privilegi.
- Concedi tutti i privilegi al nome con caratteri jolly (username_%).

Privilegi globali Seleziona tutto

SSL “none” e premiamo su “**esegui**”

Ora portiamoci nella sezione database, selezioniamo quello che abbiamo appena creato e clicchiamo su “Controlla Privilegi”

	Database	Codifica caratteri	Azione
<input type="checkbox"/>	information_schema	utf8_general_ci	Controlla i privilegi
<input checked="" type="checkbox"/>	matrix	utf8mb4_general_ci	Controlla i privilegi
<input type="checkbox"/>	mysql	utf8mb4_general_ci	Controlla i privilegi
<input type="checkbox"/>	performance_schema	utf8_general_ci	Controlla i privilegi
<input type="checkbox"/>	phpmyadmin	utf8mb4_general_ci	Controlla i privilegi
Totale: 5		utf8mb4_general_ci	

clicchiamo per modificare, subito notiamo che il “Grant” per lo specifico DB non è attivo, clicchiamo su modifica privilegi.

Utenti che hanno accesso a "matrix"

	Nome utente	Nome host	Tipo	Privilegi	Grant	Azione
<input type="checkbox"/>	matrix	localhost	globale	ALL PRIVILEGES	Sì	Modifica privilegi
			specifico del database	ALL PRIVILEGES	No	Modifica privilegi
<input type="checkbox"/>	root	localhost	globale	ALL PRIVILEGES	Sì	Modifica privilegi

Seleziona tutto Se selezionati: Esporta

una volta entrati nella sezione “flagghiamo” Grant

Modifica privilegi: Account utente 'matrix'@'localhost' - Database matrix

Privilegi specifici al database Seleziona tutto

Nota: i nomi dei privilegi di MySQL sono espressi in Inglese.

- Dati
 - SELECT
 - INSERT
 - UPDATE
 - DELETE
- Struttura
 - CREATE
 - ALTER
 - INDEX
 - DROP
- Amministrazione
 - GRANT
 - LOCK TABLES
 - REFERENCES

Ora “Esegui”

Aumentiamo il “max upload per php”

Digitiamo nel terminale:

```
nano etc/php/7.0/apache2/php.ini (invio)
```

Alla linea 656 cambia il valore da 8mb a 200mb

Alla linea 809 cambia il valore da 2mb a 200mb

Salviamo e riavviamo Apache2

```
systemctl restart apache2 (invio)
```

Nextcloud 16 non supporta PHP7.0

(si, non trattenerti... inveire contro il mondo viene facile!)

Dobbiamo aggiornare alla versione 7.X

Da terminale aggiungiamo la repo con:

```
nano /etc/apt/sources.list (invio)
```

Copiamo questa riga:

```
deb http://raspbian.raspberrypi.org/raspbian/ buster main contrib non-free rpi
```

Facciamo l'update con:

```
apt-get update && sudo apt-get upgrade (invio) e tanta pazienza
```

Rimuoviamo le altre versioni di php:

```
apt-get remove '^php.*' (invio)
```

Attenzione! Il programma ci chiederà se vogliamo cancellare il DB! NON cancellare il database di PHPMyAdmin!

```
reboot (invio)
```

Installiamo il PHP7.X

```
apt-get install php7.1-fpm php7.1-cli (invio)
```

```
systemctl restart apache2 (invio)
```

```
reboot (invio)
```

Ora dovremmo essere in grado di accedere alla pagina di installazione del Cloud.

```
https://192.168.1.90/nextcloud
```

Crea un account amministratore

root

Password

Archiviazione e database ▾

Cartella dati

/var/www/html/nextcloud/d:

Configura il database

È disponibile solo MySQL/MariaDB. Installa e attiva i moduli PHP aggiuntivi per scegliere gli altri tipi di database.
Per ulteriori dettagli, leggi la documentazione.

Utente del database

Password del database

Nome del database

localhost

Specifica il numero della porta insieme al nome

Crea un account amministratore

admin

Password così-così

Archiviazione e database ▾

Cartella dati

/var/www/html/nextcloud/d:

Configura il database

È disponibile solo MySQL/MariaDB. Installa e attiva i moduli PHP aggiuntivi per scegliere gli altri tipi di database.
Per ulteriori dettagli, leggi la documentazione.

matrix

matrix

localhost

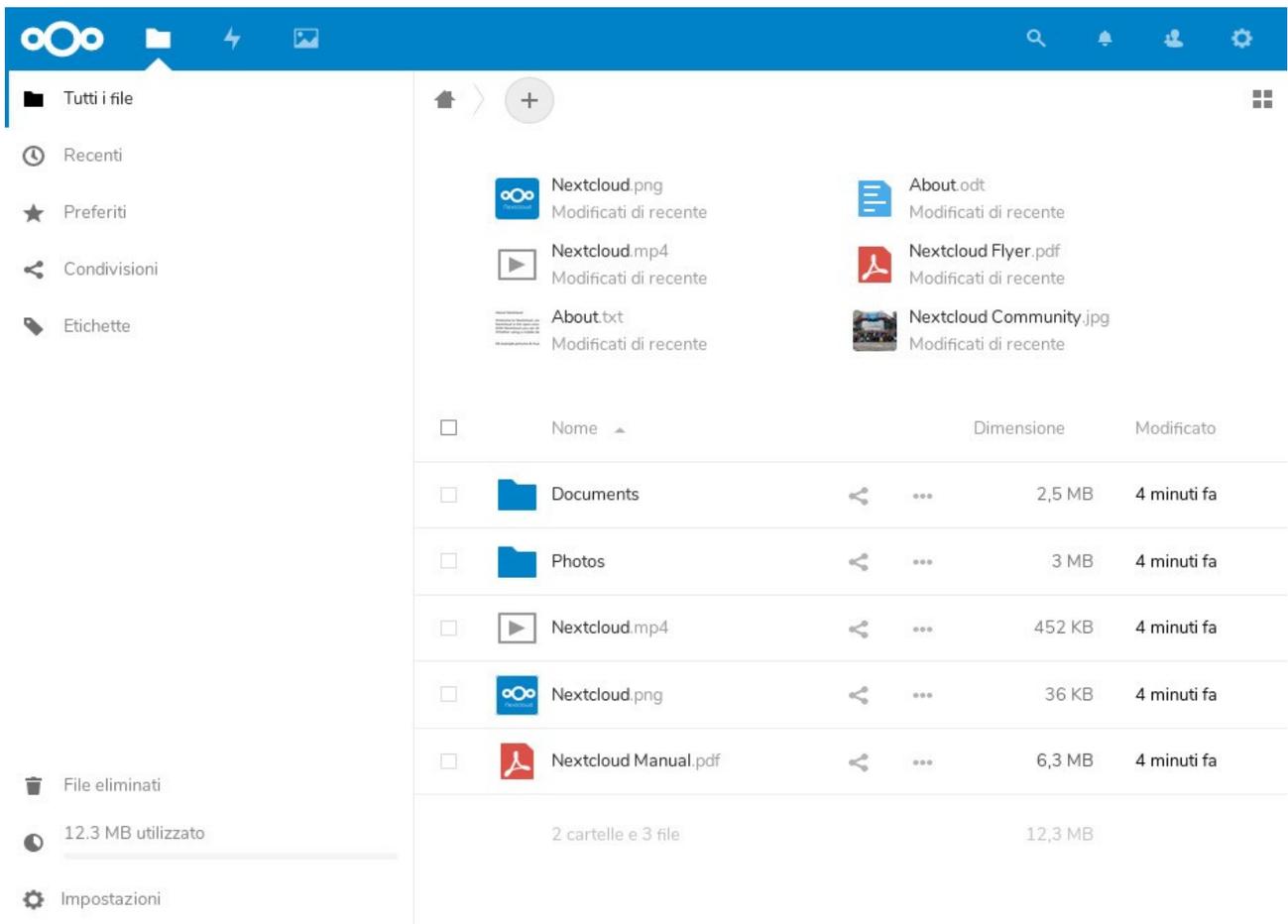
Compiliamo i campi creando l'amministratore e con i dati che abbiamo del database creato appositamente per Nextcloud.

Ora dobbiamo solo aspettare... e il Raspy ci mette un bel po' a popolare il DB... anche 15 minuti!

...fatti pure un caffettino... :-)



“It's works!”



Rendiamo “Fidati gli IP” per il nostro Nextcloud

Spostiamoci nella directory di Nextcloud dove troviamo il file di configurazione.

Da terminale digitiamo:

```
nano config/config.php (invio)
```

```
'trusted_domains' =>  
array (  
0 => 'localhost',  
1 => 'tuomatrixoideipstatico',  
),
```

Salva.

```
GNU nano 2.7.4 File: config.php Modificato  
<?php  
$CONFIG = array (  
    'instanceid' => 'ocywe7ay5xjb',  
    'passwordsalt' => 'opNxHHr474FXpHuCwN5cD150ILKZnz',  
    'secret' => 'abNFSVRJatAGwUpsBMyTMUEQsaiDEZXaQbwckSmb9JKztji5',  
    'trusted_domains' =>  
        array (  
            0 => '192.168.1.90',  
            => 'tuopstatico',  
        ),  
    'datadirectory' => '/var/www/html/nextcloud/data',  
    'dbtype' => 'mysql',  
    'version' => '16.0.0.9',  
    'overwrite.cli.url' => 'https://192.168.1.90/nextcloud',  
    'dbname' => 'matrix',  
    'dbhost' => 'localhost',  
    'dbport' => '',  
    'dbtableprefix' => 'oc_',  
    'dbuser' => 'oc_admin',  
);  
^G Guida ^O Salva ^W Cerca ^K Taglia ^J Giustifica ^G Posizione  
^X Esci ^R Inserisci ^\ Sostituisci ^U Incolla ^T Ortografia ^_ Vai a riga
```

Installiamo il FireWall

Rendiamo difficile ai ficcanaso, chiudiamo tutte le porte che non servono al server.

Da terminale digitiamo:

```
apt install ufw (invio)
```

```
ufw default deny incoming (invio) blocca in entrata
```

```
ufw default allow outgoing (invio) permette in uscita
```

```
ufw allow ssh (invio)
```

Permette le connessioni in SSH remote (non è detto ci serva, se abbiamo settato il sistema con x e lo utilizziamo anche come desktop)

```
ufw allow 80 (invio) apre la porta 80 http
```

```
ufw allow 443 (invio) apre la porta 443 https
```

Ora dobbiamo abilitarlo:

```
ufw enable (invio)
```

```
reboot (invio)
```

Per vedere lo stato del FireWall:

```
ufw status (invio)
```

```
pi@X864SERVER:~ $ sudo su
root@X864SERVER:/home/pi# ufw status
Status: active

To Action From
--
22/tcp ALLOW Anywhere
80 ALLOW Anywhere
443 ALLOW Anywhere
22/tcp (v6) ALLOW Anywhere (v6)
80 (v6) ALLOW Anywhere (v6)
443 (v6) ALLOW Anywhere (v6)

root@X864SERVER:/home/pi#
```

Sicurezza per SSH

Ultimo (lo giuro) tool che non può mancare è il Fail2Ban.
Rende un po' più difficile la vita ai visitatori malevoli.

Da terminale digitiamo:

```
apt-get install fail2ban (invio)
```

Funziona già bene senza toccare parametri ma se proprio vuoi fai riferimento al sito:

```
https://www.fail2ban.org/wiki/index.php/Main\_Page
```

Divertiti!!

Credits:

[Clelio Rossi](#)
[Roberto Perini](#)

```
https://www.x864garage.com
```

