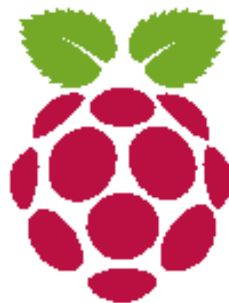
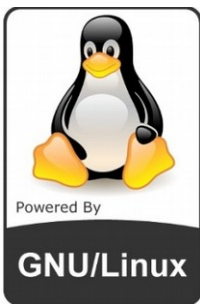


Costruisci il tuo cloud personale

con Raspberry Pi 3
e ownCloud



02/04/2016

Palermo



Via Principe di
Belmonte, 93

organizzato da



**FAB
LAB**
PALERMO

<http://fablabpalermo.org>

info@fablabpalermo.org



<http://viralids.it>



<http://senseiserver.io>



<http://manyhacks.it>



<http://pmocoworking.it>

Intro Workshop



Modalità di svolgimento

- **Durata 8 ore**

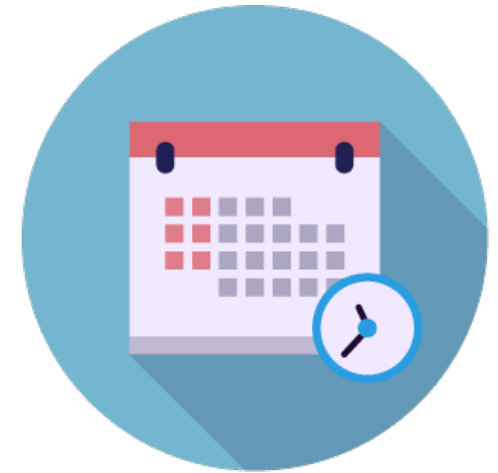
- 10:00 – 13:00 Teoria e setup
- 14:00 – 19:00 Pratica e utilizzo

- **Prima parte**

- Impareremo i fondamenti di Linux
- Installeremo ownCloud e i componenti necessari

- **Seconda parte**

- Ogni partecipante utilizzerà la propria Pi



Scopo del workshop

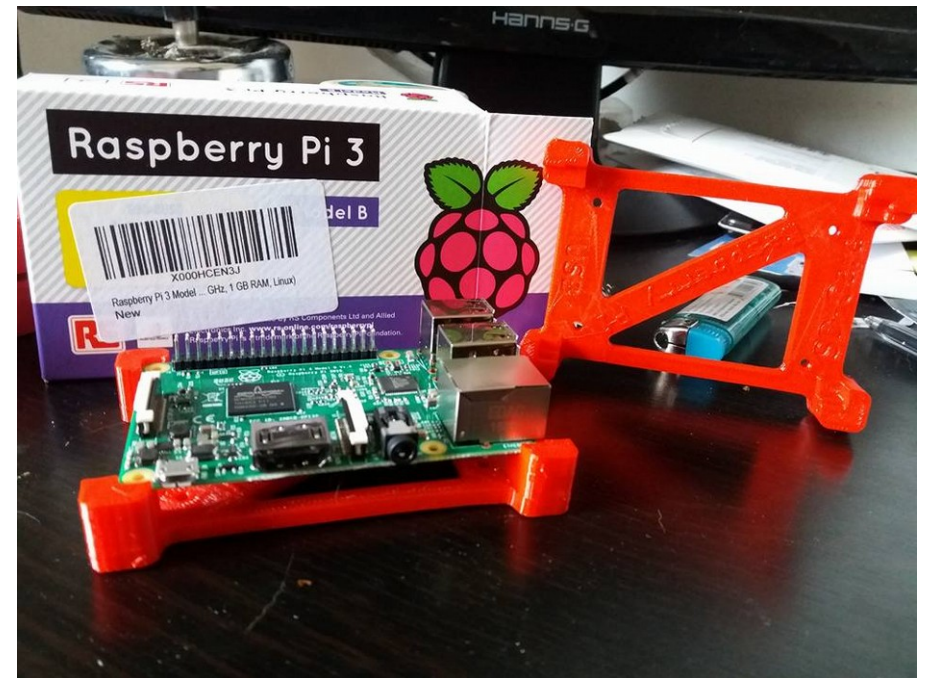
Imparare a utilizzare la Raspberry Pi e costruirsi un proprio cloud storage,

Impareremo:

- concetti base del sistema Linux
- usare Raspberry Pi3
- installare e configurare ownCloud
- effettuare il backup dei dati
- raggiungere la Pi da internet
- sync con dispositivi mobile

Contenuto del KIT

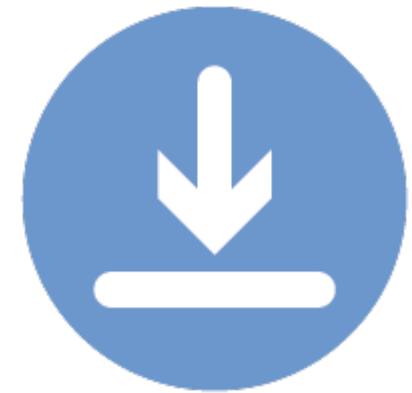
- **Raspberry Pi 3**
 - 1.2 GHz ARMv8 Quad Core
 - 1 GB DDR2
- MicroSD 16 GB
 - SanDisk Ultra
 - Classe 10 - 80 MB/s
- Alimentatore
 - micro usb 2.0 A
- 3D Printed PLA support



Materiali di supporto

● Slideshow

- Questa presentazione
 - Disponibile nella cartella progetto
 - Sul gruppo Facebook
- Altre slideshow e pdf di appunti Linux



● Altro

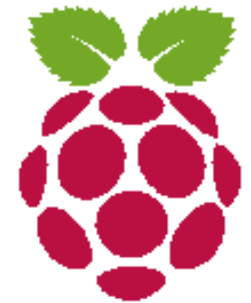
- Immagine raspbian jessie lite (OS di base)
- Bash Script di installazione per ownCloud 8
- Immagine con ownCloud 8 già installato
- Xubuntu 14 Live Linux Image USB key

Intro ARM e Pi [hardware]



Conosciamo la RASPBERRY PI

Il Pi è un **microcomputer** a basso costo, piccolo come una carta di credito, che può essere utilizzato come “cervello” per dare vita a oggetti tecnologici e invenzioni di ogni genere.

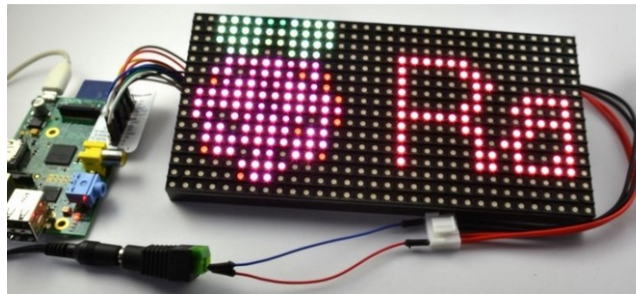
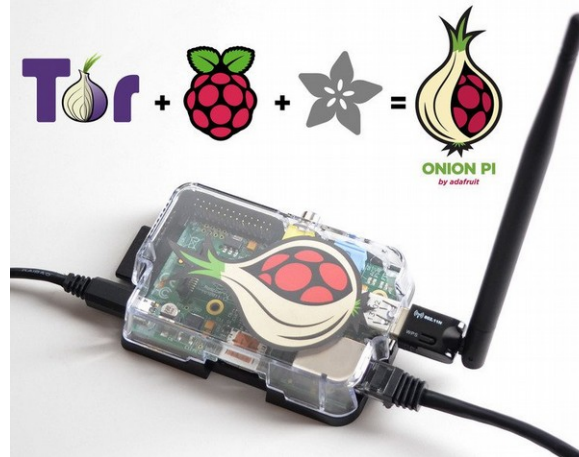


È perfetto per chi desidera imparare le basi della programmazione e dell'informatica, e per i **maker** è una manna dal cielo.

Consuma poco, costa poco, può fare tantissime cose.

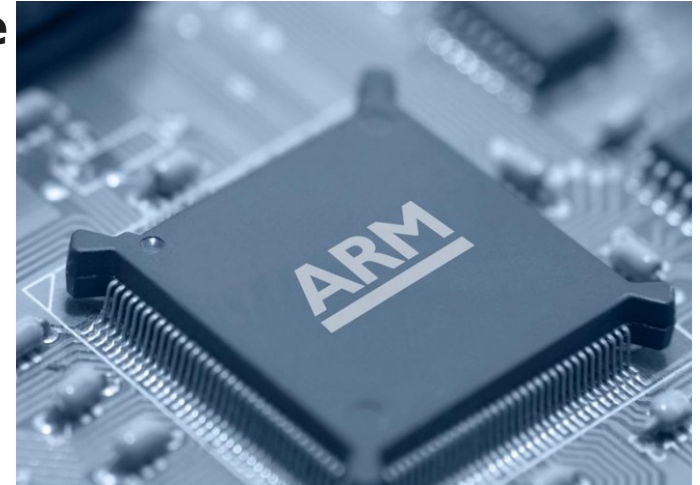
È basata su architettura **ARM** e può fare girare un sistema **GNU/Linux**.

Raspberry Pi projects



Architettura ARM

- **Rapporto potenza/consumo ottimale**
- **Poco surriscaldamento**
- **Compatti**
- **Economici**
- **32 e 64 bit**
- **multicore**
- **Li troviamo spesso nei:**
 - sistemi embedded
 - tablet
 - smartphone



Raspberry e Sistema Operativo

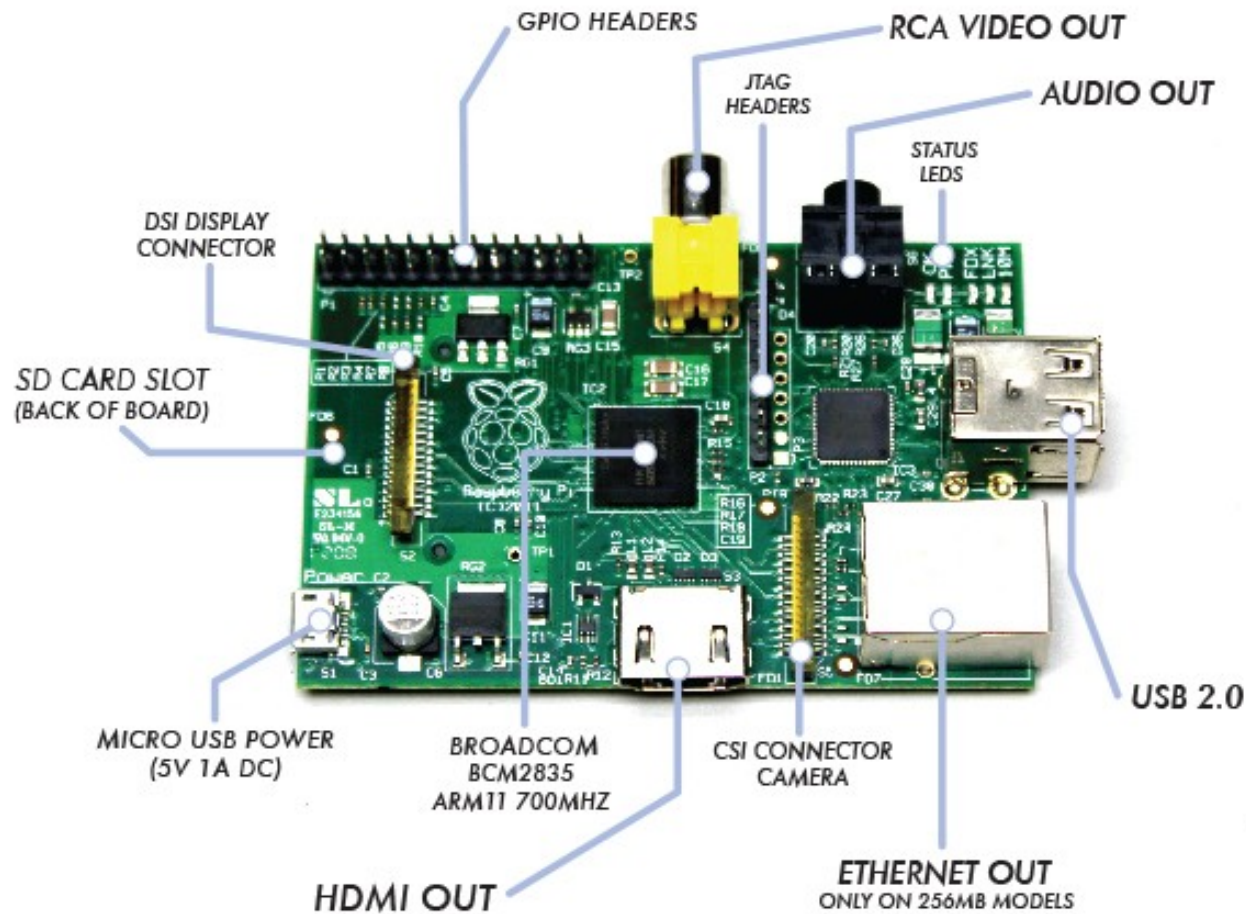
- La Pi non ha una memoria interna
- Il sistema viene ospitato nella **SD card**
 - La memoria **va preparata** come fosse un hard disk, con le partizioni necessarie e il sistema
 - Questo significa anche che semplicemente **cambiando SD** card si può passare a un sistema diverso
 - È possibile utilizzare la Pi con diversi OS o crearsi la propria struttura del sistema modificando il contenuto della SD card
 - Si suggerisce però di partire da una distribuzione raspbian come la jessie lite, scaricabile come file immagine



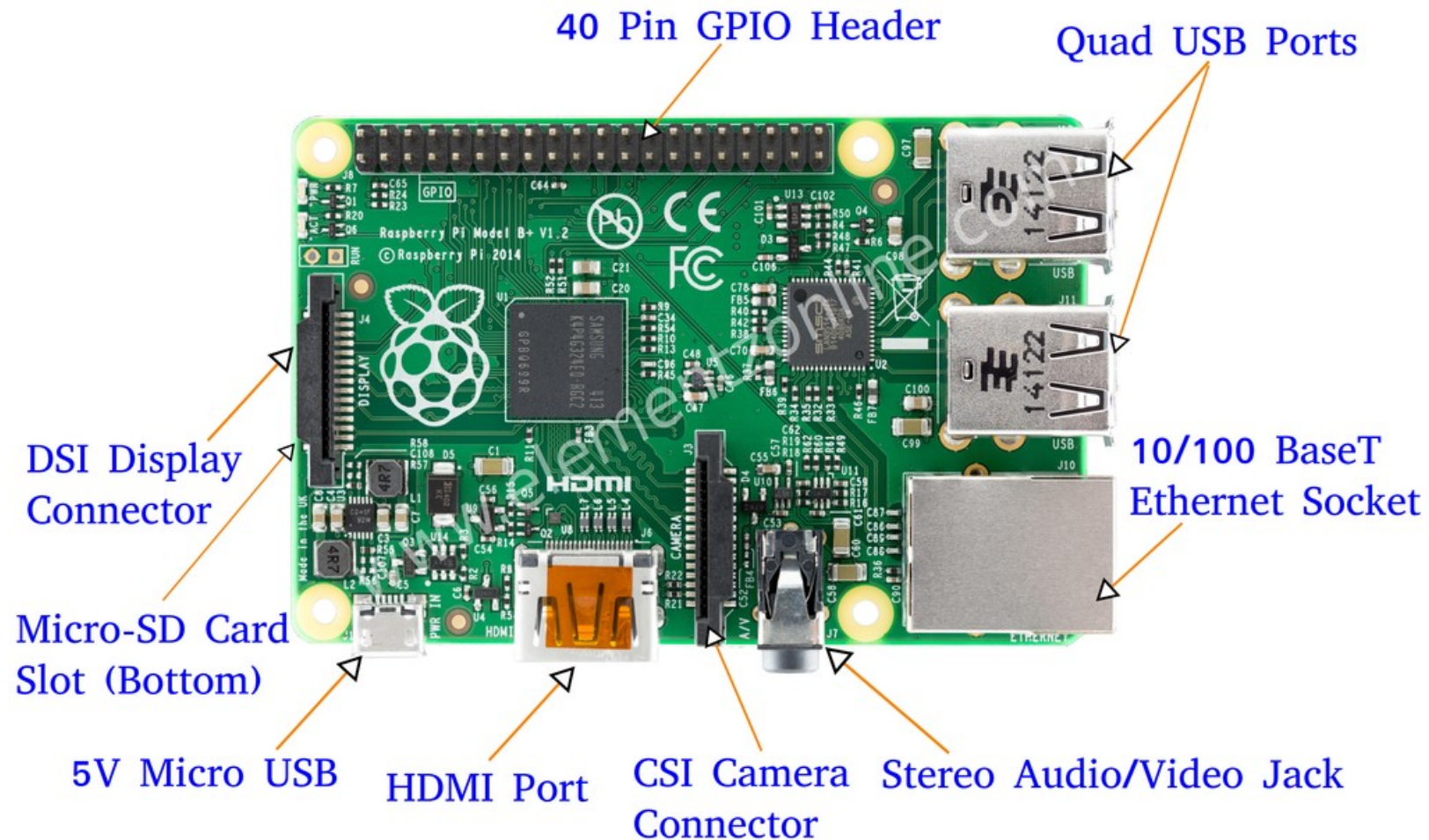
Raspberry Pi i diversi modelli

Modello	CPU	RAM	RAM Bus	Network	WiFi	USB	SATA
Raspberry Pi 3	1.2 GHz ARMv8 Quad Core	1 GB DDR2	450 MHz	100 Mbit	Yes	4	No
Raspberry Pi 2	900 MHz ARMv7 Quad Core	1 GB DDR2	450 MHz	100 Mbit	No	4	No
Raspberry Pi	700 MHz ARMv6 Single Core	512 MB SDRAM	400 MHz	100 Mbit	No	4	No

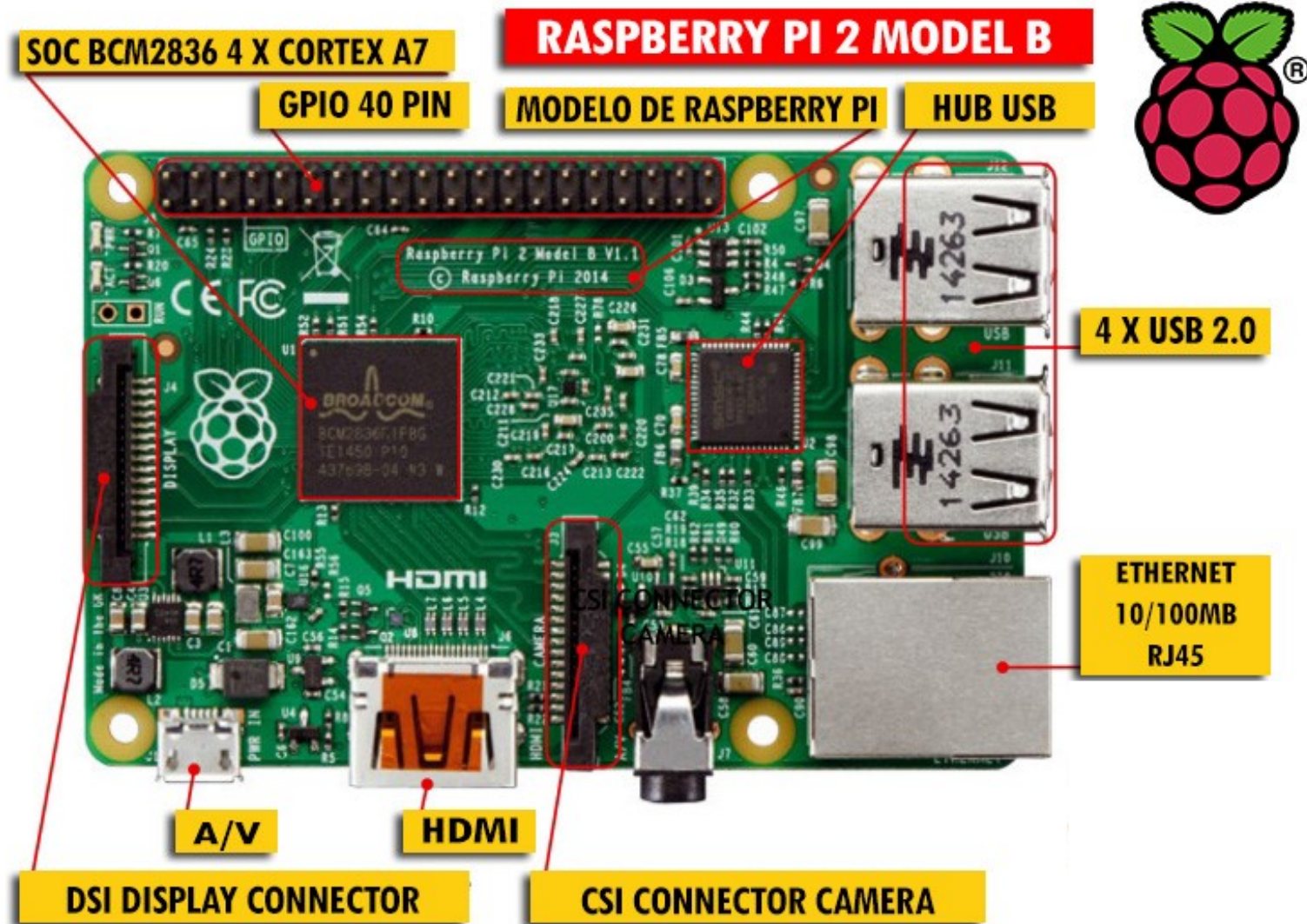
Raspberry Pi model A



Raspberry Pi model B+



Raspberry Pi 2

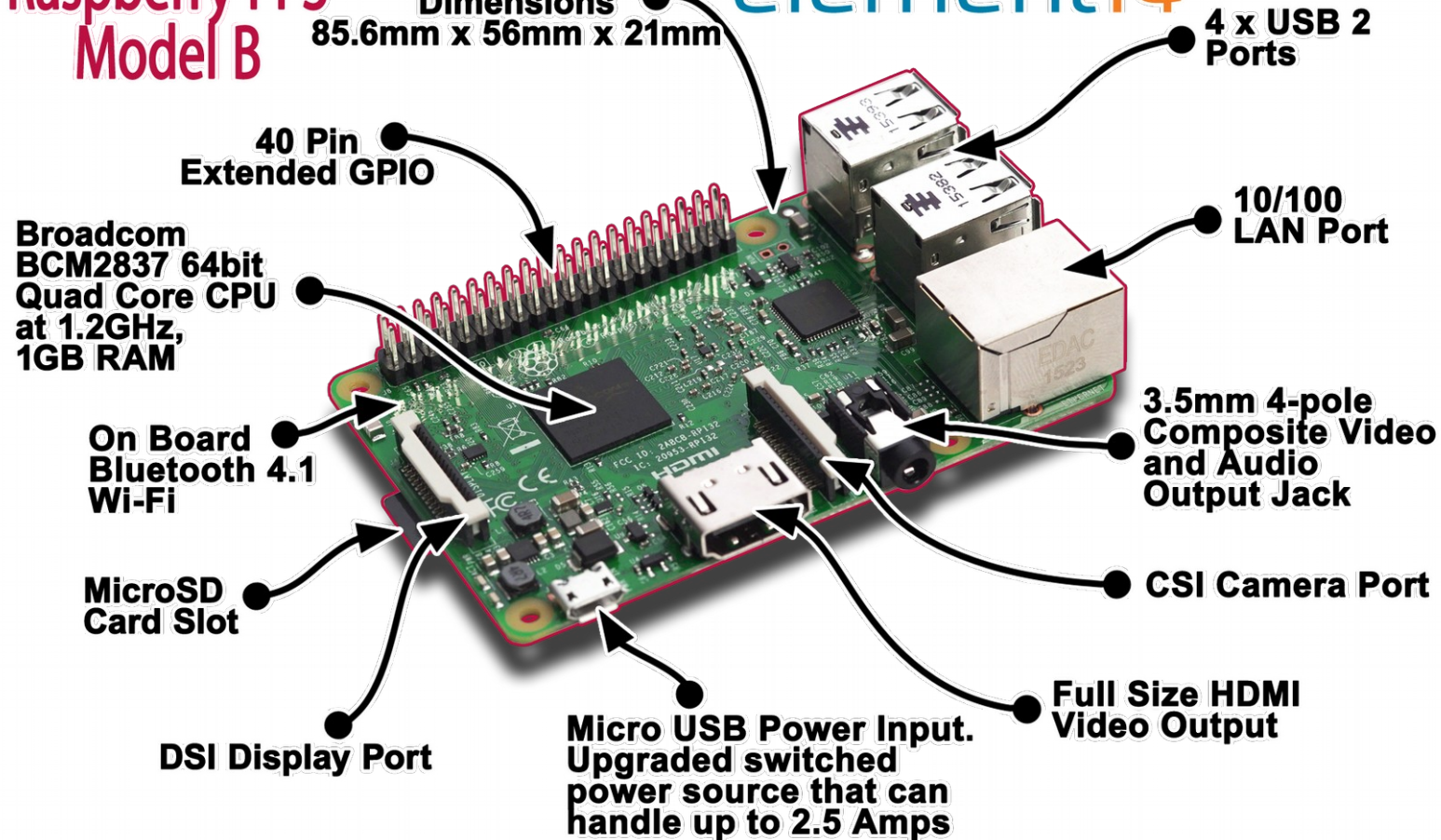


Raspberry Pi 3

Raspberry Pi 3 Model B

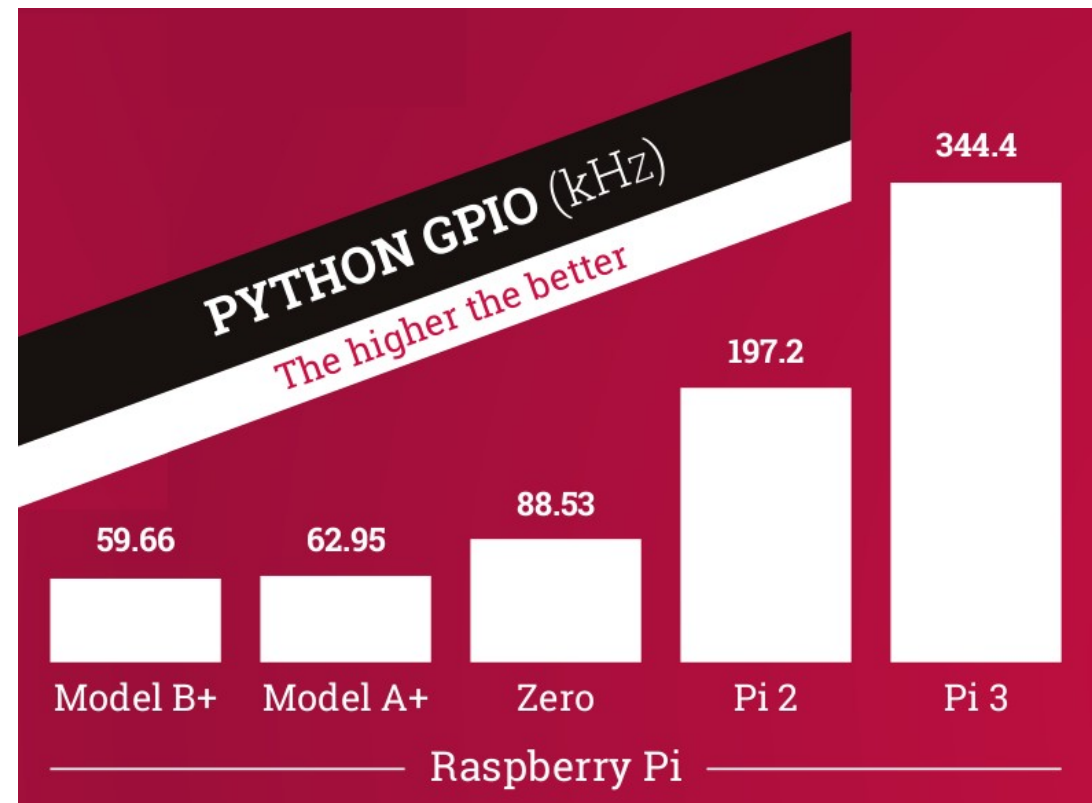
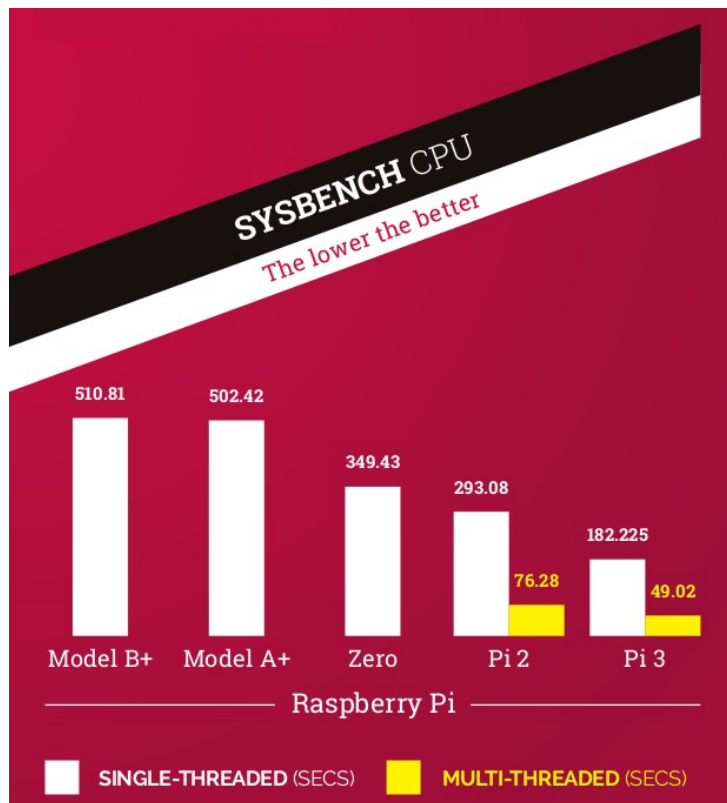
Dimensions
85.6mm x 56mm x 21mm

element14



I diversi modelli di Pi

Le prestazioni a confronto

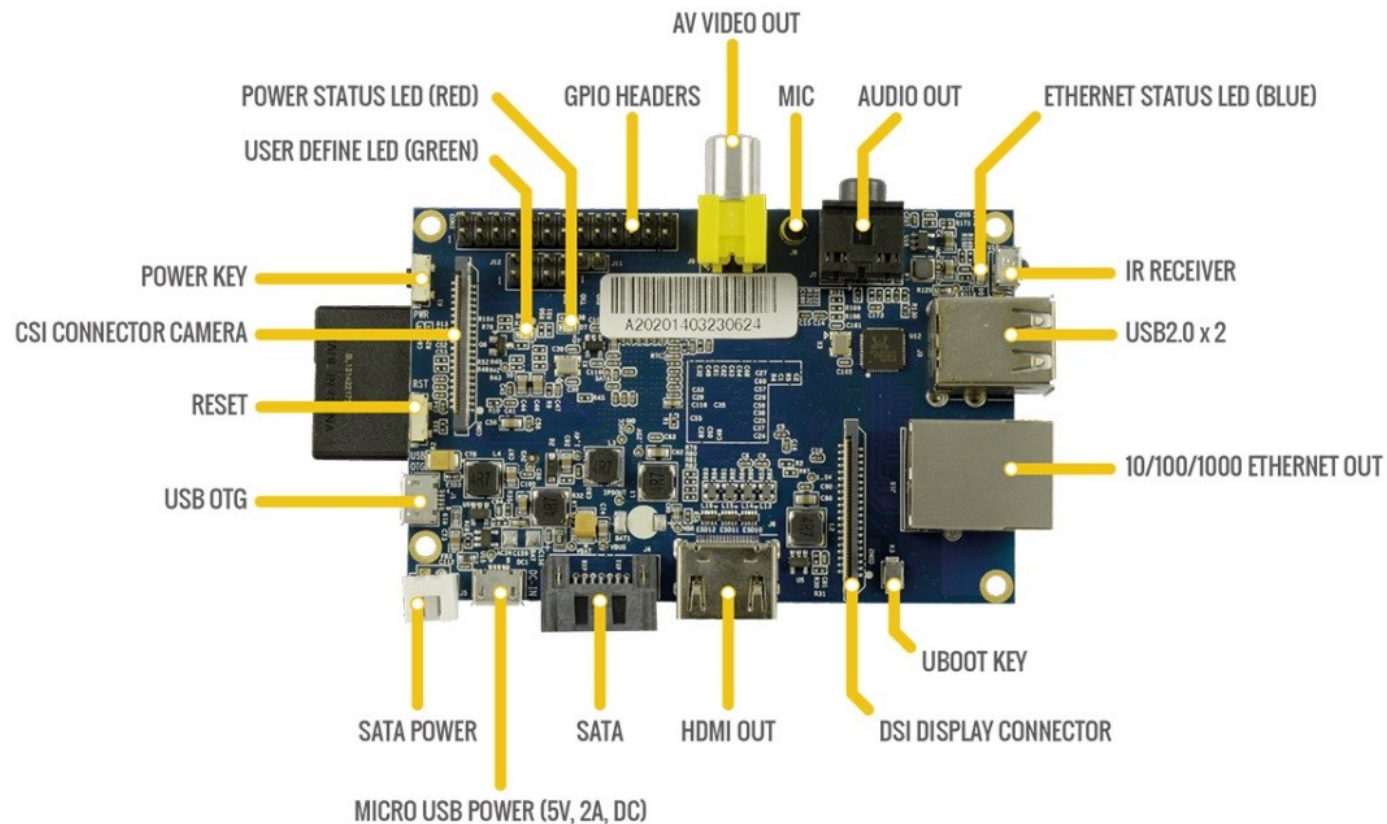


Problemi noti

- usb chipset unico: bus condiviso usbhub/eth
- latenza i/o causato da sdcard lente o incompatibili
- problemi con alcuni chipset wifi usb
- microsd si consumano in fretta:
diminuire al massimo la scrittura, o backuppare spesso.
- rpi3: surriscaldamento in caso di stress [dissipatori o ventoline]
- rpi3: in caso di device usb che consumano molta energia e non alimentati esternamente è consigliato usare un alimentatore 2.5A
- Per tutte le compatibilità hardware
http://elinux.org/RPi_Hub
- Troubleshooting http://elinux.org/R-Pi_Troubleshooting

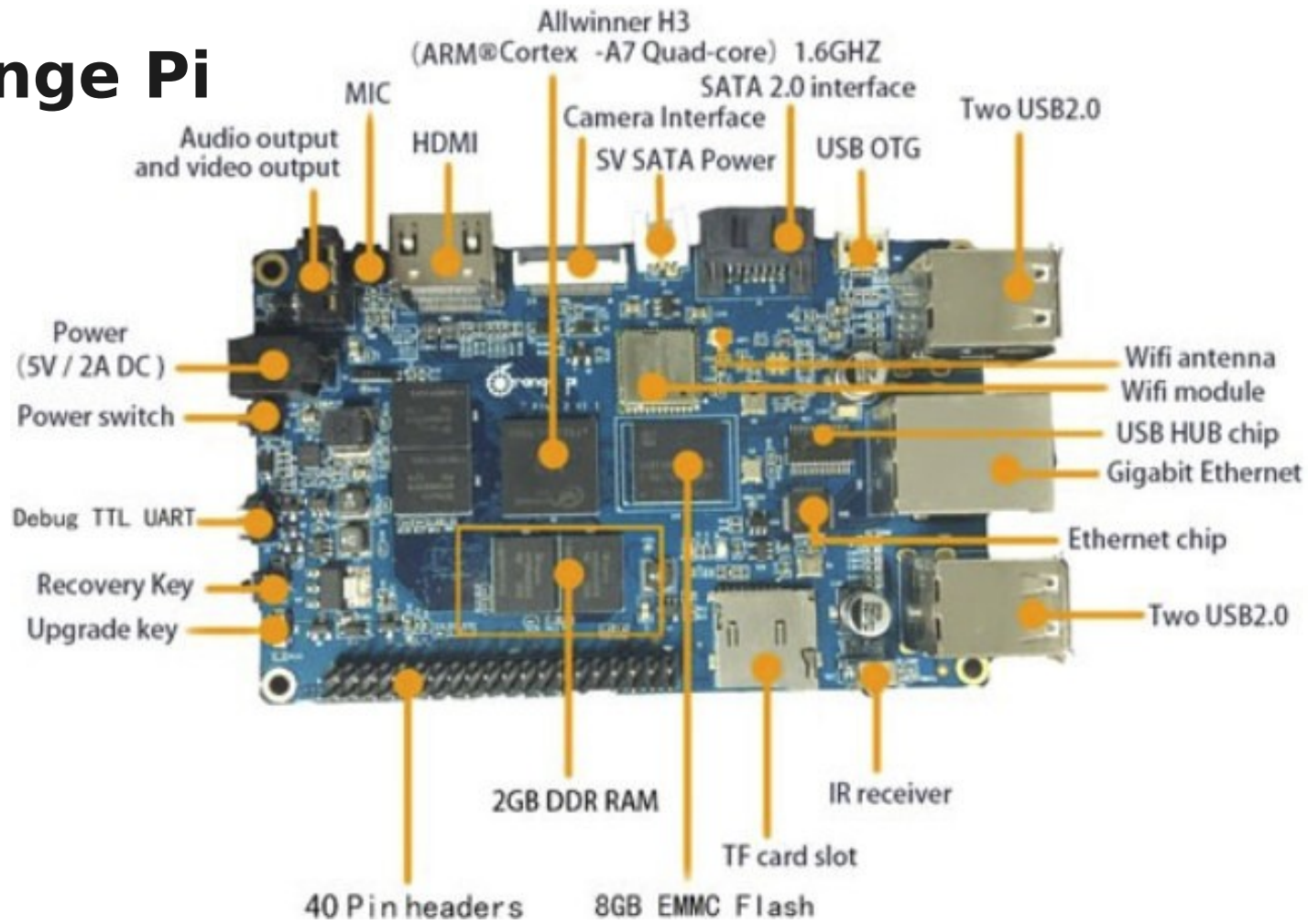
Board simili

Banana Pi



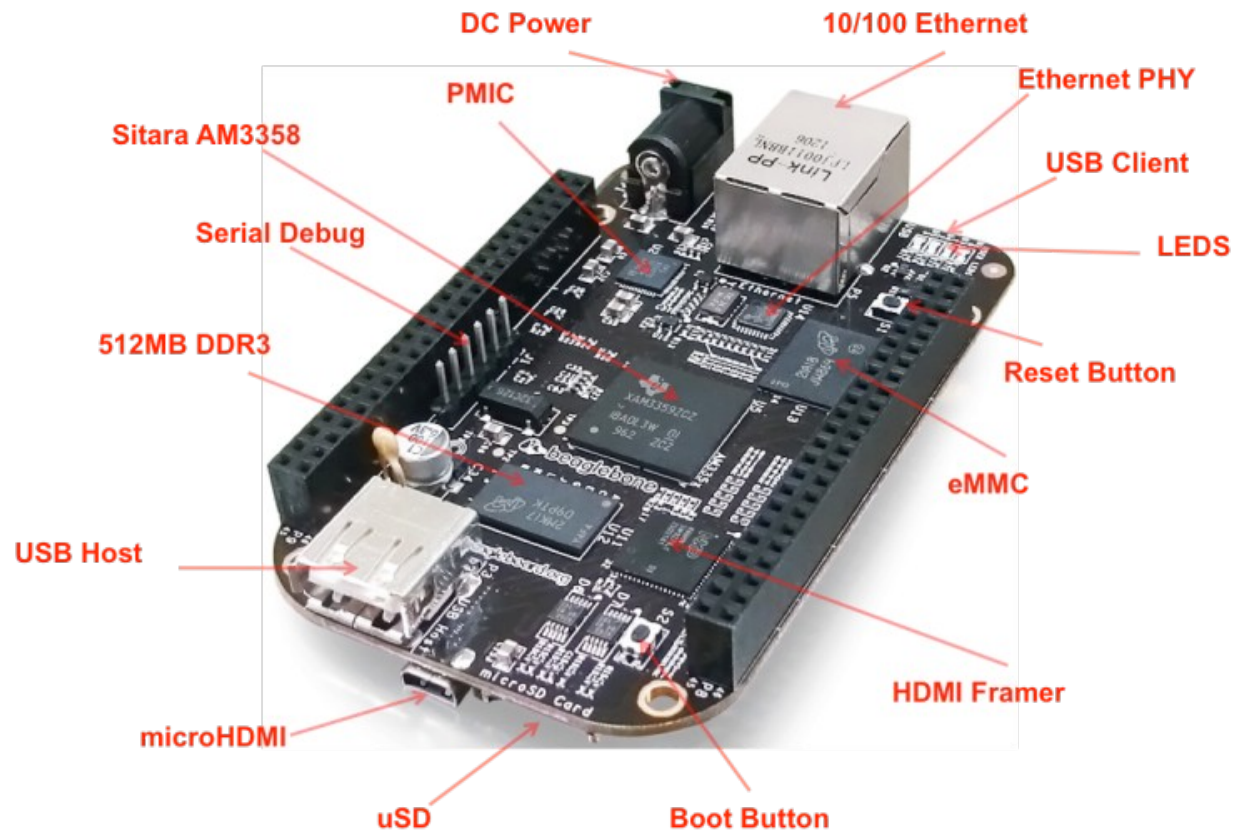
Board simili

Orange Pi



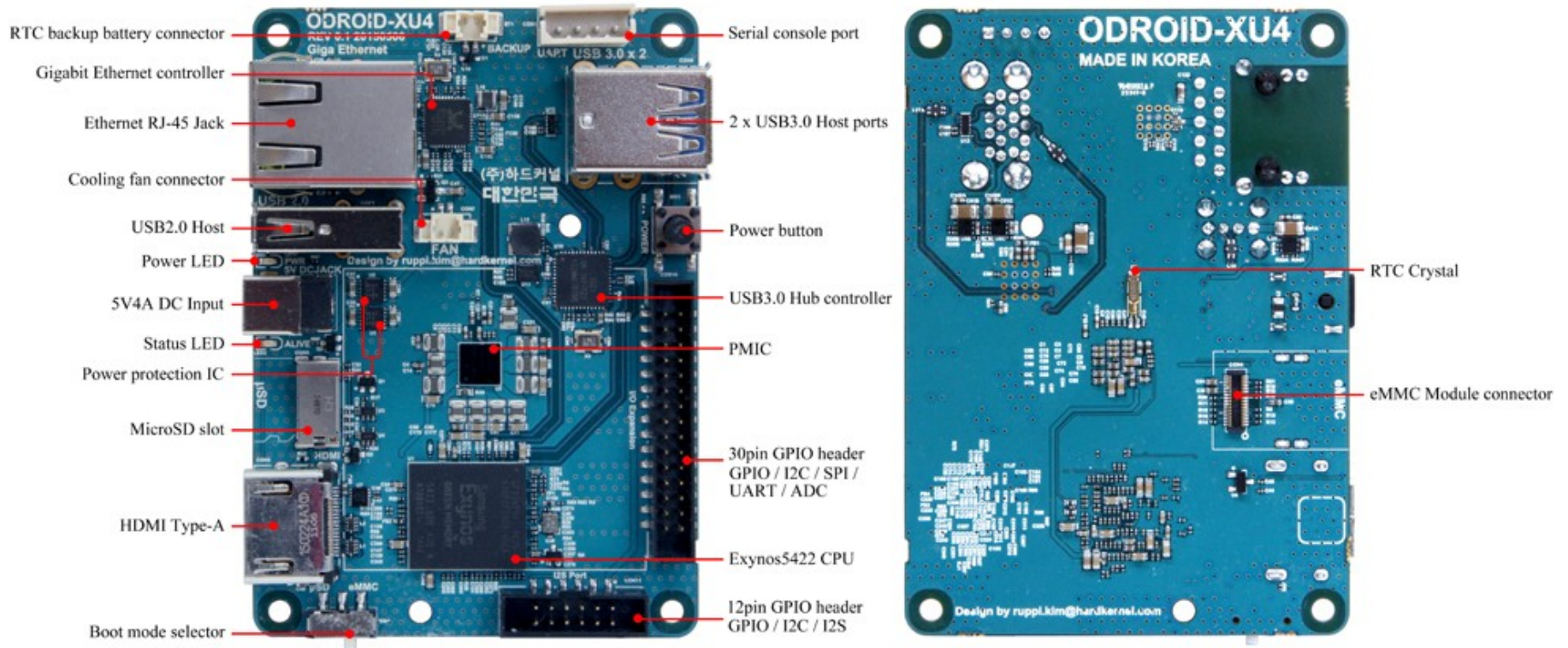
Board simili

Beaglebone Black



Board simili

Odroid XU4



Intro Linux [software]



- **Cosa è**
- **Come nasce**
- **Dove**
- **Quando**
- **Perché**

GNU/Linux



GNU/Linux – Cosa è

- **GNU** nasce nel 1983 come sistema operativo completamente **libero**
- Vuole costituire un'alternativa a Unix, sistema proprietario dominante all'epoca
- Più tardi, nel 1991, verrà integrato con il **kernel Linux** di Linus Torvalds



La filosofia OpenSource e la Free Software Foundation

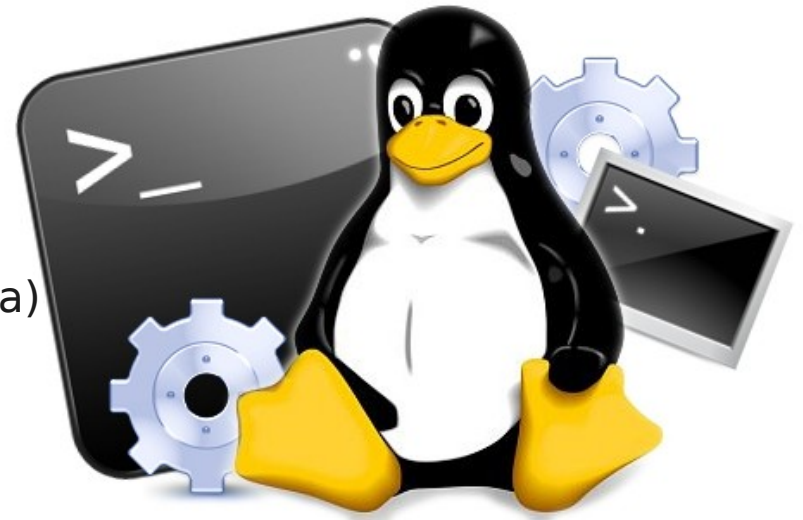
OpenSource != FREE

Open source è una pratica per produrre il software:

- Collaborativamente
- Aderente ai bisogni degli utenti
- Verso bisogni e usi innovativi
- Con più controllo sui sorgenti (bug e sicurezza)

Software libero nasce per liberare gli utenti:

- Collaborazione tra gli utenti
- Utente sopra il programma
- Indipendenza commerciale, sociale e politica



Le quattro libertà

Libertà 0: Libertà di eseguire il programma, per qualsiasi scopo

Libertà 1: Libertà di studiare come funziona il programma e di modificarlo in modo da adattarlo alle proprie necessità.

L'accesso al codice sorgente ne è un prerequisito.

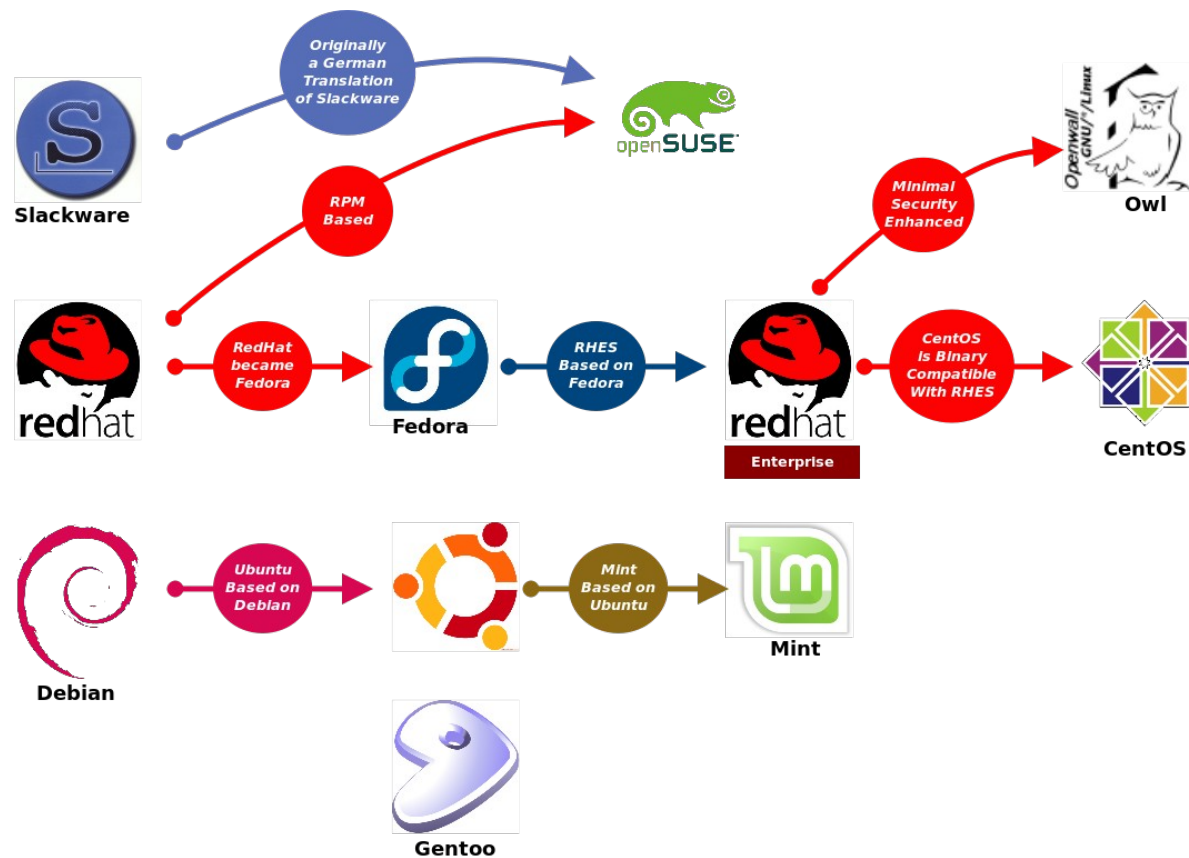
Libertà 2: Libertà di ridistribuire copie in modo da aiutare il prossimo.

Libertà 3: Libertà di migliorare il programma e distribuirne pubblicamente i miglioramenti da voi apportati (e le vostre versioni modificate in genere), in modo tale che tutta la comunità ne tragga beneficio.

L'accesso al codice sorgente ne è un prerequisito.

Distribuzioni Linux

Sistema completo (kernel, shell, librerie, GUI, Applicativi)



Distribuzioni

Release distro e Rolling distro



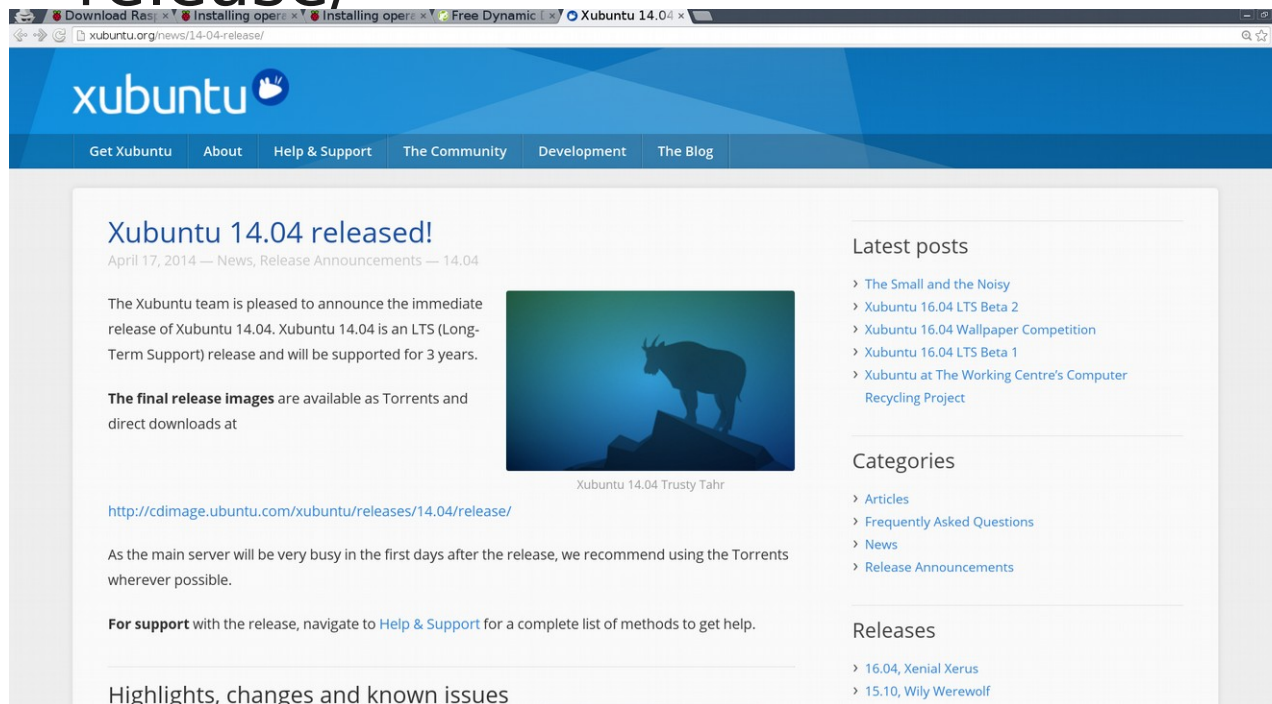
Utilizzo di Linux

- Installazione stand-alone
 - linux installato sull'intero HD del proprio pc
- Installazione collaterale
 - linux convive con altri OS
- Virtual Machine Linux
- Live CD
 - linux viene fatto partire in modalità live da CD
- Live USB
 - linux viene avviato da una USB key
- Su dispositivi embedded Linux (Es Pi)

Utilizzo di una Live Linux

Xubuntu 14.04 Live Linux on USB

<http://xubuntu.org/news/14-04-release/>



Creazione della USB key

USB bootable key

- Windows
 - con **Win32 DiskImager**
- Linux e MacOSX
 - con **dd**

*Vedremo come usare questi
tool più avanti nella
preparazione della SD card*

Utilità della Live Linux

Con la nostra Live Linux potremo effettuare:

- operazioni di backup
- installazione immagine sdcard
- mouting sdcard per editing a freddo

Queste operazioni possono essere eseguite anche da un sistema Linux non Live, da una Virtual Machine Linux o anche da sistemi Windows e Mac.

Ma possiamo usare una Live per avere un ambiente comodo e pronto all'uso.

Start with Raspbian



Preparazione della SD

- scarichiamo il file «immagine» del sistema operativo
 - ad esempio dal sito raspbrian.org
- questo file conterrà il «firmware» della nostra raspberry, cioè partizioni e file system da scrivere sulla SD card
- per scrivere il contenuto di un file immagine su una memoria si utilizzano appositi strumenti come **dd** in ambiente linux o **Win32 DiskImager** su Windows



Utilizzo di dd

- **dd** permette di copiare un file su un altro file
 - lo fa in maniera così versatile che è possibile anche utilizzarlo per partizioni e dischi interi, come fossero file
- Utilità di **dd** per le SD card
 - Copiare l'intera SD card in un file (es: per back-up)
 - Questo tipo di file viene detto file di immagine disco
 - Ripristinare immagini sulla SD card
 - da precedenti backup
 - da immagini di OS base
- **if** = input file **of** = outputfile **bs** = blocksize

Prepariamo la SD

- Individuiamo il nome del device corrispondente alla nostra SD
 - con dmesg o con df [se già montata]
 - es `/dev/mmcblk0` o `/dev/sdc`
- Smontiamo la SD [`umount`]
- Utilizziamo dd per flashare il file immagine sulla SD card

```
dd bs=4M if=raspbian-jessie-lite.img of=/dev/mmcblk0
```

** attenzione: dd non dà output e l'operazione potrebbe richiedere qualche minuto*

Prima connessione

- **Con console fisica**
 - Monitor (HDMI), tastiera (e mouse)
- **Tramite ethernet o wifi**
 - Configurando prima il sistema editando la SD card dal PC
 - Impostando la connessione ethernet o wifi
 - Accedendo alla raspberry via ssh
- **Login**
 - Utente: pi Password: raspberry

Connessione Remota

Accesso remoto via ssh possiamo usare il terminale su Linux/MacOsX oppure PuTTY se utilizziamo Windows

Download su www.putty.org

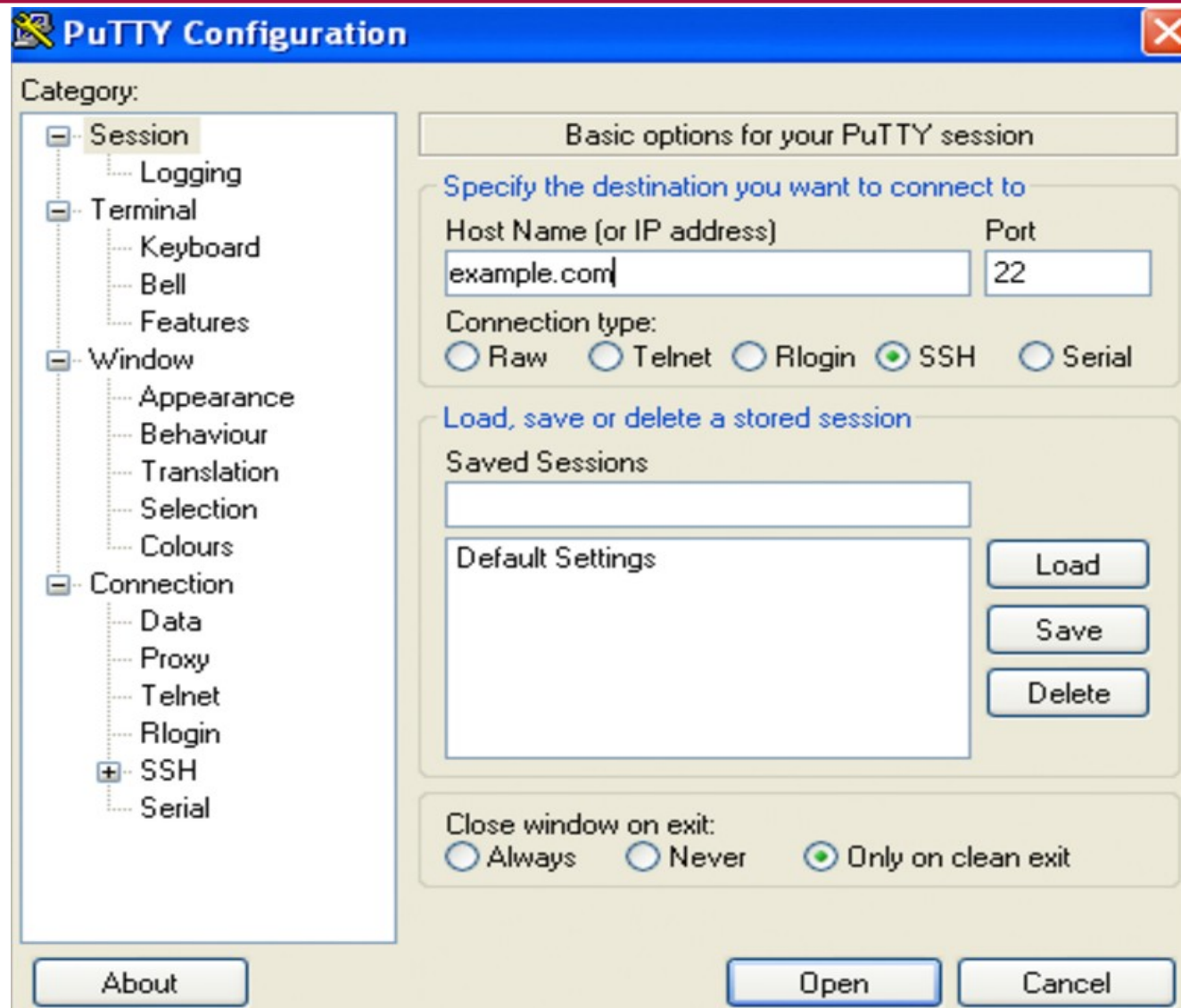
Non necessita installazione [doppio click e funziona]

Facile intuitivo veloce [vedi screenshot]

Modifica 'example.com' con l'indirizzo IP della tua raspberry owncloud

Click su open e YES per accettare la key

Connessione Remota



raspi-config

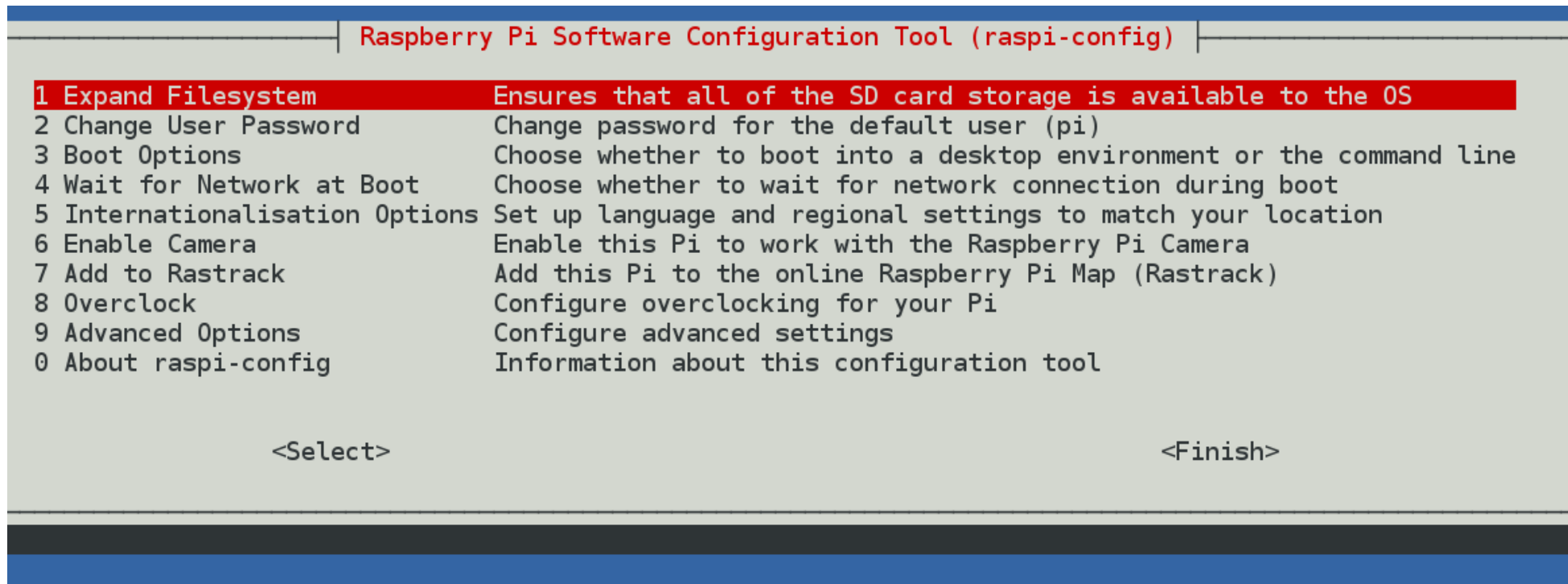
raspi-config è l'utilità di configurazione generale della raspberry pi

È un wizard che ci consente di configurare e riconfigurare rapidamente le impostazioni principali del sistema come la password, la rete, etc..

A terminal window with a blue title bar and a red close button. The terminal text shows a user prompt 'pi@raspberrypi:~' followed by the command '\$ sudo raspi-config' with a cursor at the end.

```
pi@raspberrypi:~ $ sudo raspi-config
```

raspi-config

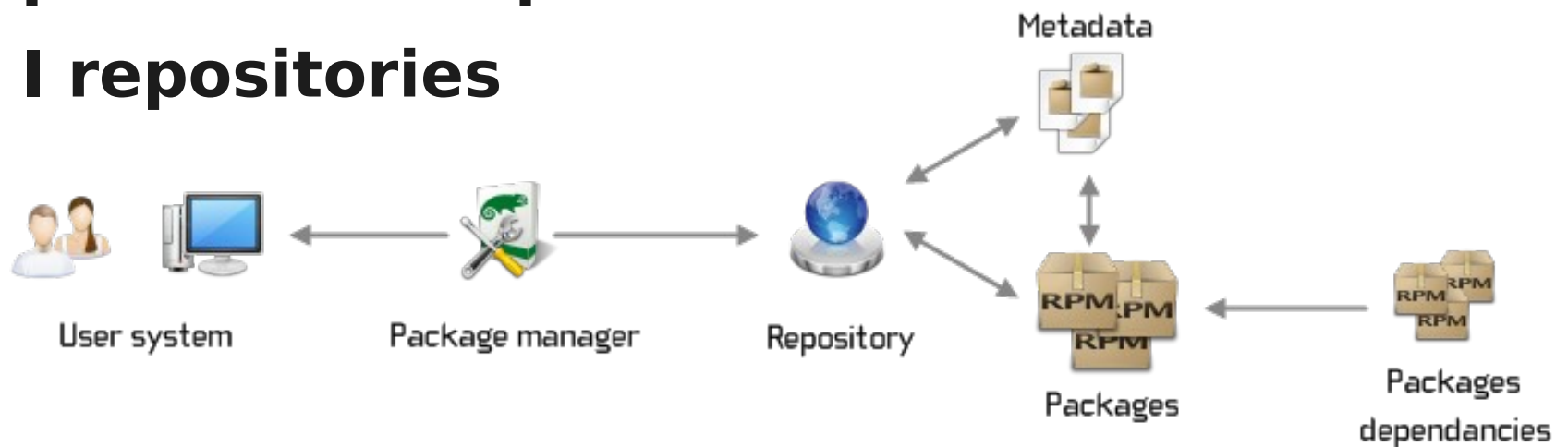


raspi-config

- **Expand Filesystem**
 - Espandiamo l'immagine del disco fino ad occupare tutta la SD
- **Change User/Password**
- **Internationalization**
 - Language, Keyboard layout, timezone
 - WiFi radio settings
- **Advanced**
 - Hostname
 - SSH

apt

- **pacchetti e dipendenze**
- **I repositories**



- **Operazioni su apt**
 - Cambio sources + key
 - Update
 - Upgrade



installare i primi tool

Modifichiamo il file `/etc/apt/source.list` e aggiungiamo la key del repository:

```
wget https://archive.raspbian.org/raspbian.public.key  
-O - | sudo apt-key add -
```

```
sudo -i
```

```
echo "deb http://archive.raspbian.org/raspbian jessie  
main contrib non-free" > /etc/apt/sources.list
```

```
echo "deb-src http://archive.raspbian.org/raspbian  
jessie main contrib non-free" >> /etc/apt/sources.list
```

```
apt-get update
```

```
apt-get upgrade [se vogliamo aggiornare il sistema,  
consigliato]
```

installare i primi tool

- iotop
 - controllo performance disco
- htop
 - controllo cpu/ram/processi
- sysstat
 - statistiche di sistema avanzate
- ifstat
 - statistiche di rete
- netstat / ss
 - controllo delle connessioni
- locate
 - localizza file

I primi comandi shell

- `cd` cambiare directory
- `cp` copiare files
- `mv` spostare files
- `rm` rimuovere file
- `ls` listare file/directory
- `df` utilizzo dischi
- `head` inizio di un file
- `tail` coda di un file
- `less` mostra un file grande con scroll
- `nano` editor di testo
- `wget` scarica un file dal web

ls

- a tutti i file
- l lista dettagliata
- h unità di memoria
- color

```
shum@sol:~$ ls -l
total 20
drwx----- 2 shum staff 4096 Jan 16 22:04 Mail
drwx----- 3 shum staff 4096 Jan 16 14:15 csc128
drwxr-xr-x  2 shum staff 4096 Jan 13 16:42 public
drwxr-xr-x  2 shum staff 4096 Jan 16 14:07 public_html
-rw-r--r--  1 shum staff 628 Jan 15 20:04 verse
```

file type

user permissions

group permissions

other (everyone) permissions

number of hard links

user (owner) name

group name

size

date/time last modified

filename

rwx

executable

writeable

readable

Utenti e root

Nei sistemi GNU/Linux esistono due tipi di utenti: gli utenti “normali” e l’utente **root**

L’utente **root** è l’amministratore che gestisce il sistema, ha poteri “infiniti” e viene usato unicamente con questo scopo (non è un utente con cui loggarsi in libertà);

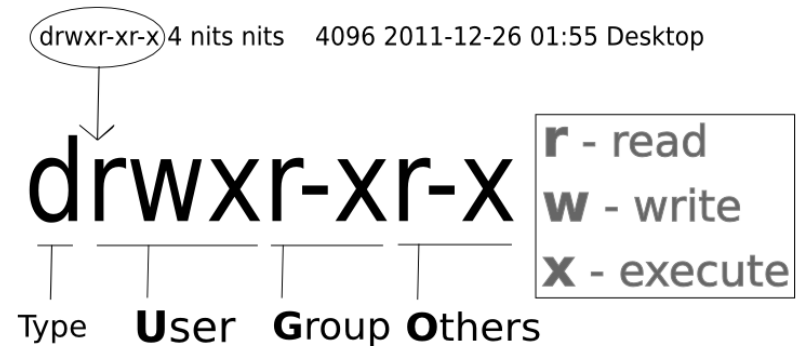
Gli **utenti** “normali” sono tutti gli altri, solitamente hanno permessi ristretti ma possono ricevere temporaneamente permessi di amministrazione;

L’amministratore può delegare alcuni permessi a determinati utenti utilizzando i **gruppi**.

Utenti e permessi

- **Utenti e Gruppi**
- **Il sistema di permessi**
- **Comandi utili**

- adduser
- userdel
- passwd
- sudo
- su
- chmod
- chown



File permissions in Linux



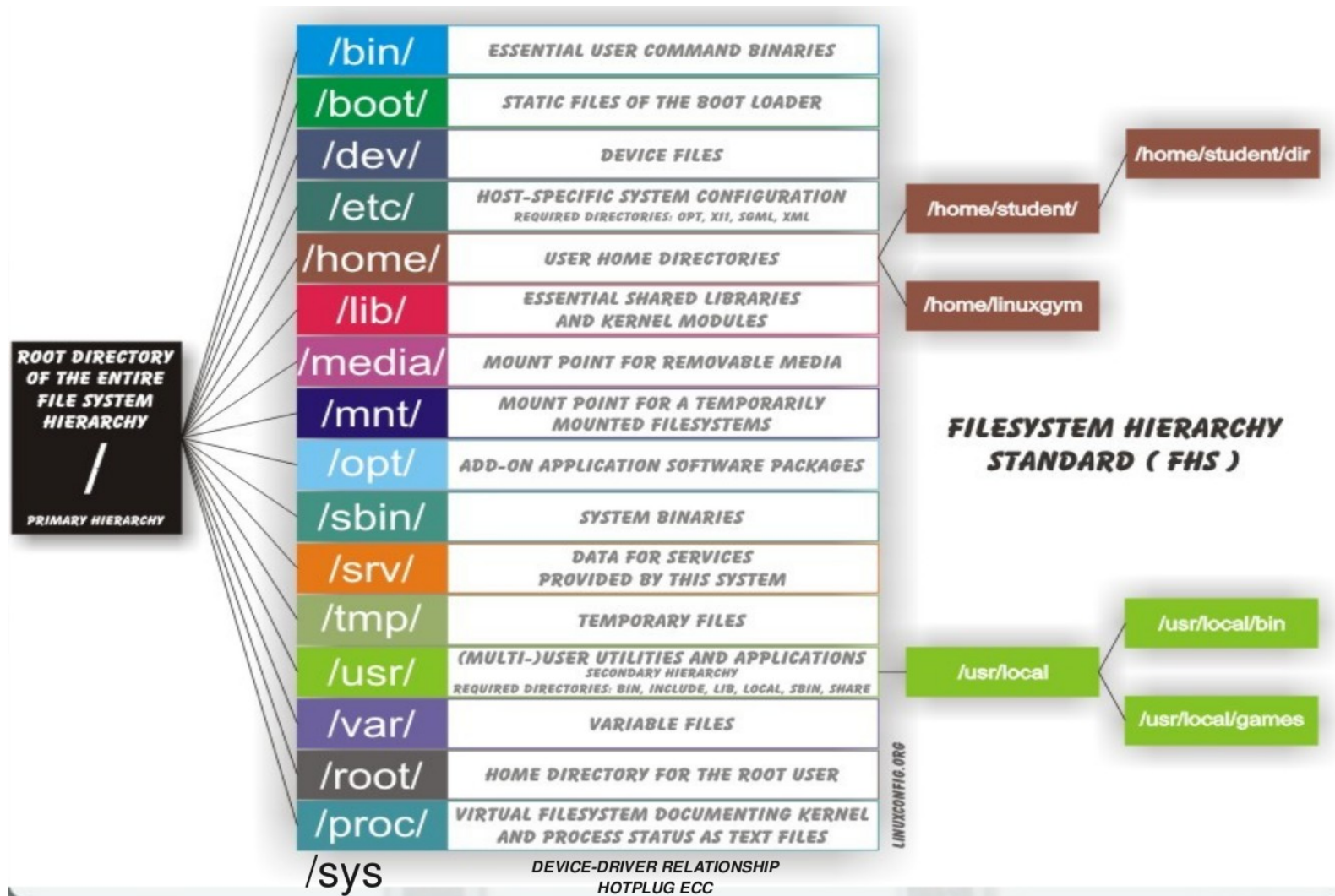
su e sudo

Gli utenti “normali” possono **acquisire permessi di root** usando i comandi:
su e **sudo**

Il comando **su** permette di loggarsi come un altro utente inserendo la sua password;

Il comando **sudo** permette di eseguire un comando con permessi di root, usando la password dell'utente o senza password [[/etc/sudoers](#)]

Linux File system Hierarchy



editor

Per editare file Linux ci offre innumerevoli possibilità: vim, emacs, ed, joe, nano sono solo alcuni degli editor possibili.

Per motivi di semplicità useremo nano che è user-friendly e lo si trova di default praticamente ovunque. Per editare un file ci basterà digitare

\$ nano nomefile.txt

Dopo aver modificato il file ci basterà premere CTRL+X e dopo Y per confermare e salvare il file.

Intro ownCloud



Architettura di ownCloud

- **ownCloud (Applicazione web)**
- PHP (Motore di scripting)
- MySQL (DataBase Server)
- Apache (Web Server)
- Exim4 (Mail Server)(non necessario, si può usare un MTA esterno tramite account user/password)
- Raspbian (Sistema Operativo)
- Raspberry (Hardware)

MTA

Mail Transport Agent

ci serve affinché la nostra Pi possa inviare email

un MTA è un **servizio** di rete,
normalmente in ascolto sulla porta 25
useremo un'altra porta la **465** per evitare
filtri sulla 25 che alcuni provider
utilizzano

Ci sono diversi MTA, noi useremo **exim4**

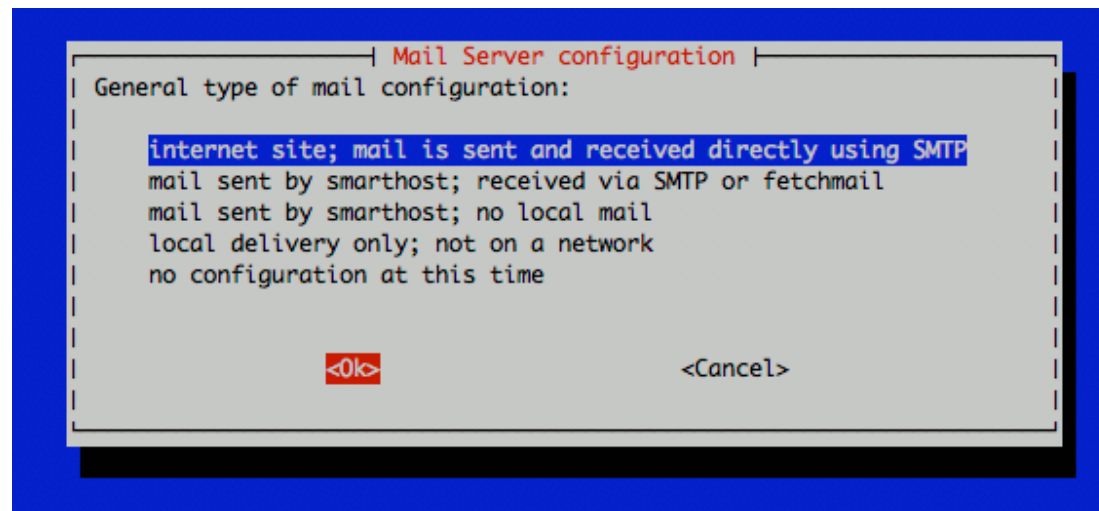
exim 4

Installiamo i pacchetti necessari

```
sudo apt-get install exim4-daemon-light exim4-config
```

Lanciamo la configurazione del pacchetto exim

```
sudo dpkg-reconfigure exim4-config
```



MAIL SERVER CONFIGURATION

General type of mail configuration:

internet site; mail is sent and received directly using SMTP

System mail name:

[generalmente l' FQDN]: hostname.local.net

IP-Addresses to listen on for incoming SMTP connections:

127.0.0.1 [remove ::1 for ipv6]

Other destinations for which mail is accepted:

hostname.local.net; hostname; localhost.localdomain; localhost

Domains to relay mail for:

Machines to relay mail for:

Delivery method for local mail: mbox format in /var/mail

Split configuration into small files? NO

Root and postmaster mail recipient: your@email.it [oppure root e fare un alias]

Keep number of DNS-queries minimal (Dial on Demand)? YES

exim4 ultimi steps

Configuriamo SMTPLISTENEROPTIONS in /etc/default/exim4

```
sed -i "s/SMTPLISTENEROPTIONS=/'SMTPLISTENEROPTIONS='-oX  
465:25 -oP \var/run/exim4/exim.pid'/" /etc/default/exim4
```

Generiamo il certificato

```
bash /usr/share/doc/exim4-base/examples/exim-gencert
```

Riavviamo exim

```
systemctl restart exim4
```

Testiamo se funziona

```
echo "testmail" | mail -s "il_subject_della_mail" latua@email.com
```

[probabilmente la troveremo nella cartella SPAM della nostra mail]

Installazione di MySQL

- Installiamo l'ultima versione di mysql server e client facendo il preseeding del pacchetto per impostare la password root di mysql [owncloudmysql] ed evitare che ci venga richiesto durante l'installazione
- ```
mysql preseeding
debconf-set-selections <<< "mysql-server mysql-server/root_password password owncloudmysql"
debconf-set-selections <<< "mysql-server mysql-server/root_password_again password owncloudmysql"
```
- `apt-get install -q -y mysql-server mysql-client`
- `mysql -u root -h localhost -p`  

```
CREATE DATABASE owncloud;
CREATE USER owncloud@localhost IDENTIFIED BY 'owncloud';
GRANT ALL PRIVILEGES ON owncloud.* TO owncloud@localhost;
FLUSH PRIVILEGES;
exit
```

# Installazione di Apache, PHP e accessori

- `apt-get install -y apache2 libapache2-mod-php5 php5-gd php5-json php5-mysql php5-curl php5-intl php5-mcrypt php5-imagick`
- Editiamo la config di php  
`/etc/php5/apache2/php.ini`
  - `; http://php.net/upload-max-filesize`
  - `upload_max_filesize = 2M      500M`  
`; Maximum number of files that can be uploaded via a single request`
  - `max_file_uploads = 20          200`  
`; http://php.net/post-max-size`
  - `post_max_size = 8M              500M`



# Configurazione di Apache

- disabilitiamo il sito di default e inseriamo
  - a2dissite 000-default.conf
- copiamo le configurazioni di apache e dei nuovi siti (http e https)
  - mpm\_prefork.conf    in /etc/apache2/mods-available/
  - owncloud.conf        in /etc/apache2/sites-available/
  - owncloud-ssl.conf    in /etc/apache2/sites-available/
- generiamo il certificato ssl
  - openssl req -x509 -sha512 -nodes -days 3650 -newkey rsa:2048 -keyout /etc/apache2/cert.key -out /etc/apache2/cert.crt -subj "/C=IT/ST=Italy/L=Palermo/O=Pa/CN=picloud.local.net"
  - openssl dhparam -out /etc/apache2/dhparams.pem 2048
- Abilitiamo i nuovi siti e riavviamo apache
  - a2ensite owncloud.conf
  - a2ensite owncloud-ssl.conf
  - systemctl restart apache2.service

# Installazione e configurazione di ownCloud

- Useremo l'installazione da pacchetto
    - Scarichiamo e aggiungiamo la chiave del repository ownCloud
- ```
wget -O- http://download.opensuse.org/repositories/isv:ownCloud:community/Debian_8.0/Release.key | apt-key add -
```
- Aggiungiamo il repository alle sources di apt
 - `echo 'deb http://download.opensuse.org/repositories/isv:/ownCloud:/community/Debian_8.0/ /' >> /etc/apt/sources.list.d/owncloud.list`
 - Aggiorniamo i repository e installiamo il pacchetto
 - `apt-get update`
 - `apt-get install owncloud-server`

Configurazione originale di ownCloud

```
<?php
$CONFIG = array (
    'instanceid' => 'ocn36dw46ex7',
    'passwordsalt' => '5pTGwCUDEftzR8itx4tlGyLpZlAol0',
    'secret' => '7IdwUSDxoftCqdmpRp3r7fVm0MwUgIgc3w0RVGlu8SED0JZu',
    'trusted_domains' =>
    array (
        0 => '192.168.23.171',
    ),
    'datadirectory' => '/var/www/owncloud/data',
    'overwrite.cli.url' => 'https://192.168.23.171',
    'dbtype' => 'mysql',
    'version' => '8.1.6.2',
    'dbname' => 'owncloud',
    'dbhost' => 'localhost',
    'dbtableprefix' => 'oc_',
    'dbuser' => 'owncloud',
    'dbpassword' => 'owncloud',
    'logtimezone' => 'UTC',
    'installed' => true,
);
```

Configurazione ottimizzata di ownCloud

```
<?php
$CONFIG = array (
    'instanceid' => 'ocn36dw46ex7',
    'passwordsalt' => '5pTGwCUDEftzR8itx4tLGyLpZlAo10',
    'secret' => '7IdwUSDxoftCqdmprp3r7fVm0MwUgIgc3w0RVGLU8SED0JZu',
    'trusted_domains' =>
        array (
            0 => '192.168.23.171',
            1 => '192.168.1.1',
            2 => '172.24.1.1',
            3 => 'localhost',
            4 => 'picloud.local.net',
        ),
    'datadirectory' => '/var/www/owncloud/data',
    'overwrite.cli.url' => 'https://172.24.1.1',
    'dbtype' => 'mysql',
    'version' => '8.1.6.2',
    'dbname' => 'owncloud',
    'dbhost' => 'localhost',
    'dbtableprefix' => 'oc_',
    'dbuser' => 'owncloud',
    'dbpassword' => 'owncloud',
    'logtimezone' => 'UTC',
    'installed' => true,
    'loglevel' => 0,
    'enabledPreviewProviders' =>
        array (
            0 => 'OC\\Preview\\PNG',
            1 => 'OC\\Preview\\JPEG',
            2 => 'OC\\Preview\\GIF',
            3 => 'OC\\Preview\\BMP',
            4 => 'OC\\Preview\\XBitmap',
            5 => 'OC\\Preview\\MP3',
            6 => 'OC\\Preview\\TXT',
            7 => 'OC\\Preview\\MarkDown',
        ),
);
```

Configurazione ottimizzata di ownCloud

- Nel caso in cui volessimo bypassare il controllo sul `trusted_domain` effettuato da owncloud basta rimuovere completamente il campo nel file di configurazione per accettare qualsiasi ip/hostname. Come ogni cosa comoda è insicura quindi è poco consigliato, ma nel caso in cui vogliate disabilitare questo check adesso sapete come farlo.

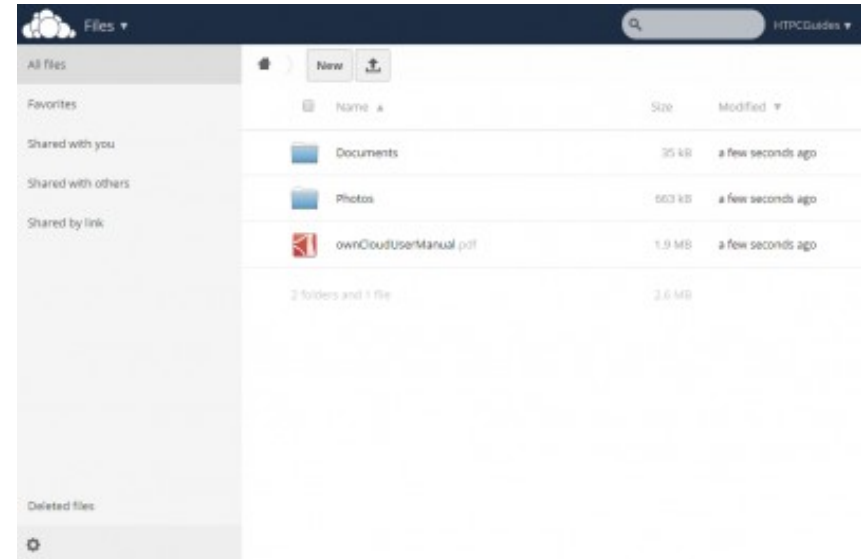
Setup Web ownCloud

- Username
- Password
- Data folder
- DB user
- DB pass
- DB name
- DB host

The image displays two sequential steps of the ownCloud web installation process. The left panel, titled 'Create an admin account', features the ownCloud logo and prompts for an admin username (set to 'admin') and a password (set to 'owncloudadmin'). It also shows the 'Data folder' path as '/var/www/owncloud/data'. The right panel, titled 'Configure the database', shows the 'MySQL/MariaDB' tab selected, with fields for database name ('owncloud'), username ('owncloud'), password ('owncloud'), and host ('localhost'). A 'Finish setup' button is visible at the bottom of the right panel.

Start with ownCloud

- File
- Upload
- admin cloud
- utente personale
- settings generali
- i primi fix



ownCloud Advanced



External storage

- Aggiungere HD esterni
- Chiavette USB

External Storage

Note: "smbclient" is not installed. Mounting of *SMB / CIFS*, *SMB / CIFS using OC login* is not possible. Please ask your system administrator to install it.

Folder name	External storage	Authentication	Configuration
<input type="text" value="Folder name"/>	<div>Add storage ▼</div>		

Federated Cloud

Your Federated Cloud ID: **firegarden@owncloud.viral.vnet**

Share it:



Plugins

- Plugins + Funzionalità avanzate
- Crittografia e altro.

Cifratura lato server *i*

☐ Abilita cifratura lato server

Server di posta *i*

Viene utilizzato per inviare le notifiche.

Modalità di invio **smtp** Cifratura **Nessuno**

Indirizzo mittente @

Metodo di autenticazione **Nessuno** ☐ Autenticazione richiesta

Indirizzo del server :

Prova impostazioni email

Backups

- Possiamo effettuare due tipi di backup
 - Backup hot
 - da dentro la raspberry mentre il sistema è in running
 - Backup cold
 - spegnendo la raspberry e staccando la microsd
 - lavorando sulla microsd attaccata al nostro PC

Backup Hot

Il backup del sistema owncloud si divide in due componenti fondamentali:

- il **sistema**
che comprende sia i file dell'applicazione web che i file utente caricati
- il **database**
che contiene le informazioni sugli utenti e sui file archiviati

Backup hot - sistema

Per copiare il sistema in una cartella nello stesso filesystem possiamo usare rsync e col potere della bash andremo ad aggiungere la data alla cartella creata

```
rsync -Aax /var/www/owncloud/ owncloud-dirbkp_`date +"%Y  
%m%d"` /
```

Dopodiché potremmo comprimere la cartella usando tar

```
tar -cvpazf owncloud_backup_data.tar.gz owncloud-dirbkp-DATE
```

Rsync possiamo usarlo anche per copiare su un device remoto o su un NAS, ovviamente avremo bisogno dell'autenticazione

```
rsync -Aax owncloud/ user@host:owncloud-dirbkp_`date +"%Y  
%m%d"` /
```

Backup Hot - database

Poi tocca al database mysql

```
mysqldump --lock-tables -h localhost -u owncloud  
-powncloud owncloud > owncloud-sqlbkp_`date +"%Y  
%m%d"`.sql
```

Anche questo potremmo comprimerlo, qui ci basterà gzip

```
gzip owncloud-sqlbkp_`DATE`.sql
```

Sposteremo anche questo file su un device esterno per una maggiore sicurezza del backup [più copie su diversi tipi di supporti è consigliato]

Backup Hot – altre considerazioni

Possiamo automatizzare tutti questi passaggi creando un piccolo script e, dopo essere sicuri che funzioni, aggiungerlo tra i cronjob della raspberry per un backup automatizzato [come compito a casa!!!]

Owncloud offre anche strumenti interni per il backup, vedremo velocemente qualche app

Backup - Cold

Per effettuare un backup cold di basterà inserire la SDcard nel nostro pc ed effettuare una copia dell'intera memoria con dd

```
dd bs=4M if=/dev/mmcblk0 of=pi-backup.img
```

Device esterni

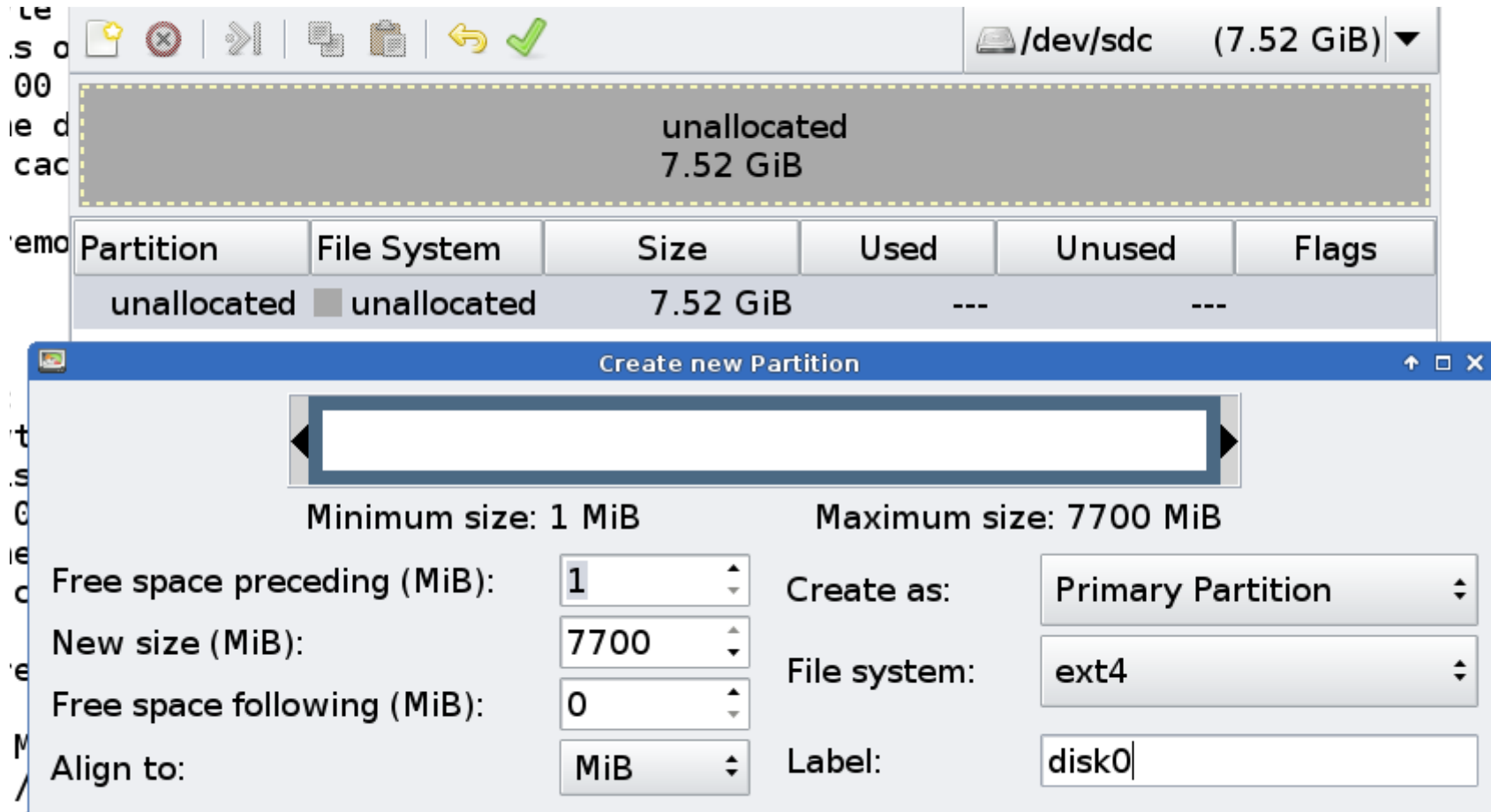
Per aggiungere un disco o una usbkey ad owncloud dobbiamo prima formattare il device e dopo farlo riconoscere ad owncloud

Possiamo aiutarci con uno strumento di linux presente anche nella live cioè gparted

In caso contrario è possibile installarlo tramite

```
sudo apt-get install gparted
```

Device esterni



Device esterni

Collegiamo il nostro device alla raspberry e

- `sudo mkdir /media/disk0`
- `sudo mount /dev/sda1 /media/disk0`

Identifichiamo lo uuid del device con

- `ls -l /dev/disk/by-uuid/ | grep sda`
- `lsblk | grep sda`

Identificato lo uuid device aggiungiamo al nostro `/etc/fstab` per il mount automatico al riavvio la seguente:

```
UUID=ecf81575-1261-4397-8047-cbc47901c541 /media/disk0 ext4
defaults,relatime 0 0
```

Sostituendo lo UUID con quello ritornato dai comandi precedenti

Device esterni

Sistemiamo i permessi della cartella esterna per permettere ad owncloud la scrittura

- `sudo mkdir /media/disk0/owncloud`
- `sudo chown www-data:www-data /media/disk0/`


A questo punto la nostra cartella è pronta per essere aggiunta come local storage [l'utente owncloud per usufruirne dovrà appartenere al gruppo admin]

Aggiungiamola tramite l'area admin owncloud

Device esterni

Archiviazione esterna

Nota: "smbclient" non è installato. Impossibile montare *SMB / CIFS* e *SMB / CIFS* utilizzando le credenziali di OC. Chiedi al tuo amministratore di sistema di installarlo.

	Nome della cartella	Archiviazione esterna	Configurazione	Disponibile per
	Local	Locale	/media/disk0/owncloud	Tutti gli utenti. Digita per selezionare utenti

Nome della cartella

Aggiungi archiviazione ▼

Abilita le anteprime ☒

Controlla le modifiche

Una volta per ogni accesso diretto ▼

☒ Abilita la memoria esterna dell'utente
Consenti agli utenti di montare la seguente memoria esterna

- ☐ Amazon S3 e conformi
- ☐ Dropbox
- ☐ FTP
- ☐ Google Drive
- ☐ OpenStack Object Storage
- ☐ ownCloud
- ☐ SFTP
- ☐ SFTP con accesso a chiave segreta
- ☐ WebDAV

Raspberry Pi e networking



Ethernet

- Connessione ethernet
 - Statica per connettersi con cavo cross
 - DHCP per connetterla alla LAN esistente

Client Wireless

Connessione alla WiFi domestica

`iwlist`

`wpa_supplicant`

Wireless AP

- **Access point**
 - Installazione e configurazione **hostapd**
 - hostapd è un pacchetto che ci permette di fare da Access Point
 - Installazione e configurazione **dnsmasq**
 - è un server DHCP e DNS integrato semplice da configurare
- **Installiamo i pacchetti**
 - `sudo apt-get install dnsmasq hostapd`

Configuriamo l'interfaccia

- Configuriamo l'interfaccia wlan0 per avere un indirizzo statico
 - `sudo nano /etc/dhcpd.conf`
 - `interface wlan0 static ip_address=172.24.1.1/24`
- Evitiamo che wpa supplicant gestisca l'interfaccia commentando l'inclusione del file di configurazione wpa
 - `sudo nano /etc/network/interfaces`
 - `iface wlan0 inet manual`
`# wpa-conf /etc/wpa_supplicant/wpa_supplicant.conf`
 - `sudo systemctl restart dhcpd.service`

Configuriamo hostapd

```
sudo nano /etc/hostapd/hostapd.conf
```

```
interface=wlan0  
driver=nl80211  
ssid=Pi3-Cloud  
hw_mode=g  
channel=1  
auth_algs=1  
ignore_broadcast_ssid=0  
wpa=2  
wpa_key_mgmt=WPA-PSK  
wpa_passphrase=raspberrypi  
rsn_pairwise=CCMP
```

```
sudo nano /etc/default/hostapd
```

```
DAEMON_CONF="/etc/hostapd/hostapd.conf"
```

Configuriamo dnsmasq

```
sudo cp /etc/dnsmasq.conf /etc/dnsmasq.conf.bkp
```

```
sudo nano /etc/dnsmasq.conf [ed aggiungiamo alla fine del file]
```

```
interface=wlan0      # Use interface wlan0
bind-interfaces      # Bind to the interface
server=8.8.8.8       # Forward DNS requests to Google DNS
domain-needed        # Don't forward short names
bogus-priv           # Never forward addresses in the non-routed spaces.
```

```
dhcp-range=172.24.1.50,172.24.1.200,48h # Assign IP addresses
```



ip forward [per instradare i pacchetti internet da ethernet a wifi]

```
sudo sh -c "echo 1 > /proc/sys/net/ipv4/ip_forward"
```

```
sudo iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
```

```
sudo iptables -A FORWARD -i eth0 -o wlan0 -m state --state RELATED,ESTABLISHED -j ACCEPT
```

```
sudo iptables -A FORWARD -i wlan0 -o eth0 -j ACCEPT
```

```
sudo sh -c "iptables-save > /etc/iptables.ipv4.nat"
```

```
sudo nano /lib/dhcpd/dhcpd-hooks/70-ipv4-nat
```

 with the contents:

```
sudo iptables-restore < /etc/iptables.ipv4.nat
```

```
sudo systemctl start hostapd
```

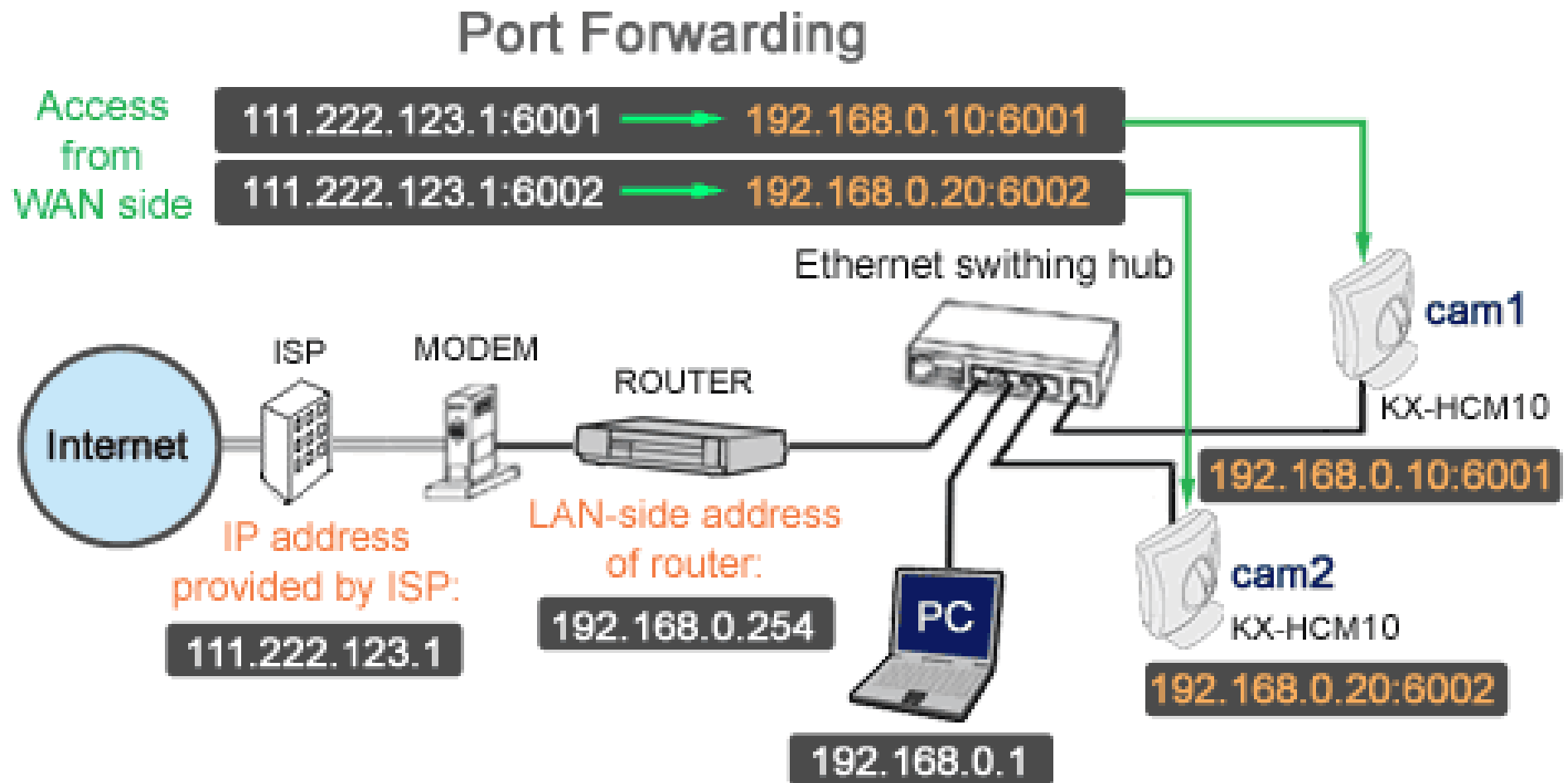
```
sudo systemctl start dnsmasq
```



Accessibilità da internet

- Configurazione del proprio router
 - Port forwarding
- IP, DNS e Dynamic DNS
 - Noip
 - Iscrizione
 - Setup
 - Configurazione

Port Forwarding



Impostare una NAT sul proprio router

- Nel pannello di amministrazione del proprio router/AP dovrebbe trovarsi una voce NAT o Virtual Server da configurare con la porta del servizio ownCloud

Device Info

Quick Setup

Advanced Set

+ Layer2 Inter

+ WAN Service

+ MAC Clone

+ LAN

- NAT

+ Virtual Ser

+ Port Trigg

+ DMZ Host

+ ALG

+ Security

+ Parental Co

+ Quality of S

+ Bandwidth C

+ Routing

+ DNS

NAT -- Virtual Servers

Select the service name, and enter the server IP address and click "Save/Apply" to forward IP packets for this service to the specified server.

NOTE: The "Internal Port End" cannot be modified directly. Normally, it is set to the same value as "External Port End". However, if you modify "Internal Port Start", then "Internal Port End" will be set to the same value as "Internal Port Start".

Remaining number of entries that can be configured:32

Use Interface:

Service Name:

☐ Select a Service:

☒ Custom Service:

Server IP Address:

External Port Start	External Port End	Protocol	Internal Port Start	Internal Port End
55000	55000	TCP	80	80
		TCP		
		TCP		

Impostare una NAT sul proprio router

Importante !!!

La stessa configurazione fatta per la porta 80 va effettuata per la porta 443 in quanto stiamo usando owncloud con un certificato SSL per una connessione cifrata con il server.

Se non apriamo anche la porta 443 il nostro browser non riuscirà a raggiungere l'owncloud.

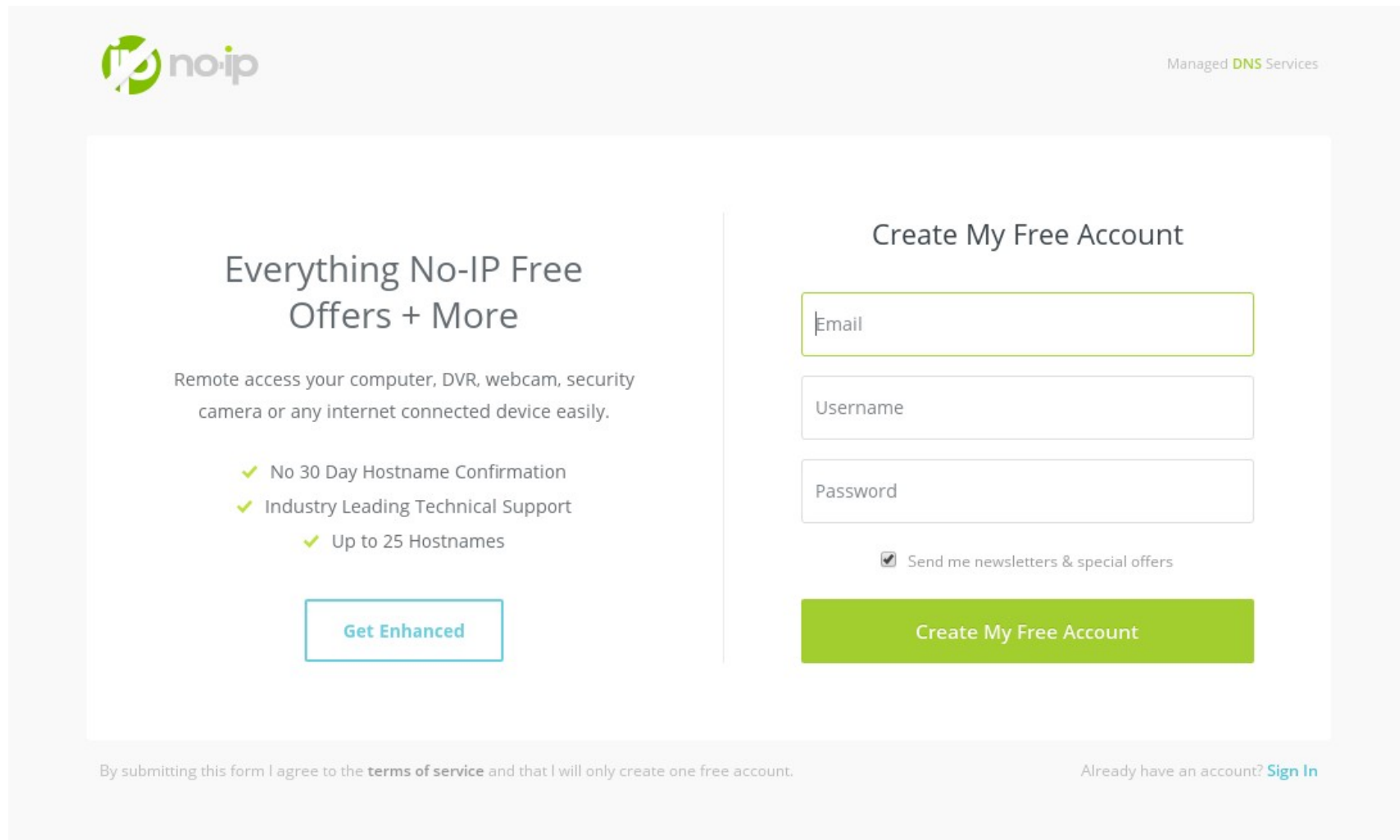
Impostare una NAT sul proprio router

Importante !!!

La stessa configurazione fatta per la porta 80 va effettuata per la porta 443 in quanto stiamo usando owncloud con un certificato SSL per una connessione cifrata con il server.

Se non apriamo anche la porta 443 il nostro browser non riuscirà a raggiungere l'owncloud.

Registrazione a noip.com



The screenshot shows the noip.com website's registration page. The header features the noip logo on the left and 'Managed DNS Services' on the right. The main content is divided into two columns. The left column, titled 'Everything No-IP Free Offers + More', describes the benefits of a free account, including remote access to devices and a list of features: 'No 30 Day Hostname Confirmation', 'Industry Leading Technical Support', and 'Up to 25 Hostnames'. A 'Get Enhanced' button is at the bottom of this section. The right column, titled 'Create My Free Account', contains a registration form with fields for 'Email', 'Username', and 'Password'. Below these fields is a checkbox for 'Send me newsletters & special offers' and a large green 'Create My Free Account' button. At the bottom of the page, there is a disclaimer about terms of service and a link to 'Sign In' for existing users.

noip

Managed DNS Services

Everything No-IP Free Offers + More

Remote access your computer, DVR, webcam, security camera or any internet connected device easily.

- ✓ No 30 Day Hostname Confirmation
- ✓ Industry Leading Technical Support
- ✓ Up to 25 Hostnames

[Get Enhanced](#)

Create My Free Account

Email

Username

Password

☒ Send me newsletters & special offers

[Create My Free Account](#)

By submitting this form I agree to the [terms of service](#) and that I will only create one free account.

Already have an account? [Sign In](#)

Aggiungere un host su noip.com

Hosts/Redirects

Add Host

Manage Hosts

Manage Groups

Download Client

Upgrade to Enhanced

Need Help?

Support Center

Troubleshooting Guide

Dynamic Update Client

Support Ticket

Contact Us

Upgrade to Priority Support

Add a host

Fill out the following fields to configure your host. After you are done click 'Create Host' to add your host.

Own a domain name?

Use your own domain name with our DNS system. [Add](#) or [Register](#) your domain name now or read more for pricing and features.

Hostname Information

Hostname:

ddns.net

Host Type:

☒ DNS Host (A) ☐ DNS Host (Round Robin) ☐ DNS Alias (CNAME)

☐ Port 80 Redirect ☐ Web Redirect ☐ AAAA (IPv6)

IP Address:

Assign to Group:

- No Group -

 [Configure Groups](#)

Enable Wildcard:

Wildcards are a Plus / Enhanced feature. [Upgrade Now!](#)

Client noip

- **download client**

- wget
<https://www.noip.com/client/linux/noip-duc-linux.tar.gz>

- **setup client**

- tar xvzf noip-duc-linux.tar.gz
- make && sudo make install
- Inseriamo l'interfaccia di uscita [eth0] e le credenziali per il servizio noip
- sudo noip2
- Crea il cronjob per l'update automatico!

Client noip

Utilizzo di noip client

Usual operation?

<code>/usr/local/bin/noip2 -C</code>	configura un client
<code>/usr/local/bin/noip2</code>	per far partire il client
<code>/usr/local/bin/noip2 -S</code>	info riguardo il client in esecuzione
<code>/usr/local/bin/noip2 -D pid</code>	toggle lo stato debug per il client pid
<code>/usr/local/bin/noip2 -K pid</code>	termina client pid
Hai altri device oltre a questo per la connessione internet?	
<code>/usr/local/bin/noip2 -M -c file</code>	start additional instances

Test noip

ping test sul nuovo dominio

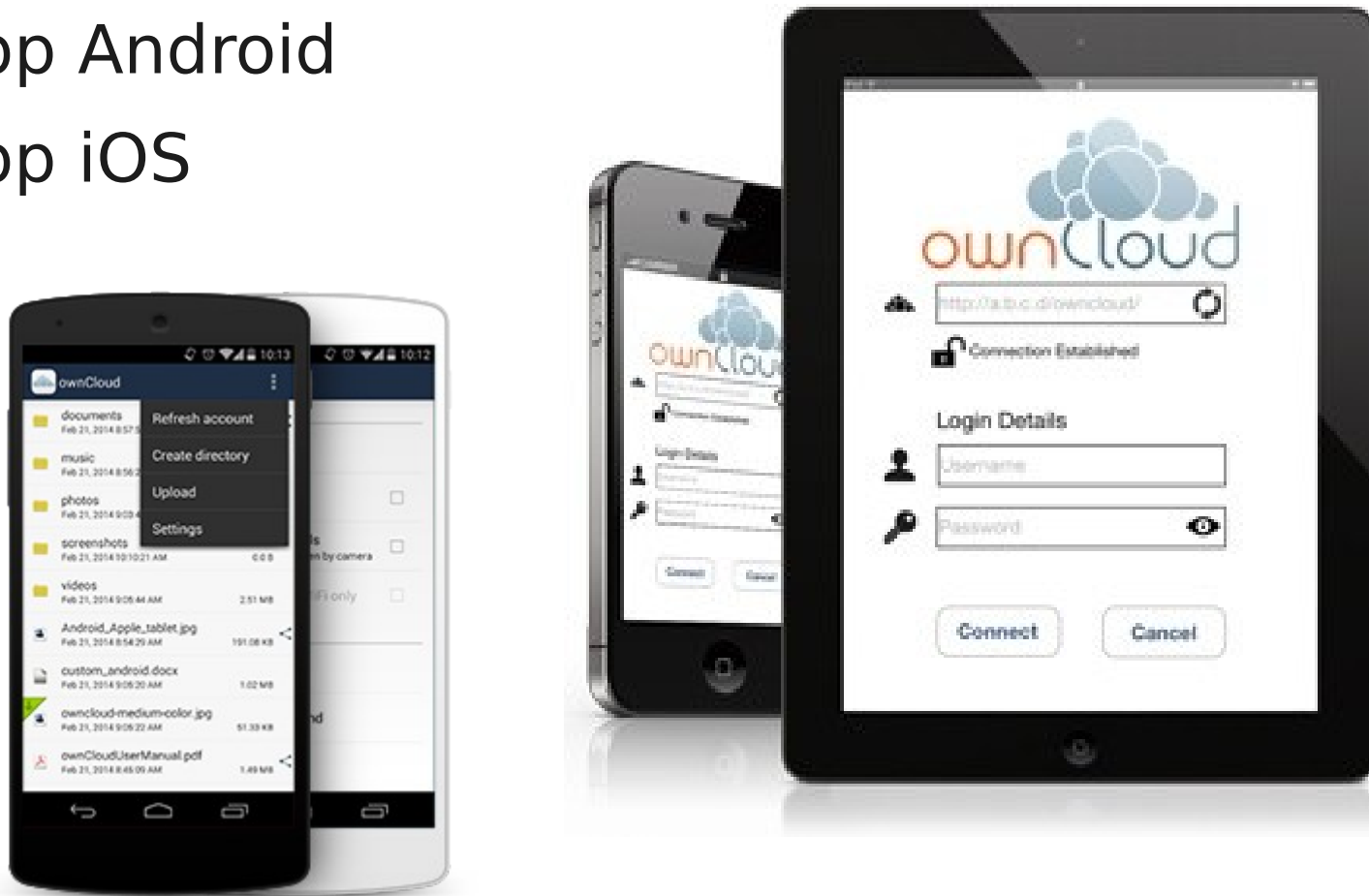
```
pi@raspberrypi: ~  
pi@raspberrypi: ~  
pi@raspberrypi:~ $ host myowncloudaddress.ddns.net  
myowncloudaddress.ddns.net has address 79.35.170.190  
pi@raspberrypi:~ $ ping myowncloudaddress.ddns.net  
PING myowncloudaddress.ddns.net (79.35.170.190) 56(84) bytes of data.  
^C  
--- myowncloudaddress.ddns.net ping statistics ---  
2 packets transmitted, 0 received, 100% packet loss, time 1007ms  
  
pi@raspberrypi:~ $ █
```

SmartDevices APP + PC Client



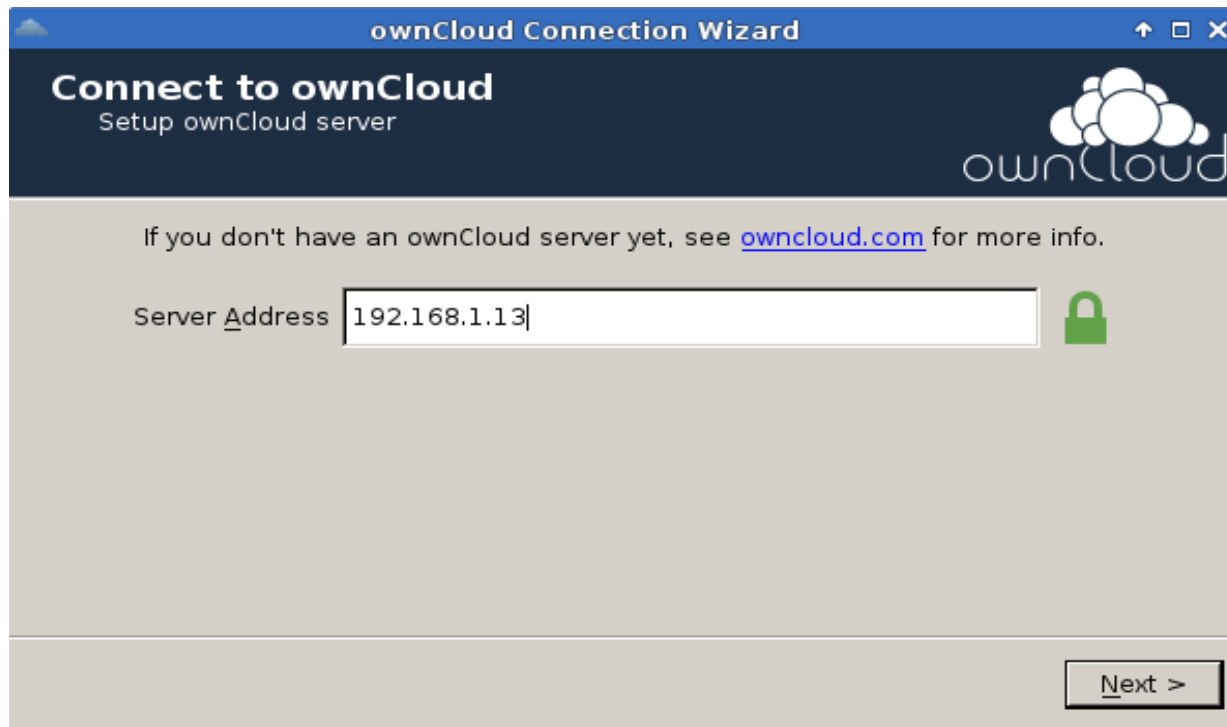
APP per smartphone e tablet

- App Android
- App iOS



Client per PC

- Installare e Configurare il CLIENT per pc [windows, linux, macosx] per il sync automatico dei file.



Considerazioni finali sulla sicurezza



Security

- **Rimuovere utente pi**
- **Cambiare tutte le password**
- **Scegliere password cazzute**
- **Aggiornare costantemente il sistema base e owncloud**
- **Backup, spesso, e su diversi supporti e/o servizi esterni**
- **Smanettare per conoscerlo a fondo**
- **Curiosare verso l'infinito e oltre**

Dubbi, Domande, Curiosità, Help



Grazie e arrivederci

FabLab Palermo

