CouchDb on a RaspberryPi

A quick walkthrough about how to install CouchDB on a tiny RaspberryPi

Posted by Massimiliano Fanciulli on February 1, 2015

My project MTrack (http://www.fanciullimassimiliano.it/2015/01/25/mtrack-is-open-source/) was initally based on Amazon DynamoDB for data storage. The system provided by Amazon AWS is great and powerful but can be a cost for small developer or for those just testing the solution. That's why when releasing MTrack as opensource i decided to replace it with CouchDB.

CouchDB is a JSON document database that is really powerful and capable of greatly scaling. Being it open source could be an optimal solution for those who may already have an instance or for developers, hosting both server and DB locally. I decide to try to install CouchDB on a RaspberryPI, which is probably capable of running the server, even though it couldn't reach great performances.

##Installation on a Raspberry Pi I installed Raspbian on a brand new RaspberryPI and installed CouchDB following a tutorial i found on internet. While below i give a reworked version of it , the full tutorial can be found here (http://jeeonthepi.blogspot.it/2014/08/installing-couchdb-1.html).

Edit the apt servers list:

```
sudo nano /etc/apt/sources.list
```

Add the following line at the end of the file:

```
deb http://packages.erlang-solutions.com/debian wheezy contrib
```

Add the Erlang Solutions public key for apt-secure using following commands:

```
wget http://packages.erlang-solutions.com/debian/erlang_solutions.asc
```

```
sudo apt-key add erlang_solutions.asc
```

Update the aptitude repository by running:

```
sudo apt-get update
```

After updating start installing all needed dependencies:

```
sudo apt-get install erlang-nox erlang-dev libmozjs185-1.0 libmozjs185-dev libcurl4-openssl-dev libicu-dev
```

Now create the couchDB account by executing:

```
sudo useradd -d /var/lib/couchdb couchdb
```

```
sudo mkdir -p /usr/local/{lib,etc}/couchdb /usr/local/var/{lib,log,run}/couchdb /var/lib/couchdb
```

```
sudo chown -R couchdb:couchdb /usr/local/{lib,etc}/couchdb /usr/local/var/{lib,log,run}/couchdb
```

```
sudo chmod -R g+rw /usr/local/{lib,etc}/couchdb /usr/local/var/{lib,log,run}/couchdb
```

All the dependencies are now set. Download the source code and unpack it:

```
wget http://ftp-stud.hs-esslingen.de/pub/Mirrors/ftp.apache.org/dist/couchdb/source/1.6.0/apache-couchdb-1.6.0.tar.gz tar xzf apache-couchdb-*.tar.gz
```

Move inside the unpacked code and configure the package:

```
cd apache-couchdb-1.6.0
```

```
./configure --prefix=/usr/local --with-js-lib=/usr/lib --with-js-include=/usr/include/js --enable-init
```

When finished it is time to build and install CouchDB:

```
make && sudo make install
```

It will take some time to build the whole package. When finished set it as daemon by running:

```
sudo chown couchdb:couchdb /usr/local/etc/couchdb/local.ini
```

```
sudo ln -s /usr/local/etc/init.d/couchdb /etc/init.d/couchdb
```

```
sudo /etc/init.d/couchdb start
```

```
sudo update-rc.d couchdb defaults
```

By default CouchDB is bound at http://localhost:5984. To test if everything works properly you can run:

```
curl http://127.0.0.1:5984/
```

If you see an output like the following everything is set properly:

```
{"couchdb":"Welcome","uuid":"dc91fe432758b49a0d4708cbfcffaedb","version":"1.6.0","vendor":{"name":"The Apache Software Foundation","version":"1.6.0"}}
```

Updating binding port

You will probably want to access CouchDB from outside the RaspberryPI. To do this, simply edit the file /usr/local/etc/couchdb/local.ini

Update the field binding_address by putting the RaspberryPI IP address. Reboot the system and check if Futon is accessible through a Web Browser at http://RASPBERRY_IP:5984/_utils

Updating database location

By default the CouchDB database is stored locally. This is not a good choice for the Raspberry Pi which runs on a MicroSD card. These indeed have limited read/write cycles and storing the database in it may make their life shorter. The solution i followed is to store the CouchDB database on a NAS, whose disk space is mounted on the local file system.

In order to do this i created a folder where the Samba share will be mounted:

```
sudo mkdir -p /media/networkshare/public
```

Then i edited the fstab: > sudo nano /etc/fstab

and added the following line, customizing the fields NAS_IP, SHARE_NAME, USER and PASSWORD:

```
//NAS_IP/SHARE_NAME /media/networkshare/public cifs
username=USER,password=PASSWORD,uid=1000,gid=1000,iocharset=utf8 0 0
```

As you can see my Samba share is password protected. If your is not, remove the username and password keywords.

Final notes

Inital tests seems to demonstrate that the Raspberry Pi can run CouchDb with acceptable performances. Unknown to me are real performances in case of heavy loads. This should be better analyzed because CouchDB supports a distributed mode and making a cluster of Raspberry PIs is not expensive.

If you don’t have a Raspberry Pi

Here it is all you need to buy for a fully functional Raspberry platform:

Raspberry Pi Model B+ (http://geni.us/1F1q)

USB Wifi dongle (http://geni.us/3RPd)

MicroSD card with adapter (http://geni.us/bL1)

Power adapter (http://geni.us/yuG)

Raspberry Pi case (http://geni.us/SSO)

---

krists_ozols · a year ago

Don't forget to set up logrotate so that couchdb log files does not fill whole SD card. It can be done by executing this command:

echo "include /usr/local/etc/logrotate.d" | sudo tee -a /etc/logrotate.conf

Massimiliano Fanciulli Mod · a year ago

Hi Krists,

thanks for hinting this! I'll put it in the post.

Massimiliano Fanciulli Mod · a year ago

Hello Marc,

no i don't have any direct experience on that but i think it shouldn't be any different from the cluster settings with "plain" hosts. I point you to the official documentation for that: http://guide.couchdb.org/draft...

Let me know if ngnix works properly on a raspberry pi. I think it will, i have seen it in a couple of youtube videos.
Cheers,
Massimiliano

Marc Geh · a year ago

Nice idea. I done the same on my B+, but my goal is to do 4 Pi's into a master-slave cluster.
You got any experience with that?
Just trying to get 'ngnix' run at the moment.
Cheers, Marc

(/feed.xml) (https://twitter.com/fanciullim)
(https://www.facebook.com/massimiliano.fanciulli)
(https://github.com/fanciulli)