Features Business Explore Pricing

This repository Search

Sign in or Sign up

jeroen / **opencpu-server**

Watch 11 Star 32 Fork 17

Code Pull requests 0 Projects 0 Pulse Graphs

Branch: master ▾ opencpu-server / **debian** /

Create new file Find file History

This branch is 39 commits behind v2.0.

Pull request Compare

**jeroen** update to 1.6.2 Latest commit edbb71a on May 28, 2016

..

| | | |
|---|---|---|
| changelog | update to 1.6.2 | 11 months ago |
| compat | many fixes | 4 years ago |
| control | update stack | 11 months ago |
| copyright | cleanups and stuff | 3 years ago |
| opencpu-cache.dirs | added readme | 4 years ago |
| opencpu-cache.install | add systemd file for ocpu-cache | 2 years ago |
| opencpu-cache.links | eof | a year ago |
| opencpu-cache.postinst | update certs and cleanup | a year ago |
| opencpu-cache.postrm | update certs and cleanup | a year ago |
| opencpu-cache.prerm | first commit | 4 years ago |
| opencpu-lib.install | first commit | 4 years ago |
| opencpu-server.dirs | fix dirs | 3 years ago |
| opencpu-server.install | update install file | 2 years ago |
| opencpu-server.links | update certs and cleanup | a year ago |
| opencpu-server.postinst | update certs and cleanup | a year ago |
| opencpu-server.postrm | update certs and cleanup | a year ago |
| opencpu-server.prerm | bunch of cleanup | 3 years ago |
| readme.md | tweak readme a bit | a year ago |
| rules | many fixes | 4 years ago |

**readme.md**

# Building OpenCPU on Debian/Ubuntu

*How to build OpenCPU on Debian or Ubuntu*

## Update R (optional, but recommended)

Because `r-base` packages included with Debian/Ubuntu are often old, we first add a repository with a recent version of R. On **Ubuntu** we can use Michael Rutter's launchpad repository:

```
sudo add-apt-repository -y ppa:marutter/rrutter
sudo apt-get update
```

Alternatively, on **Debian** use `r-base` packages from CRAN (see details). For example on Debian 8.0 ("jessie")

```
# Become root
sudo -i

# Add Wheezy CRAN repo for R 3.0+
apt-key adv --keyserver keyserver.ubuntu.com --recv-key 381BA480
echo "deb http://cran.rstudio.com/bin/linux/debian jessie-cran3/" > /etc/apt/sources.list.d/cran.list
apt-get update

# Quit root
exit
```

## Building OpenCPU packages

First install dependencies required for building OpenCPU:

```
# Update system
sudo apt-get update
sudo apt-get dist-upgrade -y

# Install build dependencies
sudo apt-get install -y wget make devscripts apache2-dev apache2 libapreq2-dev r-base r-base-dev libappa
```

Build rApache ( `libapache2-mod-r-base` ). Run this **not** as root (use a regular user).

```
cd ~
wget https://github.com/jeffreyhorner/rapache/archive/v1.2.8.tar.gz
tar xzf v1.2.8.tar.gz
cd rapache-1.2.8
dpkg-buildpackage -us -uc
```

Build OpenCPU Cloud Server ( `opencpu-server` and `opencpu-cache` ). Run this **not** as root.

```
cd ~
wget https://github.com/jeroenooms/opencpu-server/archive/v1.6.tar.gz
tar xzf v1.6.tar.gz
cd opencpu-server-1.6
dpkg-buildpackage -us -uc
```

## Installing OpenCPU server

To install the cloud server, simply install the `deb` packages in the following order:

```
cd ~
sudo dpkg -i libapache2-mod-r-base_*.deb
sudo dpkg -i opencpu-lib_*.deb
sudo dpkg -i opencpu-server_*.deb
```

## Installing OpenCPU caching server (optional)

The `opencpu-cache` package is a reverse proxy for caching and load balancing with OpenCPU. When installed, it automatically preroutes all incomming traffic on ports 80 and 443 through nginx. Only install this when you expect serious traffic.

```
# Dependencies
sudo apt-get install nginx

# Package builds
cd ~
sudo dpkg -i opencpu-cache_*.deb
```

Note that it is possible to install `opencpu-cache` on another server than `opencpu-server` if you update the nginx back-end config accordingly.

## Enable AppArmor support (optional, debian only)

OpenCPU uses AppArmor to enforce advanced security policies. AppArmor support is installed by default on Ubuntu, but in Debian we first need to enable it in the kernel. To do so, edit `/etc/default/grub` and add `security=apparmor` to the `GRUB_CMDLINE_LINUX` line. For example it would read:

```
GRUB_CMDLINE_LINUX="security=apparmor"
```

Update the grub config and reboot:

```
sudo update-grub
sudo reboot
```

After rebooting, install the apparmor packages and verify that it is enabled:

```
sudo apt-get install apparmor-utils
sudo aa-status
```

Restart OpenCPU and check the log files to confirm that apparmor works:

```
sudo service opencpu restart
sudo tail /var/log/apache2/error.log -n30
```