

eOn Install Documentation

Jonsson Group

September 3, 2010

1 Package manager

The different components of a Boinc installation might be obtained using a package manager. However, we have not tried such an installation.

2 Boinc client libraries on Ubuntu 10

Take a look at <http://boinc.berkeley.edu/trac/wiki/ServerIntro>, which served as inspiration for this text. In this document ... is used to represent a path that depends on the specific installation.

2.1 Prerequisites

Start by ensuring that all the packages listed below are installed:

- libssl-dev
- libglut3-dev
- glutg3-dev
- libglui-dev
- libglitz-glx1-dev
- libsdl1.2-dev
- libcurl4-gnutls-dev
- freeglut3
- freeglut3-dev
- libsm-dev
- libice-dev
- libxmu-dev

- libxi-dev
- libx11-dev
- libjpeg62-dev
- libgtk2.0-0
- libgtk2.0-0-dev

2.2 Making the Boinc client libraries

```
./configure --disable-server
make
sudo make install
```

Where the install option makes the created libraries and header files accessible system wide.

3 Boinc server setup on Ubuntu 10

Take a look at <http://boinc.berkeley.edu/trac/wiki/ServerIntro>, which served as inspiration for this text.

3.1 Prerequisites

Start by ensuring that all the packages listed below are installed:

- apache2-mpm-prefork
- libapache2-mod-php5
- mysql-client-5.0
- mysql-server-5.0
- php5-mysql
- php5-cli
- php5-gd
- phpmyadmin
- python-mysqldb
- libmysql++-dev
- libssl-dev

3.1.1 boincadm user

Make a new user(boincadm) to handle boinc:

```
sudo useradd -m -s /bin/bash boincadm
```

Add www-data to group boincadm with:

```
sudo usermod -G boincadm www-data
```

3.1.2 MySQL

Start mysql:

```
mysql -h localhost -u root [-p]
```

Where the `-p` only should be used if the MySQL `root` user got a password.
Configure mysql

```
GRANT ALL ON *.* TO 'boincadm'@'localhost';  
SET PASSWORD FOR 'boincadm'@'localhost'='XXX';  
exit
```

where `XXX` represents the password.

3.2 Making the Boinc server software using boincadm

Start by changing user to boincadm:

```
su boincadm
```

It might be nesseeary to use `sudo`.

Get the latest stable code from BOINC

```
svn co http://boinc.berkeley.edu/svn/branches/server_stable
```

Rename the obtained folder to boinc

```
mv server_stable boinc  
cd boinc
```

Build boinc server.

```
./_autosetup  
./configure --disable-client  
make
```

If `./autosetup` complains about a too old version of autoreconf the change line
`if check_version autoreconf 2.58` to `if check_version /usr/bin/autoreconf2.50`
in this file.

4 Making a project

Make sure the hostname is the same as the address used when connecting to the computer. This setting are specified in `/etc/hosts` and `/etc/hostname`.

Enter the Boinc tools directory and create a new project:

```
cd ../boinc/tools
./make_project eon
```

where `eon` can be replaced with another project name if preferred. Follow the instructions in the created readme file until the command `bin/xadd`.

4.1 Secure the ops page

Go to `eon/html/ops` and run:

```
cd ../eon/html/ops
htpasswd -c .htpasswd USERNAME
```

where `USERNAME` is the name of the user that should have access to the ops page. In `eon/html/project/project.inc` edit function `auth_ops()` by changing the default deny access `auth_ops_deny()` to `//auth_ops_deny()`.

```
cd ../eon/html/project/
nano project.inc
```

4.2 Prepare the eOn to use the Boinc Communicator

Go to the root of the project and replace `project.xml` with the one provided by the eon source code in the directory `boinc`

```
cd ../eon
mv project.xml project.xml.org
cp /PATH_EON_SOURCE_CODE/boinc/project.xml .
```

Furthermore copy the template files:

```
cp /PATH_EON_SOURCE_CODE/boinc/client_re.xml ./template/client_re.xml
cp /PATH_EON_SOURCE_CODE/boinc/client_wu.xml ./template/client_wu.xml
```

Compile the eon client with Boinc support but without fortran potentials:

```
cd /PATH_EON_SOURCE_CODE/client
make BOINC=1 NO_FORTRAN=1
```

Copy the obtained executable to the application area of the Boinc project

```
cd ../eon/apps/
mkdir client
cd client
mkdir client_VERSION_PLATFORM
cd client_VERSION_PLATFORM
cp /PATH_EON_SOURCE_CODE/client client_VERSION_PLATFORM
```

where the `client_VERSION_PLATFORM` would be `client_1.0_i686-pc-linux-gnu` for the first version (1.0) of the executable running on intel architecture on a linux platform (`i686-pc-linux-gnu`). Add the following deamons to the `.../eon/config.xml`:

```
<daemon>
  <cmd>
    sample_trivial_validator -d 3 -app client
  </cmd>
</daemon>
<daemon>
  <cmd>
    sample_assimilator -d 3 -app client
  </cmd>
</daemon>
```

To get the newly added application into the Boinc go to the root and run:

```
cd .../eon
./bin/xadd
./bin/update_versions
./bin/start
```

Add the environmental variable `BOINC_PROJECT_DIR` containing the path to the project:

```
cd
nano .bash_profile
BOINC_PROJECT_DIR=".../eon"
export BOINC_PROJECT_DIR
```

and finally change the paths stated in the eon configuration file `config.ini` among the eon server files.