ZENBU Installation guide

Although the convenience of using the online public zenbu http://fantom.gsc.riken.jp/zenbu has many advantages, we understand that some users will want to install a local copy of ZENBU. This Installation guide is here to help you install a ZENBU server on your machine from source code. This guide will provide you with a working ZENBU server, but will not have a local copy of data. Data will be remotely shared dynamically from the RIKEN ZENBU server.

The ZENBU source code is available for download from our sourceforge page https://sourceforge.net/projects/zenbu ZENBU is written as a client/server or web2.0 designed system. It is written in c++ and perl for server-side web services and command line tools while the web clients are written in javascript.

The installation procedure requires several key steps

- phase1: basic server machine setup: apache, fastcgi, mysql server.

- phase2: compile and install ZENBU. including all c/c++ and perl library dependencies

- phase3: configuring the server
  - creating support mysql databases
  - editing the server configuration file

- phase4: loading data
  - loading new genomes
  - loading data via the command line

Server System setup

ZENBU is a web application which requires a basic web-stack of apache2, fastcgi, mysql, sqlite3. On debian linux systems the following apt-get commands should install the required systems

```
apt-get install apache
apt-get install libapache2-mod-fcgid
apt-get install mysql-server
apt-get install mysql-client
apt-get install sqlite3
```

Compiling ZENBU from source code

C libraries dependencies

In order to compile ZENBU you will need the following c/c++ libraries : libmysql-devel, libsqlite3-devel, libfcgi, libcurl, libcurl-devel, libssl, libcrypto, rapidxml http://rapidxml.sourceforge.net/, boost 1.43 or later http://www.boost.org, bamtools https://github.com/pezmaster31/bamtools, lz4 compression library, http://code.google.com/p/lz4/

For debian linux systems the following apt-get commands should install the required libraries.

```
apt-get install g++
apt-get install cmake
apt-get install libfcgi-dev
apt-get install libmysqlclient-dev
apt-get install libmysql++-dev
apt-get install libsqlite3-dev
apt-get install expat
apt-get install libexpat1-dev
apt-get install openssl
apt-get install uuid-runtime
apt-get install libssl-dev
apt-get install libcrypto++-dev
```

ZENBU uses the boost library version 1.43 or greater. This library is sometimes available with linux distributions. In debian7 for example you can install boost from libboost-all-dev. If this is not available or there are compatibility issues, we have included a copy of boost 1.43 inside the zenbu/src/c++ directory. Or you can download directly from http://www.boost.org

```
apt-get install libboost-dev
apt-get install libboost-all-dev
```

ZENBU uses libcurl for SMTP mail functionality. http://curl.haxx.se/libcurl/c/curl_easy_setopt.html SMTP was added into libcurl 7.20 but many linuxes ship with libcurl 7.19 If your linux has the libcurl4 libraries in their package manager you can install with apt-get. If these packages are not available, please install the most recent version of libcurl from source code from http://curl.haxx.se/download.html

```
apt-get install libcurl4-nss-dev
apt-get install libcurl4-openssl-dev
```

Special libraries installed from source

ZENBU also requires several libraries that are often not pre-packaged with linux distributions. Please download and install these from source

bamtools

ZENBU provides native BAM file support through the bamtools library. Please download, compile and install from https://github.com/pezmaster31/bamtools. This can be downloaded via git or as a zip from this page. Compiling is done by

```
cd bamtools-master/
mkdir build
cd build
cmake ..
make
make install
```

After installation, you will most likely need to rebuild your shared-library cache. This is done with ldconfig. By default bamtools installs in /usr/local/lib/bamtools which may not be found by ldconfig.

On debian7

```
cd /etc/ld.so.conf.d
vi lib.conf
#add line /usr/local/lib/bamtools
#save file
ldconfig
```

Other library dependencies

ZENBU uses two other libraries: lz4 and rapidxml. These are generally not included in linux distributions but are also difficult to compile. To ease the compiling of ZENBU we have included these with the ZENBU source code and modified them as needed. We have tested compiling on debian6 and debian7. The following is for information only and to give credit to the library developers. There is no need to download again, but we provide the information if you run into difficulty.

lz4 compression library

ZENBU uses the fast compression library lz4 http://code.google.com/p/lz4/. We have include a modified copy in the zenbu source tree so there is no need to install separately.

rapidxml

rapidxml is a fast inline xml parser. http://rapidxml.sourceforge.net/ it is a header only c++ library so it does not need compiling. We have included a copy in the zenbu source tree so there is no need to install it separately

Perl Libraries

ZENBU use perl for the OpenID login webservices and requires the following libraries. They can be install usually though the standard linux distribution apt-get call, but if there are problems, we also list up the libraries.

```
apt-get install libdata-uuid-perl
apt-get install libyaml-perl
apt-get install libclass-dbi-mysql-perl
apt-get install libclass-dbi-sqlite-perl
apt-get install libcgi-application-perl
apt-get install libcgi-fast-perl
apt-get install libnet-openid-common-perl
apt-get install libnet-openid-consumer-perl
apt-get install libcrypt-openssl-bignum-perl
apt-get install libio-all-lwp-perl
apt-get install liblwp-authen-oauth-perl
apt-get install liblwpx-paranoidagent-perl
apt-get install libnet-ping-external-perl
apt-get install libxml-treepp-perl
apt-get install libcache-perl

Data::UUID
XML::TreePP
Time::HiRes
File::Temp
Cache::File
Getopt::Long
Compress::Zlib
DBI
DBD::mysql
DBD::SQLite
Math::BigInt
Math::BigRat
Math::BigInt::FastCalc
XML::SAX
XML::Simple
XML::XPath
XML::Parser
CGI
CGI::Session
CGI::Fast
Crypt
Crypt::DH
Crypt::DH::GMP
YAML LWP::UserAgent
LWPx::ParanoidAgent
Net::Yadis
Net::Ping
Net::SSL
Net::OpenID::Common
Net::OpenID::Consumer
```

Compiling ZENBU

ZENBU requires a specific directory structure to support background processing and data uploading. In particular ZENBU requires the server machine to have a /zenbu directory at the root level.

```
mkdir /zenbu /zenbu/src /zenbu/dbs /zenbu/cache /zenbu/bin /zenbu/server_config
/zenbu/users /zenbu/www

chown www-data /zenbu/cache
chown www-data /zenbu/users

#the owner must be the apache process owner, on some systems is it
#httpd, or apache or www-data alternate is to do chmod 777 or chgrp to
#allow the apache process to write into the directories cache and users
```

Next unpack your ZENBU source code. If you received as a .tar.gz it will be created with absolute paths. If not please install in the /zenbu/src directory with the version number of the ZENBU you have downloaded.

```
tar -Pxzf ZENBU_2.8.2.tar.gz

#this was made with absolute paths so it should unpack
#into /zenbu/src/ZENBU_2.8.2
```

after unpacking the source code, please follow these steps to compile. If all the libraries and dependencies are correct this should compile with out errors but maybe a few warnings depending on the version of linux you are using.

```
cd /zenbu/src/ZENBU_2.8.2/c++
make
```

Installing ZENBU web-site, cgi and tools

The ZENBU website is located in the source code in the subdirectory www/zenbu. To make setup easier we recommend using the /zenbu/www directory for different versions of the web-site code and then symbolic link from the apache html directory to the appropriate zenbu version.

```
cd /zenbu/src/ZENBU_2.8.2/www
cp -rp zenbu /zenbu/www/zenbu_2.8.2
cd /zenbu/www
ln -s zenbu_2.8.2 zenbu

cd /var/www  #or where your apache keeps the html pages
ln -s /zenbu/www/zenbu  zenbu
```

The next step is to install the commandline and background processing tools. These get installed into the /zenbu/bin directory

```
cd /zenbu/src/ZENBU_2.8.2/c++/tools
make
make install
#will copy the commandline tools into /zenbu/bin
```

Next compile and copy the ZENBU cgi programs into the zenbu website cgi directory

```
cd /zenbu/src/ZENBU_2.8.2/c++/cgi
make
cp *cgi /zenbu/src/ZENBU_2.8.2/www/zenbu/cgi
```

Configuration of ZENBU server

This section of the Installation guide we will discuss the steps of configuring the server, creating the support mysql databases, loading data via the command-line, and adding new genomes.

Configuring Apache to execute ZENBU cgi

Depending on how your apache was previously configured, you will most likely need to modify the apache config to allow ZENBU cgi webservices to execute. The general procedure for configuring cgi in Apache 2.2 is describe here http://httpd.apache.org/docs/2.2/en/howto/cgi.html and http://httpd.apache.org/docs/trunk/howto/cgi.html

On debian7 linux systems for example, apache2 is configured in /etc/apache2/. There is a directory structure as follows

```
-rw-r--r-- 1 root root  9640 Mar  4  2013 apache2.conf
drwxr-xr-x 2 root root  4096 Feb  7 20:05 conf.d
-rw-r--r-- 1 root root  1465 Mar  4  2013 envvars
-rw-r--r-- 1 root root 31063 Oct 21  2012 magic
drwxr-xr-x 2 root root  4096 Feb  8 02:11 mods-available
drwxr-xr-x 2 root root  4096 Feb  7 21:33 mods-enabled
-rw-r--r-- 1 root root   750 Mar  3  2013 ports.conf
drwxr-xr-x 2 root root  4096 Feb  8 02:13 sites-available
drwxr-xr-x 2 root root  4096 Feb  8 02:13 sites-enabled
```

Check mods-enabled and make sure there are the following for cgi and fcgi

```
rwxrwxrwx 1 root root 27 Feb  7 20:05 cgid.conf -> ../mods-available/cgid.conf
rwxrwxrwx 1 root root 27 Feb  7 20:05 cgid.load -> ../mods-available/cgid.load
rwxrwxrwx 1 root root 28 Feb  7 21:33 fcgid.conf -> ../mods-available/fcgid.conf
rwxrwxrwx 1 root root 28 Feb  7 21:33 fcgid.load -> ../mods-available/fcgid.load
```

Then in the sites-enabled subdirectory there should be a default config

```
 000-default
```

add the following section

```
<Directory "/var/www/zenbu/">
   Options +ExecCGI
   AddHandler cgi-script .cgi .fcgi .pl
   AddHandler fcgid-script .fcgi
</Directory>
```

Restart your apache server

```
sudo /etc/init.d/apache restart
```

Finally check to make sure your apache/cgi is setup and running correctly. In a web browser point at your newly installed zenbu and go to the cgi directory. For example

```
http://apache_hostname/zenbu/cgi
http://apache_hostname/zenbu/cgi/eedb_search.cgi
http://apache_hostname/zenbu/cgi/eedb_search.fcgi
```

If apache is setup and the compile was successful, it should execute and return an HTML page with some status information. The eedb_search.fcgi should return with increasing "invocation number" and the same PID on successive reloads.

If there are problems, check the apache error log

```
 sudo tail -f /var/log/apache2/error.log
```

Mysql support database creation

ZENBU uses a mysql server for the user system, visualization view/track/script config saving, data sharing and for genome sequence. ZENBU needs two specific users to be added onto the mysql server. The user "read" must be created with password "read". The user "zenbu_admin" can have any password but we recommend using the password "zenbu_admin"

```
CREATE USER 'read'@'%' IDENTIFIED BY 'read';
CREATE USER 'zenbu_admin'@'%' IDENTIFIED BY 'zenbu_admin';
```

The "read" user will need to have the follow grant permissions for each relevant database

```
GRANT SELECT, CREATE TEMPORARY TABLES, LOCK TABLES on specific_db.* to 'read'@"%";
```

Next, you need to create the three databases mentioned above

- create "user" database maybe like "zenbu_users"
- create "public sharing" database maybe like "zenbu_public_share"
- create "curation" database maybe like "zenbu_curated"

you create a zenbu database by the mysql commands. You may choose different database names if you would like.

```
CREATE DATABASE zenbu_users;
CREATE DATABASE zenbu_public_share;
CREATE DATABASE zenbu_curated;
GRANT SELECT, CREATE TEMPORARY TABLES, LOCK TABLES on zenbu_curated.* to 'read'@"%";
GRANT SELECT, CREATE TEMPORARY TABLES, LOCK TABLES on zenbu_public_share.* to 'read'@"%";

#DO NOT do give read:read grants to the user database

GRANT SELECT, CREATE TEMPORARY TABLES, LOCK TABLES, INSERT, UPDATE, CREATE, ALTER,
DELETE, INDEX on zenbu_users.* to 'zenbu_admin'@"%";

GRANT SELECT, CREATE TEMPORARY TABLES, LOCK TABLES, INSERT, UPDATE, CREATE, ALTER,
DELETE, INDEX on zenbu_curated.* to 'zenbu_admin'@"%";

GRANT SELECT, CREATE TEMPORARY TABLES, LOCK TABLES, INSERT, UPDATE, CREATE, ALTER,
DELETE, INDEX on zenbu_public_share.* to 'zenbu_admin'@"%";
```

for the "user database" run these commands from the command-line

```
 cmdline> mysql -hXXXX -uYYYY -pZZZZ -PWWWW   zenbu_users  <
/zenbu/src/ZENBU_2.8.2/sql/schema.sql

cmdline> mysql -hXXXX -uYYYY -pZZZZ -PWWWW   zenbu_users  <
/zenbu/src/ZENBU_2.8.2/sql/system_tables.sql

cmdline> /zenbu/bin/zenbu_register_peer -url
"mysql://zenbu_admin:zenbu_admin@mysql_hostname:3306/zenbu_users" -newpeer
```

Note that ZENBU uses a URL-like notation to pointing to sql databases. The format is

```
 mysql://<mysql_username>:<mysql_password>@<mysql_hostname>:<mysql_port>/<database_name>
sqlite:///<full_path_to_sqlite_database>
```

for all other databases

```
 cmdline > mysql -hXXXX -uYYYY -pZZZZ -PWWWW   my_other_db_name  < /zenbu/src/ZENBU_
2.8.2 /sql/schema.sql

cmdline > mysql -hXXXX -uYYYY -pZZZZ -PWWWW   my_other_db_name  <
/zenbu/src/ZENBU_2.8.2/sql/assembly_data.sql

cmdline> /zenbu/bin/zenbu_register_peer -url
"mysql://zenbu_admin:zenbu_admin@mysql_hostname:3306/my_other_db_name" -newpeer
```

in addition you will need to enable zenbu_admin with some "super user" privileges

```
GRANT CREATE, ALTER, DELETE, CREATE TEMPORARY TABLES, INDEX, SHOW DATABASES, LOCK TABLES
on *.* to 'zenbu_admin'@"%";

GRANT CREATE, ALTER, DELETE, CREATE TEMPORARY TABLES, INDEX,  INSERT, SHOW DATABASES,
UPDATE, SELECT, LOCK TABLES on zenbu*.* to 'zenbu_admin'@"%";
```

if the mysql server is secured or only for use by ZENBU, it will be easier to give “super-user” privileges to “zenbu_admin” for all databases on the server like the the following

```
GRANT CREATE, ALTER, DELETE, CREATE TEMPORARY TABLES, INDEX,  INSERT, SHOW DATABASES,
UPDATE, SELECT, LOCK TABLES on *.* to 'zenbu_admin'@"%";
```

ZENBU web-service configuration

First step, copy the example server configuration file to the /zenbu/server_config directory

```
cp EXAMPLE_eedb_server_config.xml /zenbu/server_config/zenbu_2.8.2_config.xml
cd /zenbu/server_config
ln -s zenbu_2.8.2_config.xml active_config.xml
```

Next you will need to edit the file related to your specific server setup. The server configuration file is in XML and hopefully is easy to read. Here is an example

```
<zenbu_server_config>
  <eedb_root>/zenbu/src/ZENBU_2.8.2/</eedb_root>
  <eedb_user_rootdir>/zenbu/users</eedb_user_rootdir>
  <cache_dir>/zenbu/cache</cache_dir>
  <user_db>mysql://zenbu_admin:zenbu_admin@mysql_hostname:3306/zenbu_users</user_db>
  <user_admin_password>zenbu_admin</user_admin_password>

  <smtp_server></smtp_server>
  <smtp_user></smtp_user>
  <smtp_password></smtp_password>
  <smtp_from></smtp_from>

  <federation_seeds>
    <seed>zenbu://fantom.gsc.riken.jp/zenbu/</seed>
  </federation_seeds>

  <public_collaboration>
    <database>mysql://read:read@mysql_hostname:3306/zenbu_public_share</database>
    <name>public sharing - ZENBU MYINSTITUTE server</name>
    <description>collaboration for sharing public data and public views/tracks/scripts on
ZENBU virtual machine</description>
  </public_collaboration>
  <curated_collaboration>
    <database>mysql://read:read@mysql_hostname:3306/zenbu_curated</database>
    <name>curated - ZENBU MYINSTITUTE server</name>
    <description>curated data collaboration for ZENBU virtual machine</description>
    <curators>
      <user>jessica.severin@gmail.com</user>
    </curators>
  </curated_collaboration>

  <default_genome>hg19</default_genome>
  <session_name>ZENBU_MYINSTITUTE_SESSID</session_name>
  <server_name>apache_hostname</server_name>
  <web_root>http://apache_hostname/zenbu</web_root>
  <web_uuid>4e6958e6-f404-4219-ac90-0d7c95f63742</web_uuid>
</zenbu_server_config>
```

In addition there is another perl specific configuration file used by the OpenID login processes. This configuration file exists inside the zenbu/cgi directory and is called eedb_server.conf Here is the example from the ZENBU virtual machine. In a future release we are planning to consolidate all the configuration into the XML formated file.

```
[
{ TYPE  => 'EEDB_URL',
'user_db' => 'mysql://zenbu_admin:zenbu_admin@mysql_hostname:3306/zenbu_users',
'session_name' => "ZENBU_MYINSTITUTE_SESSID", },

{ TYPE  => 'EEDB_ENV',
'EEDB_ROOT' => "/zenbu/src/ZENBU_2.8.2/",
'EEDB_USER_ROOTDIR' => "/zenbu/users",   },

{ TYPE  => 'ZENBU_WEB',
'WEB_ROOT'    => "http://apache_hostname/zenbu",
'SERVER_NAME' => "apache_hostname"   },

{ TYPE => 'END' }
]
```

Loading genomes and data

Most data load can be performed through the web interface “upload” system. But there are several command-line projects for loading data.

Loading new genomes

In the current version of ZENBU (2.8.2), new genomes must be loaded into mysql databases use a command-line perl script. In future versions, we will be adding genome creation/loading into the upload system.

First create a new mysql database to hold the new genome sequence. Genome sequences are often very large (Human is 3billion bases and thus requires 3GB for the mysql database). Please be aware of this.

From inside the mysql server

```
CREATE DATABASE zenbu_susScr3_pig;
GRANT SELECT, CREATE TEMPORARY TABLES, LOCK TABLES on zenbu_susScr3_pig.* to 'read'@"%";

GRANT SELECT, CREATE TEMPORARY TABLES, LOCK TABLES, INSERT, UPDATE, CREATE, ALTER,
DELETE, INDEX on zenbu_susScr3_pig.* to 'zenbu_admin'@"%";
```

From the command-line

```
cmdline> mysql -hmysql_hostname —uzenbu_admin —pzenbu_admin —P3306 zenbu_susScr3_pig <
/zenbu/src/ZENBU_2.8.2/sql/schema.sql

cmdline> /zenbu/bin/zenbu_register_peer -url
"mysql://zenbu_admin:zenbu_admin@mysql_hostname:3306/zenbu_susScr3_pig" -newpeer
```

From inside the mysql server

```
INSERT INTO `assembly` (`assembly_id`, `taxon_id`, `ncbi_version`, `ucsc_name`,
`osc_name`, `release_date`, `taxon_name`, `sequence_loaded`) VALUES
(1,9823,'Sscrofa10.2','susScr3','susScr3','2011-09-07','Sus scrofa','y');


INSERT INTO `taxon` (`taxon_id`, `genus`, `species`, `sub_species`, `common_name`,
`classification`) VALUES (9823,'Sus','scrofa',NULL,'pig','cellular organisms; Eukaryota;
Opisthokonta; Metazoa; Eumetazoa; Bilateria; Deuterostomia; Chordata; Craniata;
Vertebrata; Gnathostomata; Teleostomi; Euteleostomi; Sarcopterygii; Dipnotetrapodomorpha;
Tetrapoda; Amniota; Mammalia; Theria; Eutheria; Boreoeutheria; Laurasiatheria;
Cetartiodactyla; Suina; Suidae; Sus');
```

The taxon information can be found at the NCBI Taxonomy Browser
http://www.ncbi.nlm.nih.gov/Taxonomy/Browser/wwwtax.cgi?mode=Info&id=9825

After the database has been created, you can now load the genome sequence. For the above example, Pig genome Sscrofa10.2 can be found at NCBI at this location
http://www.ncbi.nlm.nih.gov/assembly/GCF_000003025.5

and the actual sequence fasta file are here (by clicking the “GenBank FTP site” link)
ftp://ftp.ncbi.nlm.nih.gov/genbank/genomes/Eukaryotes/vertebrates_mammals/Sus_scrofa/Sscrofa10.2/Primary_Assembly/assembled_chromosomes/FASTA/

download the chr--.fa.gz files into a local directory on your server. For example into a directory /zenbu/genomes/susScr3_pig

From the command line

```
Cmdline> /zenbu/bin/eedb_chromChunkTool.pl -url
"mysql://zenbu_admin:zenbu_admin@mysql_hostname.gsc.riken.jp:3306/zenbu_susScr3_pig" -
assembly susScr3 -seqdir /zenbu/genomes/susScr3_pig/ -withseq -create -store
```

And finally modify the ZENBU web-service configuration XML file. Add the new genome into the section

```
<zenbu_server_config>
  . . . . . snip . . .

  <federation_seeds>
    <seed> mysql://read:read@mysql_hostname.gsc.riken.jp:3306/zenbu_susScr3_pig/</seed>
    <seed>zenbu://fantom.gsc.riken.jp/zenbu/</seed>
  </federation_seeds>

  . . . . . snip . . .
</zenbu_server_config>
```

make sure that the remote-connection to the RIKEN ZENBU server is at the bottom of the federation seeds list. This will ensure that the local databases are used before doing remote searches back to RIKEN.

Bulk command-line upload of datafiles

In addition to the web interfaces for loading data, we will provide (as of version 2.8.3) a command line tool for bulk or scripted uploading of data. An early beta-release version of the program was included in version 2.8.2. The program is /zenbu/bin/zenbu_upload. Full documentation will be provided on the next release.