

# Installation Guide for *Quarantine Overseen*

This PDF provides a brief explanation of how to set up the *Quarantine Overseen* software on the University of Oregon CS web server.

## VARIABLES

The following variables will be used throughout this installation guide, where the required substitution of a *VARIABLE* is denoted by *<VARIABLE>*:

1. *USER* : This is your user ID on the ix server. It can be found through two methods:
  - a. When you ``ssh user@ix.cs.uoregon.edu`` to secure shell into the ix server, *user* is your user ID
  - b. If you are in the ix server, issue the command ``echo $USER`` to find your user ID
2. *HOMEPATH* : The path of your home directory in the ix server. This can be reliably found by issuing the following commands sequentially:
  - a. If you are in the ix server, command ``echo $HOST``

## PREREQUISITES

1. You must have an account on the ix.cs.uoregon.edu server
2. Your home directory should have a public\_html folder which will display files inside when accessing `https://ix.cs.uoregon.edu/~<USER>/`

## INSTALLATION STEPS

1. Begin in the project directory
  - a. Command: ``cd /path/to/quarantineOverseen``
2. Move/copy all data to your public\_html folder residing in your home directory
  - a. Command: ``mv * ~/public_html/`` or ``cp -r * ~/public_html/``
3. Change your working directory to public\_html
  - a. Command: ``cd ~/public_html``
4. Make sure to acquire the correct values for the variables listed in section VARIABLES
  - a. *USER* : Found by commanding ``echo $USER`` in terminal on the ix server
  - b. *HOMEPATH* : Found by commanding ``echo $HOME`` in terminal on the ix server
5. Make sure to be in ~/public\_html, and then call *setup.sh* with proper arguments - This can be found by running *setup.sh* with zero arguments
  - a. Help : Command: ``bash setup.sh``
  - b. Run Installation: Command : ``bash setup.sh <USER> <HOMEPATH>``
6. If no errors are incurred, your site can be accessed via the URL  
“`https://ix.cs.uoregon.edu/~<USER>/index.php`”

## STOPPING THE SOFTWARE

1. Simply type ``mysqlctl stop``
  - a. You should receive a terminal response "Stopping mysqld process <pid>"
2. Make sure there are no mysql processes (under your user ID)
  - a. ``ps -a`` will report running process, and you can use ``kill -9 <PID>`` to kill the PID stated by the `ps` command
3. If you wish to remove access to the pages, regardless of the connection being terminated, you must remove all files from your `public_html` folder

## POTENTIAL ERRORS

- a. A common issue is that the port we wish to mount the MySQL database on is already in use. To remediate this issue, simply:
  1. ``cd ~/public_html``
  2. Use your text editing software of choice to change the PORT value on line 12 of `setup.sh` to another port of your choosing
    - i. To decide which port to use, come up with a random number,  $1000 \leq \text{PORT} \leq 65535$ , and call ``netstat -ant | grep <PORT>``
    - ii. No output means the port is not in use, output means you should pick a new PORT
  3. Save and quit
  4. Rerun `setup.sh`
- b. For any other errors, running ``mysqlctl status`` should shed some light on the problem
  1. More often than not, `~/my.cnf` has an error where the `.socket` or `.pid` MySQL files do not exist in the designated path
    - i. To remediate this particular issue, find the proper path name by commanding ``cd ~ ; find . -name "missingFile"``
    - ii. Supplement the previous path in `~/public_html/sql/init_my.cnf` with the output