# Zombie Beatdown Guide

## Installation – Server

1. Ensure Python is installed for the server to run.
2. Copy over the zb-server folder.
3. To start the server run servertest.py
4. To configure the server, go to the configuration page on the web server and input the necessary details.

## Installation - Worker

1. Refer to Cuckoo installation for each worker that will be running it: <http://cuckoo.readthedocs.org/en/latest/installation/>, Zombie Beatdown does not require any additional dependencies than what Cuckoo requires.
2. For each worker running Cuckoo extract zb\_worker folder and place into main cuckoo folder.
3. To use Zombie Beatdown, run cuckoo.py as per normal usage, and then run worker.py inside zb\_worker.
4. The worker.py will then pull tasks from the mongodb if there are any and passes them to cuckoo for analysis.
5. The reports generated by cuckoo will be sent to the server as specified by the configuration.

## Usage

1. To pass URLs for analyses, submit them through the web server’s main page.
2. The server will then pass the URLs as tasks to the mongodb.
3. Worker machines will pull tasks from the database when there are available VMs to analyze it.
4. Once the analysis is complete, the reports generated will be sent over to the server to compile for viewing.

## Upgrading Cuckoo

1. Take note of cuckoo config files to carry over to new update.
2. Simply copy over all cuckoo files over existing cuckoo installation.
3. Under cuckoo\lib\cuckoo\core\database.py, copy a new method under the Database class.

|  |
| --- |
| def get\_status(self, status=None, tid=None):  """Gets the status of a given tid  @param status: apply a filter according to the task status  @return: task status  """  count = 0  session = self.Session()  try:  if tid and status:  count = session.query(Task).filter((Task.id == tid), (Task.status == status)).count()  except SQLAlchemyError as e:  log.debug("Database error counting tasks: {0}".format(e))  return  finally:  session.close()  return count |