Dekū

Administrator Manual Template

Client
Shawn Squire

Team 6
Boris Boiko
Raymond Chan
Gilbert Kuo
Andrew Naviasky
Jeremy Neal

dekū
Administrator Manual

Table of Contents

Notes:

- Use technical, domain-specific prose targeted to the anticipated audience (i.e., the system administrator).
    - Do **not** assume your audience has already read any of your technical documents (e.g., SRS, SDD), but reference particular sections of those documents when warranted.
- You may reuse as much of any of your previous manuals as you wish.
- It is difficult to give paragraph estimates for Sections 3 and 4, as all products are unique. That is, there may be more or less to say about a particular product. Just be as thorough as you can for your product.

1. Introduction

1.1 Purpose of This Document

The purpose of this document is to assist the installation and setup of the Dekū web application. It is intended to be read the system administrator. This manual will contain step-by-step instructions to install and setup the Dekū web application. In addition, this manual will provide additional instructions for periodic administrative tasks.

1.1 References

Dekū System Requirements Specification	
Dekū Systems Design Document	
VirtualBox	http://www.virtualbox.org
Vagrant	http://www.vagrantup.com
Apache HTTP Server	http://httpd.apache.org
Python	http://www.python.org
SQLite	http://www.sqlite.org"
Python Package Index	http://pypi.python.org"
virtualenv	http://pypi.python.org/pypi/virtualenv"
Flask	http://flask.pocoo.org"
Flask-SQLAlchemy	http://pythonhosted.org/Flask-SQLAlchemy"
SQLAlchemy	http://www.sqlalchemy.org"
Jinja	http://jinja.pocoo.org"
MarkupSafe	http://www.pocoo.org/projects/markupsafe"
Tempita	http://pythonpaste.org/tempita"
Werkzeug	http://werkzeug.pocoo.org"
flask-bcrypt	http://pythonhosted.org/Flask-Bcrypt/"
pydenticon	http://pypi.python.org/pypi/pydenticon/"

2. System Overview

2.1 Background

The Dekū web application is a social web application designed to facilitate communication amongst university members. For more information about its design and usage, please refer to the Dekū System Requirement Specifications.

Once properly installed and set up, the Dekū web application is self sustaining from a system administrator's point of view; however, it is recommended that the database is backed up on a regular basis. For step-by-step backup instructions, please refer to section 3.3 of this document.

2.2 Hardware and Software Requirements

The Dekū web application is designed to run using Vagrant and VirtualBox. The following configuration has been tested on Windows 7 and Ubuntu 12.04; however, it should run on any operating system vagrant supports.   With a moderate amount of tweaking, the Dekū web application can run on any platform; however, it is greatly recommended that the application be run inside a vagrant box for ease of installation and added security.

Section 3 outlines the steps required to set up the Dekū web application using Vagrant and VirtualBox.

3. Administrative Procedures

3.1 Installation

3.1.1 Preparing the virtual machine
Download and the latest version of VirtualBox and Vagrant from the websites below:
VirtualBox: http://www.virtualbox.org
Vagrant: http://www.vagrantup.com

Create a file named *Vagrantfile* using a text editor and add the following:

```
Vagrant.configure("2") do |config|
  config.vm.box = "hashicorp/precise64"
  config.vm.provision :shell, :path => "bootstrap.sh"
  config.vm.network :forwarded_port, host: 4567, guest: 80
  config.vm.network :forwarded_port, host: 4568, guest: 5000
end
```

Save that, then create *bootstrap.sh* which will contain the command line arguments Vagrant will run prior to its first run. The following are needed for the Dekū web application:

```
apt-get update
apt-get install -y python-dev python-pip libapache2-mod-wsgi git
pip install virtualenv
rm -rf /var/www
```

Next, download the box specified earlier using the following command:

```
vagrant box add hashicorp/precise64
```

The next command runs the virtual machine:

```
apt-get install -y python-dev python-pip libapache2-mod-wsgi git
vagrant up
```

The virtual machine is now set up and running. In order to access the virtual machine, simply type:

```
vagrant ssh
```

3.1.2 Dekū web application setup

All steps in this section are inside the vagrant virtual machine. This manual will assume that the packaged Dekū web application files are contained in the folder `/home/vagrant/Section-2-Team-6`. This can be achieved through via git or SFTP if the packaged files are on the local machine.

```
SFTP
host: localhost       port: 2222
username: vagrant     password: vagrant
```

First, create a virtual environment and install the required dependencies:

```
virtualenv .venv
source .venv/bin/activate
pip install -r requirements.txt
```

Create `/home/vagrant/deku.wsgi` with the following text:

```
activate_this = '/home/vagrant/Section-2-Team-6/.venv/bin/activate_this.py'
execfile(activate_this, dict(__file__=activate_this))

import sys
sys.path.insert(0, '/home/vagrant/Section-2-Team-6')
from app import app as application
app.db.create_all()
```

Create a log directory `/home/vagrant/logs.`

Create the file `/etc/apache2/sites-available/deku` and enter the following within (Note: you need to run `sudo` with this):

```
<VirtualHost *:5000>
     WSGIDaemonProcess deku user=www-data group=www-data threads=5
     WSGIScriptAlias / /home/vagrant/deku.wsgi

     <Directory /home/vagrant>
          WSGIProcessGroup deku
          WSGIApplicationGroup %{GLOBAL}
          Order deny,allow
          Allow from all
     </Directory>

     ErrorLog /home/vagrant/logs/error.log
     LogLevel warn
     CustomLog /home/vagrant/logs/access.log combined
</VirtualHost>
```

Add the following to `/etc/apache2/ports.conf`:

```
NameVirtualHost *:5000
Listen 5000
```

Finally, run `sudo a2ensite deku` to tell the Apache server about your new routes and restart the apache server with `sudo service apache2 reload`.

From the host os, you should now be able to go to http://localhost:4567 in a browser and see the interface. The API can be accessed via http://localhost:4568.

Running the Dekū web application requires that both ports 4567 and 4568 are accessible from outside the local area network. This may require port forwarding. Please refer to your router's instruction manual for further help.

At this time, it is recommended that ports 4567 and 4568 remain blocked to outside machines since the API has not been secured.

3.2 Routine Tasks and Periodic Administration

It is recommended that system administrator routinely backs up the database in case of software/hardware failure. This can be achieved by copying and renaming app.db to a safe and secure location.

3.3 User Support

Any additional questions can be directed to admin@deku.com. Make sure to check this e-mail often to respond to questions that may appear in the inbox.

4. **Troubleshooting**

4.1 Dealing with Error Messages and Failures

In case of software failure, the following methods may help provide insight to the problem.

Add the following lines to deku.wsgi (added lines are prefixed with +++):

```
activate_this = '/home/vagrant/.venv/bin/activate_this.py'
execfile(activate_this, dict(__file__=activate_this))

import sys
+++import logging
sys.path.insert(0, '/home/vagrant/Section-2-Team-6')
from app import app as application
+++application.debug=True
import app
app.db.create_all()

+++logging.basicConfig(stream=sys.stderr)
```

Restart the apache server

```
sudo service apache2 restart
```

HTTP requests made to the Dekū API will be logged in `/home/vagrant/logs/access.log`. Exceptions thrown by the API will be logged in `/home/vagrant/logs/error.log`.

4.2 Known Bugs and Limitations

Bugs:
At this exact point in time, there is no known unexpected behavior.

Limitations:
The Dekū web application has not yet been secured properly. While the application does not have any known bugs between the front and back ends, the application is vulnerable to malicious attacks on the API. (ie. cURL calls on API methods that do not authenticate the user)

Appendix A – Team Review Sign-off

Appendix B – Document Contributions