|  |  |
| --- | --- |
|  | **2012** |
|  | Andrew Michaelis, Caleb Drake, Brandon Knight, Jackson Melling |

|  |
| --- |
| **CodeSwap DeploYment Plan** |
|  |

Table of Contents

[1. Installing the application 2](#_Toc340057809)

[2. Enabling Twitter Authentication 2](#_Toc340057810)

[3. Generating a Self-Signed Certificate 2](#_Toc340057811)

[4. Using an SSL Certificate 2](#_Toc340057812)

[References 3](#_Toc340057813)

## 1. Installing the application

1. Install Git

sudo apt-get install git

2. Download the application

git clone https://github.com/alexlo03/CodeSwap.git

3. Navigate to the installation directory

cd CodeSwap/Installer

4. Run the installer

chmod 777 Installer.sh && ./Installer.sh

## 2. Enabling Twitter Authentication

1. [Sign up](https://dev.twitter.com/apps/new) for a Twitter application

2. In /src/config/initializers/openid.rb, add the following before the last line, replacing Consumer\_Key and Consumer\_Secret with the corresponding values supplied by Twitter:

provider :twitter, ‘Consumer\_Key’, ‘Consumer\_Secret’

## 3. Generating a Self-Signed Certificate

mkdir /etc/ssl/self-signed && cd /etc/ssl/self-signed

openssl genrsa -des3 -out server.key 1024

openssl rsa -in server.key -out server.key.insecure

mv server.key server.key.secure && mv server.key.insecure server.key

openssl req -new -key server.key -out server.csr

openssl x509 -req -days 365 -in server.csr -signkey server.key -out server.crt

Script provided by Andrew Waage on A Waage Blog [1]

## 4. Using an SSL Certificate

1. Place the .crt, .csr, and .key files into CodeSwap/src/config/cert/ss

2. In /CodeSwap/src/script/secure\_rails, replace the references to codeswap.\* files on line numbers 22 and 24 with the names of the appropriate SSL files.

## 5. Setting up the Emailer

1. Acquire access to an email account on an SMTP server (ex. Gmail)

2. Open CodeSwap/src/config/environments/development.rb

3. Fill out the following code example with your relevant SMTP server/account settings. (Note: the example below already works for the gmail server).

config.action\_mailer.smtp\_settings = {

address: "smtp.gmail.com",

port: 587,

domain: "EXAMPLE.COM",

authentication: "plain",

enable\_starttls\_auto: true,

user\_name: “GMAIL\_USERNAME”,

password: "GMAIL\_PASSWORD”

}

4. In the same file (development.rb), scroll down to the following line:

config.action\_mailer.default\_url\_options = {

:host => 'localhost:3000' }

5. Replace ‘localhost:3000’ with the root dns or ip of your server.

## 6. Setting up the Server

1. Open CodeSwap/src/db/seeds.rb

2. Clear out the file

3. Fill out the following code example with an admin’s information and password. This ` will be the only initial account on the system.

User.create(:email => 'email@example.com',

:first\_name => 'Bob',

:last\_name => 'Ross',

:password => 'password',

:password\_confirmation => 'password',

:role => :admin)

4. Repeat step 3 for each admin you wish to have on the database initially. (It is very easy to add them later through the admin homepage)

5. Migrate and seed the database:

rake db:migrate && rake db:reset

## 7. Launching the Application

1. Clear any processes running on port 80 and 443.

2. Run the following command:

cd /CodeSwap/src/script/ && chmod 777 start\_server.sh && ./start\_server.sh

## References

1. <http://qugstart.com/blog/linux/quickest-way-to-create-a-self-signed-ssl-certificate-in-ubuntu/>