CCE Search Frontend System Information and Provisioning Guide

Instance ID: i-0fb314b098444b089

Availability Zone: us-east-2b

Public DNS: ec2-3-16-54-35.us-east-2.compute.amazonaws.com

Elastic IP: 3.15.80.81

URL: cce.ebookfoundation.org

OS: Ubuntu 18.04

UFW (firewall):

- Disallowed incoming by default
- Allowed outgoing by default
- Allowed SSH, HTTP, HTTPS, NginxHTTP, OpenSSH
- Enabled UFW **after** setting up rules
- Info here

Added apt repositories:

- universe (may be enabled by default)
- ppa:certbot/certbot
- ppa:deadsnakes/ppa

  Note: remember to run sudo apt update after adding

Installing Python 3.7 and Pip3:

While Ubuntu 18.04 comes with Python 3.6, adding the deadsnakes PPA allows for the installation of Python 3.7. However, linking the python3 command to Python 3.7 may cause issues with applications that use `#!/usr/bin/python3`. See the errors section of this document for information about fixing these issues. To counteract this, use the python3.7 command instead, and perform pip installs with python3.7 -m pip install x, rather than pip3 install x.

Installed apt packages:

- git (may be installed by default)
- nginx

- software-properties-common (may be installed by default, prerequisite for installing python3.7)
- python3.7
- python3-pip
- certbot
- python-certbot-nginx

Installed Pip3 packages:

- pipenv (python3.7 -m pip install pipenv)

Nginx (web server):

- Set up as reverse proxy for app, running on port 80
    - Use NGINX as a Reverse Proxy
    - https://gunicorn.org/#deployment (we don't use gunicorn but this is the basis of the configuration)
- The following config file should be saved as `/etc/nginx/conf.d/cce-search.conf`:

    ```
    server {
      listen 80;
      server_name cce.ebookfoundation.org;
      access_log /var/log/nginx/cce-search.log;

      location / {
        proxy_pass http://127.0.0.1:5000;
      }
    }
    ```
- Nginx is automatically configured to run on server reboot
- See here

Certbot (Let's Encrypt/SSL tool):

- See here
- Get cert and configure Nginx for HTTPS with `sudo certbot --nginx`
    - Email is support@ebookfoundation.org
    - Choose to redirect HTTP traffic to HTTPS
- Certificate and chain saved at: `/etc/letsencrypt/live/cce.ebookfoundation.org/fullchain.pem`
- Key file saved at: `/etc/letsencrypt/live/cce.ebookfoundation.org/privkey.pem`
- Cert should automatically renew

Setting up the application to run as a service:

- Save the following (source) in `/etc/systemd/system/cce-search-frontend.service`

  ```
  [Unit]
  Description=The CCE Search Flask application (cce.ebookfoundation.org)
  After=network.target

  [Service]
  User=ubuntu
  WorkingDirectory=/home/ubuntu/cce-search-frontend
  ExecStart=/home/ubuntu/.local/bin/pipenv run flask run
  Restart=always

  [Install]
  WantedBy=multi-user.target
  ```

- Run `sudo systemctl daemon-reload` to load the service

Deploy the latest source code with Ansible:

- Install Ansible, then use the update-frontend-prod playbook here to deploy the application once the server is provisioned

Errors and how to fix them

Running `add-apt-repository` returns `ImportError: cannot import name '_gi' from 'gi' (/usr/lib/python3/dist-packages/gi/__init__.py)` (or something similar):

- This is caused because the `python3` command runs a different version (3.7) than the default system version
- To fix, edit the first line of `/usr/bin/add-apt-repository` to say `#!/usr/bin/python3.6`
- See here: Changed Python version, "apt-add-repository" not working ERROR

Running `sudo certbot --nginx` returns `ModuleNotFoundError: No module named '_cffi_backend'`:

- Need to install cffi

- `sudo python3 -m pip install cffi`
- See No module named _cffi_backend

Going to the IP Address in your browser shows the app, but going to the Public DNS shows the default Nginx page:

- Edit the `server_name` in `/etc/nginx/conf.d/cce-search.conf` (remember to sudo)
- Test modifications with `sudo nginx -t`
- If accepted, `sudo systemctl restart nginx`
- See Using Amazon EC2: IP address working fine, but public DNS not
- Note: I currently have Nginx set up to serve to the IP instead of the public DNS, because of some issues with the overly long address

Locked yourself out of the server because you disallowed ssh connections in ufw and then logged off

- Hahaha
- Locked myself out of SSH with UFW in EC2 AWS