⚡

REBUILD

Firewall

- Open firewall ports to allow external access

  - `sudo ufw allow 80`

nginx

- Installed nginx on three staging servers

  - `sudo apt install nginx`

- Configure nginx "sites-available"

  - `<server blocks>`

- Remove default nginx config

  - `sudo rm -rf /etc/nginx/sites-enabled/default`

- Symlink Sites Available to Sites Enabled

  - `cd /etc/nginx/sites-enabled && sudo ln -s ../etc/nginx/sites-available/staging.neonet.org`

- Restart NGINX

  - `sudo systemctl restart nginx`

SSHD

- Create a SSHD conf file to let VSCode connect

  - `sudo nano /etc/ssh/sshd_config.d/vscode.conf`

- Add line to vscode.conf

  - `AllowTcpForwarding yes`

- Restart SSHD

  - `sudo systemctl restart sshd`

Service Account

- Create "website" user as dedicated service account

  - `sudo adduser website`

- Change /var/www app directories to be owned by "website"

  - `sudo chown -R website:website /var/www/ <app-dir>`

- Switch to service account user

  - `sudo su - website`

Node Libraries

Install libraries as service account

- Install NVM (Node Version Manager)

  - `curl -o-` https://raw.githubusercontent.com/nvm-sh/nvm/v0.40.1/install.sh `| bash`

- Source NVM instead of rebooting

  - `\. "$HOME/.nvm/nvm.sh"`

- Install latest stable version of node

  - `nvm install --lts`

- Upgrade NPM to latest

- nvm install-latest-npm
- Update .npmrc with PAT and Github Packages Repository info

  - `nano ~/.npmrc`

  -

    ```
    1  //npm.pkg.github.com/:_authToken=<github-pat-token>
    2  @neonetoh:registry=https://npm.pkg.github.com/
    3  registry=https://registry.npmjs.org
    4
    5
    ```

Github

Personal

- Generate GitHub SSH Keys

  - `ssh-keygen -t ed25519 -C "your_email@example.com"`

- Add SSH Key to Github

  - `cat ~/.ssh/id_github.pub`

*Website (Service Account User)

- Generate GitHub SSH Deply Keys (for each repository)

  - `ssh-keygen -t ed25519`

- Copy SSH Key

  - `cat ~/.ssh/id_github.pub`

- Add to GitHub Repos (Individually)

  - Go to -> <repo>

- Click -> "Settings"

- Click -> "Deploy Keys"

- Add SSH public key

Deploy/Rebuild Scripts

- Make Executable

  - `sudo chmod +x /opt/scripts/deploy.sh`

- Change Ownership

  - `sudo chown website: website /opt/scripts/deploy.sh`

- Change Permissions

  - `chmod 750 /opt/scripts/deploy.sh`

- Create .env files and fill in environment variables

DB/API SERVER ONLY

- Install Nest JS Global

- Install Libreoffice

- Install Puppeteers

- Add app armor profile for puppeteer

- Install Docker

  - `sudo install -m 0755 -d /etc/apt/keyrings`

  - `sudo curl -fsSL` https://download.docker.com/linux/ubuntu/gpg `-o /etc/apt/keyrings/docker.asc`

- sudo chmod a+r /etc/apt/keyrings/docker.asc

-

```
1  echo \
2    "deb [arch=$(dpkg --print-architecture) signed-
   by=/etc/apt/keyrings/docker.asc]
   https://download.docker.com/linux/ubuntu \
3    $(. /etc/os-release && echo
   "${UBUNTU_CODENAME:-$VERSION_CODENAME}") stable" | \
4    sudo tee /etc/apt/sources.list.d/docker.list >
   /dev/null
```

- echo "deb [arch=$(dpkg --print-architecture) signed-
  by=/etc/apt/keyrings/docker.asc]
  https://download.docker.com/linux/ubuntu $(. /etc/os-
  release && echo "${UBUNTU_CODENAME:-$VERSION_CODENAME}")
  stable" | sudo tee /etc/apt/sources.list.d/docker.list >
  /dev/null

- sudo apt install docker-ce docker-ce-cli containerd.io