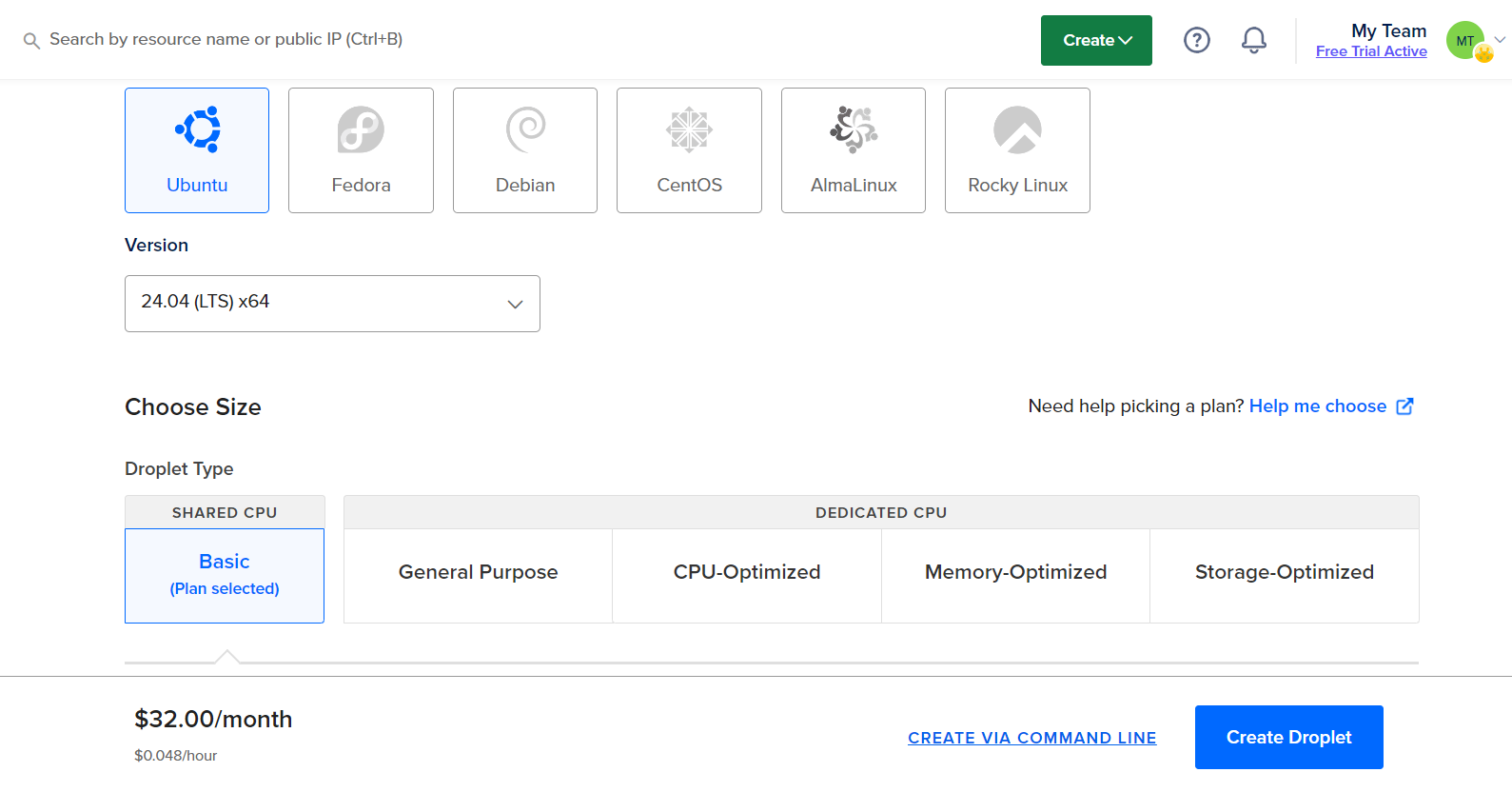
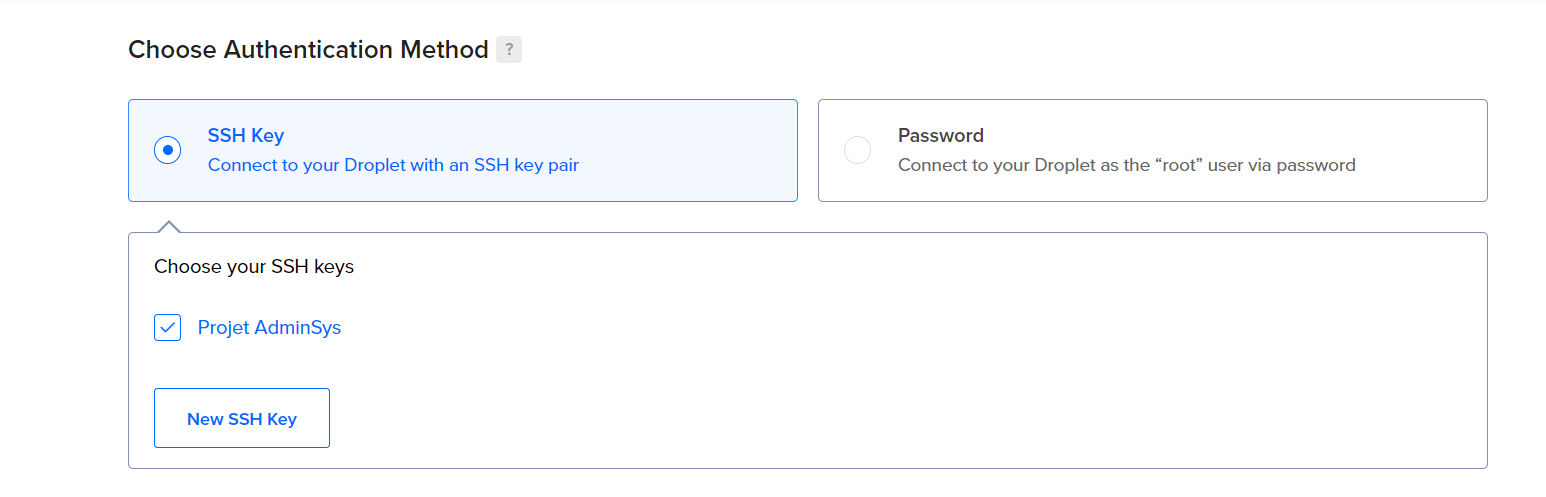
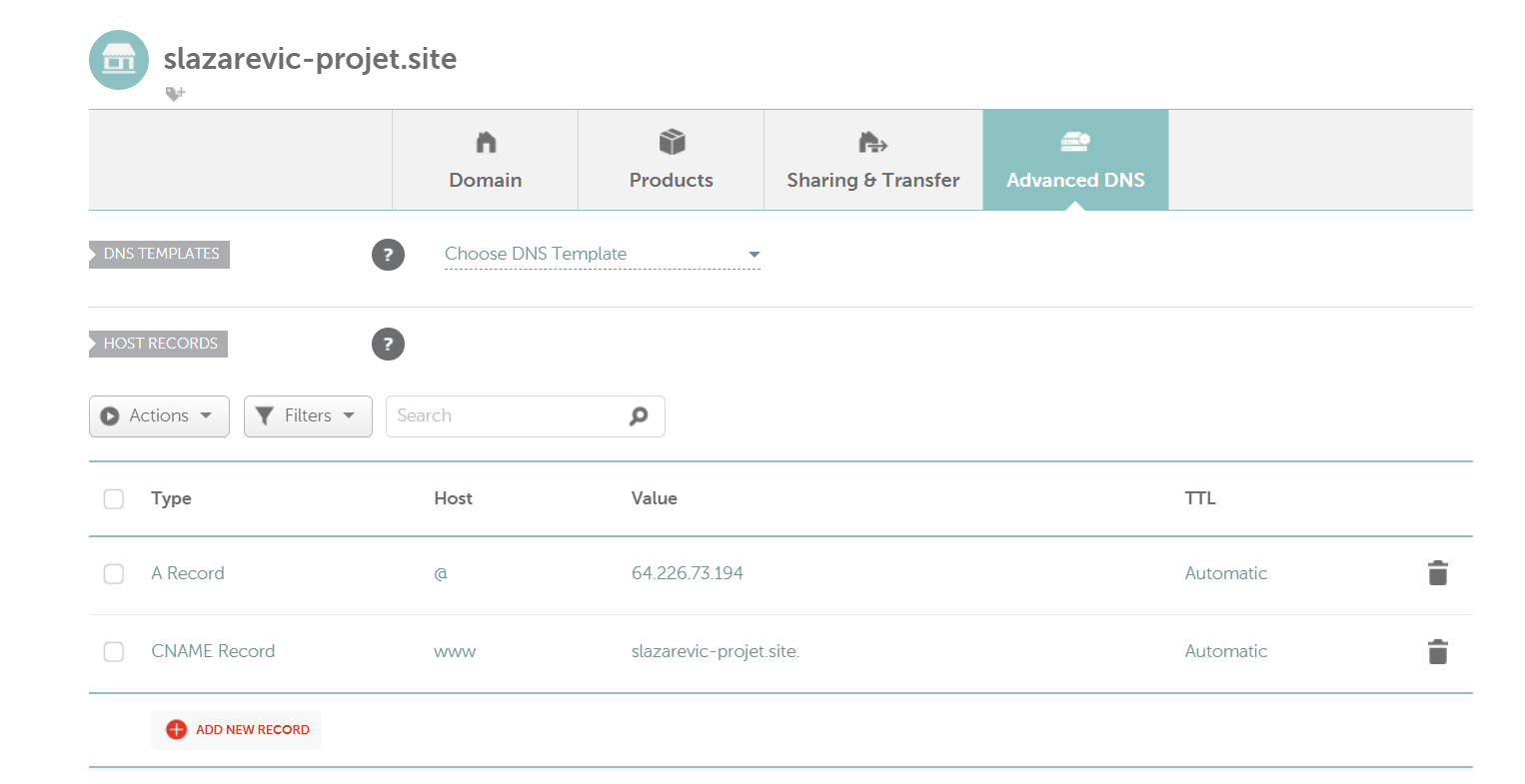
**Présentation technique projet Administration et sécurisation de serveur web**

Ce document vise à détailler les pratiques utilisées et les techniques mise en place dans la création du serveur, sa confiugration et l’hébergement d’une application via ce dernier.

Lien : https://slazarevic-projet.site/  
  
  
**1- Création du serveur et initialisation du nom de domaine**

Tout d’abord, j’ai créé une machine virtuelle sur DigitalOcean, avec la configuration dont j’avais besoin, à laquelle j’ai ajouté une clef SSH, que je pourrais utiliser par la suite pour communiquer avec mon serveur.  
  




Ensuite je me suis rendu sur NameCheap, afin d’acheter le nom de domaine que je souhaitais : <https://slazarevic-projet.site> puis dans la section gestion du nom de domaine (manage), dans l’onglet ‘Advanced DNS’ j’ai ajouté le A Record, et le CNAME Record afin de faire le lien entre l’adresse ip du serveur et l’adresse dns sur lequel il faudra taper pour le contacter   
  


**2- Configuration du serveur**

Ensuite j’ai lancé la console du serveur, et j’ai réalisé tout le reste du projet directement en ligne de commande.

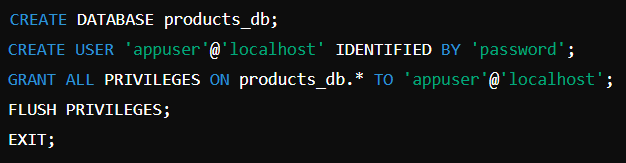
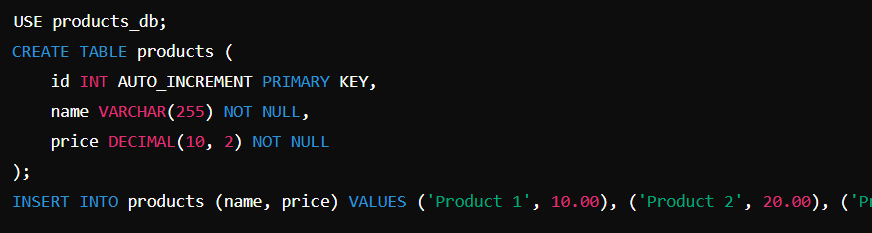
Une fois le Droplet crée, on peut se connecter directement comme ceci :  
- ssh root@your\_droplet\_ip (adresse ip du serveur)

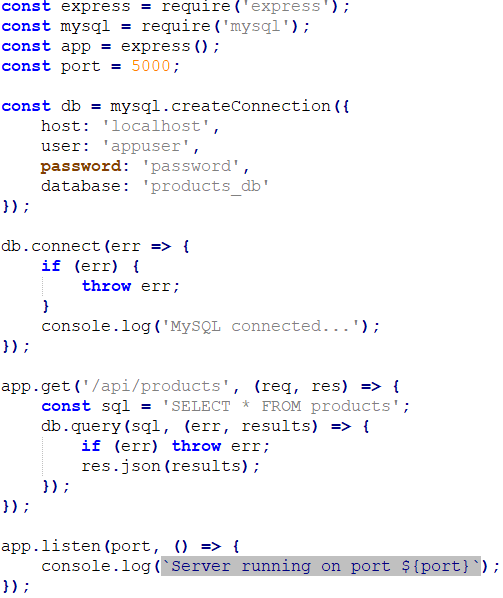
On commence par mettre à jour le serveur :  
- sudo apt update && sudo apt upgrade -y

Puis j’ai édité le fichier /etc/ssh/sshd\_config afin de désactiver l’accès root et les connexions par mot de passe

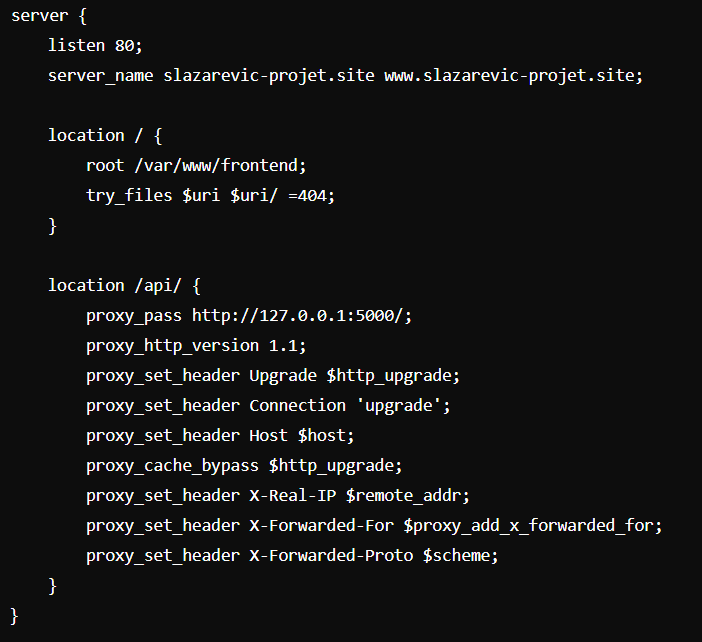
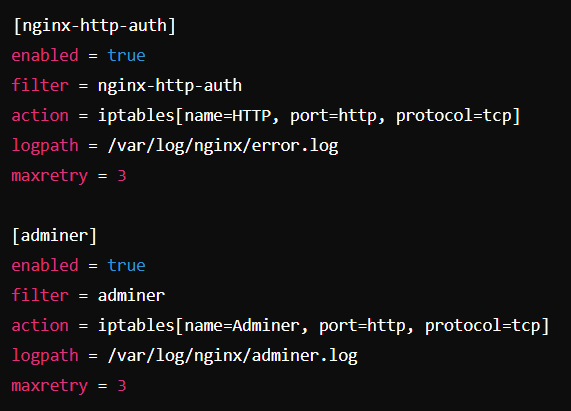
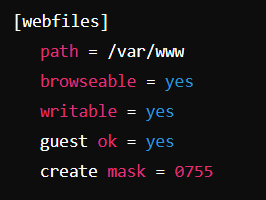
On fait   
- sudo nano /etc/ssh/sshd\_config   
puis on modifie le fichier pour avoir :  
- PermitRootLogin no  
- PasswordAuthentication no  
  
Puis j’ai redémarré le service ssh :   
- sudo systemctl restart sshd  
  
  
Ensuite on installe configure nginx :  
- sudo apt install nginx -y  
- sudo systemctl start nginx  
- sudo systemctl enable nginx

On installe et configure Node.js :  
- sudo apt install nodejs -y  
- sudo npm install pm2@latest -g

On installe et configure MySQL :  
- sudo apt install mysql-server -y  
- sudo mysql\_secure\_installation  
  
  
On créer la base de données et l’utilisateur :  
- sudo mysql -u root -p  
  
Et on exécute nos commandes SQL :  
  
  
  


Par la suite j’ai créé mes répertoires, avec mes dossiers frontend et backend contenant mes fichiers de code, je ne vais pas détailler cette étape car peu intéressante finalement, voici mon code :  
  
server.js :  
  
Par la suite j’ai remplacé mon host pour l’adresse ip sql

index.html :  


J’ai également installé les dépendances nécessaires quand / s’il le fallait.  
  
On créer ensuite le fichier de configuration nginx :  
- sudo nano /etc/nginx/sites-available/slazarevic-projet.conf  
  
Et on y ajoute notre configuration :  
  
  
  
  
Puis on active le site et on redémarre Nginx :  
- sudo ln -s /etc/nginx/sites-available/slazarevic-projet.conf /etc/nginx/sites-enabled/  
- sudo nginx -t  
- sudo systemctl restart nginx  
  
Il peut parfois être nécessaire de désactiver le site par défaut du nginx :  
- sudo rm /etc/nginx/sites-enabled/default  
  
  
Ensuite on installe et on configure le Certbot :  
- sudo apt install certbot python3-certbot-nginx -y  
- sudo certbot --nginx -d slazarevic-projet.site -d [www.slazarevic-projet.site](http://www.slazarevic-projet.site)  
  
Le certbot va automatiquement ajouter des lignes de code dans nos fichiers de configuration  
  
Puis on continue avec le Fail2Ban :  
- sudo apt install fail2ban -y  
- sudo nano /etc/fail2ban/jail.local   
  
Et on rempli le fichier comme ceci :  
  
  
  
et on redémarre :  
- sudo systemctl restart fail2ban  
  
J’ai continué avec l’installation de Samba :  
- sudo apt install samba -y  
- sudo nano /etc/samba/smb.conf  
  
Et  :   
  
  
  
On restart également :  
  
- sudo systemctl restart smbd  
  
  
A ce stade, j’ai eu certains problèmes avec la lecture des fichiers, donc j’ai vérifié les permissions, ce qui m’a débloqué :  
- sudo chown -R www-data:www-data /var/www/frontend  
- sudo chmod -R 755 /var/www/frontend

**3 – Test de l’application**

Une fois tout ceci fait, j’ai bien start ou restart mes services, back, nginx, sql etc   
On peut également vérifier l’état de certains services pour voir s’ils tournent bien, par exemple :  
- sudo systemctl status mysql  
- sudo pm2 list  
  
Vérifier si Node.js écoute bien le port 5000 aussi :  
- sudo ss -tuln | grep 5000  
  
J’ai également checker les logs du backend :  
- sudo pm2 logs backend  
  
Une fois tout ceci fait mon site était bien en ligne et fonctionnel, j’ai juste eu un petit problème lorsque je me connecte à ma base de données SQL est bien présente, le back s’y connecte bien et j’arrive également à récupérer les données de ma table depuis le serveur, cependant sur la page d’accueil du site, les données SQL que je cherche à récupérer depuis l’API ne remonte pas, celai me fait la même chose quand je tente de faire un curl sur mon adresse SQL, une erreur 504 avec un Timeout est renvoyée.  
  
A l’exception de cela, je n’ai pas vraiment rencontré de problèmes avec le reste et la partie configuration du serveur et des outils souhaités.  
  
  
LAZAREVIC Stéphane, 4IW2.