



# TP Déploiement

Page • 1 backlink • Tag

## Étapes

- Création de la VM

Nom hôte: `tpdep`

- Demande de l'ouverture des ports 80 et 443, avec ticket support DSI
- Connexion en ssh

```
ssh finxol@tpdep.istic.univ-rennes1.fr
```

La clé ssh de mon ordinateur est correctement installée, donc pas de mdp demandé

- Mise en place ufw (firewall)

Shell ▾

```
# Install ufw
sudo apt install ufw
# Paramétrage de règles par défaut
sudo ufw default deny incoming
sudo ufw default allow outgoing
# Autorisation ssh
sudo ufw allow ssh
sudo ufw limit ssh
# Ouverture du port 80 pour le web (accès internet externe
indisponible donc https et port 443 inutile)
sudo ufw allow 80

# Activer le firewall
sudo ufw enable
```

Shell ▾

```
finxol@tpdep:/var/www/tp-react$ sudo ufw status numbered
Status: active
```

To	Action	From
--	-----	----
[ 1] OpenSSH	ALLOW IN	Anywhere
[ 2] 22/tcp	LIMIT IN	Anywhere
[ 3] 80	ALLOW IN	Anywhere
[ 4] OpenSSH (v6)	ALLOW IN	Anywhere (v6)
[ 5] 22/tcp (v6)	LIMIT IN	Anywhere (v6)
[ 6] 80 (v6)	ALLOW IN	Anywhere (v6)

- Installation du serveur Nginx

Shell ▾

```
sudo apt update && sudo apt install nginx
```

- Installation du projet

Shell ▾

```
# Install unzip et git
sudo apt install unzip git
# Install fnm
curl -fsSL https://fnm.vercel.app/install | bash
# Install Node
fnm install 22
# Install deps
corepack enable
pnpm i
# Build project
pnpm run build
```

- Configuration de Nginx

```
sudo vim /etc/nginx/sites-available/tpdep.istic.univ-
rennes1.fr
```

nginx ▾

```
server {
    listen 80;
    listen [::]:80;
    server_name tpdep.istic.univ-rennes1.fr;

    root /var/www/tp-react/build;
    index index.html;

    # Enable gzip compression
    gzip on;
    gzip_types text/plain text/css application/json
```

```

application/javascript text/xml application/xml
application/xml+rss text/javascript;
    gzip_vary on;

    location / {
        try_files $uri $uri/ /index.html;
    }

    access_log /var/log/nginx/access.log;
    error_log /var/log/nginx/error.log;
}

```

Activer le site avec

Shell ▾

```

sudo ln -s /etc/nginx/sites-available/tpdep.istic.univ-rennes1.fr
/etc/nginx/sites-enabled/
sudo nginx -t

```

Démarrer nginx avec

Shell ▾

```

sudo systemctl start nginx
sudo systemctl enable nginx

```

On peut également vérifier les logs de nginx avec `sudo systemctl status nginx`

- Installation et configuration de fail2ban

Shell ▾

```

# install fail2ban
sudo apt install fail2ban
# copy config
sudo cp /etc/fail2ban/jail.conf /etc/fail2ban/jail.local

```

`vim /etc/fail2ban/jail.d/nginx.conf`

TOML ▾

```

[nginx-http-auth]
enabled = true
port = http,https
logpath = /var/log/nginx/error.log

[nginx-limit-req]

```

```
enabled = true
port = http,https
logpath = /var/log/nginx/error.log
maxretry = 5
findtime = 600
bantime = 3600
```

### Démarrer fail2ban

Shell ▾

```
sudo systemctl restart nginx
sudo systemctl start fail2ban
sudo systemctl enable fail2ban
```

- Le site est maintenant disponible sur `http://tpdep.istic.univ-rennes1.fr` !

Comme l'application déployée est simplement une SPA qui ne nécessite pas de serveur node spécifique, la fonctionnalité de serveur statique de nginx est parfaitement suffisante.

Maintenant que la version HTTP fonctionne, on peut ajouter HTTPS

- Installer Certbot pour un certificat SSL Let's Encrypt gratuit

Shell ▾

```
sudo apt install certbot python3-certbot-nginx
```

- Configurer le certificat HTTPS

Shell ▾

```
sudo certbot --nginx -d tpdep.istic.univ-rennes1.fr
```

Il suffit de renseigner notre email étudiant pour recevoir les alertes importantes

Certbot s'occupe directement de modifier notre configuration nginx pour l'utilisation de https et la redirection automatique.

- Vérifier la configuration nginx et redémarrer

Shell ▾

```
sudo nginx -t
```

```
sudo systemctl reload nginx
```

- Ouverture du port 443 sur le firewall

Shell ▾

```
sudo ufw allow 443
```

Notre site est maintenant disponible en HTTPS sur <https://tpdep.istic.univ-rennes1.fr> !