Mission (SP1) : WIFI et serveur proxy

0. Pré-requis

Pour cette mission il faut un **humain mentalement équilibré** et quelques **équipements réseaux** (switch, routeur , hyperviseur, cable RJ45, **Access Point**).

1. Introduction

L'entreprise GSB, a récemment renforcé sa politique de recrutement en mettant en place des formations pour les nouveaux souhaitant intégrer l'entreprise.

GSB charge son **DSI** (Adem Lazri) de mettre en place plusieurs **réseaux wifi**, accessibles depuis un **portail captif** pour les personnes qui viennent pour **se faire former** et pour **les employés**. Le réseau Wifi doit être **sécurisé** de manière à éviter les cyberattaques.

Un **serveur Proxy** est également à mettre en place afin de contrôler les requêtes des utilisateurs du Wifi vers des sites **inappropriés**.

2. Rendre l'Access Point accessible

2.1 Configuration du switch

1. Création des vlans

Il nous faut donc **2 nouveaux réseaux(VLANS)** :

- Un VLAN WIFI-EMPLOYE

- Un VLAN WIFI-FORMATION

> **Scission du réseau **WIFI-GSB** (10.31.196.0/23)**
> 10.31.196.0/24 —> **WIFI-EMPLOYE**
>
> **Gateway** —> 10.31.196.254
>
> 10.31.197.0/24 —> **WIFI-FORMATION**
>
> **Gateway** —> 10.31.197.254

On se connecte à la **console** du **switch** à l'aide d'un **cable console cisco** puis on crée nos VLANs.

On passe en mode privilégié et en mode configuration globale :

```
Switch> enable #passage en mode privilégié
Switch# conf t #passer en mode configuration globale
```

On crée notre vlan WIFI-FORMATION :

```
Switch(config)# Vlan 197 #Crée le VLAN 197
Switch(config-vlan)# name WIFI-FORMATION #Nommer le VLAN "WIFI-FORMATION"
Switch(config-vlan)# exit #Sortir du mode VLAN
```

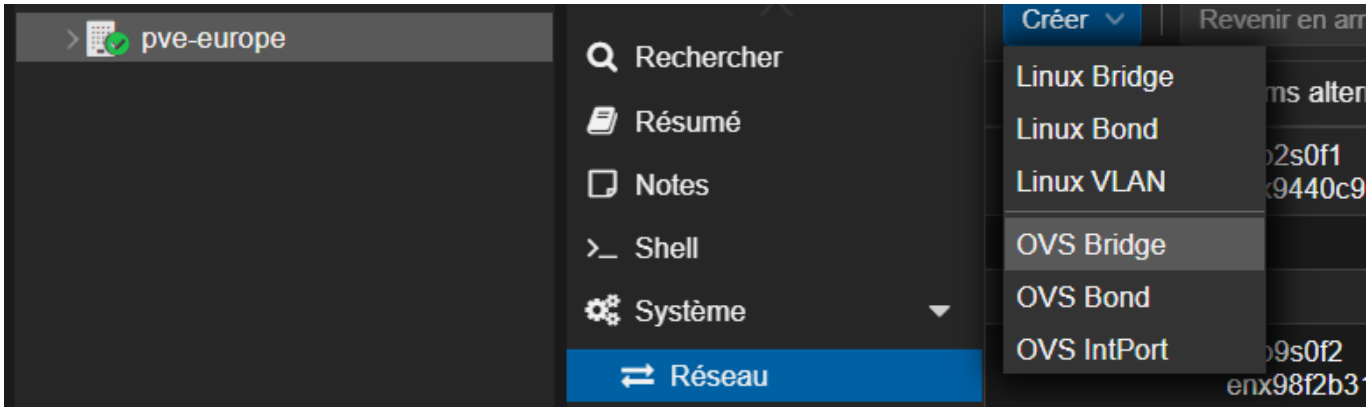
> **Important**
>
> Pour l'instant, **on ne touche pas** au vlan **198**, on expliquera plus loin pourquoi.

On **configure** le port **fa0/24** (port pour l'ACCESS POINT) :

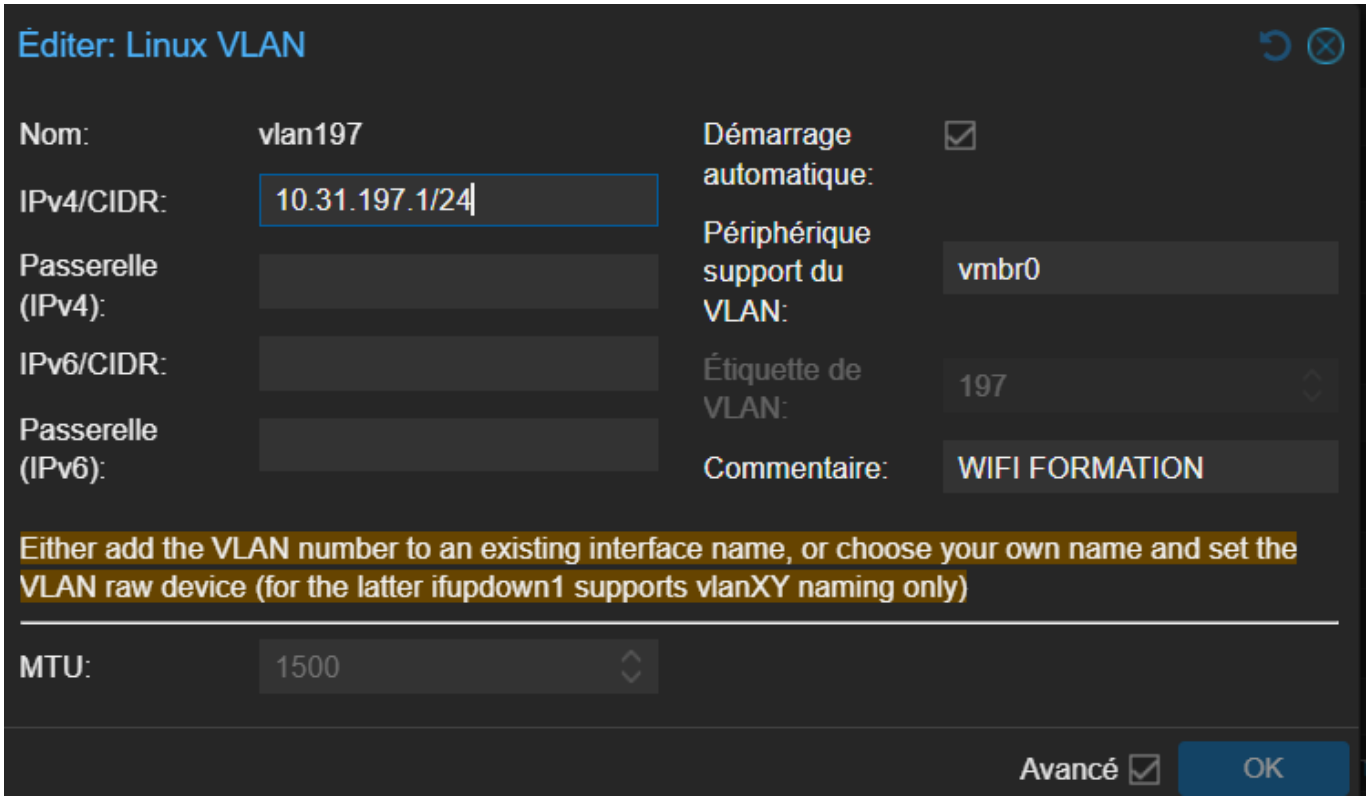
```
Switch(config)# Interface Fa0/24 #Sélectionne le port connecté à l'AP
Switch(config-if)# Switchport mode access #on met le port en access
Switch(config-if)# switchport access vlan 206 #on met le port sur le vlan
206 (management)
Switch> write memory #sauvegarde les modifications
```

2.2 Configuration du serveur ProxMox

On crée notre **vlan 197** sur ProxMox : **pve-europe —> réseau —→ Linux VLAN**

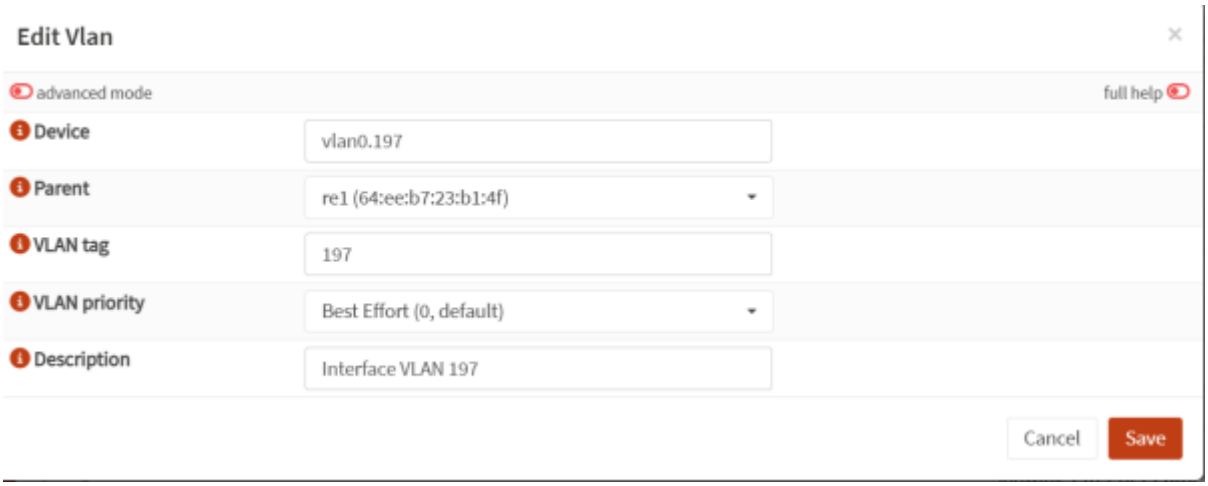


On configure le **VLAN** :

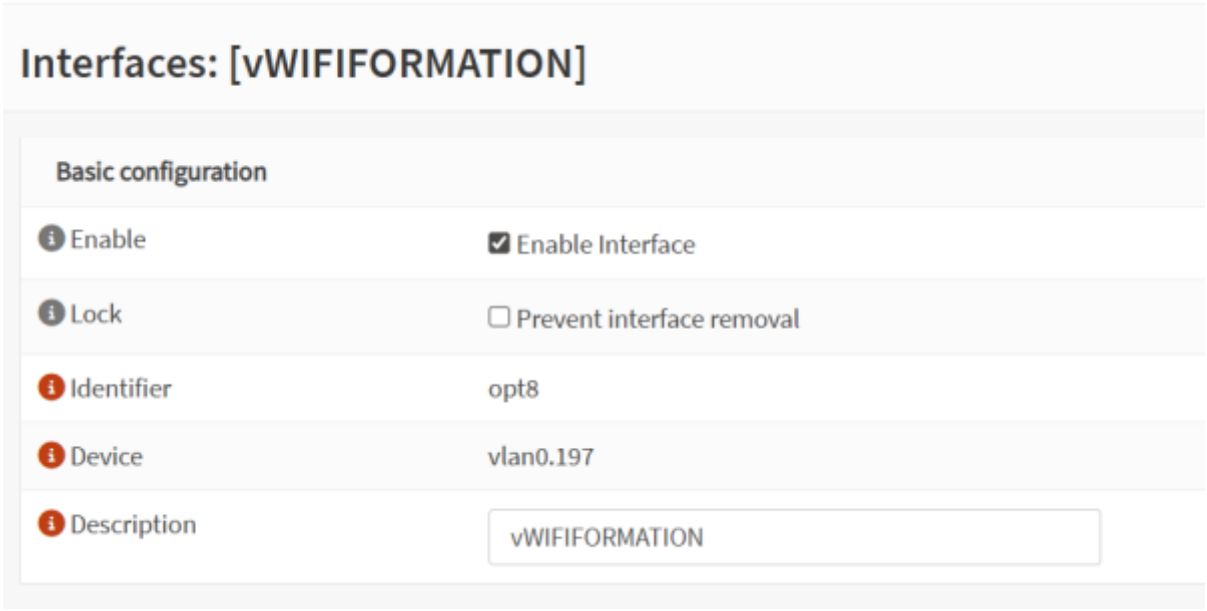


2.3 Configuration du routeur OPNSENSE

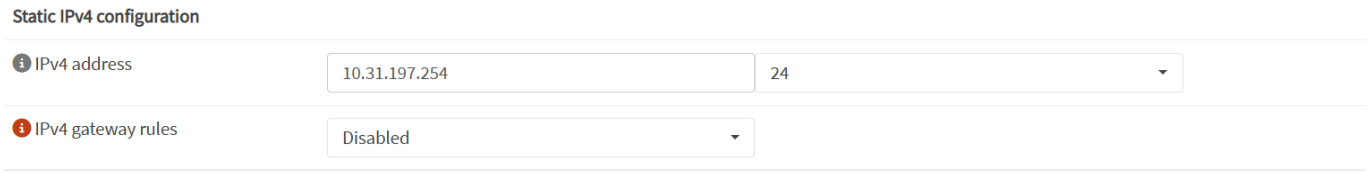
On se rend dans **Interfaces —→ Devices —→ VLAN** et on crée notre **vlan** 197 **(WIFI-FORMATION)** :



On l'assigne :



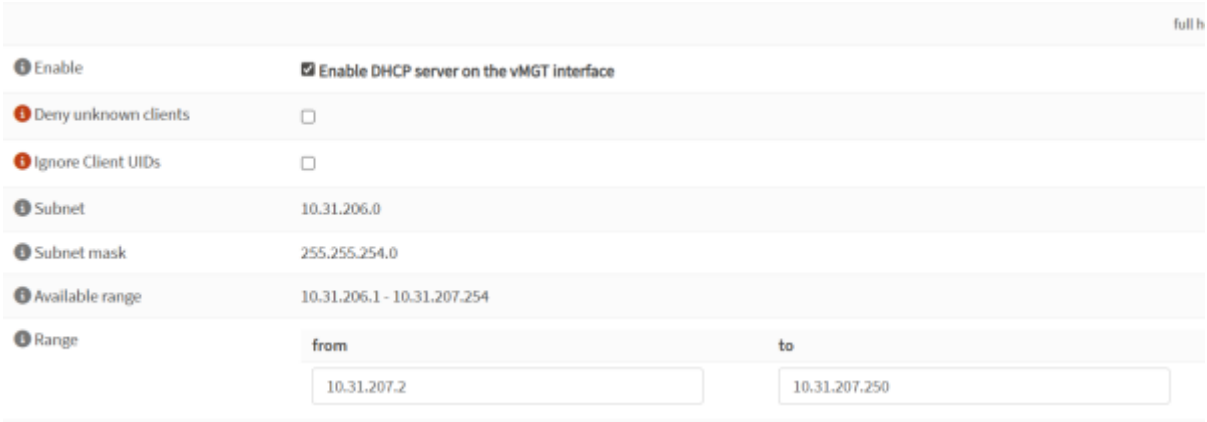
On lui configure son **adresse IP (dernière adresse du réseau)** et son masque :



Désormais, nous allons activer **l'option DHCP** sur l'interface **MGT du routeur OPNsense**, cela nous permettra d'attribuer **une adresse ip** à notre **Access Point** pour pouvoir y accéder sur son **interface via un navigateur.**

On se rend dans Services —→ ISC DHCP v4 —> vMGT

On active le **service DHCP** et définit la **plage d'adresse** attribuable :



On définit les bons **DNS** ainsi que la bonne **passerelle par défaut** :

DNS servers	10.31.200.53
	10.31.200.54
Gateway	10.31.207.254

Normalement, notre **DHCP** vient **d'attribuer une adresse IP** à notre **AP** sur **le vlan MGT (le vlan MGT (206) passe sur le port access qui relie l'AP au switch).**

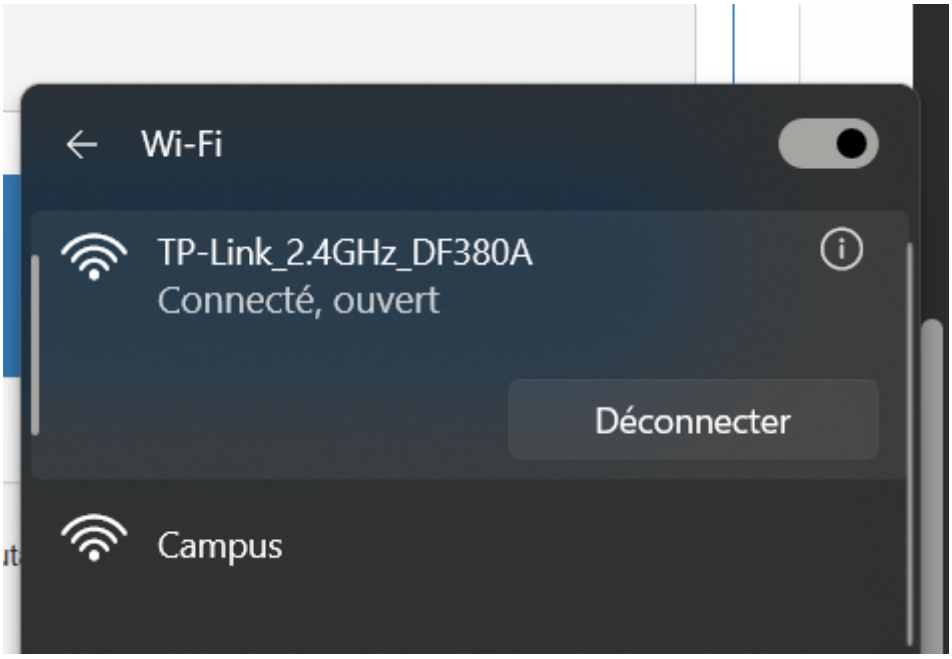
Nous **vérifions** cela dans les logs **DHCP** d'OPNsense :

vMGT	10.31.207.7	d8:44:89:df:38:0a *TP-LINK CORPORATION PTE. LTD.*	EAP110-D8-44-89-DF-38-0A		2025/12/08 16:36:46	2025/12/08 18:36:46		active	dynamic

Ici, on voit bien que notre **serveur DHCP** a attribuer **l'ip 10.31.207.5** à notre borne **Wifi/Access Point.**

On ne peut pas encore **accéder** à notre **AP** depuis une **machine** d'un autre **réseau** que le sien, on se connecte donc **au wifi** de la borne et on tape son **adresse IP** dans un **navigateur.**

Le **WIFI** de la borne apparaît **désormais** :

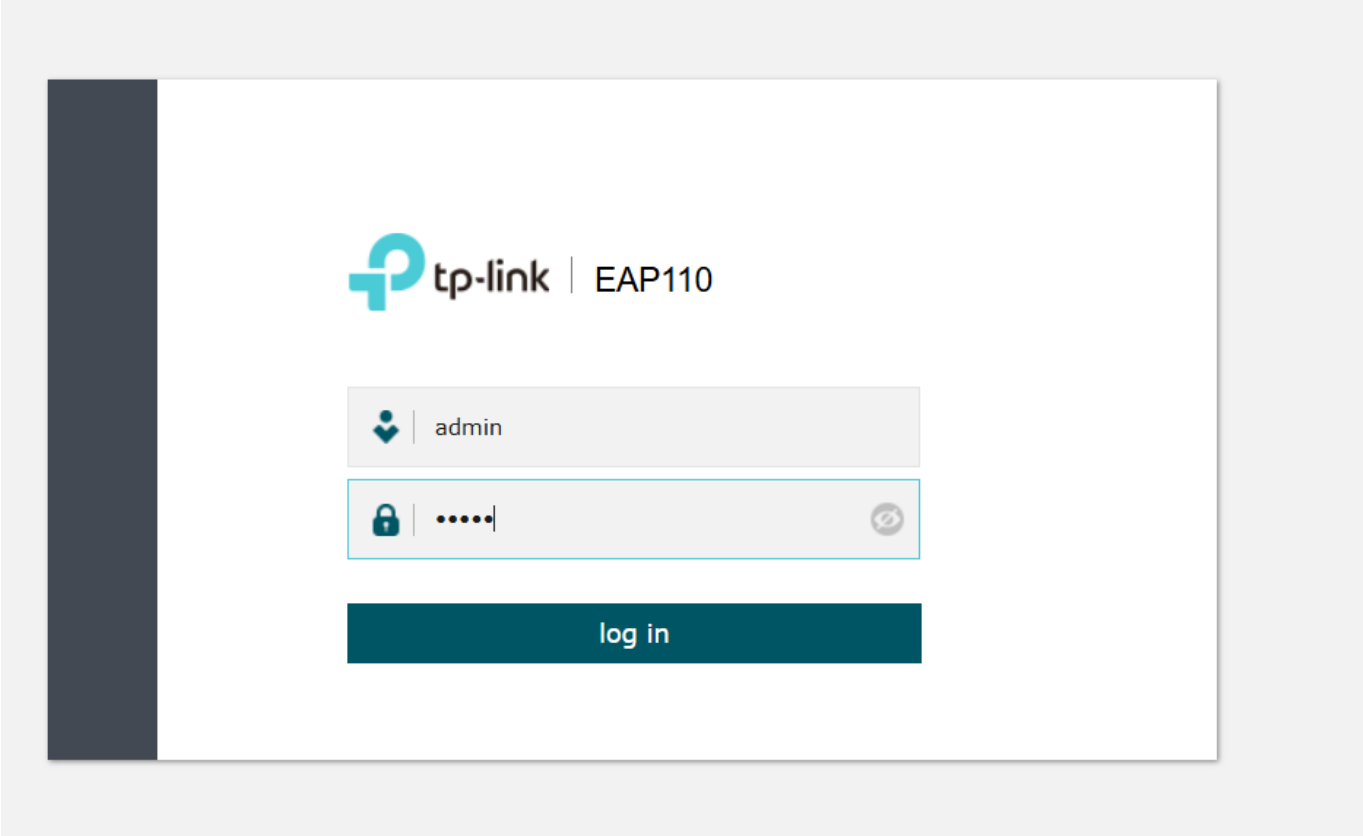


On s'y connecte puis on tape l'adresse IP de l'AP sur un navigateur : **10.31.207.7**

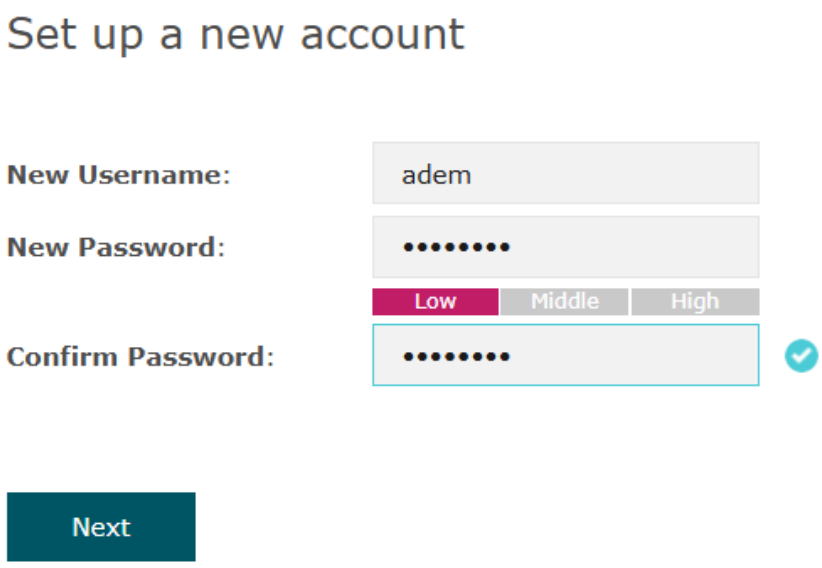
3. Configuration de l'interface TP-LINK EAP110

3.1 Première configuration

Une fois sur **l'interface web** du **TP LINK**, on se connecte en **Admin** :



On modifie l'utilisateur Admin :

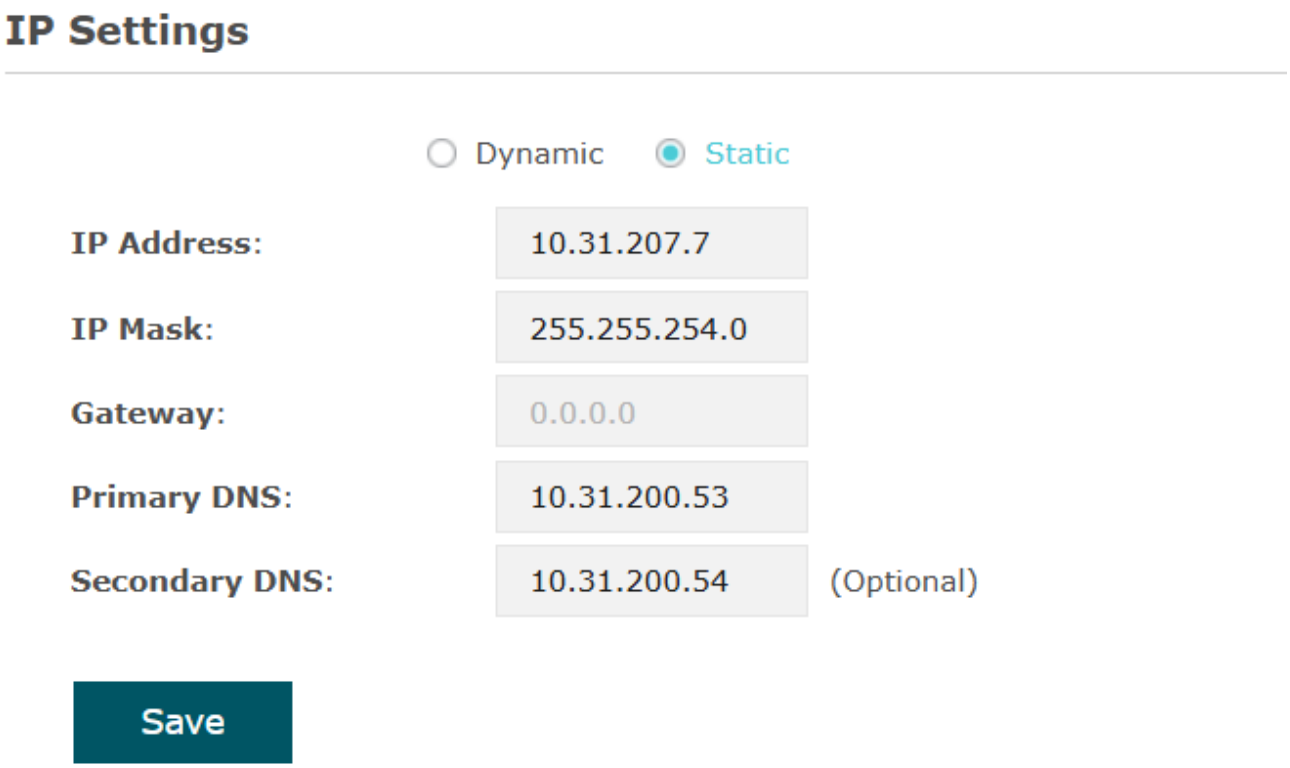


J'ai crée un **SSID “WIFI-EMPLOYE”** mais j'ai oublier de **screen** mais trkl **c'est pas compliqué**.

Voici notre premier SSID :

ID	SSID Name	Clients	Band	Security	Portal	VLAN ID	Guest Network	Down (Byte)	Up (Byte)
1	WIFI-EMPLOYE	1	2.4GHz	WPA-PSK	Disable	Disable	Disable	41124k	6059k

L'adresse IP de notre borne **est en DHCP**, on lui donne **une adresse Ipv4 static ainsi que nos DNS :**



3.2 Quelques fonctionnalités sur l'interface du TP-LINK

On va volontairement arrêter **la configuration de la borne depuis son interface** car à partir de maintenant, **nous utiliserons le controller Omada.**

Mais avant ça, découvrons, **par curiosité simplement,** ce que l'on peut faire depuis **l'interface WEB de la borne.**

B.2.1 Status

Le premier onglet Status permet d'afficher les infos sur la borne, sur les réseaux wifi diffusés, sur les clients connectés.

B.2.2 Wireless

L'onglet Wireless permet de réaliser la configuration des réseaux wifi

Wireless setting

Cette section permet de créer et configurer les SSID diffusés par la borne.

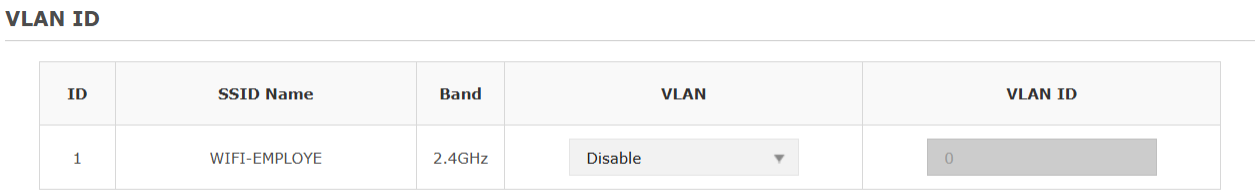
ID	SSID Name	Clients	Band	Security	Portal	VLAN ID	Guest Network	Down (Byte)	Up (Byte)
1	WIFI-EMPLOYE	1	2.4GHz	WPA-PSK	Disable	Disable	Disable	41124k	6059k

Portal

Cette section permet de créer un portail captif simpliste. On peut tester, mais on préférera le portail captif de OPNSense.

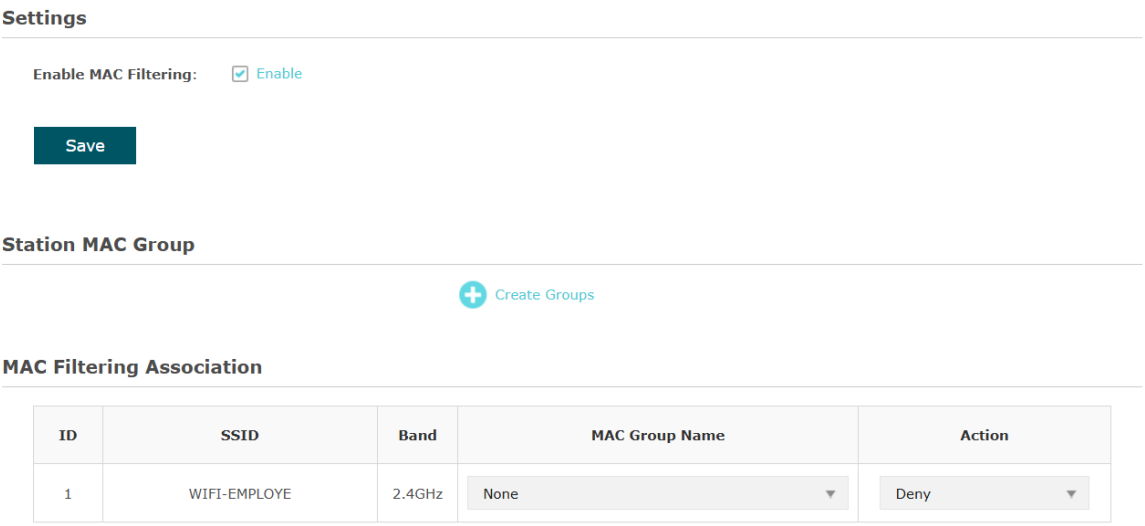
VLAN

Si la borne wifi est sur un port taggué, on peut associer un SSID avec un VLAN particulier.



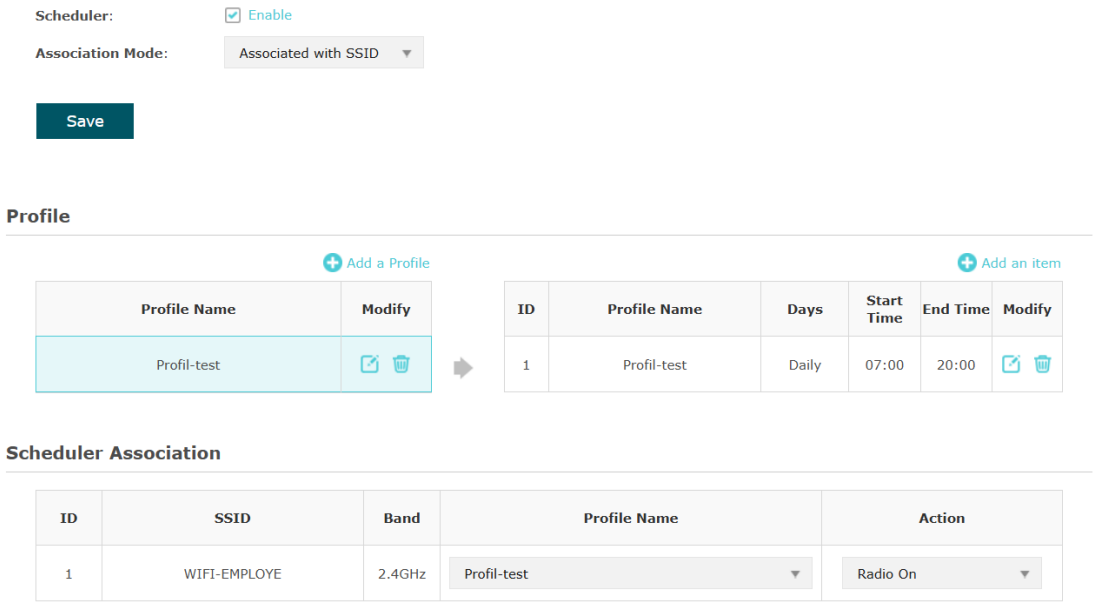
MAC Filtering

Si on veut empêcher des intrus de venir squatter notre Wifi, on peut créer des groupes d'adresses MAC afin d'appliquer une politique de filtrage.



Scheduler

La section Scheduler permet de gérer le démarrage ou l'arrêt de la borne ou de la diffusion des SSID en fonction d'une plage horaire.



Ici, les bornes diffuseront le SSID « WIFI-EMPLOYE » les jours de la semaine de 7h00 à 20h00.

B.2.3 Management & System

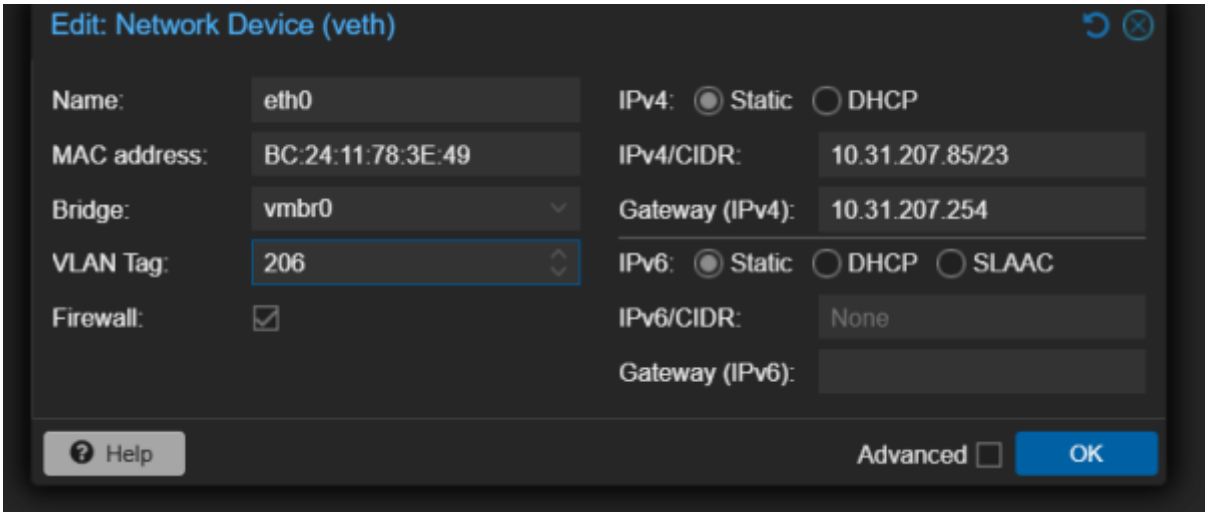
Les onglets **Management** et **System** permettent de configurer les paramètres propres à la borne Wi-Fi, parmi lesquels :

- L'adressage IP de la borne
- Les logs
- L'accès à la WebUI
- Les contrôles d'accès à la WebUI
- Le contrôle des LEDs
- L'accès SSH sur la borne
- L'activation de l'agent SNMP
- La gestion des comptes utilisateur
- La gestion du contrôleur
- La gestion des locales
- La gestion des backups
- etc.

4. Le controlleur Omada

4.1 Création et configuration du conteneur (2)

On crée une **vm "omada-mgt"**, voici sa **configuration réseau :**



Etant donné que **Debian 13 n'accepte plus les signature SHA1** car elles sont désormais considérées comme **cryptographiquement faible**, ça pose problème lors de l'installation de **mongoDB dans les dépôts.**

On va donc changer le programme utilisé pour **vérifier la signature en passant à un programme moins stricte :**

```
root@omada-mgt:~# apt update -o APT::Key::GPGVCommand=/usr/bin/gpgv
```

On met à jour la machine :

```
root@omada-mgt:~# apt update && apt upgrade -y
```

On installe Java :

```
root@omada-mgt:~# apt install openjdk-25-jre-headless
root@omada-mgt:~# apt install -y jsvc libcommons-daemon-java
```

On installe les dépendances :

```
root@omada-mgt:~# apt install -y wget curl gnupg2 sudo
```

On installe le depôt pour MongoDB (version 7.0) :

```
root@omada-mgt:~#  curl -fsSL
https://www.mongodb.org/static/pgp/server-7.0.asc | gpg -o
/usr/share/keyrings/mongodb-server-7.0.gpg --dearmor

root@omada-mgt:~# echo "deb [ signed-by=/usr/share/keyrings/mongodb-
server-7.0.gpg ] https://repo.mongodb.org/apt/debian bookworm/mongodb-
org/7.0 main" > /etc/apt/sources.list.d/mongodb-org-7.0.list
```

Mise à jour (avec bypass SHA1) :

```
root@omada-mgt:~# apt update -o APT::Key::GPGVCommand=/usr/bin/gpgv
```

On installe MongoDB :

```
root@omada-mgt:~# apt update
root@omada-mgt:~# apt install -y mongodb-org
root@omada-mgt:~# systemctl start mongod
root@omada-mgt:~# systemctl enable mongod
```

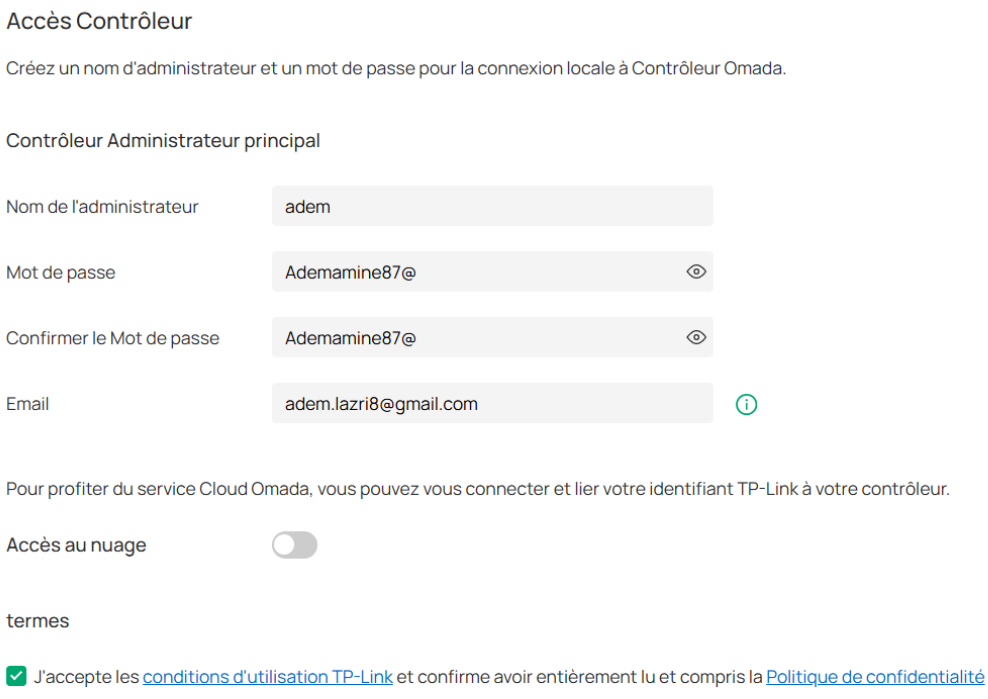
Téléchargement et installation d'Omada Controller :

```
root@omada-mgt:~# wget https://static.tp-link.com/upload/software/2025/202512/20251203/omada_v6.0.0.25_linux_x64_20251120205747.deb
root@omada-mgt:~# dpkg -i omada_v6.0.0.25_linux_x64_20251120205747.deb
```

4.2 Premiers pas sur l'interface Web de Omada

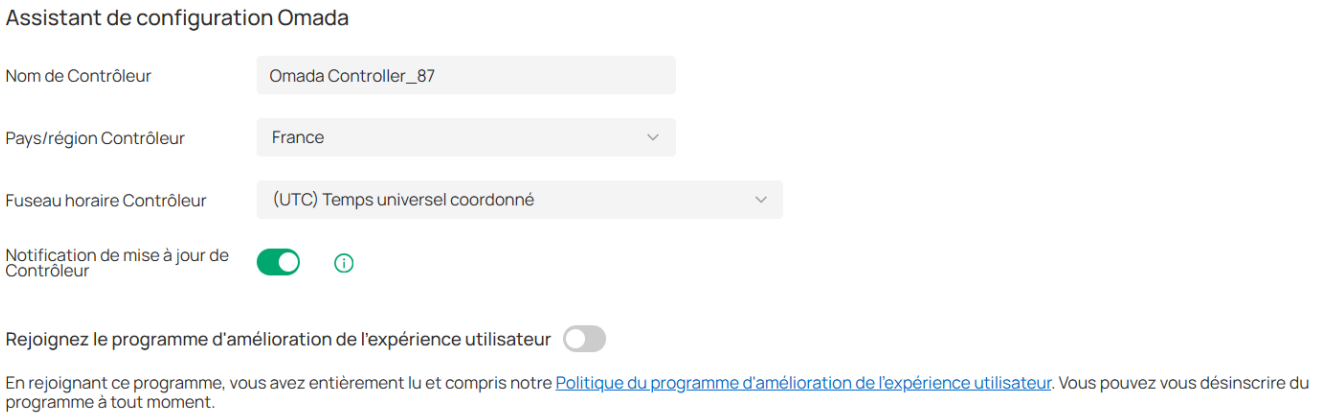
Afin d'accéder à **l'interface web d'omada** on tape **sur un navigateur** :
https://10.31.207.85:8043/

On crée un compte **administrateur** :

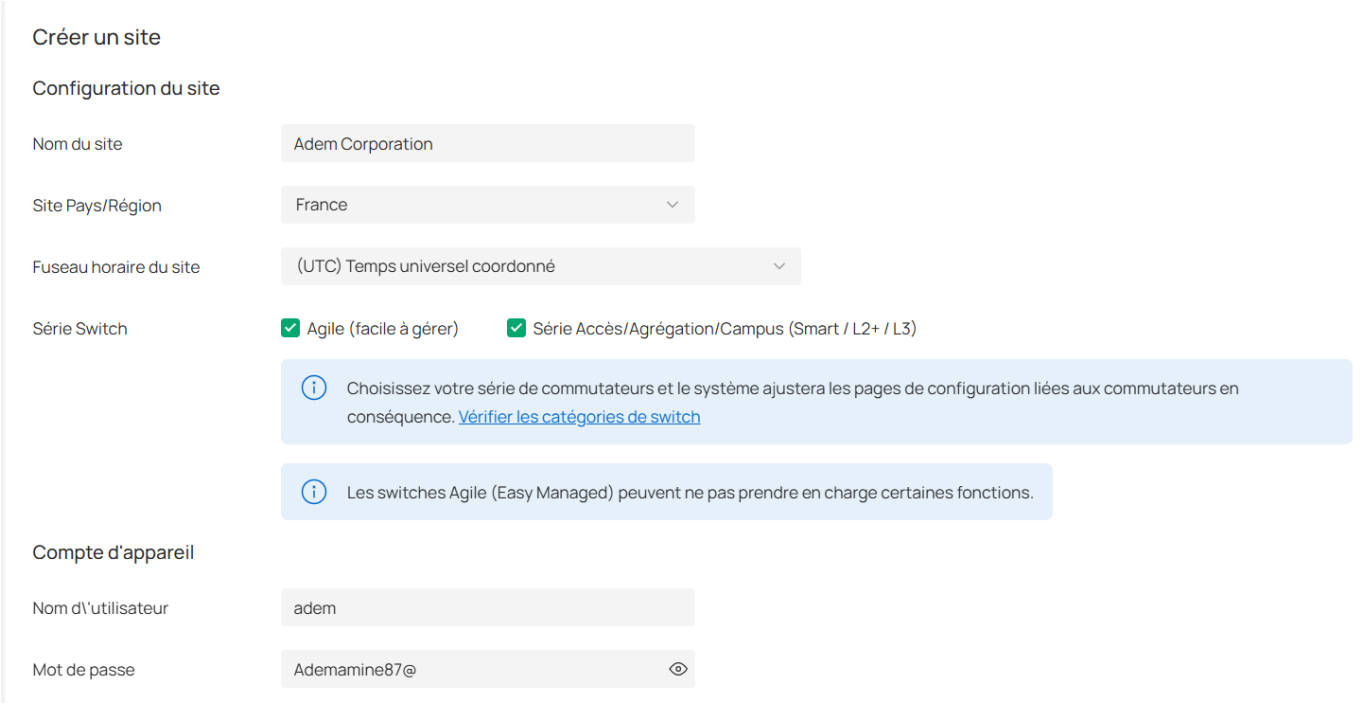


On configure un **nouveau Controller** :

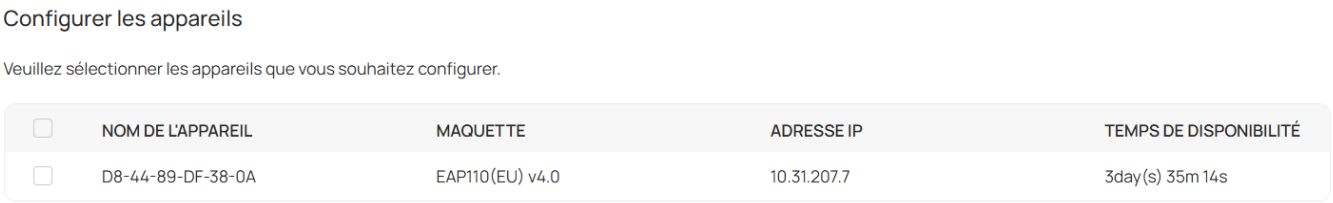
On définit un nom :



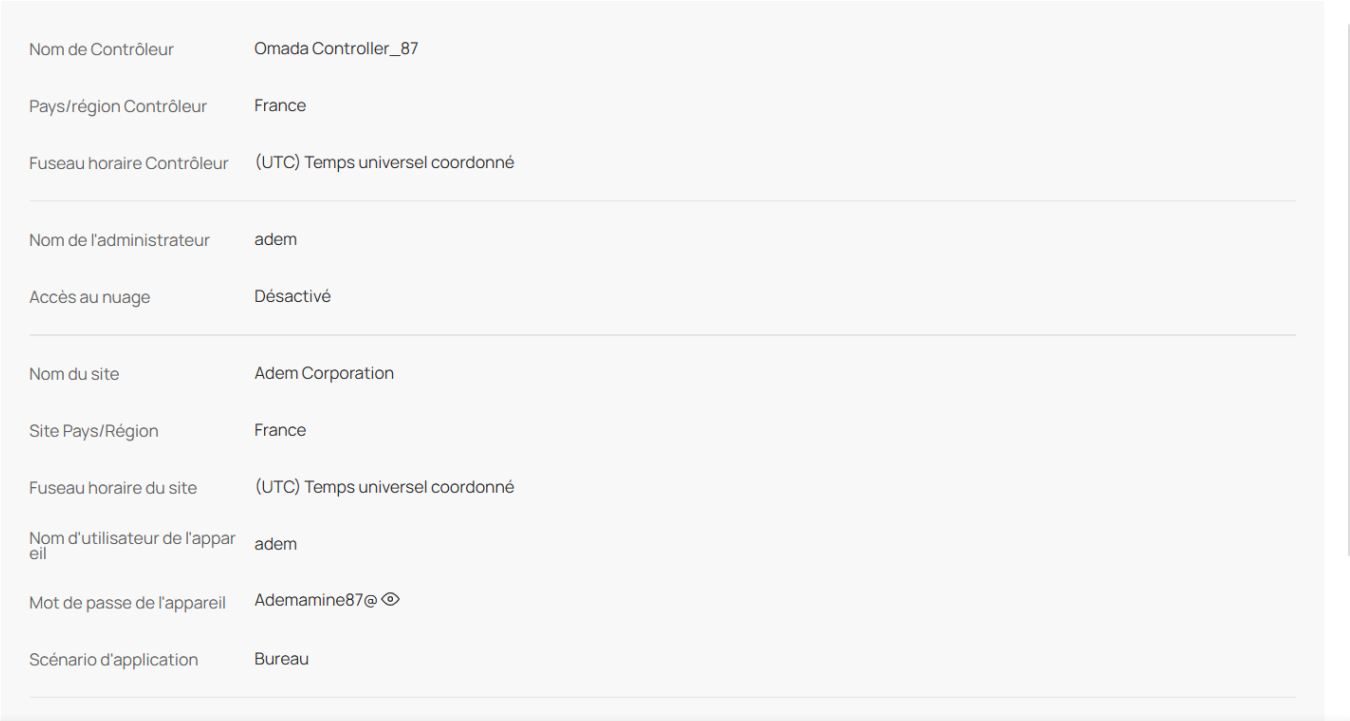
On configure le site :



Notre TP link à été détecté :



On confirme les données que l'on vient de rentrées et on valide :

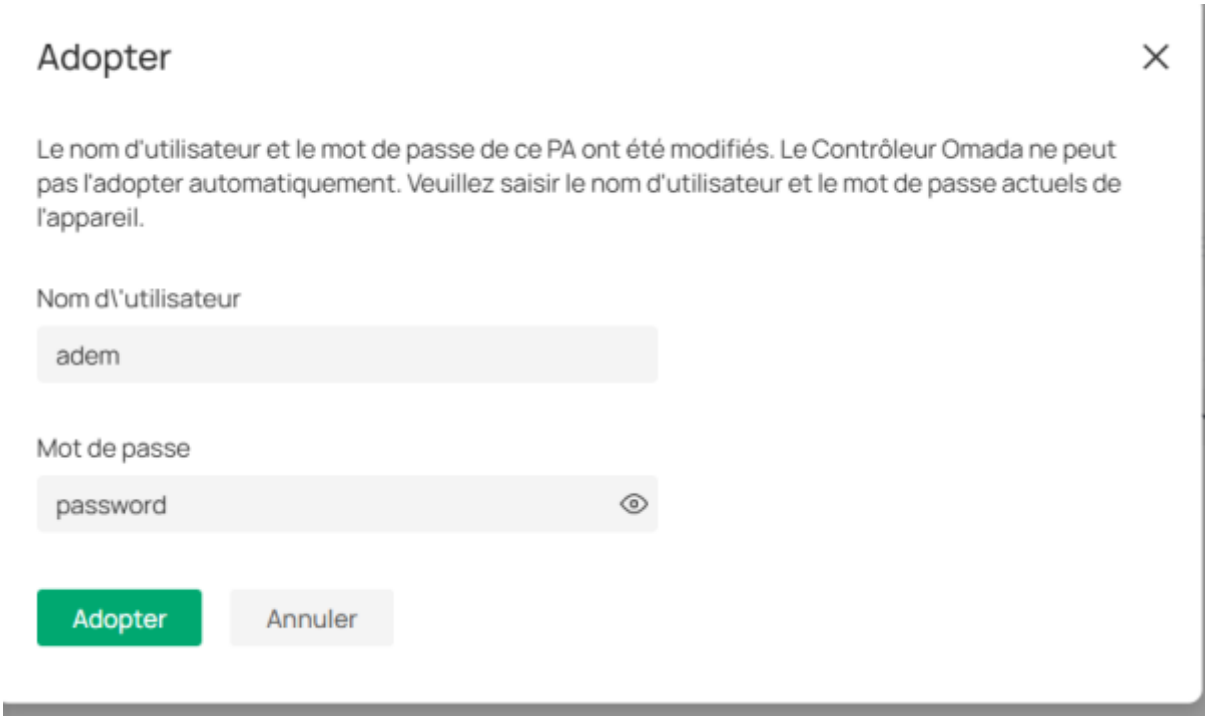


On se rend dans le site que nous venons de crée pour adopter notre appareil (AP)

Puis on se rend dans Appareil —→ adopter

Nous sommes invité à **rentrer des informations de connexions**, **il faut rentrer ceux de l'appareil que l'on souhaite adpoter !**

Voici les miennes :

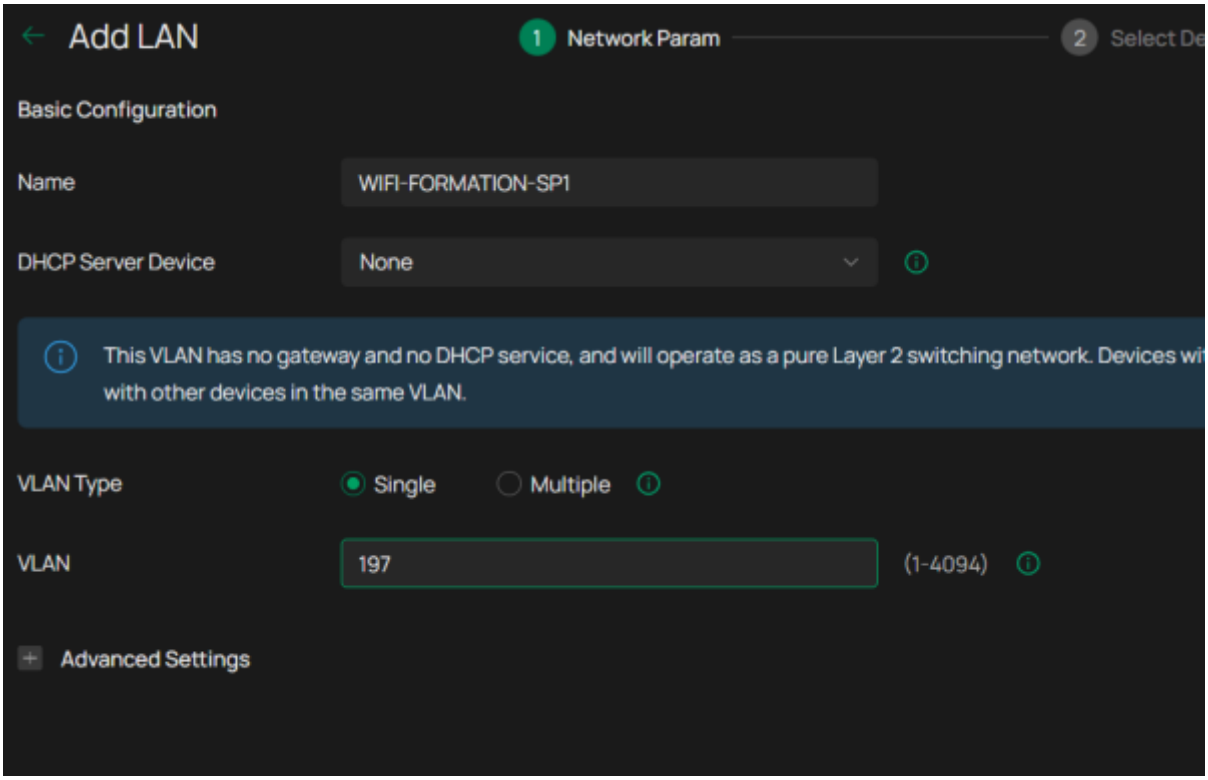


4.3 Configuration des SSID

Pour l'instant **notre AP est branché sur un switch en port access**. Ce branchement lui permet de **diffuser un SSID et uniquement un seul.** Cependant, cette configuration **ne lui permet pas de gérer de vlans taggués** et donc de diffuser plusieurs SSID sur **le même cable.**

On se rend **dans general settings —> LAN —> ADD LAN**

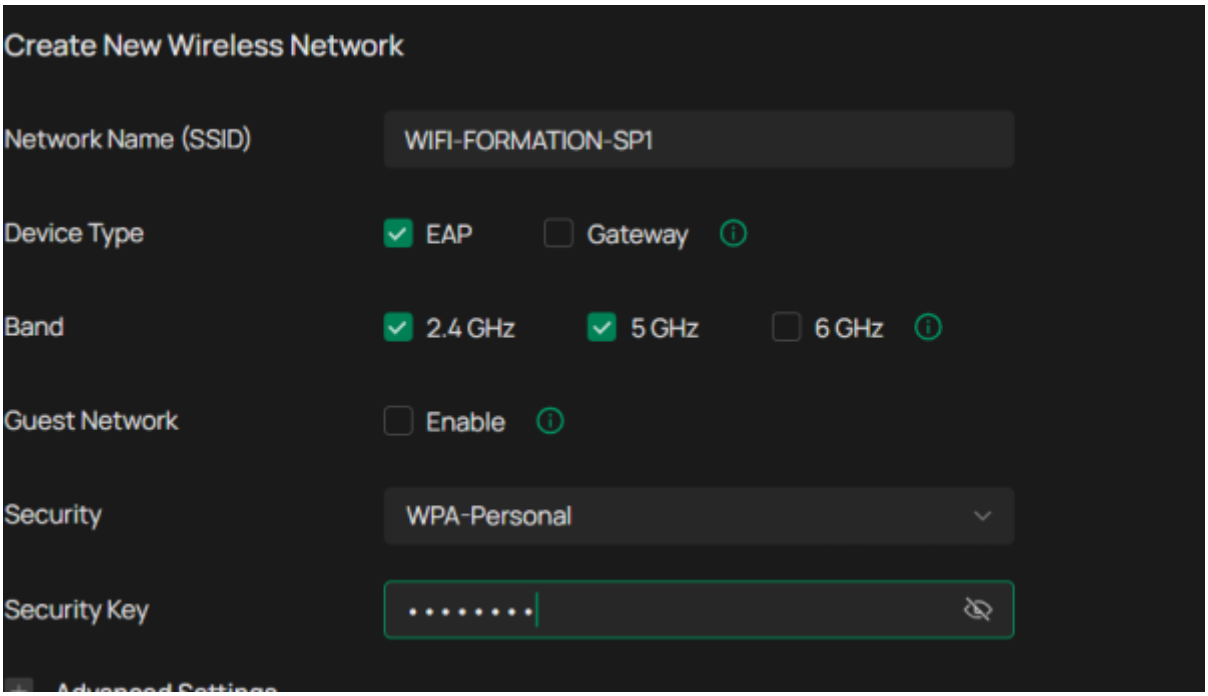
On rentre les informations du vlan que l'on souhaite diffuser :



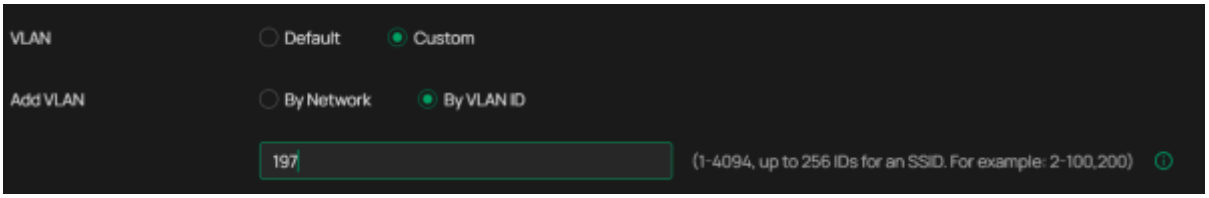
Une fois **le vlan crée**, il faut **l'attribuer à un SSID**

On se rend dans **general settings —> WLAN —> Create New Wireless Network**

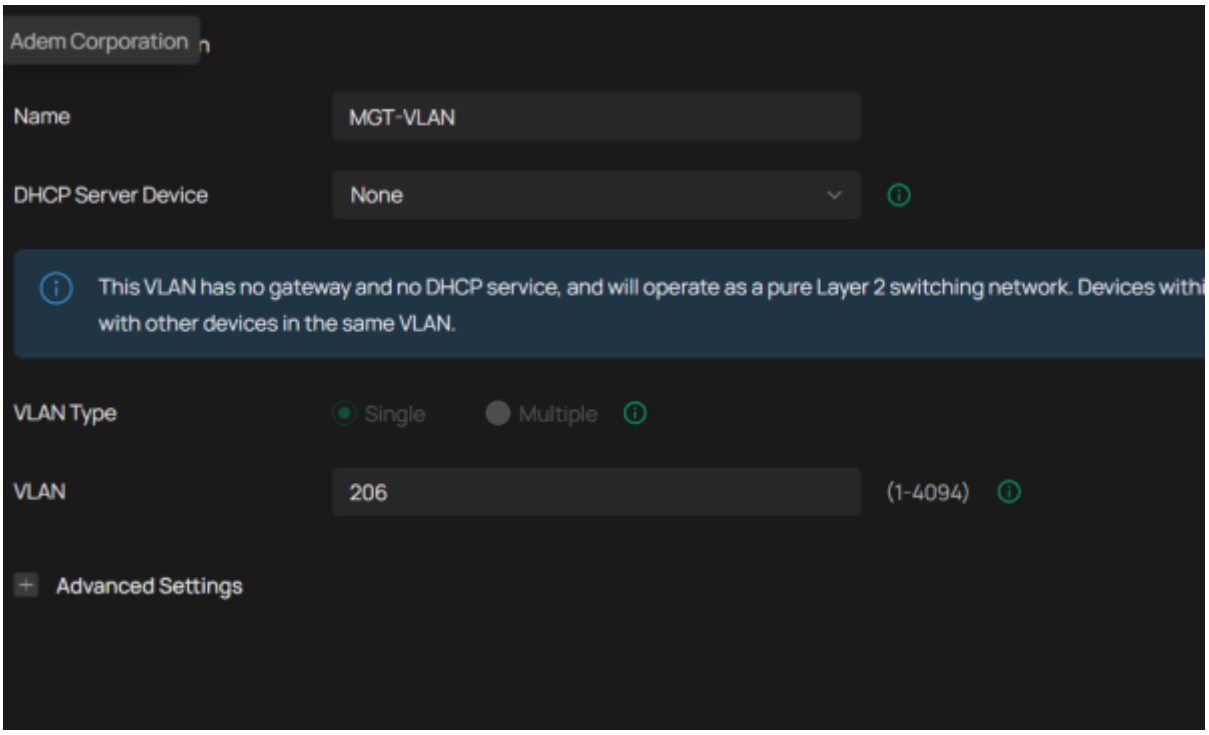
On rentre les informations du SSID :



Puis on descend jusque “advanced settings” pour rentrer le numéro du tag vlan :



Après avoir**configuré nos deux SSID** que l'on souhaite diffuser, il faut crée un **VLAN management** qui nous servira à la gestion de l'AP :



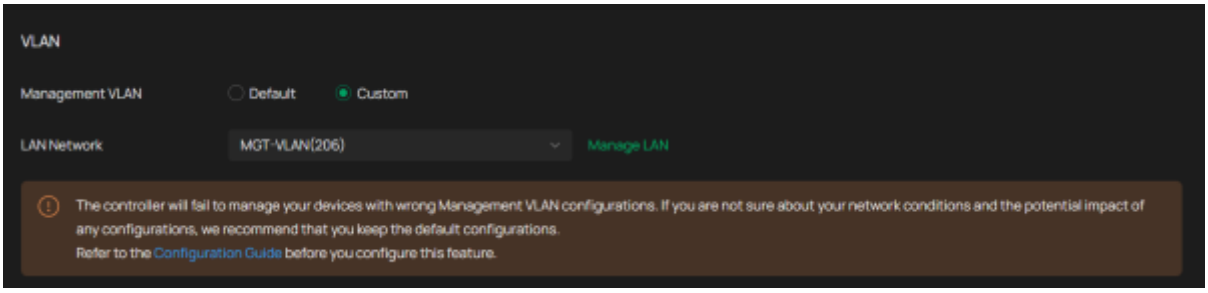
Il nous manque plus qu'à **configurer l'appareil pour qu'il puisse gérer les VLANs taggués**. Nous devrions déjà avoir **configuré un port du switch en trunk avec les VLANs dont nous avons besoin qui sont autorisés**. Voici le port ainsi que les commandes tapés sur **mon switch pour mon cas :**

```
switch# enable
switch> conf t
switch(config-if) interface fa0/11   #on configure l'interface fa0/11
switch(config-if) switchport mode trunk      #on change le port en trunk
switch(config-if) switchport trunk allowed vlan 196,197,206   #on autorise
les vlans sur le port
switch(config-if) exit
switch> exit
switch# write memory     #on sauvegarde les modifications
```

On reviens au tableau de bord de notre site puis dans :

Devices —> notre appareil —> Manage Device —> Config —> General

On attribue le vlan MGT-VLAN que l'on vient de configuré précédemment en tant que vlan management :



Attention, on perd l'accès à l'AP car il **gère maintenant les VLANs** taggués. On le branche donc au port du switch contenant **les VLANs configurés en trunk.**

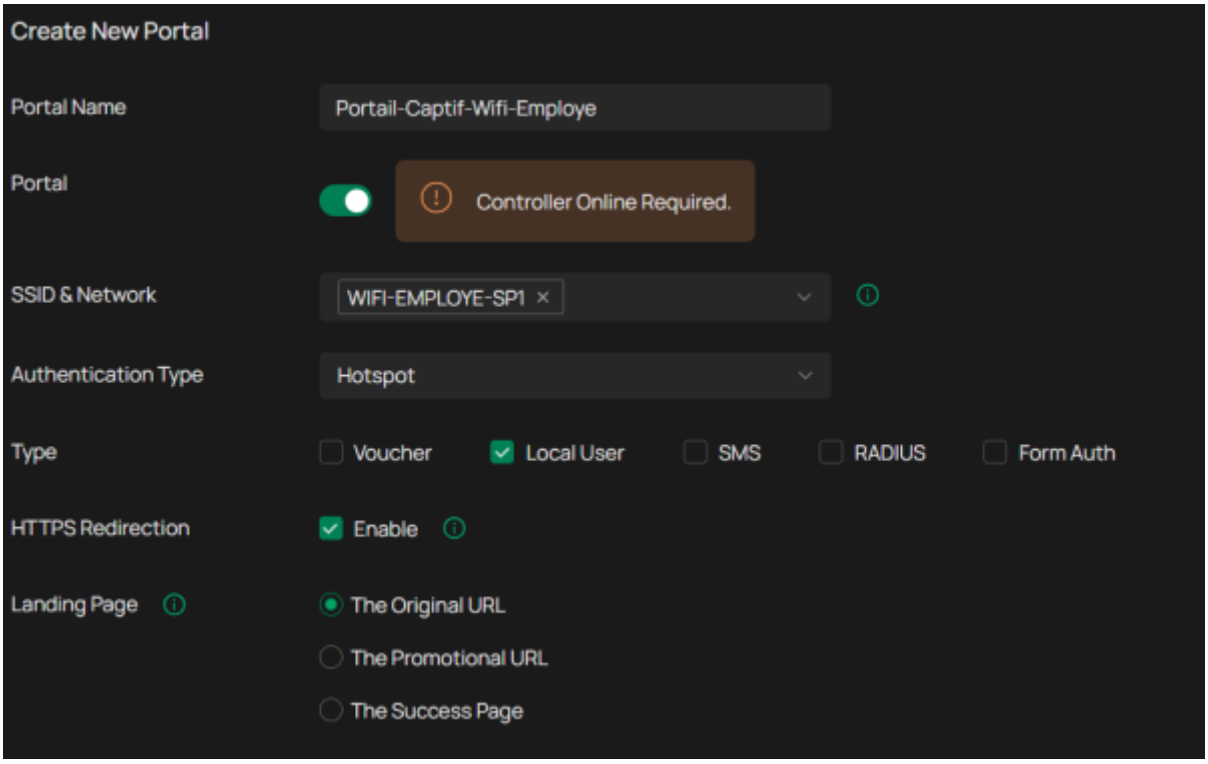
4.4 Mise en place du portail captif

Nous allons maintenant **configurer un portail captif** pour nos deux SSID configurés **précédemment** de sorte à ce que **chaque employé puisse se connecter au WIFI via son compte utilisateur**.

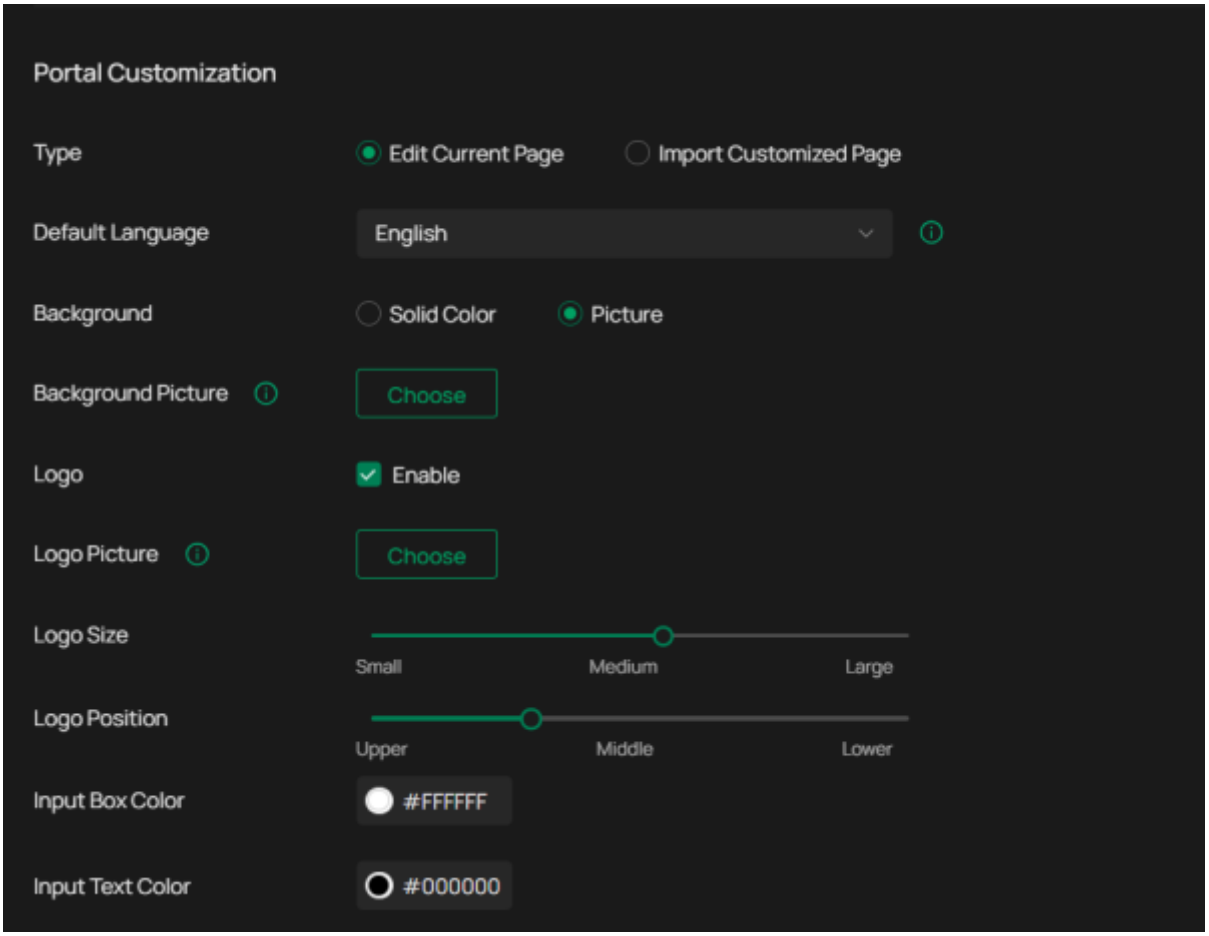
ETAPE 1 : CREATION DES PORTAILS CAPTIFS

On se rend dans **Authentication —> Portal —> Create New Portal**

On rentre les informations du portail captif :

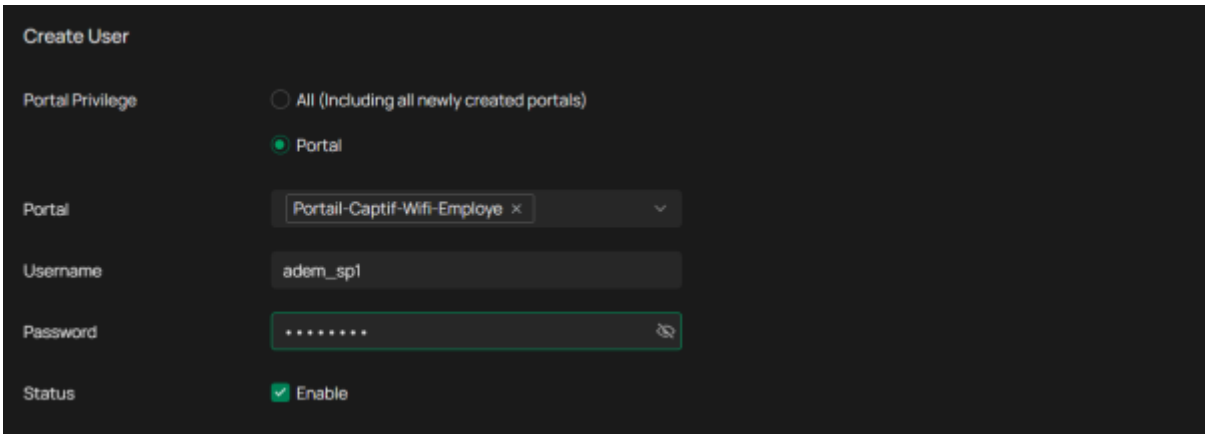


On peut le personnaliser à notre guise :



ETAPE 2 : CREATION DES UTILISATEURS

On se rend dans **Hotspot —> Local Users —> Create User** .



Chaque utilisateur **sera crée sur son portail captif** afin que les **employés ne se connectent** pas sur le portail captif **de ceux qui viennent se former et inversement.**

4.4 Mis en place du Serveur Proxy

ÉTAPE 1 — Création du conteneur Squid

On crée un **conteneur dédié** au service **Squid** sur le **vlan srv**, avec l'adresse IP **10.31.204.90/23** :

On met à jour les paquets :

```
root@squid-srv:~# apt update && apt upgrade -y
```

—

ÉTAPE 2 — Installation de Squid

```
root@squid-srv:~# apt install squid -y
```

On vérifie que le service tourne correctement :

```
root@squid-srv:~# systemctl status squid
```

—

ÉTAPE 3 — Configuration de Squid

On sauvegarde la configuration originale **avant toute modification** :

```
root@squid-srv:~# cp /etc/squid/squid.conf /etc/squid/squid.conf.bak
```

> **Port d'écoute**
>
> Squid écoute sur le **port 3128** par défaut. Les clients qui souhaitent passer par le proxy enverront leurs requêtes sur ce port.

> **Les ACL (Access Control List)**
>
> Une **ACL** est un alias que l'on attribue soit à un **réseau source** (src) autorisé à contacter le proxy, soit à un **domaine de destination** (dstdomain) à bloquer.
>
> ```
> acl WIFI_EMPLOYE src 10.31.196.0/24
> acl WIFI_FORMATION src 10.31.197.0/24
> acl RESEAUX_SOCIAUX dstdomain
> "/etc/squid/blacklist/reseaux_sociaux.txt"
> acl ADULTES dstdomain "/etc/squid/blacklist/adultes.txt"
> ```

On modifie le fichier de configuration de Squid :

```
root@squid-srv:~# nano /etc/squid/squid.conf
```

On **remplace tout le contenu** par la configuration suivante :

```
# Port d'écoute du proxy
```

```
http_port 3128

# Réseaux autorisés à utiliser le proxy
acl WIFI_EMPLOYE src 10.31.196.0/24
acl WIFI_FORMATION src 10.31.197.0/24

# Blacklist réseaux sociaux et sites adultes
acl RESEAUX_SOCIAUX dstdomain "/etc/squid/blacklist/reseaux_sociaux.txt"
acl ADULTES dstdomain "/etc/squid/blacklist/adultes.txt"

# Règles de filtrage
http_access deny RESEAUX_SOCIAUX
http_access deny ADULTES

# Autoriser les VLANs WIFI
http_access allow WIFI_EMPLOYE
http_access allow WIFI_FORMATION

# Refuser tout le reste
http_access deny all
```

> **Important**
>
> Squid **lit les règles de haut en bas**. Il est donc impératif de placer les règles de **deny** avant les **allow**, et de ne pas oublier le **deny all** final pour bloquer tout ce qui n'est pas explicitement autorisé.

On crée les blacklists :

```
root@squid-srv:~# mkdir -p /etc/squid/blacklist
root@squid-srv:~# nano /etc/squid/blacklist/reseaux_sociaux.txt
```

Contenu :

```
.facebook.com
.instagram.com
.tiktok.com
.twitter.com
.x.com
.snapchat.com
```

> **Pourquoi les points ?**
>
> Les **points en début de domaine** sont importants : par exemple **.facebook.com** bloque à la fois **www.facebook.com**, **m.facebook.com**, etc.

On crée ensuite la blacklist des **sites adultes** :

```
root@squid-srv:~# nano /etc/squid/blacklist/adultes.txt
```

Contenu :

```
.pornhub.com
.xvideos.com
.xhamster.com
.redtube.com
```

On **redémarre Squid** pour appliquer la configuration :

```
root@squid-srv:~# systemctl restart squid
```

—

ÉTAPE 4 — Pare-feu

On ajoute les règles suivantes sur **OPNsense** pour autoriser le flux vers le proxy et bloquer l'accès direct à Internet depuis les VLANs WIFI :

Règles à AUTORISER

Source	Destination	Port	Protocole	Raison
10.31.196.0/24	10.31.204.90	3128	TCP	WIFI-EMPLOYE → Squid
10.31.197.0/24	10.31.204.90	3128	TCP	WIFI-FORMATION → Squid
10.31.204.90	Any	80/443	TCP	Squid → Internet
10.31.204.90	10.31.200.53	53	TCP/UDP	Squid → DNS

Règles à BLOQUER

Source	Destination	Port	Protocole	Raison
10.31.196.0/24	Any	80/443	TCP	BLOQUER accès direct
10.31.197.0/24	Any	80/443	TCP	BLOQUER accès direct

Ces règles de **blocage de l'accès direct** sont essentielles : sans elles, les clients pourraient **contourner le proxy** en accédant directement à Internet sur les ports 80/443.

—

ÉTAPE 5 — Configuration du Proxy pour les clients

Deux options s'offrent à nous :

- **Option 1 — Manuelle** : chaque client configure son navigateur pour pointer vers le proxy (adresse : **10.31.204.90**, port : **3128**). Peu pratique à grande échelle.

- **Option 2 — Proxy transparent** : le **pare-feu redirige automatiquement** les requêtes des clients vers le proxy. Les clients n'ont **rien à configurer**.

Nous allons opter pour le **proxy transparent**.

On ouvre le fichier de configuration de Squid :

```
root@squid-srv:~# nano /etc/squid/squid.conf
```

On remplace la ligne :

```
http_port 3128
```

Par :

```
http_port 3128 intercept
```

On redémarre le service :

```
root@squid-srv:~# systemctl restart squid
```

—

6.2 Rediriger le trafic vers Squid sur OPNsense

Pour que le proxy transparent fonctionne, il faut créer une **règle NAT de redirection (port forward)** sur OPNsense qui intercepte le trafic HTTP des clients WIFI et le redirige vers Squid.

On se rend dans **Firewall —> NAT —> Port Forward —> Ajouter une règle** :

Règle NAT à créer

Champ	Valeur
Interface	WIFI-EMPLOYE / WIFI-FORMATION
Protocole	TCP
Source	10.31.196.0/24 (ou 10.31.197.0/24)
Port destination	80
Rediriger vers	10.31.204.90
Port de redirection	3128

On répète l'opération pour **chaque VLAN WIFI** (WIFI-EMPLOYE et WIFI-FORMATION).

Désormais, tout le trafic HTTP des clients connectés au WIFI sera **automatiquement intercepté** par Squid, qui appliquera les règles de filtrage configurées précédemment.