3.3.20. RADIUS 802.1X

Afin de protéger le réseau Ethernet filaire, nous préconisons la mise en en place d'un serveur Radius.

La norme 802.1x permet l'authentification du matériel IP avant tout accès au réseau filaire ou Wifi.
Les authentifications sont sécurisées, et les échanges se font :

• sur un chiffrement **Mode EAP** « simple » : md5 **ou** MSCHAPV2
Ces deux modes nécessitent : une **identité** et un **password**.

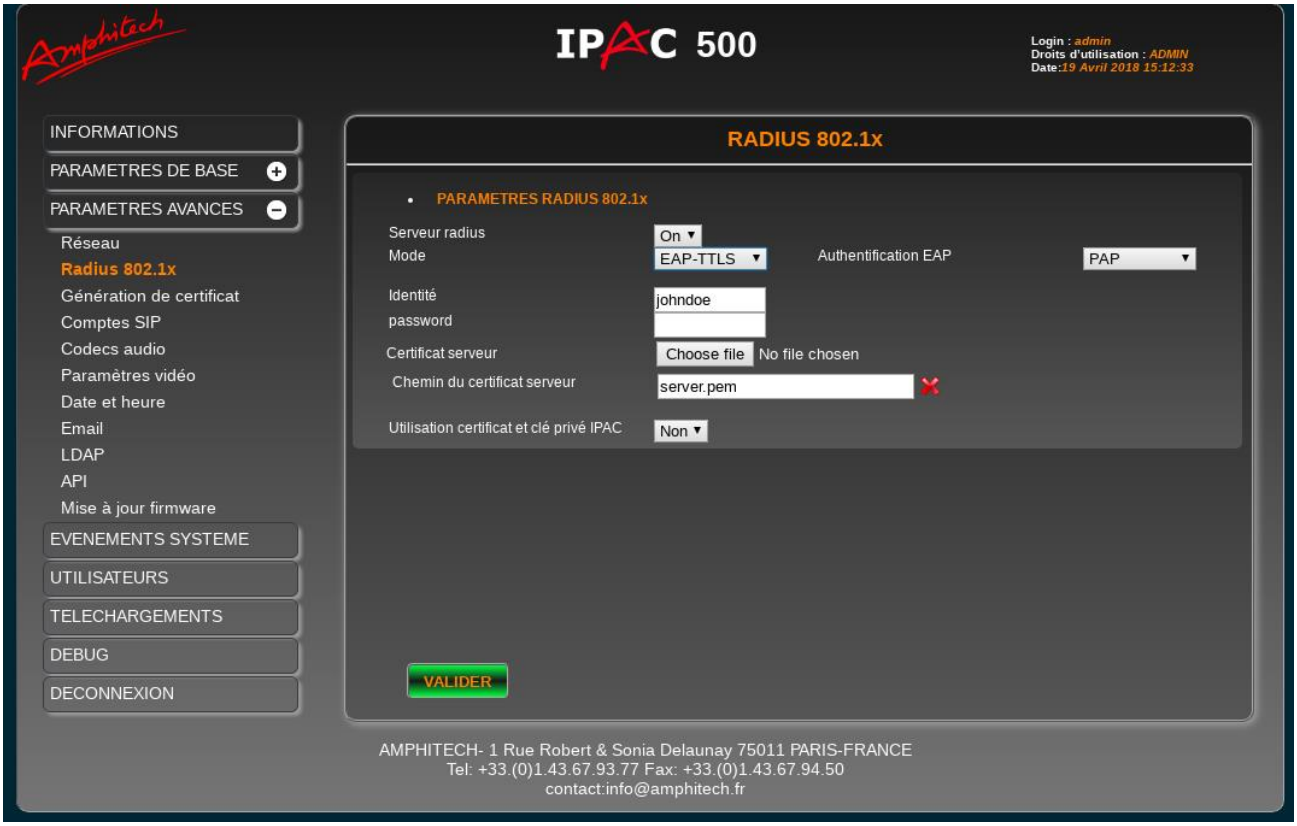
• des modes sécurisés **EAP** : PEAP, EAP-TTLS, EAP-TLS.

- En Mode EAP : **PEAP** ou **TTLS** l'ensemble fonctionne sur le principe d'un **identifiant (identité)** et d'un **password**.

1. En fonction de la configuration du serveur dans chaque mode EAP il est possible de régler le protocole d'authentification eap (2eme phase d'authentification):

   Pour le EAP-TTLS **Authentification eap** : PAP, MD5, CHAP, MSCHAPv2.

   Pour le EAP-PEAP **Authentification eap** : PAP, MD5, CHAP, MSCHAPv2 et TLS.



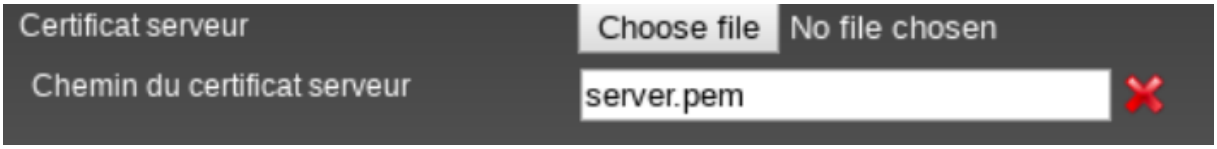
Exemple serveur (Free Radius) :

Dans la configuration générale d'EAP, si besoin selon votre version, remplacer la ligne
`default_eap_type = ttls`

Dans la configuration du TTLS

```
ttls {
# The tunneled EAP session needs a default
# EAP type which is separate from the one for
# the non-tunneled EAP module …
default_eap_type = md5
}
```

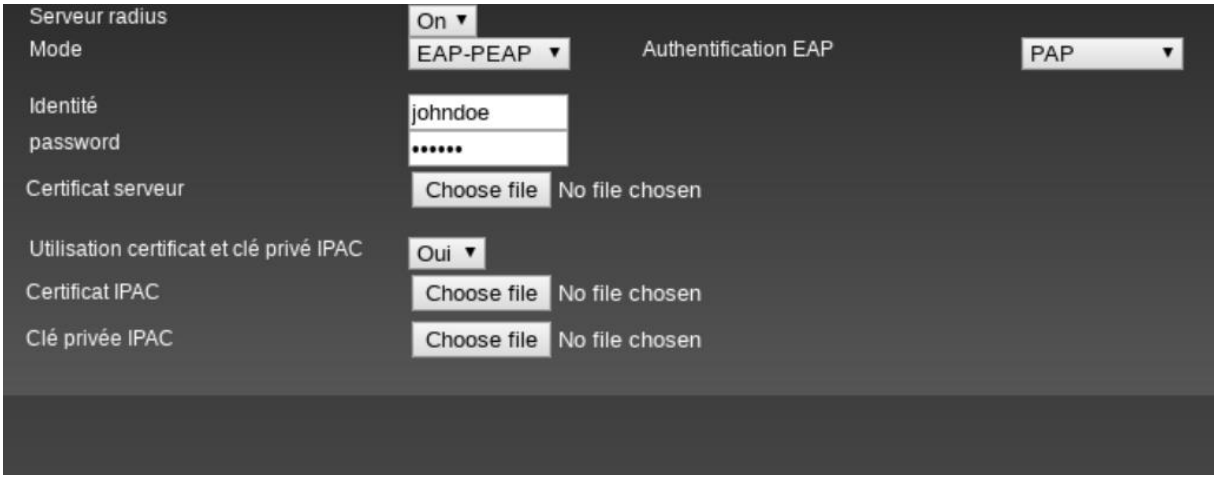
2. Ensuite, il est possible ou non d'utiliser la vérification d'un certificat serveur dans le procédé d'authentification pour le **Mode EAP** : **PEAP** et **TTLS**. Cette nécessité de certificat se paramètre côté serveur.
   Pour utiliser un certificat auto-signé ou signé par une autorité de certification, il faut importer le certificat CA.pem dans l'IPAC. Si aucun fichier de type *.pem* n'est importé , l'IPAC ne transmettra pas le certificat au serveur (si nécessité) , et l'authentification échouera.



3. Certaines configurations de serveurs ne nécessitent pas le contrôle du certificat demandeur (IPAC) et utilisent la méthode de certificat symétrique en utilisant le certificat et la clé privée du serveur lors de la phase « Certificate server Key Exchange ».

   Or dans certaines configurations serveur il est possible de demander à l'IPAC son propre certificat ainsi que sa clé privée pour le processus d'authentification.

   Si l'option « **utilisation certificat et clé privé IPAC** » est passée à « **oui** », alors,



- Ajouter manuellement un certificat et clé privé au format X.509 (auto-signé ou signé par une autorité) pour le mode EAP : PEAP ou TTLS

- Utiliser la génération automatique de cette paire par la page web « **génération de certificat et clé privé** » ATTENTION à bien vérifier l'heure et la date de l'IPAC avant de générer un certificat.

- Utiliser le certificat et clé privé Amphitech par défaut (si aucun certificat et clé importés)

➢ En Mode EAP : **TLS**

Cette méthode nécessite une authentification mutuelle entre le serveur et le demandeur (IPAC), ***Utiliser obligatoirement : certificat Serveur, clé privé pour l'IPAC, passphrase de la clé privée.***

Il n'y a plus dans ce cas d'utilisation de pair login/password, mais, l'utilisation d'un **mot de passe de clé privé** (passphrase) utilisé pour générer la clé privée et le certificat pour l'IPAC (format PKI). Dans le cas de l'utilisation d'un p12 ⚠ne pas inclure le certificat client IPAC (puisque déjà présent dans le p12).

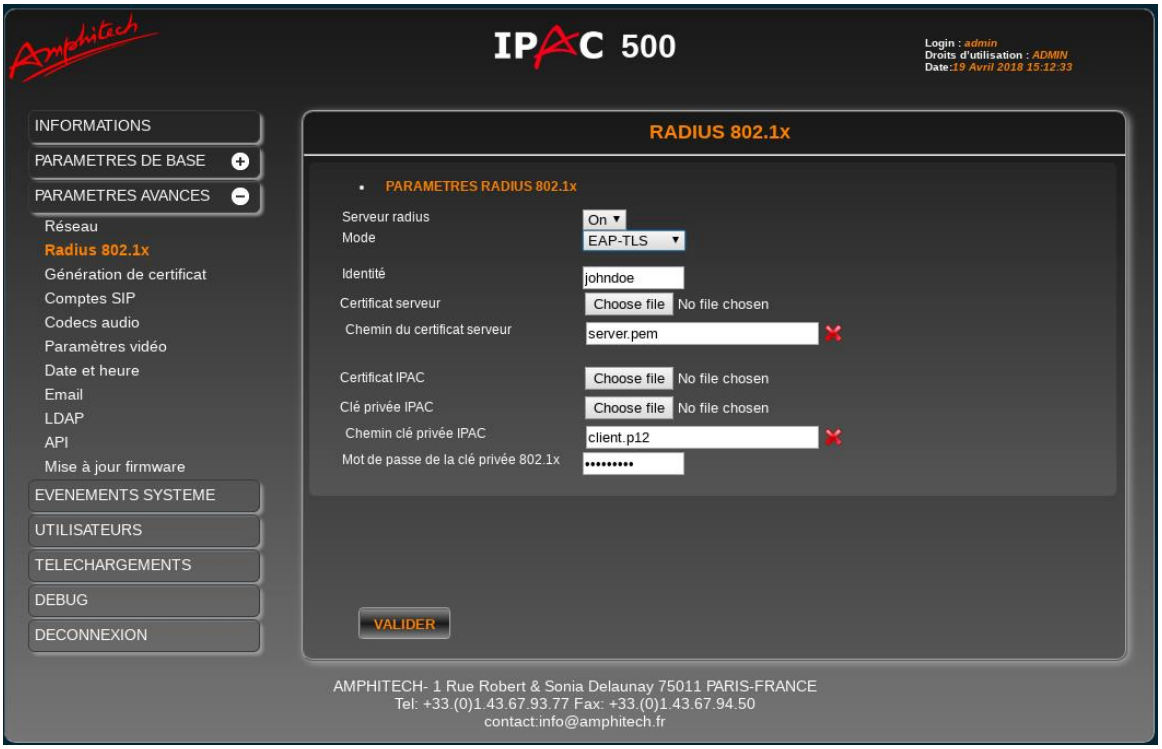
Il est possible de passer en mode Anonymous (plus d'identité au niveau du serveur) dans ce cas, dans la partier « identité » rentrer : **anonymous**.

Dans ce cas la page web de génération de certificat et ce clé privé ne peut pas être utilisée.

Le certificat émis par une PKI est sous forme d'un fichier PKCS (extension. p12) contenant :

- La clé privée
- Le certificat associé (clé publique signée par l'autorité)

Il faudra alors remplir tous les champs de la page :

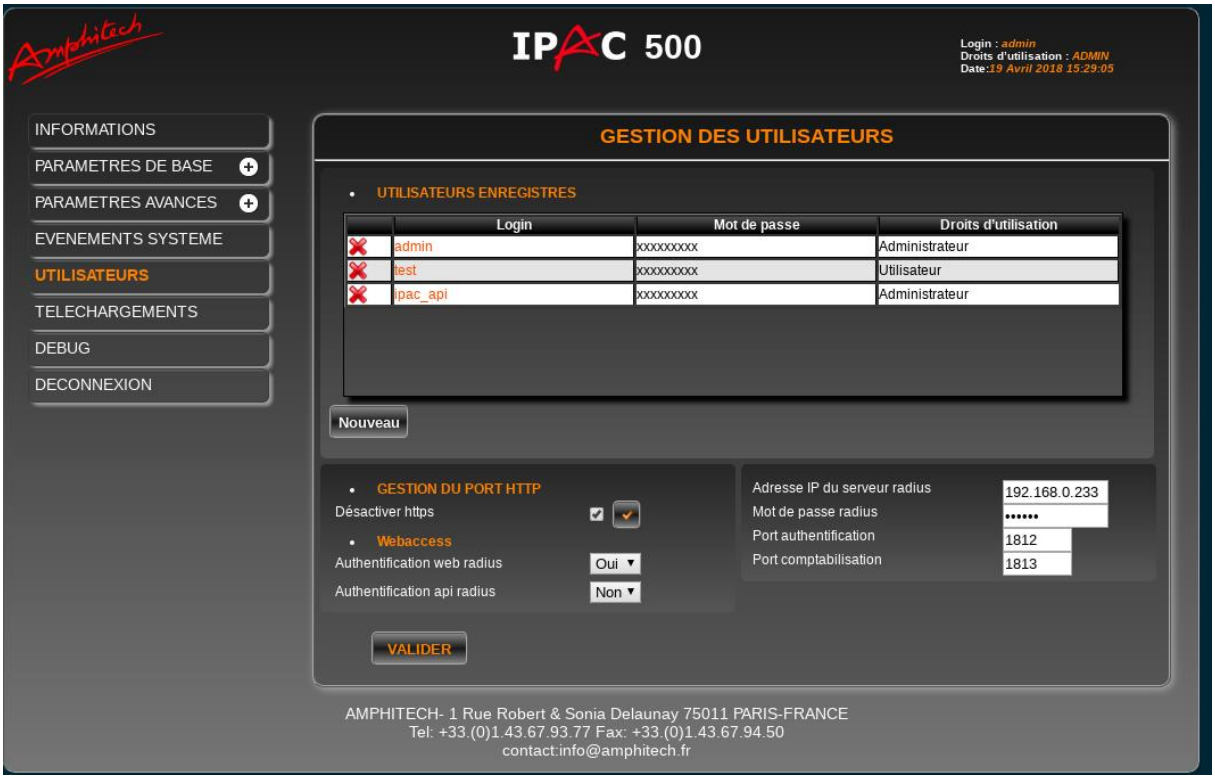


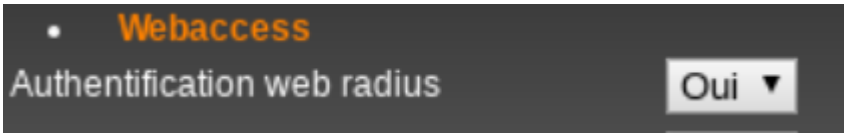
Accès web par authentification serveur Radius

Le Serveur Radius permet aussi de gérer l'authentification des comptes (Accounting) via la méthode PAP pour accéder aux pages web de paramétrage du portier.

La méthode initiale interne à l'IPAC permet de créer des comptes locaux d'administration et d'utilisation avec comme attributs :

- **Login**
- **Mot de passe**
- Droit d'utilisation : **Administrateur** ou **utilisateur**



En activant la solution 

L'authentification interne à l'IPAC fonctionnera encore, si le login et mot de passe correspondent, l'accès aux pages s'effectuera en fonction des droits d'utilisation du compte local.

Si le login et/ou le mot de passe ne correspondent pas à un compte interne à l'IPAC, et si la méthode RADIUS est activée, alors l'IPAC enverra une requête de demande d'authentification au serveur radius si :

- **L'adresse IP du serveur Radius** est renseignée.
- **Le mot de passe Radius** crée pour le client IPAC lors de la création du compte client sur le serveur est renseigné.
- Les Ports d'**authentification** et de **comptabilisation** sont renseignés.

Dans tous les cas si aucun login/password ne correspond à un compte local IPAC ou sur le serveur radius, l'authentification échouera, la connexion au pages sera impossible.

<u>Exemple pour un serveur Free Radius</u>

➢ Création d'un compte client IPAC pour l'authentification (*/etc/freeradius/client.conf*) :

- Déclaration de l'adresse IP de l'IPAC
- Password secret (ipac1234)

```
#######################################################################
#
#  Per-socket client lists.  The configuration entries are exactly
#  the same as above, but they are nested inside of a section.
#
#  You can have as many per-socket client lists as you have "listen"
#  sections, or you can re-use a list among multiple "listen" sections.
#
#  Un-comment this section, and edit a "listen" section to add:
#  "clients = per_socket_clients".  That IP address/port combination
#  will then accept ONLY the clients listed in this section.
#
#clients per_socket_clients {
#       client 192.168.3.4 {
#               secret = testing123
#        }
#}

client 192.168.0.39 {
        secret = ipac1234
}
```

- Création d'un utilisateur (login) d'accès web (user. Conf)
  - Login : **johndoe**
  - Password : **123456789**
  - Droits d'accès : **Administrative-User** *(droit admin IPAC)* ou **Login-User** *(droit utilisateur IPAC)*

```
# #
# # Last default: shell on the local terminal server.
# #
# DEFAULT
#       Service-Type = Administrative-User

# On no match, the user is denied access.

johndoe   Cleartext-Password := "123456789"
        Service-Type = Administrative-User
```

Dans cette fenêtre d'identification du login, si l'option radius est activée, il est possible de s'authentifier soit :

- Admin /mot de passe compte administrateur local (toujours valide).
- Login /mot de passe (compte créé localement sur l'IPAC)
- Login/ mot de passe via RADIUS exemple : johndoe /123456789 permettant d'ouvrir la page dans ce cas Administrateur.