

# SSH - Ajout de la 2FA

La double authentification permet d'ajouter au système d'authentification classique (mot de passe ou clé SSH) une couche supplémentaire unique avec un code unique générer régulièrement.

Personnellement je l'utilise sur toutes les machines ou le SSH est activé sur le port publique.

## Installation du module PAM Google Authenticator

! Ce produit est développé par Google mais aucune information personnelle ou donnée de tracking n'est envoyé à Google lors de son installation ou de son utilisation ! #RGPD

Le module s'installe de la manière la plus classique:

```
apt install libpam-google-authenticator -y
```

## Configuration de PAM

Modifier le fichier `/etc/pam.d/ssh` pour ajouter:

```
auth required pam_google_authenticator.so
```

## Configuration de sshd

Modifier le fichier `/etc/ssh/sshd_config` pour modifier la ligne:

```
ChallengeResponseAuthentication no
```

Par:

```
ChallengeResponseAuthentication yes
```

# Initialisation de la 2FA

Il faut être connecté avec le compte sur lequel on souhaite activer la 2FA en ssh !

Lancer la commande suivante:

```
google-authenticator
```

Et répondre aux questions de la manière suivante:

```
Do you want authentication tokens to be time-based (y/n) y
```

```
# A ce moment le QR-Code apparait, pour éviter qu'un écran petit cache le code après avoir
répondu au question, il est recommandé d'ajouter le code maintenant avec la "Ajout de la 2FA
sur son mobile"
```

```
Do you want me to update your "/root/.google_authenticator" file? (y/n) y
```

```
# La réponse à la prochaine question dépend de vous
```

```
# Si vous répondez 'n' vous acceptez qu'un code puisse être utilisé plusieurs fois
```

```
Do you want to disallow multiple uses of the same authentication
```

```
token? This restricts you to one login about every 30s, but it increases
your chances to notice or even prevent man-in-the-middle attacks (y/n) y
```

```
# La question suivante définit la possibilité d'utiliser un code dans les 4 minutes qui suivent
afin de compenser une désynchronisation de temps
```

```
# Répondre 'y' autorise les 4 minutes mais augmente les chances d'attaque
```

```
By default, a new token is generated every 30 seconds by the mobile app.
```

```
In order to compensate for possible time-skew between the client and the server,
we allow an extra token before and after the current time. This allows for a
time skew of up to 30 seconds between authentication server and client. If you
experience problems with poor time synchronization, you can increase the window
from its default size of 3 permitted codes (one previous code, the current
code, the next code) to 17 permitted codes (the 8 previous codes, the current
code, and the 8 next codes). This will permit for a time skew of up to 4 minutes
between client and server.
```

```
Do you want to do so? (y/n) n
```

```
# Ici on indique si oui ou non on veut limité le nombre d'essaie à la 2FA
# Très fortement déconseillier de mettre non car cela autoriserai un robot à faire une
attaque brut-force

If the computer that you are logging into isn't hardened against brute-force
login attempts, you can enable rate-limiting for the authentication module.
By default, this limits attackers to no more than 3 login attempts every 30s.
Do you want to enable rate-limiting? (y/n) y
```

# Redémarrage du service sshd

Redémarrer le service sshd afin d'appliquer les paramètres:

```
systemctl restart sshd
```

Revision #2

Created 25 March 2024 23:10:07 by Nicolas

Updated 13 February 2025 22:15:27 by Nicolas