Comment bloquer un compte utilisateur après plusieurs tentatives échouées


Voici, comment renforcer simplement la sécurité en bloquant les compte utilisateurs après un nombre consécutive d'authentifications échoué.
Cela peut être réalisé par l'utilisation du module `pam_faillock` ce qui permet de verrouiller temporairement le compte de l'utilisateur dans le cas de multiples tentatives authentifications et de garder un enregistrement de cette évènement.

`pam_faillock` est une partie de Linux PAM (Pluggable Authentication Modules), c'est un mécanisme dynamique pour implémenter des services d'authentification dans les applications et divers services système pour auditer l'activité du shell d'une connexion utilisateur.

Comment verrouiller des comptes d'utilisateurs après des échecs d'authentification consécutifs

Vous pouvez configuré les fonctionnalité ci-dessous dans le fichier /etc/pam.d/system-auth et /etc/pam.d/password-auth, en ajoutant les lignes suivantes:

```
auth    required       pam_faillock.so preauth silent audit deny=3
unlock_time=600
auth    [default=die]  pam_faillock.so authfail audit deny=3 unlock_time=600
```

- **audit** : Correspond à l'utilisateur audité
- **deny** : Utilisé pour définir le nombre de tentative (3 dans notre cas), après ça le compte utilisateur devrai être bloqué.
- **unlock_time** : défini le temps (300 secondes, soit 5 minutes) pour que le compte soit débloqué.


A noté que l'ordre des lignes est vraiment important, une mauvaise configurations peut bloquer l'ensemble des login de la machine.

La section `auth` dans les deux fichiers doit être organisé de cette façon :

```
auth        required      pam_env.so
auth        required      pam_faillock.so preauth silent audit deny=3
unlock_time=300
auth        sufficient    pam_unix.so  nullok  try_first_pass
auth        [default=die]  pam_faillock.so  authfail  audit  deny=3
unlock_time=300
auth        requisite     pam_succeed_if.so uid >= 1000 quiet_success
```

```
auth        required      pam_deny.so
```

Maintenant ouvrez les 2 fichiers :

```
vim /etc/pam.d/system-auth
vim /etc/pam.d/password-auth
```

L'entrée par défaut dans la section `auth` des 2 fichiers ressembles à ça:

```
#%PAM-1.0
# This file is auto-generated.
# User changes will be destroyed the next time authconfig is run.
auth        required      pam_env.so
auth        sufficient    pam_fprintd.so
auth        sufficient    pam_unix.so nullok try_first_pass
auth        requisite     pam_succeed_if.so uid >= 1000 quiet
auth        required      pam_deny.so
```

Après l'ajout des paramètres ci-dessus, ça devrai ressemblé à ça:

```
#%PAM-1.0
# This file is auto-generated.
# User changes will be destroyed the next time authconfig is run.
auth        required      pam_env.so
auth        required      pam_faillock.so preauth silent audit deny=3
unlock_time=300
auth        sufficient    pam_fprintd.so
auth        sufficient    pam_unix.so nullok try_first_pass
auth        [default=die]  pam_faillock.so  authfail  audit  deny=3
unlock_time=300
auth        requisite     pam_succeed_if.so uid >= 1000 quiet
auth        required      pam_deny.so
```

Puis ajouter la ligne suivant à la fin de la section `account` dans les 2 fichiers du dessus:

```
account     required      pam_faillock.so
```

Comment verrouiller le compte ''root'' après l'échec des tentatives de connexion

Pour verrouiller le compte root après l'échec des tentatives d'authentification, ajoutez l'option even_deny_root aux lignes des deux fichiers de la section `auth` comme ceci.

```
auth        required      pam_faillock.so preauth silent audit deny=3
even_deny_root unlock_time=300
auth        [default=die]  pam_faillock.so  authfail  audit  deny=3
even_deny_root unlock_time=300
```

Une fois que vous avez tout configuré.
Vous pouvez redémarrer les services d'accès distant comme **sshd**, pour que la stratégie ci-dessus prenne effet, si les utilisateurs utilisent ssh pour se connecter au serveur.

Comment voir les tentatives d'authentifications échoués

Vous pouvez voir tous les logs d'authentification ayant échoué à l'aide de l'utilitaire de `faillock`, qui est utilisé pour afficher et modifier le log des échecs d'authentification.
Vous pouvez voir les tentatives de connexion échouées pour un utilisateur particulier comme cela.

```
faillock --user toto
```

Pour afficher toutes les tentatives de connexion infructueuses, exécutez `faillock` sans aucun argument comme ceci:

```
faillock
```

Pour effacer les logs d'échec d'authentification d'un utilisateur, exécutez cette commande:

```
faillock --user toto --reset
```

Pour effacer tous les logs de l'ensemble des utilisateurs :

```
faillock --reset
```

Enfin, pour dire au système de ne pas verrouiller les comptes d'un utilisateur ou de plusieurs utilisateurs après plusieurs tentatives de connexion infructueuses, ajoutez l'entrée suivante, juste au-dessus de l'appel `pam_faillock` dans la section `auth` des deux ficheirs `/etc/pam.d/system-auth` et `/etc/pam.d/password-auth`comme ça (Ajoutez simplement les noms utilisateurs séparés par deux points.)

```
auth   [success=1 default=ignore] pam_succeed_if.so user in toto:tata
```

Vos fichiers devrai ressembler à cela:

```
auth  required      pam_env.so
auth   [success=1 default=ignore] pam_succeed_if.so user in toto:tata
auth   required      pam_faillock.so preauth silent audit deny=3
unlock_time=600
auth   sufficient    pam_unix.so  nullok  try_first_pass
auth   [default=die]  pam_faillock.so  authfail  audit  deny=3
unlock_time=600
auth   requisite     pam_succeed_if.so uid >= 1000 quiet_success
auth   required      pam_deny.so
```

source : tecmint.com

From:
https://www.ksh-linux.info/ - **Know Sharing**

Permanent link:
https://www.ksh-linux.info/systeme/comment-bloquer-un-compte-utilisateur-apres-plusieurs-tentatives-echouees

Last update: **28/02/2018 20:22**