Authentic 2 - Development #51239

ldap: add method to get ppolicy operational attributes

18 février 2021 11:09 - Loïc Dachary

Statut:	Fermé	**Début:**	18 février 2021
Priorité:	Normal	**Echéance:**	
Assigné à:	Loïc Dachary	**% réalisé:**	100%
Catégorie:	LDAP	**Temps estimé:**	0:00 heure
Version cible:			
Patch proposed:	Oui	**Planning:**	Non

Description

La password policy ajoute des attributs operationels à l'utilisateur. Il serait utile d'avoir une fonction qui permet de les obtenir afin, par exemple, d'adapter la rédaction d'un message d'erreur lorsqu'un echec de connexion est du à des règles imposées par la password policy. L'alternative à une fonction spécifique serait de systématiquement demander ces attributs lors d'un bind (mais seulement lorsque l'option use_controls est True). Cependant ces champs ne sont pas utilisés la plupart du temps cela alourdi un peu les transferts de données et l'occupation mémoire.

Demandes liées:

Lié à Authentic 2 - Development #51653: ldap: fetching and using ppolicy attr...	**Nouveau**	**04 mars 2021**

Révisions associées

Révision 1f637825 - 18 mai 2021 18:05 - Loïc Dachary

ldap: add method to get ppolicy operational attributes (#51239)

Fixes: #51239

License: MIT

Révision ff9e90ce - 18 mai 2021 18:10 - Benjamin Dauvergne

ldap: default use_controls to False (#51239)

Révision 4f8f5f62 - 18 mai 2021 20:58 - Benjamin Dauvergne

tests: fix tests on ppolicy (#51239)

Historique

#1 - 18 février 2021 11:32 - Benjamin Dauvergne

- Assigné à changé de Benjamin Dauvergne à Loïc Dachary

Je préférerai un ticket global montrant l'usage de l'API introduite, mais tu peux garder celui-ci et avancer sur la suite, je note ici aussi que comme dit dans le ticket initial ouvert par EE sur le sujet, que pwdUniqueAttempts ne fait pas partie de la RFC ppolicy ni du module ppolicy livré avec OpenLDAP mais est une extension commandé par Mozilla à zytrax et distribué uniquement dans le fork zytrax de slapo-ppolicy (c'est libre, c'est juste pas disponible facilement). Il faudrait en premier lieu gérer le cas général avec éventuellement le cas zytrax.

Par ailleurs tant qu'à implémenter des trucs très spécifiques je serais plus intéressé par une extraction de certains éléments de la ppolicy concernant l'utilisateur notamment pour les messages du type "votre mot de passe ne respecte pas les contraintes", « oui mes lesquelles ? » ou "votre mot de passe est trop court", « ok mais quelle longueur doit-il faire doudoudidon ? ». Je reconnais que c'est un peu chiant vu que la définition de la ppolicy peut-être un peu n'importe où dans l'arbre LDAP. Globalement l'expérience utilisateur avec ppolicy est loin d'être géniale sur ces points.

#2 - 18 février 2021 11:48 - Benjamin Renard

Concernant les attributs états, je pense utile de définir ce que nous pourrions en faire :

- **pwdChangedTime :** je vois pas bien quand cela pourrait être utile de le récupérer, peut-être lors de l'affichage du message d'expiration du mot de passe *("Vous devez changer votre mot de passe (dernière modification: XXX)")*
- **pwdFailureTime :** utile en cas d'échec de connexion et de mot de passe invalide (on affiche le nombre d'échecs passés)
- **pwdGraceUseTime :** utile pour afficher le nombre d'authentifications autorisées restantes
- **pwdHistory :** je vois pas quand on pourrait en avoir besoin : aucun intérêt à l'affichage et l'historique est complètement géré par le serveur LDAP. Il ne devrait pas être récupéré selon moi.
- **pwdReset :** à la connexion, si *pwdReset* vaut *TRUE* , il doit être renvoyé vers la modification de son mot de passe.
- **pwdUniqueAttempts :** attribut non-standard, il ne devrait pas être récupéré.

Par ailleurs, il doit dans certaine situation être possible d'économiser une requête LDAP spécifique pour récupérer ces attributs : lors d'une connexion

avec le DN de l'utilisateur, on commence par récupérer son DN à partir du login fourni et pour ça, on fait une première requête avec le compte de service. Lors de cette requête, on doit pouvoir récupérer les attributs d'états. Il faudrait cependant prévoir un cas spécifique lorsque cette première connexion n'est pas nécessaire (paramètres *user_dn_template* et *bind_with_username*).

#3 - 18 février 2021 16:17 - Loïc Dachary

- Fichier 0001-ldap-add-method-to-get-ppolicy-operational-attribute.patch ajouté

Un patch corrige:

- un appel de la fonction pour composer les messages d'erreur est ajouter pour montrer concretement l'usage de l'API introduite
- pwdUniqueAttempts est supprimé (c'est très bien expliqué dans le fil de discussion, merci pour le pointeur)
- va chercher les attributs de la ppolicy pour améliorer la lisibilité de "votre mot de passe est trop court" et éviter la question « ok mais quelle longueur doit-il faire doudoudidon ? »
- l'option "ppolicy_dn" est ajoutée parce que "la définition de la ppolicy peut-être un peu n'importe où dans l'arbre LDAP."

Mais qui ne corrige pas:

- "votre mot de passe ne respecte pas les contraintes", « oui mes lesquelles ? » parce que je ne pense pas que cette information soit disponible (ou alors je ne sais pas ou ça se trouve)
- Globalement l'expérience utilisateur avec ppolicy est loin d'être géniale sur ces points parce que je n'ai pas épaules pour m'attaquer à ça.

#4 - 18 février 2021 16:23 - Loïc Dachary

> Par ailleurs, il doit dans certaine situation être possible d'économiser une requête LDAP spécifique ...

En effet et merci de m'avoir montré que ça se passait par ici

#5 - 18 février 2021 16:37 - Loïc Dachary

Less problèmes relevés par Benjamin Renard pour archive:

- la variable ppolicy_dn est mal nommé, cela devrait plutôt être ppolicy_default_dn et surtout, celle peut-être écrasé utilisateur par utilisateur via l'attribut pwdPolicySubentry
- 'The password expired after {pwdmaxage}' => ici pwdmaxage est un entier, ce qui rend le message certainement incompréhensible pour un user normal

#6 - 25 février 2021 09:01 - Loïc Dachary

Pour éviter toute confusion, je vais attendre une revue de code du patch actuel avant d'aller plus loin. Il n'y a pas le feu au lac, c'est juste pour dire :-)

#7 - 04 mars 2021 10:56 - Loïc Dachary

> Je préférerai un ticket global montrant l'usage de l'API introduite, mais tu peux garder celui-ci et avancer sur la suite,...

J'ai créé https://dev.entrouvert.org/issues/51653 dans l'idée de faire un ticket global. Est-ce que c'est ce que tu avais en tête ou bien je suis à coté de la plaque (très possible ;-) ?

#8 - 05 mars 2021 17:55 - Benjamin Dauvergne

- Lié à Development #51653: ldap: fetching and using ppolicy attributes ajouté

#9 - 18 mars 2021 07:57 - Loïc Dachary

Est-ce que le patch proposé ici convient ? https://dev.entrouvert.org/attachments/52068

#10 - 01 avril 2021 18:15 - Loïc Dachary

Sinon je peux le corriger.

#11 - 08 avril 2021 13:21 - Loïc Dachary

- Fichier 0001-ldap-add-method-to-get-ppolicy-operational-attribute.patch ajouté

Rebase sur main, avec résolution du conflit qui empêchait le merge.

#12 - 08 avril 2021 14:00 - Valentin Deniaud

Il y avait plein de conflits parce qu'on a passé le code à la moulinette de black et isort (#52457), il faudrait désormais que tes patches soient conformes : pour cela on a un hook pre-commit de dispo, c'est expliqué dans la section « Code Style » du README.

#13 - 08 avril 2021 14:06 - Loïc Dachary

Je suis vraiment désolé de vous avoir occasionné ce travail supplémentaire. Toutes mes excuses :-(

#14 - 04 mai 2021 11:05 - Valentin Deniaud

Petites remarques sur la forme :

- Enlever le français du message de commit, il a été tapé dans le ticket et c'est très bien comme ça.
- Ne pas rajouter un .format(**attributes) sur les lignes qui n'en ont pas besoin, ça aura aussi l'avantage d'avoir un diff plus light.
- Éviter le time.sleep dans les tests, voir plutôt du côté de la fixture freezer pour ces cas d'usages.
- Il y a un print(attributes_results) qui j'imagine est du debug, à retirer.
- Certaines lignes paraissent vraiment longues, est-ce que black a bien tourné sur le code ?

#15 - 04 mai 2021 11:29 - Loïc Dachary

- Fichier 0001-ldap-add-method-to-get-ppolicy-operational-attribute.patch ajouté

Bonjour et merci pour la review.

- Le message en français a été supprimé du message de commit
- Les `.format(**attributes)` ont été enlevés lorsqu'il n'y a pas d'attributs dans la chaine en version anglaise.
- time.sleep a été laissé parce que le délai concerne le serveur ldap qui implémente la ppolicy et que la fixture n'a pas de contrôle a ce niveau
- le print superflu a été supprimé
- black est passé et a reformaté quelques lignes

#16 - 04 mai 2021 11:50 - Benjamin Dauvergne

Loïc Dachary a écrit :

> - time.sleep a été laissé parce que le délai concerne le serveur ldap qui implémente la ppolicy et que la fixture n'a pas de contrôle a ce niveau

Une attente de 3 secondes ne nous tuera pas, mais juste pour indication: pour des tests plus réalistes du même genre il y a cet outil https://github.com/wolfcw/libfaketime . C'est dispo dans Debian (paquet faketime), à voir si ça serait utilisable avec ldaptools pour lancer slapd (le test peut-être conditionné à la présence de faketime sur le système).

#19 - 18 mai 2021 18:09 - Benjamin Dauvergne

- Tracker changé de Support à Development

- Statut changé de Nouveau à Solution validée

Je vais intégrer ça mais je vais aussi passer use_controls à False par défaut pour éviter un impact sur les installations où ça n'a pas d'usage, il faudra le prendre en compte de votre coté.

#20 - 18 mai 2021 18:15 - Loïc Dachary

- Statut changé de Solution validée à Résolu (à déployer)

- % réalisé changé de 0 à 100

Appliqué par commit authentic2|1f6378256e804f04ac524d89d142f61b014c2466.

#21 - 18 mai 2021 18:21 - Loïc Dachary

Je ne comprends pas le rapport entre le changement de la valeur par défaut de use_control et cette modification ? Je veux dire par la que cette modification ne concerne qu'un chemin de code qui est actif si use_control = True. Ou bien j'ai raté un truc ?

#22 - 18 mai 2021 18:43 - Benjamin Dauvergne

Jusqu'à présent use_controls ne faisait qu'ajouter des contrôles à des requêtes existantes (ou consulter les contrôles sur les retours à simple_bind) mais là ça va générer des requêtes inutiles qui pourrait planter/etc sur d'autres implémentations que OpenLDAP (pour des raisons imprévisibles). De toute façon ça n'a pas d'impact de le désactiver.

#23 - 18 mai 2021 18:52 - Loïc Dachary

> De toute façon ça n'a pas d'impact de le désactiver.

Ca inverse la valeur par défaut et donc le comportement de tous les déploiement qui comptent sur cette valeur à la prochaine mise à jour.

#24 - 18 mai 2021 19:42 - Benjamin Dauvergne

```
commit ff9e90ce5865efc512cdcf0d53a9c847451e38df (HEAD -> main, origin/main)
Author: Benjamin Dauvergne <bdauvergne@entrouvert.com>
Date:   Tue May 18 18:10:14 2021 +0200

    ldap: default use_controls to False (#51239)

commit 1f6378256e804f04ac524d89d142f61b014c2466
Author: Loïc Dachary <ldachary@easter-eggs.com>
Date:   Fri Feb 12 19:02:26 2021 +0100

    ldap: add method to get ppolicy operational attributes (#51239)

    Fixes: #51239

    License: MIT
```

#25 - 21 mai 2021 09:32 - Frédéric Péters

- Statut changé de Résolu (à déployer) à Solution déployée

Fichiers

0001-ldap-get_ppolicy_attributes.patch	3,16 ko	18 février 2021	Loïc Dachary
0001-ldap-add-method-to-get-ppolicy-operational-attribute.patch	9,25 ko	18 février 2021	Loïc Dachary
0001-ldap-add-method-to-get-ppolicy-operational-attribute.patch	10,3 ko	08 avril 2021	Loïc Dachary
0001-ldap-add-method-to-get-ppolicy-operational-attribute.patch	8,94 ko	04 mai 2021	Loïc Dachary