Authentic 2 - Development #34316

ldap : 'NoneType' object has no attribute 'search_s'

25 juin 2019 14:04 - Thomas Noël

Statut:	Fermé	**Début:**	25 juin 2019
Priorité:	Normal	**Echéance:**	
Assigné à:	Benjamin Dauvergne	**% réalisé:**	0%
Catégorie:		**Temps estimé:**	0:00 heure
Version cible:			
Patch proposed:	Oui	**Planning:**	Non

Description

En cas de pépin pour accéder à un LDAP, on a cette première alerte "normale", prévue dans le code :

```
ERROR: could not get a connection
```

mais juste derrière, une seconde, car aucune connexion LDAP n'existe et donc "conn == None" :

```
File "/usr/lib/python2.7/dist-packages/authentic2/idp/saml/saml2_endpoints.py" in sso
  539.     return sso_after_process_request(request, login, nid_format=nid_format)

File "/usr/lib/python2.7/dist-packages/authentic2/idp/saml/saml2_endpoints.py" in sso_after_proces
s_request
  823.     add_attributes(request, login.assertion, provider)

File "/usr/lib/python2.7/dist-packages/authentic2/idp/saml/saml2_endpoints.py" in add_attributes
  223.         '__wanted_attributes': wanted_attributes,

File "/usr/lib/python2.7/dist-packages/authentic2/attributes_ng/engine.py" in get_attributes
  112.         ctx.update(source.get_attributes(instance, ctx.copy()))

File "/usr/lib/python2.7/dist-packages/authentic2/attributes_ng/sources/ldap.py" in get_attributes
  41.         ctx.update(user.get_attributes())

File "/usr/lib/python2.7/dist-packages/authentic2/backends/ldap_backend.py" in get_attributes
  370.         return self.ldap_backend.get_ldap_attributes(self.block, conn, self.dn) or {}

File "/usr/lib/python2.7/dist-packages/authentic2/backends/ldap_backend.py" in get_ldap_attributes
  948.             results = conn.search_s(dn, ldap.SCOPE_BASE, u'(objectclass=*)', attributes)

Exception Type: AttributeError at /idp/saml2/sso
Exception Value: 'NoneType' object has no attribute 'search_s'
Request information:
USER: xxxx@example.net (9befc3)
```

Sans doute faut-il mieux gérer le fait que conn puisse être None, et arrêter les frais au plus tôt.

Révisions associées

Révision dc93a544 - 01 juillet 2019 09:53 - Benjamin Dauvergne

ldap: do not block check_password and get_attributes if LDAP is down (#34316)

Historique

#1 - 25 juin 2019 14:56 - Benjamin Dauvergne

- Assigné à mis à Benjamin Dauvergne

#2 - 26 juin 2019 11:15 - Benjamin Dauvergne

- Fichier 0001-ldap-do-not-block-check_password-and-get_attributes-.patch ajouté

- Tracker changé de Support à Development

- Statut changé de Nouveau à Solution proposée

- Patch proposed changé de Non à Oui

Je suis passé sur deux méthodes qui dépendent de l'existence d'une connection (il faudrait pousser la couverture à 100% pour en trouver d'autres mais c'est les deux qui m'ont sauté aux yeux).

- get_attributes() pour rendre le souci encore moins impactant je mets les attributs en cache en fonction du user.pk et du dn pendant 8 heures
- check_password() je réponds faut par défaut si pas de connecion, mais je répond True si jamais on a mis en cache le mot de passe sur l'objet utilisateur

#3 - 26 juin 2019 13:13 - Thomas Noël

Dans les tests, après avoir coupé le ldap je m'attendais pas à « assert not user.check_password(UPASS) » ... l'idée n'était pas que comme le mot de passe reste le même, ça passe ?

#4 - 27 juin 2019 09:57 - Thomas Noël

Petite demande au passage puisque ce patch gère les soucis de connexion LDAP : on pourrait y remplacer le message "could not get a connection" par "could not get a LDAP connection" ?

#5 - 27 juin 2019 10:58 - Benjamin Dauvergne

- Statut changé de Solution proposée à En cours

#6 - 27 juin 2019 11:55 - Benjamin Dauvergne

- Fichier 0001-ldap-do-not-block-check_password-and-get_attributes-.patch ajouté

- Statut changé de En cours à Solution proposée

J'ai abandonné l'idée de tester le mot de passe via la version en cache, de toute façon ça ne sert qu'à la page de changement de mot de passe qui :

- soit est bloquée parce qu'on a pas le droit de les changer
- soit ne marchera de toute façon pas vu que le LDAP est down

J'ai modifié tous les "could not get a connection" qui deviennent des warnings et commencent tous par "ldap: ".

#7 - 01 juillet 2019 08:34 - Thomas Noël

Une question posée via jabber : pourquoi stocker les attributs du LDAP dans un cache à part, et pas dans la session ?

Réponse de Benjamin :
1. j'aime pas trop utiliser la sesssion comme cache
2. si tu mets jpegPhoto dans les attributs ça va partir en sucette
3. la session était pas vraiment accessible à cet endroit là (c'est pas vrai j'ai un RequestMiddleware qui stoque la requête en variable globale comme tout le monde mais bon) ; maintenant les backends sont censés recevoir une requête aussi
4. les sessions c'est du JSON ça peut poser souci si tout n'est pas convertible en unicode (le cache c'est du pickle (ça pose d'autres soucis dont on se fout pour l'instant))

Sachant qu'ici on palie à de très mauvaises pannes du LDAP, on fait "ce qu'on peut". La bonne solution c'est quand même que le LDAP soit plus stable que 1% du temps.

#8 - 01 juillet 2019 08:59 - Thomas Noël

- Statut changé de Solution proposée à Solution validée

Ack sur la version dans la branche (qui n'est pas celle attachée plus haut)

#9 - 01 juillet 2019 09:54 - Benjamin Dauvergne

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

```
commit dc93a5445c6039449a217d0ad795d0d0e4ff0cdd
Author: Benjamin Dauvergne <bdauvergne@entrouvert.com>
Date:   Tue Jun 25 23:23:11 2019 +0200

    ldap: do not block check_password and get_attributes if LDAP is down (#34316)
```

#10 - 02 juillet 2019 16:15 - Frédéric Péters

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

Fichiers

0001-ldap-do-not-block-check_password-and-get_attributes-.patch	6,06 ko	26 juin 2019	Benjamin Dauvergne
0001-ldap-do-not-block-check_password-and-get_attributes-.patch	7,57 ko	27 juin 2019	Benjamin Dauvergne