Hobo - Development #4573

Ajouter de l'authent

26 mars 2014 09:52 - Frédéric Péters

Statut:	Fermé	**Début:**	26 mars 2014
Priorité:	Haut	**Echéance:**	17 novembre 2014
Assigné à:	Serghei Mihai (congés, retour 15/05)	**% réalisé:**	0%
Catégorie:		**Temps estimé:**	0:00 heure
Version cible:			
Patch proposed:	Oui	**Planning:**	

Description

(encore SAMLv2 a priori)

Historique

#1 - 31 mars 2014 14:44 - Serghei Mihai (congés, retour 15/05)

ou OAuth2

#2 - 06 avril 2014 11:37 - Serghei Mihai (congés, retour 15/05)

- Assigné à mis à Serghei Mihai (congés, retour 15/05)

#3 - 07 juillet 2014 17:27 - Frédéric Péters

- Projet changé de Portail admin à Hobo

- Patch proposed mis à Non

#4 - 29 août 2014 11:00 - Frédéric Péters

Dans un message sur la liste, j'écris :

> Pour l'authentification, le déroulé devrait être selon moi le suivant, d'abord une connexion initiale par username/password, qui aurait été défini par un appel à createsuperuser, ensuite l'admin ajoute un authentic et celui-ci prend la main, SSO etc.

#5 - 14 novembre 2014 15:45 - Jérôme Schneider

- Priorité changé de Normal à Haut

Je le passe en priorité haute car c'est vraiment la fonctionnalité qui manque.

#6 - 14 novembre 2014 16:21 - Serghei Mihai (congés, retour 15/05)

- Statut changé de Nouveau à En cours

#7 - 14 novembre 2014 16:31 - Thomas Noël

Et dans une discussion je-ne-sais-plus-quand-ni-où, on a causé de django-mellon pour l'auth (en fait c'était pour passerelle, mais ça doit se faire pour hobo, àmha)

#8 - 17 novembre 2014 10:25 - Serghei Mihai (congés, retour 15/05)

- Echéance mis à 17 novembre 2014

#9 - 17 novembre 2014 18:25 - Serghei Mihai (congés, retour 15/05)

- Fichier 0001-mandatory-authentication-added.patch ajouté

- Fichier 0002-SAML-authentication-via-django-mellon-added.patch ajouté

- Patch proposed changé de Non à Oui

Authentification locale et via django-mellon ajoutées.
J'ai séparé la configuration de django-mellon afin qu'elle soit désactivée par défaut.

#10 - 17 novembre 2014 19:08 - Frédéric Péters

Il y avait un truc pour appliquer login_required à toute une hiérarchie, c'est utilisé dans Passerelle, decorated_includes(login_required, ...); ça me semble plus propre que devoir ajouter des login_required; ça ne pourrait pas être utilisé ici ?

~~~

Il reste toujours un truc que je n'aime pas dans la manière dont les settings se font ici, le fonctionnement que j'aimais bien, c'était :

```
settings.py
....

# un tas de trucs qu'on ne voudra jamais changer, que je trouve stupide de copier/coller de configs en configs
# et si ça pouvait être une valeur par défaut qui fonctionne, qu'on ne soit pas obligé de mettre ça pour fonct
ionner
# ce serait encore mieux.
MELLON_ATTRIBUTE_MAPPING = {
    'username': '{attributes[email][0]}',
    'email': '{attributes[email][0]}',
    'first_name': '{attributes[first_name][0]}',
    'last_name': '{attributes[last_name][0]}',
}

from local_settings import *  # dans ce fichier un ENABLE_MELLON = True

if ENABLE_MELLON:
  # un tas de trucs qui sont toujours pareils quand on décide d'activer Mellon
  INSTALLED_APPS += ('mellon',)
  AUTHENTICATION_BACKENDS = ( 'mellon.backends.SAMLBackend',)
  LOGIN_URL = 'mellon_login'
  LOGOUT_URL = 'mellon_logout'
```

Bref, est-ce qu'avec la manière dont on gère les settings aujourd'hui, il y a moyen d'approcher ça, de ne pas devoir recopier de fichiers de configs en fichiers de configs des trucs comme MELLON_ATTRIBUTE_MAPPING ou LOGIN_URL = 'mellon_login', d'avoir uniquement à définir les paramètres pour les valeurs qui ne se devinent pas (genre l'emplacement d'une clé) ?

#11 - 18 novembre 2014 10:28 - Jérôme Schneider

> Bref, est-ce qu'avec la manière dont on gère les settings aujourd'hui, il y a moyen d'approcher ça, de ne pas devoir recopier de fichiers de configs en fichiers de configs des trucs comme MELLON_ATTRIBUTE_MAPPING ou LOGIN_URL = 'mellon_login', d'avoir uniquement à définir les paramètres pour les valeurs qui ne se devinent pas (genre l'emplacement d'une clé) ?

Pour moi mais Benjamin ou Thomas vont peut être me corriger. Pour atteindre ce que tu proposes avec la nouvelle configuration il faut mettre MELLON_ATTRIBUTE_MAPPING et "un tas de trucs qu'on ne voudra jamais changer" dans le default_settings.py. Pour ce qui est du ENABLE_MELLON la nouvelle politique est de ne pas ajouter de magie. Donc on aurait quelque chose comme ça dans le config_example.py :

```
# Dé-commenter les lignes suivantes pour activer le support de Mellon
# un tas de trucs qui sont toujours pareils quand on décide d'activer Mellon
#INSTALLED_APPS += ('mellon',)
#AUTHENTICATION_BACKENDS = ( 'mellon.backends.SAMLBackend',)
#LOGIN_URL = 'mellon_login'
#LOGOUT_URL = 'mellon_logout'
```

De mon côté j'ai tendance à ne pas spécialement apprécier la nouvelle configuration. Les fichiers de configuration que j'ai du faire récemment ne me convienne pas. Je pense qu'une discussion sur le sujet serait intéressante. Je pense qu'il serait utile qu'on récupère un peu de "magie" au moins dans le debian.py comme par exemple pour le sentry (#5847) mais ENABLE_MELLON est un autre exemple. Pour Montpellier je n'ai pas voulu utiliser le config.py pour le portail citoyen et authentic car pour moi on est pas prêt pour le moment. Les fichiers de configuration devienne vite trop long et pas spécialement lisible.
Un autre point moins important c'est que je ne vois pas l'intérêt d'avoir un default_settings.py + un settings.py pour moi tout pourrait rentrer dans le settings.py.

#12 - 18 novembre 2014 10:37 - Serghei Mihai (congés, retour 15/05)

- Fichier 0001-mandatory-authentication-added.patch ajouté

- Fichier 0002-SAML-authentication-via-django-mellon-disabled-by-de.patch ajouté

- Fichier 0001-Mellon-authentication-disabled-by-default.patch ajouté

Oui, avec un decorateur sur le include c'est mieux.

En effet, les paramètres par défaut de MELLON auxquels on ne touchera probablement jamais doivent être dans le default_settings.py
Quant à l'activation de l'authentification avec mellon, je la verrais déclarée dans le debian_config.py, qui pourrait être écrasée par le config.py
~~~

#13 - 19 novembre 2014 11:11 - Jérôme Schneider

Pour moi le patch dans le settings.py n'est pas bon. Vu la politique actuelle il ne faut pas y toucher. En plus tu utilises une variable ETC_DIR qui n'est pas défini (sauf dans le paquet debian).

A la place de cette modif dans le settings.py je verrais plus un ajout dans config_example.py de :

```
## Django Mellon configuration
# you need to generate SSL certificates in your current directory to make it functionnal :
# openssl genpkey -algorithm rsa -pkeyopt rsa_keygen_bits:2048 -out key.cert
# openssl req -x509 -new -out cert.pem -subj '/CN=whocares' -key key.cert -days 3650
# you also need to get the idp metadata and call it idp-metadata.xml
# Uncomment the following lines to enable SAML support
#INSTALLED_APPS += ('mellon',)
#AUTHENTICATION_BACKENDS = ( 'mellon.backends.SAMLBackend',)
#LOGIN_URL = 'mellon_login'
#LOGOUT_URL = 'mellon_logout'
#MELLON_PUBLIC_KEYS = ['cert.pem']
#MELLON_PRIVATE_KEY = 'key.cert'
#MELLON_IDENTITY_PROVIDERS = [
#            {'METADATA': 'idp-metadata.xml',
#             'GROUP_ATTRIBUTE': 'role'},
#        ]
```

Pour la partie Debian je pense qu'on peut l'activer par défaut en rajoutant les lignes suivantes dans le debian/debian_config.py ces lignes :

```
        INSTALLED_APPS += ('mellon',)
        AUTHENTICATION_BACKENDS = ( 'mellon.backends.SAMLBackend',)
        LOGIN_URL = 'mellon_login'
        LOGOUT_URL = 'mellon_logout'
        MELLON_PUBLIC_KEYS = [os.path.join(ETC_DIR, 'cert.pem')]
        MELLON_PRIVATE_KEY = os.path.join(ETC_DIR, 'key.cert')
        MELLON_IDENTITY_PROVIDERS = [
            {'METADATA': os.path.join(ETC_DIR, 'idp-metadata.xml'),
             'GROUP_ATTRIBUTE': 'role'},
        ]
```

Il faudrait également rajouter la génération des certificats SSL dans le postinst et rajouter un idp-metadata.xml qui soit celui d'identity hub.

#14 - 20 novembre 2014 18:20 - Serghei Mihai (congés, retour 15/05)

- Fichier 0001-only-superusers-can-login.patch ajouté

- Fichier 0002-authentication-via-django-mellon-example-added.patch ajouté

- Fichier 0001-mellon-authentication-enabled-by-default-with-identi.patch ajouté

Corrections apportées.
J'ai rajouté également le check si l'utilisateur est admin

#15 - 25 novembre 2014 15:44 - Frédéric Péters

Je vois que ces patchs ont été poussés.

#16 - 25 novembre 2014 17:33 - Serghei Mihai (congés, retour 15/05)

- Statut changé de En cours à Solution déployée

#17 - 09 novembre 2015 12:26 - Benjamin Dauvergne

- Statut changé de Solution déployée à Fermé

Fichiers

0001-mandatory-authentication-added.patch	6,92 ko	17 novembre 2014	Serghei Mihai (congés, retour 15/05)
0002-SAML-authentication-via-django-mellon-added.patch	4,74 ko	17 novembre 2014	Serghei Mihai (congés, retour 15/05)
0001-mandatory-authentication-added.patch	6,41 ko	18 novembre 2014	Serghei Mihai (congés, retour 15/05)
0002-SAML-authentication-via-django-mellon-disabled-by-de.patch	2,27 ko	18 novembre 2014	Serghei Mihai (congés, retour 15/05)
0001-Mellon-authentication-disabled-by-default.patch	744 octets	18 novembre 2014	Serghei Mihai (congés, retour 15/05)
0001-only-superusers-can-login.patch	3,32 ko	20 novembre 2014	Serghei Mihai (congés, retour 15/05)
0002-authentication-via-django-mellon-example-added.patch	2,3 ko	20 novembre 2014	Serghei Mihai (congés, retour 15/05)
0001-mellon-authentication-enabled-by-default-with-identi.patch	5,25 ko	20 novembre 2014	Serghei Mihai (congés, retour 15/05)