Authentic 2 - Development #15146

API de génération d'un lien de connexion automatique

24 février 2017 14:46 - Benjamin Renard

Statut:	Solution proposée	**Début:**	24 février 2017
Priorité:	Normal	**Echéance:**	
Assigné à:	Benjamin Renard	**% réalisé:**	0%
Catégorie:		**Temps estimé:**	0:00 heure
Version cible:			
Patch proposed:	Oui	**Planning:**	

Description

Le but est de permettre de générer des liens de connexion automatique pour un utilisateur donné. A l'aide de ce lien, il sera alors possible de se connecter en tant qu'un utilisateur donné sans avoir à fournir de login/mot de passe. Le lien généré peut inclure une URL de redirection automatique de l'utilisateur une fois celui-ci identifié. Le lien généré a une durée de validité déterminée au moment de la génération soit à l'aide d'une valeur passée en paramètre de l'API, soit à défaut à l'aide d'une valeur par défaut. Dans le patch proposé celle-ci est hard-codée (5 minutes) mais on pourrait imaginer un paramètre de configuration pour cela.

Historique

#1 - 24 février 2017 16:07 - Benjamin Dauvergne

1. signing.dumps() peut directement gérer les problèmes d'expiration, il faudrait se servir de cette API
2. plutôt qu'autologin je verrai bien la méthode d'authentification s'appeler token dans l'appel à login()
3. je serai d'avis d'avoir 2 settings: la valeur de la durée par défaut du token, et une valeur maximum permise: A2_AUTH_TOKEN_DEFAULT_LIFETIME et A2_AUTH_TOKEN_MAX_LIFEMENT en secondes
4. plutôt que plein de logging.getLogger(__name__) un self.logger = logging.getLogger(__name__) une bonne fois pour toute dans la méthode __init__ des vues
5. ça m'embête un peu que ce soit restreint à l'email il faudrait au minimum pouvoir passer l'uuid qui lui est précis, en fait la recherche par m'email devrait être faire ds la génération du token pour ne pas que ce soit l'utilisateur qui soit surpris que le token ne marche pas, le token devrait lui contenir l'uuid seulement.
6. le paramètre de redirection s'appelle redirect_url je n'ai pas regardé mais je pense que c'est next_url dans les autres APIs
7. pour plus de sécurité je verrai bien le jeton expirer après un certains temps suivant sa première utilisation mais bien avant son expiration d'utilisation, en stockant dans le cache pour la durée correspondant à la durée de vie du jeton la date de première utilisation cache.set('token-%s' % sha1(token), now().isoformat(), settings.A2_AUTH_TOKEN_LIFETIME) avec un nouveau setting A2_AUTH_TOKEN_LIFETIME_AFTER_FIRST_USE par défaut à 3600.

#2 - 04 mai 2017 11:56 - Benjamin Renard

- Fichier 0001-Add-API-method-to-generate-auto-login-link.patch ajouté

Benjamin Dauvergne a écrit :

> 1. signing.dumps() peut directement gérer les problèmes d'expiration, il faudrait se servir de cette API

Sauf erreur, la gestion du paramètre max_age qui permet cela est fourni et évalué au moment du loads() et non du dumps() et vue qu'ici la durée d'expiration peut être différent d'un token à l'autre, je ne pense pas qu'on puisse utiliser ce mécanisme.

> 2. plutôt qu'autologin je verrai bien la méthode d'authentification s'appeler token dans l'appel à login()

C'est modifié.

> 3. je serai d'avis d'avoir 2 settings: la valeur de la durée par défaut du token, et une valeur maximum permise: A2_AUTH_TOKEN_DEFAULT_LIFETIME et A2_AUTH_TOKEN_MAX_LIFEMENT en secondes

C'est en place.

> 4. plutôt que plein de logging.getLogger(__name__) un self.logger = logging.getLogger(__name__) une bonne fois pour toute dans la méthode __init__ des vues

C'est fait.

> 5. ça m'embête un peu que ce soit restreint à l'email il faudrait au minimum pouvoir passer l'uuid qui lui est précis, en fait la recherche par m'email devrait être faire ds la génération du token pour ne pas que ce soit l'utilisateur qui soit surpris que le token ne marche pas, le token

devrait lui contenir l'uuid seulement.

C'est fait. J'ai offert la possibilité de fournir l'email ou l'uuid au moment de la génération du token mais c'est l'uuid qui est conservé dans tous les cas dans le token.

6. le paramètre de redirection s'appelle redirect_url je n'ai pas regardé mais je pense que c'est next_url dans les autres APIs

Exact, c'est corrigé.

7. pour plus de sécurité je verrai bien le jeton expirer après un certains temps suivant sa première utilisation mais bien avant son expiration d'utilisation, en stockant dans le cache pour la durée correspondant à la durée de vie du jeton la date de première utilisation cache.set('token-%s' % sha1(token), now().isoformat(), settings.A2_AUTH_TOKEN_LIFETIME) avec un nouveau setting A2_AUTH_TOKEN_LIFETIME_AFTER_FIRST_USE par défaut à 3600.

C'est en place.

Le nouveau patch est joint.

#3 - 02 juillet 2018 12:28 - Benjamin Dauvergne

- Statut changé de Nouveau à Solution proposée

- Assigné à mis à Benjamin Renard

Fichiers

0001-Add-API-method-to-generate-auto-login-link.patch	6,91 ko	24 février 2017	Benjamin Renard
0001-Add-API-method-to-generate-auto-login-link.patch	9,3 ko	04 mai 2017	Benjamin Renard