Authentic 2 - Development #24056

interrompre la connexion en cas de nouveau champ obligatoire (/demandé à l'inscription)

24 mai 2018 13:20 - Frédéric Péters

Statut:	Fermé	**Début:**	24 mai 2018
Priorité:	Normal	**Echéance:**	
Assigné à:	Benjamin Dauvergne	**% réalisé:**	0%
Catégorie:		**Temps estimé:**	0:00 heure
Version cible:			
Patch proposed:	Oui	**Planning:**	Non

Description

Quand dans un profil on définit un nouveau champ obligatoire, il faudrait qu'à la connexion suivante on arrête l'usager sur une page lui demandant de remplir ces champs.

(ça peut être particulièrement utile pour arrêter l'usager après l'ajout d'un champ "acceptation des cgu (nouvelle version)")

Demandes liées:

Lié à Hobo - Development #55865: Ajouter la case à cocher « Requis à la conne...	**Fermé**	**27 juillet 2021**

Révisions associées

Révision 5a13211f - 23 juillet 2021 17:15 - Benjamin Dauvergne

misc: add a required_on_login flag on Attribute (#24056)

Révision 34cd3c1d - 23 juillet 2021 17:15 - Benjamin Dauvergne

misc: refactor ViewRestrictionMiddleware (#24056)

The generic code for checking restrictions is separated from the code
specific to each specific check; here the code to check for
PasswordReset models is extracted into
check_password_reset_view_restriction().

Révision c5e5a14a - 23 juillet 2021 17:15 - Benjamin Dauvergne

misc: block user without required_on_login attributes (#24056)

Superuser are exempted from the restriction.

Historique

#1 - 12 juin 2018 11:30 - Benjamin Dauvergne

La mécanique pour bloquer la navigation sur un critère et rediriger sur une autre page est déjà là (authentic2.middleware.ViewRestrictionMiddleware).

Actuellement chaque plugin a2 peut proposer via une méthode check_view_restriction() qui doit retourner le nom d'une vue ou une URL. Vu que le concept de plugin est en cours de déprécation pour utiliser simplement des applications Django et leur objet AppConfig à la place il faudrait ici plutôt passer par un hook (donc modifier ViewRestrictionMiddleware pour appeler call_hooks('check_view_restriction')).

Ensuite il suffit de renvoyer l'URL de la vue de profil et de poser un message sur la requête du genre "Vous devez valider les CGUs"[1].

Ce qui m'embête c'est comment générer ce message de manière automatique et intelligible, on génèrera facilement un message du genre "Veuillez compléter les champs requis suivants: date de naissance, validation des CGUs" mais est-ce que c'est suffisant ?

[1]

```
def a2_hook_check_view_restriction(self, request):
     if 'required_fields' in request.session:
         return
     # on vérifie si tous les champs requis sont là
     if unfilled_required_fields:
        message = _('You must complete required fields: %s') % ', '.join([field.label for field in unfilled_re
quired_fields])
        messages.info(request, message)
        return 'profile_edit'
     request.session['required_fields'] = True
     return None
```

C'est bien si on ne valide une restriction qu'une fois par session (sinon ça fait des requêtes intempestives en base pour rien), i.e. si une restriction de vue passe, il faut le noter dans request.session pour ne pas le refaire.

#3 - 18 septembre 2018 14:11 - Brice Mallet

> Ce qui m'embête c'est comment générer ce message de manière automatique et intelligible, on génèrera facilement un message du genre "Veuillez compléter les champs requis suivants: date de naissance, validation des CGUs" mais est-ce que c'est suffisant ?

Dans le contexte du cas d'usage "acceptation des CGU" pour un compte créé avant obligation de celui-ci, cela me semble OK

#5 - 25 mai 2021 12:04 - Benjamin Dauvergne

- Assigné à mis à Benjamin Dauvergne

- Planning mis à Non

C'est pour moi, joie.

#7 - 16 juillet 2021 11:45 - Benjamin Dauvergne

- Fichier 0003-misc-block-user-without-required_on_login-attributes.patch ajouté

- Fichier 0002-misc-refactor-ViewRestrictionMiddleware-24056.patch ajouté

- Fichier 0001-misc-add-a-required_on_login-flag-on-Attribute-24056.patch ajouté

- Statut changé de Nouveau à Solution proposée

- Patch proposed changé de Non à Oui

Ça n'aura d'impact que là où c'est activé (en posant required_on_login sur un attribut); sera testé dans un premier temps sur GLC, l'habillage y sera fait via un template custom (pour mettre le lien vers les CGUs dans le label du champ).

#8 - 16 juillet 2021 15:35 - Paul Marillonnet

Dans 0002 :

- j'ai fait un tour (à coup de find -exec grep) des vues concernées par

  ```
  if 'logout' in url_name or '-slo' in url_name:
  ```

  Ok pour le logout.
  Le SLO c'est les vues de authentic2.idp.saml qui vont être concernées. En renommant a2-idp-saml-finish-slo en a2-idp-saml-slo-finish on peut avoir un test un peu plus carré à base de .startswith('a2-idp-saml-slo').
  Autre chose dans authentic2.idp.saml.saml2_enpdoints il me semble qu'il y a plein d'autres vues frontchannel qui ne contiennent pas logout ni -slo et qui pourtant doivent être exemptées de cette redirection vers la page de complétion des attributs manquants par l'usager.

Dans 0003 :

- avec la nouvelle méthode de classe EditRequired.get_fields on perd la logique de vérification des scopes des attributs. L'usager peut donc se voir présenter des attributs auxquels il n'a pas accès dans sa page standard d'édition de profil (EditProfile). Étrange.

De façon plus générale, à la généralisation de ces attributs requis au login, je me pose la question des perfs lorsqu'on tape cette vérification dans le middleware plutôt que d'isoler les vues candidates une à une. Mais bon la valeur de 60 secondes minimum entre deux vérifications en succès deux paraît raisonnable.

(Rien à voir, mais un petit s/succesfull/successful/ dans une ligne de commentaire de 0002.)

#9 - 16 juillet 2021 15:54 - Benjamin Dauvergne

Paul Marillonnet a écrit :

> Dans 0002 :
>
> - j'ai fait un tour (à coup de find -exec grep) des vues concernées par [...] Ok pour le logout.
>   Le SLO c'est les vues de authentic2.idp.saml qui vont être concernées. En renommant a2-idp-saml-finish-slo en a2-idp-saml-slo-finish on peut avoir un test un peu plus carré à base de .startswith('a2-idp-saml-slo').

La vue finish n'est pas concernée, c'est uniquement les vues de SLO initiée par le SP, pas toutes les vues frontchannel, disons que que ça pourrait arriver entre le moment d'émettre un SLO et de recevoir sa réponse, mais je préfère ignorer le problème.

> Dans 0003 :

- avec la nouvelle méthode de classe EditRequired.get_fields on perd la logique de vérification des scopes des attributs. L'usager peut donc se voir présenter des attributs auxquels il n'a pas accès dans sa page standard d'édition de profil (EditProfile). Étrange.

Les scopes ne servent pas à cacher des attributs, donc non ça ne sert à rien ici, on peut l'ignorer sans souci.

> De façon plus générale, à la généralisation de ces attributs requis au login, je me pose la question des perfs lorsqu'on tape cette vérification dans le middleware plutôt que d'isoler les vues candidates une à une. Mais bon la valeur de 60 secondes minimum entre deux vérifications en succès deux paraît raisonnable.

Je ne pense pas que ce soit un problème.

#10 - 16 juillet 2021 16:16 - Paul Marillonnet

Benjamin Dauvergne a écrit :

> Paul Marillonnet a écrit :
>
> > Dans 0002 :
> >
> > - j'ai fait un tour (à coup de find -exec grep) des vues concernées par [...] Ok pour le logout.
> >   Le SLO c'est les vues de authentic2.idp.saml qui vont être concernées. En renommant a2-idp-saml-finish-slo en a2-idp-saml-slo-finish on peut avoir un test un peu plus carré à base de .startswith('a2-idp-saml-slo').
>
> La vue finish n'est pas concernée, c'est uniquement les vues de SLO initiée par le SP, pas toutes les vues frontchannel, disons que que ça pourrait arriver entre le moment d'émettre un SLO et de recevoir sa réponse, mais je préfère ignorer le problème.

Où est-ce que tu vois que toutes les vues frontchannel ne vont pas tomber dans cette logique de vérification/redirection ? J'ai loupé ce passage dans le patch.

> Dans 0003 :
>
> - avec la nouvelle méthode de classe EditRequired.get_fields on perd la logique de vérification des scopes des attributs. L'usager peut donc se voir présenter des attributs auxquels il n'a pas accès dans sa page standard d'édition de profil (EditProfile). Étrange.
>
> Les scopes ne servent pas à cacher des attributs, donc non ça ne sert à rien ici, on peut l'ignorer sans souci.

Et pourtant c'est ce que je comprends à la lecture de EditProfile.get_fields :

```
        # […]
        if scopes:
            scopes = set(scopes)
            default_fields = [
                attribute.name for attribute in attributes if scopes & set(attribute.scopes.split())
            ]
        # […]
        if scopes:
            # restrict fields to those in the scopes
            fields = [field for field in fields if field in default_fields]
        # […]
```

lesquels fields sont ensuite servis au modelform_factory. Je loupe un truc ?

> De façon plus générale, à la généralisation de ces attributs requis au login, je me pose la question des perfs lorsqu'on tape cette vérification dans le middleware plutôt que d'isoler les vues candidates une à une. Mais bon la valeur de 60 secondes minimum entre deux vérifications en succès deux paraît raisonnable.
>
> Je ne pense pas que ce soit un problème.

Ok, fair enough.

#11 - 16 juillet 2021 17:51 - Benjamin Dauvergne

- Fichier 0003-misc-block-user-without-required_on_login-attributes.patch ajouté

- Fichier 0002-misc-refactor-ViewRestrictionMiddleware-24056.patch ajouté

- Fichier 0001-misc-add-a-required_on_login-flag-on-Attribute-24056.patch ajouté

Après réflexion j'ai changé mon fusil d'épaule, je passe plutôt par un flag directement sur les vues liées au logout. Le logout SOAP est hors session donc n'est pas concerné.

#12 - 16 juillet 2021 17:54 - Benjamin Dauvergne

Paul Marillonnet a écrit :

> Et pourtant c'est ce que je comprends à la lecture de EditProfile.get_fields :
> [...] lesquels fields sont ensuite servis au modelform_factory. Je loupe un truc ?

Le scope est un truc passé en paramètre de la vue, c'est pour limiter le formulaire à un certain ensemble de champs pour des raisons d'organisation, à Strasbourg je crois (genre séparer les champs d'adresse des champs d'état civil). Je ne vois pas de lien avec la fonctionnalité ici implémentée.

#13 - 19 juillet 2021 10:07 - Paul Marillonnet

Benjamin Dauvergne a écrit :

> Le scope est un truc passé en paramètre de la vue, c'est pour limiter le formulaire à un certain ensemble de champs pour des raisons d'organisation, à Strasbourg je crois (genre séparer les champs d'adresse des champs d'état civil). Je ne vois pas de lien avec la fonctionnalité ici implémentée.

D'ac je n'avais pas connaissance de cet usage.

#14 - 19 juillet 2021 10:15 - Paul Marillonnet

Benjamin Dauvergne a écrit :

> Après réflexion j'ai changé mon fusil d'épaule, je passe plutôt par un flag directement sur les vues liées au logout. Le logout SOAP est hors session donc n'est pas concerné.

Oui sur cette épaule ça vise plus juste :)
Juste une remarque de détail avant le ack : du code churn entre 0002 et 0003 qu'on pourrait s'éviter, genre dans 0002 on introduit

```
+        # prevent blocking people when they logout
+        if 'logout' in url_name or '-slo' in url_name:
```

qui est une idée finalement non retenue et qu'on vient ensuite changer complètement à nouveau dans 0003.
(Je pense qu'on peut laisser ce bout de méthode process_view inchangé dans 0002 et balancer la version finale directement dans 0003.)

#15 - 23 juillet 2021 11:59 - Benjamin Dauvergne

- Fichier 0003-misc-block-user-without-required_on_login-attributes.patch ajouté

- Fichier 0004-to-be-fixed-up-use-a-decorator-to-indicate-views-whe.patch ajouté

- Fichier 0002-misc-refactor-ViewRestrictionMiddleware-24056.patch ajouté

- Fichier 0001-misc-add-a-required_on_login-flag-on-Attribute-24056.patch ajouté

(J'ai pris en compte ta remarque sur le code churn, dans la refactorisation je ne garde que le comportement existant, i.e. view == 'auth_logout', donc juste de la refactorisation sans changement de comportement).

Encore un changement, j'ai préféré une approche opt-in, désormais les vues qui sont des points sûrs pour arrêter le parcours de l'utilisateur sont explicitement indiquées, ça concerne :

- la homepage (qui dans 99% des cas fait un redirect vers combo, et le 1% restant vers /accounts/ sur GLC)
- /accounts/
- toutes les vues de SSO ou de terminaison de SSO quand le comportement est coupé en deux comme pour CAS et SAML (sso, login, authorize, continue..)

#16 - 23 juillet 2021 12:26 - Benjamin Dauvergne

J'ai corrigé le bug dans l'IdP SAML sur make_url(continue_sso, ...).

#17 - 23 juillet 2021 15:22 - Paul Marillonnet

Benjamin Dauvergne a écrit :

> (J'ai pris en compte ta remarque sur le code churn, dans la refactorisation je ne garde que le comportement existant, i.e. view == 'auth_logout', donc juste de la refactorisation sans changement de comportement).
>
> Encore un changement, j'ai préféré une approche opt-in, désormais les vues qui sont des points sûrs pour arrêter le parcours de l'utilisateur sont

explicitement indiquées, ça concerne :
[…]

Oui en mode opt-in c'est bien aussi.
Je ne comprends pas l'idée du décorateur cependant. Pour moi ce flag view_restriction c'est inhérent à la vue, et non quelque chose qui s'applique à une ou plusieurs de ses occurrences dans les urls.py.
Du coup avec le décorateur ça donne du code de copie qui il me semble n'a pas lieu d'être : on ne se retrouve pas dans le cas où deux versions (l'une originale, l'autre copiée-décorée) de la vue coexistent.

#18 - 23 juillet 2021 15:48 - Benjamin Dauvergne

- Fichier 0003-misc-block-user-without-required_on_login-attributes.patch ajouté

- Fichier 0004-to-be-fixed-up-use-a-decorator-to-indicate-views-whe.patch ajouté

- Fichier 0002-misc-refactor-ViewRestrictionMiddleware-24056.patch ajouté

- Fichier 0001-misc-add-a-required_on_login-flag-on-Attribute-24056.patch ajouté

J'ai gardé le décorateur parce que j'aime bien ça car c'est beaucoup plus visible que les xxx.flag = True après les vues mais j'ai grandement simplifié et là c'est directement visible dans les views.py.

#19 - 23 juillet 2021 15:52 - Paul Marillonnet

- Statut changé de Solution proposée à Solution validée

D'ac ça me va comme ça. Juste attendre que Jenkins voie vert pour pousser mais dans l'esprit de la solution proposée c'est ack pour moi.

#20 - 23 juillet 2021 17:15 - Benjamin Dauvergne

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

```
commit 5ff1ccaae83d7a095c9f479adf2615aea0af0320
Author: Benjamin Dauvergne <bdauvergne@entrouvert.com>
Date:   Fri Jul 16 10:57:24 2021 +0200

    misc: block user without required_on_login attributes (#24056)

    Superuser are exempted from the restriction.

commit c8e3f9376ba1e3315be1e2a7edb50ad66df26cf6
Author: Benjamin Dauvergne <bdauvergne@entrouvert.com>
Date:   Tue Jul 6 10:12:23 2021 +0200

    misc: refactor ViewRestrictionMiddleware (#24056)

    The generic code for checking restrictions is separated from the code
    specific to each specific check; here the code to check for
    PasswordReset models is extracted into
    check_password_reset_view_restriction().

commit afd5f689cb2438f133771cd7bbcbdd30b0ddd791
Author: Benjamin Dauvergne <bdauvergne@entrouvert.com>
Date:   Fri Jul 2 11:47:58 2021 +0200

    misc: add a required_on_login flag on Attribute (#24056)
```

#21 - 23 juillet 2021 18:16 - Frédéric Péters

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

#22 - 27 juillet 2021 15:01 - Mikaël Ates (de retour le 29 avril)

- Lié à Development #55865: Ajouter la case à cocher « Requis à la connexion » sur la page de configuration des attributs de profil. ajouté

Fichiers

0003-misc-block-user-without-required_on_login-attributes.patch	8,21 ko	16 juillet 2021	Benjamin Dauvergne
0002-misc-refactor-ViewRestrictionMiddleware-24056.patch	4,11 ko	16 juillet 2021	Benjamin Dauvergne
0001-misc-add-a-required_on_login-flag-on-Attribute-24056.patch	2,3 ko	16 juillet 2021	Benjamin Dauvergne
0003-misc-block-user-without-required_on_login-attributes.patch	11,2 ko	16 juillet 2021	Benjamin Dauvergne
0002-misc-refactor-ViewRestrictionMiddleware-24056.patch	4,11 ko	16 juillet 2021	Benjamin Dauvergne

0001-misc-add-a-required_on_login-flag-on-Attribute-24056.patch	2,3 ko	16 juillet 2021	Benjamin Dauvergne
0003-misc-block-user-without-required_on_login-attributes.patch	11,8 ko	23 juillet 2021	Benjamin Dauvergne
0004-to-be-fixed-up-use-a-decorator-to-indicate-views-whe.patch	9,77 ko	23 juillet 2021	Benjamin Dauvergne
0002-misc-refactor-ViewRestrictionMiddleware-24056.patch	4,1 ko	23 juillet 2021	Benjamin Dauvergne
0001-misc-add-a-required_on_login-flag-on-Attribute-24056.patch	2,3 ko	23 juillet 2021	Benjamin Dauvergne
0003-misc-block-user-without-required_on_login-attributes.patch	11,8 ko	23 juillet 2021	Benjamin Dauvergne
0004-to-be-fixed-up-use-a-decorator-to-indicate-views-whe.patch	10,2 ko	23 juillet 2021	Benjamin Dauvergne
0002-misc-refactor-ViewRestrictionMiddleware-24056.patch	4,1 ko	23 juillet 2021	Benjamin Dauvergne
0001-misc-add-a-required_on_login-flag-on-Attribute-24056.patch	2,3 ko	23 juillet 2021	Benjamin Dauvergne