Authentic 2 - Development #32007

Niveau d'authentification et TOTP

04 avril 2019 17:06 - Valentin Deniaud

Statut:	Fermé	**Début:**	04 avril 2019
Priorité:	Normal	**Echéance:**	
Assigné à:	Valentin Deniaud	**% réalisé:**	0%
Catégorie:		**Temps estimé:**	0:00 heure
Version cible:			
Patch proposed:	Oui	**Planning:**	Non

Description

Ticket pour mettre mes patches et avoir des avis, parce que des trucs commencent à marcher.
Pour l'instant c'est un peu fourre tout parce que les niveaux d'auth et TOTP devraient être dans des tickets séparés, mais les rassembler permet de tester plus facilement et de mieux se rendre compte du design.

Tout est interne à authentic, la partie ou les autres briques respectent les niveaux viendra plus tard.

Le fonctionnement d'authentification multifacteur visé :

- Potentiellement plein de facteurs d'authentification, à hiérarchiser via des niveaux d'authentification.
  Par ex, login/pass niveau 1, code sur téléphone niveau 2, scanneur d'iris niveau 42...
- Dans authentic, on configure les endroits de publik qu'on estime critiques, et on y associe des niveaux d'authentification via les rôles.
  Par ex un agent peut traiter des demandes mais aussi ajouter des utilisateurs, on mettra le rôle/la permission ajouter des utilisateurs niveau 2, et il pourra traiter des demandes normalement, mais devra s'identifier à l'aide d'un deuxième facteur pour gérer les utilisateurs.

Ça fonctionne avec les patches de ce ticket :
- créer un nouvel utilisateur, l'associer à un rôle de traitement de certaines demandes (ou d'administrateur d'une brique autre qu'authentic, n'importe quoi qui laisse faire quelque chose dans le backoffice)
- créer un deuxième rôle en mettant un niveau d'authentification de 2 dans le formulaire de création
- ajouter à ce rôle la permission pour créer des utilisateurs, et l'associer à l'utilisateur
- se connecter avec l'utilisateur, aller dans le backoffice (par ex authentic.dev.public.love/manage)
- le menu gadjo s'affiche bien, on peut aller sur wcs sans pb
- cliquer sur le bouton "utilisateurs" du backoffice: hop, on tombe sur le deuxième facteur

Bon et là arrive un premier problème, l'utilisateur doit se dire, pourquoi le monde est-il si cruel, je ne peux pas passer cette étape.
Je traiterai ça plus tard parce que la réponse n'est pas tranchée (l'obliger à configurer le deuxième facteur la première fois? lui laisser la responsabilité de le faire et le laisser passer tant que ce n'est pas fait?), en tout cas pour l'instant c'est bien inutile parce qu'il peut aller récupérer le QR code à tout moment sur son profil authentic, bonjour la sécurité.

Demandes liées:

Lié à Authentic 2 - Development #32786: Authentification multi-facteurs	**Fermé**	**03 mai 2019**

Historique

#1 - 04 avril 2019 17:07 - Valentin Deniaud

- Fichier 0009-auth2_multifactor-add-OATH-authentication-factor.patch ajouté

- Fichier 0002-manager-display-and-configure-permission-and-role-le.patch ajouté

- Fichier 0010-wip-enable-TOTP-auth-factor.patch ajouté

- Patch proposed changé de Non à Oui

- Fichier 0011-django_rbac-add-max-authentication-level.patch ajouté

- Fichier 0003-django_rbac-remove-useless-elif.patch ajouté

- Fichier 0006-utils-allow-specifying-auth-level-when-getting-backe.patch ajouté

- Fichier 0005-decorators-add-check_auth_level-for-automatic-increa.patch ajouté

- Fichier 0004-django_rbac-check-authentication-level-along-with-pe.patch ajouté

- Fichier 0007-views-handle-authentication-level-when-logging-in.patch ajouté

- Fichier 0008-manager-handle-authentication-level-transition.patch ajouté

- Fichier 0013-manager-display-menu-button-regardless-of-authentica.patch ajouté

- Fichier 0001-django_rbac-add-auth-level-to-Role-and-Permission-cl.patch ajouté

- Fichier 0012-decorators-add-a-way-to-bypass-auth-level-check.patch ajouté

#2 - 04 avril 2019 17:20 - Benjamin Dauvergne

Push your branch.

#3 - 04 avril 2019 17:46 - Benjamin Dauvergne

Tu ne peux pas vérifier le niveau d'authentification par rapport à l'utilisateur, c'est une donnée transitoire ça doit venir de la session, après l'utilisateur peut avoir un niveau d'authentification minimum mais c'est uniquement pour bloquer l'authentification.

#4 - 04 avril 2019 17:52 - Valentin Deniaud

Benjamin Dauvergne a écrit :

> Tu ne peux pas vérifier le niveau d'authentification par rapport à l'utilisateur, c'est une donnée transitoire ça doit venir de la session, après l'utilisateur peut avoir un niveau d'authentification minimum mais c'est uniquement pour bloquer l'authentification.

J'avais fait l'erreur au début, mais normalement ce n'est plus le cas, après c'est pas exclu qu'il reste un morceau de code égaré qui le laisse penser. (si le pb c'était que c'est transitoire encore il y aurait des moyens de contournement, mais le vrai pb c'est qu'un utilisateur peut avoir plusieurs sessions, et là c'est vraiment foutu)
Tu dois pas en être encore à ce commit mais ce qu'il se passe c'est que j'annote request.user avec le niveau d'auth qui est dans la session via un décorateur, donc l'attribut user.auth_level dans django_rbac vient bien de la session.

#5 - 10 avril 2019 15:03 - Valentin Deniaud

- Description mis à jour

J'ai changé/amélioré pas mal de trucs, et maintenant les tests passent. On m'a soufflé que ça ne servait pas à grand chose de mettre autant de patches dans un ticket, du coup je n'upload pas les nouveaux que j'ai (force) pushé. J'ai mis à jour les instructions pour tester dans la description du ticket et ajouté des infos sur comment ça marche.

Ce qu'il reste à faire :

- tests, traductions, logging, pep8
- améliorations de l'interface
- réfléchir au mécanisme qui permettra d'activer l'authentification forte au niveau utilisateur comme dit dans le ticket
- faire en sorte qu'on puisse associer facilement via l'interface d'admin des backends à un niveau d'authentification (là le niveau d'un backend est codé en dur, c'est nul)

À part ça dans ses grandes lignes la logique mise en place ne devrait plus bouger, donc si quelqu'un veut jeter un œil ça ne sera à priori pas une perte de temps.
D'ailleurs pour l'instant j'arrête de travailler sur cette partie et je m'attaque au côté communication entre les briques, parce que je préfère être sûr que les deux s'emboîtent bien avant de fignoler ce ticket.

#6 - 10 avril 2019 15:04 - Valentin Deniaud

- Description mis à jour

#7 - 16 avril 2019 13:48 - Valentin Deniaud

- Tracker changé de Support à Development

#8 - 03 mai 2019 15:58 - Valentin Deniaud

Mise à jour avec un peu de nettoyage et des gros bouts qu'il manquait.

Valentin Deniaud a écrit :

> Ce qu'il reste à faire :
>
> - réfléchir au mécanisme qui permettra d'activer l'authentification forte au niveau utilisateur comme dit dans le ticket

Fait. La manière dont c'est implémenté : un utilisateur est libre d'aller activer un facteur d'auth supplémentaire à tout moment dans son profil a2. Si il tombe sur une page qui déclenche une montée d'auth et qu'il a configuré un facteur qui lui permet de l'effectuer, c'est bon. Sinon, il est redirigé vers son profil, obligé de configurer un nouveau facteur puis de s'authentifier avec valider l'activation. Puis redirection vers la page initiale une fois fini.

> - faire en sorte qu'on puisse associer facilement via l'interface d'admin des backends à un niveau d'authentification (là le niveau d'un backend est codé en dur, c'est nul)

J'ai essayé et c'est pas super joli (tout tient dans a3b17be0). Peut-être que vouloir faire ça sert pas à grand chose au final, et qu'on peut se contenter de faire cette association de manière plus statique dans un fichier de conf ?

- tests, traductions, logging, pep8
- améliorations de l'interface

Avant de m'attaquer à ces deux derniers points, dont l'achèvement doit conduire à commencer à merger des trucs, il faudrait que l'on me dise si les nouveaux mécanismes mis en place dans ce ticket conviennent (que je n'écrive pas des tests pour du code qui sera jeté au final, par ex). Notamment le moment où on effectue la vérification du niveau d'authentification d'un utilisateur bien profond dans django_rbac (93f12678, accompagné de 94766b5c).

Un autre truc qu'il reste à faire, auquel je m'attaque en attendant, c'est de gérer la multitude de cas où il faut peut-être afficher un bouton, peut-être pas, idéalement sans ajouter 400 if/else.

#9 - 03 mai 2019 16:19 - Frédéric Péters

> il faudrait que l'on me dise si les nouveaux mécanismes mis en place dans ce ticket conviennent.

Sur ce ticket précis je n'ai pas regardé.

Mais sur l'ensemble, ceci est **mon** opinion, les intrusions dans le code des autres applications m'ennuient. Ça n'empêche pas qu'il y a utilité à avoir dans authentic la gestion de multi-facteurs, mais à mon sens plutôt comme ça peut exister ailleurs, pour offrir à l'usager la possibilité de sécuriser ses accès.

C'est-à-dire, par rapport à ce que tu écris, "Si il tombe sur une page qui déclenche une montée d'auth", que cette partie resterait de côté pour le moment, que "aller activer un facteur d'auth supplémentaire à tout moment dans son profil a2" serait + "et demander à ce que facteur soit demandé lors des connexions à son compte".

Bref, pour redire la même chose encore une fois, il y a une partie multi-facteurs qui serait super et pourrait être utile dans authentic, mais la partie dans les autres applications est une trop grosse fiction aujourd'hui pour aller y faire des changements.

#10 - 03 mai 2019 18:29 - Valentin Deniaud

- Lié à Development #32786: Authentification multi-facteurs ajouté

#11 - 03 mai 2019 18:33 - Valentin Deniaud

Répondu dans #32786 (btw le présent ticket ne concerne que le fonctionnement interne à a2).

#12 - 28 mai 2019 17:25 - Valentin Deniaud

- Statut changé de Nouveau à Fermé

Les patchs dans ce ticket et ce qui y est dit datent, plutôt que de pousser une grosse màj je vais découper en petit bout et ouvrir d'autres tickets.

Le plan :

1. Ajout des niveaux d'authentification interne à a2 (les fondations : mécanique bas niveau + travail d'interface) (#33515)
2. Utilisation des niveaux d'authentification (permettre la montée de niveau au moment du login, ...)
3. Ajout du facteur TOTP
4. Interaction avec les SP

Fichiers

0011-django_rbac-add-max-authentication-level.patch	3,91 ko	04 avril 2019	Valentin Deniaud
0003-django_rbac-remove-useless-elif.patch	888 octets	04 avril 2019	Valentin Deniaud
0006-utils-allow-specifying-auth-level-when-getting-backe.patch	1,66 ko	04 avril 2019	Valentin Deniaud
0005-decorators-add-check_auth_level-for-automatic-increa.patch	1,6 ko	04 avril 2019	Valentin Deniaud
0004-django_rbac-check-authentication-level-along-with-pe.patch	12,2 ko	04 avril 2019	Valentin Deniaud
0007-views-handle-authentication-level-when-logging-in.patch	2,04 ko	04 avril 2019	Valentin Deniaud
0008-manager-handle-authentication-level-transition.patch	1,8 ko	04 avril 2019	Valentin Deniaud
0013-manager-display-menu-button-regardless-of-authentica.patch	1,68 ko	04 avril 2019	Valentin Deniaud
0001-django_rbac-add-auth-level-to-Role-and-Permission-cl.patch	3,15 ko	04 avril 2019	Valentin Deniaud
0012-decorators-add-a-way-to-bypass-auth-level-check.patch	1,29 ko	04 avril 2019	Valentin Deniaud
0009-auth2_multifactor-add-OATH-authentication-factor.patch	12 ko	04 avril 2019	Valentin Deniaud

0002-manager-display-and-configure-permission-and-role-le.patch	3,01 ko	04 avril 2019	Valentin Deniaud
0010-wip-enable-TOTP-auth-factor.patch	1,22 ko	04 avril 2019	Valentin Deniaud