

## Authentic 2 - Development #67451

### authentificateur saml : remplacer le champ texte chemin vers les métadonnées par un champ fichier

18 juillet 2022 18:18 - Valentin Deniaud

<b>Statut:</b>	Rejeté	<b>Début:</b>	18 juillet 2022
<b>Priorité:</b>	Normal	<b>Echéance:</b>	
<b>Assigné à:</b>	Valentin Deniaud	<b>% réalisé:</b>	0%
<b>Catégorie:</b>		<b>Temps estimé:</b>	0:00 heure
<b>Version cible:</b>		<b>Planning:</b>	Non
<b>Patch proposed:</b>	Oui		
<b>Description</b>			
Ce serait nettement plus propre, à voir si on arrive à écrire une migration ou si il faudra toujours se traîner le champ texte en plus.			
<b>Demandes liées:</b>			
Lié à Authentic 2 - Development #69269: auth_saml: pré-remplir l'URL des méta...			<b>Nouveau</b> <b>19 septembre 2022</b>

#### Historique

##### #1 - 18 juillet 2022 19:20 - Frédéric Péters

Même sans migration on peut juste ne plus afficher le champ texte, je pense.

##### #2 - 17 août 2022 17:47 - Valentin Deniaud

- Assigné à mis à Valentin Deniaud

##### #3 - 24 août 2022 15:47 - Valentin Deniaud

- Fichier 0001-auth\_saml-prefer-metadata-file-upload-over-metadata-.patch ajouté

- Statut changé de Nouveau à Solution proposée

- Patch proposed changé de Non à Oui

Sans migration donc, en masquant le champ.

##### #4 - 26 août 2022 08:25 - Thomas Noël

Dans le clean, on pourrait râler quand il reste les deux possibilités :

```
if self.metadata_path and self.metadata_file:
    raise ValidationError('faut choisir dans la vie (merci d'effacer metadata_path)')
```

Aussi, peut-être que "upload\_to" pourrait être du genre "saml\_authenticator", voire une fonction qui nomme bien l'affaire :

```
def metadata_directory_path(instance, filename):
    return 'saml_authenticator_{0}/metadata.xml'.format(instance.id)
```

(mais ça peut peut-être faire un bogue lors de la toute première création de l'authenticator, peut-être qu'il n'a pas encore de id, je sais pas super trop)

##### #5 - 13 septembre 2022 10:23 - Valentin Deniaud

- Fichier avec\_renommage.png ajouté

- Fichier sans\_renommage.png ajouté

Thomas Noël a écrit :

Dans le clean, on pourrait râler quand il reste les deux possibilités :

Ça rejoint l'interdiction de fournir une URL et un fichier en même temps, c'est [#66986](#), je pense qu'on peut gérer tout là bas.

Aussi, peut-être que "upload\_to" pourrait être du genre "saml\_authenticator", voire une fonction qui nomme bien l'affaire :

J'hésite, je trouve le rendu du champ fichier de Django assez moche parce qu'il affiche le chemin, cf screenshot. En plus si on upload de nouvelles métadonnées on se retrouvera à afficher « saml\_authenticator\_42/metadata\_PuFdba.xml », parce que Django ajoute automatiquement un suffixe en cas de conflit de nommage. Réutiliser le nom du fichier envoyé ça laisse à l'admin la responsabilité de bien nommer son fichier, c'est peut-être le mieux qu'on puisse faire ?

#### #6 - 13 septembre 2022 10:43 - Benjamin Dauvergne

Ici comme ailleurs j'aurai une préférence pour avoir tout dans le modèle (même si en front c'est un champs fichier), donc un TextField (c'est fait comme ça sur LibertyProvider coté IdP et je ne le regrette pas), on se fout un peu du nom du fichier des métadonnées ensuite les media d'authentic ne doivent pas être exposés au public donc faudrait en plus une vue de téléchargement de toute façon.

Par contre ce serait bien de considérer l'entity\_id issue du fichier comme l'URL et de l'ajouter aussi si on a un fort pressentiment que c'est une URL de métadonnées (genre on fait un GET et ça renvoie des métadonnées, avec un timeout à 5 secondes). Au niveau django-mellon on peut déclarer les deux et ça se mettra à jour, tout seul. Autant pour les ADFS on premise c'est rare que ça bouge mais pour AzureAD possible que ça fasse du roll-over de clé.

#### #7 - 13 septembre 2022 11:58 - Valentin Deniaud

Benjamin Dauvergne a écrit :

Ici comme ailleurs j'aurai une préférence pour avoir tout dans le modèle

Pourquoi ?

(même si en front c'est un champs fichier), donc un TextField

Le TextField existe, il n'y a donc rien à faire, je ne vais pas bricoler un champ fichier alimenté par un TextField pour s'éviter le copier coller nécessaire à remplir le TextField directement.

Ici comme ailleurs il s'agit de simplifier à la fois l'interface de configuration et le code, je ne produirai rien qui n'aille pas dans ce sens.

les media d'authentic ne doivent pas être exposés au public donc faudrait en plus une vue de téléchargement de toute façon.

Bonne remarque mais il y a juste à avoir un uuid aléatoire dans le chemin vers le fichier, c'est classique.

Par contre ce serait bien de considérer l'entity\_id issue du fichier comme l'URL

Je te laisse ouvrir le ticket correspondant.

#### #8 - 13 septembre 2022 12:33 - Benjamin Dauvergne

Valentin Deniaud a écrit :

Pourquoi ?

C'est mieux d'avoir la configuration à un seul endroit, en base, qu'à plein d'endroits, ça enlève un accès disque inutile et ça ne coûte rien.

(même si en front c'est un champs fichier), donc un TextField

Le TextField existe, il n'y a donc rien à faire, je ne vais pas bricoler un champ fichier alimenté par un TextField pour s'éviter le copier coller nécessaire à remplir le TextField directement.

Ça ne me dérange pas.

Ici comme ailleurs il s'agit de simplifier à la fois l'interface de configuration et le code, je ne produirai rien qui n'aille pas dans ce sens.

Je parle du stockage, l'interface est très bien.

les media d'authentic ne doivent pas être exposés au public donc faudrait en plus une vue de téléchargement de toute façon.

Bonne remarque mais il y a juste à avoir un uuid aléatoire dans le chemin vers le fichier, c'est classique.

Pas exposé, c'est pas exposé, autant éviter un trou de sécurité plus tard quand on exposera une clé RSA stockée dans un fichier parce qu'on a pris une mauvaise habitude.

Par contre ce serait bien de considérer l'entity\_id issue du fichier comme l'URL

Je te laisse ouvrir le ticket correspondant.

Ok.

**#9 - 13 septembre 2022 12:34 - Benjamin Dauvergne**

- Statut changé de Solution proposée à En cours

**#10 - 19 septembre 2022 15:34 - Benjamin Dauvergne**

- Lié à Development #69269: auth\_saml: pré-remplir l'URL des métadonnées via l'entity\_id référencée dans un fichier de métadonnée donné explicitement ajouté

**#11 - 19 septembre 2022 16:51 - Valentin Deniaud**

- Fichier 0001-auth\_saml-prefer-metadata-file-upload-over-metadata-.patch ajouté

- Statut changé de En cours à Solution proposée

Benjamin Dauvergne a écrit :

Ici comme ailleurs j'aurai une préférence pour avoir tout dans le modèle (même si en front c'est un champs fichier), donc un TextField (c'est fait comme ça sur LibertyProvider coté IdP et je ne le regreete pas)

Joke's on you j'ai regardé le champ en question pour ma culture et il ne fonctionne pas, erreur systématique à l'upload du fichier.

Ça vient donc appuyer mon argument de ne rien bidouiller, revoici juste avec un chemin vers le fichier aléatoire.

**#12 - 13 octobre 2022 10:56 - Benjamin Dauvergne**

- Statut changé de Solution proposée à En cours

Valentin Deniaud a écrit :

Ça vient donc appuyer mon argument de ne rien bidouiller, revoici juste avec un chemin vers le fichier aléatoire.

Oui moi aussi ne rien bidouiller, met juste un TextField, je n'ai pas dit de reprendre le widget de l'admin pour permettre l'upload.

**#13 - 19 octobre 2022 12:03 - Valentin Deniaud**

Je ne comprends plus ce qu'il faut faire, il y a déjà un TextField (SAMLAuthenticator.metadata).

Ce qui justifiait ce ticket c'est que le procédé « avoir metadata.xml, l'ouvrir, le copier, le coller dans le formulaire » paraissait moins bien que « avoir metadata.xml, l'uploader ».

**#14 - 19 octobre 2022 12:57 - Benjamin Dauvergne**

Dans ce cas je propose de simplement supprimer le champ chemin qui est effectivement une source d'erreur potentielle, et juste recopier les contenu dans le champ contenu XML. Le souci c'est surtout le champ texte qui contient un chemin sur le disque local, pas l'absence d'un champ fichier.

PS:

Ce qui justifiait ce ticket c'est que le procédé « avoir metadata.xml, l'ouvrir, le copier, le coller dans le formulaire » paraissait moins bien que « avoir metadata.xml, l'uploader ».

Il me semble que le souci c'était que quelqu'un (Fred/) a remarqué qu'il y avait des chemins vers des fichiers locaux, il a pu penser que c'était la seule façon de faire.

**#15 - 19 octobre 2022 17:02 - Valentin Deniaud**

- Statut changé de En cours à Rejeté

OK allons-y pour virer le champs chemin et migrer les données dans le textfield, je vais ouvrir un autre ticket.

## Fichiers

---

0001-auth_saml-prefer-metadata-file-upload-over-metadata-.patch	7,38 ko	24 août 2022	Valentin Deniaud
---	---------	--------------	------------------

avec_renommage.png	11,4 ko 13 septembre 2022	Valentin Deniaud
sans_renommage.png	10,2 ko 13 septembre 2022	Valentin Deniaud
0001-auth_saml-prefer-metadata-file-upload-over-metadata-.patch	9,17 ko 19 septembre 2022	Valentin Deniaud