

Authentic 2 - Bug #42646

saml, unhashable type: 'set' au SSO

07 mai 2020 22:27 - Frédéric Péters

Statut:	Fermé	Début:	07 mai 2020
Priorité:	Normal	Echéance:	
Assigné à:		% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposed:	Oui		
Description			
Traceback:			
File "/usr/lib/python2.7/dist-packages/django/core/handlers/exception.py" in inner 41. response = get_response(request)			
File "/usr/lib/python2.7/dist-packages/django/core/handlers/base.py" in _get_response 187. response = self.process_exception_by_middleware(e, request)			
File "/usr/lib/python2.7/dist-packages/django/core/handlers/base.py" in _get_response 185. response = wrapped_callback(request, *callback_args, **callback_kwargs)			
File "/usr/lib/python2.7/dist-packages/authentic2/decorators.py" in f 47. return func(request, *args, **kwargs)			
File "/usr/lib/python2.7/dist-packages/authentic2/decorators.py" in f 47. return func(request, *args, **kwargs)			
File "/usr/lib/python2.7/dist-packages/django/views/decorators/cache.py" in _wrapped_view_func 57. response = view_func(request, *args, **kwargs)			
File "/usr/lib/python2.7/dist-packages/authentic2/idp/saml/saml2_endpoints.py" in continue_sso 680. nid_format=nid_format)			
File "/usr/lib/python2.7/dist-packages/authentic2/idp/saml/saml2_endpoints.py" in sso_after_process_request 900. name_id = build_assertion(request, login, provider, nid_format=nid_format)			
File "/usr/lib/python2.7/dist-packages/authentic2/idp/saml/saml2_endpoints.py" in build_assertion 471. add_attributes(request, entity_id, assertion, provider, nid_format)			
File "/usr/lib/python2.7/dist-packages/authentic2/idp/saml/saml2_endpoints.py" in add_attributes 311. if (name, name_format, value) in seen:			
Exception Type: TypeError at /idp/saml2/continue Exception Value: unhashable type: 'set'			

Révisions associées

Révision 1c7ce145 - 07 mai 2020 23:33 - Frédéric Péters

saml: deal with set attributes (#42646)

Historique

#1 - 07 mai 2020 23:06 - Frédéric Péters

Ça se joue autour de 8b44f67d448950a5c20751ecbbf9b95e596a5db0 qui contient, simplifié,

```
- tuples = [tuple(t) for definition in qs for t in definition.to_tuples(ctx)]  
+ tuples = [tuple(t) for definition in definitions for t in definition.to_tuples(attributes_ctx)]
```

Mais definition ça sort d'une boucle au-dessus, ce qui est bizarre (mais était déjà là avant).

#2 - 07 mai 2020 23:32 - Frédéric Péters

- Fichier 0001-saml-deal-with-set-attributes-42646.patch ajouté
- Statut changé de Nouveau à Solution proposée
- Patch proposed changé de Non à Oui

Et non c'est un autre de la série le coupable, 5e87295adc8f4b256b18817a831361feecb79e11.

```
-     for text_value in normalize_attribute_values(values):
-         yield (name, name_format, friendly_name, text_value)
+     if not isinstance(values, list):
+         values = [values]
+     for value in values:
+         yield (name, name_format, friendly_name, value)
```

Et là un attribut de type set, il se trouve pris dans le not isinstance, alors que j'imagine l'intention est plutôt d'itérer dessus.

Patch attaché, testé live sur la recette.

#3 - 07 mai 2020 23:41 - Frédéric Péters

Et poussé et déployé mais je laisse ouvert, pour commentaire.

#4 - 08 mai 2020 11:00 - Frédéric Péters

- Statut changé de Solution proposée à Solution déployée

Auto validé déployé et j'ai créé [#42657](#) pour que des tests soient ajoutés.

Fichiers

0001-saml-deal-with-set-attributes-42646.patch	923 octets	07 mai 2020	Frédéric Péters
--	------------	-------------	-----------------