django-mellon - Development #69740

Corriger la gestion de next_url sur un SLO initié sur le SP

29 septembre 2022 22:02 - Benjamin Dauvergne

Statut:	Fermé	**Début:**	29 septembre 2022
Priorité:	Normal	**Echéance:**	
Assigné à:	Benjamin Dauvergne	**% réalisé:**	0%
Catégorie:		**Temps estimé:**	0:00 heure
Version cible:			
Patch proposed:	Oui	**Planning:**	Non

Description

Si le paramètre next n'est pas passé, next_url se retrouve à None, et si la vue est utilisé directement (par authentic par exemple) il faudrait pouvoir forcer la valeur de next_url.

Révisions associées

Révision 43ce1d81 - 05 octobre 2022 12:29 - Benjamin Dauvergne

utils: use same_origin() from authentic2 (#69740)

Révision 86d3cad3 - 05 octobre 2022 17:23 - Benjamin Dauvergne

views: improve handling of next_url for sp initiated logout (#69740)

Révision e98308d4 - 05 octobre 2022 17:23 - Benjamin Dauvergne

views: allow overriding the default return url after logout (#69740)

Révision 600c8cfb - 05 octobre 2022 17:23 - Benjamin Dauvergne

misc: keep nameid attributes to rebuild it (#69740)

Logout requests need a properly built NameID element, but we did not store enough information in models to do that, we uses the LassoSession dump from the session as a work-around. In order to have a session-less logout endpoint, we need to store those informations in the UserSAMLIdentifier model.

Révision 7f9602c5 - 05 octobre 2022 17:23 - Benjamin Dauvergne

utils: add method to build a session dump from models (#69740)

Storing the LassoSession dump in the Django session is no longer needed, we can rebuild it from the information in the models.

Révision 218afde9 - 05 octobre 2022 17:23 - Benjamin Dauvergne

misc: make logout work with transient NameID (#69740)

Implementation of transient NameID is special, the transient NameID is ignored and an attribut value is used as the federation key. But in order to producre a proper NameID for the logout request we need the transient NameID value. To work around this problem we add a transient_name_id attribute to the SessionIndex model representing the current SSO session, and we modify the session dump template to use this value instead of UserSAMLIdentifier.name_id if transient_name_id is not None.

Révision f335a403 - 05 octobre 2022 17:23 - Benjamin Dauvergne

views: implement a sessionless logout endpoint (#69740)

To implement SAML single logout in authentic we need a logout endpoint which works event after the user session has been killed, to do that we store the needed information in Django signed token, and use it to initiate the logout request. Afterward the next_url is stored in short-lived session cookie instead of the session.

Révision 45f81514 - 05 octobre 2022 19:53 - Benjamin Dauvergne

misc: clean SessionIndex during logout (#69740)

SessionIndex are deleted when the linked session does not exist anymore
and 5 minutes after the creation of the logout request.

Historique

#1 - 30 septembre 2022 01:09 - Benjamin Dauvergne

- Fichier 0001-utils-use-same_origin-from-authentic2-69740.patch ajouté

- Fichier 0002-views-improve-handling-of-next_url-for-sp-initiated-.patch ajouté

- Statut changé de Nouveau à Solution proposée

- Patch proposed changé de Non à Oui

Le premier patch c'est pour adopter la version de same_origin qui est dans authentic, à une modification près, le fait d'accepter directement les URLs relatives comme étant de même origine (dans authentic c'est fait dans good_next_url qui utilise same_origin).

Le deuxième fait en sorte de toujours obtenir un next_url, au pire '/', de ne plus utiliser le referer comme next_url dans aucun des cas (mauvaise idée) et de permettre de transmettre un next_url forcé comme paramètre de la vue (ce dernier point était le vrai objectif du ticket, pour servir dans #69720.

#2 - 30 septembre 2022 08:05 - Benjamin Dauvergne

- Fichier 0001-utils-use-same_origin-from-authentic2-69740.patch ajouté

- Fichier 0002-views-improve-handling-of-next_url-for-sp-initiated-.patch ajouté

Forcer le next_url au départ de la requête ne suffira pas, il faut juste changer la valeur de retour par défaut pour que ce soit /logout/ sur authentic dans tous les cas. Le souci étant qu'authentic nettoye la session après l'action de mellon.views.LogoutView et donc l'état posé en session est perdu.

#3 - 30 septembre 2022 08:18 - Benjamin Dauvergne

- Fichier 0001-utils-use-same_origin-from-authentic2-69740.patch ajouté

- Fichier 0002-views-improve-handling-of-next_url-for-sp-initiated-.patch ajouté

Et le paramètre doit s'appeler logout_next_url pour éviter de marcher sur un éventuellement retour par défaut au niveau de la vue de login dans urls.py. À utiliser ainsi:

```
url(r'^accounts/saml/', include('mellon.urls'), kwargs={'logout_next_url': '/logout/'})
```

#4 - 03 octobre 2022 10:41 - Benjamin Dauvergne

- Statut changé de Solution proposée à En cours

Je vais revoir complètement la vue pour ne plus passer par la session pour conserver l'URL de retour, ça n'est pas pratique sur un logout (qui supprime la session).

#5 - 04 octobre 2022 11:47 - Benjamin Dauvergne

- Fichier 0001-utils-use-same_origin-from-authentic2-69740.patch ajouté

- Fichier 0004-misc-keep-nameid-attributes-to-rebuild-it-69740.patch ajouté

- Fichier 0007-views-implement-a-sessionless-logout-endpoint-69740.patch ajouté

- Fichier 0008-misc-clean-SessionIndex-during-logout-69740.patch ajouté

- Fichier 0002-views-improve-handling-of-next_url-for-sp-initiated-.patch ajouté

- Fichier 0005-utils-add-method-to-build-a-session-dump-from-models.patch ajouté

- Fichier 0006-misc-make-logout-work-with-transient-NameID-69740.patch ajouté

- Fichier 0003-views-allow-overriding-the-default-return-url-after-.patch ajouté

- Statut changé de En cours à Solution proposée

L'objectif de tout ça c'était d'arriver à un endpoint de logout SAML Redirect qui fonctionne que la session existe encore ou pas.

- 0001: j'ai remarqué que la fonction same_origin était un peu foireuse, j'ai repris celle d'authentic qui est mieux
- 0002: le next_url est géré plus uniformément, c'est pareil dans tous les endpoints
- 0003: pouvoir surcharger le next_url par défaut des vues de déconnexions au niveau de l'inclusion de urls.py ; ça permettra à authentic de définir que l'URL de retour par défaut pour un logout SAML est son endpoint de logout local (/logout/) et pas (/) comme sur un service

- 0004: pour fonctionner sans la session et le dump de LassoSession (c'est du contexte qu'il faut conserver pour que Lasso fonctionne), il faut rajouter des champs à UserSAMLIdentifier pour reconstruire le dump
- 0005: suite de 0004 et utilisation dans les vues
- 0006: suite aux deux derniers le logout dans le cas d'un NameID transient (utilisé à l'université de Paris Nanterre) ne fonctionnait plus, parce qu'on ne stoque pas vraiment le NameID dans UserSAMLIdentifier.name_id mais l'email dans ce cas, pour pallier à ça je conserver le vrai NameID dans SessionIndex et j'adapte le code
- 0007: le but du ticket, avoir une méthode pour générer une URL de logout qui fonctionnera même si la session locale a disparue
- 0008: au passage je me suis aperçu que les SessionIndex n'était pas vraiment bien nettoyé sauf sur authentic[1], donc plutôt qu'en tâche de fond je les nettoye un peu à chaque logout (5 minutes après le démarrage du processus de logout et si la session n'existe plus, au passage j'ai ajouté un cas rapide quand on a des sessions en DB, le cas normal en fait)

[1] Sur le guichet principal d'une grande de ville france bien connue où on mange dans des bouchons:

```
combo=# select count(*) from mellon_sessionindex;
 count
--------
 180632
(1 ligne)
```

#6 - 05 octobre 2022 12:43 - Benjamin Dauvergne

- Fichier 0001-utils-use-same_origin-from-authentic2-69740.patch ajouté

- Fichier 0004-misc-keep-nameid-attributes-to-rebuild-it-69740.patch ajouté

- Fichier 0007-views-implement-a-sessionless-logout-endpoint-69740.patch ajouté

- Fichier 0008-misc-clean-SessionIndex-during-logout-69740.patch ajouté

- Fichier 0002-views-improve-handling-of-next_url-for-sp-initiated-.patch ajouté

- Fichier 0005-utils-add-method-to-build-a-session-dump-from-models.patch ajouté

- Fichier 0006-misc-make-logout-work-with-transient-NameID-69740.patch ajouté

- Fichier 0003-views-allow-overriding-the-default-return-url-after-.patch ajouté

Avec un test en plus dans 0008 pour vérifier que je ne supprime jamais les SesssionIndex trop tôt.

#8 - 05 octobre 2022 16:58 - Paul Marillonnet

Une première relecture :

· 0001 : Pas compris pourquoi on devait inverser les request.build_absolute_uri() avec l'autre argument dans les appels à utils.same_origin dans mellon.views.
J'en comprends que le second argument peut-être un motif, mais pourquoi d'un coup on a besoin d'inverser les deux ?

· 0002 : OK.

· 0003 : Question de forme encore sans doute qu'on peut faire en sorte d'avoir le bon nom d'argument par mot-clé pour sp_logout_response du premier coup (sans avoir 0002 qui ajoute un next_url=None puis 0003 qui s/next_url/logout_&/).

· 0004 : Ok.

· 0005 : dans l'idée ok mais j'ai un peu peur sur

```
                                session_indexes = models.SessionIndex.objects.filter(
                                    saml_identifier__user=request.user, saml_identifier__issuer__entity_id=issuer
                                ).order_by('-id')[:1]
```

pas sûr à 100% que ça va marcher de prendre le SessionIndex qui a l'id Django le plus grand, pas sûr qu'on retrouve la bonne session ici :/

· 0006 : Ok, je ne connaissais pas ce comportement en SAML qui tolère le SLO initié SP à partir d'un name id transient. Par curiosité, côté a2 idp saml on a ce qu'il faut pour gérer ça, i.e. est-ce que si un SP demande un SLO avec un attribut transient on sait faire ça ?

· 0007 : Ok. Au prochain audit de BZHunt on nous retoquera encore qu'on balance dans les URLs des jetons signés qui contiennent des données au lieu de maintenir en local une correspondance entre clé opaque signée dans la querystring d'une part, et d'autre part valeur qui elle n'y apparaît pas. Perso ça va je ne suis pas choqué que quelqu'un qui a dérobé la SECRET_KEY puisse déterminer l'url de redirection en fin de déconnexion ou l'id local de l'index de session SAML, il aura d'autres choses à faire plus intéressantes ailleurs avec cette clé secrète :)

· 0008 :
-> Petit anachronisme on utilise dans 0007 le SessionIndex.logout_timestamp déclaré ici. Pas grave.
-> Je me dis que ce serait quand même intéressant de conserver une méthode de classe cleanup() notamment pour la commande cleanupauthentic pour les déploiement où a2 est SP SAML, quitte à factoriser du code de clean_session_indexes_after_logout ?
-> J'apprends que le Model.objects.filter(…)[:N] est bien intégré et génère bien un SELECT … LIMIT N; en SQL. Cool, on en apprend tous les jours :)

#9 - 05 octobre 2022 19:53 - Benjamin Dauvergne

- Statut changé de Solution proposée à En cours

Paul Marillonnet a écrit :

> Une première relecture :
>
> · 0001 : Pas compris pourquoi on devait inverser les request.build_absolute_uri() avec l'autre argument dans les appels à utils.same_origin dans mellon.views.
> J'en comprends que le second argument peut-être un motif, mais pourquoi d'un coup on a besoin d'inverser les deux ?

L'ancienne API était symétrique, mais en vrai on veut une API ou la première URL est absolue (la notion d'origine est un spéciale, c'est pas vraiment une URL, c'est schéma://nomhote(:port) (https://developer.mozilla.org/fr/docs/web/http/headers/origin) et la deuxième n'importe quoi (relative, absolue, whatever) et qui répond oui non. L'ancienne API ne gérait pas bien cela, la nouvelle prend l'URL absolue en premier et l'autre en deuxième.

> · 0002 : OK.
>
> · 0003 : Question de forme encore sans doute qu'on peut faire en sorte d'avoir le bon nom d'argument par mot-clé pour sp_logout_response du premier coup (sans avoir 0002 qui ajoute un next_url=None puis 0003 qui s/next_url/logout_&/).

D'ac, j'ai viré l'ajout du paramètre dans 0002 pour tout mettre dans 0003, c'était des restes de tâtonnements.

> · 0004 : Ok.
>
> · 0005 : dans l'idée ok mais j'ai un peu peur sur
> [...]
> pas sûr à 100% que ça va marcher de prendre le SessionIndex qui a l'id Django le plus grand, pas sûr qu'on retrouve la bonne session ici :/

Le seul moyen de garantir des id croissants ce serait de locker (via .select_for_update()) UserSAMLIdentifier pendant la création d'un SessionIndex qui le référence, mais dans l'absolu c'est rare d'avoir plusieurs SessionIndex, et encore plus créés dans un temps très court. Faute de mieux je reste sur cette heuristique.

> · 0006 : Ok, je ne connaissais pas ce comportement en SAML qui tolère le SLO initié SP à partir d'un name id transient. Par curiosité, côté a2 idp saml on a ce qu'il faut pour gérer ça, i.e. est-ce que si un SP demande un SLO avec un attribut transient on sait faire ça ?

Je pense que oui, mais je n'ai pas testé récemment; ça n'est de toute façon pas pour authentic en IdP c'est pour Shibboleth, il n'y a que sur Renater&co qu'on voit des trucs pareils.

> · 0007 : Ok. Au prochain audit de BZHunt on nous retoquera encore qu'on balance dans les URLs des jetons signés qui contiennent des données au lieu de maintenir en local une correspondance entre clé opaque signée dans la querystring d'une part, et d'autre part valeur qui elle n'y apparaît pas. Perso ça va je ne suis pas choqué que quelqu'un qui a dérobé la SECRET_KEY puisse déterminer l'url de redirection en fin de déconnexion ou l'id local de l'index de session SAML, il aura d'autres choses à faire plus intéressantes ailleurs avec cette clé secrète :)

Je peux importer aes_encrypt_bidule qu'on utilise ailleurs, mais franchement le jeton est signé et avec un sel le rendant inutilisable pour tout autre usage, c'est tiré par les cheveux, et pour le détourner il faudrait encore qu'un autre endpoint attende les mêmes données JSON. On pourrait effectivement s'en servir sur une autre instance, je vais ajouter le host au sel, mais c'est capilotracté (et pour nous surtout du au fait que settings.SECRET_KEY ne varie pas entre les tenants sur un même SaaS).

> · 0008 :
> -> Petit anachronisme on utilise dans 0007 le SessionIndex.logout_timestamp déclaré ici. Pas grave.

J'ai cherché je n'ai pas trouvé de référence à logout_timestamp dans 0007.

> -> Je me dis que ce serait quand même intéressant de conserver une méthode de classe cleanup() notamment pour la commande cleanupauthentic pour les déploiement où a2 est SP SAML, quitte à factoriser du code de clean_session_indexes_after_logout ?

Je peux bêtement la renommer cleanup, ça fera le job.

#10 - 05 octobre 2022 19:55 - Benjamin Dauvergne

- Fichier 0001-utils-use-same_origin-from-authentic2-69740.patch ajouté

- Fichier 0004-misc-keep-nameid-attributes-to-rebuild-it-69740.patch ajouté

- Fichier 0007-views-implement-a-sessionless-logout-endpoint-69740.patch ajouté

- Fichier 0008-misc-clean-SessionIndex-during-logout-69740.patch ajouté

- Fichier 0002-views-improve-handling-of-next_url-for-sp-initiated-.patch ajouté

- Fichier 0005-utils-add-method-to-build-a-session-dump-from-models.patch ajouté

- Fichier 0006-misc-make-logout-work-with-transient-NameID-69740.patch ajouté

- Fichier 0003-views-allow-overriding-the-default-return-url-after-.patch ajouté

- Statut changé de En cours à Solution proposée

- déplacement de la modification de sp_logout_response de 0002 à 0003
- renommage de la méthode de classe de nettoyage en cleanup avec un paramètre optionnel

#11 - 06 octobre 2022 08:44 - Paul Marillonnet

Benjamin Dauvergne a écrit :

> L'ancienne API était symétrique, mais en vrai on veut une API ou la première URL est absolue (la notion d'origine est un spéciale, c'est pas vraiment une URL, c'est schéma://nomhote(:port) (https://developer.mozilla.org/fr/docs/web/http/headers/origin) et la deuxième n'importe quoi (relative, absolue, whatever) et qui répond oui non. L'ancienne API ne gérait pas bien cela, la nouvelle prend l'URL absolue en premier et l'autre en deuxième.

Ok.

> Le seul moyen de garantir des id croissants ce serait de locker (via .select_for_update()) UserSAMLIdentifier pendant la création d'un SessionIndex qui le référence, mais dans l'absolu c'est rare d'avoir plusieurs SessionIndex, et encore plus créés dans un temps très court. Faute de mieux je reste sur cette heuristique.

Ok.

> Je pense que oui, mais je n'ai pas testé récemment; ça n'est de toute façon pas pour authentic en IdP c'est pour Shibboleth, il n'y a que sur Renater&co qu'on voit des trucs pareils.

Oui je ne savais pas que ça existait ce genre de choses :)

> Je peux importer aes_encrypt_bidule qu'on utilise ailleurs, mais franchement le jeton est signé et avec un sel le rendant inutilisable pour tout autre usage, c'est tiré par les cheveux, et pour le détourner il faudrait encore qu'un autre endpoint attende les mêmes données JSON. On pourrait effectivement s'en servir sur une autre instance, je vais ajouter le host au sel, mais c'est capilotracté (et pour nous surtout du au fait que settings.SECRET_KEY ne varie pas entre les tenants sur un même SaaS).

Ok.

> J'ai cherché je n'ai pas trouvé de référence à logout_timestamp dans 0007.

Pardon, fatigue de fin de journée de mon côté hier, une hallucination de ma part.

> Je peux bêtement la renommer cleanup, ça fera le job.

Ok.

#12 - 06 octobre 2022 08:44 - Paul Marillonnet

- Statut changé de Solution proposée à Solution validée

Benjamin Dauvergne a écrit :

> - déplacement de la modification de sp_logout_response de 0002 à 0003
> - renommage de la méthode de classe de nettoyage en cleanup avec un paramètre optionnel

C'est bon pour moi.

#13 - 06 octobre 2022 08:59 - Benjamin Dauvergne

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

```
commit 45f81514bc56fe6be36bbe78c361affa52ac5625
Author: Benjamin Dauvergne <bdauvergne@entrouvert.com>
Date:   Tue Oct 4 11:33:07 2022 +0200
```

```
    misc: clean SessionIndex during logout (#69740)

    SessionIndex are deleted when the linked session does not exist anymore
    and 5 minutes after the creation of the logout request.

commit f335a403c1a0acf05ffc7209c7e8981f23a6f70c
Author: Benjamin Dauvergne <bdauvergne@entrouvert.com>
Date:   Tue Oct 4 10:44:59 2022 +0200

    views: implement a sessionless logout endpoint (#69740)

    To implement SAML single logout in authentic we need a logout endpoint
    which works event after the user session has been killed, to do that we
    store the needed information in Django signed token, and use it to
    initiate the logout request. Afterward the next_url is stored in
    short-lived session cookie instead of the session.

commit 218afde9cd7903f1d4bb0ce89143d45a48416fbd
Author: Benjamin Dauvergne <bdauvergne@entrouvert.com>
Date:   Mon Oct 3 17:23:51 2022 +0200

    misc: make logout work with transient NameID (#69740)

    Implementation of transient NameID is special, the transient NameID is
    ignored and an attribut value is used as the federation key. But in
    order to producre a proper NameID for the logout request we need the
    transient NameID value. To work around this problem we add a
    transient_name_id attribute to the SessionIndex model representing the
    current SSO session, and we modify the session dump template to use this
    value instead of UserSAMLIdentifier.name_id if transient_name_id is not
    None.

commit 7f9602c528cd8e761ae7c7af643f8657b3f85763
Author: Benjamin Dauvergne <bdauvergne@entrouvert.com>
Date:   Mon Oct 3 16:06:03 2022 +0200

    utils: add method to build a session dump from models (#69740)

    Storing the LassoSession dump in the Django session is no longer needed,
    we can rebuild it from the information in the models.

commit 600c8cfbc0a573ed1c25876abc82c42ee732f941
Author: Benjamin Dauvergne <bdauvergne@entrouvert.com>
Date:   Mon Oct 3 12:19:38 2022 +0200

    misc: keep nameid attributes to rebuild it (#69740)

    Logout requests need a properly built NameID element, but we did not
    store enough information in models to do that, we uses the LassoSession
    dump from the session as a work-around. In order to have a session-less
    logout endpoint, we need to store those informations in the
    UserSAMLIdentifier model.

commit e98308d45c04f1c1bea917825bd39c100b4ef749
Author: Benjamin Dauvergne <bdauvergne@entrouvert.com>
Date:   Mon Oct 3 11:17:37 2022 +0200

    views: allow overriding the default return url after logout (#69740)

commit 86d3cad3b8676039f454146900a3717366f30f41
Author: Benjamin Dauvergne <bdauvergne@entrouvert.com>
Date:   Fri Sep 30 00:46:05 2022 +0200

    views: improve handling of next_url for sp initiated logout (#69740)

commit 43ce1d814120da3ed96f2d8e616a650840fa5f21
Author: Benjamin Dauvergne <bdauvergne@entrouvert.com>
Date:   Fri Sep 30 00:05:15 2022 +0200

    utils: use same_origin() from authentic2 (#69740)
```

#14 - 10 octobre 2022 10:15 - Transition automatique

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

#15 - 11 décembre 2022 04:42 - Transition automatique

Automatic expiration

Fichiers

0001-utils-use-same_origin-from-authentic2-69740.patch	3,62 ko	29 septembre 2022	Benjamin Dauvergne
0002-views-improve-handling-of-next_url-for-sp-initiated-.patch	4,58 ko	29 septembre 2022	Benjamin Dauvergne
0001-utils-use-same_origin-from-authentic2-69740.patch	3,62 ko	30 septembre 2022	Benjamin Dauvergne
0002-views-improve-handling-of-next_url-for-sp-initiated-.patch	5,92 ko	30 septembre 2022	Benjamin Dauvergne
0001-utils-use-same_origin-from-authentic2-69740.patch	3,62 ko	30 septembre 2022	Benjamin Dauvergne
0002-views-improve-handling-of-next_url-for-sp-initiated-.patch	5,95 ko	30 septembre 2022	Benjamin Dauvergne
0001-utils-use-same_origin-from-authentic2-69740.patch	3,62 ko	04 octobre 2022	Benjamin Dauvergne
0004-misc-keep-nameid-attributes-to-rebuild-it-69740.patch	5,77 ko	04 octobre 2022	Benjamin Dauvergne
0007-views-implement-a-sessionless-logout-endpoint-69740.patch	7,14 ko	04 octobre 2022	Benjamin Dauvergne
0008-misc-clean-SessionIndex-during-logout-69740.patch	8,51 ko	04 octobre 2022	Benjamin Dauvergne
0002-views-improve-handling-of-next_url-for-sp-initiated-.patch	3,72 ko	04 octobre 2022	Benjamin Dauvergne
0005-utils-add-method-to-build-a-session-dump-from-models.patch	9,4 ko	04 octobre 2022	Benjamin Dauvergne
0006-misc-make-logout-work-with-transient-NameID-69740.patch	8,91 ko	04 octobre 2022	Benjamin Dauvergne
0003-views-allow-overriding-the-default-return-url-after-.patch	4,67 ko	04 octobre 2022	Benjamin Dauvergne
0001-utils-use-same_origin-from-authentic2-69740.patch	3,62 ko	05 octobre 2022	Benjamin Dauvergne
0004-misc-keep-nameid-attributes-to-rebuild-it-69740.patch	5,77 ko	05 octobre 2022	Benjamin Dauvergne
0007-views-implement-a-sessionless-logout-endpoint-69740.patch	7,14 ko	05 octobre 2022	Benjamin Dauvergne
0008-misc-clean-SessionIndex-during-logout-69740.patch	11,2 ko	05 octobre 2022	Benjamin Dauvergne
0002-views-improve-handling-of-next_url-for-sp-initiated-.patch	3,72 ko	05 octobre 2022	Benjamin Dauvergne
0005-utils-add-method-to-build-a-session-dump-from-models.patch	9,4 ko	05 octobre 2022	Benjamin Dauvergne
0006-misc-make-logout-work-with-transient-NameID-69740.patch	8,91 ko	05 octobre 2022	Benjamin Dauvergne
0003-views-allow-overriding-the-default-return-url-after-.patch	4,67 ko	05 octobre 2022	Benjamin Dauvergne
0001-utils-use-same_origin-from-authentic2-69740.patch	3,62 ko	05 octobre 2022	Benjamin Dauvergne
0004-misc-keep-nameid-attributes-to-rebuild-it-69740.patch	5,77 ko	05 octobre 2022	Benjamin Dauvergne
0007-views-implement-a-sessionless-logout-endpoint-69740.patch	7,14 ko	05 octobre 2022	Benjamin Dauvergne
0008-misc-clean-SessionIndex-during-logout-69740.patch	11,1 ko	05 octobre 2022	Benjamin Dauvergne
0002-views-improve-handling-of-next_url-for-sp-initiated-.patch	3,43 ko	05 octobre 2022	Benjamin Dauvergne
0005-utils-add-method-to-build-a-session-dump-from-models.patch	9,4 ko	05 octobre 2022	Benjamin Dauvergne
0006-misc-make-logout-work-with-transient-NameID-69740.patch	8,91 ko	05 octobre 2022	Benjamin Dauvergne
0003-views-allow-overriding-the-default-return-url-after-.patch	4,65 ko	05 octobre 2022	Benjamin Dauvergne