w.c.s. - Development #69838

compatibilité E.164 pour les numéros de téléphones préremplis

04 octobre 2022 08:37 - Paul Marillonnet

Statut:	Fermé	**Début:**	04 octobre 2022
Priorité:	Normal	**Echéance:**	
Assigné à:	Paul Marillonnet	**% réalisé:**	0%
Catégorie:		**Temps estimé:**	0:00 heure
Version cible:			
Patch proposed:	Oui	**Planning:**	Non

Description

On a actuellement la possibilité de préremplir un champ numéro de téléphone avec celui connu de l'usager, servi par authentic au SSO.
Dans #49212, on a notamment #69365 qui migre les numéros de téléphone existants dans authentic au format E.164.
Il faut que w.c.s. ait connaissance de ce changement de format, et possiblement reconstruise le numéro local lors du préremplissage des champs.

Demandes liées:

Lié à Publik - Development #49212: Création de compte avec un numéro de télép...	**En cours**	**01 octobre 2021**
Lié à w.c.s. - Development #70937: sql: virer les accesseurs en doublon SqlUs...	**Fermé**	**02 novembre 2022**

Révisions associées

Révision 00df85be - 29 novembre 2022 15:13 - Paul Marillonnet

fields: extend validate_phone_fr validation scope (#69838)

A valid phone number must either be:
· a local phone number dialable within the French numbering system;
· an international (E.164) number whose country code belongs to
French territory (metropolitan or overseas).

Révision d6148c4c - 29 novembre 2022 15:13 - Paul Marillonnet

check_hobos: add default phone field users config (#69838)

Révision d792582c - 29 novembre 2022 15:16 - Paul Marillonnet

fields: prefill users' phone number value (#69838)

Révision 98813e1f - 29 novembre 2022 15:16 - Paul Marillonnet

saml2: test retrieval and storage of user phone at sso time (#69838)

Historique

#1 - 04 octobre 2022 08:37 - Paul Marillonnet

- Lié à Development #49212: Création de compte avec un numéro de téléphone mobile ajouté

#2 - 24 octobre 2022 09:31 - Paul Marillonnet

- Statut changé de Nouveau à En cours

- Assigné à mis à Paul Marillonnet

Concernant les numéros de téléphone français, j'avoue être un peu perplexe : les filtres de gabarit w.c.s. contiennent |phonenumber_fr qui est parfaitement compatible avec un numéro au format E.164, par contre pas le filtre |is_french_mobile_phone_number, qui ne valide que les numéros locaux, et que j'imagine être utilisé ici et là en condition de sortie de page dans les formulaires.

J'en conclue qu'il faut donc que le pré-remplissage tienne compte de la potentielle présence de ce filtre, et transforme systématiquement un numéro au format E.164 en son équivalent local, pour ne pas casser les formulaires existants.

#3 - 25 octobre 2022 10:30 - Paul Marillonnet

- Fichier 0006-fields-prefill-phone-value-69838.patch ajouté

- Fichier 0005-qommon-misc-extend-validate_phone_fr-validation-scop.patch ajouté

- Fichier 0004-users-add-phone-field-69838.patch ajouté

- Fichier 0003-settings-include-phone-field-mapping-69838.patch ajouté

- Fichier 0002-fields-add-builtin-phone-prefill-field-69838.patch ajouté

- Fichier 0001-sql-add-user-phone-field-migration-69838.patch ajouté

- Statut changé de En cours à Solution proposée

- Patch proposed changé de Non à Oui

Dans l'idée, je pense qu'il faudrait quelque chose comme ça, c'est-à-dire le pan w.c.s. de l'ajout du champ côté a2 (#65173) et la compatibilité avec les numéros au format international (qui seront servis par a2 après sa migration locale, #69365).
Je vais ajouter des tests, et ajuster un peu la nouvelle version de validate_phone_fr, qui doit accepter des numéros de préfixes drom-com au format international.

#4 - 25 octobre 2022 10:43 - Benjamin Dauvergne

J'ai l'impression que tu t'es un peu avancé en ajoutant un champ "phone" sur user, je ne suis pas certain que ce soit nécessaire ou souhaité et surtout ça pose un souci du fait des champs mobile et phone qui sont déjà présent dans le user_formdef, mais si tu fais quand même cela il faudrait :

- détecter et migrer les valeurs dans f_phone / f_mobile,
- s'assurer que le provisionning fonctionne toujours de ce coté (que ça pré-remplisse user.f_phone et user.phone),
- s'assurer que les configuration de pré-remplissage sont soit migrées vers user.phone soit fonctionnent encore correctement.

Une de mes peurs signalées c'est que le pré-remplissage se mette à pré-remplir des champs de formulaire avec le format E164 alors qu'on ne transmettait, par exemple via un connecteur, que des numéros courts avant, là où on a de la validation française qui garantissait des numéros courts, il faut continuer à ne pré-remplir qu'avec un numéro court même si dans user.phone on a du E164. Dans tests on devrait avoir un cas où :

```
    assert field.get_prefill_value(user=get_request().user) == ('0123456789', False)
```

#5 - 25 octobre 2022 11:31 - Paul Marillonnet

Benjamin Dauvergne a écrit :

> J'ai l'impression que tu t'es un peu avancé en ajoutant un champ "phone" sur user, je ne suis pas certain que ce soit nécessaire ou souhaité et surtout ça pose un souci du fait des champs mobile et phone qui sont déjà présent dans le user_formdef,

Ok, j'ai loupé ce bout là. À dérouler le code avant d'écrire les patches je ne voyais pas de différence entre la co-existence d'un champ email dans le user_formdef et de la colonne email dans la table users, et la même co-existence pour phone.
Il me semblait qu'au contraire, ajouter une telle colonne avec en plus le code de 0003 nous simplifiait l'affaire.
Ça, et le fait qu'à terme, une fois #49212 passé, j'imagine il va falloir gérer dans w.c.s. des usagers a2 dont l'identifiant est le numéro de téléphone, et qui n'ont pas de mail, me confortent dans l'idée qu'on ajoute cette colonne phone tout de suite.

mais si tu fais quand même cela il faudrait :

> - détecter et migrer les valeurs dans f_phone / f_mobile,

J'ai loupé le moment où ces valeurs elles sont stockées dans w.c.s. De ma compréhension de l'affaire elles sont servies au SSO, et donc une fois #69365 passé, les nouvelles valeurs qui arrivent seront au format international. Tu as un bout de code de w.c.s. à me pointer concernent ce stockage stp, que j'ai apparemment loupé en écrivant ce patch ?

> - s'assurer que le provisionning fonctionne toujours de ce coté (que ça pré-remplisse user.f_phone et user.phone),

Testé en local sur mon devinst, le préremplissage fonctionne, je vais écrire des tests plus costauds.

> - s'assurer que les configuration de pré-remplissage sont soit migrées vers user.phone soit fonctionnent encore correctement.

Pour moi le bout de code de 0002 i.e. l'ajout de l'entrée pour configurer le mapping, avec 0003 i.e. l'ajout de l'entrée builtin dans les options de préremplissage, permet que le pré-remplissage soit migré correctement vers user.phone (d'ailleurs il me semble le test de 0006 échouerait s'il en était autrement).

> Une de mes peurs signalées c'est que le pré-remplissage se mette à pré-remplir des champs de formulaire avec le format E164 alors qu'on ne transmettait, par exemple via un connecteur, que des numéros courts avant, là où on a de la validation française qui garantissait des numéros courts, il faut continuer à ne pré-remplir qu'avec un numéro court même si dans user.phone on a du E164. Dans tests on devrait avoir un cas où :
> [...]

Oui complètement, mais je ne sais pas comment déterminer dans quel cas on préremplit avec le téléphone local. Est-ce que par exemple, dans 0002, il faudrait deux options de téléphone builtins pour le préremplissage, l'une pour le format international et l'autre pour le numéro local ? Ou bien on

décide que côté w.c.s. le préremplissage se fait systématiquement avec le numéro local ?

#6 - 25 octobre 2022 12:47 - Benjamin Dauvergne

Paul Marillonnet a écrit :

> Benjamin Dauvergne a écrit :
>
> > J'ai l'impression que tu t'es un peu avancé en ajoutant un champ "phone" sur user, je ne suis pas certain que ce soit nécessaire ou souhaité et surtout ça pose un souci du fait des champs mobile et phone qui sont déjà présent dans le user_formdef,
>
> Ok, j'ai loupé ce bout là. À dérouler le code avant d'écrire les patches je ne voyais pas de différence entre la co-existence d'un champ email dans le user_formdef et de la colonne email dans la table users, et la même co-existence pour phone.
> Il me semblait qu'au contraire, ajouter une telle colonne avec en plus le code de 0003 nous simplifiait l'affaire.
> Ça, et le fait qu'à terme, une fois #49212 passé, j'imagine il va falloir gérer dans w.c.s. des usagers a2 dont l'identifiant est le numéro de téléphone, et qui n'ont pas de mail, me confortent dans l'idée qu'on ajoute cette colonne phone tout de suite.

On pouvait déjà pré-remplir avec un numéro de téléphone, je ne sais pas ce que ça apporte d'en faire une colonne spéciale (je ne sais pas ce que ça fait pour email non plus désormais en fait, il faudrait que je me rappelle ce que ça apporte sur email).

> mais si tu fais quand même cela il faudrait :
>
> - détecter et migrer les valeurs dans f_phone / f_mobile,
>
> J'ai loupé le moment où ces valeurs elles sont stockées dans w.c.s. De ma compréhension de l'affaire elles sont servies au SSO, et donc une fois #69365 passé, les nouvelles valeurs qui arrivent seront au format international. Tu as un bout de code de w.c.s. à me pointer concernent ce stockage stp, que j'ai apparemment loupé en écrivant ce patch ?

Pour le provisionning push c'est dans wcs.ctl.hobo_notifu.CmdHoboNotify.provision_user et pour le SSO c'est dans wcs/qommon/saml2.py:Saml2Directory.fill_user_attributes mais si c'est géré par User.set_attributes_from_formdata() ça devrait aller, c'est juste qu'il faudrait migrer f_phone en avance, ne pas attendre un SSO ou un provisionning push.

> - s'assurer que les configuration de pré-remplissage sont soit migrées vers user.phone soit fonctionnent encore correctement.
>
> Pour moi le bout de code de 0002 i.e. l'ajout de l'entrée pour configurer le mapping, avec 0003 i.e. l'ajout de l'entrée builtin dans les options de préremplissage, permet que le pré-remplissage soit migré correctement vers user.phone (d'ailleurs il me semble le test de 0006 échouerait s'il en était autrement).

Je ne vois pas de migration dans 0006, on se met directement dans la bonne situation, provisionning défini sur le nouvel attribut phone et user.phone à la bonne valeur.

> > Une de mes peurs signalées c'est que le pré-remplissage se mette à pré-remplir des champs de formulaire avec le format E164 alors qu'on ne transmettait, par exemple via un connecteur, que des numéros courts avant, là où on a de la validation française qui garantissait des numéros courts, il faut continuer à ne pré-remplir qu'avec un numéro court même si dans user.phone on a du E164. Dans tests on devrait avoir un cas où :
> > [...]
>
> Oui complètement, mais je ne sais pas comment déterminer dans quel cas on préremplit avec le téléphone local. Est-ce que par exemple, dans 0002, il faudrait deux options de téléphone builtins pour le préremplissage, l'une pour le format international et l'autre pour le numéro local ? Ou bien on décide que côté w.c.s. le préremplissage se fait systématiquement avec le numéro local ?

Pour moi si tu as field.validation['type'] == 'phone_fr' (pour un StringField) c'est que ça attend un format 10 chiffres, ça vaudrait le coup d'ajouter une fonction de normalisation avec libphonenumber ici et de l'appliquer à la valeur de de prefill dans ce cas.

#7 - 26 octobre 2022 09:21 - Paul Marillonnet

- Statut changé de Solution proposée à En cours

Benjamin Dauvergne a écrit :

> On pouvait déjà pré-remplir avec un numéro de téléphone, je ne sais pas ce que ça apporte d'en faire une colonne spéciale (je ne sais pas ce que ça fait pour email non plus désormais en fait, il faudrait que je me rappelle ce que ça apporte sur email).

Ok, plus on en discute et moins ça me paraît évident qu'il faut une nouvelle colonne, je retire 0001 et 0004 (en gardant en tête qu'au moment où arriveront depuis authentic des usagers sans email il faudra ré-étudier la nécessité de l'ajout de celle-ci).

> Pour le provisionning push c'est dans wcs.ctl.hobo_notifu.CmdHoboNotify.provision_user et pour le SSO c'est dans wcs/qommon/saml2.py:Saml2Directory.fill_user_attributes mais si c'est géré par User.set_attributes_from_formdata() ça devrait aller, c'est juste qu'il faudrait migrer f_phone en avance, ne pas attendre un SSO ou un provisionning push.

Ok, la partie CmdHoboNotify j'ai loupé, je regarde comment adapter ça.

> Je ne vois pas de migration dans 0006, on se met directement dans la bonne situation, provisionning défini sur le nouvel attribut phone et user.phone à la bonne valeur.

Ok, incompréhension de ma part sur la façon dont est géré user.form_data, je revois ça.

> Pour moi si tu as field.validation['type'] == 'phone_fr' (pour un StringField) c'est que ça attend un format 10 chiffres, ça vaudrait le coup d'ajouter une fonction de normalisation avec libphonenumber ici et de l'appliquer à la valeur de de prefill dans ce cas.

Fair enough, je vais mettre ici le format à 10 chiffres comme déjà fait dans les patches a2.

#8 - 26 octobre 2022 10:40 - Benjamin Dauvergne

Paul Marillonnet a écrit :

> Benjamin Dauvergne a écrit :
>
> > On pouvait déjà pré-remplir avec un numéro de téléphone, je ne sais pas ce que ça apporte d'en faire une colonne spéciale (je ne sais pas ce que ça fait pour email non plus désormais en fait, il faudrait que je me rappelle ce que ça apporte sur email).
>
> Ok, plus on en discute et moins ça me paraît évident qu'il faut une nouvelle colonne, je retire 0001 et 0004 (en gardant en tête qu'au moment où arriveront depuis authentic des usagers sans email il faudra ré-étudier la nécessité de l'ajout de celle-ci).

Sincèrement je n'en sais rien, j'aurai bien aimé l'avis de quelqu'un d'autre que le mien, je suis plus à l'aise pour dire quoi dans authentic.

#9 - 31 octobre 2022 16:15 - Paul Marillonnet

Bon j'ai avancé dans le sens des remarques faites dans ce ticket concernant la validation de numéro de téléphone français, le provisionning et le backend saml. Pour l'instant c'est trop peu testé, je dois encore blinder tout ça, mais dans l'idée ce sera quelque chose tel que déjà visible dans la branche : https://git.entrouvert.org/wcs.git/log/?h=wip/69838-wcs-phone-prefill-e164

#10 - 02 novembre 2022 10:45 - Paul Marillonnet

- Fichier 0005-users-add-phone-field-69838.patch ajouté

- Fichier 0006-setup-add-phonenumbers-dependency-69838.patch ajouté

- Fichier 0007-qommon-misc-extend-validate_phone_fr-validation-scop.patch ajouté

- Fichier 0008-qommon-misc-add-e164-format-phone-representation-uti.patch ajouté

- Fichier 0009-qommon-misc-add-local-phone-number-representation-ut.patch ajouté

- Fichier 0010-test_templates-provide-a-valid-07-french-mobile-test.patch ajouté

- Statut changé de En cours à Solution proposée

- Fichier 0001-sql-add-user-phone-field-migration-69838.patch ajouté

- Fichier 0002-fields-add-builtin-phone-prefill-field-69838.patch ajouté

- Fichier 0003-admin-include-phone-field-mapping-69838.patch ajouté

- Fichier 0004-settings-add-default-phone-number-country-code-69838.patch ajouté

Paul Marillonnet a écrit :

> Bon j'ai avancé dans le sens des remarques faites dans ce ticket concernant la validation de numéro de téléphone français, le provisionning et le backend saml. Pour l'instant c'est trop peu testé, je dois encore blinder tout ça, mais dans l'idée ce sera quelque chose tel que déjà visible dans la branche : https://git.entrouvert.org/wcs.git/log/?h=wip/69838-wcs-phone-prefill-e164

Et la version testée est là. Ça reprend toutes les remarques ici dans ce ticket.

Finalement j'ai conservé la colonne nouvellement ajoutée, ça me simplifie le travail et je trouve ça plus rigoureux qu'un attribut qui est identifiant pour l'usager fasse l'objet d'un colonne à part entière (plutôt qu'une simple entrée dans le user.form_data dont la clé cfg['users']['field_phone'] peut varier).

#11 - 02 novembre 2022 10:45 - Paul Marillonnet

- Fichier 0011-fields-prefill-phone-value-69838.patch ajouté

- Fichier 0012-hobo_notify-save-user-s-phone-number-upon-provisionn.patch ajouté

- Fichier 0013-saml2-retrieve-and-store-user-phone-at-sso-time-6983.patch ajouté

Et les trois restants (treize patches en tout).

#15 - 02 novembre 2022 12:39 - Frédéric Péters

(première lecture de loin)

```
+    def get_users_with_phone(cls, phone):
+ (...)
```

Ça peut être réécrit de manière plus courte, façon

```
    def get_users_with_phone(cls, phone):
        return cls.select([Null('deleted_timestamp'), Equal('phone', phone])
```

(bien sûr les autres get_users_... n'ont pas cette forme, ça serait autre ticket pour faire évoluer ça, mais autant partir court ici).

(à découper les choses ça serait dans un autre commit que "add user phone field migration").

Concernant la migration :

```
+    if sql_level < 68:
+        do_user_table()
```

On fera plutôt une modification à l'appel existant, ex:

```
-    if sql_level < 65:
+    if sql_level < 68:
        # 3: introduction of _structured for user fields
        # 4: removal of identification_token
        # 12: (first part) add fts to users
        # 16: add verified_fields to users
        # 21: (first part) add ascii_name to users
        # 39: add deleted_timestamp
        # 40: add is_active to users
        # 65: index users(name_identifiers)
+       # 68: add user phone field
        do_user_table()
```

Aussi il me semble manquer le code pour créer la colonne si elle n'existait pas,

```
    if 'phone' not in existing_fields:
        cur.execute('ALTER TABLE %s ADD COLUMN ...
```

- qommon/misc: extend validate_phone_fr validation scope (#69838)

La partie préfixe des messages l'idée n'est pas de faire une redite du nom de fichier (bêtement ça pourrait juste être misc ici, dans le sens "divers" pas dans le sens "ça se passe dans le fichier misc.py", mais ça pourrait aussi être fields, dans le sens où ça concerne la validation de données dans les champs).

Je m'arrête aussi sur ce message pour qu'il soit un peu étendu, pour expliciter ce qui est désormais autorisé.

```
+    default_country_code = settings.DEFAULT_COUNTRY_CODE
```

Il ne faut pas passer via les settings, la bonne méthode dans w.c.s. c'est soit dans /backoffice/settings/, soit via get_site_option. (particulièrement il n'y a pas d'override via un settings.json dans w.c.s.).

```
+    try_e164_format,
```

Pas tout à fait à l'aise avec "try", à regarder le code je comprends que c'est "essaie de formater ainsi et sinon renvoie juste la chaine" mais je pense que je serais plus confortable à juste lire : get_phone_in_e164_format(phone) or phone (avec donc une méthode qui retournerait None en cas de numéro invalide).

(la découpe des commits, qui fait qu'on ne voit pas où/comment c'est appelé, complique).

#16 - 02 novembre 2022 14:33 - Paul Marillonnet

- Statut changé de Solution proposée à En cours

Ok merci pour la relecture, je corrige ces différents soucis (ce qui ne doit pas empêcher une relecture sur le fond de survenir entre temps).

#17 - 02 novembre 2022 15:55 - Paul Marillonnet

- Statut changé de En cours à Solution proposée

J'ai corrigé dans la branche ce qui a été remarqué ici.

Je reviens juste sur certaines remarques :

> [...]
>
> Ça peut être réécrit de manière plus courte, façon
>
> [...]
>
> (bien sûr les autres get_users_... n'ont pas cette forme, ça serait autre ticket pour faire évoluer ça, mais autant partir court ici).

Dans ma compréhension, si on prend la double définition de par exemple get_users_with_email, à la fois dans User et sa classe fille SqlUser, on a d'abord la méthode dans User qui fait appel à l'ORM :

```
    @classmethod
    def get_users_with_email(cls, email):
        return cls.select([st.Null('deleted_timestamp'), st.Equal('email', email)])
```

puis dans SqlUser à des fins d'optimisation on tape le SQL directement.

J'ai fait strictement la même chose pour get_users_with_phone, la première des deux méthodes étant rigoureusement celle que tu proposes ici. Je loupe un truc ?

> Aussi il me semble manquer le code pour créer la colonne si elle n'existait pas,

Bien vu, merci, j'ai cru à tort que c'était géré de façon générique via la liste des needed_fields, mauvaise compréhension du code de ma part. J'ai ajouté un test.

> Je m'arrête aussi sur ce message pour qu'il soit un peu étendu, pour expliciter ce qui est désormais autorisé.

C'est fait.

> Il ne faut pas passer via les settings, la bonne méthode dans w.c.s. c'est soit dans /backoffice/settings/, soit via get_site_option. (particulièrement il n'y a pas d'override via un settings.json dans w.c.s.).

Ok, passé par get_site_option.

> Pas tout à fait à l'aise avec "try", à regarder le code je comprends que c'est "essaie de formater ainsi et sinon renvoie juste la chaine" mais je pense que je serais plus confortable à juste lire : get_phone_in_e164_format(phone) or phone (avec donc une méthode qui retournerait None en cas de numéro invalide).

Ok.

> (la découpe des commits, qui fait qu'on ne voit pas où/comment c'est appelé, complique).

J'avoue ne pas aimer mettre dans un même commit la définition d'une fonction utilitaire et une occurrence très spécifique de son utilisation (provisionning, sso, etc.). J'ai collé les commits entre eux pour qu'on voie l'utilisation de la fonction dans un commit assez proche de celui de sa définition.

#18 - 02 novembre 2022 16:09 - Frédéric Péters

> Dans ma compréhension, si on prend la double définition de par exemple get_users_with_email, à la fois dans User et sa classe fille SqlUser, on a d'abord la méthode dans User qui fait appel à l'ORM :

J'ai bien du mal à capter dans le patch je vois juste du code dans SqlUser pas d'autre méthode.

Ah, c'est quelque chose qui arrive ~6 commits plus tard :/

Mais ok donc, ça ne sert à rien d'ajouter cette méthode (dans sql.py), [Null('deleted_timestamp'), Equal('phone', phone)] fait le même taf.

#19 - 02 novembre 2022 16:15 - Paul Marillonnet

Frédéric Péters a écrit :

> Dans ma compréhension, si on prend la double définition de par exemple get_users_with_email, à la fois dans User et sa classe fille SqlUser, on a d'abord la méthode dans User qui fait appel à l'ORM :

J'ai bien du mal à capter dans le patch je vois juste du code dans SqlUser pas d'autre méthode.

Ah, c'est quelque chose qui arrive ~6 commits plus tard :/

Pardon, commits mal rangés encore.

Mais ok donc, ça ne sert à rien d'ajouter cette méthode (dans sql.py), [Null('deleted_timestamp'), Equal('phone', phone)] fait le même taf.

Ok, je dégage ce commit de la branche, et ferai un ticket pour dégager toutes les méthodes get_users_with_* en doublon entre users.User et sql.SqlUser.

#20 - 02 novembre 2022 16:34 - Paul Marillonnet

- Lié à Development #70937: sql: virer les accesseurs en doublon SqlUser.get_users_with_* ajouté

#21 - 09 novembre 2022 09:49 - Benjamin Dauvergne

- fields: add builtin phone prefill field (#69838) je pense qu'il faut ajouter phone dans la branche else aussi vu que c'est un

```
+            if not users_cfg.get('field_phone'):
+                user_fields.append(('phone', _('Phone (builtin)')))
         else:
             user_fields = [('name', _('Name')), ('email', _('Email'))]
```

  en fait le code pourrait éviter une redite en enlevant le if et en faisant for user_field in (formdef.fields if formdef else []):
- admin: include phone field mapping (#69838) jamais vu un cas ou field_email était modifié pour autre chose que la colonne email par défaut, alors je ne sais pas si c'est une très bonne idée de perpétuer ça pour phone (ça doit dater du mode pickle où User n'avait peut-être pas de champ email natif...)
- users: add phone field (#69838) je mergerai ça avec sql: add user phone field migration (#69838) qui contient déjà un o.phone qui plantera sinon
- setup: add phonenumbers dependency (#69838) merger avec le suivant fields: extend validate_phone_fr validation scope (#69838)
- misc: add e164 format phone representation utility (#69838) aussi avec misc: add local phone number representation utility (#69838) (l'historique n'a pas grand intérêt)
- fields: prefill phone value (#69838) on utilise get_local_number() si la validation est phone-fr mais get_local_number() ne formate pas toujours vers la locale française vu la ligne :

```
     default_country_code = get_publisher().get_site_option('default-country-code') or 'FR'
```

  je serais d'avis d'avoir ici un appel get_local_number(user.phone, country_code='FR') or user.phone

#22 - 09 novembre 2022 23:40 - Thomas Noël

Benjamin Dauvergne a écrit :

> je serais d'avis d'avoir ici un appel get_local_number(user.phone, country_code='FR') or user.phone

Ca se comporte comment pour des numéros à la Réunion ?

#25 - 10 novembre 2022 11:17 - Benjamin Dauvergne

Bon je dis des bêtises, libphonenumber ne peut pas détecter les numéros des DOM/TOM dans le plan de numérotation national, il ne faut pas tenter de parser un numéro local et de deviner le préfixe et on ne peut pas se reposer sur le fait que le default_country_code serait bien configuré sur toutes les instances Belge et à la Réunion, c'est du stress pour rien.

Ici:

- si une validation_phone_fr est présente, et si un numéro est au format E164 (il est accepté par un bête phonenumbers.parse()) et si son préfixe est l'un des préfixes français (FR, RE, etc..) , alors ok on le reformate en ='FR' en numéro court
- si une validation_phone est faite, on parse en E164, ça passe on formate joli, si le truc n'est pas au format E164, on pré-remplit tel quel et c'est tout.

#26 - 10 novembre 2022 11:46 - Paul Marillonnet

Benjamin Dauvergne a écrit :

> […] on ne peut pas se reposer sur le fait que le default_country_code serait bien configuré sur toutes les instances Belge et à la Réunion, c'est du stress pour rien.

Ça me paraissait jouable.

> Ici:
>
> - si une validation_phone_fr est présente, et si un numéro est au format E164 (il est accepté par un bête phonenumbers.parse()) et si son préfixe est l'un des préfixes français (FR, RE, etc..) , alors ok on le reformate en ='FR' en numéro court
> - si une validation_phone est faite, on parse en E164, ça passe on formate joli, si le truc n'est pas au format E164, on pré-remplit tel quel et c'est tout.

Ok, on peut faire comme ça.

Le truc important dans ce ticket c'est que ça part de l'hypothèse que #69365 existe côté a2 et que les numéros reçus sont au format international. Là apparemment changement de plan et il faut que je me remette les choses en tête à la fois ici et côté a2 pour voir ce qui des patches proposés tient encore la route malgré ce changement, et ce qui est à revoir.

#27 - 10 novembre 2022 11:50 - Benjamin Dauvergne

Paul Marillonnet a écrit :

> Le truc important dans ce ticket c'est que ça part de l'hypothèse que #69365 existe côté a2 et que les numéros reçus sont au format international. Là apparemment changement de plan et il faut que je me remette les choses en tête à la fois ici et côté a2 pour voir ce qui des patches proposés tient encore la route malgré ce changement, et ce qui est à revoir.

Oui c'est cette hypothèse qu'il faut éliminer, on doit supposer ici qu'on continue à recevoir n'importe quoi mais qu'on essaie de bien gérer les choses si on reçoit des numéros bien formatés en E164; et dans ce cadre je ne suis pas certain que tous le boulot sur la colonne user.phone soit nécessaire, on pourrait juste corriger le pré-remplissage validate_phone_fr et ce serait suffisant j'ai l'impression.

#28 - 10 novembre 2022 12:03 - Paul Marillonnet

Benjamin Dauvergne a écrit :

> Oui c'est cette hypothèse qu'il faut éliminer, on doit supposer ici qu'on continue à recevoir n'importe quoi mais qu'on essaie de bien gérer les choses si on reçoit des numéros bien formatés en E164; et dans ce cadre je ne suis pas certain que tous le boulot sur la colonne user.phone soit nécessaire, on pourrait juste corriger le pré-remplissage validate_phone_fr et ce serait suffisant j'ai l'impression.

Ok, clairement cette colonne c'était surtout de ma part une anticipation de la situation où côté a2 on aurait des usagers créés sans courriel, et où le numéro de téléphone aurait une teneur identifiante pour l'usager. Je me voyais mal côté w.c.s. piocher un truc aussi important dans user.formdata avec une clé qui peut varier.

#29 - 10 novembre 2022 13:26 - Frédéric Péters

(pour info, perso je suis ok avec une colonne)

#31 - 16 novembre 2022 13:51 - Paul Marillonnet

- Fichier 0001-add-user-phone-field-subsequent-migration-69838.patch ajouté

- Fichier 0002-fields-add-builtin-phone-prefill-field-69838.patch ajouté

- Fichier 0003-fields-extend-validate_phone_fr-validation-scope-698.patch ajouté

- Fichier 0004-tests-provide-a-valid-07-french-mobile-test-number-6.patch ajouté

- Fichier 0005-misc-add-phone-representation-utilities-69838.patch ajouté

- Fichier 0006-fields-prefill-phone-value-69838.patch ajouté

- Fichier 0007-hobo_notify-save-user-s-phone-number-upon-provisionn.patch ajouté

- Fichier 0008-saml2-retrieve-and-store-user-phone-at-sso-time-6983.patch ajouté

La version à jour, raccord avec ce qui a été discuté plus haut, notamment :

- si une validation_phone_fr est présente, et si un numéro est au format E164 (il est accepté par un bête phonenumbers.parse()) et si son préfixe est l'un des préfixes français (FR, RE, etc..) , alors ok on le reformate en ='FR' en numéro court
- si une validation_phone est faite, on parse en E164, ça passe on formate joli, si le truc n'est pas au format E164, on pré-remplit tel quel et c'est tout.

#32 - 17 novembre 2022 15:15 - Frédéric Péters

(pur détail mais 0003, et c'était pareil dans d'autres commentaires, les lignes de la description du commit ne devraient pas être indentées).

(forme aussi : ça reste divisé en commits, avec du code ajouté sans être appelé, ce qui fait quelque chose de trop compliqué à relire, par exemple je relisais get_local_number et j'aurais aimé voir comment il était appelé).

Pour parler de get_local_number donc, perso pas fan du fonctionnement de retourner None quand il n'y a pas de default-country-code, qui fait que derrière on rattrape les choses en ajoutant des "or user.phone";

Si j'arrive à suivre quand même, get_formatted_phone c'est uniquement appelé par get_local_number, ça appelle à nettoyer simplifier dès maintenant, genre une seule fonction et cette partie ne sert plus :

```
    if number_format == 'local':
        pn_format = phonenumbers.PhoneNumberFormat.NATIONAL
    elif number_format == 'international':
        pn_format = phonenumbers.PhoneNumberFormat.E164
    else:
        return None
```

(pause à relire de l'historique de tickets et du code)

0008-saml2-retrieve-and-store-user-phone-at-sso-time-6983.patch : modifie des tests et ajoute deux lignes blanches. (???)

Grosso modo désolé c'est incroyablement confus.

Les changements de direction n'aident bien sûr pas : je n'arrive plus vraiment à capter le but général, ce qui va changer, ou pas, côté authentic, qui demande des adaptations, ou pas, ici.

Si je comprends bien il y a deux choses :

- enregistrer le téléphone dans un attribut/colonne de users, ok
- faire en sorte que si on est servi un numéro au format international, que celui-ci est utilisé pour préremplir un champ configuré pour avoir une validation téléphonique, une version "locale" du numéro soit affichée.

Si c'est bien ça, je serais pour réunir les commits du premier point en un seul commit, puis tout ce second point dans une méthode get_formatted_phone de l'utilisateur; c'est-à-dire sans avoir testé, un commit qui ferait :

```
@@ -396,6 +396,20 @@ class User(StorableObject):
         if status != 200:
             get_publisher().record_error(_('Failed to call keepalive API (status: %s)') % status)

+    def get_formatted_phone(self, country_code):
+        try:
+            pn = phonenumbers.parse(self.phone)
+        except phonenumbers.NumberParseException:
+            try:
+                pn = phonenumbers.parse(phone, country_code)
+            except phonenumbers.NumberParseException:
+                return self.phone
+
+        if phonenumbers.is_valid_number(pn):
+            return phonenumbers.format_number(pn, pn_format)
+        else:
+            return self.phone
+
```

+

```
@@ -541,13 +541,10 @@ class Field:
             if x == 'email':
                 return (user.email, explicit_lock or 'email' in (user.verified_fields or []))
             elif x == 'phone':
-                phone = user.phone
-                if hasattr(self, 'validation'):
-                    validation_type = self.validation.get('type', None)
-                    if validation_type == 'phone-fr':
-                        phone = get_local_number(user.phone, country_code='FR') or user.phone
-                    elif validation_type == 'phone':
-                        phone = get_local_number(user.phone) or user.phone
+                country_code = get_publisher().get_site_option('default-country-code')
+                if getattr(self, 'validation', None) and self.validation.get('type') == 'phone-fr':
+                    country_code = 'FR'
+                phone = user.get_formatted_phone(country_code=country_code)
                 return (phone, explicit_lock or 'phone' in (user.verified_fields or []))
```

```
            elif user.form_data:
                userform = user.get_formdef()
```

#33 - 18 novembre 2022 09:39 - Paul Marillonnet

- Statut changé de Solution proposée à En cours

Ok, merci pour la relecture en détails. En effet des changements de direction, regroupements de commits et j'ai perdu le fil sur certains points moi aussi, je reprends tout ça en fonction des retours que tu as faits ici.

#34 - 18 novembre 2022 10:06 - Benjamin Dauvergne

Toujours pas chaud que l'idée se perpétue que user.email ou user.phone soient des identifiants des utilisateurs dans w.c.s. Ce n'est pas le cas, il n'y en a que deux sources : user.id et user.name_identifiers[0/1/3/...] les autres sont là pour faire joli et rien n'en fait des identifiants (ils ne sont certainement pas garantis uniques sauf parce qu'authentic le veut bien et aucun code ne fait cette supposition dans w.c.s).

Qu'un certain champ phone soit désigné le champ numéro de téléphone par défaut pour l'envoi de SMS (comme le champ email l'est pour l'action envoi de mail via FormDef.get_submitter_email()) pas de souci, mais ça ne nécessite pas une colonne en plus, éventuellement une property sur User.

Je donne peut-être l'impression d'être doctoral sur cette question mais w.c.s. veut juste envoyer des SMS, pas associer les utilisateurs à des numéros de téléphone, c'est le boulot d'authentic ça.

#35 - 18 novembre 2022 10:16 - Frédéric Péters

(je ne vais pas me battre sur l'ajout dans une colonne dédiée)

#36 - 18 novembre 2022 10:18 - Paul Marillonnet

Ok. Et le reste du plan a2 n'étant plus du tout clair pour moi, je vais déjà avancer côté w.c.s. sans cette colonne.

#37 - 22 novembre 2022 14:59 - Paul Marillonnet

- Fichier 0001-fields-add-builtin-phone-prefill-field-69838.patch ajouté

- Fichier 0002-fields-extend-validate_phone_fr-validation-scope-698.patch ajouté

- Fichier 0003-fields-prefill-users-phone-number-value-69838.patch ajouté

- Fichier 0004-hobo_notify-test-phone-number-provisionning-69838.patch ajouté

- Fichier 0005-saml2-test-retrieval-and-storage-of-user-phone-at-ss.patch ajouté

- Statut changé de En cours à Solution proposée

Frédéric Péters a écrit :

> (pur détail mais 0003, et c'était pareil dans d'autres commentaires, les lignes de la description du commit ne devraient pas être indentées).

Ok, étrange, c'est mon vim qui fait ça, je pensais que cette indentation était le comportement normal lorsqu'il est configuré comme éditeur de message de commit git. C'est corrigé.

> (forme aussi : ça reste divisé en commits, avec du code ajouté sans être appelé, ce qui fait quelque chose de trop compliqué à relire, par exemple je relisais get_local_number et j'aurais aimé voir comment il était appelé).

Ok, désolé (mauvaise habitude de mon côté que de préférer relire la définition de fonctions et leurs usages très spécifiques dans des commits séparés), j'ai revu à la baisse le nombre de commits en regroupant les choses que j'estimais pouvoir être relues ensemble.

> 0008-saml2-retrieve-and-store-user-phone-at-sso-time-6983.patch : modifie des tests et ajoute deux lignes blanches. (???)

Étape par étape j'ai revu à la baisse les ambitions de cette partie du code, sans me rendre compte que le commit ne faisait plus rien d'autre que d'ajouter des tests, mes excuses.
J'ai retiré les deux lignes blanches et ai corrigé le message de commit.
Il en est pareil du désormais 0004, qui se contente maintenant d'ajouter des tests sur la partie provisionning hobo.

> Les changements de direction n'aident bien sûr pas : je n'arrive plus vraiment à capter le but général, ce qui va changer, ou pas, côté authentic, qui demande des adaptations, ou pas, ici.
>
> Si je comprends bien il y a deux choses :
>
> - enregistrer le téléphone dans un attribut/colonne de users, ok

Et donc cette partie est passée à la trappe désormais.

- faire en sorte que si on est servi un numéro au format international, que celui-ci est utilisé pour préremplir un champ configuré pour avoir une validation téléphonique, une version "locale" du numéro soit affichée.

> Si c'est bien ça, je serais pour réunir les commits du premier point en un seul commit, puis tout ce second point dans une méthode get_formatted_phone de l'utilisateur; c'est-à-dire sans avoir testé, un commit qui ferait :
>
> [...]
>
> +
>
> [...]

Ok, ça me va très bien de passer par une méthode de l'utilisateur.
J'ai revu tout ça et surtout ai revu à la baisse le nombre de commits, j'espère que ça facilitera une nouvelle relecture. Mes excuses pour le manque de clarté des patches précédents, écrits et mis en forme un peu précipitamment pour respecter une échéance désormais loin derrière nous.

#38 - 25 novembre 2022 10:16 - Frédéric Péters

```
+            if not users_cfg.get('field_phone'):
+                user_fields.append(('phone', _('Phone (builtin)')))
```

S'il n'y a plus de colonne native pour le téléphone ça ne me semble plus approprié; il me semble que 0001 est à faire sauter.

C'est peut-être lié à un truc plus loin, 0002,

```
+        phone = self.form_data.get('phone')
+        if not phone:
+            field_phone = users_cfg.get('field_phone')
+            phone = self.form_data.get(field_phone)
```

Si users_cfg.get('field_phone') n'est pas défini on ne doit rien prendre, pas de fallback sur self.form_data.get('phone').

Ça veut aussi dire quelque part une modification à wcs/ctl/check_hobos.py pour poser la configuration, type

```
--- a/wcs/ctl/check_hobos.py
+++ b/wcs/ctl/check_hobos.py
@@ -322,6 +322,7 @@ class CmdCheckHobos(Command):
         formdef.store()

         pub.cfg['users']['field_email'] = '_email'
+        if not pub.cfg['users'].get('field_phone'):
+            pub.cfg['users']['field_phone'] = '_phone'
         if not (pub.cfg['users'].get('fullname_template') or pub.cfg['users'].get('field_name')):
             pub.cfg['users'][
                 'fullname_template'
```

#39 - 29 novembre 2022 10:06 - Paul Marillonnet

- Fichier 0001-fields-extend-validate_phone_fr-validation-scope-698.patch ajouté

- Fichier 0002-check_hobos-add-default-phone-field-users-config-698.patch ajouté

- Fichier 0003-fields-prefill-users-phone-number-value-69838.patch ajouté

- Fichier 0004-saml2-test-retrieval-and-storage-of-user-phone-at-ss.patch ajouté

Frédéric Péters a écrit :

> [...]
>
> S'il n'y a plus de colonne native pour le téléphone ça ne me semble plus approprié; il me semble que 0001 est à faire sauter.

Ok, en effet, un oubli de ma part.

> C'est peut-être lié à un truc plus loin, 0002,
> Si users_cfg.get('field_phone') n'est pas défini on ne doit rien prendre, pas de fallback sur self.form_data.get('phone').

Ok, j'ai retiré ce fallback.

> Ça veut aussi dire quelque part une modification à wcs/ctl/check_hobos.py pour poser la configuration, type

[...]

Ok, j'ai ajouté ça, mis deux assert avant et après l'appel pertinent à CheckHobos.update_profile(…) dans les tests pour vérifier que la configuration est bien posée, et ai intégré à ça le commit tests déjà existant pour cette partie provisionning. C'est maintenant 0002.

#40 - 29 novembre 2022 14:20 - Frédéric Péters

```
+                return (phone, explicit_lock or 'phone' in (user.verified_fields or []))
```

Pas sûr de cette partie 'phone' in (user.verified_fields or []), il me semble qu'il doit y être fait référence à l'identifiant du champ, et que ça sera '_phone', via pub.cfg['users']['field_phone'].

#41 - 29 novembre 2022 15:20 - Paul Marillonnet

Frédéric Péters a écrit :

> Pas sûr de cette partie 'phone' in (user.verified_fields or []), il me semble qu'il doit y être fait référence à l'identifiant du champ, et que ça sera '_phone', via pub.cfg['users']['field_phone'].

Oui, bien vu, dans la méthode wcs.qommon.saml2.Saml2Directory.fill_user_attributes, c'est l'identifiant du champ qui est utilisé pour remplir user.verifield_fields, c'est corrigé dans la branche.

#42 - 29 novembre 2022 15:23 - Frédéric Péters

- Statut changé de Solution proposée à Solution validée

Pour moi il n'y avait que ça, go une fois jenkins content.

#43 - 29 novembre 2022 15:40 - Paul Marillonnet

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

```
commit 98813e1fbb9a53da8e0ed8fa4ec0169a2a5b1253
Author: Paul Marillonnet <pmarillonnet@entrouvert.com>
Date:   Mon Oct 31 11:05:49 2022 +0100

    saml2: test retrieval and storage of user phone at sso time (#69838)

commit d792582c6f60ea1e15200cfe46007960db1c8dab
Author: Paul Marillonnet <pmarillonnet@entrouvert.com>
Date:   Mon Nov 21 16:35:34 2022 +0100

    fields: prefill users' phone number value (#69838)

commit d6148c4c6f27d43d3611ac1e115a2a5844bbaaa3
Author: Paul Marillonnet <pmarillonnet@entrouvert.com>
Date:   Tue Nov 29 09:41:01 2022 +0100

    check_hobos: add default phone field users config (#69838)

commit 00df85be5ff5a8f7c5b7248a4535aed3a5137d99
Author: Paul Marillonnet <pmarillonnet@entrouvert.com>
Date:   Fri Oct 28 10:40:38 2022 +0200

    fields: extend validate_phone_fr validation scope (#69838)

    A valid phone number must either be:
    · a local phone number dialable within the French numbering system;
    · an international (E.164) number whose country code belongs to
    French territory (metropolitan or overseas).
```

#44 - 29 novembre 2022 20:15 - Transition automatique

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

#45 - 29 janvier 2023 04:42 - Transition automatique

Automatic expiration

Fichiers

0006-fields-prefill-phone-value-69838.patch	1,98 ko	25 octobre 2022	Paul Marillonnet

0005-qommon-misc-extend-validate_phone_fr-validation-scop.patch	1,44 ko	25 octobre 2022	Paul Marillonnet
0004-users-add-phone-field-69838.patch	2,74 ko	25 octobre 2022	Paul Marillonnet
0003-settings-include-phone-field-mapping-69838.patch	1,25 ko	25 octobre 2022	Paul Marillonnet
0002-fields-add-builtin-phone-prefill-field-69838.patch	904 octets	25 octobre 2022	Paul Marillonnet
0001-sql-add-user-phone-field-migration-69838.patch	3,09 ko	25 octobre 2022	Paul Marillonnet
0002-fields-add-builtin-phone-prefill-field-69838.patch	1,91 ko	02 novembre 2022	Paul Marillonnet
0001-sql-add-user-phone-field-migration-69838.patch	4,48 ko	02 novembre 2022	Paul Marillonnet
0003-admin-include-phone-field-mapping-69838.patch	1,25 ko	02 novembre 2022	Paul Marillonnet
0004-settings-add-default-phone-number-country-code-69838.patch	770 octets	02 novembre 2022	Paul Marillonnet
0005-users-add-phone-field-69838.patch	3,58 ko	02 novembre 2022	Paul Marillonnet
0006-setup-add-phonenumbers-dependency-69838.patch	586 octets	02 novembre 2022	Paul Marillonnet
0007-qommon-misc-extend-validate_phone_fr-validation-scop.patch	2,19 ko	02 novembre 2022	Paul Marillonnet
0008-qommon-misc-add-e164-format-phone-representation-uti.patch	1,89 ko	02 novembre 2022	Paul Marillonnet
0009-qommon-misc-add-local-phone-number-representation-ut.patch	2,3 ko	02 novembre 2022	Paul Marillonnet
0010-test_templates-provide-a-valid-07-french-mobile-test.patch	1,33 ko	02 novembre 2022	Paul Marillonnet
0011-fields-prefill-phone-value-69838.patch	4,62 ko	02 novembre 2022	Paul Marillonnet
0012-hobo_notify-save-user-s-phone-number-upon-provisionn.patch	3,97 ko	02 novembre 2022	Paul Marillonnet
0013-saml2-retrieve-and-store-user-phone-at-sso-time-6983.patch	3,67 ko	02 novembre 2022	Paul Marillonnet
0002-fields-add-builtin-phone-prefill-field-69838.patch	2,08 ko	16 novembre 2022	Paul Marillonnet
0001-add-user-phone-field-subsequent-migration-69838.patch	8,24 ko	16 novembre 2022	Paul Marillonnet
0003-fields-extend-validate_phone_fr-validation-scope-698.patch	2,81 ko	16 novembre 2022	Paul Marillonnet
0004-tests-provide-a-valid-07-french-mobile-test-number-6.patch	1,32 ko	16 novembre 2022	Paul Marillonnet
0005-misc-add-phone-representation-utilities-69838.patch	3,21 ko	16 novembre 2022	Paul Marillonnet
0006-fields-prefill-phone-value-69838.patch	8,22 ko	16 novembre 2022	Paul Marillonnet
0008-saml2-retrieve-and-store-user-phone-at-sso-time-6983.patch	3,14 ko	16 novembre 2022	Paul Marillonnet
0007-hobo_notify-save-user-s-phone-number-upon-provisionn.patch	3,62 ko	16 novembre 2022	Paul Marillonnet
0001-fields-add-builtin-phone-prefill-field-69838.patch	2,08 ko	22 novembre 2022	Paul Marillonnet
0002-fields-extend-validate_phone_fr-validation-scope-698.patch	3,62 ko	22 novembre 2022	Paul Marillonnet
0003-fields-prefill-users-phone-number-value-69838.patch	11,1 ko	22 novembre 2022	Paul Marillonnet
0004-hobo_notify-test-phone-number-provisionning-69838.patch	3,03 ko	22 novembre 2022	Paul Marillonnet
0005-saml2-test-retrieval-and-storage-of-user-phone-at-ss.patch	2,28 ko	22 novembre 2022	Paul Marillonnet
0001-fields-extend-validate_phone_fr-validation-scope-698.patch	3,62 ko	29 novembre 2022	Paul Marillonnet
0002-check_hobos-add-default-phone-field-users-config-698.patch	3,96 ko	29 novembre 2022	Paul Marillonnet
0003-fields-prefill-users-phone-number-value-69838.patch	11,1 ko	29 novembre 2022	Paul Marillonnet
0004-saml2-test-retrieval-and-storage-of-user-phone-at-ss.patch	2,28 ko	29 novembre 2022	Paul Marillonnet