

## Passerelle - Development #39650

### connecteurs SMS, option pour refuser les numéros surchargés

07 février 2020 15:56 - Frédéric Péters

<b>Statut:</b>	Fermé	<b>Début:</b>	07 février 2020
<b>Priorité:</b>	Normal	<b>Echéance:</b>	
<b>Assigné à:</b>	Nicolas Roche	<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		

#### Description

Plutôt qu'avoir à faire des validations côté client (ex: #39478) avoir une option, activée par défaut, qui fasse qu'un connecteur de diffusion SMS refuse les numéros particuliers.

#### Demandes liées:

Lié à w.c.s. - Development #39760: Pouvoir tester avec filtre Django dans une...	<b>Rejeté</b>	<b>11 février 2020</b>
Lié à Passerelle - Development #42426: Déplacer le code générique des connect...	<b>Fermé</b>	<b>04 mai 2020</b>
Lié à Passerelle - Development #42427: Déplacer le code clean_number utilisés...	<b>Fermé</b>	<b>04 mai 2020</b>
Lié à Passerelle - Development #42627: sms: ajouter une classe SMSConnectorForm	<b>Rejeté</b>	<b>07 mai 2020</b>

#### Révisions associées

##### Révision 93071920 - 01 mars 2021 10:20 - Nicolas Roche

sms: filter authorized numbers (#39650)

#### Historique

##### #1 - 11 février 2020 14:53 - Frédéric Péters

- Lié à Development #39760: Pouvoir tester avec filtre Django dans une condition qu'un numéro de téléphone est un numéro de mobile Français. ajouté

##### #2 - 11 février 2020 15:12 - Frédéric Péters

+ parallèlement option pour interdire les numéros étranger.

##### #3 - 11 février 2020 15:37 - Thomas Noël

On pourrait partir vers des "masques d'autorisation", avec un format qui ne soit pas compliqué (pas de regex), dans les IPBX tels qu'Asterisk c'est ainsi :

Pour des numéros français commençant par 06 et avec 10 chiffres (le X est un seul chiffre) :

```
0 [6-7]XXXXXXXXX
0033 [6-7]XXXXXXXXX
+33 [6-7]XXXXXXXXX
```

Pour la Belgique ou des pays où on ne connaît pas la longueur du numéro, c'est le "." qui veut dire « n'importe quoi après »

```
0032.
```

Bref, on aurait dans le connecteur un textarea avec une ligne par masque. C'est un peu moche mais super facile à comprendre/lire, je trouverais ça mieux qu'une option un peu magique qui ne sera jamais parfaitement complète.

(A noter que certains opérateurs ont de vrais outils de vérification disponibles, comme OVH et son HLR, mais ça serait l'objet d'un autre ticket spécifique, si un jour)

##### #4 - 11 mars 2020 14:57 - Valentin Deniaud

Frédéric Péters a écrit :

une option, activée par défaut, qui fasse qu'un connecteur de diffusion SMS refuse les numéros particuliers  
+ parallèlement option pour interdire les numéros étranger.

La définition de numéros particuliers varie de pays en pays, donc la première option implique la deuxième, non ? Plus que ça, il faut interdire des numéros autres que français (pas simplement étrangers). C'est un peu dommage d'avoir une case qui ne marche que sous de telles contraintes.

Thomas Noël a écrit :

On pourrait partir vers des "masques d'autorisation"

Je ne suis pas convaincu de l'intérêt de cette idée, et ça ne permet pas d'avoir quelque chose d'activé par défaut.

La notation XXXXXXXX me paraît être un nid à bug, mais les goûts et les couleurs...

Et tout comme une option magique, ça ne sera jamais parfaitement complet (ou alors ça devient des regex). En fait je crois que je préférerais les regex, sur des chaînes numériques c'est quand même assez lisible, et les 3 expressions copy/paste ready couvrant 99% des besoins seront dans la doc. Et pas de format à inventer et de parser à écrire/maintenir/faire évoluer au gré des nouveaux cas d'usage.

#### #5 - 11 mars 2020 15:06 - Frédéric Péters

La définition de numéros particuliers varie de pays en pays.

Oui mais par fournisseur de SMS ce serait toujours la même, je me disais, peut-être. i.e. si je configure un connecteur Orange Business il considérera les numéros non-français comme étrangers. (en vrai je me dis qu'un abonnement OVH peut varier de pays en pays donc mon commentaire est peut-être à ignorer).

Et tout comme une option magique, ça ne sera jamais parfaitement complet (ou alors ça devient des regex). En fait je crois que je préférerais les regex, sur des chaînes numériques c'est quand même assez lisible, et les 3 expressions copy/paste ready couvrant 99% des besoins seront dans la doc. Et pas de format à inventer et de parser à écrire/maintenir/faire évoluer au gré des nouveaux cas d'usage.

Ok regex dans le code mais côté UI ça m'irait d'avoir un <select> avec les presets, plutôt qu'avoir à trouver d'où les copier/coller.

#### #6 - 30 avril 2020 09:20 - Nicolas Roche

- Assigné à mis à Nicolas Roche

#### #7 - 01 mai 2020 16:28 - Frédéric Péters

spoiler alert : taper ces trucs spécifiques SMS dans passerelle/sms/whatever.py ou passerelle/utills/sms.py, ou autre, mais pas au milieu du reste.

#### #8 - 01 mai 2020 17:07 - Nicolas Roche

- Statut changé de Nouveau à En cours

Ne pas toucher à forms.py ?

```
class GenericConnectorForm(forms.ModelForm):
    def clean_authorization_masks(self):
```

Ni à base/models.py ?

```
class SMSResource(BaseResource):
    ...
    def get_authorization_masks_display(self):
```

ou simplement déplacer le corps des 2 fonctions ?

#### #9 - 01 mai 2020 17:49 - Frédéric Péters

Generic Whatever ne doit pas contenir de trucs parlant de SMS.

SMSResource aurait aussi du se trouver ailleurs mais j'ai raté ce coche au bon moment; il est cependant encore temps de le déplacer.

#### #10 - 01 mai 2020 18:05 - Frédéric Péters

Mais aussi, ce truc de masque, je trouve quand même ça bof et me cite :

Ok regex dans le code mais côté UI ça m'irait d'avoir un <select> avec les presets, plutôt qu'avoir à trouver d'où les copier/coller.

Et à développer sur base de l'UI, j'en serais à :

Pays autorisés

France métrop (+33)    France DOM/TOM (+262, etc.)    Belgique (+32)    Tout

Numéros surtaxés

Ne pas autoriser    Autoriser  
(attention s'applique uniquement pour la France)

Et de là ça donne dans le code juste quelques définitions des bonnes regex,

```
authorized = {
  'french': r'0033[67]\d{9}',
  'french-dom-tom': r'...',
  'belgian': r'00324\d{9}',
}
premium_rate = {
  'french': r'0033[89]\d{9}',
  'french-dom-tom': '?',
  'belgian': '?',
}
```

Et ce sont les numéros passés par clean\_numbers qui arrivent ici, donc on connaît la structure.

### #11 - 01 mai 2020 18:28 - Benjamin Dauvergne

Sinon y a python(3)-phonenumbers dans la dernière version sur toutes les debian (stretch, buster, etc..) et qui donne ces informations gratuitement, une textarea en moins :

```
In [23]: pn = phonenumbers.parse('+33630XXXXX', 'FR')
```

```
In [24]: phonenumbers.number_type(pn) in [getattr(phonenumbers.PhoneNumberType, name) for name in ['FIXED_LINE', 'MOBILE', 'FIXED_LINE_OR_MOBILE']]
Out[24]: True
```

```
In [25]: pn = phonenumbers.parse('0808080808', 'FR')
```

```
In [26]: phonenumbers.number_type(pn) in [getattr(phonenumbers.PhoneNumberType, name) for name in ['FIXED_LINE', 'MOBILE', 'FIXED_LINE_OR_MOBILE']]
Out[26]: False
```

### #12 - 01 mai 2020 18:42 - Frédéric Péters

une textarea en moins

Il n'y a pas de textarea dans ma proposition mais pointer qu'il n'y a pas besoin d'inventer tout un système de masques, ok, c'est ce que j'écrivais déjà, et ça pouvait passer par quelques regex, mais pas de soucis à ce que ça passe par une dépendance externe.

### #13 - 01 mai 2020 19:09 - Benjamin Dauvergne

Frédéric Péters a écrit :

une textarea en moins

Il n'y a pas de textarea dans ma proposition mais pointer qu'il n'y a pas besoin d'inventer tout un système de masques, ok, c'est ce que j'écrivais déjà, et ça pouvait passer par quelques regex, mais pas de soucis à ce que ça passe par une dépendance externe.

Désolé je n'avais pas lu tous les échanges, j'ai réagi dès que j'ai senti une implémentation custom de plus.

### #14 - 04 mai 2020 10:02 - Nicolas Roche

- Lié à [Development #42426](#): Déplacer le code générique des connecteurs SMS ajouté

### #15 - 04 mai 2020 10:05 - Nicolas Roche

- Lié à [Development #42427](#): Déplacer le code clean\_number utilisés par les connecteurs SMS dans SMSResource ajouté

### #16 - 04 mai 2020 10:11 - Nicolas Roche

Vu entre autres : [https://fr.m.wikipedia.org/wiki/Num%C3%A9ro\\_de\\_t%C3%A9l%C3%A9phone#F](https://fr.m.wikipedia.org/wiki/Num%C3%A9ro_de_t%C3%A9l%C3%A9phone#F)

La bibliothèque phonenumbers ne me semble pas requise ici (elle valide juste le code pays et le nombre de chiffres).

France

- +33 pour la France métropolitaine,
- Numéro surtaxé : numéros allant de 0800xxxxxx à 0899xxxxxx

Belgique

- ~~Format international~~ : +32 ZZ CC CC CC ou +32 Z CCC CC CC
- GSM (international) : de +32 45C CC CC CC à +32 49C CC CC CC
- Numéro payant a minimum 1 euro/min : 090C/ ?

Les départements d'outre-mer :

- +262 pour la Réunion, Mayotte et Terres australes et antarctiques françaises (+262 262 CC CC CC)
- +508 pour Saint-Pierre-et-Miquelon (+508 508 CC CC CC)
- +590 pour la Guadeloupe, Saint-Barthélemy et Saint-Martin, (+590 590 CC CC CC)
- +594 pour la Guyane, (+594 594 CC CC CC)
- +596 pour la Martinique. (+596 596 CC CC CC)
- +687 Nouvelle-Calédonie  
(Rien trouvé sur les numéros surtaxés et j'ai un doute sur les numéros de portables)

#### #17 - 07 mai 2020 14:45 - Nicolas Roche

- Lié à Development #42627: sms: ajouter une classe SMSConnectorForm ajouté

#### #18 - 07 mai 2020 15:13 - Nicolas Roche

- Fichier 0002-sms-authorize-numbers-using-masks-39650.patch ajouté
- Fichier 0001-sms-add-number-authorization-masks-to-SMSResource-39.patch ajouté
- Statut changé de En cours à Solution proposée
- Patch proposed changé de Non à Oui

#### #19 - 19 mai 2020 19:58 - Nicolas Roche

- Fichier 0002-sms-authorize-numbers-using-masks-39650.patch ajouté
- Fichier 0001-sms-add-number-authorization-masks-to-SMSResource-39.patch ajouté

A défaut d'avoir réussi à isoler l'ajout de la classe des formulaire sms, je l'ai mergée au premier patch.

#### #20 - 19 mai 2020 20:15 - Frédéric Péters

Mais non, ici ou ailleurs, il n'est pas question d'importer dans passerelle/base/ quoique ce soit de spécifique aux SMS.

#### #21 - 19 mai 2020 20:33 - Frédéric Péters

Mais je pense enfin pouvoir exprimer ton problème :

- tu dois pouvoir définir des widgets particuliers sur certains champs du formulaire de création/édition du connecteur;
- il y a dans GenericConnectorMixin quelque chose qui semble aller vers là;
- mais ça cherche une possibilité de get\_form\_class sur la classe d'application;
- où ça prend un truc qui part de rien;
- alors que ce serait pratique de pouvoir partir de ce qui est créé de base.

Si c'est bien ça, je me dis que ton souhait est dans

<https://git.entrouvert.org/passerelle.git/commit/?id=5e8b06043e893e9dca90fe02d80fee38570229ca> (pas testé du tout)

#### #22 - 19 mai 2020 22:39 - Frédéric Péters

De manière alternative et plus courte,

<https://git.entrouvert.org/passerelle.git/commit/?id=58769c378afc2f056526ab73d393ea12f3b16f72>

~~

Si tout ça correspond bien au problème que tu rencontres, je verrai pour agencer/ajuster ça, peut-être en combinant les deux approches.

#### #23 - 20 mai 2020 09:06 - Nicolas Roche

Oui, ça colle. Merci. (#43128 : j'ai rebasé ma branche dessus)

#### #24 - 29 mai 2020 12:35 - Nicolas Roche

- Fichier 0002-sms-authorize-numbers-using-masks-39650.patch ajouté

- Fichier 0001-sms-add-number-authorization-masks-to-SMSResource-39.patch ajouté

Branche rebasée (mais toujours avec cette adaptation : <https://dev.entrouvert.org/issues/43128#note-2>)

#### #25 - 29 mai 2020 13:23 - Frédéric Péters

Branche rebasée (mais toujours avec cette adaptation : <https://dev.entrouvert.org/issues/43128#note-2>)

Encore ici, jeu de piste, invitation à aller voir un commentaire ailleurs et ticket où je n'arrive pas à comprendre ce que tu essaies de passer comme message.

#### #26 - 29 mai 2020 16:29 - Nicolas Roche

- Fichier 0003-adaptation-de-43128.patch ajouté

Il s'agit de ce troisième patch que je dois ajouter pour utiliser [#43128](#).  
Sans ce patch :

- je dois expliciter `exclude = ('slug', 'users')` dans le Meta de la classe `SMSConnectorForm` (et donc je perd le fait que le slug soit présenté uniquement lors de l'ajout des connecteurs sms).
- je dois ajouter un attribut `id` à la classe `SMSConnector` sans quoi `__init__` se plaint de son absence lors de la configuration du logger du connecteur.

(Je cherche encore l'explication de ces 2 problèmes.)

#### #27 - 29 mai 2020 17:34 - Frédéric Péters

Tu peux dégager ton bricolage ici, valider l'autre ticket, et comme ça il y a une base de travail ?

#### #28 - 29 mai 2020 18:05 - Frédéric Péters

À nouveau traduire la demande, ce pourrait être "c'est bien beau de pouvoir définir une classe de formulaire custom mais j'aimerais bien que celle-ci soit différente pour la création ou l'édition".

Mais j'ai l'impression que ma réponse pourrait être différente aussi, de l'ordre de : il suffit d'ajouter un `get_manager_form_class()` qui se base sur le paramètre `exclude` pour retourner une classe ou l'autre, et là on pourra se dire que c'est quand même pas très naturel, qu'un paramètre explicite serait mieux, et il pourrait être ajouté.

Ou une autre réponse, qui serait pourquoi pas juste si on en est toujours à ce que j'ai cru deviner (commentaire 21), vouloir définir des widgets particuliers, pourquoi pas juste override `get_manager_form_class` et modifier le widget là-derrrière, sur la mode de ce qui est fait pour `ClearableFileInput`.

#### #29 - 29 mai 2020 18:15 - Nicolas Roche

juste override `get_manager_form_class` et modifier le widget là-derrrière

Oui, tu a anticipé ma réponse.

[#43128](#) fait que les classes peuvent définir elles même leur `form_class` via l'ajout d'une méthode `get_manager_form_class`.

```
class GenericConnectorMixin(object):
    exclude_fields = ('slug', 'users')
    ...
    def init_stuff(self, request, *args, **kwargs):
        ...
        self.form_class = self.model.get_manager_form_class(exclude=self.exclude_fields)
```

Ce qui est le cas pour le connecteur `pastell` :

```
class Pastell(BaseResource):
    ...
    @classmethod
    def get_manager_form_class(cls, *args, **kwargs):
        from .forms import PastellForm
        return PastellForm
```

Je cherchais à définir de la même façon une classe formulaire pour les connecteurs sms.  
Mais je souhaite également bénéficier de la factory pour gagner les fonctionnalités de BaseResource (gestion du slug, pas de champ user d'affiché,  
... cf <https://dev.entrouvert.org/issues/42627#note-6>)  
Je peux m'en sortir en redéfinissant la factory dans la méthode get\_manager\_form\_class :

```
class SMSResource(BaseResource):
    @classmethod
    def get_manager_form_class(cls, *args, **kwargs):
        from .forms import SMSConnectorForm

        form_class = modelform_factory(
            cls,
            form=SMSConnectorForm,
            **kwargs)
        for field in form_class.base_fields.values():
            if isinstance(field.widget, ClearableFileInput):
                field.widget.template_with_initial = '\
                    %(initial_text)s: %(initial)s '\
                    %(clear_template)s<br />%(input_text)s: %(input)s'
        return form_class
```

### #30 - 29 mai 2020 18:39 - Nicolas Roche

- Fichier 0002-sms-authorize-numbers-using-masks-39650.patch ajouté
- Fichier 0001-sms-add-number-authorization-masks-to-SMSResource-39.patch ajouté

### #31 - 29 mai 2020 18:39 - Frédéric Péters

Le fait de devoir moi poser les réponses puis avoir un commentaire qui n'oriente pas la discussion reste compliqué.

Il me semble que tu finis par dire que le seul truc dont tu as besoin, c'est pouvoir passer la classe que tu souhaites dans l'appel à modelform\_factory.

Si c'est le cas; est-ce que ça ne peut pas juste être fait en mettant manager\_form\_base\_class = SMSConnectorForm dans la classe, end of story ?

### #32 - 29 mai 2020 18:58 - Nicolas Roche

Non (mais sans encore avoir re-testé), il y a des dépendances circulaires entre le modèle et le formulaire (et sinon c'est exactement ce que j'essayais de faire jusqu'à présent).

### #33 - 29 mai 2020 19:21 - Frédéric Péters

(et sinon c'est exactement ce que j'essayais de faire jusqu'à présent).

C'est la première fois que tu parles de dépendances circulaires. Là-dessus je note en regardant le patch que ce serait parce que "model = SMSResource" dans ton SMSConnectorForm, ça ne me semble pas nécessaire.

### #34 - 31 mai 2020 14:18 - Nicolas Roche

- Fichier 0002-sms-authorize-numbers-using-masks-39650.patch ajouté
- Fichier 0001-sms-add-number-authorization-masks-to-SMSResource-39.patch ajouté

"model = SMSResource" dans ton SMSConnectorForm, ça ne me semble pas nécessaire.

Oui ça fonctionne sans.

manager\_form\_base\_class = SMSConnectorForm dans la classe, end of story ?

Oui, c'est bon tout fonctionne.

### #35 - 20 juillet 2020 10:25 - Nicolas Roche

- Fichier 0002-sms-authorize-numbers-using-masks-39650.patch ajouté
- Fichier 0001-sms-add-number-authorization-masks-to-SMSResource-39.patch ajouté

(rebasé)

#### #36 - 21 septembre 2020 12:06 - Nicolas Roche

- Fichier 0002-sms-authorize-numbers-using-masks-39650.patch ajouté
- Fichier 0001-sms-add-number-authorization-masks-to-SMSResource-39.patch ajouté

(rebasés)

#### #37 - 16 novembre 2020 10:11 - Nicolas Roche

- Fichier 0001-sms-add-number-authorization-masks-to-SMSResource-39.patch ajouté
- Fichier 0002-sms-authorize-numbers-using-masks-39650.patch ajouté

(rebasé)

#### #38 - 01 février 2021 17:06 - Nicolas Roche

- Fichier 0002-sms-authorize-numbers-using-masks-39650.patch ajouté
- Fichier 0001-sms-add-number-authorization-masks-to-SMSResource-39.patch ajouté

(re-basés)

#### #39 - 01 février 2021 22:11 - Benjamin Dauvergne

- Statut changé de Solution proposée à Solution validée

Pour commenter la description du ticket, je pense que la validation client sera de toute façon nécessaire, parce que par exemple on serait amené à exporter la liste des numéros pour une raison x ou y ou alimenter un autre système; néanmoins ça m'a l'air ok.

Au niveau de l'implémentation on aurait pu simplement utiliser un ArrayField(CharField(choices=...)) qui se serait occuper de tout (à part le widget), et premium\_rate aurait pu s'appeler allow\_premium\_rate est n'être qu'un booléen.

Je ne ferai qu'un commit, séparer l'introduction des champs et leur usage ça rend les commits plus compliqués à lire je trouve.

--

Dans authorized\_numbers :

- les regex devraient contenir le marqueur de début et fin de chaîne, sinon on va avoir du n'importe quoi.
- le code serait plus simple à lire si on construisait d'abord la regex puis on l'applique à la liste pour la filtrer (construire une regexp en concaténant les parties avec '|')
- extraire le mapping enum -> liste de regex du code

--

```
if unauthorized_numbers != []:
    raise APIError('phone numbers not authorized: %s' % ', '.join(unauthorized_numbers))
return True
```

Le return True semble inutile vu qu'on ne l'utilise et qu'on lève déjà une exception, mais plutôt que cela, je filtrerai la liste, si on envoie un SMS à un paquet de numéros on a pas envie, je pense, que tout l'envoi soit bloqué sur un seul numéro invalide; on ne lèvera une erreur que si la liste devient vide et sinon un warning pour les numéros ignorés + un retour avec la liste des numéros invalides.

#### #40 - 01 février 2021 22:12 - Benjamin Dauvergne

- Statut changé de Solution validée à En cours

#### #41 - 02 février 2021 14:08 - Nicolas Roche

- Fichier 0001-sms-filter-authorized-numbers-39650.patch ajouté
- Statut changé de En cours à Solution proposée

utiliser un ArrayField(CharField(choices=...)) et premium\_rate aurait pu s'appeler allow\_premium\_rate est n'être qu'un booléen.

fait

les regex devraient contenir le marqueur de début et fin de chaîne

fait

le code serait plus simple à lire si on construisait d'abord la regex puis on l'applique à la liste pour la filtrer

fait (du mieux que j'ai pu)

extraire le mapping enum -> liste de regex du code

(je n'ai pas compris)

je filtrerai la liste

fait

un retour avec la liste des numéros invalides.

bof, comme les SMS sont asynchrones j'envoie juste un warning

J'avais un doute sur les traductions (vu le texte inscrit dans les migrations), mais j'ai testé et ça fonctionne.

#### #42 - 02 février 2021 19:36 - Nicolas Roche

- Statut changé de Solution proposée à En cours

un retour avec la liste des numéros invalides.

(pas fait)

#### #43 - 02 février 2021 21:11 - Nicolas Roche

- Fichier 0002-sms-return-filtered-numbers-to-caller-39650.patch ajouté

- Statut changé de En cours à Solution proposée

Fait, dans un commit à part, pour faciliter la relecture.

#### #44 - 03 février 2021 18:11 - Benjamin Dauvergne

Nicolas Roche a écrit :

extraire le mapping enum -> liste de regex du code

(je n'ai pas compris)

Je voulais dire séparer les données du code, pour que la fonction ait moins de lignes, par exemple avec une globale :

```
NUMBER_REGEXS = {  
  FRANCE: ['$06\d{8}^', ...],  
  DOMTOM: ...  
}
```

#### #45 - 04 février 2021 09:52 - Nicolas Roche

- Fichier 0001-sms-filter-authorized-numbers-39650.patch ajouté

Fait, et j'ai tout remis dans un seul patch.

#### #46 - 26 février 2021 17:06 - Benjamin Dauvergne

- Statut changé de Solution proposée à Solution validée



## #47 - 01 mars 2021 10:21 - Nicolas Roche

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

```
commit 93071920dale0414f7c6ac4c21a8c35be4b401db
Author: Nicolas ROCHE <nroche@entrouvert.com>
Date: Tue Feb 2 12:20:55 2021 +0100
```

```
sms: filter authorized numbers (#39650)
```

## #48 - 02 mars 2021 18:17 - Frédéric Péters

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

### Fichiers

0002-sms-authorize-numbers-using-masks-39650.patch	9,09 ko	07 mai 2020	Nicolas Roche
0001-sms-add-number-authorization-masks-to-SMSResource-39.patch	15,3 ko	07 mai 2020	Nicolas Roche
0002-sms-authorize-numbers-using-masks-39650.patch	9,49 ko	19 mai 2020	Nicolas Roche
0001-sms-add-number-authorization-masks-to-SMSResource-39.patch	21,2 ko	19 mai 2020	Nicolas Roche
0002-sms-authorize-numbers-using-masks-39650.patch	9,51 ko	29 mai 2020	Nicolas Roche
0001-sms-add-number-authorization-masks-to-SMSResource-39.patch	17,7 ko	29 mai 2020	Nicolas Roche
0003-adaptation-de-43128.patch	2,24 ko	29 mai 2020	Nicolas Roche
0002-sms-authorize-numbers-using-masks-39650.patch	9,51 ko	29 mai 2020	Nicolas Roche
0001-sms-add-number-authorization-masks-to-SMSResource-39.patch	17,5 ko	29 mai 2020	Nicolas Roche
0002-sms-authorize-numbers-using-masks-39650.patch	9,51 ko	31 mai 2020	Nicolas Roche
0001-sms-add-number-authorization-masks-to-SMSResource-39.patch	17 ko	31 mai 2020	Nicolas Roche
0002-sms-authorize-numbers-using-masks-39650.patch	9,51 ko	20 juillet 2020	Nicolas Roche
0001-sms-add-number-authorization-masks-to-SMSResource-39.patch	18,5 ko	20 juillet 2020	Nicolas Roche
0002-sms-authorize-numbers-using-masks-39650.patch	8,96 ko	21 septembre 2020	Nicolas Roche
0001-sms-add-number-authorization-masks-to-SMSResource-39.patch	18,9 ko	21 septembre 2020	Nicolas Roche
0001-sms-add-number-authorization-masks-to-SMSResource-39.patch	19,4 ko	16 novembre 2020	Nicolas Roche
0002-sms-authorize-numbers-using-masks-39650.patch	9,12 ko	16 novembre 2020	Nicolas Roche
0002-sms-authorize-numbers-using-masks-39650.patch	8,7 ko	01 février 2021	Nicolas Roche
0001-sms-add-number-authorization-masks-to-SMSResource-39.patch	19,3 ko	01 février 2021	Nicolas Roche
0001-sms-filter-authorized-numbers-39650.patch	26,1 ko	02 février 2021	Nicolas Roche
0002-sms-return-filtered-numbers-to-caller-39650.patch	8,78 ko	02 février 2021	Nicolas Roche
0001-sms-filter-authorized-numbers-39650.patch	27,4 ko	04 février 2021	Nicolas Roche