RFC 4745 : Common Policy: A Document Format for Expressing Privacy Preferences

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 20 décembre 2007

Date de publication du RFC : Février 2007

https://www.bortzmeyer.org/4745.html

Les protocoles modernes comme SIP (qui sert surtout pour la téléphonie sur Internet mais aussi pour la messagerie instantanée) permettent d'obtenir bien des informations sensibles comme la localisation exacte de l'interlocuteur. Celui-ci n'a pas forcément envie de tenir le reste du monde au courant de ses activités. Notre RFC s'attaque donc à ce problème en décrivant un format de document exprimant des préférences en matière de protection de la vie privée.

Par exemple, une extension de SIP (RFC 3856 [1], voir aussi le RFC 2778) permet d'être informé de la présence ou de l'absence d'une personne donnée. Ce n'est pas une information qu'on donne forcément à tout le monde ! D'une manière générale, la diminution considérable de la vie privée, consécutive au rôle grandissant de l'informatique, inquiète, à juste titre, de plus en plus de gens.

S'il est nécessaire que certains usages soient tout simplement interdits, comme le fait, par exemple, la loi française informatique & Libertés, pour d'autres, on peut envisager de négocier et d'arbitrer entre l'intimité et le caractère pratique de la diffusion d'informations. À condition qu'on puisse choisir et que ces choix soient respectés. L'IETF ne peut pas garantir ce respect mais elle peut au moins définir un format pour que les choix soient transmis de manière non ambigüe et c'est ce que fait ce RFC.

Il est l'héritier d'autres formats dont le plus connu est le P3P du W3C, qui permet aux serveurs Web de signaler leurs pratiques en matière de respect de la vie privée. P3P n'a eu aucun succès, sans doute en partie car les sites Web qui ont l'habitude de déclarer leurs politiques en vingt pages d'anglais légal

1. Pour voir le RFC de numéro NNN, https://www.ietf.org/rfc/rfcNNN.txt, par exemple https://www.ietf.org/rfc/rfc3856.txt

incompréhensible ne voulaient pas les traduire dans un langage simple et non ambigü. Le W3C continue son travail sur ces « langages de politique » via l'activité Pling <http://www.w3.org/Policy/pling/>.

Notre RFC définit donc un format assez général pour couvrir d'autres besoins que ceux de SIP. La définition du format commence avec la section 4, qui détaille les principes de base. Notamment, ce format ne permet d'exprimer que des **permissions**, pas des **interdictions**. Tout est interdit par défaut et l'auteur du document doit lister les permissions. En effet, comme le sait toute personne qui a rédigé des ACL pour un routeur, on se trompe facilement, dans le sens d'une trop grande sécurité ou au contraire dans celui d'un laxisme involontaire. (La même section explique qu'il vaut mieux forcer l'utilisateur à exprimer clairement ses autorisations plutôt que de le laisser croire à tort qu'il est protégé.) Si une règle autorise `192.168.1.0/24` à faire une certaine action et qu'une autre interdit cette action à `192.168.1.0/26`, il n'est pas immédiatement évident de savoir si `192.168.1.62` est autorisé...

La solution radicale adoptée par notre RFC rend l'écriture des règles plus simple, et notamment indépendante de l'ordre dans lequel elles sont exprimées (un piège classique avec les ACL). Pour les mêmes raisons de simplicité, la section 5 explique pourquoi des fonctions intéressantes comme les expressions rationnelles ont été omises.

La section 7 est le gros morceau du RFC. Elle détaille les **conditions** qui décident si une règle s'applique ou pas. Pour l'instant (mais des extensions ultérieures pourraient changer cela), il y a trois conditions possibles. La première est l'**identité**, décrite en section 7.1. Elle fonctionne en comparant une identité stockée dans la règle avec celle obtenue de l'autre utilisateur (l'authentification de cette seconde identité n'est pas couverte dans le RFC). Ainsi, on pourra dire « si l'appelant est +33 1 38 30 83 46, alors ... ». Voici un exemple inspiré du RFC (l'élément `<identity>` correspond si **au moins** un des éléments `<one>` correspond) :

```
<identity>
     <one id="sip:alice@example.org"/>
     <one id="tel:+1-212-555-1234" />
     <one id="mailto:bob@example.net" />
 </identity>
```

Dans un autre exemple, on voit la possibilité d'autoriser tout le monde **sauf** quelques uns :

```
 <identity>
     <many>
         <except domain="example.com"/>
         <except id="sip:alice@bad.example.net"/>
         <except id="sip:bob@good.example.net"/>
         <except id="tel:+1-212-555-1234" />
         <except id="sip:alice@example.com"/>
     </many>
 </identity>
```

Une autre condition possible est la **sphère** (section 7.3). Une sphère n'est pas une identité mais une situation dans laquelle on se trouve (par exemple « au travail », « en train de manger » ou « endormi »). Le RFC n'essaie pas de définir une ontologie de ces situations, chacun est libre du vocabulaire utilisé. Enfin, la troisième espère de condition, la **validité** permet d'ajouter des contraintes temporelles. Voici un exemple (lorsqu'il y a plusieurs conditions, elles doivent toutes être vraies pour que le total soit vrai) :

```
<conditions>
   <sphere value="work"/>
   <identity>
          <one id="sip:boss@example.com"/>
   </identity>
   <validity>
                  <from>1900-01-01T08:00:00+01:00</from>
                  <until>2010-12-31T18:00:00+01:00</until>
   </validity>
</condition>
```

Une fois qu'on dispose de conditions, on peut exprimer des **actions**, qui font l'objet de la section 8 et des **transformations** (section 9). Notre RFC ne définit aucune action, cela sera laissé à des RFC ultérieurs.

Enfin, la section 13 est consacrée au schéma formel du langage, écrit en W3C Schema. Les actions et les transformations, non décrites dans ce RFC, sont marquées comme extensibles en les dérivant de `xsd:anyType`.