

Groupe de travail Réseau
Request for Comments : 3876
Catégorie : En cours de normalisation
Traduction Claude Brière de L'Isle

D. Chadwick, University of Salford
S. Mullan, Sun Microsystems
septembre 2004

Retour des valeurs de correspondance avec le protocole léger d'accès à un répertoire, version 3 (LDAPv3)

Statut de ce mémoire

Le présent document spécifie un protocole Internet en cours de normalisation pour la communauté de l'Internet, et appelle à des discussions et des suggestions pour son amélioration. Prière de se reporter à l'édition actuelle du STD 1 "Normes des protocoles officiels de l'Internet" pour connaître l'état de normalisation et le statut de ce protocole. La distribution du présent mémoire n'est soumise à aucune restriction.

Notice de copyright

Copyright (C) The Internet Society (2004).

Résumé

Le présent document décrit une commande pour la version 3 du protocole léger d'accès à un répertoire (LDAPv3, *Lightweight Directory Access Protocol version 3*) qui est utilisée pour retourner un sous ensemble des valeurs d'attribut d'une entrée. Précisément, seules les valeurs qui correspondent à un filtre "retour des valeurs". Sans la prise en charge de cette commande, un client doit restituer toutes les valeurs d'un attribut et chercher en local les valeurs spécifiques.

1. Introduction

Lorsque on lit un attribut à partir d'une entrée en utilisant le protocole léger d'accès à un répertoire version 3 (LDAPv3) [RFC2251], il est normalement seulement possible de lire soit le type d'attribut, soit le type d'attribut et toutes ses valeurs. Il n'est pas possible de lire de façon sélective juste quelques valeurs de l'attribut. Si un attribut contient de nombreuses valeurs, par exemple, l'attribut `userCertificate`, ou les sous schémas d'attributs opérationnels de publication `objectClasses` et `attributeTypes` [RFC2252], il peut alors être souhaitable que l'utilisateur soit capable de restituer de façon choisie un sous ensemble des valeurs, en particulier, celles des attributs qui correspondent à des critères de choix définis par l'utilisateur. Sans la commande spécifiée dans le présent document, un client doit lire toutes les valeurs de l'attribut et filtrer les valeurs non désirées, ce qui nécessite que le client mette en œuvre les règles de correspondance. Cela exige aussi que le client lise et éventuellement traite de nombreuses valeurs non pertinentes, ce qui peut être inefficace si les valeurs sont grandes ou complexes, ou si de nombreuses valeurs sont mémorisées par attribut.

Le présent document spécifie une commande LDAPv3 pour permettre à un usager de ne retourner que les valeurs qui correspondaient à (c'est-à-dire, qui ont retourné VRAI) à un ou plusieurs éléments d'un filtre nouvellement défini "valeurs retournées". Cette commande peut être particulièrement utile lorsque utilisée en conjonction avec les règles de correspondance extensibles qui correspondent pour un ou plusieurs composants de valeurs d'attribut binaire complexes.

Les mots clés "DOIT", "NE DOIT PAS", "EXIGE", "DEVRA", "NE DEVRA PAS", "DEVRAIT", "NE DEVRAIT PAS", "RECOMMANDE", "PEUT", et "FACULTATIF" en majuscules dans ce document sont à interpréter comme décrit dans le BCP 14, [RFC2119].

2. Commande `valuesReturnFilter`

La commande `valuesReturnFilter` (*filtre des valeurs retournées*) est critique ou non critique selon ce qui est déterminé par l'usager. Elle n'a de signification que pour l'opération `Search` (*recherche*), et DEVRAIT être ajoutée seulement à l'opération `Search` par le client. Si le serveur prend en charge la commande et si elle est présente sur une opération `Search`, le serveur DOIT obéir à la commande, sans considération de la valeur du fanion de criticité.

Si la commande est marquée critique, et si, soit le serveur ne prend pas en charge la commande, soit la commande est appliquée à une opération autre que `Search`, le serveur DOIT retourner une erreur `unavailableCriticalExtension` (*extension critique non disponible*). Si la commande n'est pas marquée comme critique, et si soit le serveur ne prend pas en charge la commande, soit la commande est appliquée à une opération autre que `Search`, alors le serveur DOIT ignorer la commande.

L'identifiant d'objet de cette comande est 1.2.826.0.1.3344810.2.3.

La valeur de la commande est une CHAINE D'OCTETS, dont la valeur est le codage en BER [X.690], conformément au paragraphe 5.1 de la [RFC2251], d'une valeur du type ASN.1 [X.680] ValuesReturnFilter (*filtre de retour de valeur*).

```

ValuesReturnFilter ::= SEQUENCE DE SimpleFilterItem
SimpleFilterItem ::= CHOIX {
equalityMatch [RFC2252] AttributeValueAssertion,           ; (assertion d'une valeur d'attribut)
substrings [RFC2119] SubstringFilter,                       ; (filtre de sous chaîne)
greaterOrEqual [X.680] AttributeValueAssertion,
lessOrEqual [X.690] AttributeValueAssertion,
present [RFC3383] AttributeDescription,                     ; (description d'attribut)
approxMatch [X.511] AttributeValueAssertion,
extensibleMatch [ISO 9594] SimpleMatchingAssertion }       ; (assertion correspondante simple)
SimpleMatchingAssertion ::= SEQUENCE {
matchingRule [RFC2026] MatchingRuleId FACULTATIF,         ; (identifiant de règle de correspondance)
type [RFC2251] AttributeDescription FACULTATIF,
                                                                    ; au moins une des valeurs ci-dessus doit être présente
matchValue [RFC2252] AssertionValue}

```

Tous les types de données ci-dessus ont la signification standard définie dans la [RFC2251].

Si le serveur prend en charge cette commande, le serveur DOIT utiliser la commande comme suit :

- 1) Le filtre Search est d'abord exécuté afin de déterminer quelles entrées satisfont le critère de recherche (ce sont les entrées filtrées). La commande n'a pas d'impact sur cette étape.
- 2) Si le paramètre typesOnly de la demande de recherche est VRAI, la commande n'a pas d'effet et la demande de recherche est traitée comme si la commande n'avait pas été spécifiée.
- 3) Si les paramètre d'attributs de la demande de recherche consistent en une liste contenant seulement l'attribut avec l'OID "1.1" (qui spécifie qu'aucun attribut n'a été retourné), la commande n'a pas d'effet et la demande de recherche est traitée comme si la commande n'avait pas été spécifiée.
- 4) Pour chaque attribut de la liste de paramètres d'attributs de la demande de recherche, le serveur DOIT appliquer la commande comme suit à chaque entrée dans l'ensemble des entrées filtrées :
 - i) Chaque valeur d'attribut qui est évaluée à VRAI contre un ou plusieurs éléments du ValuesReturnFilter est placée dans la SearchResultEntry (*entrée de résultat de recherche*) correspondante.
 - ii) Une valeur d'attribut qui est évaluée à FAUX ou à indéfini contre tous les éléments du ValuesReturnFilter n'est pas placée dans la SearchResultEntry correspondante. Un attribut qui n'a pas de valeurs choisies est retourné avec un ensemble de valeurs vide.

Note . Si la AttributeDescriptionList (voir la [RFC2251]) est vide ou comporte "*", alors la commande DOIT être appliquée à tous les attributs d'utilisateur. Si la AttributeDescriptionList contient un "+", la commande DOIT alors être appliquée à tous les attributs de fonctionnement.

3. Relations avec X.500

La commande est un sur ensemble de l'argument de recherche booléen seulementValeursCorrespondantes (MVO, matchedValuesOnly) du protocole d'accès de répertoire X.500 (DAP, *Directory Access Protocol*) [X.511], tel qu'amendé par la dernière version [ISO 9594]. Un examen attentif du booléen matchedValuesOnly par le groupe de travail Extension de LDAP (LDAPEXT) a révélé des ambiguïtés et des complexités dans le booléen MVO qui pourraient n'être pas résolues facilement. Par exemple, il n'est pas clair si le booléen MVO gouverne seulement les valeurs d'attribut qui ont contribué à la vérité globale du filtre, ou toutes les valeur d'attribut, même si l'élément de filtre qui contenait l'attribut a été évalué à faux. Pour cette raison, le groupe LDAPEXT a décidé de remplacer le booléen MVO par un simple filtre qui supprime toute incertitude sur le fait qu'une valeur d'attribut est choisie ou non.

4. Relations avec les autres commandes LDAP

L'objet de cette commande est de choisir zéro, une, ou plusieurs valeurs d'attribut à partir de chacun des attributs demandés dans une entrée de filtre, et d'éliminer le reste. Une fois que les valeurs d'attribut ont été éliminées par cette commande, elles NE DOIVENT PAS être réinstallées dans le résultat de recherche par d'autres commandes.

Cette commande agit indépendamment des autres commandes LDAP comme de tri côté serveur [RFC2891] et les entrées dupliquées [10]. Cependant, il peut y avoir des interactions entre cette commande et d'autres, de sorte qu'un ensemble différent d'entrées de résultat de recherche est retourné, ou que les entrées sont retournées dans un ordre différent, selon la séquence de cette commande et des autres commandes de la demande LDAP. Par exemple, avec le tri côté serveur, si le tri est fait d'abord, et le filtrage de retour de valeur en second, l'ensemble des résultats de recherche peut apparaître dans le mauvais ordre car le filtrage de valeurs peut retirer les valeurs d'attribut sur lesquelles le rangement a été fait. (Le document sur le tri spécifie que les entrées sans aucune valeur d'attribut de clé de tri devraient être traitées comme venant après toutes les autres valeurs d'attribut.) De même avec les entrées dupliquées, si la duplication est effectuée avant le filtrage de valeurs, l'ensemble des entrées de résultat de recherche peut contenir des entrées dupliquées identiques, dont chacune aura un ensemble vide de valeurs d'attribut, parce que le filtrage des valeurs aura supprimé les valeurs d'attribut qui ont été utilisées pour dupliquer les résultats.

Pour ces raisons, la commande ValuesReturnFilter dans une SearchRequest DEVRAIT précéder les autres commandes qui affectent le nombre et l'ordre des SearchResultEntry.

5. Exemples

Toutes les entrées sont données dans le format d'échange de données LDAP (LDIF) [RFC2849].

La représentation de chaîne de valuesReturnFilter dans les exemples ci-dessous utilise la notation ABNF [RFC2234] suivante :

```
valuesReturnFilter = "(" 1*simpleFilterItem ")"           ; (élément de filtre simple)
simpleFilterItem = "(" item ")"
```

où item est comme défini ci-dessous (adapté de la [RFC2254]).

```
item      = simple / présent / sous-chaîne / extensible
simple     = valeur d'attribut filtertype
filtertype = égal / approx / supérieur / inférieur
égal      = "="
approx    = "~="
supérieur = ">="
inférieur = "<="
extensible = attr [":" matchingrule] ":@" value / ":" matchingrule ":@" value
présent   = attr "=*"
sous-chaîne = attr "=" [initial] tout [final]
initial   = valeur
tout      = "*" *(valeur "*")
final     = valeur
attr      = AttributeDescription du paragraphe 4.1.5 de la [RFC2026]
matchingrule = MatchingRuleId du paragraphe 4.1.9 de la [RFC2026]
valeur    = AttributeValue du paragraphe 4.1.6 de la [RFC2026]
```

- 1) Le premier exemple montre comment la commande peut être réglée pour retourner toutes les valeurs d'attribut provenant d'un type d'attribut (par exemple, telephoneNumber) et un sous ensemble de valeurs provenant d'un autre type d'attribut (par exemple, mail).

Les entrées ci-dessous représentent des classes d'objet organizationalPerson situées quelque part en dessous du nom distinctif dc=ac,dc=uk.

```
dn: cn=Sean Mullan,ou=people,dc=sun,dc=ac,dc=uk
cn: Sean Mullan
sn: Mullan
objectClass: organizationalPerson
objectClass: person
objectClass: inetOrgPerson
```

```

mail: sean.mullan@hotmail.com
mail: mullan@east.sun.com
telephoneNumber: + 781 442 0926
telephoneNumber: 555-9999
dn: cn=David Chadwick,ou=isi,o=salford,dc=ac,dc=uk
cn: David Chadwick
sn: Chadwick
objectClass: organizationalPerson
objectClass: person
objectClass: inetOrgPerson
mail: d.w.chadwick@salford.ac.uk

```

Une opération de recherche LDAP est spécifiée avec un baseObject réglé au DN de la base de recherche (c'est-à-dire, dc=ac,dc=uk), une portée de sous arborescence, un filtre réglé à (sn=mullan), et la liste des attributs à retourner réglée à "mail,telephoneNumber" ou "*". De plus, une commande ValuesReturnFilter est réglée à ((mail=*hotmail.com)(telephoneNumber=*)).

Les résultats de recherche retournés par le serveur vont consister en l'entrée suivante :

```

dn: cn=Sean Mullan,ou=people,dc=sun,dc=ac,dc=uk
mail: sean.mullan@hotmail.com
telephoneNumber: + 781 442 0926
telephoneNumber: 555-9999

```

Noter que la commande n'a pas d'effet sur les valeurs retournées pour l'attribut "telephoneNumber" (toutes les valeurs sont retournées) car la commande spécifiait que toutes les valeurs devraient être retournées.

- 2) Le second exemple montre comment on peut restituer une seule définition de sous schéma de type d'attribut pour l'attribut "gunk" avec l'OID 1.2.3.4.5 provenant de la sous entrée de sous schéma.

Supposons que la sous entrée de sous schéma soit détenue en dessous de l'entrée de racine avec le DN cn=subschema subentry,o=myorg et cela contient un attribut de fonctionnement attributeTypes qui détient la description des 35 attributs connus de ce serveur (chaque description est tenue comme une seule valeur d'attribut de l'attribut attributeTypes).

```

dn: cn=subschema subentry,o=myorg
cn: subschema subentry
objectClass: subschema
attributeTypes: ( 2.5.4.3 NAME 'cn' SUP name )
attributeTypes: ( 2.5.4.6 NAME 'c' SUP name SINGLE-VALUE )
attributeTypes: ( 2.5.4.0 NAME 'objectClass' EQUALITY objectIdentifierMatch SYNTAX 1.3.6.1.4.1.1466.115.121.1.38 )
attributeTypes: ( 2.5.18.2 NAME 'modifyTimestamp' EQUALITY generalizedTimeMatch ORDERING generalizedTimeOrderingMatch SYNTAX 1.3.6.1.4.1.1466.115.121.1.24 SINGLE-VALUE NO-USER-MODIFICATION USAGE directoryOperation )
attributeTypes: ( 2.5.21.6 NAME 'objectClasses' EQUALITY objectIdentifierFirstComponentMatch SYNTAX 1.3.6.1.4.1.1466.115.121.1.37 USAGE directoryOperation )
attributeTypes: ( 1.2.3.4.5 NAME 'gunk' EQUALITY caseIgnoreMatch SUBSTR caseIgnoreSubstringsMatch SYNTAX 1.3.6.1.4.1.1466.115.121.1.44{64} )
attributeTypes: ( 2.5.21.5 NAME 'attributeTypes' EQUALITY objectIdentifierFirstComponentMatch SYNTAX 1.3.6.1.4.1.1466.115.121.1.3 USAGE directoryOperation )

```

plus 28 autres – on peut imaginer le reste

L'utilisateur crée une opération de recherche LDAP avec un baseObject réglé à cn=subschema subentry,o=myorg, une portée de base, un filtre réglé à (objectClass=subschema), la liste des attributs à retourner réglée à "attributeTypes", et le ValuesReturnFilter réglé à ((attributeTypes=1.2.3.4.5))

Le résultat de recherche retourné par le serveur consisterait en l'entrée suivante :

```
dn: cn=subschema subentry,o=myorg
attributeTypes: ( 1.2.3.4.5 NAME 'gunk' EQUALITY caseIgnoreMatch SUBSTR caseIgnoreSubstringsMatch SYNTAX 1.3.6.1.4.1.1466.115.121.1.44{64} )
```

- 3) Le dernier exemple montre comment la commande peut être utilisée pour se confronter à une valeur d'attribut `userCertificate`. Noter que cet exemple exige que le serveur LDAP prenne en charge la règle de correspondance `certificateExactMatch` définie dans [12] comme règle de correspondance `EQUALITY` pour `userCertificate` (*certificat d'utilisateur*).

L'entrée ci-dessous représente une classe d'objet `pkiUser` (*utilisateur de PKI*) mémorisée dans le répertoire.

```
dn: cn=David Chadwick,ou=people,o=University of Salford,c=gb
cn: David Chadwick
objectClass: person
objectClass: organizationalPerson
objectClass: pkiUser
objectClass: inetOrgPerson
sn: Chadwick
mail: d.w.chadwick@salford.ac.uk
userCertificate;binary: {représentation binaire d'un certificate avec
un numéro de série de 2468 produit par o=truetrust ltd,c=gb}
userCertificate;binary: {représentation binaire d'un certificate avec
un numéro de série de 1357 produit par o=truetrust ltd,c=gb}
userCertificate;binary: {représentation binaire d'un certificate avec
un numéro de série de 1234 produit par dc=certsRus,dc=com}
```

Une opération de recherche LDAP est spécifiée avec un `baseObject` réglé à `o=University of Salford,c=gb`, une portée de sous arborescence, un filtre réglé à `(sn=chadwick)`, et la liste des attributs à retourner réglée à `"userCertificate;binary"`. De plus, une commande `ValuesReturnFilter` est réglée à `((userCertificate=1357$o=truetrust ltd,c=gb))`.

Le résultat de recherche retourné par le serveur consisterait en l'entrée suivante :

```
dn: cn=David Chadwick,ou=people,o=University of Salford,c=gb
userCertificate;binary: {représentation binaire d'un certificate avec un numéro de série de 1357 produit par o=truetrust ltd,c=gb}
```

6. Considérations sur la sécurité

Le principal objet du présent document n'est pas les questions de sécurité. On notera cependant que les valeurs d'attribut `DOIVENT` n'être retournées que si le contrôle d'accès appliqué par le serveur LDAP leur permet d'être retournées, et à cet égard, l'effet de la commande `ValuesReturnFilter` n'a aucune conséquence.

Noter que la commande `ValuesReturnFilter` peut avoir un effet positif sur le déploiement des infrastructures de clé publique. Certaines opérations de PKI, comme les recherches pour des certificats spécifiques, deviennent plus adaptables, et plus pratiques lorsque combinées avec les règles de correspondance de certificat X.509 au serveur, car la commande évite le téléchargement d'un nombre éventuellement grand de certificats non pertinents qui auraient été traités et filtrés en local (ce qui dans certains cas est très difficile à effectuer).

7. Considérations relatives à l'IANA

La commande `Matched Values` a été enregistrée comme mécanisme du protocole LDAP [RFC3383] comme suit :

```
Sujet : Demande d'enregistrement de mécanisme du protocole LDAP
Identifiant d'objet : 1.2.826.0.1.3344810.2.3
Description : commande Valeurs correspondantes
Adresse personnelle & de messagerie à contacter pour plus d'informations : David Chadwick <d.w.chadwick@salford.ac.uk>
Usage : Commande
Spécification : RFC3876
Auteur/contrôleur des changements : IESG
Commentaires : aucun
```

Le présent document utilise l'OID 1.2.826.0.1.3344810.2.3 pour identifier la commande `matchedValues` décrite ici. Cet OID a été alloué par TrueTrust Ltd, sous son numéro d'entreprise de société enregistrée anglo/galloise alloué par BSI [BS7453].

8. Remerciements

Les auteurs tiennent à remercier les membres de la liste de diffusion LDAPExt de leurs commentaires constructifs sur les versions précédentes de ce document, et en particulier Harald Alvestrand qui a le premier suggéré d'avoir un filtre de retour d'attribut, et Bruce Greenblatt qui a le premier proposé la syntaxe de cette commande.

9. Références

9.1 Références normatives

- [RFC2026] S. Bradner, "Le processus de [normalisation de l'Internet](#) -- Révision 3", (BCP0009) octobre 1996. (Remplace [RFC1602](#), [RFC1871](#)) (MàJ par [RFC3667](#), [RFC3668](#), [RFC3932](#), [RFC3979](#), [RFC3978](#), [RFC5378](#), [RFC6410](#))
- [RFC2119] S. Bradner, "[Mots clés à utiliser](#) dans les RFC pour indiquer les niveaux d'exigence", BCP 14, mars 1997.
- [RFC2251] M. Wahl, T. Howes et S. Kille, "[Protocole léger d'accès à un répertoire](#) (v3)", décembre 1997.
- [RFC2252] M. Wahl, A. Coulbeck, T. Howes, S. Kille, "[Protocole léger d'accès à un répertoire](#) (v3) : Définitions de syntaxe d'attribut", décembre 1997. (Obsolète, voir [RFC4510](#), [RFC4517](#), [RFC4523](#), [RFC4512](#)) (MàJ par [RFC3377](#)) (P.S.)
- [RFC3383] K. Zeilenga, "Autorité d'allocation des numéros de l'Internet (IANA) : Considérations sur le protocole léger d'accès à un répertoire (LDAP)", septembre 2002. (Obsolète, voir [RFC4520](#))
- [X.680] Recommendation UIT-T X.680 (1997) | ISO/CEI 8824-1:1998, "Technologies de l'information – Notation de syntaxe abstraite numéro un (ASN.1) : Spécification de la notation de base".
- [X.690] Recommendation UIT-T X.690 (1997) | ISO/CEI 8825-1,2,3:1998, "Technologies de l'information – Règles de codage ASN.1 : Spécification des règles de codage de base (BER), règles de codage canonique (CER) et règles de codage distinctives (DER)".

9.2 Références pour information

- [10] Sermersheim, J., "LDAP Control for a Duplicate Entry Representation of Search Results", Travail en cours, octobre 2000.
- [12] Chadwick, D. and S.Legg, "Internet X.509 Public Key Infrastructure - Additional LDAP Schema for PKIs", Travail en cours, juin 2002
- [BS7453] BRITISH STANDARD BS 7453 Part 1. Procedures for UK Registration for Open System Standards Part 1: Procedures for the UK Name Registration Authority.
- [ISO 9594] ISO/IEC 9594 / UIT-T Rec X.511, "L'annuaire : Définition de service abstraite" (2001).
- [RFC2234] D. Crocker et P. Overell, "BNF augmenté pour les spécifications de syntaxe : ABNF", novembre 1997. (Obsolète, voir [RFC5234](#))
- [RFC2254] T. Howes, "Représentation comme chaîne des [filtres de recherche LDAP](#)", décembre 1997. (Obsolète, voir [RFC4510](#), [RFC4515](#)) (P.S.)
- [RFC2849] G. Good, "[Format d'échange de données LDAP](#) (LDIF) - Spécification technique", juin 2000. (P.S.)
- [RFC2891] T. Howes, M. Wahl, A. Anantha, "[Extension de commande LDAP](#) pour le tri côté serveur de résultats de recherche", août 2000. (P.S.)

[X.511] Recommendation UIT-T X.511, "L'annuaire : Définition de service abstraite", 1993.

10. Adresse des auteurs

David Chadwick
IS Institute
University of Salford
Salford M5 4WT
UK
téléphone : +44 161 295 5351
mél : d.w.chadwick@salford.ac.uk

Sean Mullan
Sun Microsystems
One Network Drive
Burlington, MA 01803
USA
mél : sean.mullan@sun.com

11. Déclaration complète de droits de reproduction

Copyright (C) The Internet Society (2004).

Le présent document est soumis aux droits, licences et restrictions contenus dans le BCP 78, et à www.rfc-editor.org, et sauf pour ce qui est mentionné ci-après, les auteurs conservent tous leurs droits.

Le présent document et les informations contenues sont fournis sur une base "EN L'ÉTAT" et le contributeur, l'organisation qu'il ou elle représente ou qui le/la finance (s'il en est), la INTERNET SOCIETY et la INTERNET ENGINEERING TASK FORCE déclinent toutes garanties, exprimées ou implicites, y compris mais non limitées à toute garantie que l'utilisation des informations ci-encloses ne violent aucun droit ou aucune garantie implicite de commercialisation ou d'aptitude à un objet particulier.

Propriété intellectuelle

L'IETF ne prend pas position sur la validité et la portée de tout droit de propriété intellectuelle ou autres droits qui pourraient être revendiqués au titre de la mise en œuvre ou l'utilisation de la technologie décrite dans le présent document ou sur la mesure dans laquelle toute licence sur de tels droits pourrait être ou n'être pas disponible ; pas plus qu'elle ne prétend avoir accompli aucun effort pour identifier de tels droits. Les informations sur les procédures de l'ISOC au sujet des droits dans les documents de l'ISOC figurent dans les BCP 78 et BCP 79.

Des copies des dépôts d'IPR faites au secrétariat de l'IETF et toutes assurances de disponibilité de licences, ou le résultat de tentatives faites pour obtenir une licence ou permission générale d'utilisation de tels droits de propriété par ceux qui mettent en œuvre ou utilisent la présente spécification peuvent être obtenues sur le répertoire en ligne des IPR de l'IETF à <http://www.ietf.org/ipr> .

L'IETF invite toute partie intéressée à porter son attention sur tous copyrights, licences ou applications de licence, ou autres droits de propriété qui pourraient couvrir les technologies qui peuvent être nécessaires pour mettre en œuvre la présente norme. Prière d'adresser les informations à l'IETF à ietf-ipr@ietf.org .

Remerciement

Le financement de la fonction d'édition des RFC est actuellement fourni par l'Internet Society.