Groupe de travail Réseau
Request for Comments : 4521
BCP : 118
Catégorie : Conseils d'application pratique

K. Zeilenga
OpenLDAP Foundation

Juin 2006
Traduction : Claude Brière de L'Isle

Considérations sur les extensions du protocole léger d'accès aux répertoires (LDAP)

Statut du présent Mémo

Le présent document spécifie les meilleures pratiques courantes pour l'Internet à l'usage de la communauté Internet, et invite à des discussion et suggestions d'améliorations. La diffusion du présent mémo n'est soumise à aucune restriction.

Déclaration de copyright

Résumé

Le protocole léger d'accès aux répertoires (LDAP) est extensible. Il fournit des mécanismes pour ajouter de nouvelles opérations, étendre des opérations existantes, et étendre les schémas d'utilisateur et de système. Le présent document expose les considérations pour les concepteurs d'extensions de LDAP.

Table des Matières

1 Introduction

Le protocole léger d'accès aux répertoires (LDAP) [RFC4510] est un protocole extensible.

LDAP permet que de nouvelles opérations soient ajoutées aux opérations existantes pour les améliorer [RFC4511].

LDAP permet que des schémas supplémentaires soient définis [RFC4512][RFC4517]. Ceci peut inclure des classes d'objet supplémentaires, des types d'attribut, des règles de correspondance, des syntaxes supplémentaires, et d'autres éléments de schéma. LDAP fournit la capacité d'étendre les types d'attribut avec des options [RFC4512].

LDAP prend en charge une méthode d'authentification Authentification simple et couche de sécurité (SASL, *Simple Authentication and Security Layer*) [RFC4511][RFC4513]. SASL [RFC4422] est extensible. LDAP peut être étendu pour prendre en charge des méthodes d'authentification supplémentaires [RFC4511].

LDAP prend en charge l'établissement de la sécurité de couche transport (TLS, *Transport Layer Security*) [RFC4511][RFC4513]. TLS [RFC4346] est extensible.

LDAP a un format extensible de localisateur de ressource uniforme (URL, *Uniform Resource Locator*) [RFC4516].

Enfin, LDAP permet certaines extensions à la notation de syntaxe abstraite n° 1 (ASN.1, *Abstract Syntax Notation - One*) du protocole [X.680] dont la définition reste à faire. Ceci facilite une large gamme d'améliorations du protocole, par exemple, de nouveaux codes de résultat nécessaires pour prendre en charge les extensions à ajouter à la définition de l'extension de l'ASN.1 du protocole.

Le présent document décrit les pratiques que les ingénieurs devraient prendre en compte lors de la conception des extensions à LDAP.

1.1 Terminologie

Les mots clés "DOIT", "NE DOIT PAS", "EXIGÉ", "DEVRAIT", "NE DEVRAIT PAS", "RECOMMANDÉ", "PEUT", et "FACULTATIF" dans le présent document sont à interpréter comme décrit dans le BCP 14 [RFC2119]. Dans le présent document, "la spécification", comme utilisé par le BCP 14, RFC 2119, se réfère à l'ingénierie des extensions LDAP.

Le terme "Contrôle de demande" se réfère à un contrôle attaché à un message généré par un client envoyé à un serveur. Le terme "Contrôle de réponse" se réfère à un contrôle attaché à un message généré par un serveur, envoyé à un client.

DIT signifie arbre d'information de répertoire *(Directory Information Tree)*.

DSA signifie agent de système de répertoire *(Directory System Agent)*, un serveur.

DSE signifie entrée spécifique de DSA *(DSA-Specific Entry)*.

DUA signifie agent d'utilisateur de répertoire *(Directory User Agent)*, un client.

DN signifie nom distinctif *(Distinguished Name)*.

2 Considérations générales

2.1 Portée de l'extension

Les homologues qui font l'objet d'un accord mutuel, dans le cadre d'une extension, s'accordent sur des changements significatifs à la sémantique du protocole. Cependant, les concepteurs DOIVENT considérer l'impact d'une extension sur les homologues de protocole qui ne se sont pas mis d'accord pour mettre en œuvre ou reconnaître d'une façon ou d'une autre et prendre en charge l'extension. Les extensions DOIVENT être "véritablement facultatives" [RFC2119].

2.2 Interactions entre les extensions

Les concepteurs DEVRAIENT considérer la façon dont les extensions qu'ils conçoivent interagissent avec les autres extensions.

Les concepteurs DEVRAIENT considérer l'extensibilité des extensions qu'ils spécifient. Les extensions à LDAP DEVRAIENT elles-mêmes être extensibles.

Sauf mention contraire, l'extensibilité est implicite.

2.3 Mécanisme de découverte

Les extensions DEVRAIENT fournir des mécanismes de découverte adéquats.

Comme la conception de LDAP est fondée sur le paradigme demande du client/réponse du serveur, l'approche générale de découverte est pour le client de découvrir les capacités du serveur avant d'utiliser une extension particulière. Normalement, cette découverte implique d'interroger le DSE racine et/ou d'autres DSE sur les informations fonctionnelles associées à l'extension. LDAP ne fournit pas de mécanisme pour qu'un serveur découvre les capacités d'un client.

L'attribut 'supportedControl' [RFC4512] est utilisé pour faire connaître les commandes prises en charge. L'attribut 'supportedExtension' [RFC4512] st utilisé pour faire connaître les opérations étendues prises en charge. L'attribut 'supportedFeatures' [RFC4512] est utilisé pour faire connaître les caractéristiques. D'autres attributs DSE racines PEUVENT être définis pour faire connaître d'autres capacités.

2.4 Considérations d'internationalisation

LDAP est conçu pour prendre en charge le répertoire de caractères Unicode [Unicode] complet. Les extensions DEVRAIENT éviter les applications inutilement restrictives à des sous ensembles de Unicode (par exemple, Basic Multilingual Plane, ISO 8859-1, ASCII, Printable String).

Les options d'étiquette de langage LDAP [RFC3866] fournissent un mécanisme pour étiqueter les valeurs de texte (et autres) avec des informations de langue. Les extensions qui définissent des types d'attribut DEVRAIENT permettre l'utilisation d'étiquettes de langage avec ces attributs.

2.5 Utilisation des règles de codage de base

De nombreux éléments de LDAP sont décrits à l'aide de ASN.1 [X.680] et sont codés en utilisant un sous-ensemble particulier [Protocole, paragraphe 5.2] des Règles de codage de base (BER) [X.690]. Pour permettre la réutilisation des analyseurs/générateurs utilisés dans la mise en œuvre de la spécification technique LDAP "cœur" [RFC4510], il est RECOMMENDÉ que les éléments d'extension (par exemple, les contenus spécifiques d'extension des champs controlValue, requestValue, responseValue) décrits en ASN.1 et codés en utilisant les BER soient soumis aux restrictions du [Protocole, paragraphe 5.2].

2.6 Utilisation de langages formels

Les langages formels DEVRAIENT être utilisés dans les spécifications conformément aux lignes directrices IESG [FORMAL].

2.7 Exemples

Les exemples de chaînes de noms distinctifs DEVRAIENT être conformes à la syntaxe définie dans la [RFC4518].

Les exemples de chaînes de filtre LDAP DEVRAIENT se conformer à la syntaxe définie dans la [RFC4515]. Les exemples d'URL LDAP DEVRAIENT se conformer à la syntaxe définie dans la [RFC4516]. Les entrées DEVRAIENT être représentées en utilisant LDIF [RFC2849].

2.8 Enregistrement des valeurs de protocole

Les concepteurs DOIVENT enregistrer les valeurs de protocole de leurs extensions LDAP conformément au BCP 64, RFC 4520 [RFC4520]. Les spécifications qui créent de nouveaux éléments de protocole extensibles DOIVENT étendre les registres existants ou établir de nouveaux registres pour les valeurs de ces éléments en accord avec le BCP 64, RFC 4520 [RFC4520] et le BCP 26, RFC 2434 [RFC2434].

3 Extensions d'opérations LDAP

Les extensions DEVRAIENT utiliser des commandes en définissant des extensions qui complètent des opérations existantes. Lorsque l'extension à définir ne complète pas une opération existante, les concepteurs DEVRAIENT considérer à la place la définition d'une opération étendue.

Par exemple, une opération de suppression de sous-arbre pourrait être conçue soit comme une extension de l'opération de suppression soit comme une nouvelle opération. Comme le dispositif complète l'opération de suppression existante, l'utilisation du mécanisme de commande pour étendre l'opération de suppression est vraisemblablement plus appropriée.

A titre d'exemple de compteur (et forcé), une opération de localisation de services (une opération qui retournerait pour un DN un ensemble d'URL LDAP pour des services qui pourraient contenir l'entrée dénommée par ce nom distinctif) pourrait être désignée soit par une opération de recherche soit par une nouvelle opération. Comme le dispositif ne complète pas l'opération de recherche (par exemple, l'opération ne force pas à rechercher des entrées détenues dans l'arbre d'informations du répertoire), il est vraisemblablement plus approprié de définir une nouvelle opération en utilisant le mécanisme d'opération étendue.

3.1 Commandes

Les commandes [Protocole, paragraphe 4.1.11] sont le mécanisme RECOMMENDÉ pour l'extension des opérations existantes. Les opérations existantes peuvent être une opération de base définie dans la [RFC4511] (par exemple, search, modify), une opération étendue (par exemple, Start TLS [RFC4511], Password Modify [RFC3062]), ou une opération définie comme une extension à une opération de base ou étendue.

Les extensions NE DEVRAIENT PAS retourner de commande de réponse si le serveur n'a pas une connaissance spécifique de ce que le client peut utiliser la commande. Généralement, le client demande le retour d'une commande de réponse particulière en fournissant une commande de demande en rapport.

Une opération existante PEUT être étendue pour retourner des messages IntermediateResponse [Protocole, paragraphe 4.13].

Les spécifications des commandes NE DOIVENT PAS lier une sémantique supplémentaire à la virulence des commandes au delà de ce qui est défini dans [Protocole, paragraphe 4.1.11]. Une spécification PEUT rendre obligatoire la virulence d'une valeur particulière (par exemple, VRAI ou FAUX), le cas échéant.

3.1.1 Extension de l'opération Bind avec commandes

Les commandes attachées aux messages de demande et de réponse d'une opération Bind [RFC4511] ne sont protégées par aucune couche de sécurité établie par cette opération Bind.

Les spécifications qui détailleront les commandes étendant l'opération Bind DEVRONT préciser que les couches de sécurité négociées de Bind ne protègent pas les informations contenues dans ces commandes et DEVRONT préciser comment sont protégées les informations dans ces commandes ou pourquoi les informations n'ont pas besoin de protection.

Il est RECOMMANDÉ que les concepteurs pensent à des mécanismes de remplacement pour fournir cette fonction. Par exemple, une opération étendue produite à la suite d'une opération Bind (et donc, protégée par les couches de sécurité négociées par l'opération Bind) pourrait être utilisée pour fournir la fonction souhaité.

De plus, les concepteurs d'extensions de commandes Bind DOIVENT aussi penser à la façon dont la sémantique des commandes interagit avec les étapes individuelles d'une opération Bind à plusieurs étapes. Noter que certaines étapes sont facultatives et pourrait donc requérir une attention particulière lors de la conception.

3.1.2 Extension de l'opération Start TLS avec commandes

Les commandes attachées aux messages de demande et de réponse d'une opération Start TLS [RFC4511] ne sont pas protégées par les couches de sécurité établies par l'opération Start TLS.

Les spécifications précisant les commandes qui étendent l'opération Start TLS DEVRONT préciser que les couches de sécurité négociée de Start TLS ne protègent pas les informations contenues dans

ces commandes et DEVRONT préciser comment sont protégées les informations dans ces commandes ou pourquoi les informations n'ont pas besoin de protection.

Il est RECOMMANDÉ que les concepteurs pensent à des mécanismes de remplacement pour fournir cette fonction. Par exemple, une opération étendue produite à la suite de l'opération Start TLS (et donc, protégée par les couches de sécurité négociées par l'opération Start TLS) pourraient être utilisée pour fournir la fonction désirée.

3.1.3 Extension des opérations Search avec commandes

Le traitement de l'opération Search connaît deux phases distinctes :

- trouver l'objet de base,
- rechercher des objets au niveau ou en dessous de cet objet de base.

Les spécifications des commandes qui étendent l'opération Search devraient clairement établir dans quelle ou quelles phases s'applique la sémantique de la commande. La sémantique des commandes qui ne sont pas spécifiques de l'opération Search DEVRAIT s'appliquer dans la première phase.

3.1.4 Extension des opérations Update (mise à jour) avec commandes

Les opérations Update ont des propriétés d'atomicité, de cohérence, d'isolation, et de durabilité ([ACID]).

- atomicité : Tous les changements demandés du DIT sont effectués ou aucun.
- cohérence : L'état résultant du DIT doit être conformer au schéma et aux autres contraintes.
- isolation : Les états intermédiaires ne sont pas exposés.
- durabilité : L'état résultant du DIT est préservé jusqu'à ce qu'il soit mis à jour.

Lorsqu'on définit une commande qui demande des changements supplémentaires (ou autres) du DIT, ces changements supplémentaires NE DEVRAIENT PAS être traités au titre d'une transaction distincte. Les spécifications DOIVENT être claires quant à la question de savoir si des changements de DIT supplémentaires font partie de la même transaction que les changements de DIT exprimés dans la demande de l'opération de base ou d'une transaction distincte.

Lors de la définition d'une commande qui requiert que des changements de DIT supplémentaires (ou autres) soient faits au DIT, les spécifications DOIVENT être claires sur l'ordre dans lequel ces changements et les changements de base doivent être appliqués au DIT.

3.1.5 Extension des opérations sans réponse avec commandes

Les opérations Abandon et Unbind n'incluent pas de message de réponse. Pour cette raison, les spécifications de commandes conçues pour être attachées à des demandes Abandon et Unbind DEVRAIENT rendre obligatoire que la sensibilité de la commande soit mise à FAUX.

3.2 Opérations étendues

Les opérations étendues [Protocole, paragraphe 4.12] sont le mécanisme RECOMMANDÉ pour définir de nouvelles opérations. Une opération étendue consiste en un message ExtendedRequest, zéro, un ou plusieurs messages IntermediateResponse, et un message ExtendedResponse.

3.3 Réponses intermédiaires

Les extensions DOIVENT utiliser les messages IntermediateResponse plutôt que les messages ExtendedResponse pour retourner les résultats intermédiaires.

3.4 Notifications non sollicitées

Les notifications non sollicitées [Protocole, paragraphe 4.4] offrent au serveur une capacité à notifier au client des événements non associés à l'opération en cours de traitement.

Les extensions DEVRAIENT être conçues de telles sorte que des notifications non sollicitées ne soient pas retournées sauf si le serveur a une connaissance précise de ce que le client peut utiliser la notification. Généralement, le client demande le retour d'une notification non sollicitée particulière en effectuant une opération étendue qui s'y rapporte.

Par exemple, une extension de hachage de temps pourrait être conçue pour retourner des notifications non sollicitées à des intervalles réguliers qui ont été activés par une opération étendue (qui pourrait éventuellement spécifier l'intervalle désiré).

4 Extension de la définition ASN.1 de LDAP

LDAP permet d'effectuer une extension limitée [Protocole, Section 4] de la définition ASN.1 de LDAP [Protocole, Appendice B].

4.1 Codes de résultat

Les extensions qui spécifient de nouvelles opérations ou améliorent des opérations existantes ont souvent besoin de définir de nouveaux codes de résultat. L'extension DEVRAIT être conçue de telle sorte qu'un client ait une indication raisonnablement claire de la nature réussie ou non réussie du résultat.

Les extensions DEVRAIENT utiliser les codes de résultat existants pour indiquer des conditions qui soient cohérentes avec la signification projetée [RFC4511][X.511] de ces codes. Les extensions PEUVENT introduire de nouveaux codes de résultat [RFC4520] lorsque aucun code de résultat existant ne fournit une indication adéquate de la nature du résultat.

Les extensions NE DOIVENT PAS interdire ou restreindre de quelque autre façon le retour de codes de résultat de service général, particulièrement de ceux qui font rapport d'un protocole, service, ou problème de sécurité, ou indiquent que le serveur est incapable ou non désireux d'achever l'opération.

4.2 Types de message LDAP

Bien que les extensions puissent spécifier de nouveaux types de messages LDAP en étendant le CHOIX protocolOp de la SEQUENCE LDAPMessage, ceci est généralement non nécessaire et inapproprié. Les mécanismes d'extension d'opération existants (par exemple, les opérations étendues, les notifications non sollicitées, et les réponses intermédiaires) DEVRAIENT être utilisés à la place. Cependant, il peut y avoir des cas où une extension ne convient pas bien pour ces mécanismes.

Dans de tels cas, un nouveau mécanisme d'extension DEVRAIT être défini qui puisse être utilisé par plusieurs extensions ayant des besoins similaires.

4.3 Méthodes d'authentification

L'opération Bind accepte normalement deux méthodes d'authentification, simple et SASL. SASL [RFC4422] est un cadre d'authentification extensible utilisé par plusieurs protocoles de niveau application (par exemple, BEEP, IMAP, SMTP). Il est RECOMMANDÉ que de nouveaux processus d'authentification soient définis comme mécanismes SASL. De nouvelles méthodes d'authentification LDAP PEUVENT être ajoutées pour prendre en charge de nouveaux cadres d'authentification.

La fonction principale de l'opération Bind est d'établir l'association LDAP [RFC4513]. Aucune autre opération NE DOIT être définie (ou étendue) pour établir l'association LDAP. Cependant, d'autres opérations PEUVENT être définies pour établir d'autres associations de sécurité (par exemple, IPsec).

4.4 Extensibilité ASN.1 générale

La Section 4 de la [RFC4511] établit ce qui suit :

Pour la prise en charge d'extensions futures au présent protocole, l'extensibilité est implicite lorsqu'elle est permise par l'ASN.1 (c'est-à-dire que séquence, ensemble, choix, les types énumérés son extensibles). De plus, les ellipses (...) ont été fournies dans des types ASN.1 qui sont explicitement extensibles, comme exposé dans la [RFC4520]. A cause de leur extensibilité implicite, les clients et serveurs DOIVENT (sauf mention contraire) ignorer les composants SEQUENCE de queue dont ils ne reconnaissent pas les étiquettes.

Les concepteurs DEVRAIENT éviter d'introduire des extensions qui s'appuient sur des mises en œuvre qu'on ne puisse soupçonner d'ignorer des composants de SEQUENCE dont ils ne reconnaissent pas les étiquettes.

5 Extensions de schéma

Les extensions qui définissent des éléments de schéma LDAP DOIVENT fournir des définitions de schéma conformes aux syntaxes définies dans [Modèles, paragraphe 4.1]. Lorsque les définitions fournies PEUVENT être reformatées (coupures de ligne) pour leur lisibilité, cela DOIT être noté dans la spécification.

Pour les définitions qui permettent un champ NAME, les nouveaux éléments de schéma DEVRAIENT fournir un nom et un seul. Le nom DEVRAIT être court.

Chaque définition de schéma permet un champ DESC. Le champ DESC, si il est fourni, DEVRAIT contenir une courte phrase descriptive. Le champ DESC DOIT être regardé comme informatif. C'est-à-dire que la spécification DOIT être écrite de telle sorte que son interprétation soit la même avec et sans les champs DESC fournis.

L'extension NE DOIT PAS rendre obligatoire que les mises en œuvre fournissent le même champ DESC dans le schéma qu'elles publient. Les mises en œuvre PEUVENT remplacer ou retirer le champ DESC.

Les éléments de schéma publiés NE DOIVENT PAS être redéfinis. Les éléments de schéma de remplacement (nouveaux OID, nouveaux NAME) DEVRAIENT être définis en tant que de besoin.

Les concepteurs de schéma DEVRAIENT réutiliser les éléments de schéma existants, là où c'est approprié. Cependant, toute réutilisation NE DOIT pas altérer la sémantique de l'élément.

5.1 Syntaxes LDAP

Chaque syntaxe LDAP [RFC4517] est définie en termes d'ASN.1 [X.680]. Chaque extension précisant une syntaxe LDAP DOIT spécifier la définition des données ASN.1 associée à la syntaxe. Une syntaxe LDAP distincte DEVRAIT être créée pour chaque définition de données ASN.1 distincte (y compris les contraintes).

Chaque syntaxe LDAP DEVRAIT avoir un codage de chaîne défini pour elle. Il est RECOMMANDÉ que ce codage de chaîne soit restreint aux caractères UTF-8 [RFC3629] codés en Unicode [Unicode]. L'utilisation des règles générales de codage de chaîne (GSER, *Generic String Encoding Rules*) [RFC3641] [RFC3642] ou d'autres règles génériques de codage de chaîne pour fournir des codages de chaînes pour les définitions de données ASN.1 complexes est RECOMMANDÉ. Autrement, il est RECOMMANDÉ que le codage de chaîne soit décrit en utilisant un langage formel (par exemple, ABNF [RFC4234]). Les langages formels DEVRAIENT être utilisés dans les spécifications conformément aux lignes directrices IESG [FORMAL].

Si aucun codage de chaîne n'est défini, l'extension DOIT spécifier comment le codage de transfert devra être indiqué. Généralement, l'extension DEVRAIT rendre obligatoire l'utilisation d'une option de codage de transfert binaire ou autre.

5.2 Règles de correspondance

Trois sortes de règles de correspondance de base (par exemple, EQUALITY, ORDERING, et SUBSTRING) peuvent être associées à un type d'attribut. De plus, LDAP fournit un mécanisme de règle de correspondance extensible.

La spécification de règle de correspondance DEVRAIT préciser de quelle sorte de règle de correspondance il s'agit et DEVRAIT décrire quelles sortes de valeurs on peut utiliser avec elle.

En plus des exigences établies dans la spécification technique LDAP, les règles de correspondance d'égalité DEVRAIENT être commutatives.

5.3 Types d'attribut

Les concepteurs DEVRAIENT considérer avec soin comment restreindre la structure des valeurs. Les concepteurs DEVRAIENT considérer que les serveurs ne mettront en application que les contraintes

sur la syntaxe d'attribut. C'est-à-dire qu'un attribut destiné à contenir des URI, mais qui a une syntaxe de directoryString, n'est pas restreint aux valeurs qui sont des URI.

Les concepteurs DEVRAIENT considérer avec soin les règles de correspondance, s'il en est, qui sont appropriées pour le type d'attribut. Les règles de correspondance spécifiées pour un type d'attribut DOIVENT être compatibles avec la syntaxe de ce type d'attribut.

Les extensions qui spécifient des attributs de fonctionnement DOIVENT préciser comment les serveurs vont maintenir et/ou utiliser les valeurs de chaque attribut de fonctionnement.

5.4 Classes d'objet

Les concepteurs DEVRAIENT considérer avec soin si chaque attribut d'une classe d'objet est nécessaire ("DOIT") ou permis ("PEUT").

Les extensions qui spécifient des classes d'objet qui permettent (ou exigent) des attributs de fonctionnement DOIVENT spécifier comment les serveurs vont maintenir et/ou utiliser les entrées qui appartiennent à ces classes d'objet.

6 Autres mécanismes d'extension

6.1 Options de description d'attribut

Chaque option est identifiée par une chaîne de lettres, nombres, et tirets. Cette chaîne DEVRAIT être courte.

6.2 Identités d'autorisation

Les extensions qui interagissent avec des identités d'autorisation DOIVENT prendre en charge le format LDAP authzId [RFC4513]. Le format authzId est extensible.

6.3 Extensions d'URL LDAP

Les extensions d'URL LDAP sont identifiées par une courte chaîne, un descripteur. Comme les autres descripteurs, les chaînes DEVRAIENT être courtes.

7 Considérations sur la sécurité

LDAP n'impose aucune restriction indue sur les sortes d'extensions qui peuvent être mises en œuvre. Bien que le présent document essaye de souligner les questions spécifiques que les concepteurs ont besoin de prendre en compte, il n'est pas (et ne peut pas être) exhaustif. Les concepteurs DOIVENT faire leurs propres évaluations des considérations de sécurité applicables à leurs extensions.

Les concepteurs NE DOIVENT PAS supposer que la spécification technique LDAP "cœur" [RFC4510] traite de façon adéquate les questions spécifiques qui entourent leurs extensions ou supposer que leurs extensions n'ont pas de problèmes spécifiques.

Les spécifications d'extension DEVRAIENT cependant noter si des considérations de sécurité spécifiques de la caractéristique qu'ils étendent, aussi bien que les considérations générales de sécurité de LDAP, s'appliquent à l'extension.

8 Remerciements

L'auteur remercie la communauté LDAP de l'IETF pour ses commentaires pertinents.

Ce travail s'appuie sur le "Guide du style d'extension LDAP" [GUIDE] de Bruce Greenblatt.

9 Références

9.1 Références normatives

[RFC2119] Bradner, S., "Mots clés à utiliser dans les RFC pour indiquer les niveaux d'exigence", BCP 14, RFC 2119, mars 1997.

[RFC2434] Narten, T. et H. Alvestrand, "Lignes directrices pour écrire une section Considérations relatives à l'IANA dans les RFCs" BCP 26, RFC 2434, octobre 1998.

[RFC2849] Good, G., "Le format LDAP d'échange de données (LDIF) - Spécification technique", RFC 2849, juin 2000.

[RFC3629] Yergeau, F., "UTF-8, un format de transformation d'ISO 10646", STD 63, RFC 3629, novembre 2003.

[RFC3641] Legg, S., "Règles génériques de codage de chaînes (GSER) pour les types ASN.1", RFC 3641, octobre 2003.

[RFC3642] Legg, S., "Eléments communs des règles génériques de codage de chaînes (GSER)", RFC 3642, octobre 2003.

[RFC4512] Zeilenga, K., "Protocole léger d'accès aux répertoires (LDAP) : Modèles d'informations de répertoire", RFC 4512, juin 2006.

[RFC3866] Zeilenga, K., Ed., "Étiquettes et gammes de langage dans le Protocole léger d'accès aux répertoires (LDAP)", RFC 3866, July 2004.

[RFC4234] Crocker, D. et P. Overell, "BNF augmenté pour spécifications de syntaxe : ABNF", RFC 4234, octobre 2005.

[RFC4510] Zeilenga, K., Ed., "Protocole léger d'accès aux répertoires (LDAP) : Plan d'accès des spécifications techniques, RFC 4510, juin 2006.

[RFC4511] Sermersheim, J., Ed., "Protocole léger d'accès aux répertoires (LDAP) : Le protocole", RFC 4511, juin 2006.

[RFC4512] Zeilenga, K., "Protocole léger d'accès aux répertoires (LDAP) : Modèles d'information des répertoires", RFC 4512, juin 2006.

[RFC4513] Harrison, R., Ed., "Protocole léger d'accès aux répertoires (LDAP) : Méthodes d'authentification et mécanismes de sécurité", RFC 4513, juin 2006.

[RFC4515] Smith, M., Ed. et T. Howes, "Protocole léger d'accès aux répertoires (LDAP) : Représentation de chaîne des filtres de recherche", RFC 4515, juin 2006.

[RFC4516] Smith, M., Ed. et T. Howes, "Protocole léger d'accès aux répertoires (LDAP) : Localisateur de ressource uniforme", RFC 4516, juin 2006.

[RFC4517] Legg, S., Ed., "Protocole léger d'accès aux répertoires (LDAP) : Syntaxes et règles de correspondance", RFC 4517, juin 2006.

[RFC4518] Zeilenga, K., "Protocole léger d'accès aux répertoires (LDAP) : Représentation de chaîne de noms distinctifs", RFC 4518, juin 2006.

[RFC4520] Zeilenga, K., "Autorité d'allocation des numéros Internet (IANA) Considérations pour le Protocole léger d'accès aux répertoires (LDAP)", BCP 64, RFC 4520, juin 2006.

[RFC4422] Melnikov, A., Ed. et K. Zeilenga, Ed., "Authentification simple et couche de sécurité (SASL)", RFC 4422, juin 2006.

[Unicode] Consortium Unicode, "La norme Unicode, Version 3.2.0" est définie par "Norme Unicode, Version 3.0" (Reading, MA, Addison-Wesley, 2000. ISBN 0-201-61633-5), telle qu'amendée par la Norme Unicode, Annexe n° 27 : Unicode 3.1" (http://www.unicode.org/reports/tr27/) et par la Norme Unicode, Annexe n° 28 : Unicode 3.2" (http://www.unicode.org/reports/tr28/).

[FORMAL] IESG, "Lignes directrices pour l'utilisation des langages formels dans les spécifications de l'IETF", <http://www.ietf.org/IESG/STATEMENTS/pseudo-code-in- specs.txt>, 2001.

[X.511] Union Internationale des Télécommunications - Secteur de la Normalisation des Télécommunications, "L'Annuaire : Définition de service abstraite", X.511 (1993) (aussi ISO/IEC 9594-3:1993).

[X.680] Union Internationale des Télécommunications - Secteur de la Normalisation des Télécommunications, "Notation de syntaxe abstraite n° 1 (ASN.1) - Spécification de la notation de base", X.680 (2002) (aussi ISO/IEC 8824-1:2002).

[X.690] Union Internationale des Télécommunications - Secteur de la Normalisation des Télécommunications, "Spécification des règles de codage ASN.1 : Règles de codage de base (BER), Règles de codage canoniques (CER), et règles de codage distinctives (DER)", X.690 (2002) (aussi ISO/IEC 8825-1:2002).

9.2 Références informatives

[ACID] Section 4 de ISO/CEI 10026-1:1992.

[GUIDE] Greenblatt, B., "Guide de style d'extension LDAP", travail en cours.

[RFC3062] Zeilenga, K., "Opération étendue de modification de mot de passe LDAP", RFC 3062, février 2001.

[RFC4346] Dierks, T. et E. Rescorla, "Protocole de sécurité de la couche Transport (TLS) Version 1.1", RFC 4346, avril 2006.

Adresse de l'auteur

Kurt D. Zeilenga

OpenLDAP Foundation

EMail: Kurt@OpenLDAP.org

Déclaration de copyright

Propriété intellectuelle

RemerciementLe financement de la fonction d'édition des RFC est fourni par la Administrative Support Activity (IASA) de l'IETF.