

Groupe de travail sur les réseaux
Requête pour Commentaires : 2253
Rend obsolète : 1779
Catégorie : Standard

M. Wahl
Critical Angle Inc.
S. Kille
Isode Ltd.
T. Howes
Netscape Communications Corp.
Décembre 1997
Lycée la croix-rouge - Brest

Traduction : Yves lescop

Protocole allégé d'accès à un annuaire LDAP (v3) : Représentation en chaîne de caractères UTF-8 des noms différenciés

Statut de ce document

Ce document spécifie un protocole standard d'Internet pour la communauté Internet, et ne sera éprouvé qu'après plusieurs discussions et suggestions. Merci de vous référer à l'édition courante du " Internet Official Protocol Standards " (STD1) pour l'état de standardisation et le statut de ce protocole. La distribution de ce document est illimitée.

Copyright

Copyright © "Internet society" (1997) – tous droits réservés.

Note d'IESG

Ce document décrit un protocole d'accès à un annuaire qui fournit tant l'accès en lecture que l'accès pour mise à jour. L'accès de mise à jour exige une authentification sécurisée, mais ce document n'exige la mise en place d'aucun mécanisme d'authentification adéquat.

Selon RFC 2026, section 4.4.1, cette spécification est approuvée par IESG comme norme proposée en dépit de cette limitation, pour les raisons suivantes :

- a. pour encourager la mise en place et le test d'interopérabilité de ces protocoles (avec ou sans l'accès de mise à jour) avant qu'ils soient déployés, et
- b. pour encourager le déploiement et l'utilisation de ces protocoles dans des applications à lecture seule. (par exemple applications où LDAPv3 est utilisé comme langage d'interrogation pour les annuaires qui sont mis à jour par un mécanisme sécurisé autre que LDAP), et

- c. pour éviter de retarder l'avancement et le déploiement d'autres protocoles standard d'Internet qui exigent la possibilité de questionner, mais pas de mettre à jour, des serveurs d'annuaire LDAPv3.

Les lecteurs sont avertis par la présente que jusqu'à ce que des mécanismes obligatoires d'authentification soient normalisés, les clients et les serveurs écrits selon cette spécification qui se servent de la fonctionnalité de mise à jour sont **IMPROBABLEMENT INTEROPERABLE**, ou **PEUVENT INTEROPERER SEULEMENT SI L'AUTHENTIFICATION EST RÉDUITE À UN NIVEAU INADMISSIBLEMENT FAIBLE**.

Les implanteurs sont découragés par la présente de déployer des clients ou des serveurs LDAPv3 qui mettent en œuvre la fonctionnalité de mise à jour, jusqu'à ce qu'une norme proposée pour l'authentification obligatoire dans LDAPv3 ait été approuvée et éditée comme RFC.

Résumé

L'annuaire X.500 utilise des noms différenciés comme clés primaires aux entrées dans l'annuaire. Les noms différenciés sont encodés en ASN.1 dans les protocoles d'annuaire X.500. Dans le protocole LDAP, une représentation en chaîne de caractères des noms différenciés est transférée. Cette spécification définit le format de chaîne de caractères pour représenter des noms, qui est conçu pour donner une représentation propre des noms différenciés généralement utilisés, tout en pouvant représenter tout nom différencié.

Les mots clés "DOIT", "NE DOIT PAS", "REQUIS", "DEVRA", "NE DEVRA PAS", "DEVRAIT", "NE DEVRAIT PAS", "RECOMMANDÉ", "PEUT", et "FACULTATIF" dans ce document doivent être interprétés comme décrit dans RFC 2119 [6].

1. Arrière-plan

Cette spécification présume la connaissance de X.500 [1], et le concept du nom différencié. Il est important d'avoir un format commun pour pouvoir représenter clairement un nom différencié. Le but premier de cette spécification est de faciliter l'encodage et le décodage. Un but secondaire est d'avoir des noms qui sont lisibles par les humains. On ne s'attend pas à ce que les clients LDAP avec une interface utilisateur humaine affichent ces chaînes de caractères directement à l'utilisateur, mais exécuteraient très probablement des traductions (telles qu'exprimer les noms des types d'attribut dans une des langues nationales locales).

2. Conversion des noms différenciés ASN.1 en chaîne de caractères

Dans X.501 [2] la structure ASN.1 du nom différencié est définie comme :

```
DistinguishedName ::= RDNSequence
```

```

RDNSequence ::= SEQUENCE OF RelativeDistinguishedName

RelativeDistinguishedName ::= SET SIZE (1..MAX) OF
  AttributeTypeAndValue

AttributeTypeAndValue ::= SEQUENCE {
  type AttributeType,
  value AttributeValue }

```

Les sections suivantes définissent l'algorithme pour convertir d'une représentation structurée par ASN.1 en une représentation chaîne de caractères UTF-8.

2.1. Transcodage du "RDNSequence"

Si le "RDNSequence" (séquence du nom différencié relatif) est une séquence vide, le résultat est la chaîne de caractères vide ou de longueur nulle.

Autrement, la sortie consiste en des encodages en chaîne de caractères de chaque RDN (nom différencié relatif) dans le "RDNSequence" (selon 2.2), commençant par le dernier élément de la séquence et en se déplaçant en arrière vers le premier.

Les encodages des noms différenciés relatifs voisins sont séparés par une virgule (',' ASCII 44).

2.2. Transcodage du RDN

En convertissant un nom différencié relatif (RDN = "RelativeDistinguishedName") ASN.1 en chaîne de caractères, la sortie consiste en des encodages en chaîne de caractères de chaque "AttributeTypeAndValue" (selon 2.3), dans n'importe quel ordre.

Là où il y a des RDN à valeurs multiples, les sorties des "AttributeTypeAndValues" voisins sont séparées par un signe plus ('+' caractère ASCII 43).

2.3. Transcodage de "AttributeTypeAndValue"

Le type et la valeur d'un attribut ("AttributeTypeAndValue") est encodé comme la représentation en chaîne de caractères du type d'attribut, suivie du caractère égal ('=' ASCII 61), suivi de la représentation en chaîne de caractères de la valeur de l'attribut. Le codage de la valeur de l'attribut ("AttributeValue") est donné dans la section 2.4.

Si le type de l'attribut est dans une table éditée des types d'attribut liés à LDAP [4], alors la chaîne de caractères du nom du type de cette table est utilisée, autrement il est encodé comme codage décimal pointé de l'IDENTIFICATEUR de l'OBJET du type d'attribut. La notation décimale pointée est décrite dans [3]. Comme exemple, les chaînes de caractères pour quelques-uns des types d'attribut fréquemment vus dans les RDN incluent :

```

Chaîne   Type d'attribut X.500
-----
CN       commonName

```

L	localityName
ST	stateOrProvinceName
O	organizationName
OU	organizationalUnitName
C	countryName
STREET	streetAddress
DC	domainComponent
UID	userid

2.4. Transcodage d'une valeur d'attribut ASN.1 en chaîne de caractères

Si la valeur d'attribut est d'un type qui n'a pas de représentation en chaîne de caractères définie pour lui, alors il est simplement encodé comme un caractère dièse ('#' ASCII 35) suivi de la représentation hexadécimale de chacun des octets en codage BER de la valeur d'attribut X.500. Cette forme DEVRAIT être utilisée si le type d'attribut est de la forme décimale pointée.

Autrement, si la valeur d'attribut est d'un type qui a une représentation en chaîne de caractères, la valeur est convertie d'abord en chaîne de caractères UTF-8 selon sa spécification de syntaxe (voir par exemple la section 6 de [4]).

Si la chaîne de caractères UTF-8 n'a aucun des caractères suivants qui ont besoin d'être échappé, alors cette chaîne de caractères peut être utilisée comme représentation de la valeur.

- un espace ou caractère "#" se produisant au début de la chaîne de caractères
- un caractère d'espace se produisant à la fin de la chaîne de caractères
- un des caractères ",", "+", "'", "\", "<", ">" ou ";"

Les implémentations PEUVENT échapper d'autres caractères.

Si un caractère à échapper est un de ceux de la liste montrée ci-dessus, alors il est préfixé par un antislash ('\ ASCII 92).

Autrement le caractère à échapper est remplacé par un antislash et deux chiffres hexadécimaux, qui forment un octet simple dans le code du caractère.

Des exemples du mécanisme d'échappement sont montrés dans la section 5.

3. Analyse d'une chaîne de caractères pour retrouver un nom différencié

La structure de la chaîne de caractères est spécifiée dans une grammaire BNF, basée sur la grammaire définie dans RFC 822 [5]. Les implantations de serveur analysant une chaîne de caractères de DN produite par un client LDAPv2 DOIVENT également accepter (et ignorer) les variantes données dans la section 4 de ce document.

```

distinguishedName = [nom] ; peut être la chaîne de caractères vide

name          = name-component *( "," name-component )

name-component = attributeTypeAndValue *( "+" attributeTypeAndValue )

attributeTypeAndValue = attributeType "=" attributeValue

attributeType = ( ALPHA 1*keychar ) / oid
keychar      = ALPHA / DIGIT / "-"

oid          = 1*DIGIT *( "." 1*DIGIT )

attributeValue = string

string       = *( stringchar / pair )
              / "#" hexstring
              / QUOTATION *( quotechar / pair ) QUOTATION ; seulement de v2

quotechar    = <n'importe quel caractère excepté "\" ou QUOTATION >

special      = ", " / "=" / "+" / "<" / ">" / "#" / ";"

pair         = "\" ( special / "\" / QUOTATION / hexpair )

stringchar   = <n'importe quel caractère excepté un des spéciaux, "\" ou
QUOTATION >

hexstring    = 1*hexpair
hexpair      = hexchar hexchar

hexchar      = DIGIT / "A" / "B" / "C" / "D" / "E" / "F"
              / "a" / "b" / "c" / "d" / "e" / "f"

ALPHA        = <toute lettre ASCII>
              ; (décimal 65-90 et 97-122)

DIGIT        = <tout chiffre décimal ASCII> ; (décimal 48-57)

QUOTATION    = <le caractères ASCII guillemet "'" décimal 34>

```

4. Parenté avec RFC 1779 et LDAPv2

La syntaxe donnée dans ce document est plus restrictive que la syntaxe dans RFC 1779. Les implémentations analysant une chaîne de caractères produite par un client LDAPv2 DOIVENT accepter la syntaxe de RFC 1779. Les implémentations NE DOIVENT, cependant, produire aucun des encodages de RFC 1779 qui ne sont pas décrits ci-dessus dans la section 2.

Les implémentations DOIVENT permettre à un caractère point virgule d'être employé au lieu d'une virgule pour séparer les RDN dans un nom différencié, et DOIVENT également permettre à des caractères d'espacement d'être présents de chaque côté de la virgule ou du point virgule. Les caractères d'espacement sont ignorés, et le point virgule est remplacé par une virgule.

Les implémentations DOIVENT permettre à un oid dans le type d'attribut d'être préfixé par une des chaînes de caractères "oid." ou "OID.".

Les implémentations DOIVENT permettre aux caractères espace (' ' ASCII 32) d'être présents entre le composant nom et ',', entre le type et la valeur d'attribut et '+', entre le type d'attribut et '=', et entre '=' et la valeur d'attribut. Ces caractères d'espace sont ignorés lors de l'analyse.

Les implémentations DOIVENT permettre à une valeur d'être entourées par des guillemets ("" ASCII 34), qui ne font pas partie de la valeur. À l'intérieur de la valeur entre guillemets, les caractères suivants peuvent apparaître sans être échappés :

"," , "=" , "+" , "<" , ">" , "#" et ";"

5. Exemples

Cette notation est conçue pour être commode pour les formes communes du nom. Cette section donne quelques exemples des noms différenciés écrits en utilisant cette notation. Le premier est un nom contenant trois noms différenciés relatifs (RDN) :

CN=Steve Kille,O=Isode Limited,C=GB

Voici un exemple de nom contenant trois RDN, dans lequel le premier RDN est à valeurs multiples :

OU=Sales+CN=J. Smith,O=Widget Inc.,C=US

Cet exemple montre la méthode de citation d'une virgule dans un nom d'organisation :

CN=L. Eagle,O=Sue\, Grabbit and Runn,C=GB

Un exemple de nom dans lequel une valeur contient un caractère retour chariot :

CN=Before\0DAfter,O=Test,C=GB

Un exemple de nom dans lequel un RDN était d'un type non reconnu. La valeur est le codage BER d'une CHAÎNE DE CARACTÈRES d'OCTET contenant deux octets 0x48 et 0x69.

1.3.6.1.4.1.1466.0=#04024869,O=Test,C=GB

En conclusion, un exemple d'une valeur de nom de famille de RDN se composant des 5 lettres :

Description	Lettre	Unicode	code 10646	UTF-8	Cité
LATIN CAPITAL LETTER L	L	U0000004C	0x4C	L	
LATIN SMALL LETTER U	u	U00000075	0x75	u	
LATIN SMALL LETTER C WITH CARON	č	U0000010D	0xC48D	\C4\8D	
LATIN SMALL LETTER I	i	U00000069	0x69	i	
LATIN SMALL LETTER C WITH ACUTE	ç	U00000107	0xC487	\C4\87	

Pourrait être écrit dans l'ASCII imprimable (utile pour la mise au point) :

```
SN=Lu\C4\8Di\C4\87
```

6. Références

- [1] L'annuaire -- vue d'ensemble des concepts, des modèles et des services. ITU-T Rec. X.500(1993).
- [2] L'annuaire -- Modèles. ITU-T Rec. X.501(1993).
- [3] Wahl, M., Howes, T., et S. Kille, "Lightweight Directory Access Protocol (v3)", RFC 2251, décembre 1997.
- [4] Wahl, M., Coulbeck, A., Howes, T. et S. Kille, "LDAP (v3) : Définitions de Syntaxe d'Attribut ", RFC 2252, décembre 1997.
- [5] Crocker, D., "Standard of the Format of ARPA-Internet Text Messages", STD 11, RFC 822, août 1982.
- [6] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", RFC 2119.

7. Considérations Sécuritaires

7.1. Divulgateion

Les noms différenciés se composent typiquement des informations descriptives sur les entrées qu'ils nomment, qui peuvent être des gens, des organismes, des dispositifs ou d'autres objets réels. Ceci inclut fréquemment certains des genres d'information suivants :

- le nom commun de l'objet (c.-à-d. le nom et prénoms d'une personne)
- un email ou une adresse TCP/IP
- son emplacement physique (pays, localité, ville, adresse de rue)
- attributs d'organisation (tels que le nom ou l'affiliation du service)

La plupart des pays ont des lois sur la vie privée concernant la publication d'informations sur des personnes.

7.2. Utilisation des noms différenciés dans des applications de sécurité

Les transformations d'une valeur "AttributeValue" de sa forme X.501 à une représentation de chaîne de caractères LDAP ne sont pas toujours réversibles vers la même forme BER ou DER. Un exemple d'une situation qui exige la forme DER d'un nom différencié est la vérification d'un certificat X.509.

Par exemple, un nom différencié se composant d'un RDN avec un AVA, dans lequel le type est "commonName" et la valeur est du choix "TeletexString" avec les lettres 'SAM' serait représenté dans LDAP comme chaîne de caractères CN=Sam. Un autre nom différencié dans lequel la valeur est toujours 'SAM' mais de choix "PrintableString" aurait la même représentation CN=Sam.

Les applications qui exigent la reconstruction de la forme DER de la valeur NE DEVRAIENT PAS utiliser la représentation en chaîne de caractères des syntaxes d'attribut quand elles convertissent une valeur en format LDAP. Au lieu de cela, elles DEVRAIENT utiliser la forme hexadécimale préfixée par le dièse (#) comme décrit dans le premier paragraphe de la section 2.4.

8. Adresses des Auteurs

Mark Wahl
Critical Angle Inc.
4815 W. Braker Lane #502-385
Austin, TX 78759
USA

Email : M.Wahl@critical-angle.com.

Steve Kille
Isode Ltd.
The Dome
The Square
Richmond, Surrey
TW9 1DT
England

Phone: +44-181-332-9091
EMail: S.Kille@ISODE.COM

Tim Howes
Netscape Communications Corp.
501 E. Middlefield Rd, MS MV068
Mountain View, CA 94043
USA

Phone: +1 650 937-3419
EMail: howes@netscape.com

9. Copyright intégral

Copyright © The Internet Society (1997). Tous Droits Réservés.

Le document anglais original et les traductions de celui-ci peuvent être copiés et fournis à d'autres, et les travaux dérivés qui le commente ou l'explique ou facilite son implémentation peuvent être préparés, copiés, publiés ou distribués, en totalité ou en partie, sans aucune restriction tant que les observations ci-dessus sur le copyright et ce paragraphe sont inclus dans tous ces types de copies ou de travaux dérivés. Cependant, le document anglais original lui-même ne peut être modifié de quelque façon que ce soit, comme par exemple en retirant les observations de copyright ou les références à la Internet Society ou aux autres organismes de l'Internet, excepté comme l'exige le but du développement des standards Internet où dans un tel cas les procédures pour les copyrights définis dans le processus des Standards Internet doivent être suivies, ou alors comme l'exige une traduction dans une langue autre que l'anglais.

Les autorisations limitées accordées ci-dessus sont éternelles et ne pourront être révoquées par la Internet Society, ses successeurs ou ses repreneurs.

Ce document et les informations contenues ici sont fournis de façon " TELS QUELS " et les traducteurs, la Internet Society et la Internet Engineering Task Force déclinent toute garantie, explicites ou implicites, y compris mais pas seulement toute garantie que l'utilisation des informations de ce document ne violera pas des réglementations ou des garanties implicites commerciales ou physiques pour une application particulière.