

Groupe de travail Réseau  
**Request for Comments : 4466**  
 RFC mises à jour : 2088, 2342, 3501, 3502, 3516  
 Catégorie : Sur la voie de la normalisation

A. Melnikov, Isode Ltd.  
 C. Daboo  
 avril 2006  
 Traduction Claude Brière de L'Isle

## Récapitulation des extensions à l'ABNF pour IMAP4

### Statut du présent mémoire

Le présent document spécifie un protocole de l'Internet en cours de normalisation pour la communauté de l'Internet, et appelle à des discussions et suggestions pour son amélioration. Prière de se référer à l'édition en cours des "Protocoles officiels de l'Internet" (STD 1) pour voir 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 (2006).

### Résumé

Au fil des ans, de nombreux documents provenant des groupes de travail IMAPEXT et LEMONADE, ainsi que de nombreux documents individuels, ont ajouté des extensions syntaxiques à de nombreuses commandes IMAP de base décrites dans la RFC 3501. Pour faciliter les références, le présent document collecte la plupart de ces changements d'ABNF en un seul endroit.

Le présent document suggère aussi un ensemble de schémas standard pour ajouter des options et extensions à plusieurs commandes IMAP existantes définies dans la RFC 3501. Les schémas assurent la compatibilité entre les extensions existantes et futures.

Le présent document met à jour l'ABNF dans les RFC 2088, 2342, 3501, 3502, et 3516. Il inclut aussi une partie des errata à la RFC 3501. Le présent document ne spécifie aucun changement de sémantique à ces RFC.

## Table des matières

1. Introduction.....	1
1.1 Objet du document.....	1
1.2 Conventions utilisées dans le document.....	2
2. Extensions d'ABNF IMAP.....	2
2.1 Paramètres facultatifs avec les commandes SELECT/EXAMINE.....	2
2.2 Extension de commande CREATE.....	3
2.3 Extension de commande RENAME.....	3
2.4 Extensions aux commandes FETCH et UID FETCH.....	4
2.5 Extensions aux commandes STORE et UID STORE.....	4
2.6 Extensions à la commande SEARCH.....	5
2.7 Extensions à la commande APPEND.....	6
3. Syntaxe formelle.....	6
4. Considérations sur la sécurité.....	9
5. Références normatives.....	9
6. Remerciements.....	10
Adresse des auteurs.....	10
Déclaration complète de droits de reproduction.....	10

## 1. Introduction

### 1.1 Objet du document

Le présent document vise à plusieurs objets :

1. rationaliser et généraliser l'ABNF pour certaines extensions à IMAP existantes,

2. collecter l'ABNF en un endroit afin de minimiser les références croisées entre les documents,
3. définir les blocs de construction des futures extensions afin qu'elles puissent être utilisées ensemble d'une façon compatible.

Il est prévu qu'une future révision de ce document soit incorporée dans une révision de la RFC 3501.

Le présent document met à jour l'ABNF dans les RFC 2088, 2342, 3501, 3502, et 3516. Il inclut aussi des parties des errata à la RFC 3501. Le présent document ne spécifie aucun changement de la sémantique des RFC citées.

L'ABNF de la Section 6 de la RFC 2342 a été réécrit pour se conformer à la syntaxe ABNF définie dans la RFC 4234 et pour faire référence aux nouveaux non terminaux de la RFC 3501. Il a été aussi restructuré pour permettre une meilleure lisibilité. Il n'y a pas de changement "sur le réseau".

La Section 2 étend l'ABNF pour les commandes SELECT, EXAMINE, CREATE, RENAME, FETCH/UID FETCH, STORE/UID STORE, SEARCH, et APPEND d'une façon cohérente. Les extensions à toutes les commandes sauf APPEND ont la même structure. L'extensibilité pour la commande APPEND a été faite d'une façon légèrement différente afin de préserver la rétro compatibilité avec les extensions existantes.

La Section 2 définit aussi une nouvelle réponse ESEARCH, dont l'objet est de définir une meilleure version de la réponse SEARCH définie dans la RFC 3501.

La Section 3 définit l'ABNF collecté qui remplace des parties de l'ABNF dans les RFC sus-mentionnées. L'ABNF collecté a été généralisé pour faciliter une future extensibilité.

## 1.2 Conventions utilisées dans le document

Dans les exemples, "C:" et "S:" indiquent les lignes envoyées respectivement par le client et le serveur.

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. Extensions d'ABNF IMAP

Cette Section n'est pas normative. Elle donne les fondements de l'utilisation prévue des différentes extensions et donne des directives sur la façon dont les futures extensions devraient étendre les commandes décrites.

### 2.1 Paramètres facultatifs avec les commandes SELECT/EXAMINE

Le présent document ajoute la capacité d'inclure un ou plusieurs paramètres aux commandes IMAP SELECT (paragraphe 6.3.1 de la [RFC3501]) ou EXAMINE (paragraphe 6.3.2 de la [RFC3501]) pour activer ou désactiver certains comportements standard, ou pour ajouter de nouveaux comportements facultatifs requis pour une extension particulière.

Il y a deux modes de fonctionnement possibles :

- o Un changement d'état global où une seule utilisation du paramètre facultatif va affecter l'état de session à partir de ce moment, sans considération des commandes SELECT/EXAMINE suivantes.
- o Un changement d'état par boîte aux lettres qui va affecter la session pour la seule durée du nouvel état choisi. Une commande SELECT/EXAMINE suivante sans le paramètre facultatif va annuler son effet pour la nouvelle boîte aux lettres choisie.

Les paramètres facultatifs aux commandes SELECT ou EXAMINE sont ajoutés comme une liste entre parenthèses de paires de valeur/attribut, et apparaissent après le nom de boîte aux lettres dans la commande standard SELECT ou EXAMINE. L'ordre des paramètres individuels est arbitraire. Une valeur de paramètre est facultative et peut consister en atomes, chaînes, ou listes dans un ordre spécifique. Si la valeur du paramètre est présente, elle apparaît toujours entre parenthèses (\*). Tout paramètre non défini par les extensions que le serveur prend en charge doit être rejeté avec une réponse BAD.

Exemple :

C: a SELECT INBOX (ANNOTATE)  
 S: ...  
 S: a OK SELECT achevé

Dans l'exemple ci-dessus, un seul paramètre est utilisé avec la commande SELECT.

Exemple :

C: a EXAMINE INBOX (ANNOTATE RESPONSES ("UID Responses") CONDSTORE)  
 S: ...  
 S: a OK EXAMINE achevé

Dans l'exemple ci-dessus, trois paramètres sont utilisés avec la commande EXAMINE. Le second paramètre consiste en deux éléments : un "RESPONSES" atomique suivi par une chaîne entre guillemets.

Exemple :

C: a SELECT INBOX (BLURDYBLOOP)  
 S: a BAD paramètre inconnu dans la commande SELECT

Dans l'exemple ci-dessus, un paramètre non pris en charge par le serveur est utilisé. Il en résulte la réponse BAD de la part du serveur.

(\*) - si un paramètre a une valeur obligatoire, qui peut toujours être représentée par un nombre ou un ensemble de séquences, la valeur de paramètre n'a pas besoin d'être enclose dans les parenthèses (). Voir les détails dans l'ABNF.

## 2.2 Extension de commande CREATE

Arguments : nom de boîte aux lettres  
 liste FACULTATIVE de paramètres CREATE

Réponses : pas de réponse spécifique pour cette commande

Résultat : OK - création achevée  
 NO - échec de création : ne peut pas créer de boîte aux lettres de ce nom  
 BAD - arguments invalides

Le présent document ajoute la capacité d'inclure un ou plusieurs paramètres avec la commande IMAP CREATE (voir au paragraphe 6.3.3 de la [RFC3501]) pour activer ou désactiver certains comportements standard, ou pour ajouter de nouveaux comportements facultatifs exigés pour une extension particulière. Aucun paramètre CREATE n'est défini dans le présent document.

Les paramètres facultatifs à la commande CREATE sont ajoutés comme une liste entre parenthèses de paires attribut/valeur après le nom de boîte aux lettres. L'ordre des paramètres individuels est arbitraire. Une valeur de paramètre est facultative et peut consister en atomes, chaînes, ou listes dans un ordre spécifique. Si la valeur de paramètre est présente, elle apparaît toujours entre parenthèses (\*). Tout paramètre non défini par des extensions que le serveur prend en charge doit être rejeté avec une réponse BAD.

(\*) - si un paramètre a une valeur obligatoire, qui peut toujours être représentée par un nombre ou un ensemble de séquences, la valeur de paramètre n'a pas besoin d'être entre parenthèses (). Voir les détails dans l'ABNF.

## 2.3 Extension de commande RENAME

Arguments : nom de boîte aux lettres existante  
 nouveau nom de boîte aux lettres  
 liste FACULTATIVE de paramètres RENAME

Réponses : pas de réponse spécifique pour cette commande.

Résultat : OK - changement de nom achevé  
 NO - échec de changement de nom : ne peut pas renommer la boîte aux lettres avec ce nom, etc.

## BAD - arguments invalides

Le présent document ajoute la capacité d'inclure un ou plusieurs paramètres avec la commande IMAP RENAME (voir le paragraphe 6.3.5 de la [RFC3501]) pour activer ou désactiver certains comportements standard, ou pour ajouter de nouveaux comportements facultatifs exigés pour une extension particulière. Aucun paramètre RENAME n'est défini dans le présent document.

Les paramètres facultatifs à la commande RENAME sont ajoutés comme une liste entre parenthèses de paires d'attribut/valeur après le nouveau nom de boîte aux lettres. L'ordre des paramètres individuels est arbitraire. Une valeur de paramètre est facultative et peut consister en atomes, chaînes, ou listes dans un ordre spécifique. Si la valeur de paramètre est présente, elle apparaît toujours entre parenthèses (\*). Tout paramètre non défini par des extensions que le serveur prend en charge doit être rejeté avec une réponse BAD.

(\*) - si un paramètre a une valeur obligatoire, qui peut toujours être représentée par un nombre ou un ensemble de séquences, la valeur de paramètre n'a pas besoin d'être entre parenthèses (). Voir les détails dans l'ABNF.

## 2.4 Extensions aux commandes FETCH et UID FETCH

Arguments : ensemble de séquences  
noms ou macro d'élément de données de message  
modificateurs FACULTATIFS de fetch

Réponses : réponses non étiquetées : FETCH

Résultat : OK - restitution achevée  
NO - erreur de restitution : ne peut pas aller chercher ces données  
BAD - commande inconnue ou arguments invalides

Le présent document étend la syntaxe des commandes FETCH et UID FETCH (voir au paragraphe 6.4.5 de la [RFC3501]) pour inclure des modificateurs facultatifs de FETCH. Aucun modificateur de fetch n'est défini dans le présent document.

L'ordre des modificateurs individuels est arbitraire. Chaque modificateur est une paire d'attribut/valeur. Une valeur de modificateur est facultative et peut consister en atomes et/ou chaînes et/ou listes dans un ordre spécifique. Si la valeur du modificateur est présente, elle apparaît toujours entre parenthèses (\*). Tout modificateur non défini par des extensions que le serveur prend en charge doit être rejeté avec une réponse BAD.

(\*) - si un modificateur a une valeur obligatoire, qui peut toujours être représentée par un nombre ou un ensemble de séquences, la valeur de modificateur n'a pas besoin d'être entre parenthèses (). Voir les détails dans l'ABNF.

## 2.5 Extensions aux commandes STORE et UID STORE

Arguments : ensemble de messages  
modificateurs FACULTATIFS de mémorisation  
nom d'élément de données de message  
valeur pour l'élément de données de message

Réponses : réponses non étiquetées : FETCH

Résultat : OK - mémorisation achevée  
NO - erreur de mémorisation : ne peut pas mémoriser ces données  
BAD - commande inconnue ou arguments invalides

Le présent document étend la syntaxe des commandes STORE et UID STORE (voir au paragraphe 6.4.6 de la [RFC3501]) pour inclure des modificateurs STORE facultatifs. Aucun modificateur STORE n'est défini dans le présent document.

L'ordre des modificateurs individuels est arbitraire. Chaque modificateur est une paire attribut/valeur. Une valeur de modificateur est facultative et peut consister en atomes et/ou chaînes et/ou listes dans un ordre spécifique. Si la valeur du modificateur est présente, elle apparaît toujours entre parenthèses (\*). Tout modificateur non défini par des extensions prises en charge par le serveur doit être rejetée avec une réponse BAD.

(\*) - si un modificateur a une valeur obligatoire, qui peut toujours être représentée par un nombre ou un ensemble de séquences, la valeur de modificateur n'a pas besoin d'être entre parenthèses (). Voir les détails dans l'ABNF.

## 2.6 Extensions à la commande SEARCH

### 2.6.1 Commande SEARCH étendue

Arguments : spécificateur de résultat FACULTATIF  
spécification de jeu de caractères FACULTATIF  
critères de recherche (un ou plusieurs)

Réponses : réponse non étiquetée EXIGÉE : SEARCH (\*)

Résultat : OK - recherche achevée  
NO - erreur de recherche : ne peut pas rechercher ce jeu de caractères ou critère  
BAD - commande inconnue ou arguments invalides

Ce paragraphe met à jour la définition de la commande SEARCH décrite au paragraphe 6.4.4 de la [RFC3501].

La commande SEARCH est étendue pour permettre des options de résultat. Le présent document ne définit aucune option de résultat.

L'ordre des options individuelles est arbitraire. Les options individuelles peuvent contenir des paramètres entre parenthèses (\*\*). Si une option a des paramètres, ils consistent en atomes et/ou chaînes et/ou listes dans un ordre spécifique. Toute option non définie par des extensions que le serveur prend en charge doit être rejetée avec une réponse BAD.

(\*) - Une extension à la commande SEARCH peut exiger un autre réponse non étiquetée, ou qu'une réponse non étiquetée soit retournée. Le paragraphe 2.6.2 définit une nouvelle réponse ESEARCH non étiquetée qui remplace la réponse SEARCH non étiquetée. Noter que pour une certaine commande SEARCH étendue, les réponses SEARCH et ESEARCH DEVRAIENT être mutuellement exclusives, c'est-à-dire, une seule d'entre elles devrait être retournée.

(\*\*) - si une option a un paramètre obligatoire, qui peut toujours être représentée comme un nombre ou un ensemble de séquences, le paramètre d'option n'a pas besoin des parenthèses (). Voir les détails dans l'ABNF.

### 2.6.2 Réponse ESEARCH non étiquetée

Contenu : une ou plusieurs paires de recherche-données de retour

La réponse ESEARCH DEVRAIT être envoyée comme résultat d'une commande étendue SEARCH ou UID SEARCH spécifiée au paragraphe 2.6.1.

La réponse ESEARCH commence avec un corrélateur facultatif de recherche. Si il manque, la réponse n'a alors pas été causée par une commande IMAP particulière, tandis que si il est présent, il contient l'étiquette de la commande qui a causé le retour de la réponse.

Le corrélateur de recherche est suivi par un indicateur UID facultatif. Si cet indicateur est présent, toutes les données dans la réponse ESEARCH se réfèrent aux UID, autrement toutes les données retournées se réfèrent aux numéros de message.

Le reste de la réponse ESEARCH contient une ou plusieurs paires de données de recherche. Chaque paire commence par le nom d'élément unique de retour, suivi d'une espace et des données correspondantes. Les paires de données de recherche peuvent être retournées dans n'importe quel ordre. Sauf mention contraire par une extension, tout nom d'élément de retour DEVRAIT apparaître une seule fois dans une réponse ESEARCH.

Exemple : S: \* ESEARCH UID COUNT 5 ALL 4:19,21,28

Exemple : S: \* ESEARCH (TAG "a567") UID COUNT 5 ALL 4:19,21,28

Exemple : S: \* ESEARCH COUNT 5 ALL 1:17,21

## 2.7 Extensions à la commande APPEND

L'extension IMAP BINARY [RFC3516] étend la commande APPEND pour permettre à un client d'ajouter des données contenant des NUL en utilisant la syntaxe <literal8>. L'ABNF a été réécrit pour faciliter l'extensibilité par les extensions IMAP. Le présent document n'a spécifié aucun changement sémantique aux extension de la [RFC3516].

De plus, le "literal8" non terminal défini dans la [RFC3516] a été étendu pour permettre les littéraux non synchronisants si les extensions de la [RFC3516] et de la [RFC2088] sont prises en charge par le serveur.

L'extension IMAP MULTIAPPEND [RFC3502] étend la commande APPEND pour permettre à un client d'ajouter plusieurs messages un par un. Le présent document définit une syntaxe commune pour la commande APPEND qui prend en compte les extensions de syntaxe définies dans les extensions de la [RFC3516] et la [RFC3502].

## 3. Syntaxe formelle

La spécification de syntaxe suivante utilise la notation de forme Backus-Naur augmentée (ABNF) comme spécifié dans la [RFC4234].

Les non terminaux référencés mais non définis ci-dessous sont définis dans la [RFC3501].

Sauf notation contraire, tous les caractères alphabétiques sont insensibles à la casse. L'utilisation de caractères majuscules ou minuscules pour définir des chaînes de jetons est seulement pour rendre la lecture plus claire. Les mises en œuvre DOIVENT accepter ces chaînes de façon insensible à la casse.

append = "APPEND" SP mailbox 1\*append-message  
 ;; un seul message append peut apparaître si la capacité MULTIAPPEND [RFC3502] n'est pas présente.

append-message = append-opts SP append-data

append-ext = append-ext-name SP append-ext-value  
 ;; Ce non terminal définit les extensions aux métadonnées de message.

append-ext-name = tagged-ext-label

append-ext-value= tagged-ext-val  
 ;; Ce non terminal montre la syntaxe recommandée pour de futures extensions.

append-data = literal / literal8 / append-data-ext

append-data-ext = tagged-ext  
 ;; Ce non terminal montre la syntaxe recommandée pour de futures extensions, c'est-à-dire une étiquette obligatoire suivie par des paramètres.

append-opts = [SP flag-list] [SP date-time] \*(SP append-ext)  
 ;; métadonnées de message.

charset = atom / quoted  
 ;; La syntaxe exacte est définie dans la [RFC2978].

create = "CREATE" SP mailbox [create-params]  
 ;; L'utilisation de INBOX donne une erreur NO.

create-params = SP "(" create-param \*( SP create-param) ")"

create-param-name = tagged-ext-label

create-param = create-param-name [SP create-param-value]

create-param-value= tagged-ext-val

;; Ce non terminal montre la syntaxe recommandée pour de futures extensions.

esearch-response = "ESEARCH" [search-correlator] [SP "UID"] \*(SP search-return-data)

;; Noter que les réponses SEARCH et ESEARCH DEVRAIENT être mutuellement exclusives, c'est-à-dire que un seul de ces types de réponse devrait être retourné comme résultat d'une commande.

examine = "EXAMINE" SP mailbox [select-params]

;; Modifie la commande originale IMAP EXAMINE pour accepter des paramètres facultatifs.

fetch = "FETCH" SP sequence-set SP ("ALL" / "FULL" / "FAST" / fetch-att / "(" fetch-att \*(SP fetch-att) ")") [fetch-modifiers]

;; Modifie la commande originale IMAP4 FETCH pour accepter des modificateurs facultatifs.

fetch-modifiers = SP "(" fetch-modifier \*(SP fetch-modifier) ")"

fetch-modifier = fetch-modifier-name [ SP fetch-modif-params ]

fetch-modif-params = tagged-ext-val

;; Ce non terminal montre la syntaxe recommandée pour de futures extensions.

fetch-modifier-name = tagged-ext-label

literal8 = "~{" number ["+"] "}" CRLF \*OCTET

;; Chaîne qui peut contenir des NUL.

;; <number> représente le nombre d'octets dans la chaîne de réponse.

;; Le "+" n'est permis que quand les deux extensions LITERAL+ et BINARY sont prises en charge par le serveur.

mailbox-data = / Namespace-Response / esearch-response

Namespace = nil / "(" 1\*Namespace-Descr ")"

Namespace-Command = "NAMESPACE"

Namespace-Descr = "(" string SP (DQUOTE QUOTED-CHAR DQUOTE / nil) \*(Namespace-Response-Extension) ")"

Namespace-Response-Extension = SP string SP "(" string \*(SP string) ")"

Namespace-Response = "NAMESPACE" SP Namespace SP Namespace SP Namespace

;; Cette réponse n'est actuellement permise que si le serveur IMAP prend en charge la [RFC2342].

;; Le premier espace de noms est l'espace Personnel

;; Le second espace de noms est l'espace Autres utilisateurs

;; Le troisième espace de noms est l'espace Partagé.

rename = "RENAME" SP mailbox SP mailbox [rename-params]

;; L'utilisation de INBOX comme destination donne une erreur NO, sauf si rename-params n'est pas vide.

rename-params = SP "(" rename-param \*( SP rename-param) ")"

rename-param = rename-param-name [SP rename-param-value]

rename-param-name = tagged-ext-label

rename-param-value = tagged-ext-val

;; Ce non terminal montre la syntaxe recommandée pour de futures extensions.

response-data = "\*" SP response-payload CRLF

response-payload = resp-cond-state / resp-cond-by / mailbox-data / message-data / capability-data

search = "SEARCH" [search-return-opts] SP search-program

search-correlator = SP "(" "TAG" SP tag-string ")"

search-program = ["CHARSET" SP charset SP] search-key \*(SP search-key)  
 ;; L'argument CHARSET à SEARCH DOIT être enregistré auprès de l'IANA.

search-return-data = search-modifier-name SP search-return-value  
 ;; Noter que toute option de retour de SEARCH n'est pas obligée d'avoir les données correspondantes de retour de ESEARCH.

search-return-opts = SP "RETURN" SP "(" [search-return-opt \*(SP search-return-opt)] ")"

search-return-opt = search-modifier-name [SP search-mod-params]

search-return-value = tagged-ext-val  
 ;; Données pour l'option de recherche retournée. Une seule valeur de "nz-number"/"number" peut être retournée comme atome (c'est-à-dire, sans guillemets). Un sequence-set peut aussi être retourné comme atome.

search-modifier-name = tagged-ext-label

search-mod-params = tagged-ext-val  
 ;; Ce non terminal montre la syntaxe recommandée pour de futures extensions.

select = "SELECT" SP mailbox [select-params]  
 ;; Modifie la commande originale IMAP SELECT pour accepter des paramètres facultatifs.

select-params = SP "(" select-param \*(SP select-param) ")"

select-param = select-param-name [SP select-param-value]  
 ;; Un paramètre pour SELECT peut contenir un ou plusieurs atomes et/ou chaînes et/ou listes.

select-param-name= tagged-ext-label

select-param-value= tagged-ext-val  
 ;; Ce non terminal montre la syntaxe recommandée pour de futures extensions.

status-att-list = status-att-val \*(SP status-att-val)  
 ;; Redéfinit status-att-list de la RFC 3501.  
 ;; status-att-val est défini dans l'errata à la RFC 3501.

status-att-val = ("MESSAGES" SP number) / ("RECENT" SP number) / ("UIDNEXT" SP nz-number) /  
 ("UIDVALIDITY" SP nz-number) / ("UNSEEN" SP number)  
 ;; Les extensions aux réponses STATUS devraient étendre cette production.  
 ;; Les extensions devraient utiliser la syntaxe générique définie par tagged-ext.

store = "STORE" SP sequence-set [store-modifiers] SP store-att-flags  
 ;; Étend la syntaxe de la commande STORE [RFC3501] pour permettre des modificateurs de mémorisation facultatifs.

store-modifiers = SP "(" store-modifier \*(SP store-modifier) ")"

store-modifier = store-modifier-name [SP store-modif-params]

store-modif-params = tagged-ext-val  
 ;; Ce non terminal montre la syntaxe recommandée pour de futures extensions.

store-modifier-name = tagged-ext-label

tag-string = string  
 ;; Étiquette de la commande qui a causé la réponse ESEARCH, envoyée comme chaîne.

tagged-ext = tagged-ext-label SP tagged-ext-val  
 ;; Syntaxe de sur arcade recommandée pour les extensions.

tagged-ext-label = tagged-label-fchar \*tagged-label-char  
;; C'est un "atom" valide de la RFC 3501..

tagged-label-fchar = ALPHA / "-" / "\_" / "."

tagged-label-char = tagged-label-fchar / DIGIT / ":"

tagged-ext-comp = astring / tagged-ext-comp \*(SP tagged-ext-comp) / "(" tagged-ext-comp ")"

;; Les extensions qui suivent cette syntaxe générale devraient utiliser nstring au lieu de astring quand c'est approprié dans le contexte de l'extension. Noter qu'un ensemble de message ou un "number" peut toujours être représenté comme "atom". Un URL devrait être représenté comme une chaîne "entre guillemets".

tagged-ext-simple = sequence-set / number

tagged-ext-val = tagged-ext-simple / "(" [tagged-ext-comp] ")"

#### 4. Considérations sur la sécurité

Le présent document met à jour l'ABNF des RFC 2088, 2342, 3501, 3502, et 3516. Les documents mis à jour doivent être consultés sur les considérations de sécurité pour les extensions qu'ils définissent.

Lorsque un protocole devient plus complexe, les erreurs d'analyseurs deviennent plus courantes, y compris les débordements de mémoire tampon, les dénis de service, et autres erreurs de codage courantes. Dans la mesure où le présent document rend l'analyseur plus complexe, il empire cette situation. Dans la mesure où le présent document rend l'analyseur plus cohérent et donc plus simple, la situation est améliorée. L'impact va dépendre de combien d'extensions IMAP déployées sont cohérentes avec ce document. Les mises en œuvre sont invitées à prendre ces problèmes en compte lorsque elles incorporent ces extensions. Les futures extensions IMAP devraient s'efforcer à la cohérence et à la simplicité autant que possible.

Les extensions aux commandes IMAP qui sont permises dans l'état NON AUTHENTIFIÉ sont plus sensibles à ces problèmes de sécurité à cause de la grande communauté d'attaquants possible avant l'authentification, et au fait que certains serveurs IMAP fonctionnent avec des privilèges élevés dans cet état. Le présent document n'étend aucune commande permise dans l'état NON AUTHENTIFIÉ. De futures extensions IMAP aux commandes permises dans l'état NON AUTHENTIFIÉ devraient privilégier la simplicité sur la cohérence ou l'extensibilité.

#### 5. Références normatives

- [RFC2088] J. Myers, "[Littéraux IMAP4 sans synchronisation](#)", janvier 1997. (MàJ par [RFC4466](#)) (P.S. Remplacée par [RFC7888](#))
- [RFC2119] S. Bradner, "[Mots clés à utiliser](#) dans les RFC pour indiquer les niveaux d'exigence", BCP 14, mars 1997. (MàJ par [RFC8174](#))
- [RFC2342] M. Gahrns, C. Newman, "Espace de noms IMAP4", mai 1998. (MàJ par [RFC4466](#)) (P.S.)
- [RFC2978] N. Freed et J. Postel, "Procédures d'enregistrement des jeux de caractère par l'IANA", BCP 19, octobre 2000.
- [RFC3501] M. Crispin, "Protocole d'[accès au message Internet - version 4rev1](#)", mars 2003. (P.S. ; MàJ par [RFC4466](#), [4469](#), [4551](#), [5032](#), [5182](#), [7817](#), [8314](#), [8437](#), [8474](#))
- [RFC3502] M. Crispin, "Protocole d'accès au message Internet ([IMAP](#)) - [Extension MULTIAPPEND](#)", mars 2003. (MàJ par [RFC4466](#), [RFC4469](#)) (P.S.)
- [RFC3516] L. Nerenberg, "[Extension Contenu binaire à IMAP4](#)", avril 2003. (MàJ par [RFC4466](#)) (P.S.)
- [RFC4234] D. Crocker et P. Overell, "[BNF augmenté pour les spécifications de syntaxe : ABNF](#)", octobre 2005.

(Remplace RFC2234, remplacée par RFC5234)

## 6. Remerciements

Le présent document se fonde sur des idées proposées par Pete Resnick, Mark Crispin, Ken Murchison, Philip Guenther, Randall Gellens, et Lyndon Nerenberg. Cependant, toutes les erreurs et omissions doivent être attribuées aux auteurs de ce document. Merci à Philip Guenther, Dave Cridland, Mark Crispin, Chris Newman, Elwyn Davies, et Barry Leiba pour leurs commentaires et corrections. La syntaxe de literal8 est tirée de la RFC 3516.

### Adresse des auteurs

Alexey Melnikov  
Isode Limited  
5 Castle Business Village  
36 Station Road  
Hampton, Middlesex, TW12 2BX  
UK  
mél : [Alexey.Melnikov@isode.com](mailto:Alexey.Melnikov@isode.com)

Cyrus Daboo  
mél : [cyrus@daboo.name](mailto:cyrus@daboo.name)

### Déclaration complète de droits de reproduction

Copyright (C) The IETF Trust (2006).

Le présent document est soumis aux droits, licences et restrictions contenus dans le BCP 78, et à [www.rfc-editor.org](http://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 encloses ne viole 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 pourrait 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 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](mailto:ietf-ipr@ietf.org).

### Remerciement

Le financement de la fonction d'édition des RFC est fourni par l'activité de soutien administratif (IASA) de l'IETF.