

Groupe de travail Réseau
Request for Comments : 2389
 Voir aussi la RFC 959
 Catégorie : En cours de normalisation

P. Hethmon, Hethmon Brothers
 R. Elz, University of Melbourne
 août 1998
 Traduction Claude Brière de L'Isle

Mécanisme de négociation de caractéristiques pour le protocole de transfert de fichiers

Statut du présent mémoire

Le présent document spécifie un protocole en cours de normalisation de l'Internet 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 "Normes officielles des protocoles de l'Internet" (STD 1) pour connaître l'état de la 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 (1998). Tous droits réservés.

Résumé

De temps en temps, le protocole de transfert de fichiers est étendu par de nouvelles commandes, ou de nouvelles facilités. Les mises en œuvre du protocole FTP ne peuvent pas être supposées mettre toutes immédiatement en œuvre tous les nouveaux mécanismes définis. Le présent document fournit un mécanisme par lequel les clients du protocole FTP peuvent découvrir quelles nouvelles caractéristiques sont prises en charge par un serveur FTP particulier.

Table des Matières

1. Introduction.....	1
2. Conventions du document.....	1
2.1 Jetons de base.....	2
2.2 Réponses du serveur.....	2
3. Connaissance de capacités supplémentaires – la commande FEAT.....	2
3.1 Syntaxe de la commande Feature (FEAT).....	2
3.2 Réponses à la commande FEAT.....	2
3.3 Raison de FEAT.....	4
4. Commande OPTS.....	4
5. Considérations pour la sécurité.....	4
6. Références.....	5
Remerciements.....	5
Adresse des éditeurs.....	5
Déclaration complète de droits de reproduction.....	5

1. Introduction

Le présent document amende le protocole de transfert de fichier (FTP, *File Transfer Protocol*) [RFC0959]. Deux nouvelles commandes sont ajoutées : "FEAT" et "OPTS".

Ces commandes permettent à un client de découvrir quelles options de commandes sont prises en charge par un serveur, et comment elles sont prises en charge, et de faire un choix entre diverses options que toute commande FTP peut prendre en charge.

2. Conventions du document

Le présent document utilise les conventions définies dans le BCP 14 [RFC2119]. Cela donne l'interprétation de certains mots en majuscules comme DOIT, DEVRAIT, etc.

Les termes définis dans la [RFC0959] seront utilisés ici comme ils y sont définis. Cela inclut ASCII, réponse, processus FTP de serveur, processus FTP d'utilisateur, serveur-PI, usager-PI, et usager.

La syntaxe requise est définie en utilisant le BNF augmenté défini dans la [RFC2234]. Certaines définitions générales d'ABNF sont nécessaires tout au long du présent document, et elles seront définies ici. En première lecture, il peut être sage de se rappeler simplement que ces définitions existent ici, et de passer à la section suivante.

2.1 Jetons de base

Le présent document importe les définitions données dans l'Appendice A de la [RFC2234]. Ces définitions vont se trouver pour des éléments d'ABNF de base comme ALPHA, DIGIT, VCHAR, SP, etc. À ceux là sont ajoutés les termes suivants qui sont utilisés dans le présent document.

TCHAR = VCHAR / SP / HTAB ; caractères visibles plus espace blanche

Le type TCHAR, et VCHAR de la [RFC2234], donnent des types de caractères de base tirés de sous-ensembles variables du jeu de caractères ASCII à utiliser dans diverses commandes et réponses.

réponse_d'erreur = code_d'erreur SP *TCHAR CRLF
code_d'erreur = ("4" / "5") 2DIGIT

Noter qu'en ABNF, les chaînes littérales sont insensibles à la casse. Cette convention est conservée dans le présent document. Noter cependant que ALPHA, en particulier, est sensible à la casse, comme le sont VCHAR et TCHAR.

2.2 Réponses du serveur

Le paragraphe 4.2 de la [RFC0959] définit le format et la signification des réponses par le serveur-PI aux commandes FTP provenant de l'utilisateur-PI. Ces conventions de réponse sont utilisées ici sans changement. Les mises en œuvre devraient noter que dans le présent document et les autres documents qui se rapportent à FTP, la syntaxe ABNF (qui n'était pas utilisée dans la [RFC0959]) montre parfois des réponses présentées sur une seule ligne. Sauf mention explicite contraire, ceci n'est pas destiné à impliquer que des réponses multilignes ne sont pas permises. Les mises en œuvre devraient supposer, sauf mention contraire, que toute réponse à une commande FTP quelconque (y compris QUIT) peut être du format multiligne décrit dans la [RFC0959].

Tout au long du présent document, les réponses seront identifiées par le code à trois chiffres qui est leur premier élément. Donc, le terme "500 Réponse" signifie une réponse provenant du serveur-PI qui utilise le code à trois chiffres "500".

3. Connaissance de capacités supplémentaires – la commande FEAT

On ne doit pas s'attendre à ce que tous les serveurs prennent nécessairement en charge toutes les nouvelles commandes définies dans tous les futurs amendements au protocole FTP. Afin de permettre aux clients de déterminer quelles nouvelles commandes sont prises en charge par un certain serveur, sans essayer toutes les commandes possibles, une nouvelle commande est ajoutée au répertoire des commandes FTP. Cette commande demande au serveur d'énumérer toutes les extensions de commandes, ou les extensions de mécanismes, qu'il prend en charge. C'est-à-dire que toutes les commandes et dispositifs définis et spécifiés qui ne sont pas définis dans la [RFC0959] ou dans le présent document doivent être inclus dans le résultat de la commande FEAT sous la forme spécifiée dans le document qui définit l'extension.

Les usagers-PI FTP doivent s'attendre à voir énumérés dans les réponses à la commande FEAT des dispositifs inconnus. Ce n'est pas une erreur, et cela indique simplement que la mise en œuvre de serveur FTP a vu et mis en œuvre la spécification d'un nouveau dispositif qui n'est pas connu de l'utilisateur FTP.

3.1 Syntaxe de la commande Feature (FEAT)

feat = "Feat" CRLF

La commande FEAT consiste seulement en le mot "FEAT". Elle n'a aucun paramètre ni argument.

3.2 Réponses à la commande FEAT

Lorsque un processus de serveur-FTP ne prend pas en charge la commande FEAT, il va répondre à la commande FEAT par une réponse 500 ou 502. C'est simplement la réponse normale "commande non reconnue" que déclencherait toute commande inconnue. Une erreur dans la syntaxe de commande, telle que de donner des paramètres, va résulter en une réponse 501.

Les processus de serveur-FTP qui reconnaissent la commande FEAT, mais ne mettent pas en œuvre de dispositifs étendus, et n'ont donc rien à rapporter, DEVRAIENT répondre avec le code 211 "pas de caractéristiques". Cependant, comme ce cas ne peut en pratique être distingué de celui d'un serveur-FTP qui ne reconnaît pas la commande FEAT, une réponse 500 ou 502 PEUT aussi être utilisée. La réponse "pas de caractéristiques" NE DOIT PAS utiliser le format de réponse multi-lignes : exactement une ligne de réponse est exigée et permise.

Les réponses à la commande FEAT DOIVENT se conformer à la syntaxe qui suit. Le texte sur la première ligne de la réponse est de forme libre ; il n'est pas interprété et n'a pas d'utilité pratique, car ce texte n'est pas supposé être révélé à l'utilisateur final. La syntaxe des autres lignes de réponse est définie avec précision, et si elles sont présentes, DOIT être exactement comme spécifié .

```
réponse_feat           = réponse_d_erreur / pas_de_caractéristiques / liste_des_caractéristiques
pas_de_caractéristiques = "211" SP *TCHAR CRLF
liste_des_caractéristiques = "211-" *TCHAR CRLF
                        1*( SP caractéristique CRLF )
                        "211 Fid" CRLF
caractéristique        = étiquette_de_caractéristique [ SP paramètres_de_caractéristique ]
étiquette_de_caractéristique = 1*VCHAR
paramètres_de_caractéristique = 1*TCHAR
```

Noter que chaque ligne de caractéristiques dans liste_des_caractéristiques commence par une seule espace. Cette espace n'est pas facultative, et elle n'indique pas une espace blanche générale. Cette espace garantit que la ligne de caractéristique ne peut jamais être mal interprétée comme fin de liste_des_caractéristiques, mais elle est exigée même lorsque il n'y a aucune possibilité d'ambiguïté.

Chaque extension prise en charge doit être énumérée sur une ligne séparée pour faciliter la possible inclusion des paramètres pris en charge par chaque commande d'extension. L'étiquette_de_caractéristique à utiliser dans la réponse à la commande FEAT sera spécifiée lorsque chaque nouvelle caractéristique est ajoutée à l'ensemble de commandes FTP. Ce sera souvent le nom d'une nouvelle commande ajoutée ; ceci n'est cependant pas exigé. En fait, il n'est pas exigé qu'une nouvelle caractéristique ajoute une nouvelle commande. Tous les paramètres inclus sont à spécifier avec la définition de la commande concernée. Cette spécification devra aussi spécifier comment sont à interpréter tous les paramètres présents.

L »étiquette_de_caractéristique et les paramètres_de_caractéristique sont nominalement sensibles à la casse, cependant, les définitions d'étiquettes et de paramètres spécifiques spécifient leur interprétation précise, et on s'attend à ce que ces définitions spécifient normalement l'étiquette et les paramètres d'une façon indépendante de la casse. Lorsque ceci est fait, il est recommandé aux mises en œuvre d'utiliser des lettres majuscules lors de la transmission de la réponse aux caractéristiques.

La commande FEAT elle-même n'est pas incluse dans la liste des caractéristiques prises en charge ; la prise en charge de la commande FEAT est indiquée par le retour d'une réponse autre que 500 ou 502.

Un exemple de réponse normale à la commande FEAT pourrait être une réponse multi ligne de la forme suivante :

```
C> feat
S> 211-Extensions prises en charge :
S> MLST size*;create;modify*;perm;media-type
S> SIZE
S> COMPRESSION
S> MDTM
S> 211 END
```

Les extensions particulières montrées ici sont de simples exemples de ce qui peut être défini en d'autres endroits ; aucune signification particulière ne devrait leur être attribuée. On se rappellera aussi que les noms des caractéristiques retournées ne sont pas des noms de commandes, par eux-mêmes, mais simplement l'indication que le serveur possède certains attributs ou d'autres.

L'ordre dans lequel les caractéristiques sont retournées n'a aucune importance, les processus de serveur-FTP ne sont pas obligés de mettre en œuvre un ordre quelconque, ou même de retourner dans le même ordre lorsque la commande est répétée.

Les mises en œuvre de FTP qui prennent en charge FEAT DOIVENT inclure dans la réponse à la commande FEAT toutes les extensions FTP, documentées de façon appropriée au delà des commandes et mécanismes décrits dans la [RFC0959], y compris toutes celles qui existaient avant FEAT. C'est-à-dire que lorsque un client reçoit une réponse FEAT d'un serveur

FTP, il peut supposer que les seules extensions que prend en charge le serveur sont celles qui sont énumérées dans la réponse FEAT.

Les processus d'utilisateur FTP devraient cependant être conscients que plusieurs extensions FTP ont été développées, et largement utilisées, avant l'adoption du présent document et de la commande FEAT. L'effet en est qu'une réponse d'erreur à la commande FEAT n'implique pas nécessairement que ces extensions ne sont pas prises en charge par le processus de serveur FTP. Les utilisateurs-PI devraient vérifier de telles extensions individuellement si une réponse d'erreur a été reçue à la commande FEAT.

3.3 Raison de FEAT

Bien qu'il ne soit pas absolument nécessaire, un mécanisme standard pour que le serveur-PI informe l'utilisateur-PI de toutes les caractéristiques et extensions prises en charge va aider à réduire le trafic non indispensable entre l'utilisateur-PI et le serveur-PI car plus d'extensions pourraient être introduites à l'avenir. Si aucun mécanisme n'existait pour cela, un processus d'utilisateur FTP devrait essayer chaque extension tour à tour, ce qui résulterait en une série d'échanges entre l'utilisateur-PI et le serveur-PI. À part d'être un possible gaspillage d'énergie, cette procédure ne peut pas toujours être possible, car produire une commande juste pour déterminer si elle est prise en charge ou non peut avoir des effets indésirables.

4. Commande OPTS

La commande OPTS (options) permet à un utilisateur-PI de spécifier le comportement désiré d'un processus de serveur FTP lorsque une autre commande FTP (la commande cible) est ensuite produite. Le comportement et la syntaxe exacts vont varier selon la commande cible indiquée, et seront spécifiés avec la définition de cette commande. Lorsque aucun comportement OPTS n'est défini pour une commande particulière, il n'y a pas d'option disponible pour cette commande.

Syntaxe de demande :

```
opts                = opts-cmd SP nom_de_commande [ SP options_de_commande ] CRLF
opts-cmd            = "opts"
nom_de_commande    = <toute commande FTP qui permet l'établissement d'option>
options_de_commande = <format spécifié par la commande FTP individuelle>
```

Syntaxe de réponse :

```
opts-response      = opts-good / opts-bad
opts-good          = "200" SP message_de_reponse CRLF
opts-bad           = "451" SP message_de_reponse CRLF / "501" SP message_de_reponse CRLF
message_de_reponse = *TCHAR
```

Une réponse "opts-good" (réponse 200) DOIT être envoyée lorsque le nom de commande spécifié dans la commande OPTS est reconnu, et que les options de commande, si il en est, sont reconnues, et appropriées. Une réponse "opts-bad" est envoyée dans les autres cas. Une réponse 501 est appropriée pour toute erreur permanente. C'est-à-dire dans tous les cas où la simple répétition de la commande ultérieurement, sans autre changement d'état, sera aussi une erreur. Une réponse 451 devrait être envoyée lorsque une condition temporaire chez le serveur, sans relation avec l'état de communications entre usager et serveur, empêche la commande d'être acceptée lorsque elle est produite, mais si elle est répétée ultérieurement, un environnement modifié pour le processus serveur FTP peut permettre à la commande de réussir. Si la commande OPTS elle-même n'est pas reconnue, il va bien sûr en résulter une réponse 500 ou 502.

La commande OPTS DOIT être mise en œuvre chaque fois que la commande FEAT est mise en œuvre. À cause de cela, il n'y a pas d'indication dans la liste des caractéristiques retournées par FEAT pour indiquer que la commande OPTS elle-même est prise en charge. Ni la commande FEAT, ni la commande OPTS n'ont de fonctionnalité facultative, et donc il n'y a pas de commande "OPTS FEAT" ou "OPTS OPTS".

5. Considérations pour la sécurité

Aucun nouveau problème de sécurité significatif qui ne soit déjà présent dans le protocole FTP n'est estimé avoir été créé par la présente extension. Cependant, cette extension fournit bien un mécanisme par lequel on peut déterminer les capacités d'un serveur FTP, et à partir de cela des informations supplémentaires peuvent être déduites. Bien que les mêmes informations de base puissent être obtenues en essayant les diverses commandes sur le serveur, si la commande FEAT n'était pas fournie, cette méthode peut révéler une attaque en enregistrant les tentatives d'accès à diverses commandes

d'extension. Cette possibilité n'est pas considérée comme une menace suffisamment sérieuse pour qu'elle vaille la peine de définir une action pour y remédier.

La sécurité de toute caractéristique supplémentaire qui pourrait être rapportée par la commande FEAT, et manipulée par la commande OPTS, devrait être traitée lorsque ces caractéristiques seront définies.

6. Références

[RFC0959] J. Postel et J. Reynolds, "Protocole de [transfert de fichiers](#) (FTP)", STD 9, octobre 1985.

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

[RFC2234] D. Crocker et P. Overell, "[BNF augmenté pour les spécifications](#) de syntaxe : ABNF", novembre 1997.
(*Obsolète, voir [RFC5234](#)*)

Remerciements

La présente extension de protocole a été développée dans le groupe de travail FTPEXT de l'IETF, et les membres de ce groupe sont tous reconnus comme ses créateurs.

Adresse des éditeurs

Paul Hethmon
Hethmon Brothers
2305 Chukar Road
Knoxville, TN 37923
USA
téléphone : +1 423 690 8990
mél : phethmon@hethmon.com

Robert Elz
University of Melbourne
Department of Computer Science
Parkville, Vic 3052
Australia
mél : kre@munnari.OZ.AU

Déclaration complète de droits de reproduction

Copyright (C) The Internet Society (1998). Tous droits réservés.

Ce document et les traductions de celui-ci peuvent être copiés et diffusés, et les travaux dérivés qui commentent ou expliquent autrement ou aident à sa mise en œuvre peuvent être préparés, copiés, publiés et distribués, partiellement ou en totalité, sans restriction d'aucune sorte, à condition que l'avis de droits de reproduction ci-dessus et ce paragraphe soient inclus sur toutes ces copies et œuvres dérivées. Toutefois, ce document lui-même ne peut être modifié en aucune façon, par exemple en supprimant le droit d'auteur ou les références à l'Internet Society ou d'autres organisations Internet, sauf si c'est nécessaire à l'élaboration des normes Internet, auquel cas les procédures pour les droits de reproduction définies dans les procédures des normes d'Internet doivent être suivies, ou si nécessaire pour le traduire dans des langues autres que l'anglais.

Les permissions limitées accordées ci-dessus sont perpétuelles et ne seront pas révoquées par la Société Internet ou ses successeurs ou ayants droit.

Ce document et les renseignements qu'il contient sont fournis "TELS QUELS" et l'INTERNET SOCIETY et l'INTERNET ENGINEERING TASK FORCE déclinent toute garantie, expresse ou implicite, y compris mais sans s'y limiter, toute garantie que l'utilisation de l'information ici présente n'enfreindra aucun droit ou aucune garantie implicite de commercialisation ou d'adaptation à un objet particulier.

Remerciement

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