Groupe de travail Réseau
Request for Comments : 4643
RFC mise à jour : 2980
Catégorie: Sur la voie de la normalisation

J. Vinocur, Cornell University
K. Murchison, Carnegie Mellon University
octobre 2006
Traduction Claude Brière de L'Isle

Extension d'authentification au protocole de transfert des nouvelles du réseau (NNTP)

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.

Résumé
Le présent document définit une extension au protocole de transfert des nouvelles du réseau (NNTP, *Network News Transfer Protocol*) qui permet à un client d'indiquer un mécanisme d'authentification au serveur, pour effectuer un échange de protocole d'authentification, et facultativement de négocier une couche de sécurité pour les interactions de protocole suivantes durant le reste d'une session NNTP.

Le présent document met à jour et formalise la méthode d'authentification AUTHINFO USER/PASS spécifiée dans la RFC 2980 et déconseille les méthodes d'authentification AUTHINFO SIMPLE et AUTHINFO GENERIC. De plus, le présent document définit un profil du protocole de simple authentification et couche de sécurité (SASL, *Simple Authentication and Security Layer*) pour NNTP.

Table des matières

1. Introduction

Bien que NNTP [RFC3977] ait traditionnellement été utilisé pour fournir un accès public aux groupes de nouvelles, l'authentification est souvent utile pour plusieurs objets ; par exemple, pour contrôler la consommation des ressources, pour permettre d'identifier ceux qui abusent de la commande POST, et pour restreindre l'accès aux groupes de nouvelles "locaux".

Les commandes ad-hoc AUTHINFO USER et AUTHINFO PASS, documentées dans la [RFC2980], fournissent un mécanisme d'authentification très faible très utilisé par la base installée. Du fait de leur ubiquité, elles sont formalisées dans la présente spécification mais (à cause de leur insécurité) seulement pour être utilisées en combinaison avec les couches de sécurité appropriées.

La commande ad hoc AUTHINFO GENERIC, aussi documentée dans la [RFC2980] mais beaucoup moins répandue, fournissait un équivalent spécifique de NNTP de la facilité générique SASL [RFC4422]. Le présent document déconseille AUTHINFO GENERIC en faveur d'un remplacement par AUTHINFO SASL de façon que NNTP puisse bénéficier des mécanismes d'authentification développés pour les autres protocoles d'application à capacité SASL, incluant le protocole simple de transfert de messagerie (SMTP, *Simple Mail Transfer Protocol*) [RFC2554], le protocole Post Office (POP) [RFC1734], le protocole d'accès au message Internet (IMAP, *Internet Message Access Protocol*) [RFC3501], le protocole léger d'accès à un répertoire (LDAP, *Lightweight Directory Access Protocol*) [RFC4513], et le protocole extensible d'échange de blocs (BEEP, *Blocks Extensive Exchange Protocol*) [RFC3080].

La présente spécification est à lire en conjonction avec la spécification NNTP de base [RFC3977]. Sauf mention spécifique contraire, en cas de conflit entre ces deux documents, la [RFC3977] a la préséance sur celui-ci.

Il est aussi recommandé que la présente spécification soit lue en conjonction avec la spécification SASL de base [RFC4422].

1.1 Conventions utilisées dans le présent document

Les conventions de notation utilisées dans le présent document sont les mêmes que celles de la [RFC3977], et tout terme non défini dans le présent document a la même signification que dans celle-la.

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].

Les termes relatifs à l'authentification sont définis dans "Authentification sur l'Internet" [RFC1704].

Dans les exemples, les commandes du client sont indiquées par [C], et les réponses du serveur sont indiquées par [S].

2. Extension AUTHINFO

L'extension AUTHINFO est utilisée pour authentifier un utilisateur. Noter que l'autorisation est une affaire de politique de site, non de protocole réseau, et n'est donc pas discutée dans le présent document. Le serveur détermine l'autorisation de la manière définie par sa mise en œuvre telle que configurée par l'administrateur du site.

Cette extension fournit trois nouvelles commandes : AUTHINFO USER, AUTHINFO PASS, et AUTHINFO SASL. L'étiquette de capacité pour cette extension est AUTHINFO.

2.1 Annonce de l'extension AUTHINFO

Un serveur DOIT mettre en œuvre au moins une des commandes AUTHINFO USER ou AUTHINFO SASL afin d'annoncer l'étiquette de capacité "AUTHINFO" en réponse à la commande CAPABILITIES ([RFC3977] paragraphe 5.2). Cependant, cette capacité NE DOIT PAS être annoncée après une authentification réussie (voir le paragraphe 2.2). Cette capacité PEUT être annoncée avant et après toute utilisation de la commande MODE READER ([RFC3977] paragraphe 5.3) avec la même sémantique.

L'étiquette de capacité AUTHINFO contient une liste d'arguments qui précise quelles commandes d'authentification sont disponibles.

L'argument "USER" indique que AUTHINFO USER/PASS est pris en charge comme défini au paragraphe 2.3. L'argument "USER" NE DOIT PAS être annoncé, et les commandes AUTHINFO USER/PASS NE DEVRAIENT PAS être fournies, sauf si une forte couche de chiffrement (par exemple, la sécurité de la couche transport (TLS, *Transport Layer Security*) [RFC4642]) est utilisée ou si la rétro compatibilité l'impose.

L'argument "SASL" indique que AUTHINFO SASL est pris en charge comme défini au paragraphe 2.4. Si le serveur annonce l'argument "SASL", il DOIT alors aussi annoncer la capacité "SASL" en réponse à la commande CAPABILITIES. La capacité SASL est suivie d'une liste, séparée par des espaces de noms, de mécanismes SASL disponibles.

Le serveur PEUT faire la liste des capacités AUTHINFO sans argument, ce qui indique qu'il se conforme à la présente spécification et ne permet aucune commande d'authentification dans son état présent. Dans ce cas, le client NE DOIT PAS tenter d'utiliser de commandes AUTHINFO, même si il contient une logique qui pourrait autrement l'obliger à le faire (par exemple, pour la rétro compatibilité avec des serveurs qui ne se conforment pas à la présente spécification).

De futures extensions pourront ajouter des arguments supplémentaires à cette capacité. Les arguments non reconnus DOIVENT être ignorés par le client.

Comme la commande AUTHINFO se rapporte à la sécurité, on NE DOIT PAS s'appuyer sur des CAPABILITIES provenant d'une mise en antémémoire d'une session précédente, conformément au paragraphe 12.6 de la [RFC3977]. Cependant, un client PEUT utiliser de tels résultats en antémémoire afin de détecter des attaques actives de dégradation de négociation.

Exemple de capacités AUTHINFO avant et après l'utilisation de l'extension STARTTLS [RFC4642] :

[C] CAPABILITIES
[S] 101 Liste de capacités :
[S] VERSION 2
[S] READER
[S] IHAVE
[S] STARTTLS
[S] AUTHINFO SASL
[S] SASL CRAM-MD5 DIGEST-MD5 GSSAPI
[S] LIST ACTIVE NEWSGROUPS
[S] .
[C] STARTTLS
[S] 382 Continuer avec la négociation TLS
[la négociation TLS se fait, les commandes suivantes sont protégées par TLS]
[C] CAPABILITIES
[S] 101 Liste de capacités :
[S] VERSION 2
[S] READER
[S] IHAVE
[S] AUTHINFO USER SASL
[S] SASL CRAM-MD5 DIGEST-MD5 GSSAPI PLAIN EXTERNAL
[S] LIST ACTIVE NEWSGROUPS
[S] .

2.2 Authentification avec l'extension AUTHINFO

Un serveur NNTP réponds à une commande de client par une réponse 480 pour indiquer que le client DOIT s'authentifier et/ou être autorisé afin d'utiliser cette commande ou accéder à la ressource indiquée. L'utilisation de la commande AUTHINFO comme décrit ci-dessous est une des façons dont un client peut s'authentifier/autoriser auprès du serveur. Le client PEUT donc utiliser une commande AUTHINFO après avoir reçu une réponse 480. Un client qui a l'intention d'utiliser une commande AUTHINFO DEVRAIT produire la commande CAPABILITIES pour obtenir les commandes et mécanismes d'authentification disponibles avant de tenter l'authentification.

Si un serveur annonce la capacité AUTHINFO, un client PEUT tenter la première étape de l'authentification à tout moment

durant une session pour acquérir des privilèges supplémentaires sans avoir reçu de réponse 480. Les serveurs DEVRAIENT accepter de telles demandes d'authentification non sollicitées. Un serveur NE DOIT en aucune circonstance répondre à une commande AUTHINFO par une réponse 480.

Un client NE DOIT dans aucune circonstance continuer les étapes de l'authentification au-delà de la première, sauf si le code de réponse du serveur indique que l'échange d'authentification est le bienvenu. En particulier, tout code de réponse autre qu'un 38x indique que le client NE DOIT PAS continuer l'échange d'authentification.

Après une authentification réussie, le client NE DOIT PAS produire une autre commande AUTHINFO dans la même session. Un serveur NE DOIT PAS retourner la capacité AUTHINFO en réponse à une commande CAPABILITIES, et un serveur DOIT rejeter toutes les commandes AUTHINFO suivantes avec une réponse 502. De plus, le client NE DOIT PAS produire une commande MODE READER après l'authentification, et un serveur NE DOIT PAS annoncer la capacité MODE-READER.

En accord avec la [RFC4422], le serveur DOIT continuer d'annoncer la capacité SASL en réponse à une commande CAPABILITIES avec la même liste de mécanismes SASL qu'il a donnée avant l'authentification (permettant ainsi au client de détecter une possible attaque active de dégradation de la négociation). Les autres capacités retournées en réponse à une commande CAPABILITIES reçue après l'authentification PEUVENT être différentes de celles retournées avant l'authentification. Par exemple, un serveur NNTP peut ne pas vouloir annoncer la prise en charge d'une extension spécifique avant qu'un client ait été authentifié.

Noter qu'un serveur peut effectuer un échange d'authentification réussi avec un client et lui refuser quand même l'accès à certaines ou toutes les ressources ; la réponse 502 permanente indique qu'une ressource est indisponible même si l'authentification a été effectuée (ceci est différent de l'erreur temporaire 480, qui indique qu'une ressource est actuellement indisponible mais peut devenir disponible après l'authentification).

2.3 Commande AUTHINFO USER/PASS

Ce paragraphe se substitue à la définition des commandes AUTHINFO USER et AUTHINFO PASS comme documentées au paragraphe 3.1.1 de la [RFC2980].

2.3.1 Usage

Ces commandes NE DOIVENT PAS être traitées en parallèle.

Syntaxe
AUTHINFO USER nom d'utilisateur
AUTHINFO PASS mot de passe

Réponses
281 Authentification acceptée
381 Mot de passe exigé [1]
481 Échec/rejet d'authentification
482 Commandes d'authentification produites hors séquence
502 Commande indisponible [2]

[1] Seulement valide pour AUTHINFO USER. Noter qu'à la différence des codes 3xx traditionnels, qui indiquent que le client peut continuer la commande en cours, le code 381 traditionnel signifie que la commande AUTHINFO PASS doit être utilisée pour achever l'échange d'authentification.

[2] Si l'authentification est déjà faite, AUTHINFO USER/PASS ne sont pas des commandes valides (voir paragraphe 2.2).

Note : Malgré le paragraphe 3.2.1 de la [RFC3977], le serveur NE DOIT PAS retourner 480 en réponse à AUTHINFO USER/PASS.

Paramètres
username = chaîne identifiant l'utilisateur/client
password = chaîne représentant le mot de passe de l'utilisateur

2.3.2 Description

Les commandes AUTHINFO USER et AUTHINFO PASS sont utilisées pour présenter des accréditifs en clair au serveur. Ces accréditifs consistent en un nom d'utilisateur ou un nom d'utilisateur plus un mot de passe (la distinction est qu'un mot de passe est supposé être tenu secret, tandis qu'un nom d'utilisateur ne l'est pas ; cela n'affecte pas directement le protocole mais peut avoir un impact sur les interfaces d'utilisateur). Le nom d'utilisateur est fourni par la commande AUTHINFO USER, et le mot de passe par la commande AUTHINFO PASS.

Si le serveur exige seulement un nom d'utilisateur, il NE DOIT PAS donner une réponse 381 à AUTHINFO USER et DOIT donner une réponse 482 à AUTHINFO PASS.

Si le serveur exige à la fois le nom d'utilisateur et le mot de passe, le premier DOIT être envoyé avant le second. Le serveur devra mettre le nom d'utilisateur en antémémoire jusqu'à la réception du mot de passe ; il PEUT exiger que le mot de passe soit envoyé dans la commande qui suit immédiatement (en d'autres termes, de ne mettre le nom d'utilisateur en antémémoire que jusqu'à l'envoi de la prochaine commande). Le serveur :
- DOIT retourner une réponse 381 à AUTHINFO USER ;
- DOIT retourner une réponse 482 à AUTHINFO PASS si il n'y a pas de nom d'utilisateur en antémémoire ;
- DOIT utiliser l'argument de la plus récente AUTHINFO USER pour l'authentification ; et
- NE DOIT PAS retourner une réponse 381 à AUTHINFO PASS.

Le serveur PEUT déterminer si un mot de passe est nécessaire pour un certain nom d'utilisateur. Donc le même serveur peut répondre avec un 381 et d'autres codes de réponse à AUTHINFO USER.

Si le client présente avec succès les accréditifs appropriés, le serveur produit une réponse 281. Si le serveur est incapable d'authentifier le client, il DOIT rejeter la commande AUTHINFO USER/PASS avec une réponse 481. Si une commande AUTHINFO USER/PASS échoue, le client PEUT poursuivre sans authentification. Autrement, le client PEUT essayer un autre mécanisme d'authentification ou présenter des accréditifs différents en produisant une autre commande AUTHINFO.

La commande AUTHINFO PASS permet au client d'utiliser un mot de passe en clair pour s'authentifier. Une mise en œuvre conforme NE DOIT PAS utiliser cette commande sans aussi prendre en charge TLS [RFC4642]. L'utilisation de cette commande sans une couche de chiffrement active forte est déconseillée, car cela expose le mot de passe de l'utilisateur à toute partie sur le réseau entre le client et le serveur. Toute mise en œuvre de cette commande DEVRAIT être configurable pour la désactiver chaque fois qu'une forte couche de chiffrement (comme celle fournie par la [RFC4642]) n'est pas active, et cette configuration DEVRAIT être par défaut. Le serveur utilisera le code de réponse 483 pour indiquer que le flux de données n'est pas assez sûr pour la commande tentée (voir au paragraphe 3.2.1 de la [RFC3977]).

Noter qu'un serveur PEUT (sans y être obligé) permettre des caractères d'espace dans les noms d'utilisateur et les mots de passe. Une mise en œuvre de serveur PEUT aveuglément couper les arguments de commande aux espaces et peut donc ne pas préserver la séquence exacte de caractères d'espace dans le nom d'utilisateur ou mot de passe. Donc, un client DEVRAIT examiner le nom d'utilisateur et le mot de passe et, si il détecte des espaces, avertir l'utilisateur de la probabilité de problèmes. Le mécanisme SASL PLAIN [RFC4616] est recommandé comme solution de remplacement, car il ne souffre pas de ces problèmes.

Noter aussi que historiquement le nom d'utilisateur n'est en aucune façon canonisé. Les serveurs PEUVENT utiliser le profil [RFC4013] de l'algorithme de la [RFC3454] pour préparer les comparaisons de noms d'utilisateur, mais le faire peut causer des problèmes d'interopérabilité avec les mises en œuvre anciennes. Si la canonisation est désirée, le mécanisme SASL PLAIN [RFC4616] est recommandé comme solution de remplacement.

2.3.3 Exemples

Exemple de AUTHINFO USER réussi :

[C] AUTHINFO USER wilma
[S] 281 Authentification acceptée

Exemple de AUTHINFO USER/PASS réussi :

[C] AUTHINFO USER fred
[S] 381 Entrer le mot de passe
[C] AUTHINFO PASS flintstone
[S] 281 Authentification acceptée

Exemple de AUTHINFO USER/PASS exigeant une couche de sécurité :

[C] AUTHINFO USER fred@stonecanyon.example.com
[S] 483 Chiffrement ou authentification plus forte exigé

Exemple d'échec de AUTHINFO USER/PASS :

[C] AUTHINFO USER barney
[S] 381 Entrer le mot de passe
[C] AUTHINFO PASS flintstone
[S] 481 Échec d'authentification

Exemple de AUTHINFO PASS avant AUTHINFO USER :

[C] AUTHINFO PASS flintstone
[S] 482 Commandes d'authentification produites hors séquence

2.4 Commande AUTHINFO SASL

Ce paragraphe définit un profil formel de l'authentification simple et couche de sécurité (SASL) [RFC4422]. L'utilisation de la commande AUTHINFO GENERIC documentée au paragraphe 3.1.3 de la [RFC2980], comme moyen d'effectuer l'authentification SASL est déconseillé en faveur de la commande AUTHINFO SASL. Un serveur NE DEVRAIT PAS annoncer AUTHINFO GENERIC dans la liste de capacités retournée par CAPABILITIES.

2.4.1 Usage

Cette commande NE DOIT PAS être utilisée en parallèle.

Syntaxe : mécanisme AUTHINFO SASL [réponse initiale]

Cette commande PEUT excéder 512 octets. La longueur maximum de cette commande est augmentée de façon à ce qu'elle puisse s'accommoder le la plus grande réponse initiale codée possible pour tout mécanisme SASL pris en charge par la mise en œuvre.

Réponses :
281 : Authentification acceptée
283 challenge : Authentification acceptée (avec données de succès) [1]
383 challenge : Continuer avec l'échange SASL [1]
481 : Authentification en échec/rejetée
482 : Erreur de protocole SASL
502 : Commande indisponible [2]

[1] Ces réponses PEUVENT excéder 512 octets. La longueur maximum de ces réponses est augmentée de façon à ce qu'elles puissent s'accommoder du plus grand défi codé possible pour tout mécanisme SASL pris en charge par la mise en œuvre.

[2] Si l'authentification est déjà faite, AUTHINFO SASL n'est pas une commande valide (voir le paragraphe 2.2).

Note : Malgré le paragraphe 3.2.1 de la [RFC3977], le serveur NE DOIT PAS retourner 480 en réponse à AUTHINFO SASL.

Paramètres :
mécanisme = Chaîne identifiant un mécanisme d'authentification de la [RFC4422].
réponse initiale = Réponse initiale facultative du client. Si présente, la réponse DOIT être codée comme spécifé à la Section 4 de la [RFC4648]. [3]
challenge = défi du serveur. Le défi DOIT être codé comme spécifié à la Section 4 de la [RFC4648].

[3] Cet argument PEUT excéder 497 octets. La longueur maximum de cet argument est augmentée de façon à ce qu'il puisse s'accommoder de la plus grande réponse initiale codée possible pour tout mécanisme SASL pris en charge par la

mise en œuvre.

2.4.2 Description

La commande AUTHINFO SASL initie un échange de la [RFC4422] entre le client et le serveur. Le client identifie le mécanisme SASL à utiliser avec le premier paramètre de la commande AUTHINFO SASL. Si le serveur prend en charge le mécanisme d'authentification nécessaire, il effectue l'échange SASL pour authentifier l'utilisateur. Facultativement, il négocie aussi une couche de sécurité pour les interactions de protocole suivantes durant la session. Si le mécanisme d'authentification nécessaire est invalide (par exemple, n'est pas pris en charge) le serveur rejette la commande AUTHINFO SASL avec une réponse 503 (voir au paragraphe 3.2.1 de la [RFC3977]). Si le mécanisme d'authentification nécessaire exige une couche de chiffrement, le serveur rejette la commande AUTHINFO SASL avec une réponse 483 (voir le paragraphe 3.2.1 de la [RFC3977]).

Le nom de service spécifié par ce profil de protocole de SASL est "nntp".

L'échange SASL consiste en une série de défis du serveur et de réponses du client qui sont spécifiques du mécanisme choisi [RFC4422].

Un défi de serveur est envoyé comme une réponse 383 avec un seul argument contenant la chaîne codée selon la [RFC4648] fournie par le mécanisme SASL. Un défi de serveur d'une longueur zéro DOIT être envoyé comme un seul signe égal ("=") et DOIT être inclus (afin de se conformer à l'exigence de la [RFC3977] que les réponses aient toujours le même nombre d'arguments).

Une réponse de client consiste en une ligne contenant une chaîne codée selon la [RFC4648]. Une réponse de client d'une longueur de zéro DOIT être envoyée comme un seul signe égal ("=") et DOIT être incluse (pour la cohérence avec le format de défi de serveur). Si le client souhaite annuler l'échange d'authentification, il produit une ligne avec un seul "*". Si le serveur reçoit une telle réponse, il DOIT rejeter la commande AUTHINFO SASL en envoyant une réponse 481.

Noter que ces chaînes codées selon la [RFC4648] peuvent être beaucoup plus longues que les réponses NNTP normales. Clients et serveurs DOIVENT être capables de traiter la taille codée maximum des défis et réponses générés par les mécanismes d'authentification qu'ils prennent en charge. Cette exigence est indépendante de toutes limitations de longueur de ligne que le client ou le serveur peut avoir dans d'autres parties de sa mise en œuvre du protocole.

L'argument facultatif "réponse initiale" à la commande AUTHINFO SASL est utilisé pour économiser un aller-retour quand on utilise les mécanismes d'authentification qui prennent en charge la réponse initiale de client. Si l'argument "réponse initiale" est omis et si le mécanisme choisi exige une réponse initiale de client, le serveur DOIT procéder comme défini au paragraphe 5.1 de la [RFC4422]. Dans NNTP, un défi de serveur qui ne contient pas de données est équivalent à un défi de longueur zéro et est codé comme un seul signe égal ("=").

Noter que l'argument "réponse initiale" codé selon la [RFC4648] peut excéder 497 octets, et donc que la commande AUTHINFO SASL peut excéder 512 octets. Les clients DEVRAIENT et les serveurs DOIVENT être capables de traiter la taille codée maximum de réponse initiale possible pour leurs mécanismes d'authentification pris en charge. Cette exigence est indépendante de toute limitation de longueur de commande ou d'argument que peut avoir le client ou le serveur dans d'autres parties de sa mise en œuvre du protocole. Si l'utilisation de l'argument "réponse initiale" devrait faire que la commande AUTHINFO SASL excéde 512 octets, le client PEUT choisir d'omettre le paramètre "réponse initiale" (et à la place de procéder comme défini au paragraphe 5.1 de la [RFC4422]).

Si le client devait transmettre une réponse initiale de longueur zéro, il DOIT à la place transmettre la réponse comme un seul signe égal ("="). Cela indique que la réponse est présente, mais ne contient pas de données.

Si le client utilise un argument "réponse initiale" à la commande AUTHINFO SASL avec un mécanisme SASL qui ne prend pas en charge une réponse initiale de client, le serveur DOIT rejeter la commande AUTHINFO SASL avec une réponse 482.

Si le serveur ne peut pas décoder selon la [RFC4648] une réponse de client, il DOIT rejeter la commande AUTHINFO SASL avec une réponse 504 (voir au paragraphe 3.2.1 de la [RFC3977]). Si le client ne peut pas décoder de BASE64 les défis du serveur', il DOIT annuler l'authentification en utilisant la réponse "*". En particulier, les serveurs et clients DOIVENT rejeter (et non ignorer) tout caractère non explicitement permis par l'alphabet BASE64, et ils DOIVENT rejeter toute séquence de caractères BASE64 qui contient le caractère de bourrage ('=') partout ailleurs qu'à la fin de la chaîne (par exemple, "=AAA" et "AAA=BBB" ne sont pas permis).

L'identité d'autorisation générée par cet échange de la [RFC4422] est un simple nom d'utilisateur, et le client et le serveur DOIVENT tous deux utiliser le profil de la [RFC4013] de l'algorithme de la [RFC3454] pour préparer ces noms pour la transmission ou la comparaison. Si la préparation de l'identité d'autorisation échoue ou résulte en une chaîne vide (sauf si elle a été transmise comme chaîne vide) le serveur DOIT faire échouer l'authentification avec une réponse 481.

Si le client achève l'échange avec succès, le serveur produit une réponse 281 ou 283. Si le serveur est incapable d'authentifier le client, il DOIT rejeter la commande AUTHINFO SASL avec une réponse 481. Si une commande AUTHINFO SASL échoue, le client PEUT continuer sans authentification. Autrement, le client PEUT essayer un autre mécanisme d'authentification, ou présenter des accréditifs différents en produisant une autre commande AUTHINFO.

Si le mécanisme SASL retourne des données supplémentaires lors du succès (par exemple, authentification du serveur) le serveur NNTP produit une réponse 283 avec un seul argument contenant la chaîne codée en BASE64 fournie par le mécanisme SASL. Si aucune données supplémentaires ne sont retournées lors du succès, le serveur produit une réponse 281.

Si une couche de sécurité est négociée durant l'échange SASL, elle prend effet pour le client sur l'octet qui suit immédiatement le CRLF qui conclut la dernière réponse générée par le client. Pour le serveur, elle prend effet immédiatement à la suite du CRLF de sa réponse de succès.

Quand une couche de sécurité prend effet, le protocole NNTP est remis immédiatement à l'état après l'envoi de la réponse d'accueil initial (voir le paragraphe 5.1 de la [RFC3977]) avec l'exception que si une commande MODE READER a été produite, ses effets (si il en est) ne sont pas inversés. Le serveur DOIT éliminer toutes les informations obtenues du client, comme le groupe de nouvelles en cours et le numéro d'article, qui n'ont pas été obtenues de la négociation SASL elle-même. De même, le client DEVRAIT éliminer et NE DOIT PAS s'appuyer sur des informations obtenues du serveur, comme la liste de capacités, qui n'ont pas été obtenues de la négociation SASL elle-même. (Noter qu'un client PEUT comparer les mécanismes SASL annoncés avant et après l'authentification afin de détecter une attaque de dégradation active.)

Lorsque les deux couches de sécurité TLS [RFC4642] et SASL sont en effet, le codage TLS DOIT être appliqué après le codage SASL (les données en clair sont toujours codées d'abord avec SASL, et ensuite les données résultantes sont codées avec TLS).

Pour assurer l'interopérabilité, les mises en œuvre de client et de serveur de la présente extension DOIVENT mettre en œuvre le mécanisme SASL de la [RFC2831].

Si AUTHINFO USER/PASS et AUTHINFO SASL sont tous deux mis en œuvre, le mécanisme SASL de la [RFC4616] DEVRAIT aussi être mis en œuvre, car la fonctionnalité de DIGEST-MD5 est insuffisante dans certains environnements (par exemple, le serveur peut avoir besoin de passer le mot de passe en clair à un service d'authentification externe). Le mécanisme SASL PLAIN est préféré à AUTHINFO USER, même si il n'y a pas une forte couche de chiffrement active, parce que il élimine les limitations qu'a AUTHINFO USER/PASS à l'égard de l'utilisation des caractères d'espace dans les noms d'utilisateurs et mots de passe.

2.4.3 Exemples

Exemple du mécanisme SASL de la [RFC4616] sous une couche TLS, en utilisant une réponse initiale de client :

[C] CAPABILITIES
[S] 101 Liste de capacités :
[S] VERSION 2
[S] READER
[S] STARTTLS
[S] AUTHINFO SASL
[S] SASL CRAM-MD5 DIGEST-MD5 GSSAPI
[S] LIST ACTIVE NEWSGROUPS
[S] .
[C] STARTTLS
[S] 382 Continuer avec la négociation TLS
[La négociation TLS s'effectue, les commandes suivantes sont protégées par TLS]
[C] CAPABILITIES
[S] 101 Liste de capacités :
[S] VERSION 2

[S] READER
[S] AUTHINFO USER SASL
[S] SASL CRAM-MD5 DIGEST-MD5 GSSAPI PLAIN EXTERNAL
[S] LIST ACTIVE NEWSGROUPS
[S] .
[C] AUTHINFO SASL PLAIN AHRlc3QAMTIzNA==
[S] 281 Authentification acceptée

Exemple du mécanisme EXTERNAL SASL sous une couche TLS, en utilisant l'identité d'autorisation déduite des certificats TLS du client, et donc une réponse initiale de client de longueur zéro (les commandes avant le AUTHINFO SASL sont les mêmes que dans l'exemple précédent et ont été omises) :

[C] AUTHINFO SASL EXTERNAL =
[S] 281 Authentification acceptée

Exemple du mécanisme SASL de la [RFC2831], qui inclut un défi de serveur et des données de succès du serveur (une espace blanche a été insérée pour la clarté ; les données codées en base64 sont en fait envoyées comme une seule ligne sans espace blanche incorporée) :

[C] AUTHINFO SASL DIGEST-MD5
[S] 383
bm9uY2U9InNheUFPaENFS0dJZFBNSEMwd3RsZUxxT0ljT0kyd1FZSWU0enplQXR1aVE9IixyZWFsbT0iZWFnbGUu
b2NlYW5hLmNvbSIscW9wPSJhdXRoLGF1dGgtaW50LGF1dGgtY29uZiIsY2lwaGVyPSJyYzQtNDAscmM0LTU2LHJj
NCxkZXMsM2RlcyIsbWF4YnVmPTQwOTYsY2hhcnNldD11dGYtOCxhbGdvcml0aG09bWQ1LXNlc3M=

[C]
dXNlcm5hbWU9InRlc3QiLHJlYWxtPSJlYWdsZS5vY2VhbmEuY29tIixub25jZT0ic2F5QU9oQ0VLR0lkUE1IQzB3dGxl
THFPSWNPSTJ3UVlJZTR6emVBdHVpUT0iLGNub25jZT0iMFkzSlFWMlRnOVNjRGlwK08xU1ZDMHJoVmcvLytkb
k9JaUd6LzdDZU5KOD0iLG5jPTAwMDAwMDAxLHFvcD1hdXRoLWNvbmYsY2lwaGVyPXJjNCxtYXhidWY9MTA
yNCxkaWdlc3QtdXJpPSJubnRwL2xvY2FsaG9zdCIscmVzcG9uc2U9ZDQzY2Y2NmNmZmE5MDNmOWViMDM1Nm
MwOGEzZGIwZjI=
[S] 283 cnNwYXV0aD1kZTJlMTI3ZTVhODFjZGE1M2Q5N2FjZGEzNWNkZTgzYQ==

Exemple d'échec d'authentification due à de mauvais accréditifs [RFC5802]. Noter que bien que le mécanisme puisse utiliser la réponse initiale, le client choisit de ne pas l'utiliser à cause de sa longueur, résultant en un défi de serveur de longueur zéro (ici, une espace a été insérée pour la clarté ; les données codées en base64 sont en fait envoyées comme une seule ligne sans espace incorporée) :

[C] AUTHINFO SASL GSSAPI
[S] 383 =
[C]
YIICOAYJKoZIhvcSAQICAQBuggInMIICI6ADAgEFoQMCAQ6iBwMFACAAAACjggE/YYIBOzCCATegAwIBBaE
YGxZURVNULk5FVC5JU0MuVVBFTk4uRURVoiQwIqADAgEDoRswGRsEbmV3cxsRbmV0bmV3cy51cGVubi5lZH
Wjge8wgeygAwIBEKEDAgECooHfBIHcSQfLKC8vm2i17EXmomwk6hHvjBY/BnKnvvDTrbno3198vlX2RSUt+CjuAK
hcDcj4DW0gvZEqH7t5v9yWedzztlpaThebBat6hQNr9NJPozh1/+74HUwhGWb50KtjuftO/ftQ8q0nTuYKgIq6PM4tp2ddo
1IfpjfdNR9E95GFi3y1uBT7lQOwtQbRJUjPSO3ijdue9V7cNNVmYsBsqNsaHhvlBJEXf4WJdjH8yG+Dw/gX8fUTtC5fD
pB5zLt01mkSXh6Wc4UhqQtwZBI2t/
+TpX1okbg6Hr1ZZupeH6SByjCBx6ADAgEQooG/BIG8GnCmcXWtqhXh48dGTLHQgJ04K5FjRMMq2qPSbiha9lq0osq
R2KAnQA6LioWYxU+6yPKpBDSC5WOT441fUfkM8iALkW3uNc+luFCGcnDsacrmoVU7Y6Akcp9m7Fm7orRc+TWS
WPpBg3OR2oG3ATW00NAz8TT06VOLVxIMUTINKdYVI/Ja7f3sy+/N4LGkJqScCQOwlo5tfDWn/UQFiTWo5Zw435r
H8pjy2smQCnqC14v3NMAWTu4j+dzHUNw=
[S] 481 Erreur d'authentification

Exemple d'un client qui interrompt au milieu d'un échange :

[C] AUTHINFO SASL GSSAPI
[S] 383 =
[C] *
[S] 481 Authentification interrompue comme demandé

Exemple de tentative d'utilisation d'un mécanisme qui n'est pas pris en charge par le serveur :

[C] AUTHINFO SASL EXAMPLE
[S] 503 Mécanisme non reconnu

Exemple de tentative d'utilisation d'un mécanisme qui exige une couche de sécurité :

[C] AUTHINFO SASL PLAIN
[S] 483 Chiffrement ou authentification plus forte exigée

Exemple d'utilisation d'une réponse initiale avec un mécanisme qui ne la prend pas en charge (le serveur doit commencer l'échange quand on utilise [CRAM-MD5]) :

[C] AUTHINFO SASL CRAM-MD5 AHRlc3QAMTIzNA==
[S] 482 Erreur de protocole SASL

Exemple d'une authentification qui échoue à cause d'une réponse au codage incorrect :

[C] AUTHINFO SASL CRAM-MD5
[S] 383 PDE1NDE2NzQ5My4zMjY4MzE3QHRlc3RAZXhhbXBsZS5jb20+
[C] abcd=efg
[S] 504 Erreur de codage Base64

3. Syntaxe de BNF augmenté pour l'extension AUTHINFO

Cette section décrit la syntaxe formelle de l'extension AUTHINFO en utilisant l'ABNF [RFC4234]. Elle étend la syntaxe de la Section 9 de la [RFC3977], et les non terminaux non définis dans le présent document y sont définis. L'ABNF de la [RFC3977] devrait être importé avant de tenter de valider ces règles.

3.1 Commandes

Cette syntaxe étend la "commande" non-terminal, qui représente une commande NNTP.

```
command =/ authinfo-sasl-command / authinfo-user-command / authinfo-pass-command

authinfo-sasl-command = "AUTHINFO" WS "SASL" WS mechanism [WS initial-response]
  authinfo-user-command = "AUTHINFO" WS "USER" WS username
  authinfo-pass-command = "AUTHINFO" WS "PASS" WS password

  initial-response = base64-opt
  username = 1*user-pass-char
  password = 1*user-pass-char
  user-pass-char = B-CHAR
```

Note : une mise en œuvre de serveur PEUT analyser AUTHINFO USER et AUTHINFO PASS de façon à permettre que des espaces soient utilisées dans le nom d'utilisateur ou le mot de passe. Ces mises en œuvre acceptent la syntaxe additionnelle (rendant ces deux éléments incohérents avec "token" au paragraphe 9.8 de la [RFC3977]):

```
  user-pass-char =/ SP / TAB
```

Faisant ainsi, la grammaire peut devenir ambiguë si le nom d'utilisateur ou le mot de passe commence ou se finit par une espace. Pour résoudre cette ambiguïté, ces mises en œuvre traitent normalement tout ce qui est après le premier caractère espace suivant "USER"/"PASS", jusqu'au, mais non inclus, le CRLF, comme le nom d'utilisateur/mot de passe.

3.2 Continuation de commande

Cette syntaxe étend la continuation de commande non terminal, qui représente le matériel supplémentaire envoyé par le client dans le cas de commandes multi étapes.

command-continuation =/ authinfo-sasl-383-continuation

authinfo-sasl-383-continuation = ("*" / base64-opt) CRLF

3.3 Réponses

Cette syntaxe étend le "initial-response-content" non terminal, qui représente une ligne de réponse initiale envoyée par le serveur.

initial-response-content =/ response-283-content / response-383-content
response-283-content = "283" SP base64
response-383-content = "383" SP base64-opt

3.4 Entrées de capacité

Cette syntaxe étend le "capability-entry" non terminal, qui représente une capacité qui peut être annoncée par le serveur.

capability-entry =/ authinfo-capability / sasl-capability
authinfo-capability = "AUTHINFO" *(WS authinfo-variant)
authinfo-variant = "USER" / "SASL"
sasl-capability = "SASL" 1*(WS mechanism)

3.5. Non terminaux généraux

base64-opt = "=" / base64
mechanism = 1*20mech-char
mech-char = UPPER / DIGIT / "-" / "_"

4. Résumé des codes de réponse

Cette section contient une liste de chaque nouveau code de réponse défini dans le présent document et indique si il est multi lignes, quelles commandes peuvent le générer, quels arguments il a, et quelle est sa signification.

Code de réponse 281
Généré par : AUTHINFO USER, AUTHINFO PASS, AUTHINFO SASL
Signification : authentification acceptée

Code de réponse 283
Généré par : AUTHINFO SASL
Argument : challenge
Signification : authentification acceptée (avec données de succès)

Code de réponse 381
Généré par : AUTHINFO USER
Signification : mot de passe exigé via la commande AUTHINFO PASS. Noter que ce code est utilisé pour la rétro compatibilité et ne se conforme pas à l'utilisation traditionnelle des codes 3xx.

Code de réponse 383
Généré par : AUTHINFO SASL
Argument : challenge
Signification : continuer avec l'échange SASL

Code de réponse 481
Généré par : AUTHINFO USER, AUTHINFO PASS, AUTHINFO SASL
Signification : échec/rejet d'authentification

Code de réponse 482

Généré par : AUTHINFO USER, AUTHINFO PASS, AUTHINFO SASL
Signification : commandes d'authentification produites hors séquence ou erreur de protocole SASL

5. Suivi/enregistrement d'authentification

Cette section contient des suggestions de mise en œuvre et des notes de bonnes pratiques actuelles ; elle ne spécifie pas d'autres exigences de protocole de réseau.

Une fois authentifiée, l'identité d'autorisation présentée dans l'échange AUTHINFO (le nom d'utilisateur quand on utilise USER/PASS) DEVRAIT être incluse dans un journal d'audit associant l'identité à tous les articles fournis durant une opération POST, et cette configuration DEVRAIT être celle par défaut. Cela peut être fait, par exemple, en insérant des en-têtes dans les articles postés ou par un mécanisme d'enregistrement sur le serveur. Le serveur PEUT fournir une facilité pour désactiver la procédure décrite ci-dessus, car certains utilisateurs ou administrateurs peuvent considérer que c'est une violation de l'intimité.

6. Considérations sur la sécurité

Les questions de sécurité sont discutées tout au long de ce document.

En général, les considérations sur la sécurité de la [RFC4422] et tous les mécanismes SASL mis en œuvre sont applicables ici ; seuls les plus importants sont spécifiquement mentionnés ci-dessous. Aussi, cette extension n'est pas destinée à soigner les problèmes de sécurité décrits à la section 12 de la [RFC3977] ; ces considérations restent pertinentes pour toute mise en œuvre de NNTP.

Avant qu'ait commencée la négociation de la [RFC4422], toutes les interactions de protocole peuvent avoir été effectuées en clair et peuvent avoir été modifiées par un attaquant actif. Pour cette raison, les clients et serveurs DOIVENT éliminer toutes les informations sensibles obtenues avant le début de la négociation SASL à l'établissement d'une couche de sécurité. De plus, la commande CAPABILITIES DEVRAIT être produite à nouveau à l'établissement d'une couche de sécurité, et les autres états de protocole DEVRAIENT aussi être renégociés.

Les serveurs PEUVENT mettre en œuvre une politique par laquelle la connexion est abandonnée après un certain nombre d'échecs de tentative d'authentification. Si ils le font, ils NE DEVRAIENT PAS éliminer la connexion avant qu'au moins trois tentatives d'authentification aient échoué.

Les mises en œuvre DOIVENT prendre en charge une configuration où les mécanismes d'authentification qui sont vulnérables aux attaques passives d'espionnage (comme AUTHINFO USER/PASS et SASL [RFC4616]) ne sont pas annoncés ou utilisés sans la présence d'une couche de sécurité externe comme TLS [RFC4642], et cette configuration DEVRAIT être celle par défaut.

Quand plusieurs mécanismes d'authentification sont permis par le client et le serveur, un attaquant actif peut causer une négociation dégradée du plus faible mécanisme. Pour cette raison, les clients et les serveurs DEVRAIENT être configurables à interdire l'utilisation de mécanismes faibles. La force minimum acceptable est une décision de politique qui sort du domaine d'application de la présente spécification.

7. Considérations relatives à l'IANA

7.1 Considérations de l'IANA sur les services SASL/GSSAPI

L'IANA a enregistré le nom du service SASL/GSSAPI "nntp". Ce nom de service se réfère à l'utilisation authentifiée du service de nouvelles Usenet quand il est fourni via le protocole NNTP [RFC3977].

Spécification publiée : le présent document.
Contact pour plus d'informations : les auteurs de ce document.
Contrôleur des changements : IESG <iesg@ietf.org>.

7.2 Considérations de l'IANA sur les extensions NNTP

Ce paragraphe donne une définition formelle de l'extension AUTHINFO, comme exigé par le paragraphe 3.3.3 de la [RFC3977] pour le registre de l'IANA.

o Cette extension fournit un mécanisme extensible pour l'authentification NNTP via diverses méthodes.

o L'étiquette de capacité pour cette extension est "AUTHINFO".

o L'étiquette de capacité "AUTHINFO" a deux arguments facultatifs possibles, "USER" et "SASL" (comme défini au paragraphe 2.1) qui indiquent quelles variantes de la commande AUTHINFO sont prises en charge.

o Cette extension fournit aussi l'étiquette de capacité "SASL", dont les arguments font la liste des mécanismes SASL disponibles.

o Cette extension définit trois nouvelles commandes, AUTHINFO USER, AUTHINFO PASS, et AUTHINFO SASL, dont le comportement, les arguments, et les réponses sont définis aux paragraphes 2.3 et 2.4.

o Cette extension n'associe aucune nouvelle réponse aux commandes NNTP pré existantes.

o Cette extension peut affecter le comportement global du serveur et du client en ce que la commande AUTHINFO SASL peut exiger que la communication qui suit soit transmise via une couche de sécurité intermédiaire.

o La longueur de la commande AUTHINFO SASL (comme défini dans le présent document) peut excéder 512 octets. La longueur maximum de cette commande est augmentée jusqu'à celle que peut accommoder la plus grande réponse initiale possible pour tout mécanisme SASL pris en charge par la mise en œuvre.

o Cette extension définit deux nouvelles réponses, 283 et 383, dont les longueurs peuvent excéder 512 octets. La longueur maximum de ces réponses est augmentée jusqu'à celle que accommoder le plus grand défi possible pour tout mécanisme SASL pris en charge par la mise en œuvre.

o Cette extension n'altère pas l'exécution en parallèle, mais les commandes AUTHINFO ne peuvent pas être traitées en parallèle.

o L'utilisation de cette extension peut altérer la liste des capacités ; une fois la commande AUTHINFO utilisée avec succès, la capacité AUTHINFO ne peut plus être annoncée par CAPABILITIES. De plus, la capacité MODE-READER NE DOIT PAS être annoncée après une authentification réussie.

o Cette extension ne cause la production d'une réponse 401, 480, ou 483 par aucune commande pré existante.

o Cette extension n'est pas affectée par l'usage de la commande MODE READER ; cependant, la commande MODE READER NE DOIT PAS être utilisée dans la même session à la suite d'une authentification réussie.

o Spécification publiée : le présent document.

o Contact pour plus d'information : les auteurs de ce document.

o Contrôleur des changements : IESG <iesg@ietf.org>.

8. Remerciements

La présente RFC a son origine dans un document écrit initialement par Chris Newman.

Une quantité significative du texte sur l'authentification a son origine dans la révision de NNTP ou des spécifications courantes d'authentification écrites par Stan Barber. Une quantité significative du texte sur SASL a été tirée des révisions aux RFC 1734 et RFC 2554 par Rob Siemborski.

Des remerciements particuliers vont aussi à Russ Allbery, Clive Feather, et autres qui ont commenté en privé les révisions intermédiaires de ce document, ainsi que les membres du groupe de travail NNTP de l'IETF pour leurs conseils continuels

(bien que sporadiques) dans la discussion.

9. Références

9.1 Références normatives

[RFC1704] N. Haller et R. Atkinson, "Authentification sur l'Internet", octobre 1994. *(Information)*

[RFC2119] S. Bradner, "Mots clés à utiliser dans les RFC pour indiquer les niveaux d'exigence", BCP 14, mars 1997. *(MàJ par* RFC8174)

[RFC2831] P. Leach et C. Newman, "Utilisation de l'authentification par résumé comme mécanisme SASL", mai 2000. (*Obsolète, voir* RFC6331)

[RFC3454] P. Hoffman et M. Blanchet, "Préparation de chaînes internationalisées ("stringprep")", décembre 2002. *(P.S.)*

[RFC3977] C. Feather, "Protocole de transfert de nouvelles du réseau (NNTP)", octobre 2006. *(P.S., MàJ par* RFC 6048

[RFC4013] K. Zeilenga, "SASLprep : Profil Stringprep pour les noms d'utilisateur et mots de passe", février 2005.

[RFC4234] D. Crocker et P. Overell, "BNF augmenté pour les spécifications de syntaxe : ABNF", octobre 2005. *(Remplace* RFC2234, *remplacée par* RFC5234)

[RFC4422] A. Melnikov et K. Zeilenga, éd, "Authentification simple et couche de sécurité (SASL)", juin 2006. *(P.S.)*

[RFC4642] K. Murchison et autres, "Utilisation de la sécurité de la couche transport (TLS) avec le protocole de transfert des nouvelles du réseau (NNTP)", octobre 2006. *(P.S. ; MàJ par* RFC8143)

[RFC4648] S. Josefsson, "Codages de données Base16, Base32 et Base64", octobre 2006. (*Remplace* RFC3548) *(P.S.)*

9.2 Références pour information

[CRAM-MD5] Nerenberg, L., "The CRAM-MD5 SASL Mechanism", Travail en cours.

[RFC1734] J. Myers, "Commande POP3 AUTHentification", décembre 1994. *(P.S., remplacée par la* RFC5034)

[RFC2554] J. Myers, "Extension de service SMTP pour l'authentification", mars 1999. (*Obsolète, voir* RFC4954) *(P.S.)*

[RFC2980] S. Barber, "Extensions communes à NNTP", octobre 2000. (*MàJ par* RFC3977, RFC4643, RFC4644) *(Information)*

[RFC3080] M. Rose, "Cœur du protocole extensible d'échange de blocs (BEEP)", mars 2001. *(P.S.)*

[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)

[RFC4513] R. Harrison, éditeur, "Protocole léger d'accès à un répertoire (LDAP) : Méthodes d'authentification et mécanismes de sécurité", juin 2006.

[RFC4616] K. Zeilenga, éd., "Mécanisme PLAIN de l'authentification simple et couche de sécurité (SASL)", août 2006. *(P.S.)*

[RFC5802] C. Newman, A. Menon-Sen, A. Melnikov, N. Williams, "Mécanisme d'authentification par mise en cause/réponse avec sel (SCRAM) dans les mécanismes SASL et GSS-API", juillet 2010. *(P. S. ; MàJ par* RFC7677)

Adresse des auteurs

Jeffrey M. Vinocur
Department of Computer Science
Upson Hall
Cornell University
Ithaca, NY 14853
USA
mél : vinocur@cs.cornell.edu

Kenneth Murchison
Carnegie Mellon University
5000 Forbes Avenue
Cyert Hall 285
Pittsburgh, PA 15213
USA
mél : murch@andrew.cmu.edu

Déclaration complète de droits de reproduction

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