Groupe de travail Réseau
Request for Comments : 2941
RFC rendue obsolète : 1416
Catégorie :En cours de normalisation

T. Ts'o, éditeur, VA Linux Systems
J. Altman , Columbia University
septembre 2000
Traduction Claude Brière de L'Isle

Option d'authentification Telnet

Résumé
Le présent document décrit l'option d'authentification pour le protocole Telnet [RFC0854] comme méthode générique pour la négociation d'un type et mode d'authentification qui inclut le choix d'utilisation du chiffrement et de la transmission des accréditifs. Bien que ce document résume les commandes et les types actuellement utilisés, il ne définit pas un type d'authentification spécifique. Des documents séparés seront publiés pour définir chaque type d'authentification.

Le présent document met à jour une spécification précédente de l'option d'authentification Telnet, la [RFC1416], de sorte qu'elle puisse être utilisée pour activer en toute sécurité l'option de chiffrement Telnet [RFC2946].

1. Noms et codes des commandes

AUTHENTICATION 37

Commandes d'authentification :

IS	0
SEND	1
REPLY	2
NAME	3

Types d'authentification :

NULL	0
KERBEROS_V4	1
KERBEROS_V5	2
SPX*	3
MINK*	4
SRP	5
RSA*[aussi utilisé par SRA*]	6
SSL*	7
[non alloué]	8
[non alloué]	9
LOKI*	10
SSA*	11
KEA_SJ	12
KEA_SJ_INTEG	13
DSS	14
NTLM*	15

Les types d'authentification suivis par (*) n'ont jamais été soumis à l'IETF pour être pris en considération comme norme de l'Internet.

Par suite d'une pratique historique, les numéros futurs de type d'authentification et de modificateurs d'authentification seront alloués par l'IANA sur la base de la politique du premier arrivé, premier servi, comme établi dans la [RFC2434]. En

dépit du fait que les numéros de type d'authentification sont alloués dans un espace de numéros de 8 bits (comme c'est le cas de la plupart des valeurs dans la spécification Telnet) il n'est pas envisagé que l'espace des numéros soit en danger d'épuisement ou le devienne. Cependant, si cela devait devenir un problème, lorsque plus de 50 % de l'espace des numéros sera alloué, l'IANA devra adresser, pour approbation, les demandes d'allocation à l'IESG ou à un expert désigné. L'IANA a reçu pour instruction de ne pas produire de nouvelles valeurs de sous-option sans que soit soumise la documentation de leur usage.

Modificateurs :

AUTH_WHO_MASK	1
AUTH_CLIENT_TO_SERVER	0
AUTH_SERVER_TO_CLIENT	1
AUTH_HOW_MASK	2
AUTH_HOW_ONE_WAY	0
AUTH_HOW_MUTUAL	2
ENCRYPT_MASK	20
ENCRYPT_OFF	0
ENCRYPT_USING_TELOPT	4
ENCRYPT_AFTER_EXCHANGE	16
ENCRYPT_RESERVED	20
INI_CRED_FWD_MASK	8
INI_CRED_FWD_OFF	0
INI_CRED_FWD_ON	8

2. Signification des commandes

Le présent document fait référence à un "serveur" et un "client". Pour les besoins du présent document, le "serveur" est le côté de la connexion qui effectue l'ouverture passive de TCP (état TCP LISTEN) et le "client" est le côté de la connexion qui fait l'ouverture active.

IAC WILL AUTHENTICATION
Le côté client de la connexion envoie cette commande pour indiquer qu'il veut envoyer et recevoir des informations d'authentification.

IAC DO AUTHENTICATION
Le côté serveur de la connexion envoie cette commande pour indiquer qu'il veut envoyer et recevoir des informations d'authentification.

IAC WONT AUTHENTICATION
Le côté client de la connexion envoie cette commande pour indiquer qu'il refuse d'envoyer et recevoir des informations d'authentification ; le côté serveur doit envoyer cette commande si il reçoit une commande DO AUTHENTICATION.

IAC DONT AUTHENTICATION
Le côté serveur de la connexion envoie cette commande pour indiquer qu'il refuse d'envoyer ou recevoir des informations d'authentification ; le côté client doit envoyer cette commande si il reçoit une commande WILL AUTHENTICATION.

IAC SB AUTHENTICATION SEND authentication-type-pair-list IAC SE
L'envoyeur de cette commande (le serveur) demande que le côté distant envoie des informations d'authentification pour un des types d'authentification énumérés dans "authentication-type-pair-list". La liste "authentication-type-pair-list" est une liste ordonnée de paires de "authentication-type". Seul le côté serveur (DO AUTHENTICATION) est autorisé à l'envoyer.

IAC SB AUTHENTICATION IS authentication-type-pair <auth data> IAC SE
L'envoyeur de cette commande (le client) envoie les informations d'authentification pour le type d'authentification "authentication-type-pair". Seul le côté client (WILL AUTHENTICATION) est autorisé à l'envoyer.

IAC SB AUTHENTICATION REPLY authentication-type-pair <auth data> IAC SE
L'envoyeur de cette commande (le serveur) envoie une réponse aux informations d'authentification reçues dans une commande IS précédente. Seul le côté serveur (DO AUTHENTICATION) est autorisé à l'envoyer.

IAC SB AUTHENTICATION NAME remote-user IAC SE
Cette commande facultative est envoyée pour spécifier le nom du compte que l'usager souhaite être autorisé à utiliser sur l'hôte distant. Noter que l'authentification peut réussir, mais que l'autorisation d'utiliser un compte particulier peut quand même échouer. Certains mécanismes d'authentification peuvent ignorer cette commande.

La "authentication-type-pair" est de deux octets, le premier est le type d'authentification, et le second est un modificateur du type. Le type d'authentification peut inclure ou non un codage incorporé. Par exemple, lorsque le type d'authentification Kerberos 4 est négocié, le chiffrement doit être négocié avec l'option Telnet ENCRYPT. Cependant, les types d'authentification SSL et KEA_SJ fournissent un canal chiffré au titre d'une négociation d'option Telnet AUTH réussie.

Il y a actuellement cinq champs de un bit définis dans le modificateur. Les deux premiers de ces bits sont traités comme une paire, le bit AUTH_WHO_MASK et le bit AUTH_HOW_MASK. Il y a quatre combinaisons possibles de ces deux bits :

AUTH_CLIENT_TO_SERVER
AUTH_HOW_ONE_WAY
Le client va envoyer au serveur les informations d'authentification sur l'utilisateur local. Si la négociation réussit, le serveur aura authentifié l'utilisateur sur le côté client de la connexion.

AUTH_SERVER_TO_CLIENT
AUTH_HOW_ONE_WAY
Le serveur va s'authentifier auprès du client. Si la négociation réussit, le client va savoir qu'il est connecté au serveur auquel il veut être connecté.

AUTH_CLIENT_TO_SERVER
AUTH_HOW_MUTUAL
Le client va envoyer les informations d'authentification sur l'utilisateur local au serveur, et le serveur va alors s'authentifier auprès du client. Si la négociation réussit, le serveur aura authentifié l'utilisateur du côté client de la connexion, et le client va savoir qu'il est connecté au serveur auquel il voulait être connecté.

AUTH_SERVER_TO_CLIENT
AUTH_HOW_MUTUAL
Le serveur va s'authentifier auprès du client, et le client va alors s'authentifier auprès du serveur. Si la négociation réussit, le client va savoir qu'il est connecté au serveur auquel il veut être connecté, et le serveur va savoir que le client est bien celui qu'il prétend être.

Les troisième et cinquième bits du modificateur sont les bits ENCRYPT_MASK. Ces bits sont utilisés pour déterminer si et comment le chiffrement devrait être activé. Des quatre combinaisons possibles seules trois sont actuellement définies :

ENCRYPT_OFF
Le chiffrement ne sera pas utilisé pour cette session. TELOPT ENCRYPT NE DEVRAIT PAS être négocié. Ce mode DOIT être utilisé avec tous les types AUTH qui ne fournissent pas un secret partagé à utiliser comme clé de session.

ENCRYPT_USING_TELOPT
Le chiffrement sera négocié via l'utilisation de TELOPT ENCRYPT. Immédiatement après l'achèvement de l'authentification, TELOPT ENCRYPT DOIT être négocié dans les deux directions. Il est exigé que ceci se fasse avant la transmission des accréditifs ; les autres options Telnet sont négociées ; ou toutes les données d'utilisateur sont transmises. L'échec de la négociation de TELOPT ENCRYPT dans l'une ou l'autre direction DOIT résulter en l'arrêt immédiat de la session.

ENCRYPT_AFTER_EXCHANGE
Le chiffrement sera activé dans les deux directions immédiatement après la réussite de l'échange su secret partagé à utiliser comme clé de session. L'algorithme de chiffrement à utiliser DOIT être impliqué par le type AUTH.

Le champ du quatrième bit dans le modificateur est le bit INI_CRED_FWD_MASK. Ce bit est soit réglé à INI_CRED_FWD_ON, soit à INI_CRED_FWD_OFF. Ce bit est réglé par le client pour aviser le serveur de s'attendre à la transmission d'accréditifs de la part du client.

INI_CRED_FWD_OFF
Le client ne va pas transmettre d'accréditifs au serveur. Ce mode doit être utilisé si la méthode d'authentification choisie ne prend pas en charge la transmission des accréditifs.

INI_CRED_FWD_ON
Une fois l'authentification, et peut-être le chiffrement terminés, le client va immédiatement transmettre les accréditifs d'authentification au serveur.

La motivation de ce bit pour avis est que le serveur pourrait souhaiter attendre jusqu'à ce que les accréditifs transmis aient été envoyés avant de commencer aucune procédure de connexion spécifique du système d'exploitation qui pourrait dépendre de ces accréditifs. Noter que la transmission des accréditifs peut n'être pas prise en charge par tous les mécanismes d'authentification. C'est une erreur de protocole d'établir ce bit si le mécanisme d'authentification sous-jacent ne prend pas en charge la transmission des accréditifs.

Les transmission des accréditifs NE DOIT PAS être effectuée si AUTH_CLIENT_TO_SERVER | AUTH_HOW_ONE_WAY a été utilisé car l'identité du serveur ne peut pas être assurée. Les accréditifs NE DEVRAIENT PAS être transmis si la connexion Telnet n'est pas protégée en utilisant des services de chiffrement ou de protection d'intégrité.

Noter que les anciennes mises en œuvre de l'option d'authentification Telnet ne vont pas comprendre les bits ENCRYPT_MASK et INI_CRED_FWD_MASK. Donc une mise en œuvre qui souhaite offrir ces bits devrait offrir les paires de type d'authentification avec ces deux bits établis et ne pas les établir si la rétro compatibilité est exigée.

3. Spécification par défaut

La spécification par défaut pour cette option est

WONT AUTHENTICATION DONT AUTHENTICATION

qui signifie qu'il n'y aura aucun échange d'informations d'authentification.

4. Motivation

Une des faiblesses du protocole Telnet est que pour se connecter aux systèmes distants, les utilisateurs doivent taper leurs mots de passe, qui sont passés en clair à travers le réseau. Si les connexions passent par des réseaux qui ne sont pas de confiance, il existe une possibilité que les mots de passe soient compromis par quelqu'un qui surveillerait les paquets en transit.

L'objet de l'option AUTHENTICATION est de fournir un cadre pour le passage d'informations d'authentification à travers la session Telnet, et un mécanisme pour permettre le chiffrement du flux de données comme effet collatéral de la réussite de l'authentification ou via l'utilisation ultérieure de l'option Telnet ENCRYPT. Cela signifie que : 1) le mot de passe des usagers ne sera pas envoyé en clair sur le réseau ; 2) si le processus Telnet du côté frontal a les informations d'authentification appropriées, il peut automatiquement envoyer les informations, et l'usager n'aura pas à taper de mot de passe ; 3) une fois l'authentification réussie, le flux de données peut être chiffré pour fournir une protection contre les attaques actives.

Il est prévu que l'option AUTHENTICATION soit assez générale pour qu'elle puisse être utilisée pour passer des informations pour tout système d'authentification et de chiffrement.

5. Implications pour la sécurité

La capacité à négocier un mécanisme commun d'authentification entre client et serveur est une caractéristique de l'option d'authentification qui devrait être utilisée avec prudence. Lorsque la négociation est effectuée, aucune authentification n'a encore eu lieu. Chaque système n'a donc aucun moyen de savoir si il parle au système qu'il avait prévu ou non. Un intrus pourrait tenter de négocier l'utilisation du système d'authentification qui le plus faible, ou qui est déjà compromis par l'intrus.

Si le type d'authentification exige que le chiffrement soit activé au titre d'une négociation facultative séparée (l'option ENCRYPT) cela va ouvrir une fenêtre de faiblesse à un attaquant actif entre le moment où l'authentification s'achève jusque et y compris celui de la négociation pour activer le chiffrement. Une attaque active est celle où le flux TCP sous-jacent peut être modifié ou passer sous le contrôle de l'attaquant actif. Si le serveur n'offre que des paires de type d'authentification qui comportent le ENCRYPT_USING_TELOPT établi dans le champ ENCRYPT_MASK, cela évitera la

fenêtre de faiblesse, car les deux parties seront d'accord pour que l'option Telnet ENCRYPT doivent être négociée immédiatement après le bon achèvement de l'AUTH Telnet.

D'autres types d'authentification lient l'activation du chiffrement comme effet collatéral de la réussite de l'authentification. Cela donne aussi une protection contre les attaques actives. Le bit ENCRYPT_AFTER_EXCHANGE permet ces types d'authentification pour négocier le chiffrement de sorte qu'il puisse devenir facultatif.

Une autre opportunité pour les attaques actives se présente lorsque le chiffrement peut être activé et désactivé sans nouvelle authentification. Une fois que le chiffrement est désactivé, un attaquant peut capturer le flux Telnet et interférer avec les tentatives de redémarrer le chiffrement. Donc, un client NE DEVRAIT PAS accepter la capacité de désactiver le chiffrement. Une fois le chiffrement désactivé, si une tentative de réactiver le chiffrement échoue, le client DOIT mettre un terme à la connexion Telnet.

Il est important que dans les deux cas la paire de types d'authentification soit protégée en intégrité à la fin de l'échange d'authentification. Cela doit être spécifié pour chaque type d'authentification pour s'assurer que le résultat de la négociation de l'option d'authentification Telnet est accepté à la fois par le client et le serveur. Certaines sous-options de type d'authentification peuvent souhaiter inclure tous les échanges de la négociation d'authentification Telnet dans la somme de contrôle d'intégrité, pour protéger pleinement la totalité de l'échange.

Chaque côté DOIT vérifier la cohérence des paires auth-type dans chaque message reçu. Toute variation dans la paire auth-type DOIT être traitée comme une erreur fatale de protocole.

6. Règles de mise en œuvre

WILL et DO ne sont utilisés qu'au début de la connexion pour obtenir et accorder la permission de négociations futures.

L'authentification n'est négociée que dans une seule direction ; le serveur doit envoyer le "DO",et le client doit envoyer le "WILL". Cette restriction est due à la nature de l'authentification ; il y a trois cas possibles : le serveur authentifie le client, le client authentifie le serveur, et serveur et client s'authentifient l'un l'autre. En ne négociant l'option que dans une direction, et en déterminant ensuite lequel des trois cas est utilisé via la sous-option, l'ambiguïté potentielle est supprimée. Si le serveur reçoit un "DO", il doit répondre par un "WONT". Si le client reçoit un "WILL", il doit répondre par un "DONT".

Une fois que les deux hôtes ont échangé un DO et un WILL, le serveur a toute liberté pour demander les informations d'authentification. Dans la demande est envoyée une liste de types d'authentification pris en charge. Seul le serveur peut envoyer des demandes ("IAC SB AUTHENTICATION SEND authentication-type-pair-list IAC SE"). Seul le client peut transmettre les informations d'authentification via la commande "IAC SB AUTHENTICATION IS authentication-type ... IAC SE". Seul le serveur peut envoyer des réponses ("IAC SB AUTHENTICATION REPLY authentication-type ... IAC SE"). Autant de sous-options IS et REPLY peuvent être échangées que nécessaire pour le schéma d'authentification choisi.

Si le client n'accepte aucun des types d'authentification énumérés dans la liste de paires de types d'authentification, un type de NULL devrait être utilisé pour l'indiquer dans la réponse IS. Noter que si le client répond par un type NULL, le serveur peut choisir de clore la connexion.

Lorsque le serveur a tiré la conclusion que l'authentification ne peut pas être négociée avec le client, il devrait lui envoyer "IAC DONT AUTH".

Les types d'authentification DOIVENT être ordonnés par préférence en ordre de préférence décroissante.

Tant que le serveur est "WILL AUTH" il peut demander à tout moment les informations d'authentification. Cela se fait en envoyant une nouvelle liste de types d'authentification pris en charge. Demander les informations d'authentification peut être fait comme moyen de vérifier la validité des accréditifs du client après un certain temps ou pour négocier une nouvelle clé de session à utiliser durant le chiffrement.

7. Interface d’utilisateur

Normalement les spécifications de protocole ne traitent pas de la spécification de l'interface d'utilisateur. Cependant, du fait que l'utilisateur va probablement vouloir être capable de configurer l'authentification et le chiffrement et savoir si les négociations ont réussi ou non, il est nécessaire de donner quelques lignes directrices pour que les mises en œuvre fournissent un niveau minimum de contrôle à l'usager.

L'usager DOIT être capable de spécifier si l'authentification doit être utilisée ou non, et si le chiffrement sera utilisé ou non en cas de réussite de l'authentification. Il DEVRAIT y avoir au moins quatre réglages, REQUIRE, PROMPT, WARN et DISABLE. Régler le commutateur d'authentification à REQUIRE *(exigé)* signifie que si l'authentification échoue, un message d'erreur approprié doit être affiché et la connexion Telnet doit être fermée. Régler le commutateur d'authentification à PROMPT *(invite)* signifie que si l'authentification échoue, un message d'erreur approprié doit être affiché et l'usager doit être invité à fournir une confirmation pour que la connexion Telnet se poursuive. Régler le commutateur d'authentification à WARN *(avertissement)* signifie que si l'authentification échoue, un message d'erreur approprié doit alors être affiché avant que se poursuive la session Telnet. Régler le commutateur d'authentification à DISABLE *(désactiver)* signifie que l'authentification ne sera pas tentée. Le commutateur de chiffrement DEVRAIT avoir les mêmes réglages que celui d'authentification ; les réglages ne sont cependant utilisés que lorsque l'authentification réussit. Les réglage par défaut pour les deux commutateurs devrait être WARN. Ces deux commutateurs peuvent être mis en œuvre comme un seul commutateur, bien que de les avoir séparés donne plus de contrôle à l'usager.

8. Exemples

Ce qui suit est un exemple d'utilisation de l'option :

Client **Serveur**

IAC DO AUTHENTICATION

IAC WILL AUTHENTICATION

[Le serveur est maintenant libre de demander les informations d'authentification.]

IAC SB AUTHENTICATION SEND KERBEROS_V4 CLIENT|MUTUAL KERBEROS_V4 CLIENT|ONE_WAY IAC SE

[Le serveur a demandé l'authentification mutuelle Kerberos, mais il veut faire juste une authentification Kerberos unidirectionnelle. Le client va alors répondre par le nom d'usager sous lequel il veut se connecter, et le ticket Kerberos.]

IAC SB AUTHENTICATION NAME "joe"
IAC SE
IAC SB AUTHENTICATION IS
KERBEROS_V4 CLIENT|MUTUAL AUTH 4
7 1 67 82 65 89 46 67 7 9 77 0 48 24 49 244 109 240 50 208 43 35 25 116 104 44 167 21 201 224 229 145 20 2 244 213 220 33 134 148 4 251 249 233 229 152 77 2 109 130 231 33 146 190 248 1 9 31 95 94 15 120 224 0 225 76 205 70 136 245 190 199 147 155 13
IAC SE

[Le serveur répond par une commande ACCEPT pour déclarer que l'authentification a réussi.]

IAC SB AUTHENTICATION REPLY
KERBEROS_V4 CLIENT|MUTUAL ACCEPT
IAC SE

[Ensuite, le client envoie un CHALLENGE pour vérifier qu'il parle bien au bon serveur.]

IAC SB AUTHENTICATION IS
KERBEROS_V4 CLIENT|MUTUAL
CHALLENGE xx xx xx xx xx xx xx xx IAC SE

[Enfin, le serveur envoie une RESPONSE pour prouver qu'il est réellement le bon serveur.]

IAC SB AUTHENTICATION REPLY
KERBEROS_V4 CLIENT|MUTUAL
RESPONSE yy yy yy yy yy yy yy yy
IAC SE

Ce qui suit est un exemple d'utilisation de l'option avec chiffrement négocié via le ENCRYPT de Telnet :

Client **Serveur**

IAC DO AUTHENTICATION

IAC WILL AUTHENTICATION

[Le serveur est maintenant libre de demander les informations d'authentification.]

IAC SB AUTHENTICATION SEND
KERBEROS_V4
CLIENT|MUTUAL|ENCRYPT_USING_TELOPT
KERBEROS_V4 CLIENT|ONE_WAY IAC SE

[Le serveur a demandé l'authentification mutuelle Kerberos, mais il veut juste faire l'authentification Kerberos unidirectionnelle. Dans les deux cas, il veut chiffrer le flux de données. Le client va maintenant répondre avec le nom d'usager sous lequel il veut se connecter, et le ticket Kerberos.]

IAC SB AUTHENTICATION NAME "joe"
IAC SE
IAC SB AUTHENTICATION IS
KERBEROS_V4
CLIENT|MUTUAL|ENCRYPT_USING_TELOPT
AUTH 4 7 1 67 82 65 89 46 67 7 9 77 0 48 24 49 244 109 240 50 208 43 35 25 116 104 44 167 21 201 224 229 145 20 2 244 213 220 33 134 148 4 251 249 233 229 152 77 2 109 130 231 33 146 190 248 1 9 31 95 94 15 120 224 0 225 76 205 70 136 245 190 199 147 155 13 IAC SE

[Le serveur répond par une commande ACCEPT pour déclarer que l'authentification a réussi.]

IAC SB AUTHENTICATION REPLY
KERBEROS_V4
CLIENT|MUTUAL|ENCRYPT_USING_TELOPT
ACCEPT IAC SE

[Ensuite, le client envoie un CHALLENGE pour vérifier qu'il parle bien au bon serveur.]

IAC SB AUTHENTICATION IS
KERBEROS_V4
CLIENT|MUTUAL|ENCRYPT_USING_TELOPT
CHALLENGE xx xx xx xx xx xx xx xx IAC SE

[Le serveur envoie une RESPONSE pour prouver qu'il est réellement le bon serveur.]

IAC SB AUTHENTICATION REPLY
KERBEROS_V4
CLIENT|MUTUAL|ENCRYPT_USING_TELOPT
RESPONSE yy yy yy yy yy yy yy yy IAC SE

[À ce point, le client et le serveur commencent à négocier l'option Telnet ENCRYPT dans chaque direction pour un canal sûr. Si l'option échoue dans l'une ou l'autre direction pou n'importe quelle raison, la connexion doit être immédiatement interrompue.]

Ce qui suit est un exemple d'utilisation de l'option avec chiffrement intégré :

Client	**Serveur**
	IAC DO AUTHENTICATION
IAC WILL AUTHENTICATION	
[Le serveur est maintenant libre de demander les informations d'authentification.]	
	IAC SB AUTHENTICATION SEND KEA_SJ CLIENT|MUTUAL|ENCRYPT_AFTER_EXCHANGE IAC SE
[Le serveur a demandé l'authentification mutuelle KEA avec le chiffrement SKIPJACK. Le client répond alors par le nom d'usager sous lequel il veut se connecter et le certificat KEA.]	
IAC SB AUTHENTICATION NAME "joe" IAC SE IAC SB AUTHENTICATION IS KEA_SJ CLIENT|MUTUAL|ENCRYPT_AFTER_EXCHANGE '1' CertA||Ra IAC SE	
[Le serveur répond avec son certificat KEA.]	
	IAC SB AUTHENTICATION REPLY KEA_SJ CLIENT|MUTUAL|ENCRYPT_AFTER_EXCHANGE '2' CertB||Rb||IVb||Encrypt(NonceB) IAC SE
[Ensuite, le client envoie un CHALLENGE pour vérifier qu'il parle bien au bon serveur.]	
IAC SB AUTHENTICATION IS KEA_SJ CLIENT|MUTUAL|ENCRYPT_AFTER_EXCHANGE '3' IVa||Encrypt(NonceB xor 0x0C18 || NonceA) IAC SE	
[À ce point, le client commence à chiffrer le flux de données sortant, et le serveur, après avoir reçu cette commande, commence à déchiffrer le flux de données entrant. Enfin, le serveur envoie une RESPONSE pour prouver qu'il est bien le bon serveur.]	
	IAC SB AUTHENTICATION REPLY KEA_SJ CLIENT|MUTUAL|ENCRYPT_AFTER_EXCHANGE '4' Encrypt(NonceA xor 0x0C18) IAC SE
[À ce point, le serveur commence à chiffrer son flux de données sortantes, et le client, après avoir reçu cette commande, commence à déchiffrer le flux de données entrantes.]	

On s'attend à ce que toute mise en œuvre qui prend en charge l'option Telnet AUTHENTICATION prenne en charge la totalité de la présente spécification.

9. Considérations pour la sécurité

Le présent mémoire décrit un cadre général pour ajouter l'authentification et le chiffrement au protocole Telnet. Le mécanisme d'authentification réel est décrit dans les spécifications de sous-option d'authentification, et la sécurité de l'option d'authentification dépend de la force et des faiblesses de la sous-option d'authentification.

On notera que la négociation de la paire type/authentification n'est pas protégée, ce qui permet donc à un attaquant de forcer le résultat de l'authentification à être la méthode la plus faible mutuellement acceptable. (Par exemple, même si les deux côtés de la négociation peuvent accepter un mécanisme "fort" et un mécanisme à "40 bits", un attaquant pourrait forcer le choix du mécanisme à "40 bits".) Une mise en œuvre devrait donc n'accepter qu'un mécanisme d'authentification soit négocié que si elle veut croire qu'il est sûr.

On notera aussi que la négociation du nom d'utilisateur dans le message SE IAC du nom IAC SB AUTHENTICATION NAME n'est pas protégée. Les mises en œuvre devraient vérifier la valeur par une méthode sûre avant d'utiliser cette valeur qui n'est pas de confiance.

11. Remerciements

De nombreuses personnes ont travaillé au présent document depuis de nombreuses années. Dave Borman a été éditeur et auteur de beaucoup du texte original. D'autres personnes qui ont contribué à ce texte par des idées et des suggestions sont David Carrel, Jeff Schiller, et Richard Basch.

10. Références

[RFC0854] J. Postel et J. Reynolds," Spécification du protocole TELNET", STD 8, mai 1983.

[RFC1416] D. Borman, "Option Telnet Authentification", janvier 1993. *(Expérimentale, remplacée par la présente RFC)*

[RFC2434] T. Narten et H. Alvestrand, "Lignes directrices pour la rédaction d'une section Considérations relatives à l'IANA dans les RFC", BCP 26, octobre, 1998. *(Rendue obsolète par la RFC5226)*

[RFC2946] T. Ts'o, "Option Telnet de chiffrement des données", septembre 2000. *(P.S.)*

12. Adresse des auteurs

Theodore Ts'o, Editor
VA Linux Systems
43 Pleasant St.
Medford, MA 02155
téléphone : (781) 391-3464
mél : tytso@mit.edu

Jeffrey Altman
Columbia University
Watson Hall Room 716
612 West 115th Street
New York NY 10025
téléphone : +1 (212) 854-1344
mél : jaltman@columbia.edu

Liste de diffusion : telnet-wg@BSDI.COM

13 Déclaration complète de droits de reproduction

nécessaire à l'élaboration des normes Internet, auquel cas les procédures pour les droits de reproduction définies dans les processus des normes de l'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 assuré par la Internet Society.