Groupe de travail Réseau
Request for Comments : 2952
Catégorie : Information

T. Ts'o, VA Linux Systems
septembre 2000
Traduction Claude Brière de L'Isle

Chiffrement Telnet : DES en chiffrement à rebouclage par le chiffre à 64 bits

Résumé
Le présent document spécifie comment utiliser l'algorithme de chiffrement DES en mode à rebouclage par le chiffre avec l'option de chiffrement Telnet.

1. Noms et codes des commandes

Type de chiffrement

DES_CFB64	1

Commandes de sous-option

CFB64_IV	1
CFB64_IV_OK	2
CFB64_IV_BAD	3

2. Signification des commandes

IAC SB ENCRYPT IS DES_CFB64 CFB64_IV <vecteur initial> IAC SE
L'envoyeur de cette commande génère un vecteur initial aléatoire de 8 octets et l'envoie à l'autre côté de la connexion en utilisant la commande CFB64_IV. Le vecteur initial est envoyé en clair. Seul le côté de la connexion qui est WILL ENCRYPT peut envoyer la commande CFB64_IV.

IAC SB ENCRYPT REPLY DES_CFB64 CFB64_IV_OK IAC SE
IAC SB ENCRYPT REPLY DES_CFB64 CFB64_IV_BAD IAC SE
L'envoyeur de ces commandes accepte ou rejette le vecteur initial reçu dans une commande CFB64_IV. Seul le côté de la connexion qui est DO ENCRYPT peut envoyer les commandes CFB64_IV_OK et CFB64_IV_BAD. La commande CFB64_IV_OK DOIT être envoyée pour la rétro compatibilité avec les mises en œuvre existantes ; il n'y a aucune raison pour qu'un envoyeur ait besoin d'envoyer la commande CFB64_IV_BAD sauf dans le cas d'une violation de protocole où le vecteur initial envoyé n'aurait pas la longueur correcte (c'est à dire, ne ferait pas 8 octets).

3. Règles de mise en œuvre

Une fois qu'une commande CFB64_IV_OK a été reçue, le côté WILL ENCRYPT de la connexion devrait faire la négociation d'id_de_clé en utilisant la commande ENC_KEYID. Une fois que la négociation du id_de_clé a bien identifié un id_de_clé commun, les commandes START et END peuvent alors être envoyées par le côté de la connexion qui est WILL ENCRYPT. Les données seront chiffrées en utilisant l'algorithme DES à rebouclage par le chiffre à 64 bits.

Si le chiffrement (déchiffrement) est désactivé et réactivé à nouveau, et si le même id_de_clé est utilisé lors du redémarrage du chiffrement (déchiffrement) le texte en clair qui intervient ne doit pas changer l'état de la machine de chiffrement (déchiffrement).

Si une commande START est envoyée (reçue) avec un id_de_clé différent, la machine de chiffrement (déchiffrement) doit être réinitialisée immédiatement après la fin de la commande START avec la nouvelle clé et le vecteur initial envoyés

(reçus) dans la dernière commande CFB64_IV.

Si une nouvelle commande CFB64_IV est envoyée (reçue) et si le chiffrement (déchiffrement) est activé, la machine de chiffrement (déchiffrement) doit être réinitialisée immédiatement après la fin de la commande CFB64_IV avec le nouveau vecteur initial et le id_de_clé envoyés (reçus) dans la dernière commande START.

Si le chiffrement (déchiffrement) n'est pas activé lorsque une commande CFB64_IV est envoyée (reçue) la machine de chiffrement (déchiffrement) doit être réinitialisée après la prochaine commande START, avec le id_de_clé envoyé (reçu) dans cette commande START, et le vecteur initial envoyé (reçu) dans cette commande CFB64_IV.

4. Algorithme

Sachant que V[i] est le vecteur initial de 64 bits, V[n] est le n^{e} vecteur de 64 bits, D[n] est le n^{e} tronçon de 64 bits de données à chiffrer (déchiffrer), et O[n] est le n^{e} tronçon de 64 bits de données chiffrées (déchiffrées), alors :

V[0] = DES(V[i], clé)
O[n] = D[n] <OU exclusif> V[n]
V[n+1] = DES(O[n], clé)

5. Intégration avec l'option Telnet AUTHENTICATION

Comme il est noté dans les spécifications de l'option Telnet ENCRYPTION, une valeur d'id_de_clé de zéro indique la clé de chiffrement par défaut, comme on peut la déduire de l'option Telnet AUTHENTICATION. Si la clé de chiffrement par défaut négociée par suite de l'option Telnet AUTHENTICATION contient moins de 8 octets, l'option DES_CFB64 ne doit alors pas être offerte ni utilisée comme option de chiffrement Telnet valide. Si la clé de chiffrement négociée par suite de l'option Telnet AUTHENTICATION fait plus de 16 octets, les huit premiers octets de la clé devraient être utilisés comme id_de_clé 0 pour les données envoyées du client Telnet au serveur Telnet, et les huit octets suivants de la clé devraient être utilisés comme id_de_clé 0 pour les données envoyées par le serveur Telnet au client Telnet. Autrement, les huit premiers octets de la clé de chiffrement sont utilisés comme id_de_clé zéro pour l'option Telnet ENCRYPTION dans les deux directions (avec le client comme WILL ENCRYPT et le serveur comme WILL ENCRYPT).

Dans tous les cas, si la clé négociée par l'option Telnet AUTHENTICATION n'était pas une clé DES, la clé utilisée par le DES_CFB64 doit avoir sa parité corrigée après qu'il a été déterminé d'utiliser l'algorithme ci-dessus.

Noter que l'algorithme ci-dessus suppose qu'il est sûr d'utiliser une clé non DES (ou une partie d'une clé non DES) comme une clé DES. Ceci n'est pas nécessairement vrai de tous les systèmes de chiffrement, mais on spécifie ce comportement comme le comportement par défaut car il est vrai pour la plupart des systèmes d'authentification d'utilisation courante aujourd'hui, et pour la compatibilité avec les mises en œuvre existantes. De nouveaux mécanismes d'authentification Telnet pourront spécifier d'autres méthodes pour déterminer dans leur spécification les clés à utiliser pour cette suite de chiffrement, si la clé de session négociée par ce mécanisme d'authentification n'est pas une clé DES et lorsque cet algorithme peut n'être pas utilisé de façon sûre.

6. Considérations pour la sécurité

Le chiffrement qui utilise le rebouclage par le chiffre n'assure pas la protection de l'intégrité des données ; un attaquant actif a une capacité limitée à modifier le texte, si il peut prédire le texte en clair qui a été transmis. Les limitations auxquelles fait face l'attaquant (le fait que seuls 8 octets peuvent être modifiés à la fois, et que le bloc de huit octets de données suivant sera corrompu, ce qui rend donc la détection vraisemblable) sont significatives, mais il est possible qu'un attaquant actif soit néanmoins capable d'exploiter cette faiblesse.

Le compromis est ici d'ajouter un code d'authentification de message (MAC) ce qui augmente de façon significative le nombre d'octets nécessaires pour envoyer un seul caractère dans le protocole Telnet, ce qui va impacter les performances sur les liaisons lentes (c'est-à-dire, les liaisons téléphoniques).

7. Remerciements

Ce document a été rédigé à l'origine par Dave Borman de Cray Research avec le concours du groupe de travail Telnet de l'IETF.

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

8 Déclaration complète de droits de reproduction

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