

Groupe de travail Réseau  
**Request for Comments : 2951**  
 Catégorie : Information  
 Traduction Claude Brière de L'Isle

R. Housley, T. Horting, P. Yee  
 SPYRUS  
 septembre 2000

## Authentification Telnet avec KEA et SKIPJACK

### Statut de ce mémoire

Le présent mémoire apporte des informations pour la communauté de l'Internet. Il ne spécifie aucune norme de l'Internet. La distribution du présent mémoire n'est soumise à aucune restriction.

### Copyright Notice

Copyright (C) The Internet Society (2000). Tous droits réservés.

### Résumé

Le présent document définit une méthode pour authentifier Telnet en utilisant l'algorithme d'échange de clé (KEA, *Key Exchange Algorithm*) et le chiffrement du flux Telnet en utilisant SKIPJACK. Deux modes de chiffrement sont spécifiés ; l'un assure l'intégrité des données et l'autre non. La méthode s'appuie sur l'option Telnet AUTHENTICATION.

## 1. Noms et codes des commandes

AUTHENTICATION 37

Commandes d'authentification :

IS	0
SEND	1
REPLY	2
NAME	3

Types d'authentification :

KEA_SJ	12
KEA_SJ_INTEG	13

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

Commandes de sous-option :

KEA_CERTA_RA	1
KEA_CERTB_RB_IVB_NONCEB	2
KEA_IVA_RESPONSEB_NONCEA	3
KEA_RESPONSEA	4

## 2. Extensions de sécurité Telnet

En tant que protocole, Telnet n'a pas de concept de sécurité. Sans négociation d'options, il passe simplement les caractères entre les terminaisons de réseau virtuel représentées par les deux processus Telnet. Dans son utilisation la plus courante comme protocole pour l'accès au terminal distant (accès TCP 23) Telnet se connecte normalement à un serveur qui exige une authentification de niveau utilisateur au moyen d'un nom d'utilisateur et d'un mot de passe en clair. Le serveur ne s'authentifie pas auprès de l'utilisateur.

L'option Authentication de Telnet fournit :

- \* l'authentification de l'utilisateur -- remplaçant ou augmentant le mécanisme normal de mot de passe d'hôte ;
- \* l'authentification du serveur – normalement faite en conjonction avec l'authentification d'utilisateur ;
- \* la négociation des paramètres de session -- en particulier, la clé et les attributs de chiffrement ;
- \* la protection de session – principalement par le chiffrement des données et du flux de commandes incorporées, mais l'algorithme de chiffrement peut aussi fournir la protection de l'intégrité des données.

Afin de prendre en charge ces services de sécurité, les deux entités Telnet doivent d'abord négocier leur volonté de prendre en charge l'option Authentication de Telnet. Une fois obtenu l'accord pour la prise en charge de cette option, les parties sont alors capables d'effectuer la négociation des sous-options pour déterminer le protocole d'authentification à utiliser, et éventuellement le nom d'utilisateur distant à utiliser pour la vérification des autorisations. Le chiffrement est négocié avec le type d'authentification.

L'authentification et la négociation des paramètres surviennent au sein d'une série non limitée d'échanges. Le serveur propose une liste rangée par ordre de préférence des types (mécanismes) d'authentification qu'il accepte. En plus de faire la liste des mécanismes qu'il accepte, le serveur qualifie chaque mécanisme avec un modificateur qui spécifie si le chiffrement des données est désiré. Le client choisit un mécanisme dans la liste et répond au serveur en indiquant son choix et le premier ensemble de données d'authentification nécessaire pour le type d'authentification choisi. Le client peut ignorer une demande de chiffrer les données et l'indiquer, mais le serveur peut aussi terminer la connexion si le client refuse le chiffrement. Le serveur et le client procèdent alors au nombre d'itérations nécessaire pour arriver à l'authentification demandée.

Le chiffrement démarre immédiatement après que la négociation de l'option Authentication est achevée.

## 3. Utilisation de l'algorithme d'échange de clé (KEA)

Le présent document spécifie la méthode dans laquelle KEA est utilisé pour réaliser l'authentification Telnet. KEA (conjointement avec SKIPJACK) [4] fournit l'authentification et la confidentialité. La protection de l'intégrité peut aussi être fournie.

Les entités Telnet peuvent utiliser KEA pour fournir l'authentification mutuelle et la prise en charge de l'établissement de clés de chiffrement des données. Un simple format de jeton et un ensemble d'échanges assurent ces services.

NonceA et NonceB utilisés dans cet échange sont des chaînes de 64 bits. Le client génère NonceA, et le serveur génère NonceB. La valeur du nom occasionnel est choisie au hasard. Le nom occasionnel est envoyé en forme gros bouton (*poids fort en premier*). Le chiffrement du nom occasionnel sera fait avec le même mécanisme qui sera utilisé par la session, et qui est détaillé au paragraphe suivant.

Ra et Rb utilisés dans cet échange sont des chaînes de 1024 bits et sont définies par l'algorithme KEA [4].

IVa et IVb sont les vecteurs d'initialisation de 24 octets. Ils sont composés de "THIS IS NOT LEAF" suivi de 8 octets aléatoires.

CertA est le certificat du client. CertB est le certificat du serveur. Les deux certificats sont des certificats X.509 [6] qui contiennent des clés publiques KEA [7]. Le client doit valider le certificat du serveur avant d'utiliser la clé publique KEA qu'il contient. De même, le serveur doit valider le certificat du client avant d'utiliser la clé publique KEA qu'il contient.

À l'achèvement de ces échanges, les parties ont une clé SKIPJACK commune. L'authentification mutuelle est fournie par la vérification des certificats utilisés pour établir la clé de chiffrement SKIPJACK et par la bonne utilisation de la clé de session SKIPJACK déduite. Pour se protéger contre les attaques actives, le chiffrement va avoir lieu après la réussite de l'authentification. Il n'y aura pas de moyen de désactiver le chiffrement et de le réactiver en toute sécurité ; la répétition de la procédure complète d'authentification est le seul moyen sûr de le redémarrer. Si l'utilisateur ne veut pas utiliser le chiffrement, il peut désactiver le chiffrement après l'établissement de la session.

### 3.1 Modes SKIPJACK

Il y a deux modes distincts de chiffrement des flux Telnet ; l'un d'eux fournit la protection de l'intégrité et pas l'autre. Comme Telnet fonctionne normalement en mode caractère par caractère, SKIPJACK avec mécanisme de protection de l'intégrité des flux exige la transmission de quatre octets pour chaque octet de données Telnet. Cependant, un mode simplifié de SKIPJACK sans mécanisme de protection de l'intégrité ne va exiger que la transmission d'un octet pour chaque octet de données Telnet.

Le mode cryptographique pour SKIPJACK avec intégrité des flux est le rebouclage du chiffre sur 32 bits de données (CFB-32, *Cipher Feedback on 32 bits of data*) et le mode de SKIPJACK est le rebouclage de chiffrement sur 8 bits de données (CFB-8, *Cipher Feedback on 8 bits of data*).

#### 3.1.1 SKIPJACK sans protection de l'intégrité du flux

Le premier mode, qui est le moins compliqué utilise SKIPJACK CFB-8. Ce mode ne fournit pas de protection de l'intégrité du flux.

Pour SKIPJACK sans protection de l'intégrité du flux, la paire de deux octets du type d'authentification type est KEA\_SJ AUTH\_CLIENT\_TO\_SERVER | AUTH\_HOW\_MUTUAL | ENCRYPT\_AFTER\_EXCHANGE | INI\_CRED\_FWD\_OFF. Cela indique que le mécanisme SKIPJACK sans protection de l'intégrité sera utilisé pour l'authentification mutuelle et le chiffrement du flux Telnet. La Figure 1 illustre le mécanisme d'authentification de KEA suivi par SKIPJACK sans protection de l'intégrité du flux.

Client (Partie A)	Serveur (Partie B)
IAC WILL AUTHENTICATION -->	<-- IAC DO AUTHENTICATION
	<-- IAC SB AUTHENTICATION SEND
	<liste des options d'authentification> IAC SE
IAC SB AUTHENTICATION NAME <nom d'usager> -->	
IAC SB AUTHENTICATION IS KEA_SJ	
AUTH_CLIENT_TO_SERVER   AUTH_HOW_MUTUAL	
ENCRYPT_AFTER_EXCHANGE   INI_CRED_FWD_OFF	
KEA_CERTA_RA CertA  Ra IAC SE ----->	
	<-- IAC SB AUTHENTICATION REPLY KEA_SJ
	AUTH_CLIENT_TO_SERVER   AUTH_HOW_MUTUAL
	ENCRYPT_AFTER_EXCHANGE   INI_CRED_FWD_OFF
	IVA_RESPONSEB_NONCEA
	KEA_CERTB_RB_IVB_NONCEB CertB  Rb  IVb
	Chiffre( NonceB ) IAC SE
IAC SB AUTHENTICATION IS KEA_SJ	
AUTH_CLIENT_TO_SERVER   AUTH_HOW_MUTUAL	
ENCRYPT_AFTER_EXCHANGE   INI_CRED_FWD_OFF	
KEA_IVA_RESPONSEB_NONCEA	
Iva  Chiffre( NonceB OUX 0x0C12)  NonceA ) IAC SE -->	
<le client commence le chiffrement>	
	<-- IAC SB AUTHENTICATION REPLY KEA_SJ
	AUTH_CLIENT_TO_SERVER   AUTH_HOW_MUTUAL
	ENCRYPT_AFTER_EXCHANGE   INI_CRED_FWD_OFF
	KEA_RESPONSEA Chiffre( NonceA OUX 0x0C12 ) IAC
SE	
	<le serveur commence le chiffrement>

Figure 1.

#### 3.1.2 SKIPJACK avec protection de l'intégrité du flux

SKIPJACK avec protection de l'intégrité du flux est plus compliqué. Il utilise la fonction de hachage unidirectionnel SHA-1 [3] pour fournir la protection de l'intégrité du flux de chiffrement comme suit :

Établir H0 comme étant le hachage SHA-1 d'une chaîne de longueur zéro.

Cn est le n° caractère dans le flux Telnet.

Hn = SHA-1( Hn-1||Cn ) où Hn est la valeur du hachage associée au n° caractère dans le flux.

ICVn est défini comme les trois octets de plus fort poids de Hn.

Transmettre Chiffre( Cn||ICVn ).

Le texte chiffré qui est transmis est le chiffrement SKIPJACK CFB-32 de ( Cn||ICVn ). L'extrémité réceptrice de la liaison Telnet inverse le processus, en commençant par déchiffrer le texte chiffré, en séparant Cn et ICVn, en recalculant Hn, en recalculant ICVn, puis en comparant le ICVn reçu avec le ICVn recalculé. L'intégrité est indiquée si la comparaison réussit, et Cn peut alors être traité normalement au titre du flux Telnet. L'échec de la comparaison indique une perte de l'intégrité, qu'elle soit due à une manipulation active ou à une perte de la synchronisation cryptographique. Dans l'un et l'autre cas, le seul recours est d'abandonner la connexion Telnet et de recommencer.

Pour SKIPJACK avec protection de l'intégrité du flux, la paire de deux octets de type d'authentification est KEA\_SJ\_INTEG\_AUTH\_CLIENT\_TO\_SERVER | AUTH\_HOW\_MUTUAL | ENCRYPT\_AFTER\_EXCHANGE | INI\_CRED\_FWD\_OFF. Cela indique que le mécanisme KEA SKIPJACK avec protection de l'intégrité sera utilisé pour l'authentification mutuelle et le chiffrement du flux Telnet. La Figure 2 illustre le mécanisme d'authentification de KEA SKIPJACK avec protection de l'intégrité du flux.

Client (Partie A)	Serveur (Partie B)
IAC WILL AUTHENTICATION -->	<-- IAC DO AUTHENTICATION
	<-- IAC SB AUTHENTICATION SEND
	<liste des options d'authentification> IAC SE
IAC SB AUTHENTICATION NAME <nom d'utilisateur>-->	
IAC SB AUTHENTICATION IS KEA_SJ_INTEG	
AUTH_CLIENT_TO_SERVER AUTH_HOW_MUTUAL	
ENCRYPT_AFTER_EXCHANGE   INI_CRED_FWD_OFF	
KEA_CERTA_RA CertA  Ra IAC SE ----->	
	<-- IAC SB AUTHENTICATION REPLY KEA_SJ_INTEG
	AUTH_CLIENT_TO_SERVER   AUTH_HOW_MUTUAL
	ENCRYPT_AFTER_EXCHANGE   INI_CRED_FWD_OFF
	IVA_RESPONSEB_NONCEA
	KEA_CERTB_RB_IVB_NONCEB CertB  Rb  IVb
	Chiffre( NonceB ) IAC SE
IAC SB AUTHENTICATION IS KEA_SJ_INTEG	
AUTH_CLIENT_TO_SERVER   AUTH_HOW_MUTUAL	
ENCRYPT_AFTER_EXCHANGE	
INI_CRED_FWD_OFF KEA_IVA_RESPONSEB_NONCEA	
IVa  Chiffre( NonceB OUX 0x0D12)  NonceA ) IAC SE -->	
<le client commence le chiffrement>	
	<-- IAC SB AUTHENTICATION REPLY KEA_SJ_INTEG
	AUTH_CLIENT_TO_SERVER   AUTH_HOW_MUTUAL
	ENCRYPT_AFTER_EXCHANGE   INI_CRED_FWD_OFF
	KEA_RESPONSEA Chiffre( NonceA XOR 0x0D12 )
	IAC SE
	<le serveur commence le chiffrement>

Figure 2

#### 4. Considérations pour la sécurité

La totalité du présent mémoire est consacrée aux mécanismes de sécurité. Pour que KEA fournisse l'authentification exposée, la mise en œuvre doit protéger la clé privée contre toute divulgation. De même, les clés SKIPJACK doivent être protégées de la divulgation.

Les mises en œuvre doivent générer les clés privées KEA de façon aléatoire, ainsi que les vecteurs d'initialisation (les IV), et les noms occasionnels (*nonce*). L'utilisation de générateurs de nombres pseudo aléatoires inadéquats (PRNG, *pseudo-random number generator*) pour générer des clés de chiffrement peut résulter en une sécurité affaiblie ou inexistante. Un attaquant peu trouver beaucoup plus facile de reproduire l'environnement du PRNG qui a produit les clés, en cherchant dans l'ensemble de possibilités restreint résultant, plutôt qu'une recherche en force brute dans la totalité de l'espace de clés.

La génération de nombres aléatoires de qualité est difficile. La RFC1750 [8] offre des lignes directrices importantes dans ce domaine, et l'appendice 3 de FIPS Pub 186 [9] donne une technique pour avoir des PRNG de qualité.

En liant l'activation du chiffrement comme effet collatéral de la réussite de l'authentification, la protection est assurée contre un attaquant actif. Si le chiffrement était activé par une négociation séparée, cela fournirait une fenêtre de vulnérabilité depuis le moment de l'achèvement de la négociation de l'authentification jusque et y compris au moment de la négociation sur l'activation du chiffrement. Le seul moyen sûr de redémarrer le chiffrement si il est désactivé est de répéter la totalité du processus d'authentification.

## 5. Considérations relatives à l'IANA

Les types d'authentification KEA\_SJ et KEA\_SJ\_INTEG et leurs valeurs de sous-option associées sont enregistrés auprès de l'IANA. Toute valeur de sous-option utilisée pour étendre le protocole tel que décrit dans le présent document doit être enregistré auprès de l'IANA avant utilisation. L'IANA a reçu pour instruction de ne pas délivrer de nouvelles valeurs de sous-option sans la soumission de la documentation de leur utilisation.

## 6. Remerciements

Nous tenons à remercier William Nace pour son soutien durant la mise en œuvre de cette spécification.

## 7. Références

- [1] J. Postel et J. Reynolds, "Spécification du [protocole TELNET](#)", RFC0854, STD 8, mai 1983.
- [2] T. Ts'o, éd., J. Altman, "[Option d'authentification Telnet](#)", RFC2941, septembre 2000. (P.S.)
- [3] FIPS Pub 180-1. "Secure Hash Standard". 17 avril 1995.
- [4] "SKIPJACK and KEA Algorithm Specification", Version 2.0, 29 mai 1998. Disponible à <http://csrc.nist.gov/encryption/skipjack-kea.htm>
- [5] J. Postel et J. Reynolds, "Spécifications des [options TELNET](#)", RFC0855, mai 1983.
- [6] R. Housley, W. Ford, W. Polk et D. Solo, "Profil de [certificat d'infrastructure de clé publique X.509](#) et de CRL pour l'Internet", RFC2459, janvier 1999. (*Obsolète, voir la RFC5280*) (P.S.)
- [7] R. Housley, W. Polk, "Représentation de [l'infrastructure de clé publique X.509](#) pour l'Internet des clés de l'algorithme d'échange de clés (KEA) dans les certificats d'infrastructure de clé publique X.509 de l'Internet" RFC2528, mars 1999. (*Info.*)
- [8] D. Eastlake, 3<sup>rd</sup> et autres, "Recommandations d'[aléa pour la sécurité](#)", RFC1750, décembre 1994. (*Info., remplacée par la RFC4086*)
- [9] National Institute of Standards and Technology. FIPS Pub 186: "Digital Signature Standard". 19 mai 1994.

## 8. Adresse des auteurs

Russell Housley  
SPYRUS  
381 Elden Street, Suite 1120  
Herndon, VA 20170  
USA  
mél : [housley@spyrus.com](mailto:housley@spyrus.com)

Todd Horting  
SPYRUS  
381 Elden Street, Suite 1120  
Herndon, VA 20170  
USA  
mél : [thorting@spyrus.com](mailto:thorting@spyrus.com)

Peter Yee  
SPYRUS  
5303 Betsy Ross Drive  
Santa Clara, CA 95054  
USA  
mél : [yee@spyrus.com](mailto:yee@spyrus.com)

## 9. Déclaration complète de droits de reproduction

Copyright (C) The Internet Society (2000). 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 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.