Groupe de travail Réseau
Request for Comments : 4535
Catégorie : Sur la voie de la normalisation
Traduction Claude Brière de L'Isle

H. Harney, U. Meth
A. Colegrove, SPARTA, Inc.
G. Gross, IdentAware
June 2006

GSAKMP : protocole de gestion de clés d'association de groupe sécurisé

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 spécifie le protocole de gestion de clé d'association de groupe sécurisé (GSAKMP, *Group Secure Association Key Management Protocol*). GSAKMP fournit un cadre de sécurité pour créer et gérer des groupes cryptographiques sur un réseau. Il fournit des mécanismes pour distribuer la politique de groupe et authentifier les utilisateurs, les règles pour prendre les décisions de contrôle d'accès durant l'établissement et la récupération de groupe, les capacités pour récupérer de la compromission de membres du groupe, la délégation des fonctions de sécurité de groupe, et les capacités pour détruire le groupe. Il génère aussi les clés de groupe.

Table des matières

1. Introduction

GSAKMP assure la distribution de la politique, la mise en application de la politique, la distribution des clés, et la gestion des clés pour des groupes cryptographiques. Les groupes cryptographiques partagent tous une clé commune (ou un ensemble de clés) pour le traitement des données. Ces clés prennent toutes en charge une politique de sécurité au niveau du système afin que le groupe cryptographique puisse être de confiance pour effectuer des services pertinents pour la sécurité.

La capacité d'un groupe d'entités à effectuer des services de sécurité exige qu'une association de groupe sécurisée (GSA, *Group Secure Association*) soit établie. Une GSA assure qu'il y a une définition commune "au niveau du groupe" de la politique de sécurité et de l'application de cette politique. La distribution des clés de chiffrement est un mécanisme qui utilise les applications de politique au niveau du groupe.

1.1 Vue d'ensemble de GSAKMP

La protection des informations de groupe exige la définition d'une politique de sécurité et l'application de cette politique par tous les participants. Le contrôle de la dissémination de la clé de chiffrement est le principal mécanisme pour appliquer la politique de contrôle d'accès. C'est le principal objet de GSAKMP de générer et disséminer une clé de groupe de façon sûre.

GSAKMP sépare les fonctions et les responsabilités de gestion de la sécurité de groupe en trois rôles majeurs :1) Propriétaire du groupe, 2) Contrôleur de groupe/Serveur de clés, et 3) Membre du groupe. Le propriétaire du groupe est responsable de la création des règles de politique de sécurité pour un groupe et les exprime dans le jeton de politique. Le Contrôleur de groupe/Serveur de clés (GC/KS) est responsable de la création et du maintien des clés et de l'application de la politique de groupe en accordant l'accès aux membres du groupe (GM, *Group Member*) potentiels en accord avec le jeton de politique. Pour appliquer la politique d'un groupe, les membres de groupe potentiels ont besoin de connaître la politique de contrôle d'accès pour le groupe, une identification non ambiguë de toutes les parties qui leur téléchargent des clés, et des chaînes d'autorité vérifiables pour leur téléchargement de clé. En d'autres termes, les membres du groupe ont besoin de savoir qui va potentiellement être dans le groupe et de vérifier que celui qui dissémine la clé est autorisé à agir dans cette capacité.

Afin d'établir une association de groupe sécurisée (GSA, *Group Secure Association*) pour prendre en charge ces activités, l'identité de chaque partie au processus DOIT être affirmée sans ambiguïté et authentifiée. Il DOIT aussi être vérifié que chaque partie est autorisée, comme défini par le jeton de politique, pour fonctionner dans ce rôle dans le protocole (par exemple, GM ou GC/KS).

Les caractéristiques de sécurité du protocole d'établissement pour la GSA incluent :
- l'identification de la politique de groupe
- la dissémination de la politique de groupe
- l'établissement d'une SA du GM au GC/KS pour protéger les données
- la vérification du contrôle d'accès.

GSAKMP fournit des mécanismes pour la création et la gestion de groupes cryptographiques. D'autres protocoles peuvent être utilisés en conjonction avec GSAKMP pour permettre à diverses applications de créer des groupes fonctionnels en accord avec les exigences spécifiques de leur application. Par exemple, dans une vidéo conférence à petite échelle, l'organisateur pourrait utiliser un protocole d'invitation à une session comme SIP [RFC3261] pour transmettre des informations sur l'heure de la conférence, l'adresse de la session, et les formats à utiliser. Pour une transmission vidéo à grande échelle, l'organisateur pourrait utiliser un protocole d'annonce en diffusion groupée comme SAP [RFC2974].

Le présent document décrit un ensemble utile par défaut d'algorithmes et configurations de sécurité, Sécurité Suite 1. Cette suite permet de décrire un ensemble complet d'algorithmes et réglages aux membres potentiels du groupe d'une manière concise. D'autres suites de sécurité PEUVENT être définies comme nécessaire et PEUVENT être disséminées durant l'annonce hors bande d'un groupe.

Les architectures réparties prennent en charge des groupes cryptographiques à grande échelle. Les architectures réparties sécurisées exigent une délégation autorisée des actions de GSA aux ressources du réseau. Le jeton de politique pleinement spécifié est le mécanisme qui facilite cette autorisation. La transmission de ce jeton de politique à tous les GM qui se joignent au groupe permet à GSAKMP de prendre en charge en toute sécurité les architectures réparties et les sources de données multiples.

Les communications de groupe de beaucoup à beaucoup exigent des sources de données multiples. Les sources de données multiples sont prises en charge à cause de l'inclusion d'un jeton de politique et de charges utiles de politique qui permet aux membres du groupe de revoir le contrôle d'accès de groupe et les paramètres d'autorisation. Ce processus de revue par les membres donne à chaque membre (chaque source de données potentielle) la capacité de déterminer si le groupe fournit une protection adéquate pour les données des membres.

1.2 Organisation du document

Le reste de ce document est organisé comme suit :la Section 2 présente la terminologie et les concepts utilisés pour formuler les exigences de ce protocole. La Section 3 présente les considérations sur la sécurité par rapport à GSAKMP. La Section 4 définit l'architecture de GSAKMP. La Section 5 décrit le cycle de vie de la gestion de groupe. La Section 6 décrit la définition de la suite de sécurité. La Section 7 présente les types et formats de message utilisés durant chaque phase du cycle de vie. La Section 8 définit les diagrammes d'état pour le protocole.

2. Terminologie

La terminologie suivante est utilisée dans le présent document.

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].

Certificat : structure de données utilisée pour lier d'une façon vérifiable une identité à une clé cryptographique (par exemple, X.509v3).

Récupération de compromission : acte de récupération d'un état de fonctionnement sûr après la détection qu'un membre du groupe n'est pas fiable. Ceci peut être accompli par un changement de clé.

Groupe cryptographique : ensemble d'entités partageant ou désirant partager une GSA.

Contrôleur de groupe/serveur de clés (GC/KS, *Group Controller/Key Server*) : membre du groupe qui a autorité pour effectuer des actions critiques du protocole, incluant de créer et distribuer les clés et construire et maintenir les structures de changement de clés. Avec l'évolution du groupe, il PEUT devenir souhaitable d'avoir plusieurs contrôleurs pour effectuer ces fonctions.

Membre de groupe (GM) : toute entité qui a accès aux clés de groupe. Sans considération de la façon dont un membre adhère au groupe ou dont le groupe est structuré, les GM vont effectuer les actions suivantes :

- Authentifier et valider les identités et les autorisations des entités qui effectuent les actions pertinentes pour la sécurité
- Accepter les clés de groupe venant du GC/KS

- Demander les clés de groupe au GC/KS
- Appliquer les politiques de groupe coopératives comme déclaré dans le jeton de politique de groupe
- Effectuer la revue d'homologue des actions de gestion des clés
- Gérer la clé locale.

Propriétaire du groupe (GO, *Group Owner*) : c'est l'entité autorisée à générer et modifier un jeton de politique authentifiable pour le groupe, et à notifier au GC/KS de commencer le groupe.

Politique de groupe : elle décrit complètement les mécanismes de protection et les comportements pertinents pour la sécurité pour le groupe. Cette politique DOIT être couramment comprise et appliquée par le groupe pour un fonctionnement sûr cohérent.

Association de groupe sécurisée (GSA, *Group Secure Association*) : association logique d'utilisateurs ou hôtes qui partagent des clés de chiffrement. Ce groupe peut être établi pour prendre en charge des associations entre les applications ou protocoles de communication.

Clé de protection du trafic de groupe (GTPK, *Group Traffic Protection Key*) : la ou les clés créées pour la protection des données du groupe.

Élément de clé *(Key Datum)* : une seule clé et ses attributs associés pour son utilisation.

Clé de chiffrement de clé (KEK, *Key Encryption Key*) : clé utilisée dans un mécanisme de chiffrement pour envelopper une autre clé.

Bride de clé *(bride de clé)* : identifiant d'une instance ou version particulière d'une clé.

Identifiant de clé *(Key Identifier)* : identifiant pour une clé qui DOIT rester statique pour tout le cycle de vie de cette clé.

Paquetage de clé *(Key Package)* : format Type/Longueur/Données contenant un élément de clé.

Matrice de hiérarchie logique de clés (LKH, *Logical Key Hierarchy*) : groupe des clés créé pour faciliter la méthodologie de récupération de LKH compromise.

Jeton de politique (PT, P*olicy Token*) : c'est une structure de données utilisée pour disséminer la politique de groupe et les mécanismes pour l'appliquer. Le jeton de politique est produit et signé par un propriétaire de groupe autorisé. Chaque membre du groupe DOIT vérifier le jeton, satisfaire à la politique d'adhésion au groupe, et appliquer la politique du groupe (par exemple, chiffrer les données d'application avec un algorithme spécifique). Le jeton de politique de groupe va contenir diverses informations, incluant :
- la version du protocole GSAKMP
- la méthode de création de clé
- la politique de dissémination des clés
- la politique de contrôle d'accès
- la politique d'autorisation de groupe
- la politique de récupération de compromission
- les mécanismes de protection des données.

Changement de clé *(Rekey)* : acte de changer les clés au sein d'un groupe selon la politique définie.

Matrice de changement de clé *(Rekey Array)* : construction qui contient toutes les informations de changement de clé pour un certain membre.

Clé de changement de clé *(Rekey Key)* : c'est la KEK utilisée pour chiffrer les clés pour un sous ensemble du groupe.

Contrôleur de groupe/serveur de clés subordonné (S-GC/KS, *Subordinate Group Controller/Key Server*) : tout membre de groupe qui a le traitement approprié et les caractéristiques de confiance, définies dans la politique de groupe, qui a le potentiel d'agir comme S-GC/KS. Cela permet au traitement de groupe et aux exigences de communication d'être répartis équitablement parmi le réseau (par exemple, pour distribuer la clé de groupe). L'utilisation facultative de GSAKMP avec des contrôleurs de groupe/serveurs de clés subordonnés sera documentée dans un document séparé.

Identifiant de clé d'enveloppe *(Wrapping KeyID)* : identifiant de clé de la clé utilisée pour envelopper un paquetage de clé.

Bride de clé d'enveloppe *(Wrapping bride de clé)* : bride de clé de la clé utilisée pour envelopper le paquetage de clé.

3. Considérations sur la sécurité

En plus de la spécification de GSAKMP lui-même, la sécurité de la mise en œuvre d'un système GSAKMP est affectée par des facteurs de prise en charge, qui sont exposés ici.

3.1 Hypothèses sur la sécurité

Les hypothèses suivantes sont faites comme base de la discussion de la sécurité :

1. GSAKMP suppose que sa plate forme de soutien peut fournir le traitement et les services de séparation des données au niveau d'assurance approprié pour prendre en charge ses groupes.

2. La fonction de génération de clés du moteur cryptographique ne va générer que des clés fortes.

3. La sécurité de ce protocole est critiquement dépendante de l'aléa des paramètres choisis au hasard. Ils devraient être générés par une source fortement aléatoire ou une source pseudo-aléatoire d'un germe appropriés [RFC4086].

4. La sécurité d'un groupe peut être affectée par la précision de l'horloge système. Donc, GSAKMP suppose que l'horloge système est proche de l'heure correcte. Si un hôte GSAKMP s'appuie sur un service de l'heure du réseau pour régler son horloge locale, ce protocole doit alors être sécurisé contre les attaquants. La dérive maximum admissible d'horloge entre les membres du groupe est configurable par la politique, avec une valeur par défaut de 5 minutes.

5. Comme décrit dans la section de traitement du message, l'utilisation de la valeur de nom occasionnel utilisée pour la fraîcheur d'une signature est le mécanisme utilisé pour déjouer les attaques en répétition. Dans toute utilisation d'un nom occasionnel, une exigence clé est l'imprévisibilité du nom occasionnel, du point de vue de l'attaquant. L'utilité du nom occasionnel s'appuie sur l'incapacité d'un attaquant de réutiliser les vieux noms occasionnels ou de prédire la valeur du nom occasionnel.

6. GSAKMP ne fournit pas de protection de l'identité.

7. L'infrastructure d'acheminement de diffusion groupée du groupe n'est pas sécurisée par GSAKMP, et donc, il est possible de créer une attaque de déni de service par inondation de diffusion groupée en utilisant le flux de données de l'application de diffusion groupée. Un infiltré (c'est-à-dire un GM félon) ou un non membre pourrait conduire les routeurs de diffusion groupée à arroser de données un système victime.

8. La compromission d'un S-GC/KS force au réenregistrement de tous les GM sous son contrôle. Le GM reconnaît cette situation en trouvant le certificat du S-GC/KS sur une liste de révocation de certificats (CRL, *Certificate Revocation List*) fournie par un service comme LDAP.

9. La compromission du GO force la terminaison du groupe. Le GM reconnaît cette situation en trouvant le certificat du GO sur une CRL fournie par un service comme LDAP.

3.2 Protocoles en relation

GSAKMP est dérivé de deux protocoles existants : ISAKMP [RFC2408] et FIPS Pub 196 [FIPS196]. En accord avec la suite de sécurité n° 1, les mises en œuvre de GSAKMP DOIVENT prendre en charge l'utilisation de l'échange de clés Diffie-Hellman [DH77] pour la création de clés à deux parties et PEUVENT utiliser la hiérarchie logique de clé (LKH, *Logical Key Hierarchy*) [RFC2627] pour la capacité de changement de clés. La conception de GSAKMP a aussi été influencée par les protocoles suivants : [HHMCD01], [RFC2093], [RFC2094], [BMS], et [RFC2412].

3.2.1 ISAKMP

ISAKMP fournit une structure souple de charges utiles enchaînées à l'appui d'un échange de clés authentifié et la gestion d'association de sécurité pour des communications d'homologue à homologue. GSAKMP construit sur ces caractéristiques pour fournir des caractéristiques d'application de politique à l'appui de diverses communications de groupe.

3.2.2 FIPS Pub 196

FIPS Pub 196 fournit un protocole d'authentification mutuelle.

3.2.3 LKH

Quand la politique de groupe impose une récupération de la sécurité du groupe après la découverte de la compromission d'un GM, GSAKMP s'appuie alors sur une capacité de changement de clés (c'est-à-dire, LKH) pour activer la récupération de groupe après une compromission [RFC2627]. Ceci est facultatif car dans de nombreuses instances il peut être meilleur de détruire le groupe compromis et de reconstruire un groupe sûr.

3.2.4 Diffie-Hellman

Un groupe peut s'appuyer sur des mécanismes de création de clé en deux parties, c'est-à-dire, Diffie-Hellman, pour protéger des données sensibles durant le téléchargement.

Les informations de ce paragraphe empruntent beaucoup à la [RFC4306], car ce protocole a déjà travaillé sur des problèmes similaires et GSAKMP utilise les mêmes considérations de sécurité pour ses besoins. Ce paragraphe contient une paraphrase des paragraphes de la [RFC4306] modifiés comme approprié pour GSAKMP.

La force d'une clé déduite d'un échange Diffie-Hellman utilisant des valeurs spécifiques p et g dépend de la force inhérente des valeurs, de la taille de l'exposant utilisé, et de l'entropie fournie par le générateur de nombres aléatoires utilisé. Un générateur de nombres aléatoire fort, combiné avec les recommandations de la [RFC3526] sur la taille de l'exposant Diffie-Hellman est recommandé comme suffisant. Une mise en œuvre devrait noter cette estimation prudente lors de l'établissement d'une politique et de la négociation des paramètres de sécurité.

Noter que ces limitations sont sur les valeurs Diffie-Hellman elles mêmes. Il n'y a rien dans GSAKMP qui interdise l'utilisation de valeurs plus fortes, ni rien qui puisse diluer la force obtenue de valeurs plus fortes. En fait, le cadre extensible de GSAKMP invite à la définition de plus de suites de sécurité.

On suppose que les exposants Diffie-Hellman dans cet échange sont écrasés de la mémoire après usage. En particulier, ces exposants NE DOIVENT PAS être déduits de secrets à longue durée tels que le germe d'un générateur pseudo aléatoire qui n'est pas détruit après usage.

3.3 Attaques de déni de service (DoS)

Cette spécification de GSAKMP traite de l'atténuation d'attaque répartie d'usurpation d'identité IP (un sous ensemble de possibles attaques de DoS) au paragraphe 5.2.2, "Mouchards : établissement de groupe avec protection contre le déni de service".

3.4 Disponibilité du changement de clés

En plus de la capacité de GSAKMP de faire des opérations de changement de clés, GSAKMP DOIT aussi avoir la capacité de tenir ces informations de changement de clé à la disposition des GM. La nécessité que les GM reçoivent les messages de changement de clé exige l'utilisation de méthodes pour augmenter la probabilité de réception des messages de changement de clés. Ces méthodes PEUVENT inclure de multiples transmissions du message de changement de clé, l'envoi du message de changement de clé sur un panneau d'affichage, etc. Les mises en œuvre de GSAKMP conformes qui prennent en charge la capacité facultative de changement de clé DOIVENT prendre en charge la retransmission de messages de changement de clé.

3.5 Preuve de hiérarchie de confiance

Comme défini par [HCM], la politique de sécurité de groupe DOIT être définie d'une manière vérifiable. GSAKMP ancre sa confiance dans le créateur du groupe, le GO.

Le jeton de politique définit explicitement tous les paramètres qui créent une infrastructure sûre vérifiable. Le jeton de politique GSAKMP est produit et signé par le GO. Le GC/KS va le vérifier et accorder l'accès aux GM seulement si ils satisfont aux règles du jeton de politique. Les nouveaux GM ne vont accepter l'accès que si 1) le jeton se vérifie, 2) le

GC/KS est un disséminateur autorisé, et 3) les mécanismes du groupe sont acceptables pour protéger les données des GM.

4. Architecture

Cette architecture présente un modèle de confiance pour GSAKMP et un concept de fonctionnement pour établir une infrastructure répartie de confiance pour la distribution de la clé de groupe et de la politique.

GSAKMP se conforme aux concepts architecturaux de sécurité de la diffusion groupée de l'IETF MSEC comme spécifiés dans le document d'architecture de sécurité de la diffusion groupée [RFC3740]. GSAKMP utilise les composants de la sécurité de la diffusion groupée pour créer un modèle ce confiance pour les opérations qui mettent en œuvre les principes de sécurité de suspicion mutuelle et d'autorités de confiance de création de politique.

4.1 Modèle de confiance

4.1.1 Composants

Le modèle de confiance contient quatre composants clés :
- Le propriétaire du groupe (GO),
- Le contrôleur de groupe/serveur de clés (GC/KS),
- Le GC/KS subordonné (S-GC/KS), et
- Le membre du groupe (GM).

Le but du modèle de confiance GSAKMP est de déduire la confiance d'une autorité commune de confiance de création de politique pour un groupe. Toutes les décisions et actions pertinentes pour la sécurité mises en œuvre par GSAKMP sont fondées sur des informations qui sont en fin de compte traçables et vérifiées par l'autorité de confiance de création de politique. Il y a deux autorités de confiance de création de politique pour GSAKMP : le GO (autorité de création de politique) et la racine de l'infrastructure de clé publique (PKI, *Public Key Infrastructure*) qui nous permet de vérifier le GO.

4.1.2 GO

Le GO est l'autorité de création de politique pour le groupe. Le GO a une identité bien définie qui est pertinente pour le groupe. Cette identité peut être une personne ou un composant de confiance pour le groupe. Toutes les entités potentielles dans le groupe doivent reconnaître le GO comme l'individu qui a l'autorité de spécifier la politique pour le groupe.

La politique reflète les exigences de protection des données dans un groupe. En fin de compte, les données et l'environnement d'application conduisent la politique de sécurité pour le groupe.

Le GO doit déterminer les règles et mécanismes de sécurité qui sont appropriés pour les données protégées par les clés de groupe. Toutes ces informations sont capturées dans un jeton de politique (PT, *Policy Token*). Le GO crée le PT et le signe.

4.1.3 GC/KS

Le GC/KS est autorisé à effectuer plusieurs fonctions : création de clé, distribution des clés, changement des clés, et gestion des membres du groupe.

Comme autorité de création de clé, le GC/KS va créer l'ensemble des clés pour le groupe. Ces clés incluent les clés de protection du trafic du groupe (GTPK, *Group Traffic Protection Key*) et les clés de changement de clé de premier rang. Il peut y avoir des arborescences de changement de clé de second rang si une structure répartie de gestion de changement de clé est requise pour le groupe.

En tant qu'autorité de distribution des clés (enregistrement) il doit notifier au groupe sa localisation des services d'enregistrement. Le GC/KS va devoir appliquer le contrôle d'accès aux clés au titre des processus de distribution des clés et d'enregistrement.

En tant qu'autorité de changement de clés de groupe, il effectue le changement de clé afin de changer la GTPK du groupe. Le changement de la GTPK limite l'exposition des données chiffrées avec une seule GTPK.

Finalement, en tant qu'autorité de gestion des membres du groupe, le GC/KS peut gérer les membres du groupe (enregistrement, éviction, désenregistrement, etc.). Ceci peut être fait en partie en utilisant une approche d'arborescence de clés, comme des hiérarchies logiques de clés (LKH, *Logical Key Hierarchy*) qui est facultative.

4.1.4 GC/KS subordonné

Un GC/KS subordonné est utilisé pour distribuer la fonction de GC/KS à plusieurs entités. Le S-GC/KS va avoir toute l'autorité du GC/KS sauf une : il ne va pas créer la GTPK. Il est supposé ici que le groupe va transmettre des données avec une seule GTPK à tout moment. Cette GTPK vient du GC/KS.

Noter qu'à l'égard du GC/KS, le S-GC/KS est responsable d'une vérification de sécurité supplémentaire : le S-GC/KS doit s'enregistrer comme membre auprès du GC/KS, et durant ce processus, il doit vérifier l'autorité du GC/KS.

4.1.5 GM

Le GM a deux tâches : s'assurer que toutes les actions pertinentes pour la sécurité sont autorisées et utiliser de façon appropriée les clés de groupe. Durant le processus d'enregistrement, le GM va vérifier que le PT est signé par un GO reconnu. De plus, il va vérifier que le GC/KS ou S-GC/KS engagé dans le processus d'enregistrement est autorisé, comme spécifié dans le PT. Si un changement de clés et de nouveaux PT sont distribués au groupe, le GM va vérifier qu'ils sont appropriés et que toutes les actions sont autorisées.

L'accès aux données du groupe est accordé au GM par la réception des clés de groupe. Cela va avec sa responsabilité de protéger la clé contre une divulgation non autorisée.

GSAKMP n'offre aucun mécanisme d'application pour contrôler quels GM sont des locuteurs de diffusion groupée à un moment donné. Cette politique et son application dépendent de l'application de diffusion groupée et de ses protocoles. Cependant, GSAKMP permet à un groupe d'avoir une des trois configurations de locuteur de diffusion groupée d'association de sécurité de groupe suivantes :

- Il y a un seul GM autorisé à être le porte-parole du groupe. Il y a une SA d'application de diffusion groupée allouée par le GO en soutien de ce porte-parole. Le PT initialise cette SA d'application de diffusion groupée et identifie le GM qui a été autorisé à être porte-parole. Tous les GM partagent une seule TPK avec ce GM porte-parole. La vérification du numéro de séquence pour la protection contre la répétition est faisable et activée par défaut. C'est la configuration de groupe par défaut. Les mises en œuvre de GSAKMP DOIVENT prendre en charge cette configuration.

- Le GO autorise tous les GM à être les porte-parole du groupe. Le GO alloue une SA d'application de diffusion gropée en soutien de ces porte-parole. Le PT initialise cette SA d'application de diffusion groupée et indique que tout GM peut être un porte-parole. Tous les GM partagent une seule GTPK et les autres informations d'état de la SA. Par conséquent, certaines caractéristiques de sécurité de la SA comme la vérification du numéro de séquence pour la protection contre la répétition ne peuvent pas être prises en charge par cette configuration. Les mises en œuvre de GSAKMP DOIVENT prendre en charge cette configuration de groupe.

- Le GO autorise un sous ensemble des GM à être porte-parole du groupe (qui peut être le sous ensemble composé de tous les GM). Le GO alloue une SA d'application de diffusion groupée distincte pour chacun de ces porte-parole. Le PT identifie les porte-parole autorisés et initialise chacune de leurs associations de sécurité d'application de diffusion groupée. Les porte-parole partagent quand même une TPK commune à travers leur SA, mais chaque porte-parole a une instance d'informations d'état de SA séparée à chaque GM homologue. Par conséquent, cette configuration prend en charge les caractéristiques de sécurité de SA, comme la vérification de numéro de séquence pour la protection contre la répétition, ou des mécanismes d'authentification de source qui exigent un état par porte-parole chez le receveur. L'inconvénient de cette configuration est qu'elle ne s'adapte pas à un grand nombre de porte-parole. Les mises en œuvre de GSAKMP PEUVENT prendre en charge cette configuration de groupe.

4.1.6 Hypothèses

Les hypothèses pour ce modèle de confiance sont que :
- le GCKS n'est jamais compromis,
- le GO n'est jamais compromis,
- la PKI, sous réserve de validation du certificat, est digne de confiance,
- le GO est capable de créer une politique de sécurité pour satisfaire les demandes du groupe,

- la compromission d'un membre du groupe va être détectable et rapportée au GO d'une manière fiable,
- la récupération subséquente d'une compromission va refuser au membre compromis un accès inapproprié aux données protégées,
- aucune action pertinente pour la sécurité ne dépend d'une heure réseau précise,
- il y a des mécanismes de protection de la confidentialité, de l'intégrité, d'authentification de la source de diffusion groupée, et de protection contre la répétition pour tous les messages de contrôle GSAKMP.

4.2 Politique de sécurité fondée sur la règle

Le modèle de confiance pour GSAKMP tourne autour de la définition et l'application de la politique de sécurité. En fait, l'utilisation de la clé n'est pertinent, au sens de la sécurité, que si elle représente l'application réussie de la politique de sécurité du groupe.

Les opérations du groupe se prêtent elles-mêmes à une politique de sécurité fondée sur la règle. Le besoin de distribution des données à de nombreux points d'extrémité conduit souvent à la définition de ces points d'extrémité autorisés sur la base des règles. Par exemple, tous les participants à une certaine conférence de l'IETF pourraient être définis comme un seul groupe.

Si les règles de la politique de sécurité doivent être pertinentes, elles doivent être couplées à des mécanismes de validation. Le principe central est ici que le niveau de confiance qu'on peut accorder à une politique de sécurité est exactement égal au niveau de confiance accordé au mécanisme de validation utilisé pour prouver cette politique. Par exemple, si tous les participants à l'IETF sont admis, ils peuvent alors enregistrer leur identité à partir de leur certificat à l'entrée des réunions. Ce certificat est produit par une autorité de création de politique de confiance (racine de PKI) qui est autorisée à identifier quelqu'un comme participant à l'IETF. Le GO pourrait faire des règles d'admission au groupe IETF sur la base des certificats d'identité produits à partir des PKI de confiance.

Dans GSAKMP, chaque règle de politique de sécurité est couplée avec un mécanisme explicite de validation. Pour des considérations d'interopérabilité, GSAKMP exige que les mises en œuvre de PKI qui le prennent en charge DOIVENT être conformes à la RFC 3280.

Si le certificat de clé publique d'un GM est révoqué, l'entité qui a produit cette révocation DEVRAIT alors le signaler au GO, afin que celui-ci puisse expulser ce GM. La méthode pour signaler cet événement au GO n'est pas normalisée par la présente spécification.

Une transposition directe de règle en mécanisme de validation permet l'utilisation de plusieurs règles et PKI pour créer des groupes. Cela permet à un GO de définir une politique de sécurité de groupe qui s'étende sur plusieurs domaines de PKI, chacun avec son propre certificat de clé publique d'autorité de certification.

4.2.1 Contrôle d'accès

La politique de contrôle d'accès pour les clés de groupe est équivalente à la politique de contrôle d'accès pour les données d'application de diffusion groupée que les clés protègent.

Dans un groupe, chaque source de données est chargée de s'assurer que l'accès aux données de la source est approprié. Cela implique que chaque source de données devrait avoir connaissance de la politique de contrôle d'accès pour les clés de groupe.

Dans le cas général, GSAKMP offre une suite de services de sécurité à ses applications et ne prescrit pas comment elles utilisent ces services.

GSAKMP prend en charge la création de GSA avec des sources de données multiples. Il prend aussi en charge des architectures où le GC/KS n'est pas lui-même une source de données. Dans les multiples architectures de source de données, GSAKMP exige que la politique de contrôle d'accès soit définie avec précision et distribuée à chaque source de données. La référence pour cette structure de données est le jeton de politique GSAKMP [RFC4534].

4.2.2 Autorisations pour des actions pertinentes pour la sécurité

Un aspect critique du modèle de confiance GSAKMP est l'autorisation des actions pertinentes pour la sécurité. Cela inclut le téléchargement de la clé de groupe, le changement de clé, et la création et la mise à jour du jeton de politique. Ces

actions pourraient être utilisées pour perturber le groupe sécurisé, et toutes les entités dans le groupe doivent vérifier que ces actions sont à l'instigation des entités autorisées au sein du groupe.

4.3 Fonctionnement réparti

L'adaptabilité est une caractéristique centrale de GSAKMP. L'approche de GSAKMP des opérations adaptables est l'établissement de S-GC/KS. Cela permet aux systèmes GSAKMP de répartir la charge de travail d'établissement et de gestion de très grands groupes.

Un autre aspect du fonctionnement réparti des S-GC/KS est l'activation d'autorités de gestion locales. Dans les très grands groupes des enclaves subordonnées peuvent être mieux adaptées pour assurer la gestion locale des membres du groupe de l'enclave, due à une connaissance directe des membres du groupe.

Une des questions critiques impliquées par le fonctionnement réparti est la découverte de la localisation de l'infrastructure de sécurité et de la suite de sécurité. De nombreuses applications de groupe qui ont des interactions dynamiques doivent "se trouver" l'une l'autre pour fonctionner. La découverte de l'infrastructure de sécurité est juste un autre élément d'information qui doit être connu par le groupe afin de fonctionner en toute sécurité.

Il y a plusieurs méthodes pour la découverte de l'infrastructure :
- les annonces
- l'envoi à la cantonade
- les points de rendez-vous/enregistrement

Une méthode pour distribuer la localisation de l'infrastructure de sécurité est d'utiliser des annonces. SAP est couramment utilisé pour annoncer l'existence d'une nouvelle application ou service de diffusion groupée. Si une application utilise SAP [RFC2974] pour annoncer l'existence d'un service sur un canal de diffusion groupée, ce service pourrait être étendu à inclure la localisation de l'infrastructure de sécurité pour un groupe particulier.

Les annonces peuvent aussi être utilisées par GSAKMP dans un des deux modes suivants : les recherches par anneau d'expansion (ERS, *expanding ring search*) de l'infrastructure de sécurité et les ERS pour la découverte d'infrastructure. Dans l'un et l'autre cas, GSAKMP va utiliser une diffusion groupée qui va lentement augmenter sa portée en incrémentant les bonds de diffusion groupée. La source de diffusion groupée contrôle la portée de diffusion groupée des paquets en réglant explicitement leur compte de durée de vie.

Dans une annonce d'anneau d'expansion, le GC/KS annonce son existence pour un groupe particulier. Le nombre de bonds que cette annonce va traverser va être configuré en local. Les GM vont écouter sur une adresse de diffusion groupée bien connue pour les GC/KS qui fournissent le service pour des groupes d'intérêt. Si plusieurs GC/KS se trouvent fournir le service, le GM va alors prendre le plus proche (en termes de bonds de diffusion groupée). Le GM va alors envoyer un message GSAKMP Demande d'adhésion (RTJ, *Request To Join*) au GC/KS annoncé. Si l'annonce se trouve être une annonce parasite, cela est rapporté aux autorités de gestion appropriées. Le concept d'ERA est légèrement différent de SAP en ce qu'il pourrait survenir sur l'adresse de diffusion groupée du canal de données, au lieu d'une adresse spéciale de diffusion groupée dédiée au service SAP.

Une recherche par anneau d'expansion fonctionne dans l'ordre inverse de l'ERA. Dans ce cas, le GM est l'entité qui annonce. Les (S-)GC/KS écoutent les demandes de service, spécifiquement, les RTJ. Le (S-)GC/KS répond à la RTJ. Si le GM reçoit plus d'une réponse, il va ignorer les réponses ou envoyer des NACK sur la base de sa configuration locale.

L'envoi à la cantonade est un service très similaire à ERS. Il peut aussi être utilisé pour fournir la connexion à l'infrastructure de sécurité. Dans ce cas, le GM va envoyer la RTJ à une adresse de demande de service bien connue. Ce service d'envoi à la cantonade va acheminer la RTJ à un GC/KS approprié. Le service d'envoi à la cantonade va avoir une connaissance de l'infrastructure de sécurité et de connexité au réseau pour faciliter cette connexion.

Les points d'enregistrement peuvent être utilisés pour distribuer de nombreuses données pertinentes pour le groupe, incluant l'infrastructure de sécurité. De nombreuses applications de groupe s'appuient sur des points d'enregistrement bien connus pour annoncer la disponibilité des groupes. Il n'y a pas de raison pour que GSAKMP ne puisse pas utiliser la même approche pour annoncer l'existence et la localisation de l'infrastructure de sécurité. C'est un processus simple si l'application prise en charge accepte l'enregistrement. L'infrastructure GSAKMP peut toujours fournir un site d'enregistrement si l'existence de cette plate-forme de découverte d'infrastructure est nécessaire. L'enregistrement des S-GC/KS à ce site pourrait être un moyen efficace de permettre l'enregistrement du GM.

La découverte de l'infrastructure GSAKMP peut utiliser tout mécanisme qui convient aux exigences d'une application de diffusion groupée particulière, incluant des mécanismes qui n'ont pas été discutés dans cette architecture. Cependant, la découverte de l'infrastructure GSAKMP n'est pas normalisée par la présente version de la spécification GSAKMP.

4.4 Concept de fonctionnement

Ce concept de fonctionnement montre comment les différents rôles dans GSAKMP interagissent pour établir un groupe sûr. Ce concept particulier de fonctionnement se concentre sur un groupe sécurisé qui utilise les services répartis de dissémination de clés du S-GC/KS.

4.4.1 Hypothèses

L'hypothèse de base est qu'il y a une ou plusieurs PKI dignes de confiance pour le groupe. Cette PKI de confiance va être utilisée pour créer et vérifier les règles de politique de sécurité.

Il y a un GO que tous les GM reconnaissent comme ayant l'autorité de création de politique de groupe. Tous les GM doivent être préconfigurés de façon sûre à connaître la clé publique du GO.

Tous les GM ont accès aux informations de PKI du GO, les clés publiques d'ancre de confiance et les règles de validation de chemin de certificats.

Il y a une connexité suffisante entre les entités GSAKMP.
- La SA d'enregistrement exige que le GM puisse se connecter au GC/KS ou S-GC/KS en utilisant TCP ou UDP.
- La SA de changement de clé exige que le service de communication de diffusion groupée de couche de données soit disponible. Ce peut être IP en diffusion groupée, des réseaux superposés utilisant TCP, ou des tunnels de NAT.
- GSAKMP peut prendre en charge de nombreuses applications sûres de couche de données différentes, chacune avec des exigences de connexité uniques.

4.4.2 Création d'un jeton de politique

Le GO crée et signe le jeton de politique pour un groupe. Le jeton de politique contient les règles de contrôle d'accès et d'autorisation pour un groupe particulier.

Le PT comporte les informations suivantes :
- Identification : cela permet une identification non ambiguë du PT et du groupe.
- Règles de contrôle d'accès : ces règles spécifient qui peut avoir accès aux clés de groupe.
- Règles d'autorisation : ces règles spécifient qui peut être un S-GC/KS.
- Mécanismes : ces règles spécifient les mécanismes de sécurité qui vont être utilisés par le groupe. Ceci est nécessaire pour s'assurer qu'il n'y a pas de maillon faible dans le profil de sécurité du groupe. Par exemple, pour IPsec, cela pourrait inclure les données de configuration SPD/SAD.
- L'authentification de la source du PT au GO : le PT est un objet de CMS signé, et cela permet à tous les GM de le vérifier.

4.4.3 Création d'un groupe

Le PT est envoyé à un GC/KS potentiel. Cela peut se produire de plusieurs façons, et la méthode de transmission sort du domaine d'application de GSAKMP. Le GC/KS potentiel va vérifier la signature du GO sur le PT pour s'assurer qu'il vient bien d'un GO de confiance. Ensuite, le GC/KS va vérifier qu'il est autorisé à devenir le GC/KS, sur la base des règles d'autorisation du PT. En supposant que le GC/KS fait confiance au PT, qu'il est autorisé à être GC/KS, et qu'il est configuré en local pour devenir un GC/KS pour un certain groupe et ce GO, alors le GC/KS va créer les clés nécessaires pour lancer le groupe. Le GC/KS va effectuer toutes les actions nécessaires (si il en est) pour annoncer sa capacité à distribuer la clé pour le groupe. Le GC/KS va alors se mettre à l'écoute des RTJ.

Le PT a un numéro de séquence. Chaque fois qu'un PT est distribué au groupe, les membres du groupe vérifient que le numéro de séquence sur le PT est croissant. La durée de vie du PT n'est pas limitée par un intervalle de temps particulier, autre que les durées de vie imposées par certains de ses attributs (par exemple, durée de vie de clé de signature). Le numéro de séquence actuel du PT est téléchargé au GM dans le message "Téléchargement de clé". Aussi, pour éviter des attaques en répétition, ce numéro de séquence n'est jamais remis à une valeur inférieure (c'est-à-dire, ne revient jamais à zéro) tant que l'identifiant de groupe reste valide et utilisé. Le GO DOIT préserver ce numéro de séquence à travers les réamorçages.

4.4.4 Découverte de GC/KS

Les GM potentiels vont recevoir l'avis du nouveau groupe via un mécanisme : annonce, envoi à la cantonade, ou recherche d'enregistrement. Le GM va envoyer une RTJ au GC/KS.

4.4.5 Application de la politique d'enregistrement au GC/KS

Le GC/KS peut ou non exiger des mouchards, selon l'environnement de DoS et la configuration locale.

Une fois que la RTJ a été reçue, le GC/KS va vérifier que le GM est admis à avoir accès aux clés de groupe. Le GC/KS va alors vérifier la signature sur la RTJ pour s'assurer qu'elle a été envoyée par l'identité prétendue. Si la vérification réussit, le GC/KS va envoyer un message Téléchargement de clé pour le GM. Sinon, le GC/KS peut notifier au GM un problème ne relevant pas de la sécurité.

4.4.6 Application de la politique d'enregistrement au GM

À réception du message Téléchargement de clé, le GM va vérifier la signature sur le message. Puis le GM va récupérer le PT dans le message Téléchargement de clé et vérifier que le GO a créé et signé le PT. Une fois vérifiée la validité du PT, le GM va vérifier que le GC/KS est autorisé à distribuer la clé pour ce groupe. Puis le GM va vérifier que les mécanismes utilisés dans le groupe sont disponibles et acceptables pour la protection des données des GM (en supposant que le GM est une source de données). Le GM va alors accepter son adhésion à ce groupe.

Le GM va alors vérifier si il est admis comme S-GC/KS pour ce groupe. Si le GM est admis à être un S-GC/KS ET si la configuration locale du GM lui permet d'agir comme S-GC/KS pour ce groupe, alors le GM change son état de fonctionnement en S-GC/KS. Le GO doit allouer l'autorité de devenir S-GC/KS d'une manière qui prend en charge l'intégrité globale et le fonctionnement du groupe.

4.4.7 Opérations GSAKMP autonomes réparties

En mode autonome, chaque S-GC/KS opère sur un sous groupe largement auto contenu pour lequel le GC/KS principal délègue la responsabilité de la gestion des membres du sous groupe au S-GC/KS. En général, le S-GC/KS traite en local l'enregistrement et le désenregistrement de chaque membre du groupe sans aucune interaction avec le GC/KS principal. Périodiquement, le GC/KS principal envoie en diffusion groupée un message Événement de changement de clé adressé seulement à son ou ses S-GC/KS.

Après qu'un S-GC/KS a traité avec succès un message Événement de changement de clé provenant du GC/KS principal, le S-GC/KS transmet à son sous groupe son propre message Événement de changement de clé contenant une copie de la nouvelle GTPK et du nouveau jeton de politique du groupe. Le S-GC/KS chiffre les informations de gestion des clés du sous groupe de son message Événement de changement de clé en utilisant la hiérarchie logique de clés ou un protocole de changement de clés comparable. Le S-GC/KS utilise le protocole de changement de clé pour réaliser le secret vers l'avant et vers l'arrière, afin que seuls les membres autorisés du sous du groupe puissent déchiffrer et acquérir l'accès à la nouvelle GTPK et au nouveau jeton de politique. La fréquence de transmission par le GC/KS principal du message Événement de changement de clé est un paramètre de jeton de politique.

Pour le cas particulier d'un S-GC/KS qui détecte un membre expulsé ou compromis du groupe, un mécanisme est défini pour déclencher un changement immédiat de clés de groupe plutôt que d'attendre l'achèvement de la période de changement de clés du groupe. Voir les détails ci-après.

Chaque S-GC/KS va être enregistré par le GC/KS comme nœud de gestion avec la responsabilité de la distribution de la GTPK, de l'application de la politique de contrôle d'accès, de la création de l'arborescence de LKH, et de la distribution des matrices de clés de LKH. Le S-GC/KS va être enregistré dans l'arborescence principale de LKH comme point d'extrémité. Chaque S-GC/KS va détenir une matrice complète de clé LKH pour l'arborescence de clé LKH du GC.

Dans un souci de clarté, le processus de création d'un groupe GSAKMP réparti est expliqué en ordre chronologique.

D'abord, le propriétaire du groupe va créer un jeton de politique qui autorise un sous ensemble des membres du groupe à assurer le rôle de S-GC/KS.

Le GO doit s'assurer que les règles du S-GC/KS dans le jeton de politique vont être assez strictes pour assurer la confiance dans les S-GC/KS. Ce jeton de politique est écarté du GC principal.

Le GC va créer la GTPK et initialiser l'arborescence de clés de LKH. Le GC va ensuite attendre qu'un S-GC/KS potentiel envoie un message de demande d'adhésion (RTJ).

Un S-GC/KS potentiel va éventuellement envoyer une RTJ. Le GC va appliquer la politique de contrôle d'accès comme définie dans le jeton de politique. Le S-GC/KS va accepter le rôle de S-GC/KS et créer sa propre arborescence de clés de LKH pour les membres de son sous groupe.

Le S-GC/KS va alors offrir des services d'enregistrement pour le groupe. Ce sont des décisions de gestion locale qui sont facultatives pour contrôler la portée des membres du groupe qui peuvent être desservis par un S-GC/KS. Ce sont des questions de gestion vraiment locales qui permettent aux administrateurs d'un S-GC/KS de restreindre le service aux GM potentiels. Ces contrôles locaux n'affectent pas la politique de sécurité globale du groupe, comme définie dans le jeton de politique.

Un membre potentiel du groupe va envoyer une RTJ au S-GC/KS. Le S-GC/KS va appliquer la politique de contrôle d'accès toute entière comme définie dans le PT. Le GM va recevoir une matrice de clés LKH qui correspond à l'arborescence de LKH du S-GC/KS. L'arborescence de clés générée par le S-GC/KS est indépendante de l'arborescence de clés générée par le GC/KS ; ils n'ont pas de clés en commun.

Le GM a alors les clés dont il a besoin pour recevoir le trafic du groupe et être soumis au changement de clés provenant du S-GC/KS. Pour les besoins de la présente discussion, on suppose que le GM va être expulsé du groupe.

Le S-GC/KS va recevoir une notification que le GM est à expulser. Ce mécanisme sort du domaine d'application du protocole.

À réception d'une notification qu'un GM qui détient une matrice de clés au sein de son arborescence de LKH doit être expulsé, le S-GC/KS fait deux choses. D'abord, le S-GC/KS initie un échange de désenregistrement avec le GC/KS qui identifie le membre à expulser. (Le S-GC/KS relaie le désenregistrement d'un membre du groupe informant le GC/KS que le membre du groupe a été expulsé du groupe.) Ensuite, le S-GC/KS va attendre une action de changement de clé par le GC/KS. L'immédiateté de l'action de changement de clé par le GC/KS est une décision de gestion du GC/KS. La sécurité est mieux servie par une expulsion rapide des membres qui ne sont plus de confiance.

À réception de la notification de désenregistrement de la part du S-GC/KS, le GC/KS va enregistrer que le membre est expulsé. Le GC/KS va alors suivre la procédure du groupe pour initier une action de changement de clés (qui sort du domaine d'application de ce protocole). Le GC/KS va communiquer au GO les informations sur le membre expulsé (cela sort du domaine de ce protocole). Avec ces informations, le GO va créer un nouveau PT pour le groupe avec l'identité du GM expulsé ajoutée à la liste des exclus dans les règles de contrôle d'accès du groupe. Le GO fournit ce nouveau PT au GC/KS pour distribution avec le message Événement de changement de clé.

Le GC/KS va lancer une opération de changement de clé avec un nouveau PT. Le S-GC/KS va recevoir le changement de clé et le traiter. Au même moment, tous les autres S-GC/KS vont recevoir le changement de clé et noter l'identité du GM exclu. Tous les S-GC/KS vont revoir les identités locales pour s'assurer que le GM exclu n'est pas un membre local. Si il l'est, le S-GC/KS va alors créer un message de changement de clé. Les S-GC/KS doivent toujours créer un message de changement de clé, que le membre de groupe expulsé soit ou non membre de leur sous arborescence.

Le S-GC/KS va alors créer un message de changement de clé local. Le S-GC/KS va envoyer la TPK de groupe enveloppée à tous les membres de son arborescence locale de LKH, sauf aux membres exclus.

5. Cycle de vie du groupe

La gestion d'un groupe cryptographique suit un cycle de vie : définition du groupe, établissement du groupe, et maintenance du groupe quant à la sécurité. La définition du groupe implique de définir les paramètres nécessaires pour prendre en charge un groupe sûr, incluant son jeton de politique. L'établissement du groupe est le processus d'accorder l'accès aux nouveaux membres. Les messages de maintenance du groupe quant à la sécurité incluent des changements de clés, des changements de politique, des suppressions de membres, et la destruction du groupe. Chacune de ces phases du cycle de vie est traitée dans les paragraphes qui suivent.

L'utilisation et le traitement de la charge utile facultative Identifiant de fabricant pour tous les messages se trouve au paragraphe 7.10.

5.1 Définition du groupe

Un groupe cryptographique est établi pour prendre en charge des communications sûres parmi un groupe d'individus. Les activités nécessaires pour créer un jeton de politique à l'appui d'un groupe cryptographique incluent :
- de déterminer une politique d'accès : identifier les entités qui sont autorisées à recevoir la clé du groupe ;
- de déterminer la politique d'autorisation : identifier quelles entités sont autorisées à effectuer les actions pertinentes pour la sécurité, incluant la dissémination de la clé, la création de la politique, et l'initiation des actions de gestion de la sécurité ;
- de déterminer les mécanismes : définir les algorithmes et protocoles utilisés par GSAKMP pour sécuriser le groupe ;
- de créer un jeton de politique de groupe : formater les politiques et mécanismes dans un jeton de politique, et appliquer la signature du GO.

5.2 Établissement du groupe

L'établissement de groupe GSAKMP consiste en trois messages de mise en œuvre obligatoire : la demande d'adhésion (RTJ, *Request to Join*), le téléchargement de clé (KeyDL, *Key Download*) et l'accusé de réception/échec de téléchargement de clé (KeyDL-A/F, *Key Download Ack/Failure*). L'échange peut aussi inclure deux messages d'erreur FACULTATIFS : les messages Erreur de demande d'adhésion et Absence d'accusé de réception. Le fonctionnement qui utilise seulement les messages obligatoires est appelé le "mode concis", tandis que l'inclusion des messages d'erreur est appelé le "mode verbeux". Les mises en œuvre de GSAKMP DOIVENT prendre en charge le mode concis et PEUVENT prendre en charge le mode verbeux. L'établissement de groupe est discuté au paragraphe 5.2.1.

Un groupe est établi en mode concis ou verbeux par un paramètre du jeton de politique. Tous les (S-)GC/KS dans un groupe en mode verbeux DOIT prendre en charge le mode verbeux. GSAKMP permet que les groupes en mode verbeux aient des GM qui ne prennent pas en charge le mode verbeux. Les GM candidats qui ne prennent pas en charge le mode verbeux et reçoivent un message RTJ-Erreur ou Absence d'accusé de réception doivent traiter ces messages en douceur. De plus, un GM ne va pas savoir à l'avance si il interagit avec le (S-)GC/KS en mode verbeux ou concis avant que le jeton de politique soit reçu.

Pour la protection contre le déni de service, un échange de mouchards PEUT précéder l'échange d'établissement de groupe. L'échange de mouchards est décrit au paragraphe 5.2.2.

Sans considération du mode, tout message d'erreur envoyé entre des membres composants indique la première erreur rencontrée lors du traitement du message.

5.2.1 Établissement de groupe standard

Après la réception hors bande d'un jeton de politique, un contrôleur de groupe/serveur de clés (GC/KS) potentiel vérifie le jeton et son éligibilité à effectuer la fonction de GC/KS. Il lui est alors permis de créer toutes les clés de groupe nécessaires et de commencer à établir le groupe.

Le diagramme d'échelonnement des messages GSAKMP à la Figure 1, illustre le processus d'établissement d'un groupe cryptographique. Le côté gauche du diagramme représente les actions du GC/KS. Le côté droit du diagramme représente les actions des GM. Les composants de chaque message montrés dans le diagramme sont présenté aux paragraphes 5.2.1.1 à 5.2.1.5.

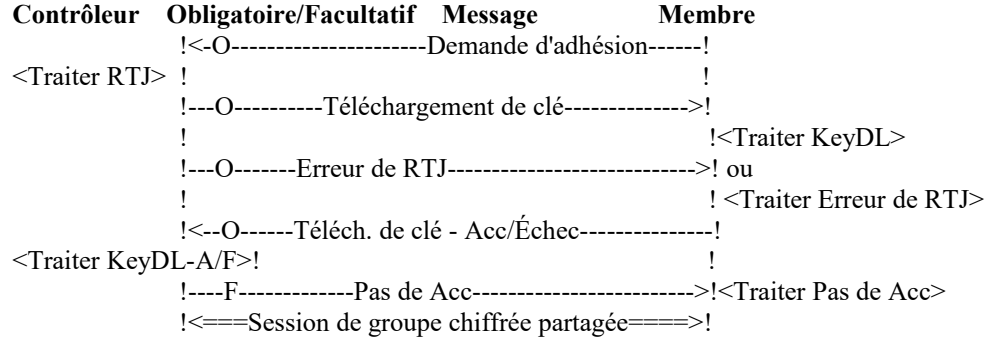


Figure 1 : Diagramme de l'échelonnement des messages GSAKMP

Le message Demande d'adhésion est envoyé d'un GM potentiel au GC/KS pour demander l'admission au groupe cryptographique. Le message contient le matériel de création de clé, la fraîcheur des données, un choix facultatif de mécanismes, et la signature du GM.

Le message Téléchargement de clé est envoyé du GC/KS au GM en réponse à une demande d'adhésion acceptée. Ce message signé du GC/KS contient l'identifiant du GM, des données sur la fraîcheur, le matériel de création de clé, les clés chiffrées et le jeton de politique chiffré. Le jeton de politique est utilisé pour faciliter une création de groupe bien ordonnée et DOIT inclure l'identification du groupe, les permissions du groupe, la politique d'adhésion au groupe, l'identité du serveur de clé/contrôleur du groupe, les informations de gestion du groupe, et la signature numérique du GO. Cela va permettre au GM de déterminer si la politique de groupe est compatible avec la politique locale.

Le message Erreur de demande d'adhésion est envoyé du GC/KS au GM en réponse à une demande d'adhésion non acceptée. Ce message n'est pas signé par le GC/KS pour deux raisons : 1) le GM, à ce point, n'a pas connaissance de qui est autorisé à agir comme GC/KS, et donc la signature n'aurait aucune signification pour le GM, et 2) signer les réponses pour refuser les demandes d'adhésion offrirait un potentiel d'attaque de déni de service. Le message contient l'indication de la condition d'erreur. Les valeurs possibles de cette condition d'erreur sont : Type de charge utile invalide, Version invalide, Identifiant de groupe invalide, Identifiant de séquence invalide, Charge utile mal formée, Informations d'identifiant invalides, Certificat invalide, Type de certificat non pris en charge, Autorité de certification invalide, Échec d'authentification, Certificat indisponible, Demande non autorisée, Interdit par la politique du groupe, et Interdit par la politique configurée localement.

Le message Échec d'accusé de réception de téléchargement de clé indique l'état de réception du téléchargement de clé chez le GM. C'est un message signé du GM qui contient des données de fraîcheur et d'état.

Le message Pas d'accusé de réception est envoyé du GC/KS au GM en réponse à un message Échec d'accusé de réception de téléchargement de clé invalide ou absent. Le message signé contient des données de fraîcheur et d'état et est utilisé pour avertir le GM d'une éviction imminente du groupe si un message Échec d'accusé de réception de téléchargement de clé valide n'est pas envoyé. L'éviction signifie que le membre va être exclu du groupe après le prochain événement de changement de clé. La politique sur quand un groupe particulier a besoin de changer ses clés est déclarée dans le jeton de politique. L'éviction est discutée plus en détails au paragraphe 5.3.2.1.

Pour les paragraphes suivants sur la structure du message, les détails sur le format et le traitement de la charge utile se trouvent à la Section 7. Chaque message est identifié par son type d'échange dans l'en-tête du message. Les noms occasionnels *(nonces)* DOIVENT être présents dans les messages sauf si l'heure de synchronisation est disponible pour le système.

5.2.1.1 Demande d'adhésion

Le type d'échange pour la demande d'adhésion est huit (8).

Les composants d'un message Demande d'adhésion sont montrés dans le Tableau 1.

Tableau 1 : Définition du message Demande d'adhésion (RTJ)

Nom du message : Demande d'adhésion (RTJ, *Request to Join*)
Dissection : {HDR-GrpID, Création de clé, Nonce_I, [VendorID], : [Notif_Mechanism_Choices], [Notif_Cookie],
: [Notif_IPValue]} SigM, [Cert]
Types de charge utile : En-tête GSAKMP, Création de clé, [Nom occasionnel], [VendorID], Signature, [Certificat],
[Notifications]

SigM : Signature du membre du groupe
Cert : Certificats nécessaires, zéro, un ou plusieurs
{}SigX : Indique les champs utilisés dans la signature
[] : Indique un élément de données facultatif

Comme montré à la Figure 1, un GM potentiel DOIT générer et envoyer un message RTJ pour demander la permission de se joindre au groupe. Au minimum, le GM DOIT être capable de configurer manuellement la destination pour la RTJ. comme définie dans la dissection du message RTJ, ce message DOIT contenir une charge utile Création de clé pour la détermination de la KEK. Une charge utile Nom occasionnel DOIT être incluse pour la fraîcheur et la valeur Nonce_I DOIT être sauvegardée pour un usage ultérieur potentiel. Le GC/KS ne va utiliser ce nom occasionnel fourni que si le jeton de politique pour ce groupe définit l'utilisation de noms occasionnels plutôt qu'une heure de synchronisation. Une charge utile FACULTATIVE Notification de type Choix de mécanisme PEUT être incluse pour identifier les mécanismes que le GM veut utiliser. L'absence de cette charge utile va causer le choix par le GC/KS des mécanismes appropriés spécifiés par défaut par le jeton de politique pour le téléchargement de clé.

En réponse, le GC/KS accepte ou refuse la demande sur la base de la configuration locale. <Traiter la RTJ> indique les actions du GC/KS qui vont déterminer si la RTJ va être traitée. Les vérifications suivantes DEVRAIENT être effectuées dans l'ordre présenté.

Dans cette procédure, le GC/KS DOIT vérifier que l'en-tête du message est correctement formé et confirmer que ce message est pour ce groupe en vérifiant la valeur de l'identifiant de groupe. Si la vérification d'en-tête réussit, l'identité de l'envoyeur est alors extraite de la charge utile Signature. Cette identité DOIT être utilisée pour effectuer les vérifications de contrôle d'accès et trouver les accréditifs des GM (par exemple, un certificat) pour les vérifications de message. Elle DOIT aussi être utilisée dans le message Téléchargement de clé. Ensuite, le GC/KS va vérifier la signature sur le message pour s'assurer de son authenticité. Le GC/KS DOIT utiliser du matériel d'authentification vérifié et de confiance provenant d'une racine connue. Si la signature du message est vérifiée, le GC/KS confirme alors que toutes les charges utiles requises sont présentes et proprement formatées sur la base des mécanismes annoncés et/ou demandés. Si toutes les vérifications réussissent, le GC/KS va créer et envoyer le message Téléchargement de clé comme décrit au paragraphe 5.2.1.2.

Si le GM ne reçoit pas de réponse à la RTJ dans la valeur de temporisation localement configurée du GM, le GM DEVRAIT envoyer à nouveau le message RTJ jusqu'à trois (3) fois.

Note : À tout moment, un GC/KS NE DOIT traiter pas plus d'un (1) message RTJ valide provenant d'un certain GM par groupe jusqu'à ce que son échange de protocole d'enregistrement se conclut.

Si une erreur se produit durant le traitement du message RTJ, et si le GC/KS fonctionne en mode concis, la session d'enregistrement DOIT être terminée, et toutes les informations d'état sauvegardées DOIVENT être éliminées.

La charge utile FACULTATIVE Notification de type Mouchard est discutée au paragraphe 5.2.2.

La charge utile FACULTATIVE Notification de type Valeur IP peut être utilisée par le GM pour porter une valeur IP spécifique au GC/KS.

5.2.1.2 Téléchargement de clé

Le type d'échange pour Téléchargement de clé est neuf (9).

Les composants d'un message Téléchargement de clé sont montrés au Tableau 2:

Tableau 2 : Définition du message Téléchargement de clé

Nom du message : Téléchargement de clé (KeyDL *Key Download*)
Dissection : {HDR-GrpID, Identifiant de membre, [Nonce_R, Nonce_C], Création de clé, (jeton de politique)*,
(Téléchargement de clé)*, [VendorID]} SigC, [Cert]
Types de charge utile : En-tête GSAKMP, Identification, [Nom occasionnel], Création de clé, Jeton de politique,
Téléchargement de clé, [VendorID], Signature, [Certificat]

SigC : Signature du Contrôleur de groupe/Serveur de clés
Cert : certificats nécessaires, zéro, un ou plusieurs
{}SigX : indique les champs utilisés dans Signature
[] : indique un élément de données facultatif
(xxx)* : indique des informations chiffrées

En réponse à un message RTJ formé et vérifié de façon appropriée, le GC/KS crée et envoie le message KeyDL. Comme défini dans la dissection du message, ce message DOIT contenir des charges utiles contenant les informations suivantes : identification du GM, matériel de création de clé, jeton de politique chiffré, informations de clé chiffrées, et informations de signature. Si l'heure synchronisée n'est pas disponible, les charges utiles Nom occasionnel DOIVENT être incluses dans le message pour en vérifier la fraîcheur.

Si elles sont présentes, les valeurs de nom occasionnel transmises DOIVENT être la valeur Nonce_R générée par le GC/KS et la valeur Nonce_C combinée qui a été générée en utilisant la valeur Nonce_R du GC/KS et la valeur et Nonce_I reçue du GM dans la RTJ.

Si la détermination de clé en deux parties est utilisée, le matériel de création de clé fourni par le GM et/ou le GC/KS va être utilisé pour générer la clé. La génération de cette clé dépend de l'échange de clé, comme défini au paragraphe 7.11, "Charge utile Création de clé". Le jeton de politique et le matériel de clé sont chiffrés dans la clé générée.

Le GM DOIT être capable de traiter le message Téléchargement de clé. <Traiter KeyDL> indique les actions du GM qui vont déterminer comment le message Téléchargement de clé va être traité. Les vérifications suivantes DEVRAIENT être effectuées dans l'ordre présenté.

Dans cette procédure, le GM va vérifier que l'en-tête de message est proprement formé et confirmer que ce message est pour ce groupe en vérifiant la valeur de l'identifiant de groupe. Si la vérification de l'en-tête réussit, le GM DOIT confirmer que ce message lui était destiné en comparant l'identifiant de membre dans la charge utile Identification à son identité.

Après confirmation de l'identification, les valeurs de fraîcheur sont vérifiées. Si on utilise des noms occasionnels, le GM DOIT utiliser sa valeur de Nonce_I sauvegardée, extraite de la valeur Nonce_R reçue du GC/KS, calculer la valeur combinée de Nonce_C, et la comparer à la valeur de Nonce_C reçue. Si on utilise pas de noms occasionnels, le GM DOIT vérifier l'horodatage dans la charge utile Signature pour déterminer si le message est nouveau.

Après la confirmation de la fraîcheur, la signature DOIT être vérifiée pour s'assurer de son authenticité. Le GM DOIT utiliser du matériel d'authentification vérifié et de confiance provenant d'une racine connue. Si la signature du message est vérifiée, le matériel de création de clé est extrait de la charge utile Création de clé pour générer la KEK. Cette KEK est alors utilisée pour déchiffrer les données du jeton de politique. La signature sur le jeton de politique DOIT être vérifiée. Les vérifications de contrôle d'accès DOIVENT être effectuées sur le GO et sur le GC/KS pour déterminer leur autorité au sein de ce groupe. Après la réussite de toutes ces vérifications, la KEK peut alors être utilisée pour déchiffrer et traiter le matériel de clé provenant de la charge utile Téléchargement de clé. Si tout est réussi, le GM va créer et envoyer le message Accusé de réception/échec de téléchargement de clé comme décrit au paragraphe 5.2.1.4.

Les charges utiles Jeton de politique et Téléchargement de clé sont envoyées chiffrées dans la KEK générée par les informations de charge utile Création de clé en utilisant les mécanismes définis dans l'annonce de groupe. Cela garantit que la politique sensible et les données de clé pour le groupe et les données de changement de clé potentiel pour cet individu ne peuvent pas être lues par quelqu'un d'autre que le receveur prévu.

Si une erreur se produit durant le traitement du message KeyDL, sans considération de si le GM est en mode concis ou verbeux, la session d'enregistrement DOIT être terminée, le GM DOIT envoyer un message Accusé de réception/échec de téléchargement de clé, et toutes les informations d'état sauvegardées DOIVENT être éliminées. En mode concis, la charge utile Notification va être du type NACK pour indiquer la terminaison. En mode verbeux, la charge utile Notification va contenir le type de l'erreur rencontrée.

5.2.1.3 Erreur de demande d'adhésion

Le type d'échange pour l'erreur de demande d'adhésion est onze (11).

Les composants du message Erreur de demande d'adhésion sont montrés au Tableau 3 :

Tableau 3 : Définition du message Erreur de demande d'adhésion (RTJ-Err)

Nom du message : Erreur de demande d'adhésion (RTJ-Err)
Dissection : {HDR-GrpID, [Nonce_I], Notification, [VendorID]}
Types de charges utiles : En-tête GSAKMP , [Nom occasionnel] Notification, [VendorID]

En réponse à une RTJ non acceptable, le GC/KS PEUT envoyer un message Erreur de demande d'adhésion (RTJ-Err, *Request to Join Error*) contenant une charge utile Notification appropriée. Noter que le message RTJ-Err n'est pas un message signé pour les raisons suivantes : l'absence de la connaissance par le GM de qui est un GC/KS valide ainsi que le besoin de protéger le GC/KS contre la signature des messages et la consommation de ressources précieuses. À la suite de l'envoi d'une RTJ-Err, le GC/KS DOIT terminer la session, et toutes les informations d'état sauvegardées DOIVENT être éliminées.

À réception d'un message RTJ-Err, le GM va valider ce qui suit : l'identifiant de groupe dans l'en-tête appartient à un groupe auquel le GM a envoyé une RTJ, et, si présent, le Nonce_I correspond au Nonce_I envoyé dans une RTJ à ce groupe. Si les vérifications ci-dessus réussissent, le GM PEUT terminer l'état associé à cet identifiant de groupe et nom occasionnel. Le GM DEVRAIT être capable de recevoir un message Téléchargement de clé valide pour cet identifiant de groupe et nom occasionnel après la réception d'une RTJ-Err pendant une durée configurée en local.

5.2.1.4 Accusé de réception/échec de téléchargement de clé

Le type d'échange pour Accusé de réception/échec de téléchargement de clé est quatre (4).

Les composants du message Accusé de réception/échec de téléchargement de clé sont montrés au Tableau 4:

Tableau 4 : Définition de message Accusé de réception/échec de téléchargement de clé (KeyDL-A/F)

Nom du message : Accusé de réception/échec de téléchargement de clé (KeyDL-A/F, *Key Download - Ack/Failure*)
Dissection : {HDR-GrpID, [Nonce_C], Notif_Ack, [VendorID]}SigM
Types de charge utile : En-tête GSAKMP, [Nom occasionnel], Notification, [VendorID], Signature

SigM : signature du membre du groupe
{}SigX : indique les champs utilisés dans Signature

En réponse à un message KeyDL correctement traité, le GM crée et envoie le message KeyDL-A/F. Comme défini dans la dissection du message, ce message DOIT contenir des charges utiles pour les informations suivantes : charge utile Notification du type Accusé de réception (ACK) et Informations de signature. Si l'heure synchronisée n'est pas disponible, la charge utile Nom occasionnel DOIT être présente pour la fraîcheur, et la valeur de nom occasionnel transmise DOIT être la valeur de Nonce_C générée par le GM. Si le GM ne reçoit pas de message KeyDL dans un délai configuré en local, le GM PEUT envoyer une nouvelle RTJ. Si le GM reçoit un message Absence d'accusé de réception (LOA, *Lack of Ack*) du GC/KS avant de recevoir un message KeyDL, le GM DEVRAIT envoyer un message KeyDL-A/F de type NACK suivi par une nouvelle RTJ.

Le GC/KS DOIT être capable de traiter le message KeyDL-A/F. <Traiter KeyDL-A/F> indique les actions du GC/KS qui vont déterminer comment le message KeyDL-A/F va être traité. Les vérifications suivantes DEVRAIENT être effectuées dans l'ordre présenté.

Dans cette procédure, le GC/KS va vérifier que l'en-tête de message est correctement formé et confirmer que ce message est pour ce groupe en vérifiant la valeur de l'identifiant de groupe. Si la vérification d'en-tête réussit, le GC/KS DOIT vérifier la fraîcheur du message. Si on utilise des noms occasionnels, le GC/KS DOIT utiliser sa valeur Nonce_C sauvegardée et la comparer pour égalité avec la valeur de Nonce_C reçue. Si on n'utilise pas de noms occasionnels, le GC/KS DOIT vérifier l'horodatage dans la charge utile Signature pour déterminer si le message est nouveau. Après la confirmation de la fraîcheur, la signature DOIT être vérifiée pour s'assurer de son authenticité. Le GC/KS DOIT utiliser du matériel d'authentification vérifié et de confiance provenant d'une racine connue. Si la signature du message se vérifie, le GC/KS traite la charge utile Notification. Si le type de notification est ACK, l'enregistrement s'est achevé avec succès, et

les deux parties DEVRAIENT supprimer les informations d'état associées à cet enregistrement de GM.

Si le GC/KS ne reçoit pas un message KeyDL-A/F de forme appropriée ou est incapable de traiter correctement le message KeyDL-A/F, le type de charge utile Notification est toute valeur sauf ACK ; ou si aucun message KeyDL-A/F n'est reçu dans le délai configuré en local, le GC/KS DOIT évincer ce GM du groupe dans le prochain événement de changement de clé défini par la politique. Le GC/KS PEUT envoyer le message FACULTATIF Absence d'accusé de réception quand il fonctionne en mode verbeux comme défini au paragraphe 5.2.1.5.

5.2.1.5 Absence d'accusé de réception (LOA)

Le type d'échange pour Absence d'accusé de réception est douze (12).

Les composants d'un message Absence d'accusé de réception sont montrés au Tableau 5:

Tableau 5 : Définition du message Absence d'accusé de réception (LOA, *Lack of Ack*)

Nom du message : Absence d'accusé de réception (LOA, *Lack of Ack*)
Dissection : {HDR-GrpID, Identifiant de membre, [Nonce_R, Nonce_C], Notification, [VendorID]} SigC, [Cert]
Types de charge utile : En-tête GSAKMP, Identification, [Nom occasionnel], Notification, [Vendor ID], Signature, [Certificat]

SigC : signature du contrôleur de groupe/serveur de clés
Cert : certificats nécessaires, zéro, un ou plusieurs
{}SigX : indique les champs utilisés dans Signature
[] : indique un élément de données facultatif

Si la valeur de la temporisation locale du GC/KS expire avant la réception d'un KeyDL-A/F de la part du GM, le GC/KS PEUT créer et envoyer un message LOA au GM. Comme défini dans la dissection du message, ce message DOIT contenir des charges utiles contenant les informations suivantes : identification du GM, notification d'erreur, et informations de signature.

Si l'heure synchronisée n'est pas disponible, les charges utiles Nom occasionnel DOIVENT être présentes pour la fraîcheur, et les valeurs de nom occasionnel transmises DOIVENT être la valeur Nonce_R générée par le GC/KS et la valeur combinée Nonce_C qui a été générée en utilisant la valeur Nonce_R du GC/KS et la valeur Nonce_I reçue du GM dans la RTJ. Ces valeurs ont déjà été générées durant la phase Téléchargement de clé du message.

Le GM PEUT être capable de traiter le message LOA sur la base de la configuration locale. <Traiter LOA> indique les actions du GM qui vont déterminer comment il va être agit sur le message LOA. Les vérifications suivantes DEVRAIENT être effectuées dans l'ordre présenté.

Dans cette procédure, le GM DOIT vérifier que l'en-tête du message est correctement formé et confirmer que ce message est pour ce groupe en vérifiant la valeur de l'identifiant de groupe. Si la vérification de l'en-tête réussit, le GM DOIT confirmer que ce message lui était destiné en comparant l'identifiant de membre dans la charge utile Identification à son identité. Après la confirmation de l'identification, les valeurs de fraîcheur sont vérifiées. Si il utilise des noms occasionnels, le GM DOIT utiliser sa valeur sauvegardée de Nonce_I, extraite de la valeur Nonce_R reçue du GC/KS, calculer la valeur combinée de Nonce_C, et la comparer à la valeur de Nonce_C reçue. Si il n'utilise pas de nom occasionnel, le GM DOIT vérifier l'horodatage dans la charge utile Signature pour déterminer si le message est nouveau. Après la confirmation de la fraîcheur, les vérifications de contrôle d'accès DOIVENT être effectuées sur le GC/KS pour déterminer son autorité au sein de ce groupe. Puis la signature DOIT être vérifiée pour s'assurer de son authenticité, Le GM DOIT utiliser du matériel d'authentification vérifié et de confiance provenant d'une racine connue.

Si les vérifications réussissent, le GM DEVRAIT renvoyer un KeyDL-A/F pour cette session.

5.2.2 Mouchards : établissement de groupe avec protection contre le déni de service

Ce paragraphe définit une capacité FACULTATIVE qui PEUT être mise en œuvre dans GSAKMP quant on utilise des groupes fondés sur IP. Les informations de ce paragraphe empruntent largement à la [RFC4306] car ce protocole a déjà réglé ce problème et GSAKMP copie ce concept. Ce paragraphe paraphrase ceux de la [RFC4306] modifiés pour GSAKMP pour définir l'objet des mouchards.

Un mode facultatif "Mouchard" est défini pour GSAKMP contre les attaques de DoS.

Le terme de "mouchard" a son origine dans Photuris de Karn et Simpson [RFC2522], proposition ancienne de gestion des clés avec IPsec. L'en-tête fixe de message ISAKMP inclut deux champs de huit octets intitulés "mouchards". Au lieu de placer ces données de mouchard dans l'en-tête, dans GSAKMP ces données sont déplacées dans une charge utile Notification.

Une attaque prévue contre GSAKMP est l'épuisement d'état et de CPU, où le GC/KS cible est inondé de demandes d'adhésion provenant d'adresses IP falsifiées. Cette attaque peut être rendue moins efficace si une mise en œuvre de GC/KS utilise une CPU minimale et n'engage pas d'état pour la communication avant de savoir si le GM potentiel initiateur peut recevoir des paquets à l'adresse d'où il prétend qu'il les envoie. Pour réaliser cela, le GC/KS (quand il fonctionne en mode mouchard) DEVRAIT rejeter les message initiaux de demande d'adhésion si ils ne contiennent pas une charge utile Notification de type "mouchard". Il DEVRAIT plutôt envoyer un message Téléchargement de mouchard en réponse à la RTJ et inclure un mouchard dans une charge utile Notification de type Mouchard_exigé. Les GM potentiels qui reçoivent de telles réponses DOIVENT réessayer le message Demande d'adhésion avec le mouchard fourni par le GC/KS qui répond dans sa charge utile Notification de type Mouchard, comme défini dans la charge utile Notification facultative du message Demande d'adhésion du paragraphe 5.2.1.1. Cet échange initial va être alors comme montré à la Figure 2 avec les composants du nouveau message Téléchargement de mouchard montré au Tableau 6. Le type d'échange pour Téléchargement de mouchard est dix (10).

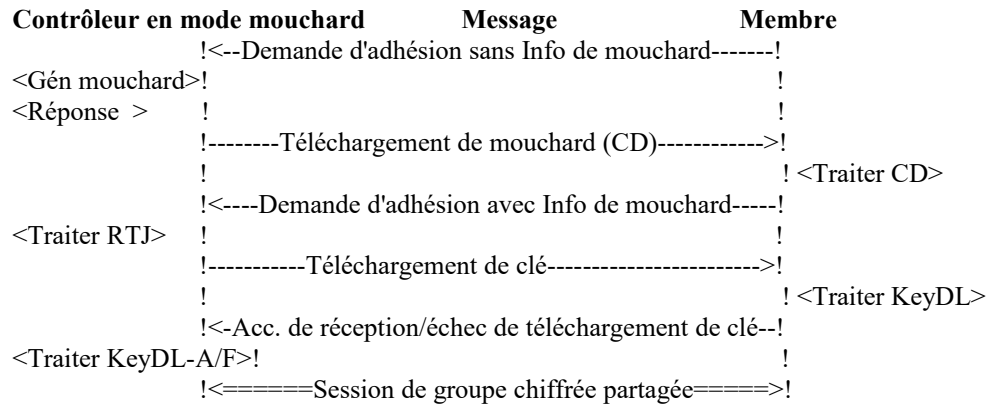


Figure 2 : Diagramme d'échelle GSAKMP avec mouchards

Tableau 6 : Définition du message Téléchargement de mouchard

Nom du message : Téléchargement de mouchard
Dissection : {HDR-GrpID, Notif_MOUCHARD_EXIGÉ, [VendorID]}
Types de charge utile : En-tête GSAKMP, Notification, [VendorID]

Les deux premiers messages n'affectent aucun état de GM ou du GC/KS sauf pour communiquer le mouchard.

Une mise en œuvre de GSAKMP DEVRAIT appliquer sa génération de mouchard de GC/KS de telle façon qu'elle n'exige de sauvegarder aucun état pour reconnaître son mouchard valide quand le second message Demande d'adhésion arrive. Les algorithmes et la syntaxe exacte qu'elle utilise pour générer les mouchards n'affectent pas l'interopérabilité et donc ne sont pas spécifiés ici.

Voici un exemple de la façon dont un point d'extrémité pourrait utiliser les mouchards pour mettre en œuvre une protection limitée contre le DoS.

Une bonne façon de faire est de régler le mouchard à être :

Mouchard = <NuméroDeVersionSecret> | Hachage(Ni | IPi | <secret>)

où <secret> est un secret généré au hasard connu seulement du GC/KS qui répond et changé périodiquement, Ni est la valeur du nom occasionnel tiré du GM potentiel initiateur, et IPi est l'adresse IP affirmée par le GM candidat. L'adresse IP est soit l'adresse IP de source de l'en-tête IP, soit l'adresse IP contenue dans la charge utile facultative Notification "IPvalue" (si elle est présente). <NuméroDeVersionSecret> devrait être changé chaque fois que <secret> est régénéré. Le

mouchard peut être recalculé quand arrive la "Demande d'adhésion avec mouchard" et qu'il est comparé au mouchard dans le message reçu. Si ils correspondent, le GC/KS qui répond sait que toutes les valeurs ont été calculées depuis le dernier changement du <secret> et que IPi DOIT être la même que l'adresse de source qu'il a vu la première fois. Incorporer Ni dans le hachage assure qu'un attaquant qui voit seulement le message Téléchargement de mouchard ne peut pas réussir à falsifier un message "Demande d'adhésion avec informations de mouchard". Cette valeur de Ni DOIT être la même valeur de Ni que dans le message original "Demande d'adhésion" pour que le calcul réussisse.

Si une nouvelle valeur pour <secret> est choise alors que les connexions sont en cours d'initialisation, une "Demande d'adhésion avec informations de mouchard" pourrait être retournée avec un <NuméroDeVersionSecret> autre que l'actuel. Le GC/KS qui répond PEUT dans ce cas rejeter le message en envoyant une autre réponse avec un nouveau mouchard, ou il PEUT garder la vieille valeur de <secret> pendant un bref instant et accepter les mouchards calculés à partir de l'une ou l'autre. Le GC/KS qui répond NE DEVRAIT PAS accepter indéfiniment des mouchards après que <secret> est changé, car cela réduirait une partie de la protection contre le déni de service. Le GC/KS qui répond DEVRAIT changer la valeur de <secret> fréquemment, en particulier en cas d'attaque.

Un autre exemple de génération de valeur de mouchard dans un environnement de NAT est de substituer à la valeur de IPi la valeur IP reçue dans la charge utile Notification dans le message RTJ. Ce scénario est indiqué par la présence de la charge utile Notification de type IPValue. Avec cette substitution, un calcul similaire à celui décrit ci-dessus peut être utilisé.

5.2.3 Établissement de groupe pour membres en réception seule

Ce paragraphe décrit une capacité FACULTATIVE qui peut être mise en œuvre dans un système structuré où le (S-)GC/KS est connu à l'avance par des moyens hors bande et où l'heure synchronisée est disponible.

À la différence de l'établissement de groupe standard, dans le système à réception seule, les GM et les (S-)GC/KS opèrent en mode concis et échangent seulement un message : le téléchargement de clé. Les nouveaux GM potentiels n'envoient pas de RTJ. Les (S-)GC/KS n'attendent pas de message Accusé de réception/échec de téléchargement de clé et ne suppriment pas de GM pour manque ou réception du message.

Le fonctionnement est le suivant : sur notification via un événement autorisé hors bande, le (S-)GC/KS forme et envoie un message Téléchargement de clé au nouveau membre avec les charges utiles Nom occasionnels ABSENTES. Le GM vérifie que :
- la charge utile ID identifie ce GM,
- l'horodatage dans le message est frais,
- le message est signé par un (S-)GC/KS autorisé,
- la signature se vérifie sur le message.

Quand on utilise un type de création de clé Diffie-Hellman pour les membres en réception seule, un modèle statique éphémère est supposé : la charge utile Création de clé dans le message Téléchargement de clé contient le composant public du (S-)GC/KS. Le composant public du membre est supposé être obtenu par des moyens hors bande sûrs.

5.3 Maintenance de groupe

La phase de maintenance du groupe inclut des adhésions et des départs de membres, des activités de changement de clés de groupe, des mises à jour de politique, et des destructions de groupes. Ces activités sont présentées dans les paragraphes suivants.

5.3.1 Gestion de groupe

5.3.1.1 Événements de changement de clé

Un événement de changement de clé est toute action, incluant un rapport de compromission ou une expiration de clé, qui exige la création d'une nouvelle clé de groupe et/ou des informations de changement de clé.

Une fois qu'un événement a été identifié (comme défini dans le jeton de politique de sécurité de groupe) le GC/KS DOIT créer et fournir un message signé contenant la GTPK et les informations de changement de clé pour le groupe.

Chaque GM qui reçoit ce message DOIT vérifier la signature sur le message pour s'assurer de son authenticité. Si la signature du message ne correspond pas, le message DOIT être éliminé. Après vérification, le GM va trouver le paquet approprié de téléchargement de changement de clé et va déchiffrer les informations avec une ou des clés de changement de clé mémorisées. Si un nouveau jeton de politique est distribué avec le message, il DOIT être chiffré avec la vieille GTPK.

Le type d'échange pour Événement de changement de clé est cinq (5).

Les composants d'un message Événement de changement de clé sont montrés au Tableau 7:

Tableau 7 : Définition du message Événement de changement de clé

Nom du message : Événement de changement de clé
Dissection : {HDR-GrpID, ([jeton de politique])*, Rekey Array, [VendorID]}SigC, [Cert]
Types de charge utile : En-tête GSAKMP, [jeton de politique], Événement de changement de clé, [VendorID], Signature, [Certificat],

SigC : signature du contrôleur de groupe/serveur de clés
Cert : certificats nécessaires, zéro, un ou plusieurs
{}SigX : indique les champs utilisés dans Signature
(xxx)* : indique des informations chiffrées
[] : indique un élément de données facultatif

5.3.1.2. Mises à jour de politique

Les nouveaux jetons de politique sont envoyés via le message Événement de changement de clé. Ces mises à jour de politique peuvent être couplées à un événement existant de changement de clé ou peuvent être envoyées dans un message avec le type d'événement de changement de clé de Charge utile d'événement de changement de clé réglé à Aucun (0) (voir au paragraphe 7.5.1).

Un jeton de politique NE DOIT PAS être traité si le traitement du message Événement de changement de clé qui le porte échoue. Le traitement du jeton de politique dépend du type et sort du domaine d'application du présent document.

5.3.1.3 Destruction de groupe

La destruction du groupe est aussi réalisée via le message Événement de changement de clé. Dans un message Événement de changement de clé pour une destruction de groupe, l'identifiant de séquence est réglé à 0xFFFFFFFF. À réception de ce message Événement de changement de clé authentifié, les composants du groupe DOIVENT terminer le traitement des informations associées au groupe indiqué.

5.3.2 Sortie d'un groupe

Il y a plusieurs conditions dans lesquelles un membre va quitter un groupe : éviction, départ volontaire sans avis, et départ volontaire avec avis (désenregistrement). Chacune d'elles est discutée dans les paragraphes qui suivent.

5.3.2.1 Éviction

À un moment de la vie du groupe, il peut être désirable d'évincer un ou plusieurs membres d'un groupe. Du point de vue de la gestion des clés, cela implique de révoquer l'accès aux données protégées du groupe en "désactivant" les clés des membres qui partent. Ceci est accompli avec un événement de changement de clé, qui est discuté plus en détails au paragraphe 5.3.1.1. Si le futur accès au groupe va aussi être refusé, les membres DOIVENT être ajoutés à une liste de contrôle d'accès refusé, et les règles d'autorisation du jeton de politique DOIVENT être mises à jour de façon appropriée afin qu'elles excluent le ou les GM expulsés. Après la réception d'un nouveau PT, les GM DEVRAIENT évaluer le niveau de confiance de toutes les données d'application récentes originaires du ou des GM expulsés.

5.3.2.2 Départ volontaire sans avis

Si un membre souhaite quitter un groupe pour lequel la qualité de membre n'impose pas de coût ni de responsabilité à ce membre, celui-ci PEUT alors simplement supprimer les copies locales des clés de groupe et cesser les activités de groupe.

5.3.2.3 Désenregistrement

Si l'adhésion au groupe n'impose pas de coût ou de responsabilités au membre partant, le membre DEVRAIT alors se désenregistrer du groupe quand il souhaite partir. Le désenregistrement consiste en un échange de trois messages entre le GM et le GC/KS du membre : la Demande de départ, la Réponse de départ, et l'Accusé de réception de départ. Les mises en œuvre conformes de GSAKMP pour les GM DEVRAIENT prendre en charge les messages de désenregistrement. Les mises en œuvre conformes de GSAKMP pour les GC/KS DOIVENT prendre en charge les messages de désenregistrement.

5.3.2.3.1 Demande de départ

Le type d'échange pour un message Demande de départ est treize (13). Les composants d'un message Demande de départ sont montrés au Tableau 8.

Tout GM qui désire initier le processus de désenregistrement DOIT générer et envoyer un message RTD pour notifier son intention au GC/KS. Comme défini dans la dissection du message RTD, ce message DOIT contenir des charges utiles pour les informations suivantes : l'identification du GC/KS et la notification du désir de quitter le groupe. Quand l'heure de synchronisation n'est pas disponible pour le système comme défini par le jeton de politique, une charge utile Nom occasionnel DOIT être incluse pour la fraîcheur, et la valeur Nonce_I DOIT être sauvegardée pour usage ultérieur. Ce message DOIT ensuite être signé par le GM.

Tableau 8 : Définition du message Demande de départ (RTD, *Request_to_Depart*)

Nom du message : Demande de départ (RTD)
Dissection : {HDR-GrpID, GC/KS_ID, [Nonce_I], Notif_Leave_Group, [VendorID]} SigM, [Cert]
Types de charge utile : En-tête GSAKMP, Identification, [Nom occasionnel], Notification, [VendorID], Signature, [Certificat]

SigM : Signature du membre du groupe
Cert : certificats nécessaires, zéro, un ou plusieurs
{}SigX : indique les champs utilisés dans Signature
[] : indique un élément de données facultatif

À réception du message RTD, le GC/KS DOIT vérifier que l'en-tête du message est proprement formé et confirmer que ce message est pour ce groupe en vérifiant la valeur de l'identifiant de groupe. Si la vérification d'en-tête réussit, la valeur de l'identifiant dans la charge utile Identification est comparée à la sienne, l'identité du GC/KS, pour confirmer que le GM a l'intention de converser avec ce GC/KS, celui qui a enregistré ce membre dans le groupe. Puis l'identité de l'envoyeur est extraite de la charge utile Signature. Cette identité DOIT être utilisée pour confirmer que ce GM est membre du groupe desservi par ce GC/KS. Puis le GC/KS va confirmer à partir de la charge utile Notification que le GM demande à quitter le groupe. Ensuite, le GC/KS va vérifier la signature sur le message pour s'assurer de son authenticité. Le GC/KS DOIT utiliser du matériel d'authentification vérifié et de confiance sur une racine connue. Si toutes les vérifications réussissent et si le message est traité avec succès, le GC/KS DOIT alors former un message Réponse de départ comme défini au paragraphe 5.3.2.3.2.

Si le traitement du message échoue, la session de désenregistrement DOIT être terminée, et tout état associé à cette session est supprimé. Si le GC/KS fonctionne en mode concis, aucun message d'erreur n'est alors envoyé au GM. Si le GC/KS fonctionne en mode verbeux, le GC/KS envoie alors un message Réponse de départ avec une charge utile Notification de type Erreur de demande de départ

5.3.2.3.2 Réponse de départ

Le type d'échange pour un message Réponse de départ (DR) est quatorze (14). Les composants du message Réponse de départ sont montrés au Tableau 9.

En réponse à un message RTD proprement formé et vérifié, le GC/KS DOIT créer et envoyer le message DR. Comme défini dans la dissection du message, ce message DOIT contenir des charges utiles avec les informations suivantes : identification du GM, notification de l'acceptation du départ, et informations de signature. Si l'heure de synchronisation n'est pas disponible, les charges utiles Nom occasionnel DOIVENT être incluses dans le message pour la fraîcheur.

Tableau 9 : Définition du message Réponse de départ (DR)

Nom du message : Réponse de départ (DR)
Dissection : {HDR-GrpID, Identifiant de membre, [Nonce_R, Nonce_C], Notification, [VendorID]} SigC, [Cert]
Types de charge utile : En-tête GSAKMP, Identification, [Nom occasionnel], Notification, [VendorID], Signature, [Certificat]

SigC : Signature du membre du groupe
Cert : certificats nécessaires, zéro, un ou plusieurs
{}SigX : indique les champs utilisés dans Signature
[] : indique un élément de données facultatif

Si il en est de présentes, les valeurs de nom occasionnel transmises DOIVENT être la valeur de Nonce-R générée par le GC/KS et la valeur combinée de Nonce_C qui a été générée en utilisant la valeur de Nonce_R du GC/KS et la valeur de Nonce_I reçue du GM dans le RTD. Cette valeur de Nonce_C DOIT être sauvegardée par rapport à l'identifiant du GM qui part.

Le GM DOIT être capable de traiter le message Réponse de départ. Les vérifications suivantes DEVRAIENT être effectuées dans l'ordre présenté.

Le GM DOIT vérifier que l'en-tête de message est proprement formé et confirmer que ce message est pour ce groupe en vérifiant la valeur de l'identifiant de groupe. Si la vérification d'en-tête réussit, le GM DOIT confirmer que ce message lui était destiné en comparant l'identifiant de membre dans la charge utile Identification à son identité. Après la confirmation de l'identification, les valeurs de fraîcheur sont vérifiées. Si il utilise les noms occasionnels, le GM DOIT utiliser sa valeur de Nonce_I sauvegardée, extraite de la valeur reçue du Nonce_R du GC/KS, calculer la valeur combinée de Nonce_C, et la comparer pour égalité avec la valeur reçue de Nonce_C. Si il n'utilise pas les noms occasionnels, le GM DOIT vérifier l'horodatage dans la charge utile de signature pour déterminer si le message est nouveau. Après la confirmation de la fraîcheur, la confirmation que l'identité du signataire du message DR est le GC/KS autorisé pour les GM est effectuée. Ensuite, la signature DOIT être vérifiée pour s'assurer de son authenticité. Le GM DOIT utiliser du matériel d'authentification vérifié et de confiance provenant d'une racine connue. Si la signature du message se vérifie, le GM DOIT alors vérifier que la Notification est du type Départ accepté ou Erreur de demande de départ.

Si le traitement est réussi, et si la charge utile Notification est du type Départ acepté, le membre DOIT former le message Accusé de réception de départ comme défini au paragraphe 5.3.2.3.3. Si le traitement est réussi, et si la charge utile Notification est du type Erreur de demande de départ, le membre DOIT supprimer tous les états associés à la session de désenregistrement. Si le membre désire encore se désenregistrer du groupe, il DOIT recommencer le processus de désenregistrement.

Si le traitement du message échoue, la session de désenregistrement DOIT être terminée, et tout état associé à cette session est supprimé. Si le GM fonctionne en mode concis, un message Accusé de réception de départ avec une charge utile Notification de type NACK est envoyé au GC/KS. Si le GM fonctionne en mode verbeux, le GM envoie alors un message Accusé de réception de départ avec une charge utile Notification du type d'échec approprié.

5.3.2.3.3 Accusé de réception de départ

Le type d'échange pour un message Accusé de réception de départ est quinze (15). Les composants du message Accusé de réception de départ sont montrés dans le Tableau 10 :

Tableau 10 : Définition du message Accusé de réception de départ (DA)

Nom du message : Accusé de réception de départ (DA, *Departure_ACK*)
Dissection : {HDR-GrpID, [Nonce_C], Notif_Ack, [VendorID]}SigM
Types de charge utile : En-tête GSAKMP, [Nom occasionnel], Notification, [VendorID], Signature

SigM : Signature du membre du groupe
{}SigX : indique les champs utilisés dans Signature

En réponse à un message Réponse de départ correctement traité, le GM DOIT créer et envoyer le message Accusé de réception de départ. Comme défini dans la dissection du message, ce message DOIT contenir des charges utile pour porter les informations suivantes : charge utile Notification de type Accusé de réception (ACK) et informations de signature. Si

l'heure de synchronisation n'est pas disponible, la charge utile Nom occasionnel DOIT être présente pour la fraîcheur, et la valeur du nom occasionnel transmise DOIT être la valeur de Nonce_C générée par le GM.

À réception de l'accusé de réception de départ, le GC/KS DOIT effectuer les vérifications suivantes. Ces vérifications DEVRAIENT être effectuées dans l'ordre présenté.

Dans cette procédure, le GC/KS DOIT vérifier que l'en-tête de message est proprement formé et confirmer que ce message est pour ce groupe en vérifiant la valeur de l'identifiant de groupe. Si la vérification de l'en-tête réussit, le GC/KS DOIT vérifier la fraîcheur du message. Si il utilise des noms occasionnels, le GC/KS DOIT utiliser la valeur de Nonce_C qu'il a sauvegardée et la comparer à la valeur du Nonce_C reçue. Si il n'utilise pas de nom occasionnel, le GC/KS DOIT vérifier l'horodatage dans la charge utile de signature pour déterminer si le message est nouveau. Après confirmation de la fraîcheur, la signature DOIT être vérifiée pour s'assurer de son authenticité. Le GC/KS DOIT utiliser du matériel d'authentification vérifié et de confiance provenant d'une racine connue. Si la signature du message est vérifiée, le GC/KS traite la charge utile Notification. Si le type de notification est ACK, ceci est considéré comme un traitement réussi du message.

Si le traitement du message est réussi, le GC/KS DOIT supprimer le membre du groupe. Cela PEUT impliquer d'initier un événement de changement de clés pour le groupe.

Si le traitement du message échoue ou si aucun accusé de réception de départ n'est reçu, le GC/KS PEUT produire un message LOA.

6. Suite de sécurité

La suite n° 1 de définition de sécurité DOIT être prise en charge. D'autres définitions de suite de sécurité PEUVENT être définies dans d'autres spécifications Internet.

6.1 Hypothèses

Tous les GM potentiels auront assez d'informations disponibles pour utiliser la suite de sécurité correcte pour se joindre au groupe. Ceci peut être accompli par une suite par défaut bien connue, "Suite de sécurité n° 1 ", ou par l'annonce/envoi d'une autre suite.

6.2 Définition de la suite n° 1

Les mises en œuvre de GSAKMP DOIVENT prendre en charge la suite d'algorithmes et configurations ci-après. La définition de la suite n° 1 emprunte beaucoup à la définition du groupe 2 Oakley de IKE et à Oakley lui-même.

La définition de Suite 1 GSAKMP donne toutes les définitions d'algorithme et de chiffrement exigées pour traiter les messages d'établissement de groupe. Il est important de noter que GSAKMP ne négocie pas ces mécanismes de chiffrement. Cette définition est réglée par le propriétaire du groupe (GO, *Group Owner*) via le jeton de politique (passé durant l'échange GSAKMP pour les besoin de vérification des membres).

La définition de la suite n°1 GSAKMP est :

Définition de l'algorithme de téléchargement de clé et de chiffrement de jeton de politique :
Algorithme : AES
Mode : CBC
Longueur de clé : 128 bits

L'algorithme de signature numérique de jeton de politique est : DSS-ASN1-DER
L'algorithme de hachage est : SHA-1
L'algorithme de hachage de nom occasionnel est : SHA-1

La définition de création de clé est :
Le type d'algorithme est Diffie Hellman
Définition de groupe MODP :
g : 2

p : "FFFFFFFF FFFFFFFF C90FDAA2 2168C234 C4C6628B 80DC1CD1"
"29024E08 8A67CC74 020BBEA6 3B139B22 514A0879 8E3404DD"
"EF9519B3 CD3A431B 302B0A6D F25F1437 4FE1356D 6D51C245"
"E485B576 625E7EC6 F44C42E9 A637ED6B 0BFF5CB6 F406B7ED"
"EE386BFB 5A899FA5 AE9F2411 7C4B1FE6 49286651 ECE65381 FFFFFFFF FFFFFFFF"

Note : les valeurs p et g viennent du paragraphe 6.2 de la [RFC2409], "Second groupe Oakley", et p fait 1024 bits.

L'algorithme de signature numérique de message GSAKMP est : DSS-SHA1-ASN1-DER
Le type d'identifiant de signature numérique est : ID-DN-STRING

7. Structure de charge utile GSAKMP

Un message GSAKMP se compose d'un en-tête GSAKMP (paragraphe 7.1) suivi d'au moins une charge utile GSAKMP. Toutes les charges utiles GSAKMP sont composées de l'en-tête générique de charge utile (paragraphe 7.2) suivi par les données spécifiques de la charge utile. Le message est chaîné par une charge utile précédente qui définit la charge utile qui la suit. Les charges utiles ne sont pas obligées d'être dans l'ordre exact indiqué dans la dissection de message de la Section 5, pourvu que toutes celles requises soient présentes. Sauf si il est explicitement déclaré dans une dissection que plusieurs charges utiles d'un seul type peuvent être présentes, pas plus d'une charge utile de chaque type permis par le message ne doit apparaître. La charge utile finale dans un message ne va pointer sur aucune charge utile suivante.

Tous les champs de type entier dans la structure En-tête et Charge utile qui font plus d'un octet DOIVENT être convertis en l'ordre des octets du réseau avant la transmission des données.

Aucun bourrage des champs NE DOIT être fait car cela conduit à des erreurs de traitement.

Quand un message contient une charge utile VendorID, le traitement des charges utiles de ce message est modifié comme défini au paragraphe 7.10.

7.1 En-tête GSAKMP

7.1.1 Structure d'en-tête GSAKMP

Les champs d'en-tête GSAKMP sont montrés à la Figure 3 et définis comme :

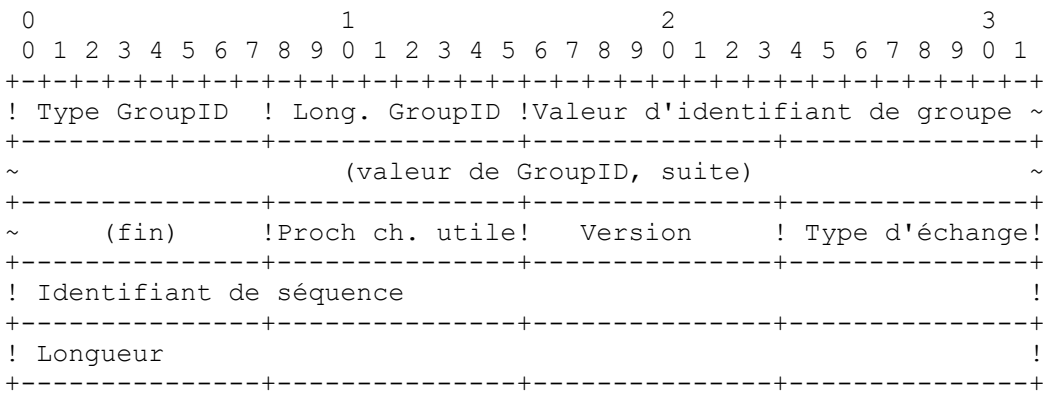


Figure 3 : Format d'en-tête GSAKMP

Type d'identifiant de groupe (1 octet) – le Tableau 11 présente les types d'identifiant de groupe. Ce champ est traité comme une valeur non signée.

Tableau 11 : Types d'identifiant de groupe

Type	Valeur	Description
Réservé	0	
UTF-8	1	Format défini au paragraphe 7.1.1.1.1.
Chaîne d'octets	2	Ce type DOIT être mis en œuvre. Format défini au paragraphe 7.1.1.1.2.
IPv4	3	Format défini au paragraphe 7.1.1.1.3.
IPv6	4	Format défini au paragraphe 7.1.1.1.4.
Réservé à l'IANA	5 – 192	
Usage privée	193 – 255	

Longueur d'identifiant de groupe (1 octet) - Longueur du champ Valeur d'identifiant de groupe en octets. Cette valeur NE DOIT PAS être zéro (0). Ce champ est traité comme une valeur non signée.

Valeur d'identifiant de groupe (longueur variable) - indique le nom/titre du groupe. Tous les types de GroupID devraient fournir un nom unique entre les groupes. Les types de GroupID DEVRAIENT fournir cette capacité en incluant un élément aléatoire généré par le créateur (propriétaire) du groupe d'au moins huit (8) octets, donnant une probabilité extrêmement faible de collision des noms de groupes. La valeur de GroupID est statique pour toute la vie du groupe.

Prochaine charge utile (1 octet) - indique le type de la prochaine charge utile du message. Le format de chaque charge utile est défini dans les paragraphes qui suivent. Le Tableau 12 présente les types de charge utile. Ce champ est traité comme une valeur non signée.

Tableau 12 : Types de charge utile

Type de prochaine charge utile	Valeur
Aucune	0
Jeton de politique	1
Paquet Téléchargement de clé	2
Événement de changement de clé	3
Identification	4
Réservé	5
Certificat	6
Réservé	7
Signature	8
Notification	9
Identifiant de fabricant	10
Création de clé	11
Nom occasionnel	12
Réservé à l'IANA	13 - 192
Usage privé	193 - 255

Version (1 octet) - indique la version de protocole GSAKMP utilisée. La valeur actuelle est un (1). Ce champ est traité comme une valeur non signée.

Type d'échange (1 octet) - indique le type de l'échange (aussi appelé type de message). Le Tableau 13 présente les valeurs de type d'échange. Ce champ est traité comme une valeur non signée.

Tableau 13 : Types d'échange

Type d'échange	Valeur
Réservé	0 - 3
Accusé de réception/échec de téléchargement de clé	4
Événement de changement de clé	5
Réservé	6 - 7
Demande d'adhésion	8
Téléchargement de clé	9
Téléchargement de mouchard	10
Erreur de demande d'adhésion	11
Pas d'accusé de réception	12
Demande de départ	13

Réponse de départ	14
Accusé de réception de départ	15
Réservé à l'IANA	16 - 192
Usage privé	193 - 255

Identifiant de séquence (4 octets) - l'identifiant de séquence est utilisé pour la protection des messages de gestion de groupe contre la répétition. Si le message n'est pas un message de gestion de groupe, cette valeur DOIT être réglée à zéro (0). La première valeur utilisée par un (S-)GC/KS DOIT être un (1). Pour chaque message de gestion de groupe distinct que ce (S-)GC/KS transmet, cette valeur DOIT être incrémentée de un (1). Les receveurs de ce message de gestion de groupe DOIVENT confirmer que la valeur reçue est supérieure à la valeur de l'identifiant de séquence reçu avec le dernier message de gestion de groupe provenant de ce (S-)GC/KS. Les composants de groupe (par exemple, GM, S-GC/KS) DOIVENT terminer le traitement à réception d'un message de gestion de groupe authentifié contenant un identifiant de séquence de 0xFFFFFFFF. Ce champ est traité comme un entier non signé dans l'ordre des octets du réseau.

Longueur (4 octets) – longueur totale du message (en-tête + charges utiles) en octets. Ce champ est traité comme un entier non signé dans l'ordre des octets du réseau.

7.1.1.1 Structure d'identifiant de groupe

Ce paragraphe définit les formats des types d'identifiants de groupe définis.

7.1.1.1.1 UTF-8

Le format du type UTF-8 [RFC3629] est montré à la Figure 4.

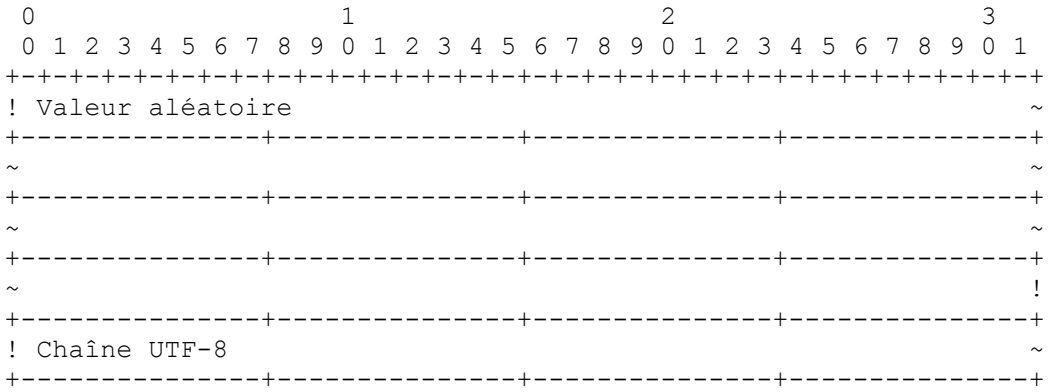


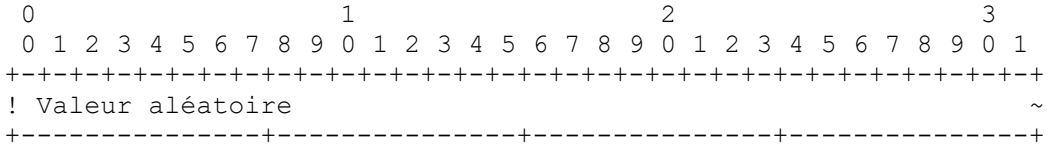
Figure 4 : Format d'identifiant de groupe UTF-8

Valeur aléatoire (16 octets) - pour le type d'identifiant de groupe UTF-8, la valeur aléatoire est représentée par une chaîne de exactement 16 chiffres hexadécimaux convertis de leurs valeurs d'octet dans l'ordre des octets du réseau. Les chiffres zéro hexadécimaux de tête et les chiffres zéro hexadécimaux de queue sont toujours inclus dans la chaîne, plutôt que d'être tronqués.

Chaîne UTF-8 (longueur variable) - ce champ contient la portion lisible par l'homme de l'identifiant de groupe en format UTF-8. Sa longueur est calculée comme la (longueur d'identifiant de groupe - 16) pour le champ Valeur aléatoire. La longueur minimum de ce champ est un (1) octet.

7.1.1.1.2 Chaîne d'octets

Le format du type Chaîne d'octet est montré à la Figure 5.



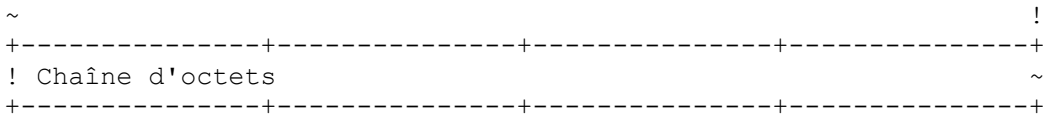


Figure 5 : Format de l'identifiant de groupe Chaîne d'octets

Valeur aléatoire (8 octets) – valeur aléatoire de 8 octets non signés dans le format des octets du réseau.

Chaîne d'octets (longueur variable) - ce champ contient la portion Chaîne d'octets de l'identifiant de groupe Sa longueur est calculée comme la (longueur d'identifiant de groupe - 8) pour le champ Valeur aléatoire. La longueur minimum de ce champ est un (1) octet.

7.1.1.1.3 Identifiant de groupe IPv4

Le format du type Identifiant de groupe IPv4 est montré à la Figure 6.

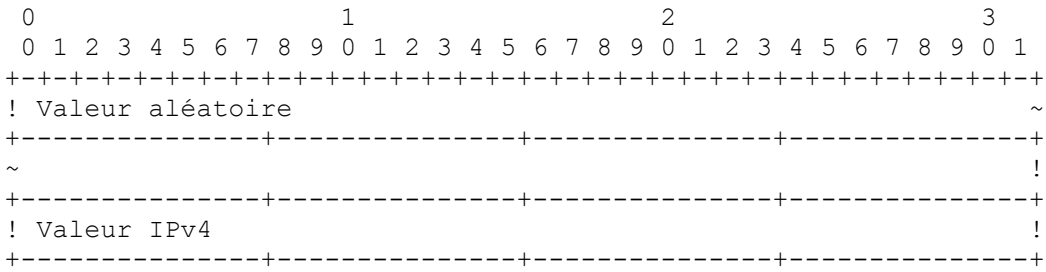


Figure 6 : Format de l'identifiant de groupe IPv4

Valeur aléatoire (8 octets) – valeur aléatoire de 8 octets non signés dans le format des octets du réseau.

Valeur IPv4 (4 octets) - valeur IPv4 dans le format des octets du réseau. Cette valeur PEUT contenir l'adresse de diffusion groupée du groupe.

7.1.1.1.4 Identifiant de groupe IPv6

Le format du type d'identifiant de groupe IPv6 est montré à la Figure 7.

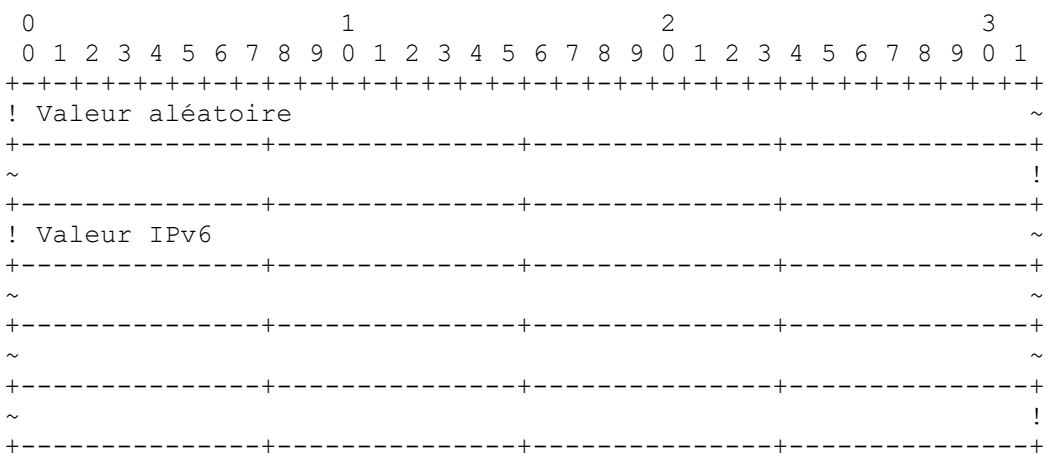


Figure 7 : Format de l'identifiant de groupe IPv6

Valeur aléatoire (8 octets) – valeur aléatoire de 8 octets non signés dans le format des octets du réseau.

Valeur IPv6 (16 octets) - valeur IPv6 dans le format des octets du réseau. Cette valeur PEUT contenir l'adresse de diffusion groupée du groupe.

7.1.2 Traitement de l'en-tête GSAKMP

Lors du traitement de l'en-tête GSAKMP, il DOIT être vérifié que les valeurs des champs suivants sont correctes :

1. Type d'identifiant de groupe – la valeur du type d'identifiant de groupe DOIT être vérifiée comme étant un type de charge utile d'identifiant de groupe valide comme défini au Tableau 11. Si la valeur n'est pas valide, une erreur est alors enregistrée. En mode verbeux, un message approprié contenant une valeur de notification Charge utile mal formée sera envoyé.

2. Identifiant de groupe - le GroupID du message reçu DOIT être vérifié par rapport aux identifiants de groupe valides du composant de groupe. Si aucune correspondance n'est trouvée, une erreur est alors enregistrée ; de plus, en mode verbeux, un message approprié contenant une valeur de notification de Identifiant de groupe invalide sera envoyé.

3. Prochaine charge utile – il DOIT être vérifié que la valeur de prochaine charge utile est un type de charge utile valide comme défini au Tableau 12. Si la valeur n'est pas valide, une erreur est alors enregistrée. En mode verbeux, un message approprié contenant une valeur de notification Type de charge utile invalide sera envoyée.

4. Version – il DOIT être vérifié que la valeur du numéro de version GSAKMP est un (1). Pour les autres valeurs, voir le traitement ci-dessous. Il DOIT être vérifié que le numéro de version GSAKMP est cohérent avec la politique du groupe comme spécifié dans son jeton de politique. Si la version n'est pas prise en charge ou autorisée, une erreur est alors enregistrée. En mode verbeux, un message approprié contenant une valeur de notification Version invalide sera envoyé.

5 Type d'échange - il DOIT être vérifié que le type d'échange est un type d'échange valide comme défini au Tableau 13 et il DOIT être du type que l'automate à états GSAKMP s'attend à recevoir. Si le type d'échange n'est pas valide, une erreur est alors enregistrée. En mode verbeux, un message approprié contenant la valeur de notification Type d'échange invalide va être envoyé.

6. Identifiant de séquence - il DOIT être vérifié que la valeur de l'identifiant de séquence est correcte. Pour les messages de négociation, cette valeur DOIT être zéro (0). Pour les messages de gestion de groupe, cette valeur DOIT être supérieure au dernier identifiant de séquence reçu de ce (S-)GC/KS. La réception d'un identifiant de séquence incorrect sur les messages de gestion de groupe NE DOIT PAS causer la génération d'un message de réponse. À réception d'un identifiant de séquence incorrect sur des messages non de gestion de groupe, une erreur est enregistrée. En mode verbeux, un message approprié contenant une valeur de notification de Identifiant de séquence invalide va être envoyé.

Les champs Longueur dans l'en-tête GSAKMP (Longueur d'identifiant de groupe et Longueur) sont utilisés pour aider au traitement du message. Si il se trouve qu'un champ est incorrect, une erreur est alors enregistrée. En mode verbeux, un message approprié contenant la valeur de notification Charge utile mal formée va être envoyé.

Afin de permettre à une mise en œuvre de GSAKMP version un (v1) d'interopérer avec de futures versions du protocole, quelques idées sont discutées ici à cet effet.

Un (S-)GC/KS qui fonctionne dans un groupe multi versions comme défini par le jeton de politique peut avoir de nombreuses approches de la façon d'interagir avec les GM dans ce groupe pour un message de changement de clé.

Une solution possible est que le (S-)GC/KS envoie plusieurs messages de changement de clé, un par niveau de version qu'il prend en charge. Puis chaque GM ne va traiter que le message de la version dans laquelle il fonctionne.

Une autre approche que toutes les mises en œuvre de GM v1 DOIVENT prendre en charge est l'incorporation d'un message de v1 dans un message de version deux (v2). Si un GM fonctionnant en v1 reçoit un message GSAKMP qui a une valeur de version supérieure à un (1), le GM va tenter de traiter les informations immédiatement après l'en-tête de groupe comme un en-tête de groupe pour la v1 du protocole. Si c'est en fait un en-tête de groupe v1, alors le reste de ce message v1 va être traité à la place. Après le traitement de ce message incoropré en v1, les données qui suivent le message v1 devraient être la charge utile telle qu'identifiée par le champ Prochaine charge utile dans l'en-tête original du message et vont être ignorées par le membre v1. Cependant, si la charge utile qui suit l'en-tête initial n'est pas un en-tête de groupe v1, alors le GM va traiter en douceur le message non reconnu.

7.2 En-tête générique de charge utile

7.2.1 Structure d'en-tête générique de charge utile

Chaque charge utile GSAKMP définie dans les paragraphes qui suivent commence par un en-tête générique, montré dans la Figure 8, qui fournit une capacité de "chaînage" de charge utile et définit clairement les limites d'une charge utile. Les champs de l'en-tête générique de charge utile sont définis comme suit :

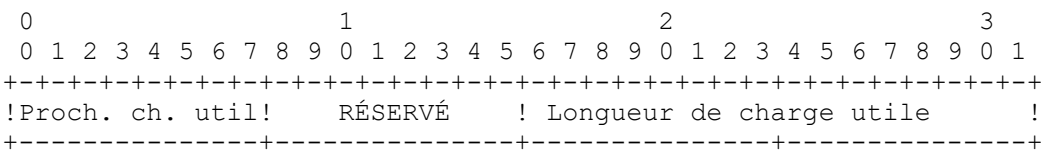


Figure 8 : en-tête générique de charge utile

Prochaine charge utile (1 octet) – identifiant du type de charge utile de la prochaine charge utile dans le message. Si la charge utile actuelle est la dernière du message, ce champ sera alors 0. Ce champ fournit la capacité de "chaînage". Le Tableau 12 identifie les types de charge utile. Ce champ est traité comme une valeur non signée.

RÉSERVÉ (1 octet) - non utilisé, réglé à 0.

Longueur de charge utile (2 octets) – longueur en octets de la charge utile courante, incluant l'en-tête de charge utile générique. Ce champ est traité comme un entier non signé dans le format des octets du réseau.

7.2.2 Traitement de l'en-tête générique de charge utile

Lors du traitement de l'en-tête générique de charge utile, il DOIT être vérifié que les valeurs des champs suivants sont correctes :

1. Prochaine charge utile – la valeur de Prochaine charge utile DOIT être vérifiée comme étant un type de charge utile valide comme défini au Tableau 12. Si le type de charge utile n'est pas valide, une erreur est alors enregistrée. En mode verbeux, un message approprié contenant une valeur de notification Type de charge utile invalide sera envoyé.

2. RÉSERVÉ - Ce champ DOIT contenir la valeur zéro (0). Si la valeur de ce champ n'est pas zéro (0), une erreur est alors enregistrée. En mode verbeux, un message approprié contenant une valeur de notification Charge utile mal formée sera envoyé.

Le champ Longueur dans l'en-tête générique de charge utile est utilisé pour traiter le reste de la charge utile. Si ce champ se trouve être incorrect, une erreur est alors enregistrée. En mode verbeux, un message approprié contenant une valeur de notification de Charge utile mal formée sera envoyé.

7.3 Charge utile Jeton de politique

7.3.1 Structure de charge utile Jeton de politque

La charge utile Jeton de politique contient des informations authentifiables spécifiques du groupe qui décrivent les comportement pertinents du groupe pour la sécurité, les paramètres de contrôle d'accès, et les mécanismes de sécurité. La Figure 9 montre le format de la charge utile.

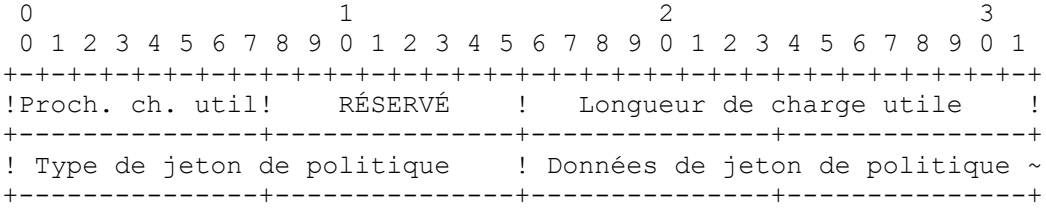


Figure 9 : Format de charge utile Jeton de politique

Les champs de la charge utile Jeton de politique sont définis comme suit :

Prochaine charge utile (1 octet) – Identifiant du type de charge utile de la prochaine charge utile dans le message. Si la charge utile en cours est la dernière du message, ce champ sera alors à 0. Ce champ fournit la capacité de "chaînage". Le Tableau 12 identifie les types de charge utile. Ce champ est traité comme une valeur non signée.

RÉSERVÉ (1 octet) - non utilisé, réglé à 0.

Longueur de charge utile (2 octets) – longueur en octets de la charge utile en cours, incluant l'en-tête de charge utile générique. Ce champ est traité comme un entier non signé dans le format des octets du réseau.

Type de jeton de politique (2 octets) – spécifie le type de jeton de politique utilisé. Le Tableau 14 identifie les types de jetons de politique. Ce champ est traité comme un entier non signé dans le format des octets du réseau.

Tableau 14 : Types de jeton de politique

Type de jeton de politique	Valeur	Définition
Réservé	0	
GSAKMP_ASN.1_PT_V1	1	Toutes les mises en œuvre de GSAKMP DOIVENT prendre en charge ce format de PT. Format spécifié dans la [RFC4534].
Réservé à l'IANA	2 - 49152	
Usage privé	49153 - 65535	

Données de jeton de politique (longueur variable) – contient les informations de jeton de politique. Les valeurs de ce champ sont spécifiques du jeton, et le format est spécifié par le champ Type de PT.

Si cette charge utile est chiffrée, seul le champ Données de jeton de politique est chiffré.

Le type de charge utile pour la charge utile Jeton de politique est un (1).

7.3.2 Traitement de la charge utile Jeton de politique

Lors du traitement de la charge utile Jeton de politique, il DOIT être vérifié que les champs suivants ont les valeurs correctes :

1. Prochaine charge utile, RÉSERVÉ, Longueur de charge utile - ces champs sont traités comme défini au paragraphe 7.2.2, "Traitement d'en-tête générique de charge utile".

2. Type de jeton de politique – la valeur du type de jeton de politique DOIT être vérifiée comme étant un type valide de jeton de politique comme défini au Tableau 14. Si la valeur n'est pas valide, une erreur est alors enregistrée. En mode verbeux, un message approprié contenant une valeur de notification de Charge utile mal formée sera envoyé.

3. Données de jeton de politique - ces données de jeton de politique DOIVENT être traitées conformément au type de jeton de politique spécifié. Le type va définir le format des données.

7.4 Charge utile Téléchargement de clé

Se reporter à la section de terminologie pour les différents termes relatifs aux clés utilisés dans ce paragraphe.

7.4.1 Structure de charge utile Téléchargement de clé

La charge utile Téléchargement de clé contient les clés de groupe (par exemple, clés de groupe, clés initiales de changement de clé, etc.). Ces charges utiles Téléchargement de clé peuvent avoir plusieurs attributs de sécurité qui leur sont appliqués sur la base de la politique de sécurité du groupe. La Figure 10 montre le format de la charge utile.

La politique de sécurité du groupe dicte que la charge utile Téléchargement de clé DOIT être chiffrée avec une clé de chiffrement de clé (KEK). Le mécanisme de chiffrement utilisé est spécifié dans le jeton de politique. Les membres du groupe DOIVENT créer la KEK en utilisant la méthode de création de clé identifiée dans la charge utile Création de clé.

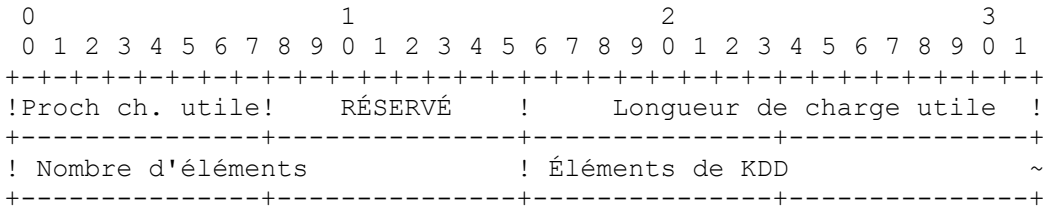


Figure 10 : Format de charge utile Téléchargement de clé

Les champs de charge utile Téléchargement de clé sont définis comme suit :

Prochaine charge utile (1 octet) – Identifiant pour le type de charge utile de la prochaine charge utile dans le message. Si la charge utile en cours est la dernière du message, ce champ sera alors à 0. Ce champ fournit la capacité de "chaînage". Le Tableau 12 identifie les types de charge utile. Ce champ est traité comme une valeur non signée.

RÉSERVÉ (1 octet) - non utilisé, réglé à 0.

Longueur de charge utile (2 octets) – longueur en octets de la charge utile en cours, incluant l'en-tête de charge utile générique. Ce champ est traité comme un entier non signé dans le format des octets du réseau.

Nombre d'éléments (2 octets) – contient le nombre total de clés de protection du trafic de groupe et de matrices de changement de clés passé dans ce bloc de données. Ce champ est traité comme un entier non signé dans le format des octets du réseau.

Éléments de KDD (données de téléchargement de clé) (longueur variable) – contient les informations de téléchargement de clé. Les données de téléchargement de clé sont une séquence de Type/Longueur/Données du nombre d'éléments. Le format de chaque élément est défini dans la Figure 11.

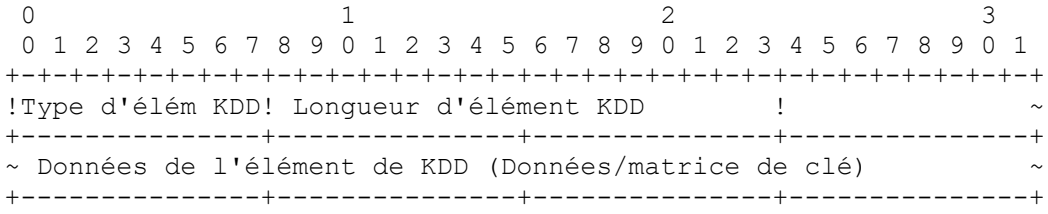


Figure 11 : Format d'élément de données de téléchargement de clé

Pour chaque élément de données de téléchargement de clé, le format des données est comme suit :

Type d'élément de données de téléchargement de clé (KDD, *Key Download Data*) (1 octet) – Identifiant du type de données contenant cet élément KDD. Voir au Tableau 15 les valeurs possibles de ce champ. Ce champ est traité comme une valeur non signée.

Longueur d'élément de KDD (2 octets) - longueur en octets des données pour l'élément de données de téléchargement de clé qui suit ce champ. Ce champ est traité comme un entier non signé dans le format des octets du réseau.

Données pour l'élément de KDD (longueur variable) – contient les clés et les informations qui s'y rapportent. Le format de ce champ est spécifique de la valeur du champ Type d'élément de KDD. Pour le type d'élément de KDD GTPK, ce champ va contenir un élément Clé comme défini au paragraphe 7.4.1.1. Pour le type d'élément de KDD Rekey - LKH, ce champ va contenir une matrice de changement de clé comme défini au paragraphe 7.4.1.2.

Tableau 15 : Types d'éléments de données de téléchargement de clé

Type d'élément de KDD	Valeur	Définition
GTPK	0	Ce type DOIT être mis en œuvre. Il identifie que les données contiennent des informations de clé de protection du trafic du groupe.
Rekey – LKH	1	Facultatif
Réservé à l'IANA	2 – 192	

Usage privé	193 – 255

Le chiffrement de cette charge utile ne couvre que les données qui suivent l'en-tête générique de charge utile (champs Nombre d'éléments et Éléments de KDD).

Le type de charge utile pour le paquet Téléchargement de clé est deux (2).

7.4.1.1 Structure Élément de clé

Un élément de clé *(Key Datum)* contient toutes les informations pour une clé. La Figure 12 montre le format de cette structure.

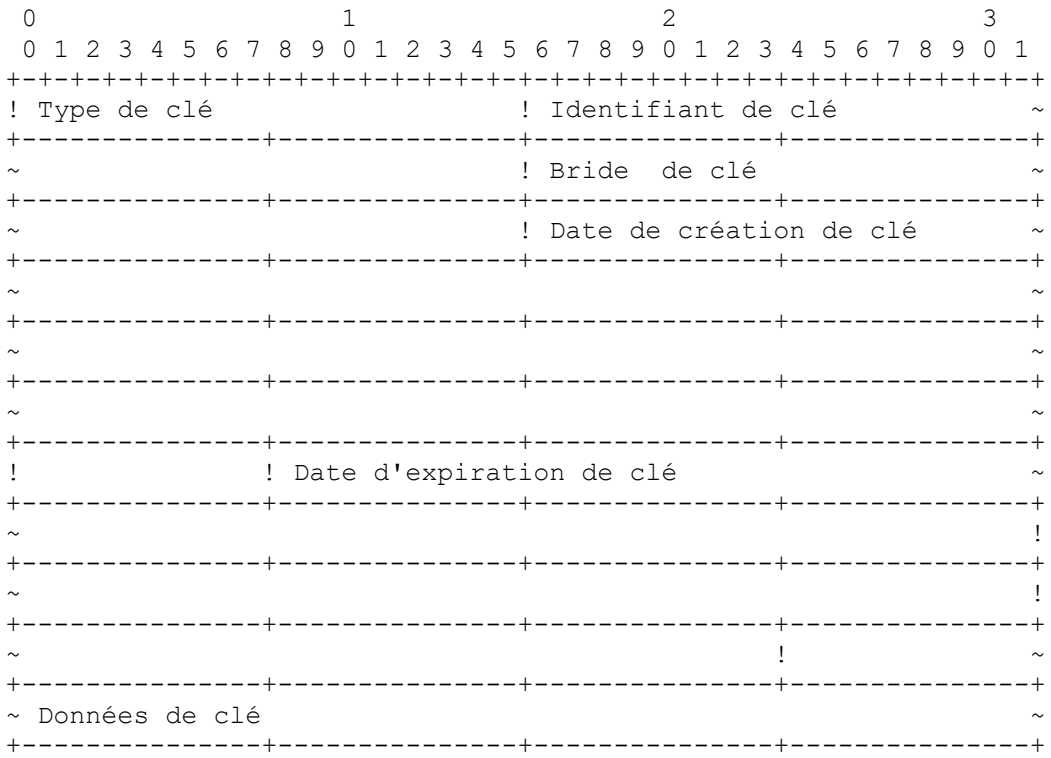


Figure 12 : Format d'élément de clé

Type de clé (2 octets) - c'est l'algorithme cryptographique pour lequel ces données de clé vont être utilisées. Cette valeur est spécifiée dans le jeton de politique. Voir au Tableau 16 les valeurs possibles de ce champ. Ce champ est traité comme une valeur non signée.

Tableau 16 : Types de clés de chiffrement

Types de clés de chiffrement	Valeur	Description
Réservé	0 – 2	
3DES_CBC64_192	3	Voir la [RFC2451].
Réservé	4 – 11	
AES_CBC_128	12	Ce type DOIT être pris en charge. Voir la [RFC4306].
AES_CTR	13	Voir la [RFC4306].
Réservé à l'IANA	14 - 49152	
Usage privé	49153 - 65535	

Identifiant de clé (4 octets) – c'est l'identifiant permanent de toutes les versions de la clé. Cette valeur PEUT être définie par le jeton de politique. Ce champ est traité comme une chaîne d'octets.

Bride de clé (4 octets) - c'est la valeur qui identifie de façon univoque une version (instance particulière) d'une clé. Ce champ est traité comme une chaîne d'octets.

Date de création de clé (15 octets) – c'est la valeur de temps à laquelle ces données de clé ont été générées. Ce champ contient l'horodatage en format UTF-8 AAAAMMJJHHMMSSZ, où AAAA sont l'année (0000 - 9999), MM sont la valeur numérique du mois (01 - 12), JJ sont le jour du mois (01 - 31), HH est l'heure du jour (00 - 23), MM est la minute dans l'heure (00 - 59), SS sont les secondes dans la minute (00 - 59), et la lettre Z indique que c'est l'heure Zoulou *(UTC, Temps universel coordonné)*. Ce format est vaguement fondé sur la [RFC3161].

Date d'expiration de clé (15 octets) – c'est la valeur du moment où l'utilisation de cette clé ne sera plus valide. Ce champ contient l'horodatage en format UTF-8 AAAAMMJJHHMMSSZ, où AAAA sont l'année (0000 - 9999), MM sont la valeur numérique du mois (01 - 12), JJ sont le jour du mois (01 - 31), HH sont l'heure du jour (00 - 23), MM est la minute dans l'heure (00 - 59), SS sont les secondes dans la minute (00 - 59), et la lettre Z indique que c'est l'heure Zoulou. Ce format est vaguement fondé sur la [RFC3161].

Données de clé (longueur variable) – ce sont les données de clé réelles, qui dépendent de l'algorithme de type de clé pour ce format.

Note : la combinaison de l'identifiant de clé et de la bride de clé DOIT être unique au sein du groupe. Cette combinaison va être utilisée pour identifier une clé de façon univoque.

7.4.1.2 Structure de matrice de changement de clé

Une matrice de changement de clé contient les informations pour l'ensemble de KEK qui est associé à un membre du groupe. La Figure 13 montre le format de cette structure.

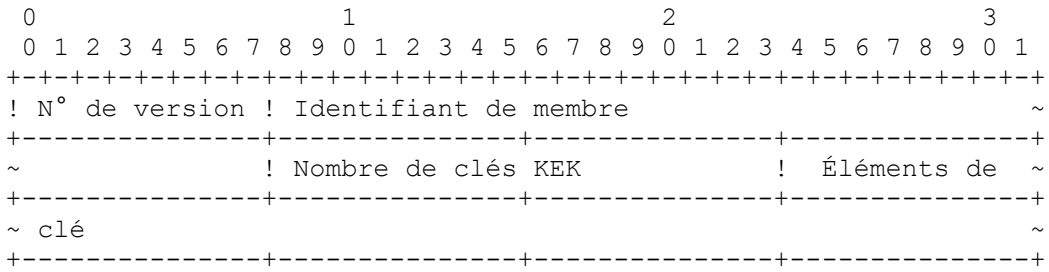


Figure 13 : Format de structure de matrice de changement de clé

Numéro de version (1 octet) – contient la version du protocole de changement de clé dans lequel les données sont formatées. Pour le type d'élément de KDD Rekey - LKH, se reporter au paragraphe A.2 pour la description de cette valeur. Ce champ est traité comme une valeur non signée.

Identifiant de membre (4 octets) - c'est l'identifiant de membre de la séquence de changement de clé contenue dans cette matrice de changement de clé. Ce champ est traité comme une chaîne d'octets. Pour le type d'élément de KDD Rekey - LKH, se reporter au paragraphe A.2 pour la description de cette valeur..

Nombre de clés KEK (2 octets) - cette valeur est le nombre de clés de KEK distinctes dans cette séquence. Cette valeur est traitée comme un entier non signé dans le format des octets du réseau.

Éléments de clé (longueur variable) - séquence des KEK dans le format d'élément de clé. Le format de chaque élément de clé dans cette séquence est défini au paragraphe 7.4.1.1.

Identifiant de clé (pour un identifiant de clé au sein du changement de clé) – espace de LKH, se reporter au paragraphe A.2 pour la description de cette valeur.

7.4.2 Traitement de la charge utile Téléchargement de clé

Avant de traiter ces données, le contenu de la charge utile DOIT être déchiffré.

Lors du traitement de la charge utile Téléchargement de clé, la valeur correcte des champs suivants DOIT être vérifiée :

1. Prochaine charge utile, RÉSERVÉ, Longueur de charge utile - ces champs sont traités comme défini au

paragraphe 7.2.2, "Traitement de l'en-tête générique de charge utile".

2. Type d'élément de KDD – tous les champs de type d'élément de KDD DOIVENT être vérifiés comme étant d'un type d'élément de KDD valide comme défini au Tableau 15. Si la valeur n'est pas valide, une erreur est alors enregistrée. En mode verbeux, un message approprié contenant une valeur de notification de Charge utile mal formée sera envoyé.

3. Type de clé – tous les champs de type de clé DOIVENT être vérifiés comme étant d'un type de chiffrement valide comme défini au Tableau 16. Si la valeur n'est pas valide, une erreur est alors enregistrée. En mode verbeux, un message approprié contenant une valeur de notification de Informations de clé invalides va être envoyé.

4. Date d'expiration de clé – tous les champs de la date d'expiration de clé DOIVENT être vérifiés pour confirmer que leur valeur représente une valeur future et non passée. Si la valeur n'est pas valide, une erreur est alors enregistrée. En mode verbeux, un message approprié contenant une valeur de notification de Informations de clé invalides va être envoyé.

Les champs de longueur et de compteur dans la charge utile sont utilisés pour aider au traitement de la charge utile. Si un des champs se trouve être incorrect, une erreur est alors enregistrée. En mode verbeux, un message approprié contenant une valeur de notification de Charge utile mal formée sera envoyé.

7.5 Charge utile Événement de changement de clé

Se reporter à la section de terminologie pour les différents termes relatifs aux clés utilisées dans ce paragraphe.

7.5.1 Structure de charge utile Événement de changement de clé

La charge utile Événement de changement de clé PEUT contenir plusieurs clés chiffrées dans des KEK enveloppantes. La Figure 14 montre le format de la charge utile. Si les données qui doivent être contenues dans une charge utile Événement de changement de clé sont trop grosses pour la charge utile, la séquence peut être partagée entre plusieurs charges utiles Événement de changement de clé à une limite de données d'événement de changement de clé.

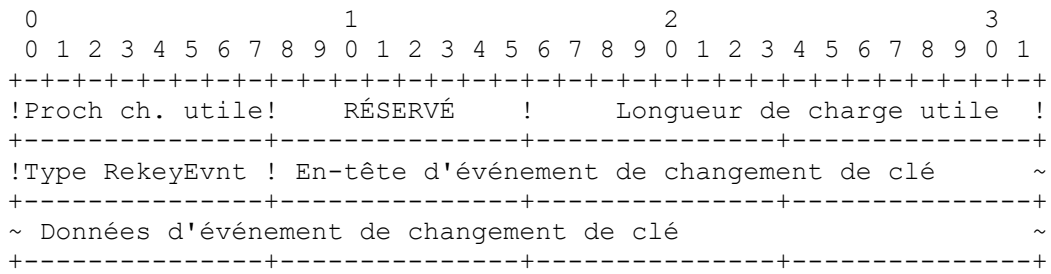


Figure 14 : Format de Charge utile Événement de changement de clé

Les champs de charge utile Événement de changement de clé sont définis comme suit :

Prochaine charge utile (1 octet) – Identifiant du type de charge utile de la prochaine charge utile dans le message. Si la charge utile en cours est la dernière du message, ce champ sera alors à 0. Ce champ fournit la capacité de "chaînage". Le Tableau 12 identifie les types de charge utile. Ce champ est traité comme une valeur non signée.

RÉSERVÉ (1 octet) - non utilisé, réglé à 0.

Longueur de charge utile (2 octets) – longueur en octets de la charge utile en cours, incluant l'en-tête de charge utile générique. Ce champ est traité comme un entier non signé dans le format des octets du réseau.

Type d'événement de changement de clé (1 octet) – spécifie le type d'événement de changement de clé utilisé. Le Tableau 17 présente les types d'événements de changement de clé. Ce champ est traité comme une valeur non signée.

En-tête d'événement de changement de clé (longueur variable) – ce sont les informations d'en-tête pour l'événement de changement de clé. Le format est défini au paragraphe 7.5.1.1, "Structure d'en-tête d'événement de changement de clé".

Données d'événement de changement de clé (longueur variable) – ce sont les informations de changement de clé pour l'événement de changement de clé. Le format est défini au paragraphe 7.5.1.2, "Structure de données d'événement de

changement de clé".

Le type de charge utile Événement de changement de clé est trois (3).

Tableau 17 : Types d'événement de changement de clé

Type d'événement de changement de clé	Valeur	Définition
Aucun	0	Ce type DOIT être mis en œuvre. Dans ce cas, la taille du champ Données d'événement de changement de clé sera long de zéro octets. L'objet d'une charge utile Événement de changement de clé de type Aucun est quand il est nécessaire d'envoyer un nouveau jeton sans informations de changement de clé. Le message Changement de clé GSAKMP exige une charge utile Événement de changement de clé, et dans cette instance il aurait des données de changement de clé de type Aucun.
GSAKMP_LKH	1	Les données de changement de clé vont avoir le type LKH formaté selon GSAKMP. Le format de ce champ est défini au paragraphe 7.5.1.2.
Réservé à l'IANA	2 – 192	
Usage privé	193 – 255	

7.5.1.1 Structure d'en-tête d'événement de changement de clé

Le format de l'en-tête d'événement de changement de clé est montré à la Figure 15.

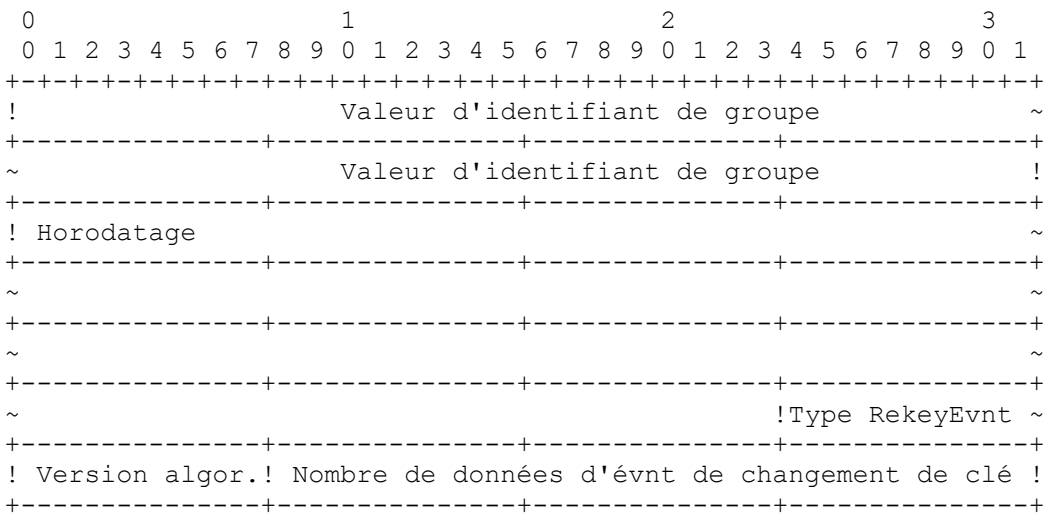


Figure 15 : Format d'en-tête d'événement de changement de clé

Valeur d'identifiant de groupe (longueur variable) - indique le nom/titre du groupe dont les clés sont à changer. Ce sont les mêmes format, longueur, et valeur que la valeur d'identifiant de groupe du paragraphe 7.1, "En-tête GSAKMP".

Horodatage (15 octets) - c'est la valeur de l'heure à laquelle les données d'événement de changement de clé ont été générées. Ce champ contient l'horodatage en format UTF-8 AAAAMMJJHHMMSSZ, où AAAA sont l'année (0000 - 9999), MM est la valeur numérique du mois (01 - 12), JJ est le jour du mois (01 - 31), HH est l'heure du jour (00 - 23), MM est la minute dans l'heure (00 - 59), SS sont les secondes dans la minute (00 - 59), et la lettre Z indique que c'est l'heure Zoulou. Ce format est vaguement fondé sur la [RFC3161].

Type d'événement de changement de clé (1 octet) – c'est l'algorithme de changement de clé utilisé pour ce groupe. Les valeurs de ce champ se trouvent dans le Tableau 17. Ce champ est traité comme une valeur non signée.

Version d'algorithme (1 octet) - indique la version du type de changement de clé utilisé. Pour le type d'événement de changement de clé de GSAKMP_LKH, se référer au paragraphe A.2 pour la description de cette valeur. Ce champ est traité comme une valeur non signée.

Nombre de données d'événement de changement de clé (2 octets) - nombre de données d'événement de changement de clé

contenues dans les données de changement de clé. Cette valeur est traitée comme un entier non signé dans l'ordre des octets du réseau.

7.5.1.2 Structure des données d'événement de changement de clé

Comme défini dans l'en-tête d'événement de changement de clé, champ Nombre de données de changement de clé, plusieurs éléments d'information sont envoyés dans les données d'événement de changement de clé. Chaque utilisateur final ne sera intéressé que par une donnée d'événement de changement de clé parmi toutes les informations envoyées. Chaque donnée d'événement de changement de clé va contenir tous les paquetages de clé qu'un utilisateur exige. Pour chaque donnée d'événement de changement de clé, les données qui suivent les champs enveloppants sont chiffrées avec la clé identifiée dans l'en-tête d'enveloppe. La Figure 16 montre le format de chaque donnée d'événement de changement de clé.

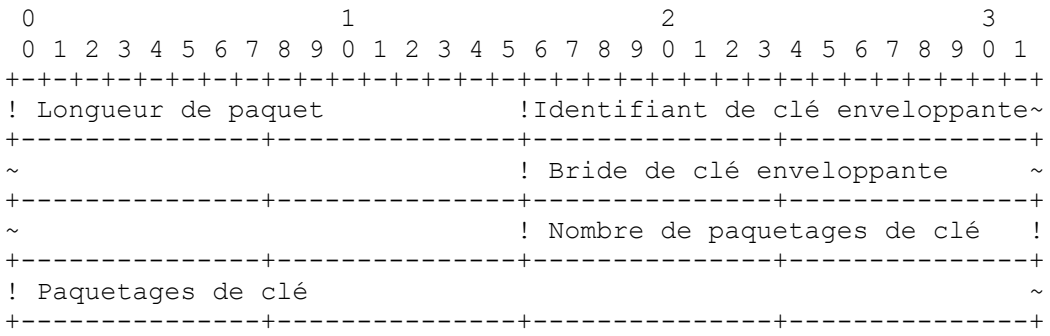


Figure 16 : Format de données d'événement de changement de clé

Longueur de paquet (2 octets) – longueur en octets des données d'événement de changement de clé, qui consistent en le nombre de paquetages de clés et le ou les paquetages de clés. Cette valeur est traitée comme un entier non signé dans l'ordre des octets du réseau.

Identifiant de clé enveloppante (4 octets) - c'est l'identifiant de clé de la KEK utilisée pour le chiffrement/déchiffrement des nouvelles clé (changées). Pour le type d'événement de changement de clé de Rekey - LKH, se reporter au paragraphe A.2 pour la description de cette valeur.

Bride de clé enveloppante (4 octets) - c'est une bride de clé de la KEK utilisée pour le chiffrement/déchiffrement des nouvelles clés (changées). Voir au paragraphe 7.4.1.1 les valeurs de ce champ.

Nombre de paquetages de clés (2 octets) - nombre de paquetages de clé contenus dans ces données d'événement de changement de clé. Cette valeur est traitée comme un entier non signé dans l'ordre des octets du réseau.

Paquetages de clé (longueur variable) – format de type/longueur/valeur d'un élément de clé. Ce format est défini au paragraphe 7.5.1.2.1.

7.5.1.2.1 Structure de paquetage de clé

Chaque paquetage de clé contient toutes les informations sur la clé. La Figure 17 montre le format d'un paquetage de clé.

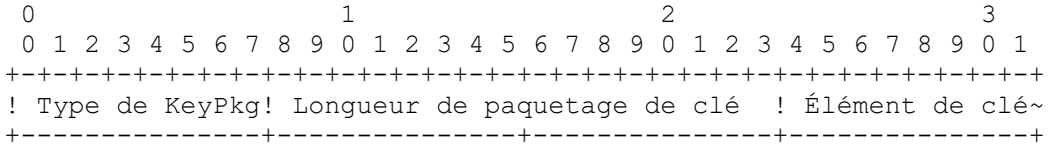


Figure 17 : Format de paquetage de clé

Type de paquetage de clé (1 octet) - type de clé dans ce paquetage de clé. Les valeurs légales pour ce champ sont définies au Tableau 15, Types de données de téléchargement de clé. Ce champ est traité comme une valeur non signée.

Longueur de paquetage de clé (2 octets) – longueur de l'élément de clé. Ce champ est traité comme un entier non signé dans le format des octets du réseau.

Élément de clé (longueur variable) – données réelles de la clé. Le format de ce champ est défini au paragraphe 7.4.1.1, "Structure d'élément de clé".

7.5.2 Traitement de la charge utile Événement de changement de clé

Lors du traitement de la charge utile Événement de changement de clé, il DOIT être vérifié que les champs suivants ont des valeurs correctes :

1. Prochaine charge utile, RÉSERVÉ, Longueur de charge utile - ces champs sont traités comme défini au paragraphe 7.2.2, "Traitement d'en-tête générique de charge utile".

2. Champ Type d'événement de changement de clé au sein de l'en-tête de charge utile "Événement de changement de clé" - le type d'événement de changement de clé DOIT être vérifié comme type valide d'événement de changement de clé comme défini au Tableau 17. Si le type d'événement de changement de clé n'est pas valide, une erreur est enregistrée sans considération du mode (concis ou verbeux). Aucun message de réponse d'erreur n'est généré pour la réception d'un message de gestion de groupe.

3. Valeur d'identifiant de groupe – la valeur de l'identifiant de groupe de l'en-tête Événement de changement de clé du message reçu DOIT être confrontée à l'identifiant de groupe du composant de groupe. Si aucune correspondance n'est trouvée, la charge utile est éliminée, puis une erreur est enregistrée, sans considération du mode (concis ou verbeux). Aucun message de réponse d'erreur n'est généré pour la réception d'un message de gestion de groupe.

4. Horodatage – la valeur de l'horodatage de l'en-tête Événement de changement de clé PEUT être vérifiée pour déterminer si l'heure de génération de l'événement de changement de clé est récente par rapport au délai du réseau et aux temps de traitement. Si l'horodatage est jugé n'être pas récent, une erreur est enregistrée. Aucun message de réponse d'erreur n'est généré pour la réception d'un message de gestion de groupe.

5. Champ Type d'événement de changement de clé au sein de l'en-tête Événement de changement de clé - le type d'événement de changement de clé de l'en-tête Événement de changement de clé du message reçu DOIT être vérifié comme type d'événement de changement de clé valide, comme défini au Tableau 17, et la même valeur de type d'événement de changement de clé antérieurement dans cette charge utile. Si le type d'événement de changement de clé n'est pas valide ou non égal à la valeur précédente de type d'événement de changement de clé, une erreur est alors enregistrée, sans considération du mode (concis ou verbeux). Aucun message de réponse d'erreur n'est généré pour la réception d'un message de gestion de groupe.

6. Version d'algorithme – le numéro de version de l'algorithme de changement de clé DOIT être vérifié pour s'assurer que la version indiquée est prise en charge. Si il n'est pas pris en charge, une erreur est alors enregistrée, sans considération du mode (concis ou verbeux). Aucun message de réponse d'erreur n'est généré pour la réception d'un message de gestion de groupe.

Les champs Longueur et Compteur sont utilisés pour aider au traitement du message. Si un champ se trouve être incorrect, l'arrêt du traitement DOIT être initié.

Un GM DOIT traiter toutes les données d'événement de changement de clé comme fondées sur la méthode de changement de clé utilisée. Il y a un potentiel que plusieurs événements de changement de clé soient pour ce GM. Les données d'événement de changement de clé sont traitées dans l'ordre jusqu'à ce que toutes les données d'événement de changement de clé soient consommées.

1. Identifiant de clé enveloppante - il DOIT être confronté à la liste des KEK mémorisées que détient ce GM. Si une correspondance est trouvée, le traitement de ces données d'événement de changement de clé se poursuit. Autrement, sauter les prochaines données d'événement de changement de clé.

2. Bride de clé enveloppante - Si un identifiant de clé enveloppante correspondant a été trouvé, la bride de clé enveloppante DOIT alors être confrontée à la bride de la KEK pour laquelle l'identifiant de clé correspondait. Si les brides correspondent, alors le GM va traiter les paquetages de clé associés à ces données d'événement de changement de clé. Autrement, sauter les prochaines données d'événement de changement de clé.

Si un GM a trouvé un identifiant de clé enveloppante et une bride de clé enveloppante qui correspondent, le GM déchiffre les données restantes dans ces données d'événement de changement de clé conformément à la politique en utilisant la KEK

définie par l'identifiant de clé enveloppante et la bride de clé enveloppante. Après le déchiffrement des données, le GM extrait le champ Nombre de paquetages de clé pour aider à traiter les paquetages de clé suivants. Les paquetages de clé sont traités comme suit :

1. Type de paquetage de clé – il DOIT être vérifié qu'il est valide comme défini au Tableau 15. Si il n'est pas valide, une erreur est enregistrée, sans considération du mode (concis ou verbeux). Aucun message de réponse d'erreur n'est généré pour la réception d'un message de gestion de groupe.

2. Longueur de paquetage de clé – il est utilisé pour traiter les informations d'élément de clé suivantes.

3. Type de clé – il DOIT être vérifié qu'il est valide comme défini au Tableau 16. Si le type de paquetage de clé n'est pas valide, une erreur est enregistrée, sans considération du mode (concis ou verbeux). Aucun message de réponse d'erreur n'est généré pour la réception d'un message de gestion de groupe.

4. Identifiant de clé - l'identifiant de clé DOIT être confronté à l'ensemble des identifiants de clés que cet utilisateur conserve pour ce type de clé. Si aucune correspondance n'est trouvée, une erreur est enregistrée, sans considération du mode (concis ou verbeux). Aucun message de réponse d'erreur n'est généré pour la réception d'un message de gestion de groupe.

5. Bride de clé - la bride de clé est extraite comme elle est et est utilisée pour être la nouvelle bride de clé pour la clé actuellement associée à l'identifiant de clé du paquetage de clé.

6. Date de création de clé - la date de création de clé DOIT être vérifiée comme étant postérieure à la date de création de clé de la clé actuellement détenue. Si cette date est antérieure à celle de la clé actuellement détenue, une erreur est enregistrée, sans considération du mode (concis ou verbeux). Aucun message de réponse d'erreur n'est généré pour la réception d'un message de gestion de groupe.

7. Date d'expiration de clé - la date d'expiration de clé DOIT être vérifiée comme étant postérieure à la date de création de clé juste reçue et que les règles de temps se conforment à la politique. Si la date d'expiration n'est pas postérieure à la date de création ou ne se conforme pas à la politique, une erreur est enregistrée, sans considération du mode (concis ou verbeux). Aucun message de réponse d'erreur n'est généré pour la réception d'un message de gestion de groupe.

8. Données de clé - les données de clé sont extraites sur la base des informations de longueur dans le paquetage de clé.

Si il n'y avait pas d'erreur dans le traitement du paquetage de clé, la clé représentée par l'identifiant de clé va avoir toutes ses données mises à jour sur la base des informations reçues.

7.6 Charge utile Identification

7.6.1 Structure de charge utile Identification

La charge utile Identification contient des données spécifiques de l'entité utilisées pour échanger les informations d'identification. Ces informations sont utilisées pour vérifier les identités des membres. La Figure 18 montre le format de la charge utile Identification.

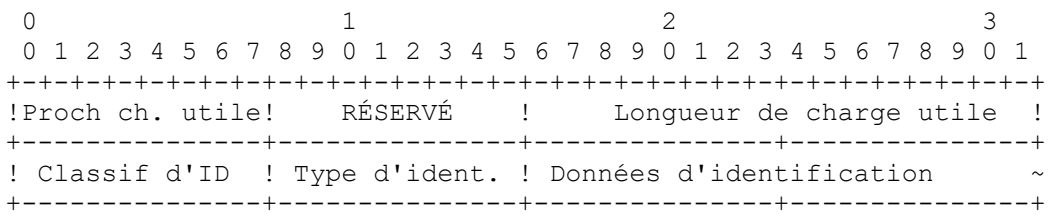


Figure 18 : Format de la charge utile Identification

Les champs de la charge utile Identification sont définis comme suit :

Prochaine charge utile (1 octet) – Identifiant du type de charge utile de la prochaine charge utile dans le message. Si la charge utile en cours est la dernière du message, ce champ sera alors à 0. Ce champ fournit la capacité de "chaînage". Le Tableau 12 identifie les types de charge utile. Ce champ est traité comme une valeur non signée.

RÉSERVÉ (1 octet) - non utilisé, réglé à 0.

Longueur de charge utile (2 octets) – longueur en octets de la charge utile en cours, incluant l'en-tête de charge utile générique. Ce champ est traité comme un entier non signé dans le format des octets du réseau.

Classification d'identification (ID) (1 octet) – Classifie le propriétaire des données d'identification. Le Tableau 18 identifie les valeurs possibles pour ce champ. Ce champ est traité comme une valeur non signée.

Tableau 18 : Classification d'identification

Classification	**Valeur**
Envoyeur	0
Receveur	1
Tiers	2
Réservé à l'IANA	3 - 192
Usage privé	193 - 255

Type d'identification (ID) (1 octet) – Spécifie le type de l'identification utilisée. Le Tableau 19 identifie les valeurs possibles pour ce type. Ce champ est traité comme une valeur non signée. Tous les types définis sont FACULTATIFS sauf mention contraire.

Données d'identification (longueur variable) – contient les informations d'identité. Les valeurs pour ce champ sont spécifiques du groupe, et le format est spécifié par le champ Type d'ID. Le format de ce champ est déclaré en conjonction avec le type dans le Tableau 19.

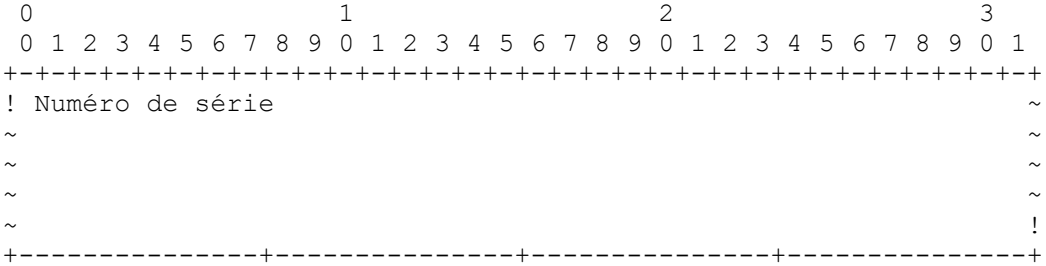
Le type de charge utile pour la charge utile Identification est quatre (4).

Tableau 19 : Types d'identification

Type d'ID	**Valeur**	**Champ Cert PKIX**	**Défini dans**
Réservé	0		
ID_IPV4_ADDR	1	SubjAltName iPAddress	Voir [RFC4306] § 3.5.
ID_FQDN	2	SubjAltName dNSName	Voir [RFC4306] § 3.5.
ID_RFC822_ADDR	3	SubjAltName rfc822Name	Voir [RFC4306] § 3.5.
Réservé	4		
ID_IPV6_ADDR	5	SubjAltName	Voir [RFC4306] § 3.5.
Réservé	6 - 8		
ID_DER_ASN1_DN	9	Sujet entier, comparé au bit près	Voir [RFC4306] § 3.5.
Réservé	10		
ID_KEY_ID	11	N/A	Voir [RFC4306] § 3.5.
Réservé	12 - 29		
Nom non codé (ID_U_NAME)	30	Sujet	Le format pour ce type est défini au paragraphe 7.6.1.1.
ID_DN_STRING	31	Sujet	Voir la [RFC4514]. Ce type DOIT être mis en œuvre.
Réservé à l'IANA	32 - 192		
Usage privé	193 - 255		

7.6.1.1 Structure de ID_U_NAME

Le format pour le type Nom non codé (ID_U_NAME) est montré à la Figure 19.



```
! Longueur                                                     !
+---------------+---------------+---------------+---------------+
! Données de DN                                                ~
+---------------+---------------+---------------+---------------+
```

Figure 19 : Format de nom non codé (ID-U-NAME)

Numéro de série (20 octets) – numéro de série du certificat. Ce champ est traité comme un entier non signé dans le format des octets du réseau.

Longueur (4 octets) – longueur en octets du champ Données de DN. Ce champ est traité comme un entier non signé dans le format des octets du réseau.

Données de DN (longueur variable) – valeur réelle du DN UTF-8 (champ Sujet) en utilisant le caractère barre oblique (/) comme délimiteur de champ (par exemple, "/C=US/ST=MD/L=Quelquepart/O=ACME, Inc./OU=DIV1/CN=usager1/Email=usager1@acme.com" sans les guillemets).

7.6.2 Traitement de la charge utile Identification

Lors du traitement de la charge utile Identification, il DOIT être vérifié que les champs suivants ont les valeurs correctes :

1. Prochaine charge utile, RÉSERVÉ, Longueur de charge utile - ces champs sont traités comme défini au paragraphe 7.2.2, "Traitement de l'en-tête générique de charge utile".

2. Classification d'ID – la valeur de la classification d'ID DOIT être vérifiée comme étant un type valide de classification d'identification comme défini au Tableau 18. Si la valeur n'est pas valide, une erreur est alors enregistrée. En mode verbeux, un message approprié contenant une valeur de notification de Charge utile mal formée sera envoyé.

3. Type d'identification – la valeur du type d'identification DOIT être vérifiée comme étant un type valide d'identification comme défini au Tableau 19. Si la valeur n'est pas valide, une erreur est alors enregistrée. En mode verbeux, un message approprié contenant une valeur de notification de Charge utile mal formée sera envoyé.

4. Données d'identification - ces données d'identification DOIVENT être traitées en accord avec le type d'identification spécifié. Le type va définir le format des données. Si les données d'identification sont utilisées pour trouver une correspondance et si aucune n'est trouvée, une erreur est alors enregistrée. En mode verbeux, un message approprié contenant une valeur de notification de Informations d'identification invalides va être envoyé.

7.6.2.1 Traitement de ID_U_NAME

Lors du traitement des données d'identification de type ID_U_NAME, il DOIT être vérifié que les champs suivants ont les valeurs correctes :

1. Numéro de série - le numéro de série DOIT être supérieur ou égal à un (1) pour être un numéro de série valide d'une CA conforme [RFC3280]. Si la valeur n'est pas valide, une erreur est alors enregistrée. En mode verbeux, un message approprié contenant une valeur de notification de Charge utile mal formée sera envoyé.

2. Données de DN – les données de DN sont traitées comme une chaîne UTF-8.

3. La CA DOIT être une autorité de confiance valide de création de politique comme défini par le jeton de politique.

Ces deux éléments d'information, Numéro de série et Données de DN, conjointes, vont alors être utilisées pour l'identification des parties. Ces valeurs sont aussi utilisées pour aider à identifier le certificat lorsque nécessaire.

7.7 Charge utile Certificat

7.7.1 Structure de la charge utile Certificat

La charge utile Certificat fournit le moyen de transport des certificats ou autres informations relatives aux certificats via

GSAKMP et peut apparaître dans tout message GSAKMP. Les charges utiles Certificat DEVRAIENT être incluses dans un échange chaque fois qu'un service de répertoire approprié (par exemple, LDAP [RFC4523]) n'est pas disponible pour distribuer les certificats. Plusieurs charges utiles de certificats PEUVENT être envoyées pour permettre la vérification des chaînes de certificats. À l'inverse, zéro (0) charge utile de certificat peut être envoyée, et le GSAKMP receveur DOIT s'appuyer sur d'autres méchanismes pour récupérer les certificats et les vérifier. La Figure 20 montre le format de la charge utile Certificat.

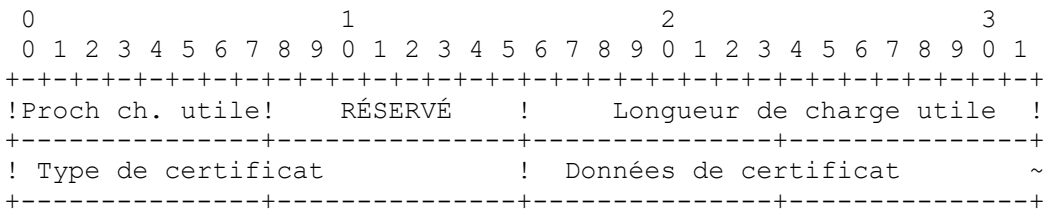


Figure 20 : Format de charge utile Certificat

Les champs de charge utile Certificat sont définis comme suit :

Prochaine charge utile (1 octet) – Identifiant du type de charge utile de la prochaine charge utile dans le message. Si la charge utile actuelle est la dernière du message, ce champ sera alors à 0. Ce champ fournit une capacité de "chaînage". Le Tableau 12 identifie les types de charge utile. Ce champ est traité comme une valeur non signée.

RÉSERVÉ (1 octet) - non utilisé, réglé à 0.

Longueur de charge utile (2 octets) – longueur en octets de la charge utile actuelle, incluant l'en-tête de charge utile générique. Ce champ est traité comme un entier non signé dans le format des octets du réseau.

Type de certificat (2 octets) - Ce champ indique le type de certificat ou les informations relatives au certificat contenues dans le champ Données de certificat. Le Tableau 20 présente les types de charge utile de certificat. Ce champ est traité comme un entier non signé dans le format des octets du réseau.

Données de certificat (longueur variable) – codage réel des données de certificat. Le type du certificat est indiqué par le champ Type/codage de certificat.

Le type de charge utile pour la charge utile Certificat est six (6).

Tableau 20 : types de charge utile Certificat

Type de certificat	**Valeur**	**Description/Défini dans**
Aucun	0	
Réservé	1 - 3	
Certificat X.509v3	4	Ce type DOIT être mis en œuvre
-- Signature		
-- Codage DER		Contient un certificat X.509 codé en DER.
Réservé	5 - 6	
Liste de révocation de certificat	7	Contient une CRL X.509 codée en BER.
Réservé	8 - 9	
Certificat X.509 – Attribut	10	Voir le paragraphe 3.6 de la [RFC4306].
Clé RSA brute	11	Voir le paragraphe 3.6 de la [RFC4306].
Hachage et URL de certificat X.509	12	Voir le paragraphe 3.6 de la [RFC4306].
Hachage et URL de bouquet X.509	13	Voir le paragraphe 3.6 de la [RFC4306].
Réservé à l'IANA	14 -- 49152	
Usage privé	49153 -- 65535	

7.7.2 Traitement de la charge utile Certificat

Lors du traitement de la charge utile Certificat, il DOIT être vérifié que les champs suivants ont les valeurs correctes :

1. Prochaine charge utile, RÉSERVÉ, Longueur de charge utile - ces champs sont traités comme défini au

paragraphe 7.2.2, "Traitement de l'en-tête générique de charge utile".

2. Type de certificat – la valeur du type de certificat DOIT être vérifiée comme type valide de certificat comme défini au Tableau 20. Si la valeur n'est pas valide, une erreur est alors enregistrée. En mode verbeux, un message approprié contenant une valeur de notification de Type de certificat non pris en charge sera envoyé.

3. Données de certificat - ces données de certificat DOIVENT être traitées en accord avec le type de certificat spécifié. Le type va définir le format des données. La réception d'un certificat de l'autorité de confiance de création de politique dans une charge utile Certificat cause l'élimination de la charge utile. Ce certificat reçu NE DOIT PAS être utilisé pour vérifier le message. Le certificat de l'autorité de confiance de création de politique DOIT être récupéré par d'autres moyens.

7.8 Charge utile Signature

7.8.1 Structure de charge utile Signature

La charge utile Signature contient des données générées par la fonction de signature numérique. La signature numérique, comme définie par la dissection de chaque message, couvre le message de l'en-tête de message GSAKMP jusqu'à la charge utile Signature, sans inclure la Longueur des données de signature. La Figure 21 montre le format de la charge utile Signature.

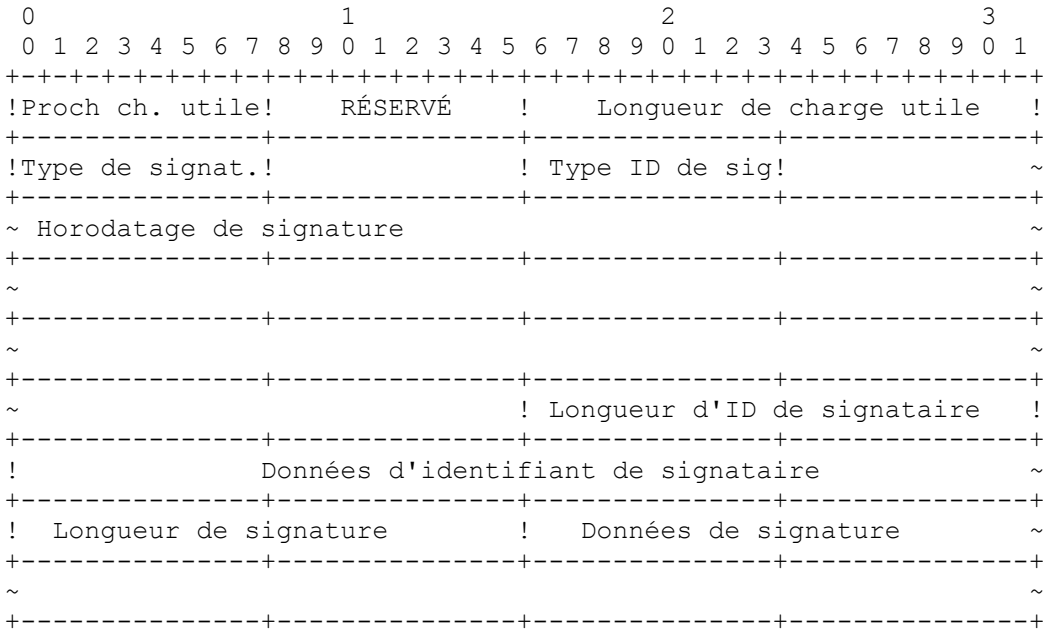


Figure 21 : Format de charge utile Signature

Les champs de la charge utile Signature sont définis comme suit :

Prochaine charge utile (1 octet) – Identifiant pour le type de charge utile de la prochaine charge utile dans le message. Si la charge utile en cours est la dernière du message, ce champ sera alors à 0. Ce champ fournit la capacité de "chaînage". Le Tableau 12 identifie les types de charge utile. Ce champ est traité comme une valeur non signée.

RÉSERVÉ (1 octet) - non utilisé, réglé à 0.

Longueur de charge utile (2 octets) - longueur en octets de la charge utile en cours, incluant l'en-tête de charge utile générique. Ce champ est traité comme un entier non signé dans le format des octets du réseau.

Type de signature (2 octets) - indique le type de signature. Le Tableau 21 présente les types de signature admissibles. Ce champ est traité comme un entier non signé dans le format des octets du réseau.

Tableau 21 : Types de signature

Type de signature	Valeur	Description/Définie dans
DSS/SHA1 avec codage ASN.1/DER (DSS-SHA1-ASN1-DER)	0	Ce type DOIT être pris en charge.
RSA1024-MD5	1	Voir la [RFC3447].
ECDSA-P384-SHA3	2	Voir [FIPS186-2].
Réservé à l'IANA	3 - 41952	
Usage privé	41953 - 65536	

Type d'ID de signature (1 octet) - indique le format des données d'ID de signature. Ces valeurs sont les mêmes que celles définies pour les types d'identifiant de charge utile Identification, qui se trouvent au Tableau 19. Ce champ est traité comme une valeur non signée.

Horodatage de signature (15 octets) – C'est la valeur de l'heure où la signature numérique a été appliquée. Ce champ contient l'horodatage en format UTF-8 AAAAMMJJHHMMSSZ, où AAAA est l'année (0000 - 9999), MM est la valeur numérique du mois (01 - 12), JJ est le jour du mois (01 - 31), HH est l'heure du jour (00 - 23), MM est la minute dans l'heure (00 - 59), SS est la seconde dans la minute (00 - 59), et la lettre Z indique l'heure Zoulou. Ce format est vaguement fondé sur la [RFC3161].

Longueur de l'ID de signataire (2 octets) – longueur en octets de l'identifiant de signataire. Ce champ est traité comme un entier non signé dans le format des octets du réseau.

Données d'ID de signataire (longueur variable) – données qui identifient l'identifiant du signataire (par exemple, DN). Le format de ce champ se fonde sur le champ Type d'ID de signature et est montré lorsque ce type est défini. Le contenu de ce champ DOIT être confronté au jeton de politique pour déterminer l'autorité et l'accès du signataire au sein du contexte du groupe.

Longueur de signature (2 octets) – longueur en octets des données de signature. Ce champ est traité comme un entier non signé dans le format des octets du réseau.

Données de signature (longueur variable) – données qui résultent de l'application de la fonction de signature numérique au message et/ou charge utile GSAKMP.

Le type de charge utile pour la charge utile Signature est huit (8).

7.8.2 Traitement de la charge utile Signature

Lors du traitement de la charge utile Signature, il DOIT être vérifié que les champs suivants ont les valeurs correctes :

1. Prochaine charge utile, RÉSERVÉ, Longueur de charge utile - ces champs sont traités comme défini au paragraphe 7.2.2, "Traitement de l'en-tête générique de charge utile".

2. Type de signature – la valeur de type de signature DOIT être vérifiée comme étant un type de signature valide comme défini au Tableau 21. Si la valeur n'est pas valide, une erreur est alors enregistrée. En mode verbeux, un message approprié contenant une valeur de notification de Charge utile mal formée sera envoyé.

3. Type d'identifiant de signature – la valeur de type d'identifiant de signature DOIT être vérifiée comme étant un type valide d'identifiant de signature comme défini au Tableau 19. Si la valeur n'est pas valide, une erreur est alors enregistrée. En mode verbeux, un message approprié contenant une valeur de notification de Charge utile mal formée sera envoyé.

4. Horodatage de signature - ce champ PEUT être vérifié pour déterminer si l'heure de signature de la transaction est fraîche par rapport aux délais attendus du réseau. Une telle vérification est appropriée pour les systèmes dans lesquels sont désirées des séquences archivées d'événements.

Note : l'age maximum acceptable d'un horodatage de signature par rapport à l'horloge du système local est un paramètre configuré en local qui peut être réglé par son interface de gestion GSAKMP.

5. Données d'identifiant de signature - ce champ va être utilisé pour identifier l'envoyeur. Cette information DOIT alors être utilisée pour confirmer que c'est la bonne partie qui a envoyé ces informations. Ce champ est aussi utilisé pour récupérer la clé publique appropriée du certificat pour vérifier le message.

6. Données de signature – cette valeur DOIT être comparée à la signature recalculée pour vérifier le message. Les informations sur la façon de vérifier les certificats utilisés pour s'assurer de la validité de la signature se trouvent dans la [RFC3280]. C'est seulement après qu'est vérifié le certificat identifié par les données d'identifiant de signature que la signature peut être calculée pour comparer les données de signature pour la vérification de signature. Une erreur potentielle qui peut survenir durant la vérification de signature est Échec d'authentification. Les erreurs potentielles qui peuvent survenir lors du traitement des certificats pour la vérification de signature sont : Certificat invalide, Autorité de certification invalide, Type de certificat non pris en charge, et Certificat indisponible.

Les champs de longueur dans la charge utile Signature sont utilisés pour traiter le reste de la charge utile. Si un champ se trouve être incorrect, la terminaison du traitement DOIT être initiée.

7.9 Charge utile Notification

7.9.1 Structure de charge utile Notification

La charge utile Notification peut contenir des données spécifiques de GSAKMP et du groupe et est utilisée pour transmettre des données d'information, comme des conditions d'erreur, à un homologue GSAKMP. It est possible d'envoyer plusieurs charges utiles Notification indépendantes dans un seul message GSAKMP. La Figure 22 montre le format de la charge utile Notification.

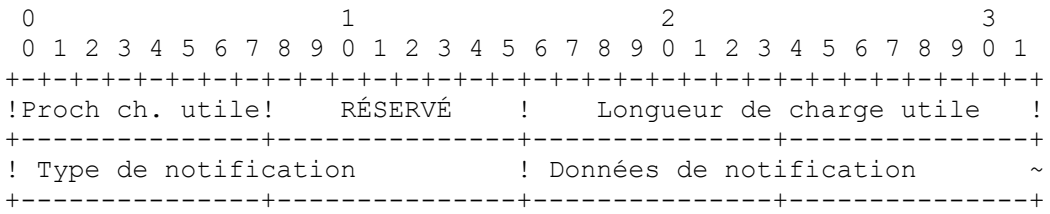


Figure 22 : Format de charge utile Notification

Les champs de la charge utile Notification sont définis comme suit :

Prochaine charge utile (1 octet) – identifiant pour le type de charge utile de la prochaine charge utile dans le message. Si la charge utile en cours est la dernière du message, ce champ sera alors à 0. Ce champ fournit une capacité de "chaînage". Le Tableau 12 identifie les types de charge utile. Ce champ est traité comme une valeur non signée.

RÉSERVÉ (1 octet) - non utilisé, réglé à 0.

Longueur de charge utile (2 octets) – longueur en octets de la charge utile actuelle, incluant l'en-tête de charge utile générique. Ce champ est traité comme un entier non signé dans le format des octets du réseau.

Type de notification (2 octets) – spécifie le type du message de notification. Le Tableau 22 présente les types de charge utile Notification. Ce champ est traité comme un entier non signé dans le format des octets du réseau.

Données de notification (longueur variable) – données d'information ou d'erreur transmises en plus du type de charge utile Notification. Les valeurs de ce champ sont spécifiques du domaine d'interprétation (DOI).

Le type de charge utile pour la charge utile Notification est neuf (9).

Tableau 22 : Types de notification

Type de notification	Valeur
Aucune	0
Type de charge utile invalide	1
Réservé	2 - 3
Version invalide	4

ID de groupe invalide	5
ID de séquence invalide	6
Charge utile mal formée	7
Informations de clé invalides	8
Informations d'ID invalides	9
Réservé	10 - 11
Type de certificat non pris en charge	12
Autorité de certification invalide	13
Échec d'authentification	14
Réservé	15 - 16
Certificat indisponible	17
Réservé	18
Demande non autorisée	19
Réservé	20 - 22
Accusé de réception	23
Réservé	24 - 25
Accusé de non réception	26
Mouchard exigé	27
Mouchard	28
Choix de mécanismes	29
Quitte le groupe	30
Départ accepté	31
Erreur de demande de départ	32
Type d'échange invalide	33
Valeur IPv4	34
Valeur IPv6	35
Interdit par la politique du groupe	36
Interdit pas la politique configurée en local	37
Réservé à l'IANA	38 - 49152
Usage privé	49153 -- 65535

7.9.1.1 Données de notification – type de charge utile Accusé de réception (ACK)

La portion des données de la charge utile Notification de type ACK sert de confirmation de la réception correcte du message Téléchargement de clé ou, lorsque nécessaire, fournit d'autres informations de réception quand elle est incluse dans un message signé. La Figure 23 montre le format des données de notification – type de charge utile Accusé de réception.

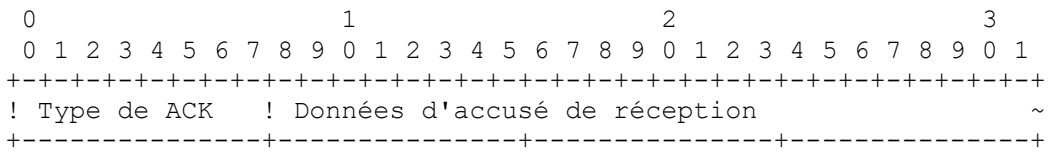


Figure 23 : Format des données de notification – type de charge utile Accusé de réception

Les champs des données de notification - type de charge utile Accusé de réception sont définis comme suit :

Type de ACK (1 octet) – spécifie le type d'accusé de réception. Le Tableau 23 présente les types de charge utile d'accusé de réception de notification. Ce champ est traité comme une valeur non signée.

Tableau 23 : Types d'accusé de réception

Type d'ACK	Valeur	Définition
Simple	0	Portion de données nulle.
Réservé à l'IANA	1 - 192	
Usage privé	193 - 255	

7.9.1.2 Données de notification – type de charge utile Mouchard exigé et Mouchard

La portion de données de la charge utile Notification des types Mouchard exigé et Mouchard contient la valeur du mouchard. La valeur pour ce champ aura été calculée par le GC/KS qui répond et été envoyée au GM. Le GM va prendre la valeur reçue et la copier dans le champ de données de notification de la charge utile Notification de type Mouchard qui est retransmise dans la "Demande d'adhésion avec informations de mouchard" au GC/KS. La valeur du mouchard NE DOIT PAS être modifiée.

Le format de cela est déjà décrit dans la discussion sur les mouchards au paragraphe 5.2.2.

7.9.1.3 Données de notification - type de charge utile Choix de mécanisme

La portion de données de la charge utile Notification de type Choix de mécanisme contient les mécanismes dont le GM demande l'utilisation pour la négociation avec le GC/KS. Ces informations vont être fournies par le GM dans un message RTJ. La Figure 24 montre le format des données de notification – type de charge utile Choix de mécanisme. Plusieurs choix de type|longueur|données sont collés ensemble dans une charge utile de notification pour permettre à un usager de transmettre toutes les informations pertinentes au sein d'une charge utile Notification. La longueur de la charge utile va contrôler l'analyse du champ du choix de mécanisme de données de notification.

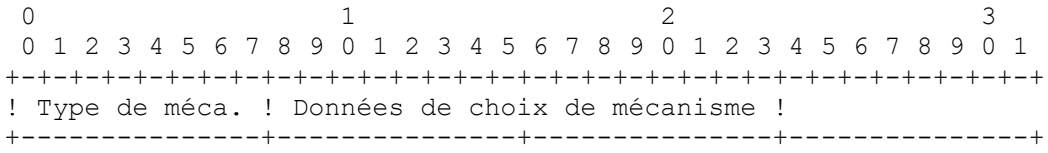


Figure 24 : Format de Données de notification - type de charge utile Choix de mécanisme

Les champs de Données de notification - type de charge utile Choix de mécanisme sont définis comme suit :

Type de mécanisme (1 octet) – spécifie le type de mécanisme. Le Tableau 24 présente les types de mécanisme de choix de mécanisme de notification. Ce champ est traité comme une valeur non signée.

Tableau 24 : Types de mécanismes

Type de mécanisme	Valeur	Référence du tableau de mécanisme
Algorithme de création de clé	0	Tableau 26
Algorithme de chiffrement	1	Tableau 16
Algorithme de hachage de nom occasionnel	2	Tableau 25
Réservé à l'IANA	3 - 192	
Usage privé	193 - 255	

Données de choix de mécanisme (2 octets) – valeur des données pour le type de mécanisme choisi. Les valeurs sont spécifiques de chaque type de mécanisme défini. Tous les tableaux nécessaires pour définir les valeurs qui ne sont pas définies ailleurs (dans cette spécification ou d'autres) sont définis ici. Ce champ est traité comme un entier non signé dans le format des octets du réseau.

Tableau 25 : Types de hachage de nom occasionnel

Type de hachage de nom occasionnel	Valeur	Description
Réservé	0	
SHA-1	1	Ce type DOIT être pris en charge
Réservé à l'IANA	2 - 49152	
Usage privé	49153 - 65535	

7.9.1.4 Données de notification - type de charge utile Valeur IPv4 et IPv6

La portion des données de la charge utile Notification de type Valeur IPv4 et IPv6 contient la valeur IP appropriée dans l'ordre des octets du réseau. Cette valeur va être réglée par le créateur du message pour l'usage du receveur du message.

7.9.2 Traitement de la charge utile Notification

Lors du traitement de la charge utile Notification, il DOIT être vérifié que les champs suivants ont les valeurs correctes :

1. Prochaine charge utile, RÉSERVÉ, Longueur de charge utile - ces champs sont traités comme défini au paragraphe 7.2.2, "Traitement de l'en-tête générique de charge utile".

2. Type de notification – la valeur du type de notification DOIT être vérifiée comme étant un type de notification défini au Tableau 22. Si la valeur n'est pas valide, une erreur est alors enregistrée. En mode verbeux, un message approprié contenant une valeur de notification de Charge utile mal formée sera envoyé.

3. Données de notification - ces données de notification DOIVENT être traitées en accord avec le type de notification spécifié. Le type va définir le format des données. Lors du traitement de ces données, tout champ de type DOIT être confronté au tableau approprié pour les valeurs correctes. Si le contenu des données de notification n'est pas valide, une erreur est alors enregistrée. En mode verbeux, un message approprié contenant une valeur de notification de Charge utile mal formée sera envoyé.

7.10 Charge utile Identifiant de fabricant

7.10.1 Structure de charge utile Identifiant de fabricant

La charge utile Identifiant de fabricant contient une constante définie par le fabricant. La constante est utilisée par les fabricants pour identifier et reconnaître les instances distantes de leurs mises en œuvre. Ce mécanisme permet à un fabricant d'expérimenter de nouvelles caractéristiques tout en conservant la rétro compatibilité. La Figure 25 montre ce format de charge utile.

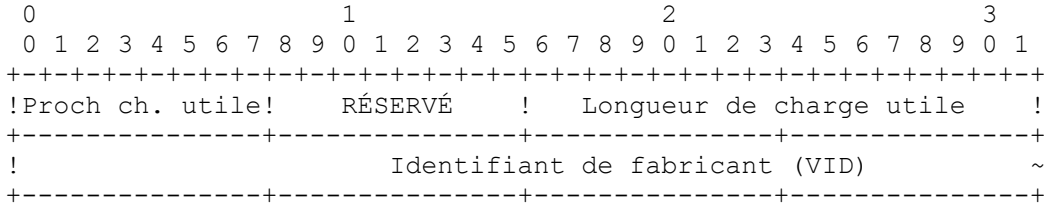


Figure 25 : Format de charge utile Identifiant de fabricant

Une charge utile Identifiant de fabricant PEUT annoncer que l'envoyeur est capable d'accepter certaines extensions au protocole, ou elle PEUT simplement identifier la mise en œuvre comme une aide au débogage. Une charge utile Identifiant de fabricant NE DOIT PAS changer l'interprétation des informations définies dans la présente spécification. Plusieurs charges utiles Identifiant de fabricant PEUVENT être envoyées. Une mise en œuvre N'EST PAS OBLIGÉE d'envoyer du tout de charge utile Identifiant de fabricant.

Une charge utile Identifiant de fabricant peut être envoyée au titre de tout message. La réception d'une charge utile Identifiant de fabricant familière permet à une mise en œuvre d'utiliser les numéros d'usage privé décrits dans la présente spécification – charges utiles privées, échanges privés, notifications privées, etc. Cela implique que toutes les règles de traitement définies pour toutes les charges utiles sont maintenant modifiées pour reconnaître toutes les valeurs définies par cet identifiant de fabricant pour tous les champs de toutes les charges utiles. Les identifiants de fabricants qui ne sont pas familiers DOIVENT être ignorés.

Les rédacteurs de projets Internet qui souhaitent étendre le présent protocole DOIVENT définir une charge utile Identifiant de fabricant pour annoncer la capacité de mettre en œuvre l'extension dans le projet Internet. Il est prévu que les projets Internet qui obtiennent leur acceptation et sont normalisés vont recevoir des valeurs allouées dans la gamme Réservé de l'IANA, et l'exigence d'utiliser une charge utile Identifiant de fabricant va disparaître.

Les champs de la charge utile Identifiant de fabricant sont définis comme suit :

Prochaine charge utile (1 octet) – Identifiant pour le type de charge utile de la prochaine charge utile dans le message. Si la charge utile en cours est la dernière du message, ce champ sera alors à 0. Ce champ fournit la capacité de "chaînage". Le Tableau 12 identifie les types de charge utile. Ce champ est traité comme une valeur non signée.

RÉSERVÉ (1 octet) - non utilisé, réglé à 0.

Longueur de charge utile (2 octets) – longueur en octets de la charge utile en cours, incluant l'en-tête de charge utile générique. Ce champ est traité comme un entier non signé dans le format des octets du réseau.

Identifiant de fabricant (longueur variable) – valeur de l'identifiant de fabricant. La longueur minimum de ce champ est de quatre (4) octets. Il est de la responsabilité de la personne qui choisit l'identifiant de fabricant de le choisir de façon à assurer son unicité en dépit de l'absence de tout registre central des identifiants. La bonne pratique est d'inclure un nom de société, un nom de personne, ou des données de type similaire. Un résumé de message d'une longue chaîne unique est préférable à la longue chaîne unique elle-même.

Le type de charge utile pour la charge utile Identifiant de fabricant est dix (10).

7.10.2 Traitement de la charge utile Identifiant de fabricant

Lors du traitement de la charge utile Identifiant de fabricant, il DOIT être vérifié que les champs suivants ont les valeurs correctes :

1. Prochaine charge utile, RÉSERVÉ, Longueur de charge utile - ces champs sont traités comme défini au paragraphe 7.2.2, "Traitement de l'en-tête générique de charge utile".

2. Identifiant de fabricant – les données de l'identifiant de fabricant DOIVENT être traitées pour déterminer si la valeur de l'identifiant de fabricant est reconnue par la mise en œuvre. Si la valeur de l'identifiant de fabricant n'est pas reconnue, ces informations sont enregistrées, sans considération du mode (concis ou verbeux). Le traitement du message DOIT se poursuivre sans considération de la reconnaissance de cette valeur.

Il est recommandé que les mises en œuvre qui veulent utiliser des informations spécifiques de l'identifiant de fabricant tentent de traiter les charges utiles Identifiant de fabricant d'un message entrant avant le traitement du reste du message. Cela va permettre à la mise en œuvre de reconnaître que quand elle traite d'autres charges utiles elle peut utiliser le plus grand ensemble de valeurs de champs de charge utile (valeurs d'usage privé, etc.) comme défini par les identifiants de fabricant reconnus.

7.11 Charge utile Création de clé

7.11.1 Structure de la charge utile Création de clé

La charge utile Création de clé contient des informations utilisées pour créer les clés de chiffrement de clé. Les attributs de sécurité pour cette charge utile sont fournies par le jeton de politique. La Figure 26 montre le format de cette charge utile.

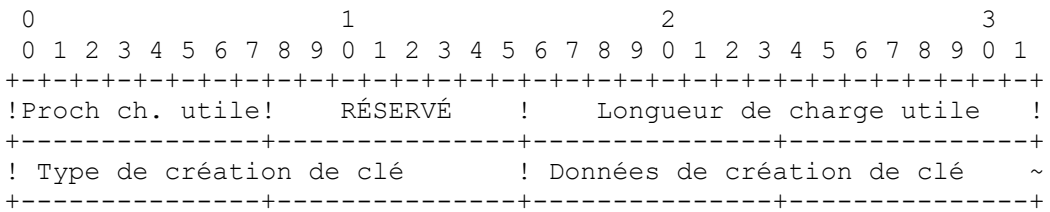


Figure 26 : Format de charge utile Création de clé

Les champs de la charge utile Création de clé sont définis comme suit :

Prochaine charge utile (1 octet) – Identifiant du type de charge utile de la prochaine charge utile dans le message. Si la charge utile en cours est la dernière du message, ce champ sera alors à 0. Ce champ fournit la capacité de "chaînage". Le Tableau 12 identifie les types de charge utile. Ce champ est traité comme une valeur non signée.

RÉSERVÉ (1 octet) - non utilisé, réglé à 0.

Longueur de charge utile (2 octets) – longueur en octets de la charge utile en cours, incluant l'en-tête de charge utile générique. Ce champ est traité comme un entier non signé dans le format des octets du réseau.

Type de création de clé (2 octets) – spécifie le type de création de clé utilisé. Le Tableau 26 identifie les types d'informations de création de clé. Ce champ est traité comme un entier non signé dans le format des octets du réseau.

Données de création de clé (longueur variable) – contient les informations de création de clé. Les valeurs pour ce champ sont spécifiques du groupe, et le format est spécifié par le champ Type de création de clé.

Le type de charge utile pour le paquet Création de clé est onze (11).

Tableau 26 : Types d'informations de création de clé

Type de création de clé	Valeur	Définition/Défini dans
Réservé	0 – 1	
Groupe MODP Diffie-Hellman à 1024 bits tronqué	2	Ce type DOIT être pris en charge. Défini au paragraphe B.2 de la [RFC4306]. Si le résultat du traitement est plus long que nécessaire pour le mécanisme défini, on utilise les X premiers bits de moindre poids et on tronque le reste.
Réservé	3 – 13	
Groupe MODP Diffie-Hellman à 2048 bits tronqué	14	Défini dans la [RFC3526]. Si le résultat du traitement est plus long que nécessaire pour le mécanisme défini, on utilise les X premiers bits de moindre poids et on tronque le reste.
Réservé à l'IANA	15 – 49152	
Usage privé	49153 – 65535	

7.11.2 Traitement de la charge utile Création de clé

Les spécificités de la charge utile Création de clé sont définies au paragraphe 7.11.
Lors du traitement de la charge utile Création de clé, il DOIT être vérifié que les champs suivants ont les valeurs correctes :

1. Prochaine charge utile, RÉSERVÉ, Longueur de charge utile - ces champs sont traités comme défini au paragraphe 7.2.2, "Traitement de l'en-tête générique de charge utile".

2. Type de création de clé – la valeur du type de création de clé DOIT être vérifiée comme étant un type de création de clé valide comme défini au Tableau 26. Si la valeur n'est pas valide, une erreur est alors enregistrée. En mode verbeux, un message approprié contenant une valeur de notification de Charge utile mal formée va être envoyé.

3. Données de création de clé - ces données de création de clé DOIVENT être traitées en accord avec le type de création de clé spécifié pour générer la KEK pour protéger les informations à envoyer dans le message approprié. Le type va définir le format des données.

Les mises en œuvre qui veulent déduire d'autres clés du matériel initial de création de clé (par exemple, matériel DH de chiffrement de secret) DOIVENT définir un type de création de clé autre qu'un de ceux montrés au Tableau 26. Le nouveau type de création de clé doit spécifier cet algorithme de déduction, pour lequel la KEK PEUT être une des clés déduites.

7.12 Charge utile Nom occasionnel

7.12.1 Structure de charge utile Nom occasionnel

La charge utile Nom occasionnel contient des données aléatoires utilisées pour garantir la fraîcheur durant un échange et protéger contre les attaques en répétition. La Figure 27 montre le format de la charge utile Nom occasionnel.

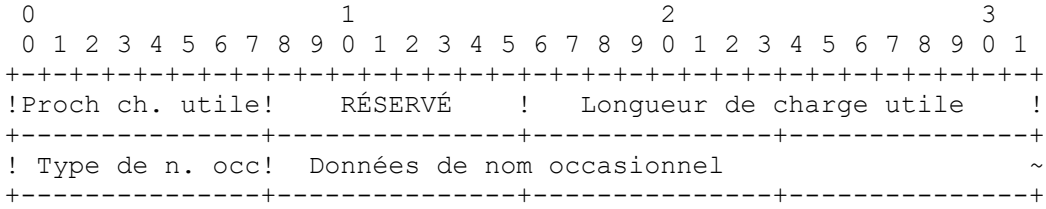


Figure 27 : Format de la charge utile Nom occasionnel

Les champs de charge utile Nom occasionnel sont définis comme suit :

Prochaine charge utile (1 octet) – Identifiant pour le type de charge utile de la prochaine charge utile dans le message. Si la charge utile en cours est la dernière du message, ce champ sera alors à 0. Ce champ fournit une capacité de "chaînage". Le Tableau 12 identifie les types de charge utile. Ce champ est traité comme une valeur non signée.

RÉSERVÉ (1 octet) - non utilisé, réglé à 0.

Longueur de charge utile (2 octets) – longueur en octets de la charge utile en cours, incluant l'en-tête de charge utile générique. Ce champ est traité comme un entier non signé dans le format des octets du réseau.

Type de nom occasionnel (1 octet) – spécifie le type de nom occasionnel utilisé. Le Tableau 27 identifie les types de noms occasionnels. Ce champ est traité comme une valeur non signée.

Tableau 27 : Types de nom occasionnel

Type de nom occasionnel	Valeur	Définition
Aucun	0	
Initiateur (Nonce_I)	1	
Répondant (Nonce_R)	2	
Combiné (Nonce_C)	3	Hachage de (valeur d'initiateur, valeur de répondant). Le type de hachage vient de la politique (par exemple, définition de suite de sécurité du jeton de politique).
Réservé à l'IANA	4 - 192	
Usage privé	192 - 255	

Données de nom occasionnel (longueur variable) – contient les informations de nom occasionnel. Les valeurs pour ce champ sont spécifiques du groupe, et le format est spécifié par le champ Type de nom occasionnel. Si aucune information spécifique du groupe n'est fournie, la longueur minimum de ce champ est 4 octets.

Le type de charge utile pour la charge utile Nom occasionnel est douze (12).

7.12.2 Traitement de la charge utile Nom occasionnel

Lors du traitement de la charge utile Nom occasionnel, il DOIT être vérifié que les champs suivants ont les valeurs correctes :

1. Prochaine charge utile, RÉSERVÉ, Longueur de charge utile - ces champs sont traités comme défini au paragraphe 7.2.2, "Traitement de l'en-tête générique de charge utile".

2. Type de nom occasionnel – il DOIT être vérifié que la valeur du nom occasionnel est d'un type de nom occasionnel valide comme défini au Tableau 27. Si la valeur n'est pas valide, une erreur est alors enregistrée. En mode verbeux, un message approprié contenant une valeur de notification de Charge utile mal formée sera envoyé.

3. Données de nom occasionnel – ce sont les données du nom occasionnel et elles doivent être vérifiées en accord avec leur contenu. La taille de ce champ est définie au paragraphe 7.12, "Charge utile Nom occasionnel". Voir au paragraphe 5.2, "Établissement de groupe" l'interprétation de ce champ.

8. Diagrammes d'état GSAKMP

La Figure 28 présente les états rencontrés dans l'utilisation de ce protocole. Le Tableau 28 définit les états. Le Tableau 29 définit les transitions.

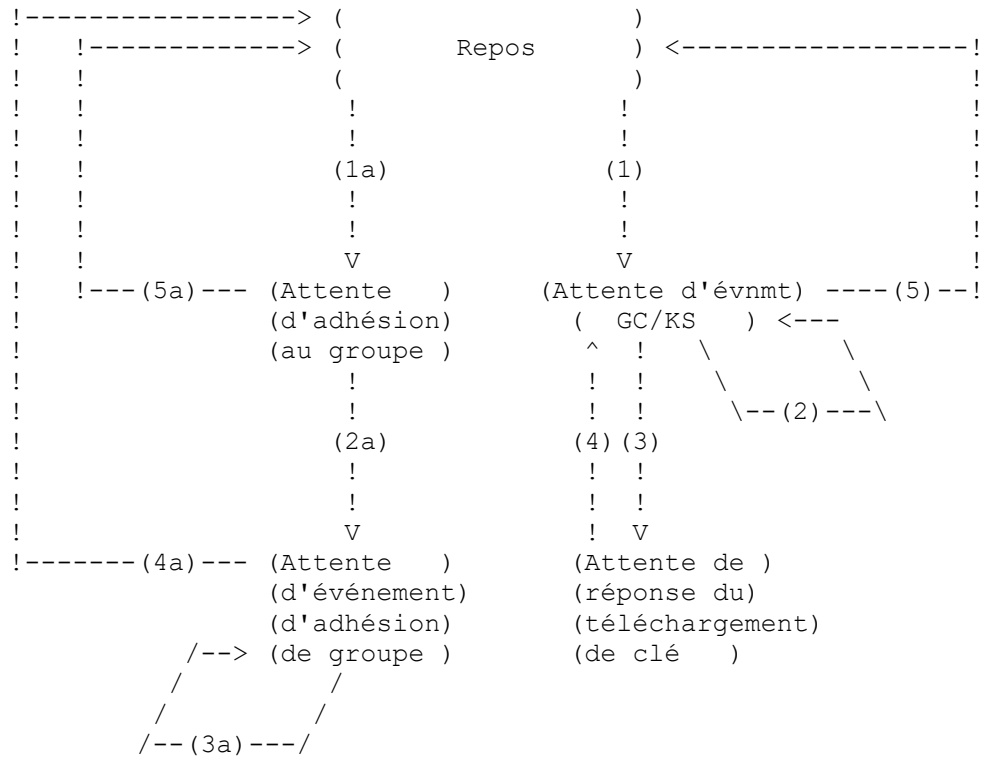


Figure 28 : Diagramme des états GSAKMP

Tableau 28 : États GSAKMP

Repos : l'application GSAKMP est en attente d'entrées
Attente d'événement du GC/KS : le GC/KS est sous tension et en fonctionnement, en attente d'événements
Attente de réponse du téléchargement de clé : le GC/KS a envoyé le téléchargement de clé, il attend la réponse du GM.
Attente de l'adhésion au groupe : le GM est en cours d'adhésion au groupe.
Attente d'événement d'adhésion au groupe : le GM a la clé du groupe, il attend des messages de gestion de groupe.

Tableau 29 : Événements de transition d'états

Transition 1 : Commande de création de groupe
Transition 2 : Réception d'une mauvaise RTJ
: Réception d'une commande valide pour changer une adhésion au groupe
: Envoi de message compromis x fois
: Désenregistrement de membre
Transition 3 : Réception d'une RTJ valide
Transition 4 : Fin de temporisation
: Réception d'accusé de réception
: Réception d'accusé de non réception
Transition 5 : Commande de suppression de groupe
Transition 1a : Commande d'adhésion au groupe
Transition 2a : Envoi d'accusé de réception
Transition 3a : Réception de messages de gestion de groupe
Transition 4a : Commande de suppression de groupe
: Commande de désenregistrement
Transition 5a : Fin de temporisation
: Erreurs d'échec de message

9. Considérations relatives à l'IANA

9.1 Allocation des numéros d'accès

L'IANA a fourni le numéro d'accès GSAKMP 3761 dans les deux espaces UDP et TCP. Toutes les mises en œuvre DOIVENT utiliser cette allocation d'accès de la manière appropriée

9.2 Contenu du registre initial de l'IANA

Les entrées de registre suivantes ont été créées :
Types d'identification de groupe GSAKMP (paragraphe 7.1.1)
Types de charge utile GSAKMP (paragraphe 7.1.1)
Types d'échange GSAKMP (paragraphe 7.1.1)
Types de jeton de politique GSAKMP (paragraphe 7.3.1)
Types d'éléments de données de téléchargement de clé GSAKMP (paragraphe 7.4.1)
Types de clé de chiffrement GSAKMP (paragraphe 7.4.1.1)
Types d'événement de changement de clé GSAKMP (paragraphe 7.5.1)
Classification d'identification GSAKMP (paragraphe 7.6.1)
Types d'identification GSAKMP (paragraphe 7.6.1)
Types de certificat GSAKMP (paragraphe 7.7.1)
Types de signature GSAKMP (paragraphe 7.8.1)
Types de notification GSAKMP (paragraphe 7.9.1)
Types d'accusé de réception GSAKMP (paragraphe 7.9.1.1)
Types de mécanisme GSAKMP (paragraphe 7.9.1.3)
Types de hachage de nom occasionnel GSAKMP (paragraphe 7.9.1.3)
Types de création de clé GSAKMP (paragraphe 7.11.1)
Types de nom occasionnel GSAKMP (paragraphe 7.12.1)

Les changements et ajouts aux registres suivants sont par action de normalisation de l'IETF :
Types d'identification de groupe GSAKMP
Types de charge utile GSAKMP
Types d'échange GSAKMP
Types de jeton de politique GSAKMP
Types d'éléments de données de téléchargement de clé GSAKMP
Types d'événement de changement de clé GSAKMP
Classification d'identification GSAKMP
Types de notification GSAKMP
Types d'accusé de réception GSAKMP
Types de mécanisme GSAKMP
Types de nom occasionnel GSAKMP

Les changements et ajouts aux registres suivants sont par revue d'expert :
Types de clé de chiffrement GSAKMP
Types d'identification GSAKMP
Types de certificat GSAKMP
Types de signature GSAKMP
Types de hachage de nom occasionnel GSAKMP
Types de création de clé GSAKMP

10. Remerciements

Le présent document est le résultat de l'effort de collaboration de nombreuses personnes. Si il n'y avait pas de limite au nombre des auteurs qui peuvent apparaître sur une RFC, les personnes suivantes auraient été citées, par ordre alphabétique : Haitham S. Cruickshank de l'Université du Surrey, Sunil Iyengar de l'Université du Surrey, Gavin Kenny de LogicaCMG, Patrick McDaniel de AT&T Labs Research, et Angela Schuett du NSA.

Les personnes suivantes méritent la reconnaissance et des remerciements pour leurs contributions, qui ont largement amélioré ce protocole : Eric Harder est un auteur de GSAKMP tunnelé, dont les concepts se trouvent aussi dans GSAKMP.

Rod Fleischer, aussi auteur de GSAKMP tunnelé, et Peter Lough ont tous deux réalisé le codage d'un prototype du logiciel GSAKMP et ont aidé à définir de nombreuses parties du protocole qui étaient au mieux imprécises. Andrew McFarland et Gregory Bergren ont fourni une analyse critique des premières versions de la spécification. Ran Canetti a analysé la sécurité du protocole et fourni des suggestions sur le déni de service qui ont conduit à la "protection par mouchard" facultative.

11. Références

11.1 Références normatives

[DH77] Diffie, W., and M. Hellman, "New Directions in Cryptography", IEEE Transactions on Information Theory, juin 1977.

[FIPS186-2] NIST, "Digital Signature Standard", FIPS PUB 186-2, National Institute of Standards and Technology, U.S. Department of Commerce, janvier 2000.

[FIPS196] NIST, "Entity Authentication Using Public Key Cryptography," Federal Information Processing Standards Publication 196, février 1997.

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

[RFC2409] D. Harkins et D. Carrel, "L'échange de clés Internet (IKE)", novembre 1998. *(Obsolète, voir la* RFC4306)

[RFC2412] H. Orman, "Protocole OAKLEY de détermination de clés", novembre 1998. *(Information)*

[RFC2627] D. Wallner, E. Harder, R. Agee,"Gestion de clés en diffusion groupée : problèmes et architectures", juin 1999. *(Info.)*

[RFC3280] R. Housley, W. Polk, W. Ford et D. Solo, "Profil de certificat d'infrastructure de clé publique X.509 et de liste de révocation de certificat (CRL) pour l'Internet", avril 2002. *(Obsolète, voir* RFC5280)

[RFC3629] F. Yergeau, "UTF-8, un format de transformation de la norme ISO 10646", STD 63, novembre 2003.

[RFC4306] C. Kaufman, "Protocole d'échange de clés sur Internet (IKEv2)", décembre 2005. *(Obsolète, voir la* RFC5996)

[RFC4514] K. Zeilenga, éd., "Protocole léger d'accès à un répertoire (LDAP) : Représentation de chaîne des noms distinctifs", juin 2006.

[RFC4534] A Colegrove, H Harney, "Jeton de politique de sécurité de groupe, v1", juin 2006. *(P.S.)*

11.2 Références pour information

[BMS] Balenson, D., McGrew, D., and A. Sherman, "Key Management for Large Dynamic Groups: One-Way Function Trees and Amortized Initialization", Travail en cours, février 1999.

[HCM] H. Harney, A. Colegrove, P. McDaniel, "Principles of Policy in Secure Groups", Proceedings of Network and Distributed Systems Security 2001 Internet Society, San Diego, CA, février 2001.

[HHMCD01] Hardjono, T., Harney, H., McDaniel, P., Colegrove, A., and P. Dinsmore, "Group Security policy token: Definition and Payloads", Travail en cours, août 2003.

[RFC2093] H. Harney, C. Muckenhirn, "Spécification du protocole de gestion de clé de groupe (GKMP)", juillet 1997. *(Exp)*

[RFC2094] H. Harney, C. Muckenhirn, "Architecture du protocole de gestion de clé de groupe (GKMP)", juillet 1997.

(Exp.)

[RFC2408] D. Maughan, M. Schertler, M. Schneider et J. Turner, "Protocole Internet d'association de sécurité et gestion de clés (ISAKMP)", novembre 1998. *(Obsolète, voir la* RFC4306)

[RFC2451] R. Pereira, R. Adams, "Algorithmes de chiffrement ESP en mode CBC", novembre 1998. *(P.S.)*

[RFC2522] P. Karn, W. Simpson, "Photuris : Protocole de gestion de clé de session", mars 1999. *(Expérimentale)*

[RFC2974] M. Handley, C. Perkins, E. Whelan, "Protocole d'annonce de session (SAP)", octobre 2000. *(Expérimentale)*

[RFC3161] C. Adams, P. Cain, D. Pinkas et R. Zuccherato, "Protocole d'horodatage (TSP) d'infrastructure de clé publique X.509 pour l'Internet", août 2001.

[RFC3261] J. Rosenberg et autres, "SIP : Protocole d'initialisation de session", juin 2002. (*Mise à jour par* 3265, 3853, 4320, 4916, 5393, 6665, 8217, 8760)

[RFC3447] J. Jonsson et B. Kaliski, "Normes de cryptographie à clés publiques (PKCS) n° 1 : Spécifications de la cryptographie RSA version 2.1", février 2003. *(Obsolète, remplacée par* RFC8017) *(Information)*

[RFC3526] T. Kivinen et M. Kojo, "Groupes supplémentaires d'exponentiation modulaire (MODP) Diffie-Hellman pour l'échange de clés Internet (IKE)", mai 2003.

[RFC3740] T. Hardjono et B. Weis, "Architecture de sécurité de groupe de diffusion groupée", mars 2004. *(Information)*

[RFC4086] D. Eastlake 3rd, J. Schiller, S. Crocker, "Exigences d'aléa pour la sécurité", juin 2005. (*Remplace* RFC1750) (BCP0106)

[RFC4523] K. Zeilenga, "Protocole léger d'accès à un répertoire (LDAP) : Définitions de schémas pou les certificats X.509", juin 2006.

Appendice A. Informations sur LKH

Cet appendice donne une vue d'ensemble de LKH, définit les valeurs des champs dans les messages GSAKMP qui sont spécifiques de LKH, et donne un exemple d'un message Événement de changement de clé qui utilise le schéma de LKH.

A.1 Généralités sur LKH

LKH fournit une topologie de distribution des clés pour un changement de clés de groupe. Il change les clés d'un groupe sur la base d'une structure arborescente et de clés de sous groupe. Dans l'arborescence LKH de la Figure 29, les membres sont représentés par les nœuds feuilles de l'arborescence, Tandis que les nœuds intermédiaires représentent des groupes de clés abstraits. Un membre va posséder plusieurs clés : la clé de protection du trafic du groupe (GTPK), les clés de sous groupe pour chaque nœud sur son chemin vers la racine de l'arborescence, et une clé personnelle. Par exemple, le membre marqué #3 aura la GTPK, la clé A, la clé D, et la clé 3.

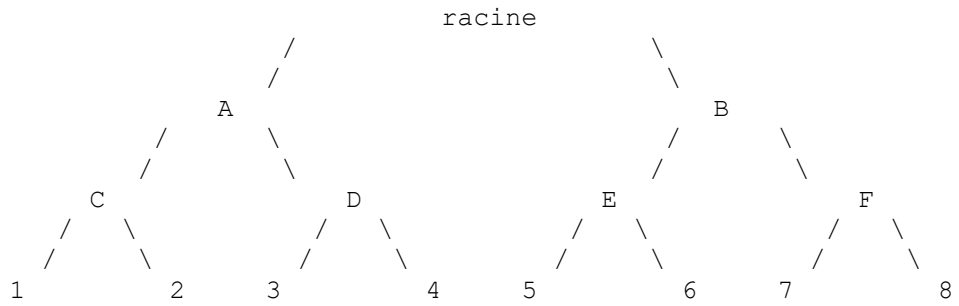


Figure 29 : Arborescence LKH

Cette topologie de fourniture de clés assure un changement de clés rapide à tous les membres du groupe, sauf compromis. Si le membre 3 est compromis, la nouvelle GTPK (GTPK') va devoir être distribuée au groupe avec une clé non possédée par le membre 3. De plus, les nouvelles clés A et D (Clé A' et Clé D') vont aussi avoir besoin d'être distribuées en toute sécurité aux autres membres de ces sous arborescences. Chiffrer la GTPK' avec la clé B distribuerait en toute sécurité cette clé aux membres 5, 6, 7, et 8. La clé C peut être utilisée pour chiffrer la GTPK' et la clé A' pour les membres 1 et 2. Le plus proche voisin du membre 3, le membre 4, peut obtenir la GTPK', la clé D', et la clé A' chiffrées sous sa clé personnelle, la clé 4. À la fin de ce processus, les clés du groupe sont changées en toute sécurité avec le membre 3 complètement exclu.

A.2 LKH et GSAKMP

Quand on utilise LKH avec GSAKMP, les problèmes suivants exigent une certaine attention :

1. Numéro de version de changement de clé – le numéro de version de changement de clé dans la matrice de changement de clé de la charge utile Téléchargement de clé DOIT contenir la valeur un (1).

2. Version d'algorithme – la version d'algorithme dans la charge utile Événement de changement de clé DOIT contenir la valeur un (1).

3. Degré de l'arborescence - l'arborescence LKH utilisée peut être de n'importe quel degré ; il n'est pas nécessairement binaire.

4. Identification de nœud - chaque nœud dans l'arborescence est traité comme une KEK. Une KEK est juste une clé particulière. Comme le déclare la règle pour toutes les clés dans GSAKMP, l'ensemble de KeyID et de KeyHandle DOIT être unique. Une suggestion sur la façon de faire cela est donnée dans cette section.

5. Identifiant de clé et bride enveloppants - c'est l'identifiant de clé et la bride du nœud LKH utilisé pour envelopper/chiffrer les données dans les données d'événement de changement de clé.

Pour la discussion qui suit, se référer à la Figure 30.

Légende :
o : nœud dans l'arborescence LKH
N : ligne contenant le numéro du nœud KeyID
L : ligne contenant le numéro de MemberID pour SEULEMENT toutes les feuilles

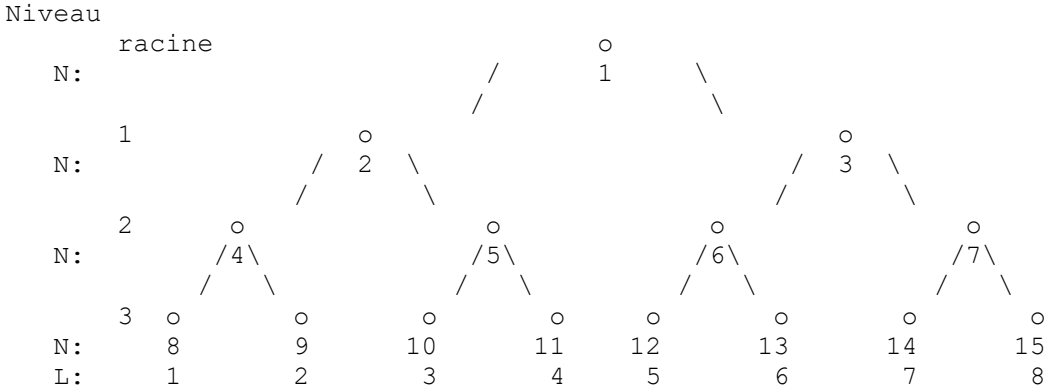


Figure 30 : Arborescence LKH de GSAKMP

Pour garantir l'unicité de l'identifiant de clé, le contrôleur de changement de clé DEVRAIT construire une arborescence virtuelle et étiqueter l'identifiant de clé de chaque nœud, en faisant une recherche en première largeur d'une arborescence complètement remplie sans considérer si l'arborescence est ou non réellement pleine. Pour simplifier cet exemple, la racine de l'arborescence a reçu la valeur d'identifiant de clé de un (1). Ces valeurs d'identifiant de clé vont être statiques tout au long de la vie de cette arborescence. De plus, la matrice de changement de clés distribuée aux GM exige une valeur d'identifiant de membre associée pour être distribuée avec la charge utile Téléchargement de clé. Ces valeurs d'identifiant de membre DOIVENT être uniques. Donc, l'ensemble associé à chaque nœud feuille (les nœuds de cette feuille jusqu'à la racine) reçoivent un identifiant de membre. Dans cet exemple, le nœud feuille de gauche reçoit la valeur d'identifiant de membre de un (1). Ces deux ensembles de valeurs, les identifiants de clé (représentés sur les lignes N) et les identifiants de

membre (représentés sur la ligne L) vont donner des informations suffisantes dans les charges utiles Téléchargement de clé et Événement de changement de clé pour disséminer les informations. La bride de clé associée à ces clés est régénérée chaque fois que la clé est remplacée dans l'arborescence du fait d'une compromission.

A.3 Exemples de LKH

Définition des valeurs:
0xLLLL – valeur de longueur
0xHHHHHHHH# - valeur de bride
AAAAMMJJHHMMSSZ – valeur de l'heure

A.3.1 Exemple de téléchargement de clé LKH

Ce paragraphe donne un exemple des données pour la charge utile Téléchargement de clé. Le GM va être MemberID 1 et ses clés associées. Les données montrées sont celles qui suivent l'en-tête générique de charge utile.

| GTPK | MemberID 1 | KeyID 2 | KeyID 4 | KeyID 8

Nombre d'éléments - 0x0002
Élément n° 1 :
Type d'élément de données de téléchargement de clé - 0x00 (GTPK)
Longueur d'élément de données de téléchargement de clé - 0xLLLL
Type de clé - 0x03 (3DES`CBC64`192)
Identifiant de clé - KEY1
Bride de clé - 0xHHHHHHH0
Date de création de clé - AAAAMMJJHHMMSSZ
Date d'expiration de clé - AAAAMMJJHHMMSSZ
Données de clé - variable, selon la définition de la clé
Élément n° 2 :
Type d'élément de données de téléchargement de clé - 0x01 (Rekey - LKH)
Longueur d'élément de données de téléchargement de clé - 0xLLLL
Numéro de version de changement de clé - 0x01
Identifiant de membre - 0x00000001
Nombre de clés KEK - 0x0003
KEK n° 1 :
Type de clé - 0x03 (3DES`CBC64`192)
Identifiant de clé - 0x00000002
Bride de clé - 0xHHHHHHH2
Date de création de clé - AAAAMMJJHHMMSSZ
Date d'expiration de clé - AAAAMMJJHHMMSSZ
Données de clé - variable, selon la définition de la clé
KEK n° 2 :
Type de clé - 0x03 (3DES`CBC64`192)
Identifiant de clé - 0x00000004
Bride de clé - 0xHHHHHHH4
Date de création de clé - AAAAMMJJHHMMSSZ
Date d'expiration de clé - AAAAMMJJHHMMSSZ
Données de clé - variable, selon la définition de la clé
KEK n° 3 :
Type de clé - 0x03 (3DES`CBC64`192)
Identifiant de clé - 0x00000008
Bride de clé - 0xHHHHHHH8
Date de création de clé - AAAAMMJJHHMMSSZ
Date d'expiration de clé - AAAAMMJJHHMMSSZ
Données de clé - variable, selon la définition de la clé

A.3.2 Exemple d'événement de changement de clé LKH

Ce paragraphe donne un exemple de données pour la charge utile Événement de changement de clé. Le GM qui a

MemberID 6 va être expulsé du groupe. Les données montrées suivent l'en-tête générique de charge utile.

| Type d'événement de changement de clé | GroupID | Date/heure | Type de changement de clé | Version d'algorithme | Nombre de paquets | { (GTPK)2, (GTPK, 3', 6')12, (GTPK, 3')7 }

Ces données montrent que trois paquets sont transmis. On lit chaque paquet comme :
a) GTPK enveloppé dans KeyID 2 LKH
b) GTPK, KeyID LKH 3' & 6', tous enveloppés dans KeyID 12 LKH
c) GTPK et KeyID LKH 3', tous enveloppés dans KeyID 7 LKH

Note : bien que dans cet exemple plusieurs clés soient chiffrées sous une seule clé, d'autres appariements sont légaux (par exemple, (GTPK)2, (GTPK)3', (3')6', (3')7', (6')12).

On montre le format de toutes les données d'en-tête et le paquet (b).

Type d'événement de changement de clé - 0x01 (GSAKMP`LKH)
Identifiant de groupe - 0xAABBCCDD 0x12345678
Horodatage - AAAAMMJJHHMMSSZ
Type d'événement de changement de clé - 0x01 (GSAKMP`LKH)
Version d'algorithme - 0x01
Nombre de paquets RkyEvt - 0x0003
Pour le paquet (b) :
Longueur de paquet - 0xLLLL
Identifiant de clé enveloppante - 0x000C
Bride de clé enveloppante - 0xHHHHHHHD
Nombre de paquetages de clés - 0x0003
- Paquetage de clé 1 :
  - Type de paquetage de clé - 0x00 (GTPK)
  - Longueur du paquetage - 0xLLLL
  - Type de clé - 0x03 (3DES`CBC64`192)
  - Identifiant de clé - KEY1
  - Bride de clé - 0xHHHHHHH0
  - Date de création de clé - AAAAMMJJHHMMSSZ
  - Date d'expiration de clé - AAAAMMJJHHMMSSZ
  - Données de clé - variable, selon la définition de la clé
- Paquetage de clé 2 :
  - Type de paquetage de clé - 0x01 (Rekey - LKH)
  - Longueur du paquetage - 0xLLLL
  - Type de clé - 0x03 (3DES`CBC64`192)
  - Identifiant de clé - 0x00000003
  - Bride de clé - 0xHHHHHHH3
  - Date de création de clé - AAAAMMJJHHMMSSZ
  - Date d'expiration de clé - AAAAMMJJHHMMSSZ
  - Données de clé - variable, selon la définition de la clé
- Paquetage de clé 3 :
  - Type de paquetage de clé - 0x01 (Rekey - LKH)
  - Longueur du paquetage - 0xLLLL
  - Type de clé - 0x03 (3DES`CBC64`192)
  - Identifiant de clé - 0x00000006
  - Bride de clé - 0xHHHHHHH6
  - Date de création de clé - AAAAMMJJHHMMSSZ
  - Date d'expiration de clé - AAAAMMJJHHMMSSZ
  - Données de clé - variable, selon la définition de la clé

Adresse des auteurs

Hugh Harney
SPARTA, Inc.
7110 Samuel Morse Drive
Columbia, MD 21046
USA
téléphone : (443) 430-8032
mél : hh@sparta.com

Uri Meth
SPARTA, Inc.
7110 Samuel Morse Drive
Columbia, MD 21046
USA
téléphone : (443) 430-8058
mél : umeth@sparta.com

Andrea Colegrove
SPARTA, Inc.
7110 Samuel Morse Drive
Columbia, MD 21046
USA
téléphone : (443) 430-8014
mél acc@sparta.com

George Gross
IdentAware Security
82 Old Mountain Road
Lebanon, NJ 08833
USA
téléphone : (908) 268-1629
mél : gmgross@identaware.com

Déclaration complète de droits de reproduction

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