Groupe de travail Réseau
Request for Comments : 2093
Catégorie : Expérimentale
Traduction Claude Brière de L'Isle

H. Harney
C. Muckenhirn
SPARTA, Inc.
juillet 1997

Spécification du protocole de gestion de clés de groupe (GKMP)

Statut de ce mémoire

Le présent mémoire définit un protocole expérimental pour la communauté de l'Internet. Il ne spécifie en aucune façon une norme de l'Internet. On invite à des discussions et suggestions pour son amélioration. La distribution du présent mémoire n'est soumise à aucune restriction.

Résumé

La présente spécification propose un protocole pour créer des clés symétriques groupées et les distribuer entre des homologues communicants. Ce protocole présente les avantages suivants : 1) il est virtuellement invisible à l'opérateur, 2) aucun site central de distribution de clés n'est nécessaire, 3) seuls les membres du groupe ont la clé, 4) le fonctionnement est en mode envoyeur ou receveur, 5) on peut utiliser des protocoles de communications en diffusion groupée.

Table des matières

1 Fondements

La distribution de gestion de clés traditionnelle a imité le système militaire de comptabilité de clés fondée sur le papier. Les clés étaient distribuées, ordonnées, et comptabilisées physiquement, conduisant à de longues durées de préparation et à un fonctionnement coûteux.

Il existe des algorithmes coopératifs de gestion de clés qui permettent de générer des paires de clés entre deux équipements. Cela donne une structure de gestion de clés plus rapide et plus fiable, capable de prendre en charge un grand nombre de communications sûres. Malheureusement, seules des clés appariées sont aujourd'hui prises en charge par ces méthodes.

Le présent document décrit un protocole pour établir et changer les groupes de clés de chiffrement (plus de deux) sur l'Internet. On appelle cette approche le protocole de gestion de clés de groupe (GKMP, *Group Key Management Protocol*).

1.1 Généralités sur le protocole

GKMP crée des clés pour des groupes de chiffrement, distribue les clés aux membres du groupe, assure (via des contrôles d'homologue à homologue) un contrôle d'accès aux clés fondé sur une règle, refuse l'accès aux hôtes connus pour être compromis, et permet un contrôle hiérarchique des actions du groupe.

Le concept de génération de clés utilisé par le GKMP est une génération coopérative entre deux entités de protocole. Il y a plusieurs algorithmes de génération de clés viables dans le GKMP (c'est-à-dire, RSA, Diffie-Hellman, courbes elliptiques). Tous ces algorithmes utilisent des techniques de clés asymétriques pour passer les informations entre deux entités pour créer une seule clé de chiffrement.

Le protocole GKMP distribue alors les clés de groupe aux entités GKMP qualifiées. Ce processus de distribution est un processus de défiance mutuelle (toutes les actions et identités doivent être vérifiées).

GKMP fournit un processus de vérification d'homologue à homologue. Les entités du protocole passent des certificats de permission (PC, *permission certificate*) au titre du processus de distribution de clés de groupe. Les PC contiennent des informations de contrôle d'accès sur un site particulier. Ces informations de contrôle d'accès sont allouées par une autorité supérieure qui signe alors le PC. Donc, chaque entité peut vérifier les permissions de chaque autre entité GKMP mais ne peut en modifier aucune. Chaque entité de protocole vérifie les permissions et les compare au niveau de service demandé. Si les permissions ne sont pas égales ou supérieures à la demande, le service est refusé.
GKMP prend en charge la récupération des compromis. Une liste d'entités GKMP compromises est distribuée aux membres du groupe durant les actions de gestion de clés. Par essence, une liste de récupération de compromis (CRL, *Compromise Recovery List*) permet aux membres des groupes d'abandonner les connexions qui ont des entités compromises. Le GKMP délègue le contrôle des groupes à des contrôleurs de groupe spécifiques afin qu'il soit plus facile de répartir la CRL sur les

entités GKMP les plus importantes. Durant chaque action de gestion de clé, le numéro de version de CRL est transmis ; lorsque une mise à jour de CRL est détectée, elle est téléchargée et vérifiée (elle est signée par une autorité supérieure).

Le GKMP permet le contrôle des actions du groupe. Dans certains réseaux, il est souhaitable qu'une autorité supérieure contrôle strictement la génération des groupes. Ces réseaux ont généralement une autorité centrale de fonctionnement du réseau. Le GKMP permet à ces autorités d'ordonner à distance les actions du groupe. Ces ordres sont signés par cette autorité et vérifiés par toutes les entités impliquées avec le groupe.

Le GKMP est un protocole de couche application. Il est indépendant du protocole de communication sous-jacent. Cependant, si le service de diffusion groupée est disponible, il va accélérer le changement de clés des groupes cryptographiques. Donc, le GKMP utilise les services de diffusion groupée si ils sont disponibles.

2 Généralités : les rôles dans GKMP

La création et la distribution de clés groupées exige une répartition des rôles. Cela identifie quelles fonctions effectuent les hôtes individuels dans le protocole. Les deux rôles principaux sont ceux de distributeur de clés et de membre. Le contrôleur initie la création de la clé, forme les messages de distribution des clés, et collecte les accusés de réception des clés des receveurs. Les membres attendent un message de distribution, déchiffrent, valident, et accusent réception de la nouvelle clé.

2.1 Contrôleur de groupe

Le contrôleur de groupe (GC, *group controller*) est le membre du groupe qui a autorité pour effectuer les actions critiques du protocole (c'est-à-dire, création de clé, distribution de clé, création de messages de changement de clé de groupe, et rapport sur les progrès de ces actions). Tous les membres du groupe ont la capacité d'être GC et pourraient assumer cette tâche si ils sont désignés pour le faire.

Le GC aide le groupe cryptographique à atteindre et conserver la synchronisation de clés. Un groupe doit fonctionner sur la même clé de chiffrement symétrique. Si une partie du groupe perd sa clé ou la change de façon inappropriée, il ne sera pas capable d'envoyer ou recevoir des données à un autre hôte qui fonctionne sur la clé correcte. Donc, il est important que les opérations qui créent ou changent une clé soient sans ambiguïté et soient contrôlées (c'est-à-dire qu'il ne serait pas approprié que plusieurs hôtes essayent de changer simultanément les clés d'un réseau). Donc, quelqu'un doit en être chargé, et c'est le contrôleur.

2.2 Membre de groupe

Dit simplement, un membre de groupe est tout hôte d'un groupe qui n'agit pas comme contrôleur. Les membres du groupe vont assister le contrôleur pour créer la clé, valider l'autorisation du contrôleur d'effectuer des actions, accepter la clé provenant du contrôleur, demander la clé au contrôleur, entretenir les listes de CRL locales, effectuer les révisions d'homologue des actions de gestion de clés, et gérer la clé locale.

3. Primitives d'éléments de données

3.1 Liste de membres de groupe

Dans un groupe en mode envoyeur, le GC doit avoir reçu une liste des membres du réseau. Le contrôleur va alors initier le contact avec ces membres du réseau et créer le groupe.

3.2 Jeton de groupe

Le jeton de groupe est créé par l'autorité qui commande un groupe. Le jeton contient des informations dont les membres du réseau ont besoin pour s'assurer qu'un contrôleur est autorisé à créer un groupe et savoir exactement quelles contraintes il est prévu de faire peser sur le groupe. Le jeton de groupe contient les champs suivants :

- Identification de groupe
- Identifiant de GC
- Action de groupe (créer, changer, supprimer les clés)
- Permissions du groupe (règles pour guider le contrôle d'accès)
- Intervalle de changement de clé (durée de vie de la clé de groupe)
- Version de jeton (identifiant du jeton en cours)

- o Signature du jeton (signature asymétrique utilisant la clé privée des commandants de groupe)
- o Clé publique des commandants de groupe (cette clé publique est elle-même signée par le gestionnaire de la sécurité du réseau pour lier la clé publique à un identifiant de membre d'un réseau spécifique).

3.3 Identifiant de groupe

Le groupe doit être identifié de façon univoque pour permettre que plusieurs groupes différents coexistent sur un réseau.

3.4 Identifiant de GTEK

C'est l'identifiant univoque de la GTEK (il peut inclure des informations d'état).

3.5 Identifiant de GKEK

C'est l'identifiant univoque de la GKEK (il peut inclure des informations d'état).

3.6 Champ de création de GTEK

Dans un protocole de création de clé coopératif, chaque partie contribue à certains champs utilisés pour créer la clé.

3.7 Champ de création de GKEK

Dans un protocole de création de clé coopératif, chaque partie contribue à certains champs utilisés pour créer la clé.

3.8 Distributeur de signature

Signature asymétrique qui utilisé la clé privée des GC.

3.9 Distributeur de clé publique

Moitié publique de la paire de clés de signature des GC. (Cette clé publique est elle-même signée par le gestionnaire de la sécurité du réseau pour lier la clé publique à un identifiant de membre du réseau spécifique.)

3.10 Signature de membre

Signature asymétrique qui utilise la clé privée des membres choisis.

3.11 Clé publique de membre

Moitié publique de la paire de clé de signature des membres choisis. (Cette clé publique est elle-même signée par le gestionnaire de la sécurité du réseau pour lier la clé publique à un identifiant de membre spécifique du réseau.

3.12 Permissions de contrôleur

Les permissions de contrôleur sont allouées par le gestionnaire de la sécurité. La signature des gestionnaires de sécurité va lier les permissions à l'identifiant de contrôleur.

3.13 Identifiant de SKEK

Ce champ identifie exactement quelle SKEK est en train d'être créée. Cela permet que plusieurs groupes interopèrent simultanément sur un réseau.

3.14 Champ de création de SKEK

Ce champ contient les informations fournies en contribution pour être utilisées dans le processus de création de KEK.

3.15 Permissions de membre

Les permissions des membres sont allouées par le gestionnaire de la sécurité. La signature des gestionnaires de sécurité va lier les permissions à l'identifiant du contrôleur.

3.16 Clés de groupe chiffrées

Cet élément de données est chiffré dans la KEK (de session ou de groupe) créée pour le téléchargement des clés. Ce sont la GTEK et la GKEK créées pour un groupe. Une somme de contrôle est aussi chiffrée. Cela assure la confidentialité et l'intégrité des données de la GTEK et de la GKEK.

3.17 Confirmation de déchiffrement

C'est un court champ (un octet) qui indique le déchiffrement du message et exactement quel type de message a été déchiffré.

3.18 Demande

Un champ D0mande contient la demande spécifique qu'un membre d'un réseau peut faire à un autre. Les demandes couvrent l'adhésion au groupe, la mise à jour de la CRL, la génération de paire de TEK, la détection, la création de groupe, l'état.

3.18 Liste de suppression de membres

C'est une liste des membres du groupe qui sont supprimés administrativement du groupe.

4. Définitions de message

4.1 Commande Créer un Groupe

Ce message contient les primitives d'élément de données suivantes : Membres du groupe, Identifiant de Grp, Identifiant de contrôleur de Grp, Action de Grp, Permissions de Grp, Intervalle de changement de clé, Version de jeton, Signature de jeton, Clé publique de jeton. Ce message peut être confidentiel du fait du champ Permissions de groupe. Dans les systèmes sensibles, il sera nécessaire de le chiffrer avant la transmission.

4.2 Créer Grp Keys_1

Ce message passe les informations nécessaires pour créer les clés de groupe à partir du GC vers le membre choisi du réseau. Ce message contient : Identifiant de Grp, Demande, Identifiant de GTEK, Identifiant de GKEK, champ Création de GTEK, champ Création de GKEK, jeton de Grp, Signature du contrôleur, Clé publique du contrôleur.

4.3 Créer Grp Keys_2

Ce message passe les informations nécessaires pour créer les clés de groupe à partir du membre du réseau choisi au GC. Ce message contient : Identifiant de Grp, Identifiant de GTEK, Identifiant de GKEK, Champ Création de GTEK, Champ Création de GKEK, Signature du membre, Clé publique du membre.

4.4 Négocier Grp Keys_1

Ce message passe le jeton de groupe et les permissions de GC au membre choisi. Ces informations peuvent être sensibles et devoir être protégées. Donc, ce message est chiffré avec la GTEK juste créée. Ce chiffrement comporte les vérifications d'intégrité des données appropriées. Ce message1 contient : Identifiant de Grp, Identifiant de TEK, Identifiant de KEK, Jeton de groupe, Permissions de contrôleur.

4.5 Négocier Grp Keys_2

Ce message passe les permissions des membres choisis du réseau au GC. Ce message1 contient : Identifiant de Grp, Identifiant de GTEK, Identifiant de GKEK, Permissions de membre. Ces informations peuvent être sensibles et devoir être

protégées. Donc, ce message est chiffré avec la GTEK juste créée. Ce chiffrement comporte les vérifications appropriées d'intégrité des données.

4.6 Créer Session KEK_1

Ce message envoie des informations pour créer une KEK à utilisation unique entre le GC et le membre choisi du réseau.

4.7 Créer Session KEK_2

Ce message envoie des informations pour créer une KEK à usage unique entre le membre choisi du réseau et le GC.

4.8 Négocier Session Keys_1

Ce message passe l'identifiant de groupe, l'identifiant de SKEK, le numéro de version de la CRL, le jeton de groupe et les permissions des GC aux membres choisis du réseau. Ces informations peuvent être sensibles et doivent être protégées. Donc, ce message est chiffré. Si une paire de clés appropriée est disponible, elle devrait alors être utilisée. Sinon, la KEK juste créée pourrait être utilisée pour chiffrer le message.

4.9 Négocier Session Keys_2

Ce message identifie le groupe, la SKEK, le numéro de version de la CRL et les permissions du membre. Ces informations peuvent aussi être sensibles et doivent être protégées.

4.10 Télécharger les clés de groupe

Ce message comporte un identifiant de GRP et des éléments de données de clés de groupe chiffrées.

4.11 Accusé de réception de téléchargement de clé

Ce message contient l'identifiant de GRP et des éléments de données de Confirmation_déchiffrement. Il confirme la réception et la vérification du déchiffrement de la GTEK et de la GKEK.

4.12 Diffusion groupée de changement de clés

Ce message contient : Identifiant de groupe ID, Identifiant de GTEK, Identifiant de GKEK, jeton de groupe, et les permissions du contrôleur. Le message de changement de clés est chiffré dans la GKEK déjà résidente dans tous les sites de membres du groupe. Cela conduit à un seul message capable d'être accepté par tous les membres du groupe.

4.13 Demande_se_joindre_au_groupe

Ce message contient Demande, Identifiant de groupe, Signature de membre, Clé publique de membre.

4.14 Supprimer_Clés_de_Groupe

Ce message contient : Identifiant de groupe, Demande, Liste de suppression de membres, Signature du contrôleur, Clés publiques des contrôleurs.

4.15 Accusé de réception de suppression des clés de groupe

Ce message contient Identifiant de groupe, Identifiant de membre, Signature de membre, Clé publique de membre.

5 Définitions d'états

Il y a treize états distincts dans le protocole. Ils sont décrits ci-dessous.

5.1 État 1

L'adresse de source est vérifiée pour s'assurer qu'elle n'est pas sur la CRL.

Le champ jeton est validé avec la clé publique de la source.

Le numéro de version du jeton est vérifié pour s'assurer que ce jeton est actuel.

L'identifiant de groupe est vérifié pour vois si ce groupe existe.

Le champ Identifiant de contrôleur est ensuite lu. Si le receveur figure comme GC sur la liste, le receveur suppose qu'il joue le rôle de contrôleur. Sinon, le rôle supposé est celui de receveur.

Le GC lit le champ Permission de groupe dans le jeton de groupe. Il vérifie alors que ses permissions personnelles excèdent ou sont égales à celles du groupe.

Le GC va créer sa portion du message de création de clé.

Le message Créer Grp Keys_1 est achevé et transmis.

5.2 État 2

Le champ Signature de source est validé en utilisant la clé publique de la source.

Le champ Identifiant de source est comparé à la CRL locale. Si la source est sur la CRL, l'association est terminée.

Le champ Demande est lu. Les contributions locales aux clés de groupe sont créées.

Les clés de groupe sont créées et mémorisées en attendant la négociation.

Le tableau des clés est mis à jour pour montrer la clé de groupe en attendant la négociation.

5.3 État 3

Le certificat de permission est restitué et validé en utilisant la clé publique du gestionnaire de sécurité. Les permissions de la source du message sont vérifiées pour s'assurer qu'elles satisfont ou dépassent celles du groupe.

Le jeton de groupe est restitué et validé en utilisant la clé publique appropriée.

Le numéro de version du jeton est vérifié pour s'assurer que le jeton est actuel.

L'identifiant de groupe spécifié dans le jeton est comparé à l'identifiant de groupe réel. Si ils sont différents, l'échange est terminé.

L'identifiant de contrôleur spécifié dans le jeton est comparé à l'identifiant de GC. Si ils ne correspondent pas, l'échange est terminé.

Les permissions locales sont comparées aux permissions spécifiées pour le groupe. Si elles ne sont pas égales ou supérieures aux permissions du groupe, l'échange est terminé et un rapport est généré.

L'intervalle de changement de clés spécifié dans le jeton est mémorisé localement.

Le tableau des clés est mis à jour pour refléter les permissions de clés, l'intervalle de changement de clés, l'identifiant de groupe et l'heure courante.

5.4 État 4

Le certificat de permission est restitué et validé en utilisant la clé publique de sécurité des membres. Les permissions de la source du message sont vérifiées pour s'assurer qu'elle sont égales ou supérieures à celles du groupe.

Le tableau des clés est mis à jour pour refléter les permissions de clés, l'intervalle de changement de clés, l'identifiant de groupe et l'heure courante.

5.5 État 5

Le champ Signature de source est validé en utilisant la clé publique de la source.

Le champ Identifiant de source est comparé à la CRL locale. Si la source est sur la CRL, l'association est terminée.

Le champ Demande est lu. La contribution locale à la SKEK est créée. La SKEK est créée et mémorisée en attendant la négociation.

Le tableau des clés est mis à jour pour montrer la SKEK en cours de négociation.

5.6 État 6

Le certificat de permission est restitué et validé en utilisant la clé publique des gestionnaires de sécurité. Les permissions de la source du message sont vérifiées pour s'assurer qu'elles sont égales ou supérieures à celles du groupe.

Le jeton de groupe est restitué et validé en utilisant la clé publique appropriée.

Le numéro de version du jeton est vérifié pour s'assurer que le jeton est actuel.

L'identifiant de groupe spécifié dans le jeton est mémorisé.

L'identifiant de contrôleur spécifié dans le jeton est comparé à l'identifiant de GC. Si ils ne correspondent pas, l'échange est terminé.

Les permissions locales sont comparées aux permissions spécifiées pour le groupe. Si elles ne sont pas égales ou supérieures aux permissions du groupe, l'échange est terminé et un rapport est généré.
L'intervalle de changement de clés spécifié dans le jeton est mémorisé localement.

Le tableau des clés est mis à jour pour refléter les permissions de clés, l'intervalle de changement de clés, l'identifiant de groupe et l'heure courante.

5.7 État 7

Le certificat de permission est restitué et validé en utilisant la clé publique des gestionnaires de sécurité. Les permissions de la source du message sont vérifiées pour s'assurer qu'elles sont égales ou supérieures à celles du groupe.

Le tableau des clés est mis à jour.

5.8 État 8

L'Identifiant de groupe est vérifié.

Les clés de groupe sont déchiffrées en utilisant la SKEK. Les vérifications d'intégrité des données sont validées pour s'assurer d'un déchiffrement approprié.

Le tableau des clés est mis à jour pour refléter les nouvelles clés de groupe, les permissions de clés, l'intervalle de changement de clés, l'identifiant de groupe et l'heure actuelle.

5.9 État 9

Mise à jour du journal de gestion de groupe.

5.10 État 10

Le certificat de permission est restitué et validé en utilisant la clé publique des gestionnaires de sécurité. Les permissions de la source du message sont vérifiées pour s'assurer qu'elles sont égales ou supérieures à celles du groupe.

Le jeton du groupe est restitué et validé en utilisant la clé publique appropriée.

Le numéro de version du jeton est vérifié pour s'assurer que le jeton est actuel.

L'identifiant de groupe spécifié dans le jeton est vérifié.

L'identifiant de contrôleur spécifié dans le jeton est comparé à l'identifiant de GC. Si ils ne correspondent pas, l'échange est terminé.

Les permissions locales sont comparées aux permissions spécifiées pour le groupe. Si elles ne sont pas égales ou supérieures aux permissions du groupe, l'échange est terminé et un rapport est généré.

L'intervalle de changement de clés spécifié dans le jeton est mémorisé localement.

Les nouvelles clés de groupe sont déchiffrées avec la GKEK en cours. Le champ Intégrité des données est vérifié pour s'assurer que le déchiffrement est approprié.

Le tableau des clés est mis à jour pour refléter les permissions de clés, l'intervalle de changement de clés, l'identifiant de groupe et l'heure courante.

5.11 État 11

Valider la signature en utilisant la clé publique de la source.

Vérifier que l'adhésion au groupe à l'initiative des membres est disponible. Sinon, l'ignorer. Dans ce cas, commencer la distribution des clés de groupe.

5.12 État 12

Valider la signature en utilisant la clé publique des GC.

Restituer la liste des suppressions. Vérifier pour voir si elle est sur la liste des suppressions, si il en est ainsi, continuer.

Créer un message Acc_Suppression_Clés_de_Grp.

Supprimer les clés de groupe.

5.13 État 13

Valider la signature en utilisant la clé publique des GC.

Restituer la liste des suppressions. Si la liste est une suppression globale, vérifier les clés de remplacement.

Passer les opérations du groupe sur les clés de remplacement.

Créer un message Acc_Suppression_Clés_de_Grp.

Supprimer les clés.

6. Définitions fonctionnelles – Protocole de gestion de clés de groupe

GKMP consiste en plusieurs fonctions nécessaires pour créer, distribuer, changer les clés et gérer les groupes de clés symétriques. Ces fonctions sont :

- o Création de groupe (groupe initié par l'envoyeur)
  - -- Créer les clés de groupe
  - -- Distribuer les clés de groupe
- o Changement des clés de groupe
  - -- Créer les clés de groupe
  - -- Changer les clés de groupe
- o Adhésion initiée par le membre
- o Suppression de membre de groupe

Les paragraphes qui suivent vont décrire chaque fonction, y compris les primitives des données et la construction des messages. Les diagrammes associés représentent les spécificités (séquence, localisation, sources et destinations de communications) des messages et traitements nécessaires.

6.1 Création de groupe

L'initialisation de membre est une fonction en trois étapes qui implique de commander la création du groupe, la création des clés du groupe, puis la distribution de ces clés aux "autres" membres du groupe. Les messages entre le GC et le premier membre génèrent deux clés pour les futures actions du groupe : la clé de chiffrement du trafic du groupe (GTEK, *Group Traffic Encryption Key*) et la clé de chiffrement de clé de groupe (GKEK, *Group Key Encryption Key*). Les messages entre le GC et les autres membres servent aux besoins de la distribution des clés. Ces fonctions sont décrites dans les paragraphes qui suivent.

6.1.1 Commande de groupe

La toute première action est pour qu'une entité commande le groupe. Cette commande est envoyée au GC.

6.1.2 Créer les clés de groupe

Le premier membre doit coopérer avec le GC pour créer les futures clés de groupe. S'appuyer sur deux hôtes distincts pour créer les clés de groupe maximise la probabilité que la clé résultante ait les propriétés cryptographiques appropriées. Un seul hôte pourrait créer la clé si la fonction d'aléation est robuste et de confiance. Malheureusement cela exige habituellement des matériels spécialisés qui ne sont pas disponibles sur la plupart des sites. L'intention du présent protocole était d'utiliser des matériels génériques pour améliorer l'extensibilité du GKMP. Donc, on utilise des mécanismes coopératifs de génération de clé.

Pour faciliter une création de groupe bien ordonnée, les informations de gestion doivent être échangées entre le contrôleur et les membres du groupe. Ces informations identifient de façon univoque l'identité du GC, ses permissions, l'autorisation de créer les clés, les futures permissions du groupe, l'état actuel de la liste des compromis, et les informations de gestion relevant des clés en cours de création. Toutes ces informations sont protégées contre la falsification par des technologies de signature asymétrique. La clé publique utilisée pour vérifier les paramètres dont la portée s'étend sur l'ensemble du réseau (par exemple, les permissions des hôtes individuels) est largement détenue. La clé publique pour vérifier les informations générées en local, comme l'identité de l'homologue, est envoyée avec les messages. Ceci allège les exigences de mémorisation de clé publique des hôtes.

Les objectifs du processus de création de clés sont :
- o de générer en coopération une GTEK et une GKEK,
- o de permettre aux créateurs de clés de vérifier l'identité du partenaire de création de la clé en vérifiant les signatures des messages,
- o de partager les clés publiques,
- o de permettre la validation du GC, en signant l'identification de groupe, l'identification du GC, et les permissions du groupe,
- o d'envoyer au premier membre l'identité du groupe, l'identité du GC, les identités de membre du groupe, les permissions du groupe, et l'intervalle de changement de clés du groupe, signés par le commandant de groupe (lorsque le groupe a été commandé à distance).

Cette fonction consiste en quatre messages entre le GC et le premier membre. Les messages initiaux sont pour

l'établissement de la GTEK et de la GKEK. Cela est fait par le GC qui envoie un message Créer_Group_Keys_1 signé au premier membre. Ce message contient deux valeurs aléatoires nécessaires pour générer la GTEK et la GKEK. Ce message contient aussi la clé publique du GC.

Le premier membre validé le message Créer_Group_Keys_1 signé, construit et envoie un message Créer_Group_Keys_2 signé au GC. Il génère la GTEK et la GKEK, et mémorise la clé publique reçue. Le message Créer_Group_Keys_2 contient les valeurs aléatoires nécessaires pour que le GC génère la GTEK et la GKEK. Ce message contient aussi la clé publique du premier membre.

Le GC valide le message Créer_Group_Keys_2 signé, génère la GTEK et la GKEK, construit le message Négocier_Group_Keys_1 pour sa transmission au premier membre, et mémorise la clé publique reçue.

Le GC envoie le message Négocier_Group_Keys_1 au premier membre, chiffré avec la GTEK qui vient d'être générée.

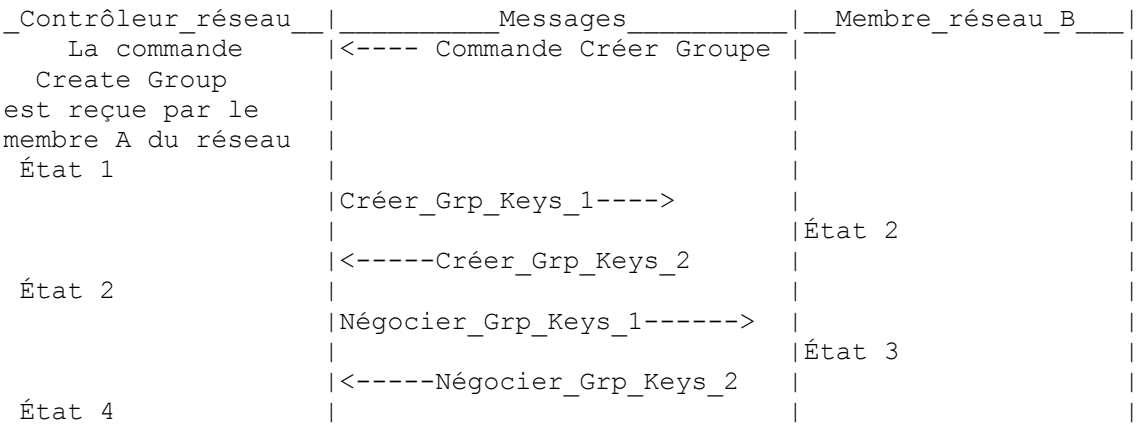

Figure 1 : Diagramme d'état : Créer_Group_Keys

Le premier membre déchiffre le message Négocier_Group_Keys_1 et extrait l'identification du groupe, l'identification du GC, les membres du groupe, les permissions du groupe, l'intervalle de changement de clés, le numéro de version de CRL, et la signature de l'autorité de certification. Les champs Identification du groupe, Identification du GC, et Permissions du groupe sont validés sur la base de la signature des commandants de groupe extraits (si c'est un groupe commandé à distance, cette signature identifie l'hôte distant). Si ces champs sont validés, les structures internes des premiers membres sont mises à jour.

6.1.3 Distribution des clés de groupe aux autres membres

Les autres membres du groupe doivent obtenir les clés de groupe avant que celui-ci soit pleinement opérationnel. L'objet de l'initialisation des autres membres du groupe est le suivant :

- o générer de façon coopérative une clé de chiffrement de clé de session (SKEK) pour la transmission de la GTEK et de la GKEK à partir du GC,
- o permettre à chaque membre de vérifier l'identité du contrôleur et vice versa,
- o permettre à chaque membre de vérifier l'autorisation du contrôleur de créer le groupe,
- o envoyer le paquet de clé (KP, *key packet*) (consistant en la GTEK, GKEK), l'identité de groupe, l'identité du GC, les identités des membres du groupe, les permissions du groupe, et l'intervalle de changement des clés du groupe aux autres membres.

Cette fonction consiste en six messages entre le GC et les autres membres. Les messages initiaux sont pour l'établissement d'une SKEK. Cela se fait par l'envoi par le GC d'un message Créer_Session_KEK_1 signé aux autres membres. Ce message contient la valeur aléatoire nécessaire pour que l'autre membre génère la SKEK. Ce message contient aussi la clé publique du GC.

L'autre membre valide le message Créer_Session_KEK_1, construit et envoie un message Créer_Session_KEK_2 au GC, génère la SKEK, et mémorise la clé publique reçue. Le message Créer_Session_KEK_2 contient la valeur aléatoire nécessaire pour que le GC génère la SKEK. Ce message contient aussi la clé publique de l'autre membre.

Le GC valide le message Créer_Session_KEK_2, génère la SKEK, construit le message Négocier_Session_ KEK_1 pour le transmettre à l'autre membre, et mémorise la clé publique reçue.

Le GC envoie à l'autre membre le message Négocier_Session_KEK_1 chiffré avec la SKEK qui vient d'être générée. Le

message Négocier_Session_KEK_1 comporte l'identifiant de groupe, le jeton de groupe, les permissions du contrôleur, et le numéro de version de la CRL.

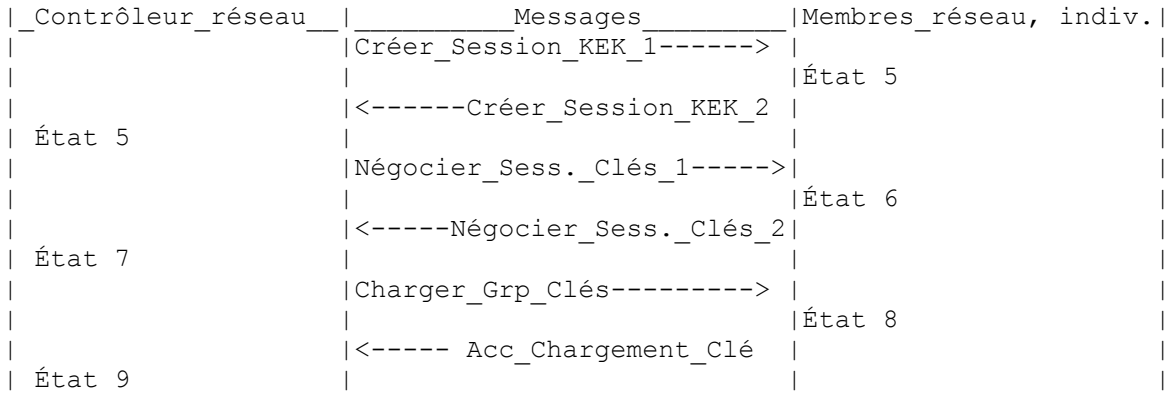

Figure 2 : Diagramme d'état : Distribution des clés

L'autre membre déchiffre le message Négocier_Session_KEK_1, vérifie l'autorité et l'identification du contrôleur,s'assure que la CRL locale est à jour, et construit un message Négocier_Session_KEK_2 pour le transmettre au GC.

Le GC reçoit le message Négocier_Session_KEK_2 et construit un message Charger_Grp_Clés pour le transmettre à l'autre membre.

Le GC envoie à l'autre membre le message Charger_Grp_Clés chiffré avec la SKEK qui vient juste d'être générée. (Noter que la clé utilisée pour chiffrer les messages de négociation peut être combinée différemment pour créer la KEK.)

Les autres membres déchiffrent le message Charger_Grp_Clés et extraient le KP, l'identification de groupe, l'identification du GC, les membres du groupe, les permissions du groupe, l'intervalle de changement de clés, et la signature des commandants de groupe. Les champs Identification de groupe, Identification du GC, et Permissions du groupe sont validés sur la base de la signature. Si ces champs sont valides, les autres tableaux de mémorisation de clés internes des membres sont mis à jour avec les nouvelles clés.

6.2 Changement des clés de groupe

Le changement de clés est une fonction en deux étapes qui implique un échange de messages entre le GC et un "premier membre" et les "autres membres". Les messages entre le GC et le premier membre sont exactement comme décrit pour la création de groupe. Les messages entre le GC et les autres membres sont destinés à la distribution de la nouvelle GTEK et de la nouvelle GKEK. Ces fonctions dont décrites dans les paragraphes qui suivent.

6.2.1 Création des clés de groupe

La fonction de premier membre pour une opération de changement de clés est la même que pour l'initialisation des clés. Prière de se référer au paragraphe 2.1 sur la création de groupe intitulé "Création de clés de groupe".

6.2.2 Changement de clés

L'objet du changement de clés est le suivant :

- o envoyer la nouvelle GTEK et la nouvelle GKEK aux autres membres,
- o permettre à chaque membre de vérifier l'identité du contrôleur,
- o permettre à chaque membre de vérifier l'autorisation du contrôleur de changer les clés du groupe, l'identification du groupe, et l'identification du GC,
- o envoyer l'identité du groupe, l'identité du GC, les identités des membres du groupe, les permissions du groupe, et l'intervalle de changement de clés du groupe aux autres membres,

Les messages pour créer et négocier les clés de groupe sont les mêmes que ceux de la création de groupe. C'est la raison pour laquelle ils ne sont pas reproduits ici.

La portion changement de clés de cette fonction consiste en un message entre le GC et les autres membres. Le GC construit un message DiffGrP_Chgt_Clés signé à transmettre aux autres membres. Comme son nom l'implique, ce message peut être en diffusion groupée à l'ensemble du groupe. Le GC envoie aux autres membres le message DiffGrP_Chgt_Clés signé chiffré avec la GKEK en cours.

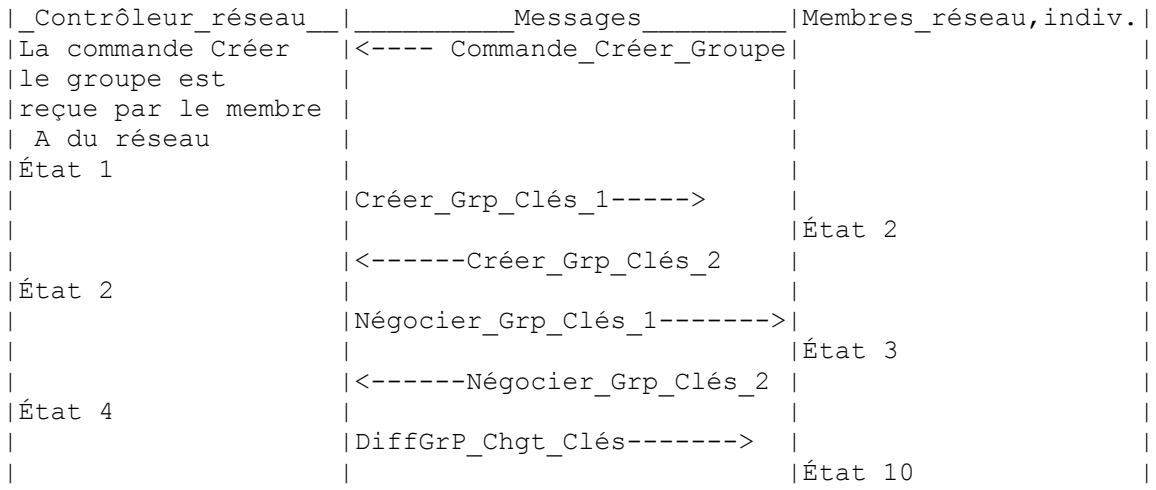

Figure 3 : Diagramme d'état : Changement de clés

Les autres membres déchiffrent et valident le message DiffGrP_Chgt_Clés signé et extraient le nouveau KP, l'identification de groupe, l'identification de GC, les membres du groupe, les permissions du groupe, l'intervalle de changement de clés, et la signature de commande de changement de clés. Les champs Identification de groupe, Identification de GC, et Permissions du groupe sont validés sur la base de la signature de commande de changement de clés extraite. Si ces champs sont validés, les tableaux de la base de données des clés sont mis à jour.

6.3 Adhésion à l'initiative du membre

GKMP prend en charge l'adhésion au groupe à l'initiative du membre. Ce type de service est très intéressant lorsque l'initiateur du groupe n'a pas besoin de contrôler les adhésions autrement qu'en vérifiant que tous les membres du groupe se conforment à des règles préalablement acceptées.

Un exemple de ce type de groupe est celui des vacances d'emploi d'une entreprise. Une entreprise veut garder ses vacances d'emploi confidentielles et peut décider de chiffrer les annonces. Le créateur du groupe ne se soucie pas de savoir qui reçoit les annonces du moment qu'il est dans l'entreprise. Lorsque un employé essaye d'accéder aux informations, le GC regarde les permissions d'employés (signées par une autorité supérieure). Si elles indiquent que l'employé fait partie de l'entreprise, le contrôleur permet l'accès au groupe.

Avant qu'un membre potentiel du groupe puisse se joindre aux opérations du groupe, il doit demander la clé au GC, s'identifier sans ambiguïté, passer ses permissions, et recevoir les clés. Cela exige plusieurs messages passés entre le GC et l'adhérent postulant. L'objet de ces messages est le suivant :

- o demander l'adhésion au groupe au contrôleur,
- o générer de façon coopérative une SKEK pour la transmission du chiffrement du trafic de groupe et la GKEK de la part du GC,
- o permettre à chaque membre de vérifier l'identité du contrôleur et vice versa,
- o permettre à chaque membre de vérifier l'autorisation des contrôleurs de créer le groupe,
- o envoyer le KP, l'identité de groupe, l'identité du GC, les identités des membres du groupe, les permissions du groupe, et l'intervalle de changement de clés aux autres membres,

Les séries de messages pour une adhésion à l'initiative d'un membre sont très similaires aux séries de messages pour distribuer les clés de groupe durant la création de groupe. En fait, les séries sont identiques excepté l'ajout d'un message de demande d'adhésion envoyée du membre adhérent au contrôleur lorsque l'adhésion est à l'initiative du membre. Ce message ne devrait pas exiger le chiffrement car il ne contient probablement pas d'informations sensibles. Cependant, dans certains systèmes militaires, le fait qu'un membre veuille se joindre à un groupe peut être une information sensible du point de vue de l'analyse du trafic. Dans ces instances spécialisées, une paire de TEK peut être créée, s'il n'en existe déjà une, pour cacher la demande de service.

Cette fonction consiste en sept messages entre le GC et le membre adhérent. Le premier message est créé par le membre adhérent et envoyé au GC. Il demande simplement l'adhésion au groupe au contrôleur. Le contrôleur prend la décision de répondre à la demande sur la base des paramètres du groupe – limites d'adhésion, listes de membres.

Les messages suivants sont pour l'établissement d'une SKEK. Cela se fait par l'envoi par le GC aux autres membres d'un

message Créer_Session_KEK_1 signé. Ce message contient la valeur aléatoire nécessaire pour que les autres membres génèrent la SKEK. Ce message contient aussi la clé publique du GC.

L'autre membre valide le message Créer_Session_KEK_1, construit et envoie un message Créer_Session_KEK_2 au GC, génère la SKEK, et mémorise la clé publique reçue. Le message Créer_Session_KEK_2 contient la valeur aléatoire nécessaire pour que le GC génère la SKEK. Ce message contient aussi la clé publique de l'autre membre.

Le GC validé le message Créer_Session_KEK_2, génère la SKEK, construit le message Négocier_Session_ KEK_1 pour le transmettre aux autres membres, et mémorise la clé publique reçue.

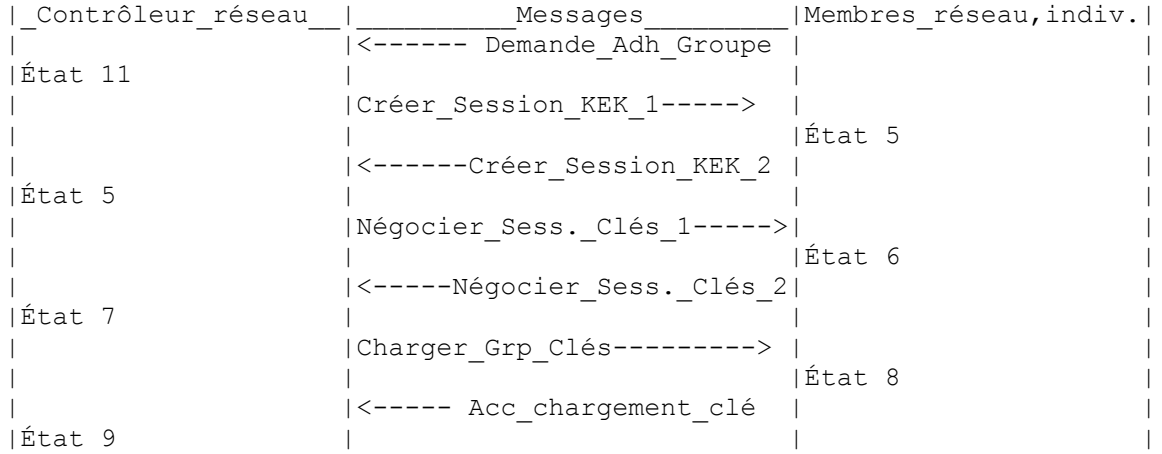

Figure 4 : Diagramme d'état : Adhésion de membre

Le GC envoie à l'autre membre le message Négocier_Session_KEK_1 chiffré avec la SKEK qui vient d'être générée.

L'autre membre déchiffre le message Négocier_Session_KEK_1 et construit un message Négocier_Session_KEK_2 à transmettre au GC.

Le GC reçoit le message Negocier_Session_KEK_2 et construit un message Charger_Grp_Clés à transmettre à l'autre membre.

Le GC envoie à l'autre membre le message Download_Grp_Keys chiffré avec la SKEK qui vient d'être générée. (noter que la clé utilisée pour chiffrer les messages de négociation peuvent être combinés de façon différente pour créer la KEK.)

Les autres membres déchiffrent le message Charger_Grp_Clés et extraient le KP, l'identification de groupe, l'identification de GC, les membres du groupe, les permissions du groupe, l'intervalle de changement de clés, et la signature des commandants de groupe. Les champs Identification de groupe, Identification du GC, et Permissions du groupe sont validés sur la base de la signature. Si ces champs sont validés, les tableaux internes de mémorisation de clé des autres membres sont mis à jour avec les nouvelles clés.

6.4 Suppression de membre

Il y a deux types de scénarios de suppression de membre – coopératif et hostile. Le scénario de suppression coopératif est le retrait d'un membre de groupe de confiance pour une raison liée à la gestion (c'est-à-dire, pour réduire la taille du groupe, préparer le membre pour un déplacement). La suppression hostile résulte habituellement en une perte de l'état sécurisé sur le site du membre (c'est-à-dire, compromission, panne d'équipement).

Les deux scénarios représentent des enjeux différents pour le réseau. La minimisation de l'impact sur le réseau est de la plus haute importance dans le scénario coopératif. On veut laisser intact le groupe de clés et pouvoir être sûr que la suppression coopérative du membre du groupe n'aura pas d'impact sur la sécurité future des opérations du groupe. Dans le cas d'une suppression hostile, le but est de revenir à un état de fonctionnement sûr aussi vite que possible. En fait, c'est un compromis. On peut éliminer le groupe compromis aussitôt que la compromission est découverte, mais cela peut handicaper une base importante. De sorte que les soucis de sécurité doivent être mis en balance avec les préoccupations du fonctionnement.

6.4.1 Suppression coopérative

La fonction de suppression coopérative survient entre un membre de confiance et le GC. Il en résulte une suppression fiable du chiffrement de la clé de groupe et des GTEK chez le membre supprimé. Cette suppression est destinée à être une fonction administrative.

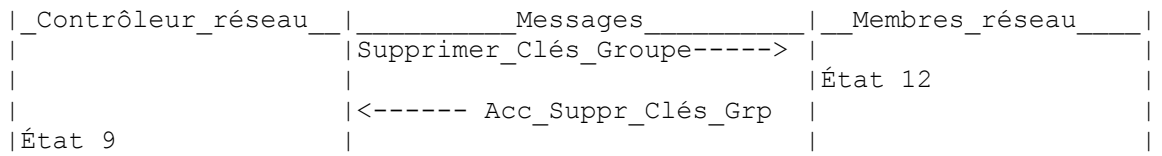

Figure 5 : Diagramme d'état : Suppression coopérative

Cette fonction consiste en deux messages entre le GC et le membre. Le GC envoie le message Supprimer_Clés_Groupe au groupe, chiffré avec la GTEK. Le message identifie le ou les membres qui ont besoin de supprimer les clés de groupe. Le ou les membres déchiffrent le message Supprimer_Clés_Groupe, extraient l'identification de groupe, vérifient la liste des membres supprimés, suppriment les clés de trafic de groupe et de chiffrement de clé pour ce groupe, et construisent le message Acc_Suppr_Clés_Grp pour le transmettre au GC.

Le message Acc_Suppr_Clés_Grp est chiffré avec la clé de trafic de groupe. Le GC reçoit le message Acc_Suppr_Clés_Grp, le déchiffre, et met à jour la définition du groupe.

6.4.2 Suppression hostile (sur compromission)

La suppression hostile survient lorsque un groupe perd la confiance envers un membre. On suppose que toutes les clés qui résident sur le site du membre ont été perdues. On suppose aussi que le membre ne va pas coopérer. Donc, on doit essentiellement créer un autre groupe, moins le membre suspect, et transférer les opérations de groupe à ce nouveau groupe. Lorsque le contrôleur perd la confiance du groupe, un autre contrôleur doit être désigné et le processus de suppression hostile peut ensuite se poursuivre.

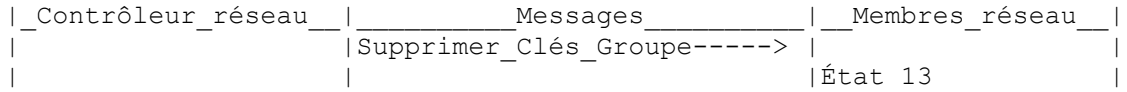

Figure 6 : Diagramme d'état : Suppression hostile

Quelques question de gestion de la sécurité et du fonctionnement se posent autour de la récupération de compromission. Ces questions concernent essentiellement le compromis entre continuité du fonctionnement et vulnérabilité de la sécurité. Si un membre se trouve être mauvais, du point de vue de la sécurité, tout le trafic devrait s'arrêter sur le réseau. Cependant, si ce trafic porte une opération critique, il se peut que le groupe préfère vivre avec la faiblesse de la sécurité plutôt que d'interrompre la communication du groupe.

GKMP fournit deux mécanismes pour aider à interdire l'accès aux membres compromis. D'abord, il met en œuvre une liste de révocation de certificat (CRL, *Certificate Revocation List*) qui est vérifiée durant le processus de création du groupe. Ainsi, elle ne permettra pas qu'un membre compromis soit inclus dans un nouveau groupe. Ensuite, GKMP facilite la création d'un autre groupe (moins le ou les membres compromis). Cependant, il n'impose pas la solution à la question de savoir si le groupe peut continuer de fonctionner avec un membre compromis.

Le mécanisme qu'utilise GKMP pour retirer un membre compromis est de le laisser dehors. Cela implique de créer un nouveau groupe, sans le membre compromis, et d'y passer les opérations du groupe. Le vieux groupe est annulé par plusieurs diffusions groupées d'un message de suppression de groupe.

Cette fonction consiste en un message du GC à tous les membres. Le GC envoie le message Supprimer_Groupe à tous les membres, chiffré avec la GTEK. Il en résulte la suppression des clés de trafic de groupe et de chiffrement de clés chez tous les membres du groupe. Tous les membres déchiffrent le message Supprimer_Groupe reçu, valident l'autorisation, extraient l'identification de groupe, et suppriment les clés de trafic de groupe et de chiffrement de clés.

7. Considérations sur la sécurité

Le présent document, est entièrement consacré aux problèmes de sécurité.

8. Adresse des auteurs

Hugh Harney
SPARTA, Inc.
Secure Systems Engineering Division
9861 Broken Land Parkway, Suite 300
Columbia, MD 21046-1170
USA
téléphone : +1 410 381 9400 (ext. 203)
mél : hh@columbia.sparta.com

Carl Muckenhirn
SPARTA, Inc.
Secure Systems Engineering Division
9861 Broken Land Parkway, Suite 300
Columbia, MD 21046-1170
USA
téléphone : +1 410 381 9400 (ext. 208)
mél : cfm@columbia.sparta.com