Sécurité des réseaux
Authentification avec Kerberos

A. Guermouche

Plan

Plan

Présentation

Kerberos est un service d'authentification développé par comme partie du projet Athena du MIT.

Problèmatique :

- Comment assurer l'authentification pour les demandes de services dans un environnement distribué ouvert?
- Problème de confiance en les mécanismes d'authentification de chacun des postes de travail.
  - Un utilisateur peut obtenir l'accès à un poste de travail et feindre d'être un autre utilisateur.
  - Un utilisateur peut changer l'adresse réseau d'un poste de travail pour que les demandes envoyées depuis ce dernier semblent venir d'un autre poste de travail.
  - Un utilisateur peut espionner des échanges et lancer une attaque par rejeu pour perturber des opérations.
- Kerberos fournit un serveur centralisé d'authentification dont la fonction est d'authentifier les utilisateurs vis-à-vis des serveurs et inversement.

Exigences

Le premier rapport publié sur Kerberos énumère les exigences suivantes :

sûr. un espion sur le réseau ne doit pas pouvoir obtenir l'information nécessaire pour se faire passer pour un utilisateur.

fiable. Kerberos doit pouvoir reposer sur une architecture de serveurs distribuée avec des systèmes pouvant en remplacer d'autres.

transparent. idéalement, l'utilisateur ne doit pas être conscient du processus d'authentification.

évolutif. le système doit être capable de gérer un grand nombre de clients et de serveurs.

Le schéma complet de Kerberos est celui d'un service d'authentification par un tiers de confiance utilisant un protocole s'inspirant de celui de Needham Schroeder.

Rappel sur le protocole de Needham Schroeder

Soient A et B deux systèmes communiquant via un réseau pas sûr devant déterminer une clé de session.

Notations :

- S. est un serveur de confiance pour A et B.
- K_{AS}. clé symétrique connue uniquement par A et S.
- K_{BS}. clé symétrique connue uniquement par B et S.
- N_A et N_B. des nonces.

Protocole :

$$A \rightarrow S : A, B, N_A$$
$$S \rightarrow A : \{N_A, K_{AB}, B, \{K_{AB}, A\}_{K_{BS}}\}_{K_{AS}}$$
$$A \rightarrow B : \{K_{AB}, A\}_{K_{BS}}$$
$$B \rightarrow A : \{N_B\}_{K_{AB}}$$
$$A \rightarrow B : \{N_B - 1\}_{K_{AB}}$$

Protocole d'authentification de Kerberos (1/2)

- ⋆ Utilisation d'un serveur d'authentification (SA) qui connait tous les mots de passe.
- ⋆ Le serveur partage une clé secrète unique avec chaque serveur.

Lorsqu'un utilisateur demande un service à un serveur :

1. Le module client du poste de travail de l'utilisateur envoie le mot de passe saisi au SA dans un message contenant l'ID de l'utilisateur, son mot de passe et l'ID du serveur.
2. Le SA authentifie l'utilisateur.
3. Le SA génère un ticket contenant l'ID de l'utilisateur et l'ID du serveur et le chiffre avec la clé secrète partagée par le SA et le serveur.
4. Le ticket est envoyé au poste client.

Remarque :

- ⋆ Le ticket contient l'ID du serveur pour qu'il puisse vérifier qu'il a correctement déchiffrer le ticket,
- ⋆ l'ID du client pour dire que le ticket a été émis par le module client C
- ⋆ et l'adresse réseau du poste client (ou le nom) pour éviter l'usurpation d'identité en passant par une autre machine.

Protocole d'authentification de Kerberos (2/2)

- ⋆ Réduire le nombre de fois où l'utilisateur doit saisir son mot de passe.
- ⋆ Le mot de passe du client transitait en clair sur le réseau.

Solution : Utilisation d'un serveur d'octroi de ticket (TGS).

1. Le client demande un ticket d'octroi de tickets de la part de l'utilisateur au SA.
2. Le SA répond par un ticket chiffré avec une clé dérivée du mot de passe de l'utilisateur. Quand cette réponse arrive, le client demande son mot de passe à l'utilisateur, génère la clé et essaye de déchiffrer le ticket.
3. Le client demande un ticket d'octroi de service de la part de l'utilisateur au TGS (le message correspondant contient l'ID de l'utilisateur , l'ID du service, et le ticket d'octroi de tickets).
4. Le TGS déchiffre le ticket et vérifie le succès du déchiffrement par la présence de son ID. Si tout est bon, il génère un ticket pour accorder l'accès au service.
5. Le client demande l'accès à un service de la part de l'utilisateur (transmission d'un message contenant l'ID de l'utilisateur et le ticket d'octroi de service).

Remarque :

- ⋆ Inclusion d'un horodatage dans le ticket d'octroi de tickets pour éviter le rejeu.
- ⋆ Le ticket de service est chiffré avec une clé connue uniquement du TGS et du serveur (Le ticket d'octroi est lui chiffré par une clé connue par le SA et le TGS).

Protocole Kerberos

Problème du protocole précédent :

- Problème de durée de vie du ticket d'octroi de ticket et octroi de service (ainsi que l'authentification de la personne utilisant le ticket vis à vis du ticket).
- Authentification des serveurs vis à vis des clients.

Solution :

- Le SA fournit à la fois au TGS et au client une information secrète de façon sûre (utilisation d'une clé de session).
  - La distribution de la clé se fait par le biais d'un message, du SA vers le client, contenant la clé de session ainsi que le ticket et chiffré avec une clé dérivée du mot de passe.
  - Cette même clé est incluse dans le ticket (qui ne peut être lu que par le TGS).

Royaumes Kerberos

Un “royaume” Kerberos est l’environnement contenant un serveur et un ensembles de clients. Le royaume nécessite :

- Le serveur Kerberos doit connaître l’ID utilisateur (UID) et le mot de passe haché de tous les utilisateurs.
- Le serveur Kerberos doit partager une clé secrète avec chaque serveur. Tous les serveurs sont enregistrés sur le serveur Kerberos.

Le royaume est une notion centrale dans la configuration des différentes versions de serveurs Kerberos (MIT, Heimdal, . . .).