Sécurité Informatique

Gestion et distribution des clés & Protocoles d'authentification

October 17, 2018

Houcemeddine HERMASSI

houcemeddine.hermassi@enit.rnu.tn

École Nationale d'Ingénieurs de Carthage ENI-CARTHAGE
Université Carthage
Tunisie

Plan de cour

Distribution des clés secrètes dans les alg symétriques
- Problèmatique
- Distribution des clés
- Distribution des clés et Protocoles d'authentification
- Distribution des clés :Needham-Schroeder
- Distribution des clés : protocole orienté connexion

Distribution des clés publiques ds les alg asymétriques
- Distribution des clés publiques
- Distribution par Annonce publique
- Distribution par autorité de clé publique
- Distribution de clé par Certificats

Protocoles d'authentification
- Kerberos

Distribution des clés secrètes dans les algs symétriques

Problème

Problèmatique

- Le problème de génération et distribution des clés est un problème majeure dans les communications sécurisées.
- La sécurité des protocoles et des alg de cryptage est basé sur ce problème fondamental
- La gestion des différents clés des différents entités est aussi un problème majeur
- Les alg symétriques requiert que les deux interloculteurs partagent la même clé secrète
- les alg asymétriques requiert que les interlocuteurs possédent des clés publiques valides de leurs correspondants

Alternatives

Alice et Bob ont beaucoup d'alternatives pour distribuer une clé

- Alice peut sélectionner une clé et la délivre physiquement à Bob (main à main)
- Une tierce partie peut sélectionner et délivrer la clé à Alice et Bob
- Si Alice et bob ont communiqué auparavant, ils peuvent utiliser l'ancienne clé pour chiffrer la nouvelle
- Si Alice et Bob ont des lignes sécurisés avec une tierce partie Charlie, alors Charlie peut relayer la clé entre Alice et bob.

Hiérarchie des clés

Généralement deux types de clés

- **Clé de session**
  - clé temporaire
  - utilisée por chiffrer les données entre deux interlocuteurs
  - utilisée pour une seule session puis rejeté
- **Clé principale** :
  - utilisée pour chiffrer les clés de sessions
  - partagée par les utilisateurs et un centre de distribution de clé (KDC)

Protocoles d'authentification

- utilisés pour convaincre les entités de leurs identités et pour échanger des clés de session
- Peut être dans un seul sens ou mutuel (ds les deux sens)
- Les protocoles d'authentification permettre de garantir :
  - **Confidentialité** : pour protéger les clés de session
  - **Timeliness (tiens compte du timing)** : pour empêcher les attaques replay (rejeu)

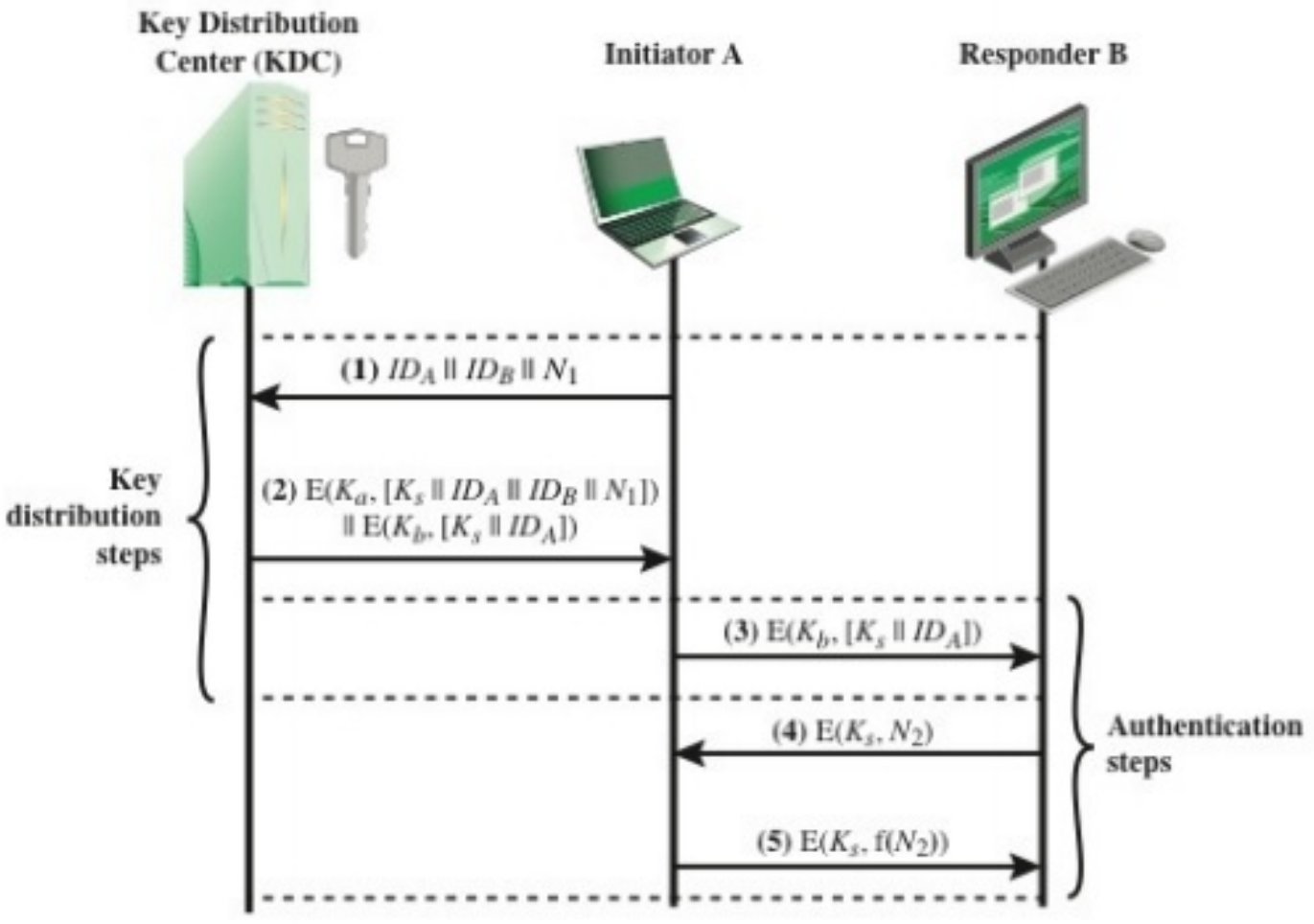
Authentification dans un seul sens

- ce type d'authentification est requis lorsque émetteur et récepteur ne sont pas en communication en même temps (ex : e-mail)
- l'entête de ce type de protocole doit être clair (non chiffré) pour être délivré sans problème par un système d'email
- le contenu du corps d'email peut être chiffré
- l'émetteur doit être authentifié

Authentification par cryptographie symétrique

- peut être faite moyennant **un centre de distribution de clé (KDC)**
- chaque entité partage sa clé principale avec le KDC
- le KDC génère les clés de session utilisés pour la connexion entre les différents entités
- les clés principales sont utilisées pour distribuer les clés de session

Scénario de distribution de clé : Needham-Schroeder



Protocole de Needham-Schroeder

- distribution de clés par une tierce partie
- pour une session entre deux entités **A** et **B** orchestré par un KDC
- le protocole est comme suit :
  1. $A \rightarrow KDC : ID_A \parallel ID_B \parallel N_1$
  2. $KDC \rightarrow A: E(K_a, [K_s \parallel ID_B \parallel N_1 \parallel E(K_b, [K_s \parallel ID_A])])$
  3. $A \rightarrow B : E(K_b, [K_s \parallel ID_A])$
  4. $B \rightarrow A : E(K_s, [N_2])$
  5. $A \rightarrow B : E(K_s, [f(N_2)])$

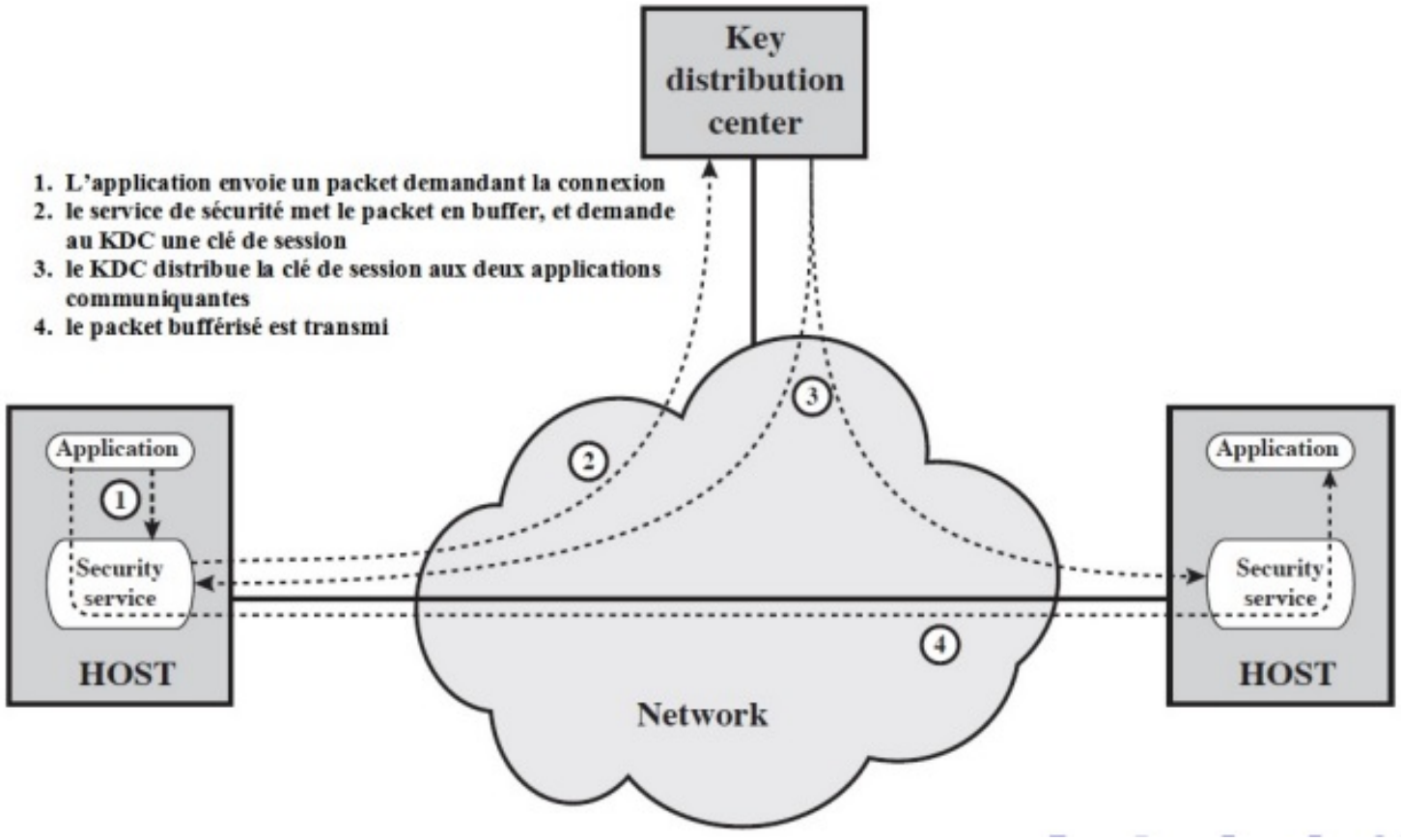
Replay attack sur Needham-Schroeder

- Le protocole est vulnérable à une replay-attack : le message de l'étape 3 peut être retransmit convainquant B qu'il est en communication avec A
- solution pour résoudre ce problème :
  - ajouter des timestamps dans l'étape 2 et 3
  - ajouter un nombre aléatoire à usage unique externe pour chaque échange de clés K_s

Distribution des clés secrètes dans les algorithmes symétriques

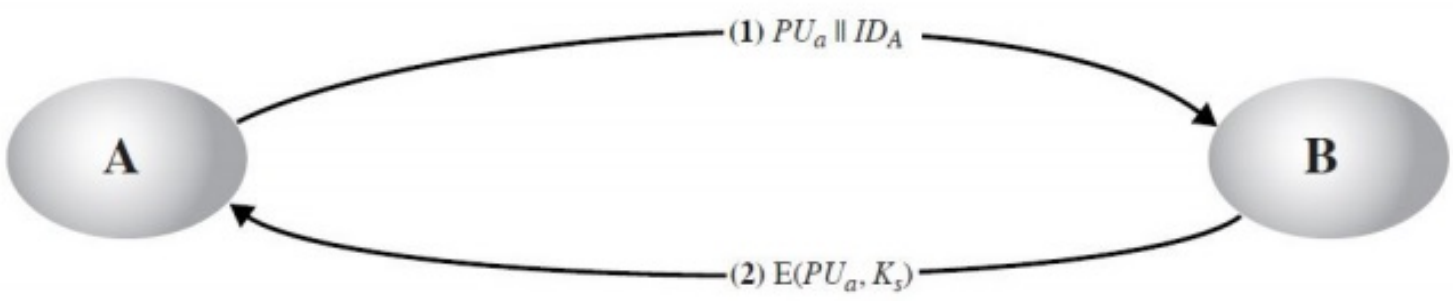
Distribution des clés

Distribution de clé automatique dans un protocole orienté connexion



Distribution d'une clé à l'aide d'une alg asymétrique : une simple distribution de clé

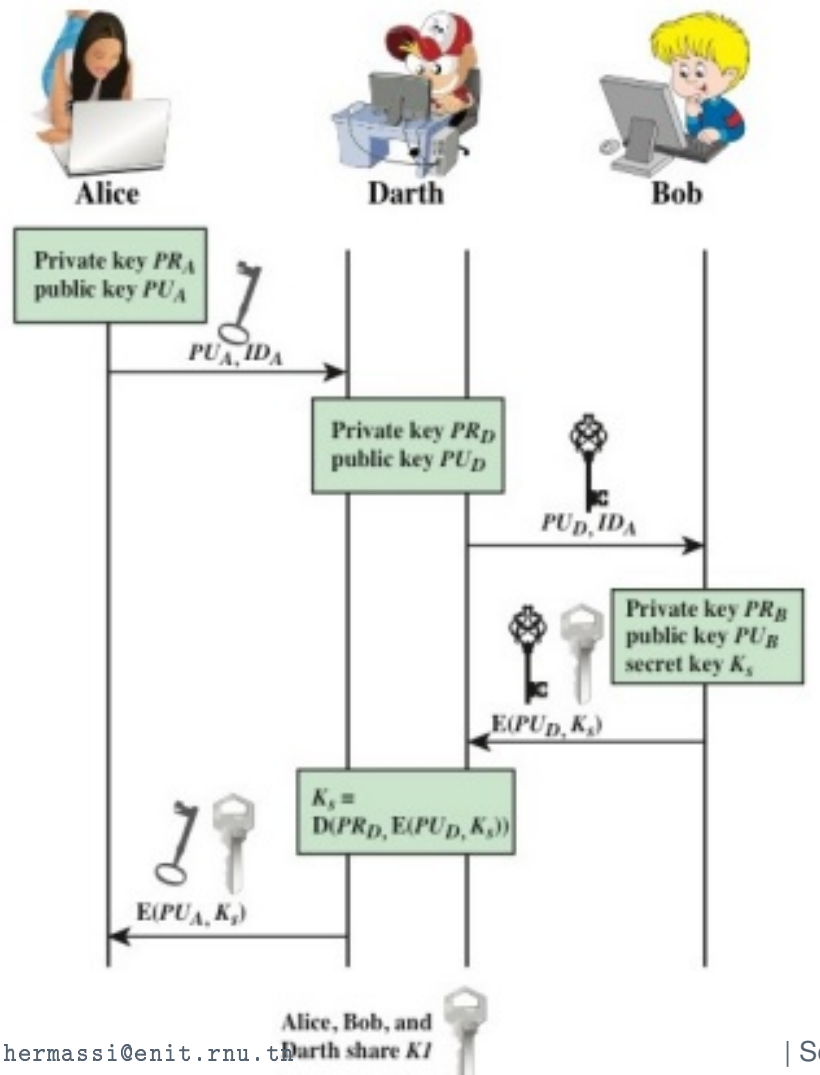
Merkle a proposé le protocole suivant pour distribuer une clé :



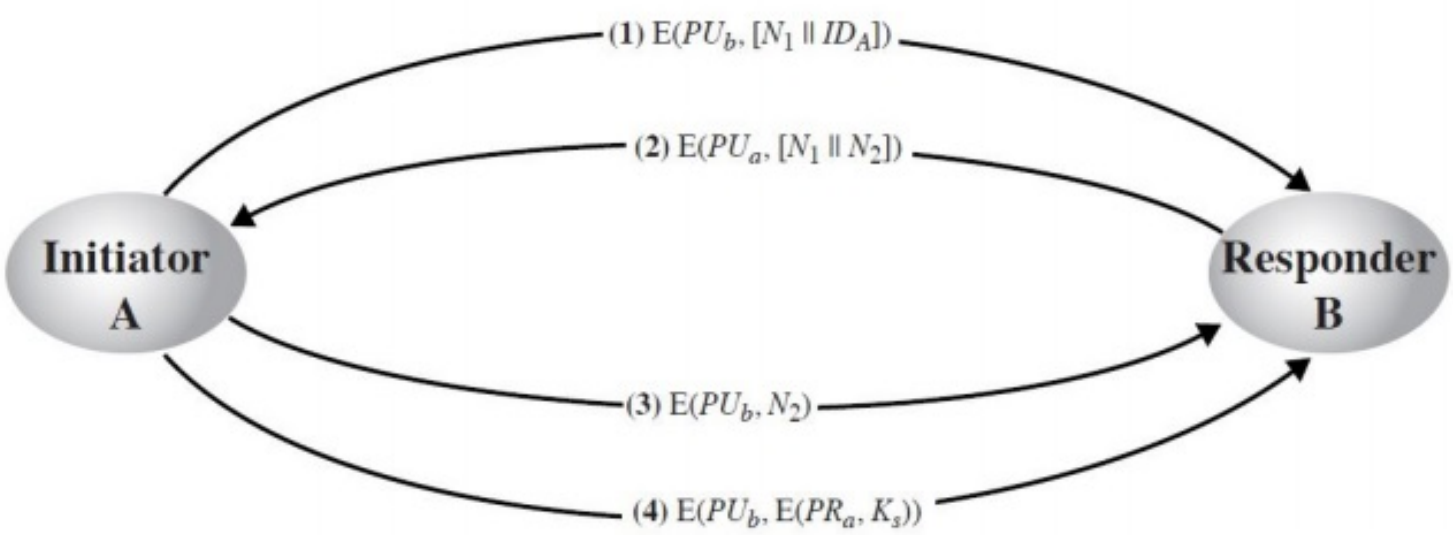
Distribution des clés secrètes dans les algorithmes symétriques

Distribution des clés

Man-in-the-middle-attack sur le protocole de Merkle



Distribution d'une clé à l'aide d'une alg asymétrique : confidentialité et authentification



Distribution des clés publiques

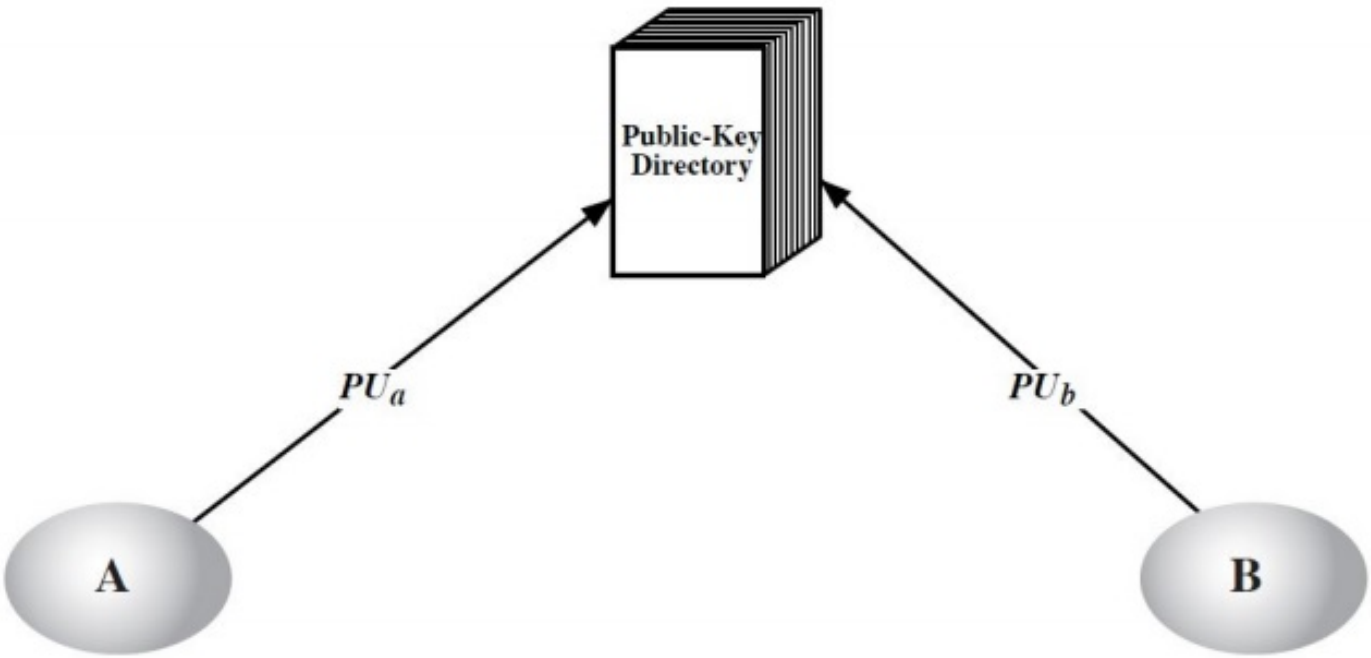
Peut se faire par :

- **annonce publique**
- à travers **une archive (répertoire)** disponible publiquement
- à travers **Une autorité de clé publique**
- **Certificats de clés publiques**

Distribution par Annonce publique

- Les utilisateurs distribuent leurs clés publiques aux bénéficiaires (intéressés) ou par diffusion à la communauté
  - attacher les clés publiques de PGP aux e-mails, ou les envoyer aux nouveaux groupes ou diffusion aux mails du carnet d'adresses
- Principal problème de cette méthode est **la modification = contrefaçon**
  - n'importe qui peut créer une clé prétendant être quelqu'un d'autre et la diffuser
  - jusqu'à ce que la falsification soit découverte, l'adversaire peut communiquer comme si c'était l'utilisateur légitime

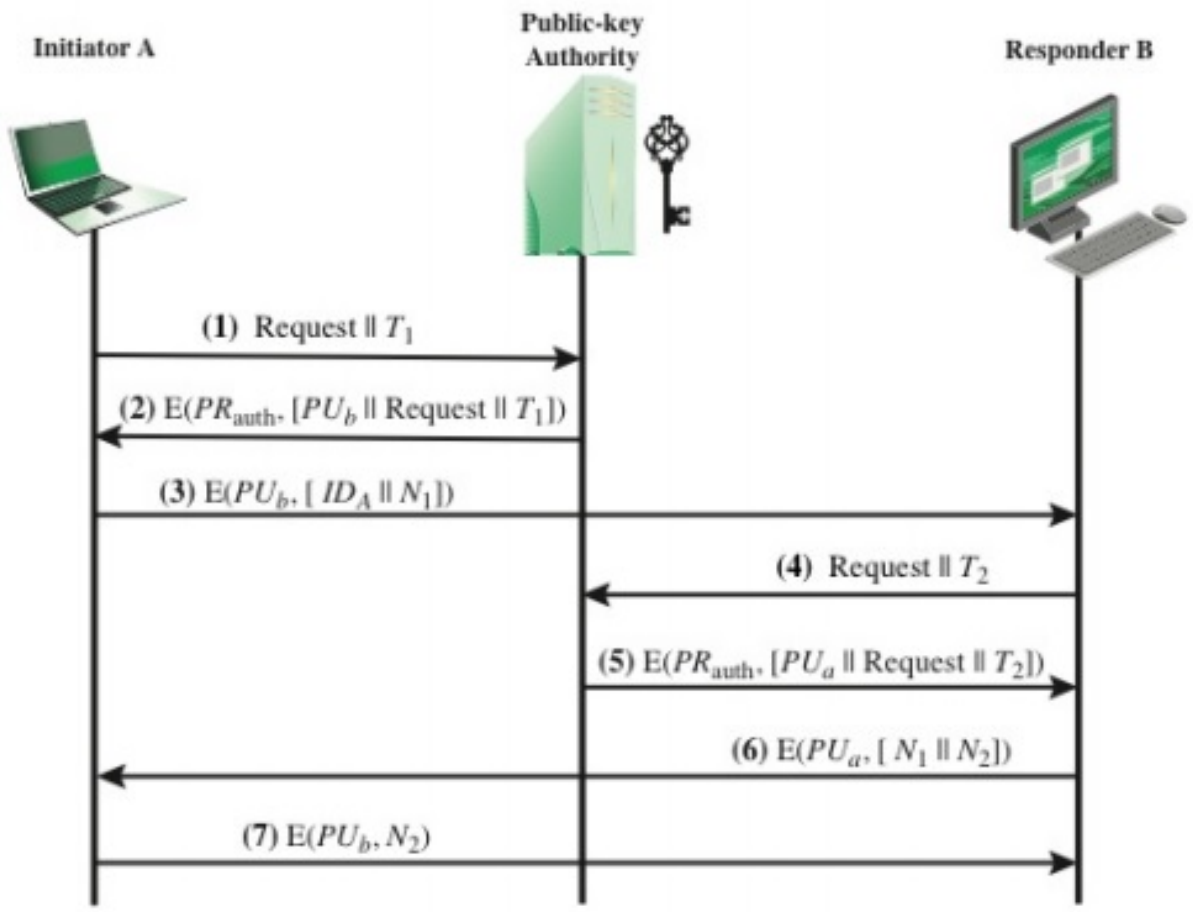
Distribution des clés par répertoire disponible publiquement



Distribution des clés publiques ds les algs asymétriques

Distribution des clés publiques

Distribution de clé par autorité de clé publique



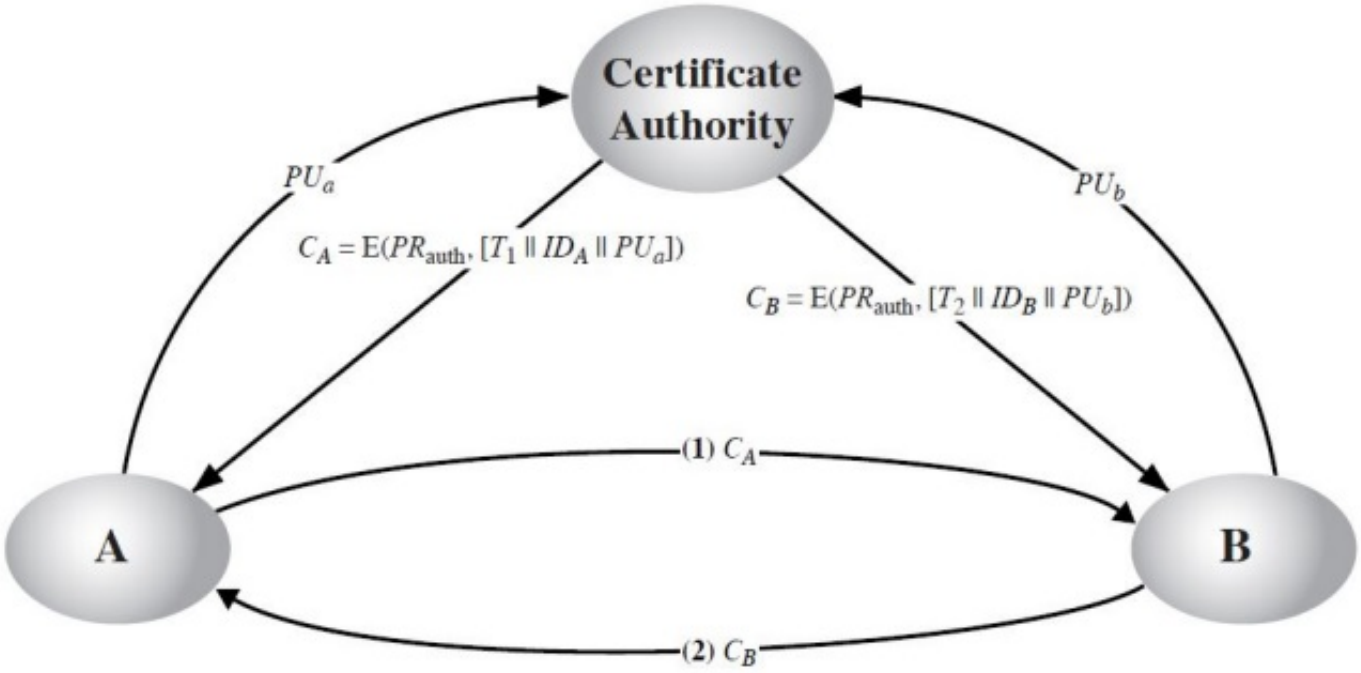
Distribution de clé par Certificats

- Les certificats permettent l'échange de clés publiques sans acces en temps réel à une autorité de clé publique
- Le certificat fait le lien entre **l'identité** et **la clé publique**
  - généralement avec d'autre info tel que période de validité, droits d'utilisation
- le contenu de la certificat est **signé** par une une autorité de clé publique ou une autorité de certification (AC)
- n'importe quel utilisateur connaissant la clé publique de l'AC peut vérifier la signature

Distribution des clés publiques ds les algs asymétriques

Distribution des clés publiques

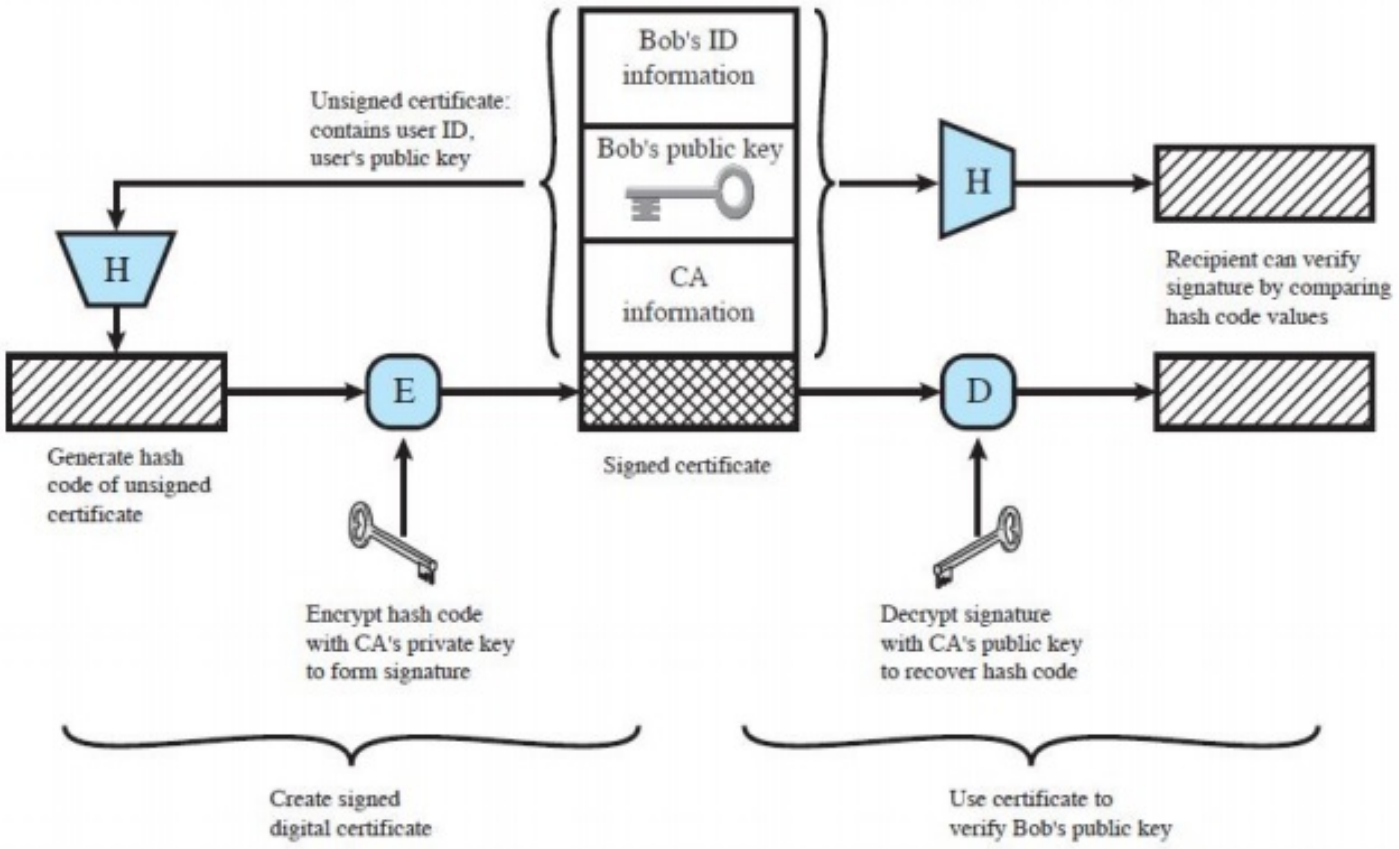
Création et échange des certificats



Distribution des clés publiques ds les algo asymétriques

Distribution des clés publiques

Utilisation du certificat



Distribution des clés publiques ds les algs asymétriques

Distribution de clé par Certificats

Certificat X.509

- X.509 est une partie de la serie de recommandations X.500 qui deéfinit un annuaire de services.
  - l'annuaire est un serveur ou un ensemble de serveurs qui maintient une base de donnée sur les utilisateurs
- X.509 définit une méthode de travail pour la fourniture de services d'authentification par l'annuaire X.500 à ses utilisateurs
  - paru en 1988 et révisé dernièrement en 2000.
  - basé sur l'utilisation de la cryptographie asymétrique et les signatures numériques
  - n'impose pas d'utiliser un alg spécifique mais recommande l'utilisation de RSA
  - n'impose pas une fonction de hachage spécifique
- chaque certificat contient la clé publique de l'utilisateur et signé par la clé privée de l'AC
- X.509 propose des protocoles d'authentifications basé sur l'utilisation des certificats numériques

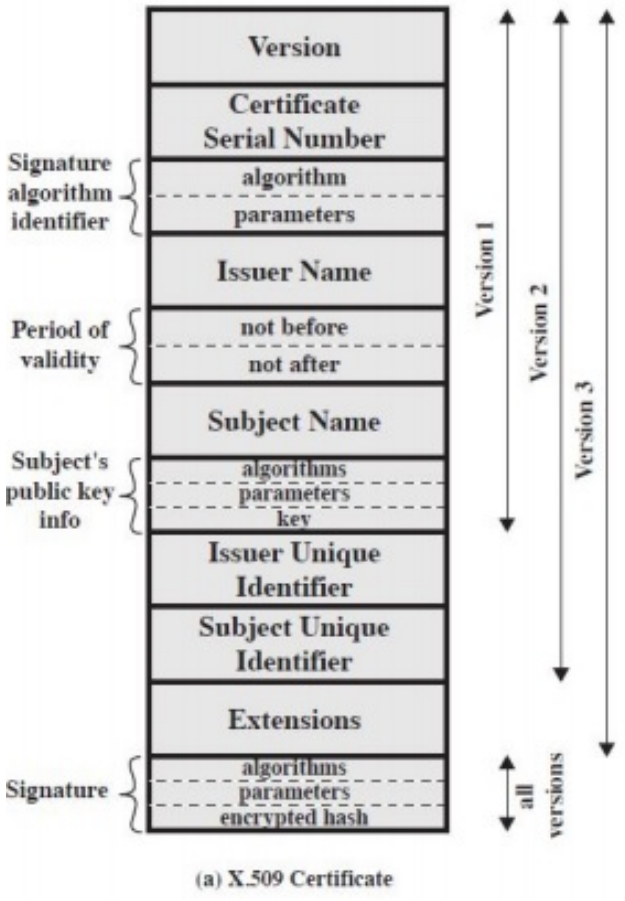
Certificat X.509

- le certificat est généré par une authorité de Certification (CA), et contient :
  - version V (1, 2, ou 3)
  - numéro de série SN (unique dans une CA) identifiant le certificat
  - l'identifiant de l'algorithme de signature (AI)
  - l'CA créateur du certificat)
  - periode de validité TA (de - à dates)
  - Le nom du sujet (nom du propriétaire du certificat)
  - les info concernant la clé publique du sujet(algorithme, paramètres, clé publique)
  - l'identifiant unique du créateur de la signature (v2+)
  - l'identifiant unique du sujet (v2+)
  - champs d'extension (v3)
  - signature (du hash de tous les champs dans le certificat)
- la notation CA«A» indique que le certificat de A est signé par CA

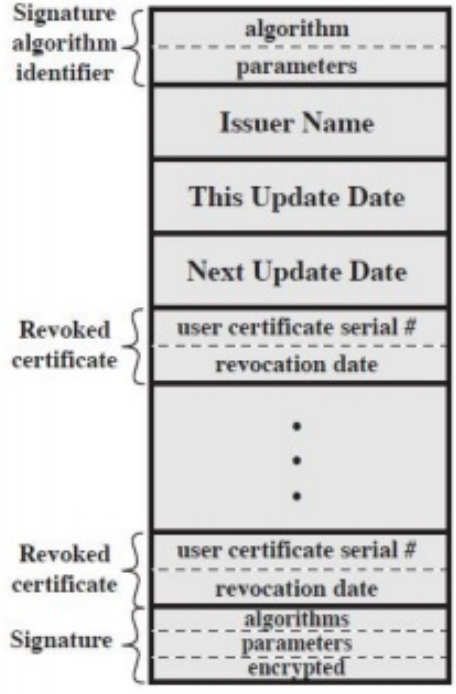
Distribution des clés publiques ds les asymétriques

Distribution de clé par Certificats

Format X.509



(a) X.509 Certificate



(b) Certificate Revocation List

Distribution des clés publiques ds les asymétriques

Distribution de clé par Certificats

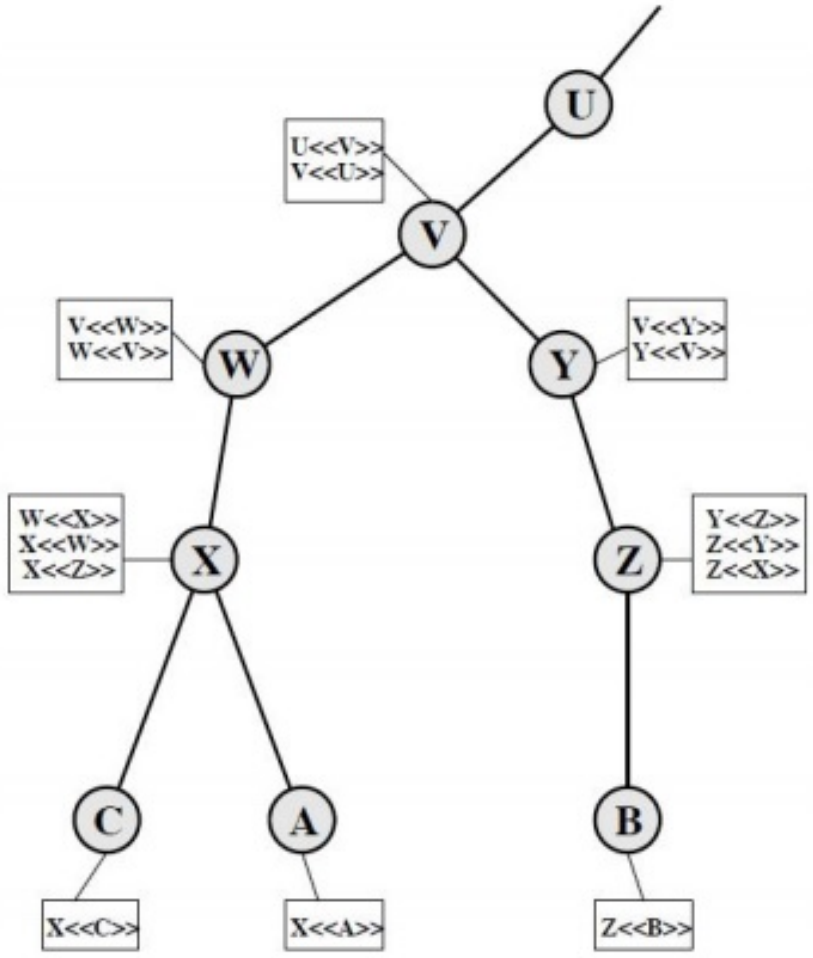
La hiérarchie des autorités de certification

- Si les deux utilisateurs partage la même CA, alors ils connaissent tous les deux sa clé publique
- sinon les autorité de certifications forment une héarchie
- utiliser des certificats liant les membres de la hiérarchie pour valider les autres CA
- chaque CA a des certificats pour ses clients et ses parents
- chaque client fait confiance à ses parents
- la hiérarchie permette la vérification de n'importe quel certificat d'un CA par les utilisateurs des autres CA dans la héarchie

Distribution des clés publiques ds les asymétriques

Distribution de clé par Certificats

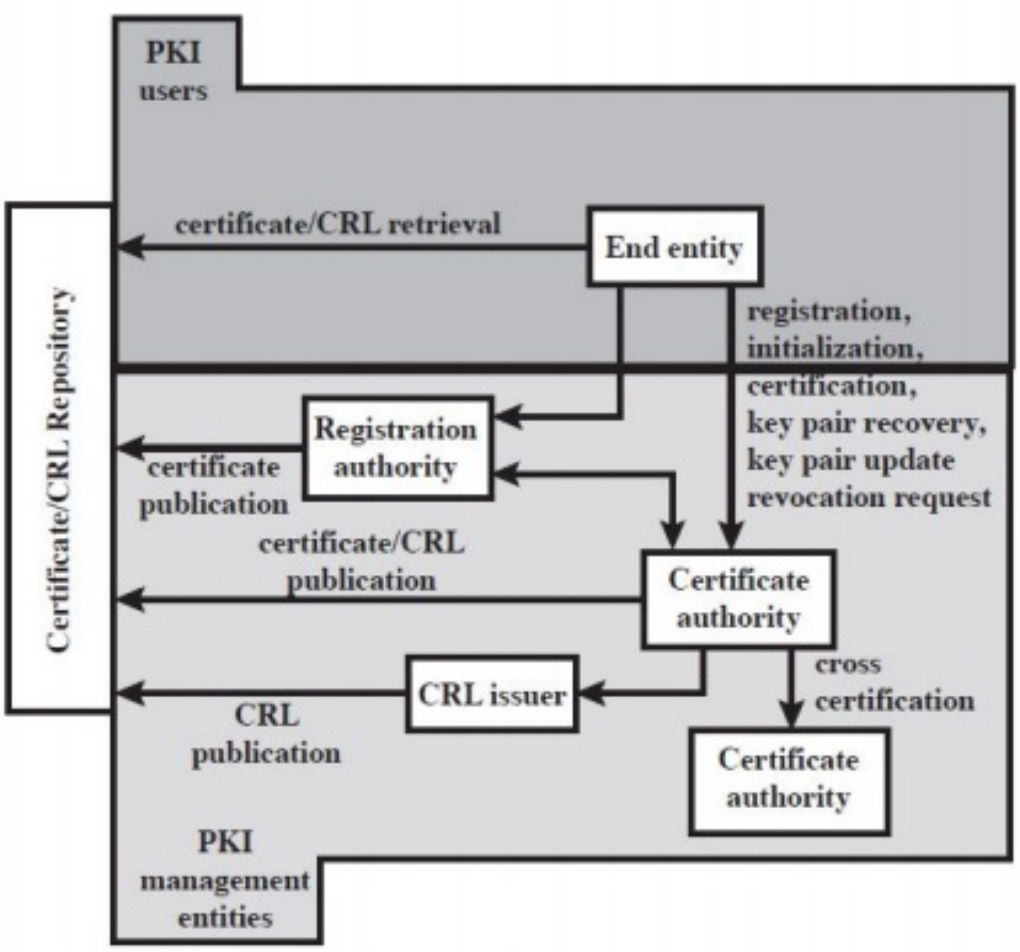
La hiérarchie des autorités de certification



Distribution des clés publiques ds les algs asymétriques

PKI : Public Key Infrastructure

PKI : Public Key Infrastructure



gestion des certificats par PKI

Les fonctions de PKI :

- registration
- initialization
- certification
- récupération de la paire de clé
- mise à jour de la paire de clé
- demande de révocation
- solution pour certification croisées
- protocoles : CMP, CMC

Protocoles d'authentification
Kerberos

Principe

- Service d'authentification dévéloppé au MIT
- Pourquoi Kerberos ?
  - Un utilisateur peut accéder à un poste de travail particulier et faire semblant d'être un autre utilisateur opérant à partir de ce poste.
  - Un utilisateur peut modifier l'adresse de réseau d'une station de travail de sorte que les requêtes envoyées par le poste de travail modifié semblent provenir de la station de travail usurpée
  - Un utilisateur peut espionner les échanges et utiliser une attaque replay pour gagner l'entrée à un serveur ou à perturber les opérations
- Kerberos fournit un service d'authentification centralisé dont la fonction est d'authentifier les utilisateurs aux serveurs et les serveurs aux utilisateurs
- Kerberos est basé totalement sur la cryptographie symétrique. La cryptographie à clé publique n'est pas utilisée
- il y a deux version de Kerberos : 4 et 5

Kerberos V4

- Utilise DES pour l'authentification
- le serveur d'authentification (AS) :
  - connaît les mots de passe de tous les utilisateurs et les enregistre dans une base de donnée centralisé
  - partage une clé secrète avec chaque serveur
- Ticket :
  - un ticket est crée une fois l'AS authentifie l'utilisateur. Le ticket contient l'ID de l'utilisateur, l'adresse réseau, et l'Id serveur
  - le ticket est chiffré par la clé secrète partagé par l'AS et le serveur
- TGS : Ticket-granting Service
  - crée et distribue les tickets aux utilisateurs qui ont été authentifié par l'AS
  - à chaque fois que l'utilisateur requiert l'accès à un nouveau service, le client demande cette connexion au TGS en utilisant le ticket pour s'authentifier au près du TGS
  - Le TGS accorde alors un ticket pour le service particulier
  - Le client enregistre les tickets de chaque service et les utilise pour s'authentifier à un serveur à chaque fois un service particulier est demandé

Protocoles d'authentification
Kerberos

L'échange de messages par Kerberos V4

(1) C → AS $ID_c \| ID_{tgs} \| TS_1$

(2) AS → C $E(K_c, [K_{c,tgs} \| ID_{tgs} \| TS_2 \| Lifetime_2 \| Ticket_{tgs}])$

$Ticket_{tgs} = E(K_{tgs}, [K_{c,tgs} \| ID_C \| AD_C \| ID_{tgs} \| TS_2 \| Lifetime_2])$

(a) Authentication Service Exchange to obtain ticket-granting ticket

(3) C → TGS $ID_v \| Ticket_{tgs} \| Authenticator_c$

(4) TGS → C $E(K_{c,tgs}, [K_{c,v} \| ID_v \| TS_4 \| Ticket_v])$

$Ticket_{tgs} = E(K_{tgs}, [K_{c,tgs} \| ID_C \| AD_C \| ID_{tgs} \| TS_2 \| Lifetime_2])$

$Ticket_v = E(K_v, [K_{c,v} \| ID_C \| AD_C \| ID_v \| TS_4 \| Lifetime_4])$

$Authenticator_c = E(K_{c,tgs}, [ID_C \| AD_C \| TS_3])$

(b) Ticket-Granting Service Exchange to obtain service-granting ticket

(5) C → V $Ticket_v \| Authenticator_c$

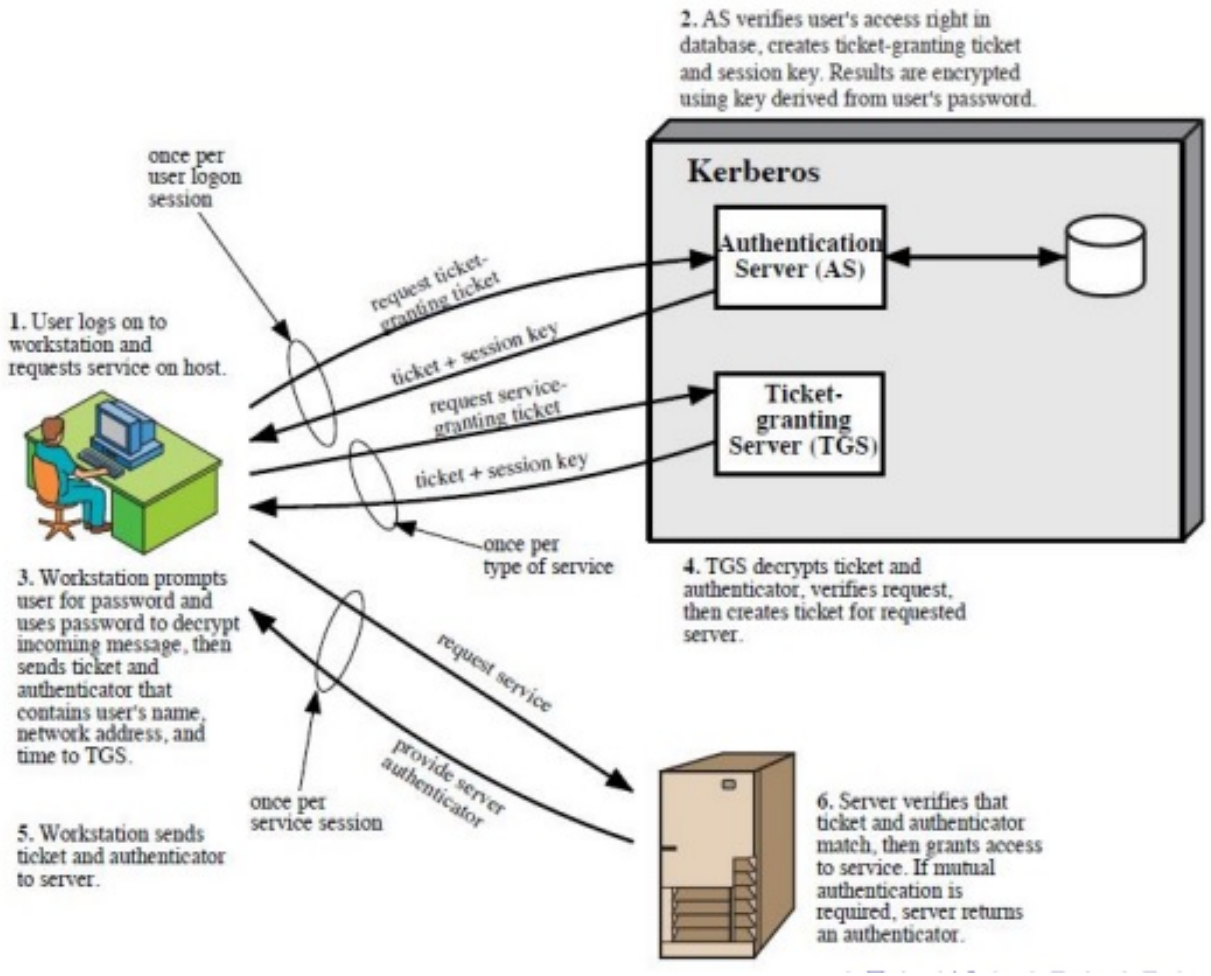
(6) V → C $E(K_{c,v}, [TS_5 + 1])$ (for mutual authentication)

$Ticket_v = E(K_v, [K_{c,v} \| ID_C \| AD_C \| ID_v \| TS_4 \| Lifetime_4])$

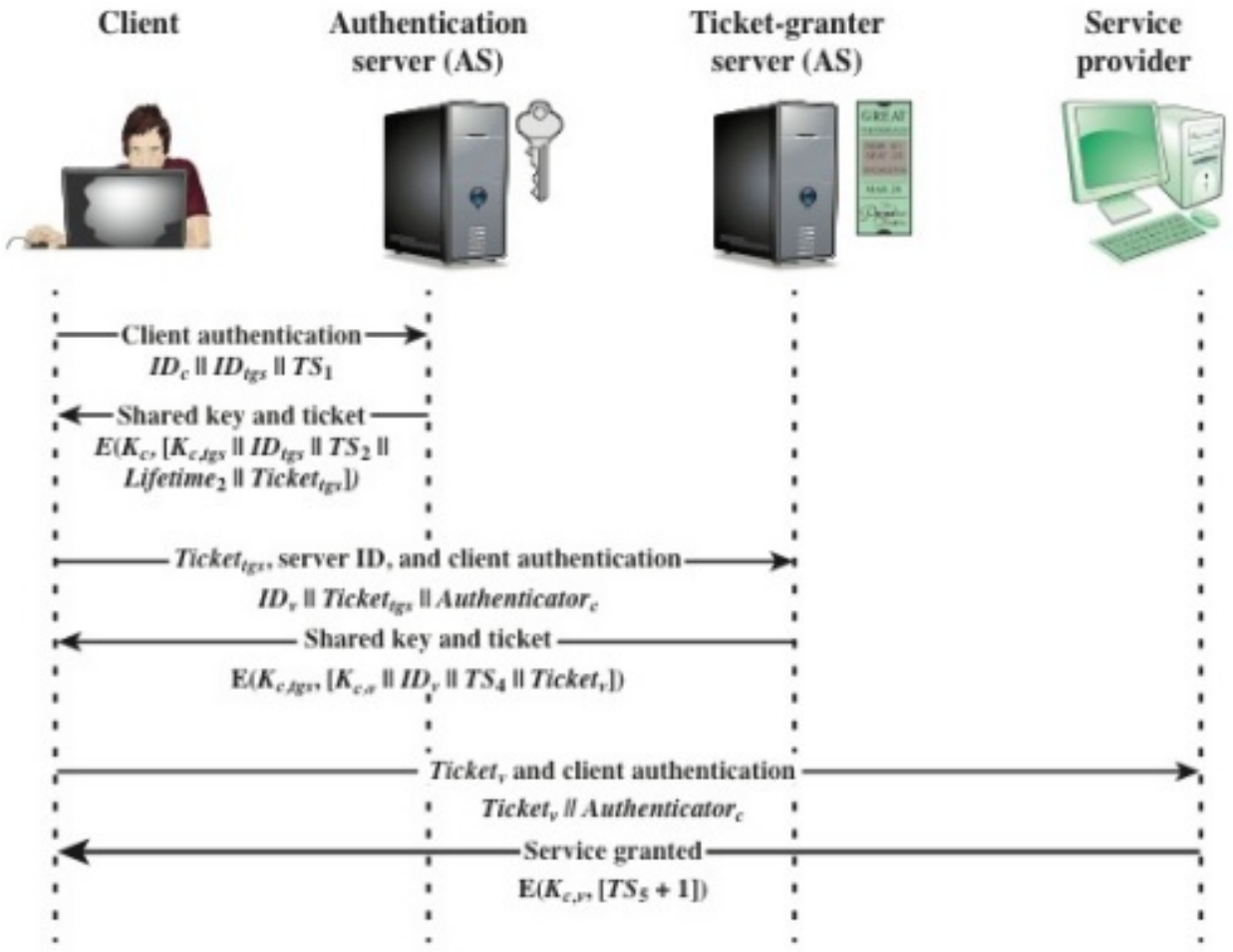
$Authenticator_c = E(K_{c,v}, [ID_C \| AD_C \| TS_5])$

(c) Client/Server Authentication Exchange to obtain service

L'échange de messages par Kerberos V4



L'échange de messages par Kerberos V4



Merci pour votre attention!