SSO

I) Généralités :

Le SSO (Single Sign on, authentification unique et une seule fois) est un système qui permet à l'utilisateur de saisir son mot de passe une seule fois pour accéder à toutes les applications web, améliorant ainsi l'ergonomie d'accès aux applications et la sécurité du système d'information en limitant la circulation des mots de passe.

Lors de l'unique saisie du mot de passe, l'identifiant de l'utilisateur et ses attributs sont ensuite propagés vers les applications.

Certains logiciels SSO assurent la fermeture des sessions applicatives de l'utilisateur lorsqu'il se déconnecte.

L'architecture SSO se base sur un serveur d'authentification (certificat X509, Kerberos).
Pour authentifier les utilisateurs le serveur d'authentification peut se baser sur un annuaire LDAP.

II) Fonctionnement :

2.1) Architecture :

L'architecture d'un grand nombre de SSO est inspirée de Kerberos.

L'authentification est assurée par le serveur d'authentification qui délivre des tickets aux clients et aux applications.

Le ticket application transite par le client.

Les applications font confiance au serveur d'authentification.

Kerberos utilise la cryptographie symétrique. Pour simplifier l'architecture du système, l'utilisation de certificats X509 utilisant des algorithmes asymétriques est possible.

2.2) Le serveur d'authentification :

Le serveur d'authentification est l'élément central du SSO puisqu'il assure :
- L'authentification.
- La persistance de la connexion.
- La propagation de l'identité de l'utilisateur auprès des applications.

Le serveur à la charge de vérifier le mot de passe de l'utilisateur auprès d'une base de référence (NIS, LDAP, /etc/passwd …).

S'il s'agit d'un certificat, il vérifie la validité de certificat, la chaîne de certification et les listes de révocation.

Une fois l'utilisateur authentifié, le serveur maintient la session en plaçant un cookie http sur son poste. Les données contenues dans le cookie sont protégées (cryptage ou utilisation d'un ticket interprété par le serveur).

Le cookie http est le seul moyen technique fiable pour que l'utilisateur soit reconnu comme authentifié lors de son prochain accès au serveur.

Pour que l'utilisateur puisse utiliser une application, le serveur fournit son identité à l'application. L'identité transite par le poste de l'utilisateur soit par redirection http, soit par un document html incluant un javascript de redirection.

L'application ne fait appel qu'une fois au serveur d'authentification.

2.3) L'agent d'authentification :

Agent intégré :

L'agent d'authentification est la brique SSO intégrée aux applications (librairie, module apache).

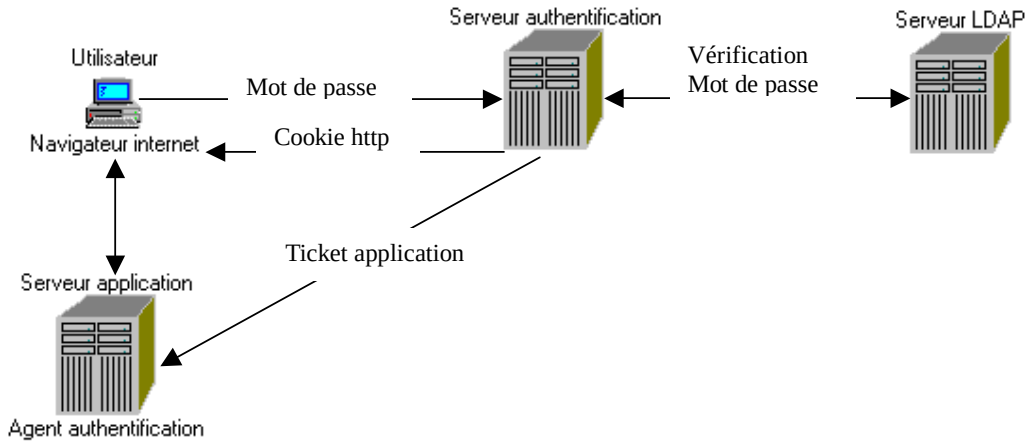
L'agent vérifie que l'utilisateur est authentifié. S'il n'est pas authentifié, il le redirige vers le serveur d'authentification.

Si le client est authentifié (présence cookie http), l'agent vérifie l'origine des données (les données peuvent être signées) et les transmet à l'application.

Dans le cas d'architecture multi tiers, certains SSO véhiculent des jetons d'authentification au lieu des attributs de l'utilisateur entre le serveur d'authentification et l'agent. L'agent contacte directement le serveur (https ou web service) et lui présente le jeton pour validation, en retour il reçoit les données sur l'utilisateur.

Schéma :

Attention, le ticket application transite par le poste utilisateur.



Agent reverse proxy :

L'intégration d'un agent d'authentification peut être coûteuse et difficile. Il est possible d'utiliser un reverse proxy qui consiste à installer en frontal des applications un agent d'authentification.

L'agent reverse proxy intercepte les accès utilisateurs afin de les identifier pour transmettre les données d'identification à l'application via des variables d'environnement (module serveur web) ou dans des champs d'entête http.

La plupart des SSO sont fournis avec un module de ce type pour apache.