Single Sign-On (SSO) – 1ère partie

1) Qu'est ce que le SSO ?

La multiplication des accès et des mots de passe amène les utilisateurs à faire de leur mieux pour conserver les informations relatives à leurs comptes. Malheureusement cela inclut souvent le recours à des pratiques dangereuses telles que : inscrire les codes secrets sur leur agenda papier ou sur des post-it, utiliser le même pour la plupart de leurs accès, ou laisser les connexions ouvertes lorsqu'ils quittent leur poste de travail.
Une bonne solution pour ce problème est de s'appuyer sur une solution de Single Sign-On. Le Single Sign-On (SSO) est un processus qui permet à un utilisateur de s'authentifier une seule fois pour accéder à plusieurs applications ou ressources.

Mais aussi pratiques soient-ils, les SSO sont rarement déployés simplement pour faciliter la vie des utilisateurs. Ils s'intègrent généralement à un projet de sécurité plus large, dans lequel ils sont considérés comme un éléments secondaires.

Les types d'authentification SSO
Il existe deux types d'authentification SSO; la première dite Web SSO, le seconde dite Enterprise SSO (eSSO).

Le **Web SSO** prend en charge toutes les applications qui utilisent un navigateur Web pour vous connecter à des applications.

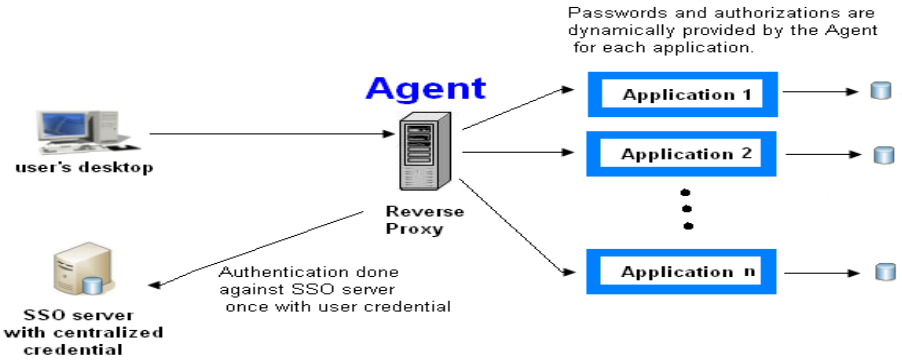

Fig 1: Web SSO architecture

Pour leur part, les systèmes **eSSO** ne sont pas limités aux applications web et sont conçus pour minimiser le nombre de fois qu'un utilisateur doit taper son login et son mot de passe pour se connecter à de multiples applications de l'entreprise.

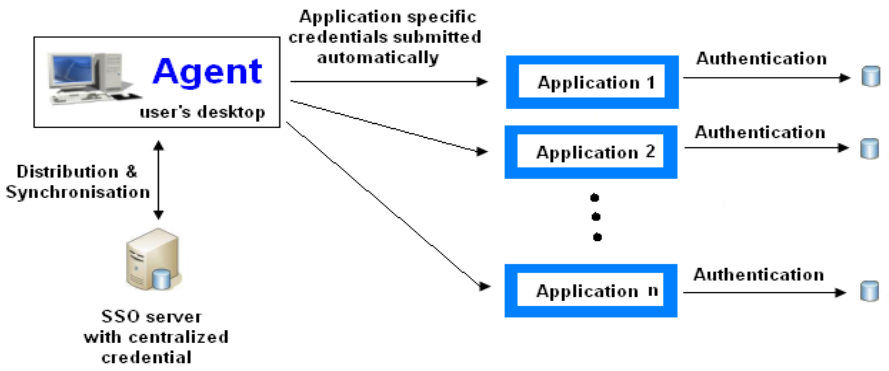

Fig 2: Enterprise SSO architecture

Les composantes SSO

Dans le système SSO, nous trouvons généralement les éléments suivants:

- Le client qui demande l'accès à l'application. Généralement, il s'agit d'un navigateur Web (cas d'un WebSSO). Dans le cas plus général d'une application client / serveur, le client peut être par exemple un client telnet.
- Le serveur d'authentification qui conserve la base de données des informations d'indentification. C'est l'élément central du système de SSO.
- Le serveur d'application qui fournit les ressources en fonction du résultat du processus d'authentification
- L'agent SSO qui s'interpose entre le client et les serveurs d'applications prend en charge (au moins en partie) la phase d'authentification du client vis-à-vis du serveur d'application. L'agent peut être localisé à différents endroits selon l'architecture SSO, et peut être matériel ou logiciel.

2) Les différentes approches de SSO

2.1 L'approche centralisée

Le principe est de disposer d'une base de données centralisée contenant tous les utilisateurs. Cela permet également de centraliser la gestion de la politique de sécurité. Un exemple de mise en œuvre est LDAP.

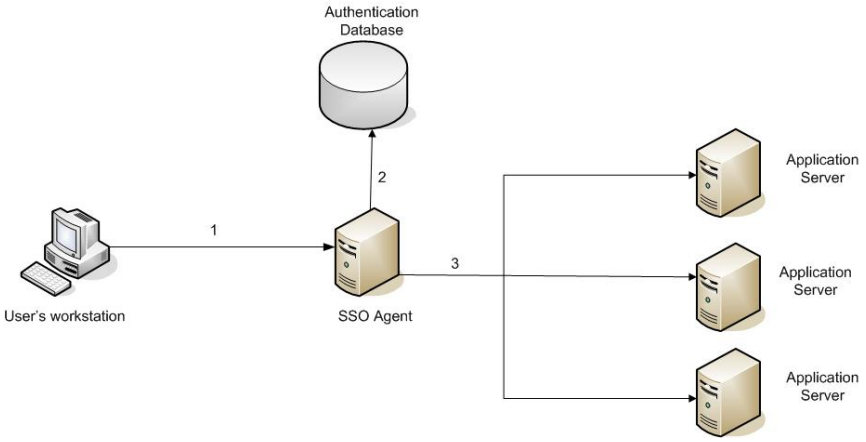

Fig 3: Centralized SSO

1. Le client souhaite accéder à une application. Dans ce cas, l'agent s'exécute sur un Reverse Proxy et intercepte la demande.
2. L'agent authentifie l'utilisateur d'après une base de données d'authentification qui peut être un annuaire LDAP.
3. Une fois l'utilisateur authentifié, il peut accéder à l'application.

2.2 L'approche fédérative

Le grand défi dans les infrastructures d'authentification aujourd'hui est d'étendre le SSO pour couvrir plusieurs autorités d'authentification «différentes » (mise en œuvre sur différentes plates-formes ou gestion par des organisations différentes). La fédération permet d'étendre le contrôle d'accès et le SSO à travers les frontières organisationnelles. En effet, la mise en œuvre de l'identité fédérée et l'extension du SSO dans les entreprises permet de distribuer le contrôle et la maintenance des activités, et donc d'avoir plus de commodité et de temps à la fois pour les organisations et les utilisateurs. L'approche fédérée permet à un utilisateur de manière transparente de parcourir les différents sites, services au sein d'une fédération donnée. Chaque service gère une partie des données d'un utilisateur mais partage les informations de cet utilisateur avec les services partenaires.

Cette approche a été développée pour répondre à un besoin de gestion décentralisée des utilisateurs, où chaque service partenaire désire conserver le contrôle de sa politique de sécurité.

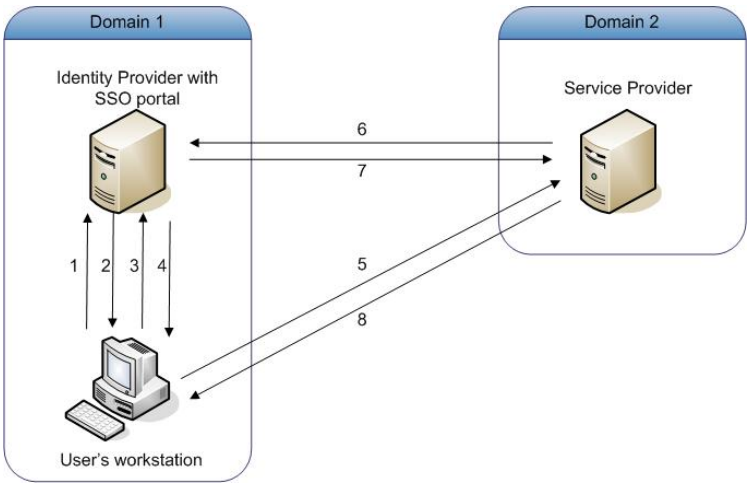

Fig 4: Federated SSO

1. L'utilisateur se connecte au fournisseur d'identité (IdP).
2. Après une authentification réussie, l'IdP envoie à l'utilisateur des informations sur les applications auxquelles il peut accéder.
3. L'utilisateur clique sur le lien Service Provider (SP) dans le portail. Il s'agit d'un lien spécial, qui ne se connecte pas directement au SP.
4. L'IdP reçoit la demande et crée une « Identity Assertion » (un identifiant d'identité). L'IdP conserve cette « Identity Assertion » avec un "artefact" pointeur dans son cache. Puis, l'IdP renvoie une réponse redirigée vers le navigateur client.
5. Le navigateur est redirigé vers le SP avec «l'artefact».
6. Le SP reçoit cette demande et contacte l'IdP avec « l'artefact » "pour demander l' »Identity Assertion » réelle.
7. L'IdP reçoit la demande, et vérifie cette entrée dans la table des « Identity Assertion » en cache en utilisant « l'artefact » comme index. Il crée une « Identity Assertion » au format SAML, et le renvoie à la SP.
8. Le SP extrait les informations utilisateur de l' « Identity Assertion » reçue. Enfin, après une authentification locale réussie, l'utilisateur est autorisé à accéder au service.

Le principal exemple de l'approche fédérée est Liberty Alliance. Elle s'appuie principalement sur la norme SAML, ainsi que sur les protocoles http et SSL.
Un autre protocole pour établir la confiance entre des systèmes hétérogènes d'identité existe. C'est un protocole relativement nouveau qui est appelé Web Service Federation (WS-Federation).

2.3 L'approche coopérative

Cette approche est similaire à l'approche de la Fédération. Elle répond aux besoins de structures institutionnelles, par exemple, des laboratoires de recherche ou des administrations. Dans l'approche coopérative, chaque utilisateur dépend de l'une des entités partenaires. Quand il essaie de parvenir à un service du réseau, l'utilisateur est authentifié par le partenaire dont il dépend. Comme dans l'approche fédérative, tous les services du réseau gèrent indépendamment leur propre politique de sécurité. Avec cette approche, les identifiants de sécurité de l'utilisateur ne sont pas échangés. Les principaux représentants de cette approche sont Shibboleth et Central Authentication Service (CAS).

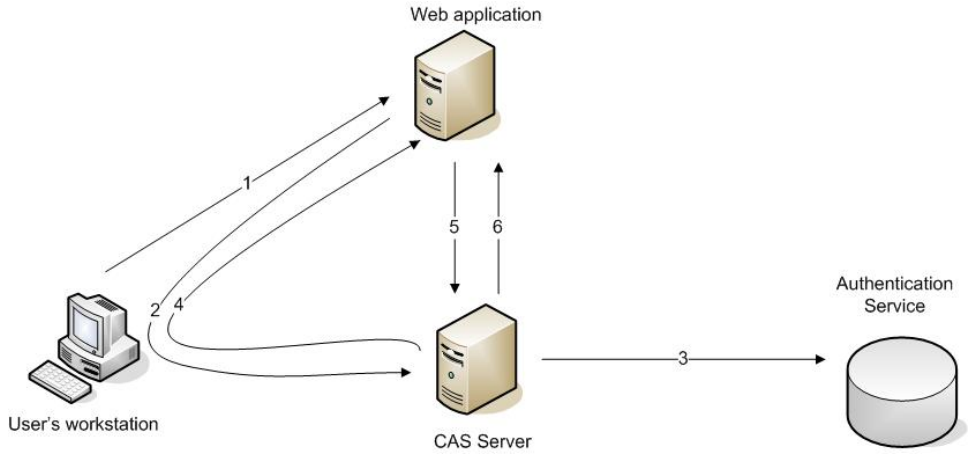

Fig 5: functional scheme of CAS

1. Le client se rend sur une application nécessitant une authentification.
2. L'application le redirige vers le CAS.
3. Le CAS valide l'authenticité du client, généralement par le contrôle d'un nom d'utilisateur et d'un mot de passe.
4. Si l'authentification réussit, le CAS renvoie le client vers l'application, avec un ticket de service (ST).
5. L'application valide le ticket en communiquant avec le CAS au travers d'une connexion sécurisée.
6. Le CAS donne ensuite à l'application l'information fiable que l'utilisateur est authentifié avec succès.

3) Les architectures SSO

Les principaux modèles d'architecture SSO sont :

- le SSO côté client (client-side SSO),
- le SSO côté serveur (server-side SSO)
- et des systèmes hybrides.

La plupart des eSSO sont basés sur une architecture SSO côté client alors que le Web SSO utilise un modèle SSO côté serveur.

3.1 SSO côté client

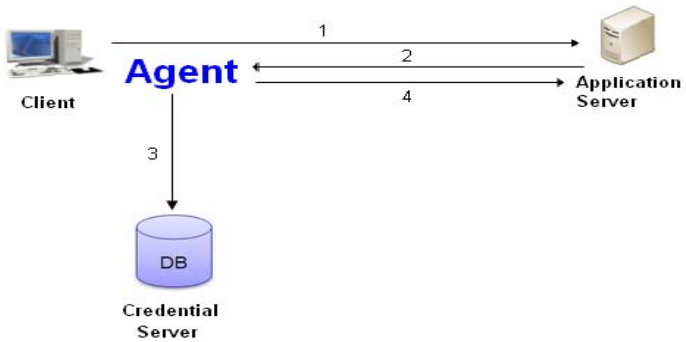

Fig 6: Client side SSO with a central credential database

1. Le client souhaite accéder à une application.
2. L'application demande ses informations d'identification. L'agent présent sur le poste client intercepte la demande.
3. L'agent vérifie les informations d'identification dans le serveur centralisé des identifications.

4. L'agent simule l'utilisateur réel et envoie les informations d'identification à l'application. Ainsi, l'identification est transparente pour l'utilisateur.

3.2 SSO côté serveur

Il existe deux types de SSO "Server-Side" : ceux qui utilisent un reverse proxy et ceux utilisant des agents "serveurs". Avec cette architecture, il n'est pas nécessaire d'installer un agent sur chacun des PC de l'utilisateur.

Agent

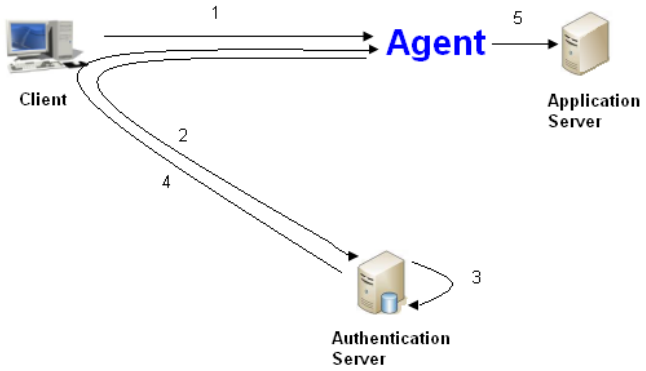

Fig 7: Server Side SSO with Agent on Application server

1. Le client souhaite accéder à une application. L'agent présent sur le serveur d'application intercepte la demande.
2. L'agent vérifie que l'utilisateur est authentifié : s'il ne l'est pas, l'agent le redirige vers le serveur d'authentification. Cette redirection peut apparaître comme un portail ou une fenêtre. L'utilisateur fournit ses informations d'identification pour le serveur d'authentification.
3. Le serveur d'authentification vérifie l'identité de l'utilisateur dans la base de données de référence.
4. Une fois l'utilisateur authentifié, le serveur d'authentification renvoie un cookie HTTP sur le poste de l'utilisateur qui permet de maintenir la session de l'utilisateur.
5. L'agent le transfère sur le serveur d'application.

Reverse Proxy

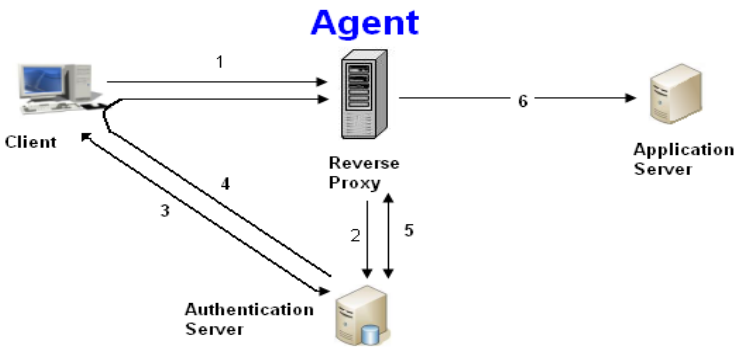

Fig 8: Server Side SSO with Agent on Reverse Proxy

1. Le navigateur Web tente de se connecter à l'application Web. Toute demande de connexion pour une application est redirigée vers le reverse proxy.

2. L'agent sur le reverse proxy intercepte la demande et vérifie l'authentification de l'utilisateur via le serveur d'authentification.
3. Si l'utilisateur n'est pas authentifié, le serveur d'authentification demande à l'utilisateur des informations d'identification. L'utilisateur fournit ses informations d'identification pour le serveur d'authentification.
4. Le serveur d'authentification envoie un jeton jouant le rôle de cookie et redirige le navigateur.
5. L'agent sur le reverse proxy intercepte la demande et vérifie l'authentification de l'utilisateur via le serveur d'authentification avec le jeton. Le serveur d'authentification envoie le login et l'autorisation des informations qui est associée avec le jeton.
6. L'agent permet d'accéder à la demande.

3.3 SSO hybride

Il existe de nombreuses approches hybrides combinant client-side SSO et server-side SSO. Cette architecture permet de réduire certains problèmes du SSO côté client.

Pour plus d'informations

http://www.01net.com/article/256916.html
http://fr.wikipedia.org/wiki/Authentification_unique
http://www.cesnet.cz/doc/techzpravy/2006/web-sso/
http://www.cru.fr/documentation/federation/index
http://www.ufinity.com/media/pdf/whitepapers/SSO%20Architecture%20Comparisons.pdf
http://www.projectliberty.org/

Nota : Le mois prochain, la deuxième partie de cet article vous décrira les fonctionnalités du SSO et quelques outils.

Fin du document