République algérienne démocratique et populaire
Université Saad-Dahlab Blida 1
Faculté d'informatique
Département des science

Central Authentication Service

2018/
2019

Fait par :

Mehdi Nacer KERKAR

Etapes de presentation

- SSO (Single Sign-On)
- Central Authentication Service (CAS)
- Fonctionnement de base
- Fonctionnement multi-tiers
- Implémentation

Bonjour !

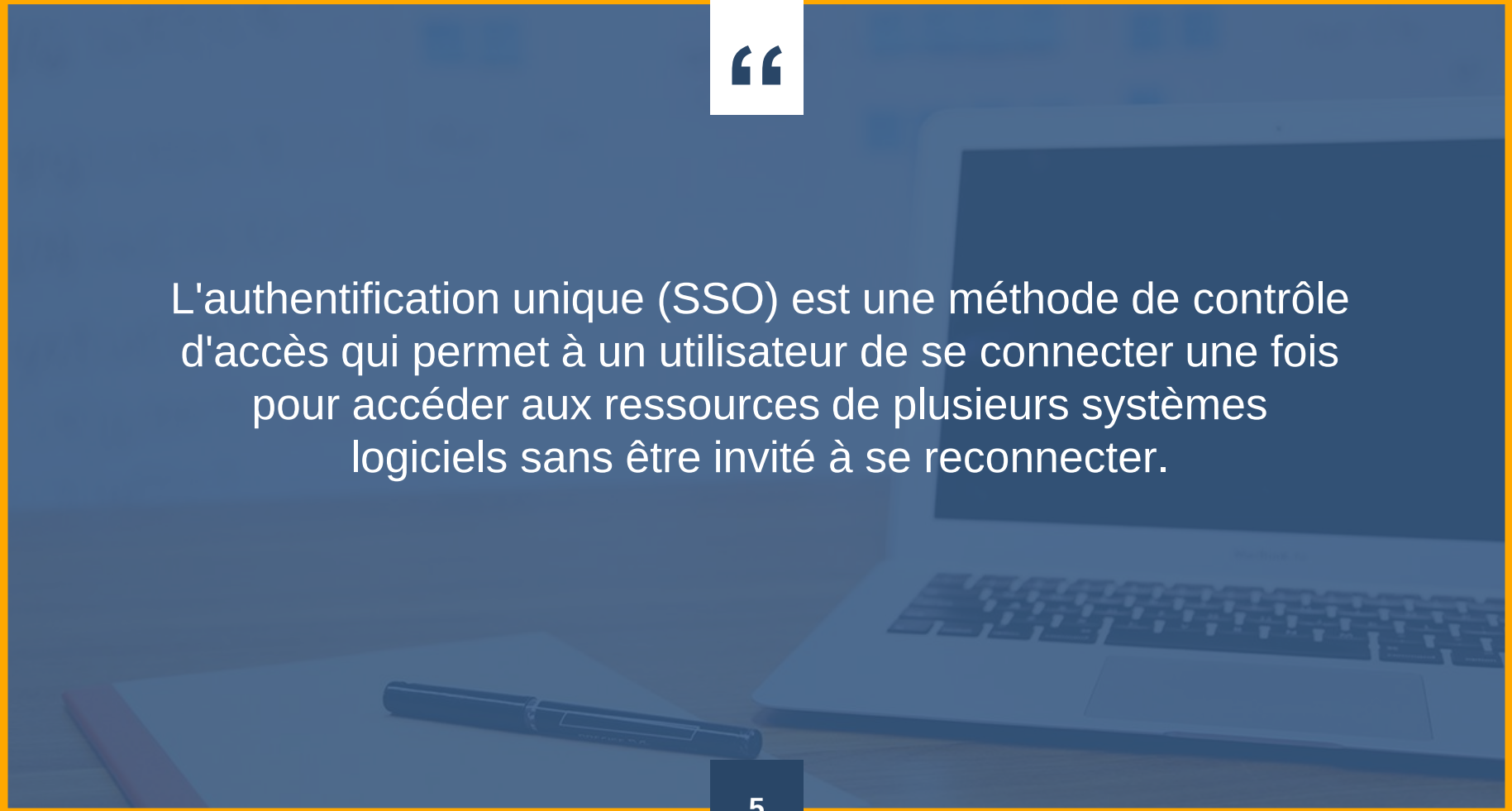
Je suis Mehdi Nacer KERKAR

Etudiant en Sécurité System d’Information, Fan de Cyber Sécurité.

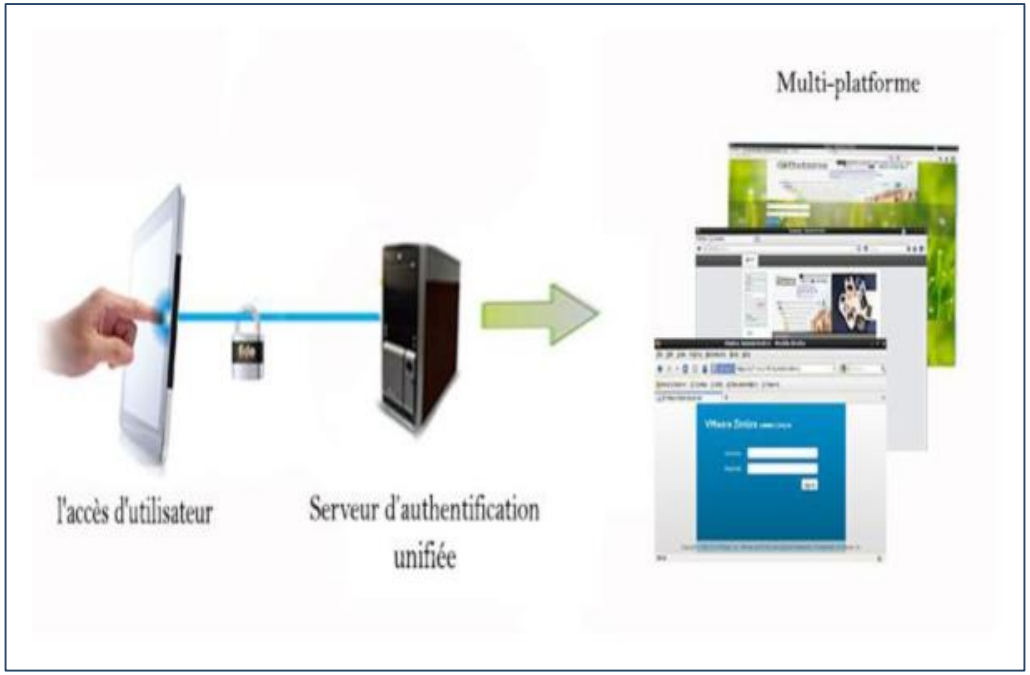
Je suis là, parce que j’aime présenter.

SSO (Single Sign-On)

Commencons avec la définition

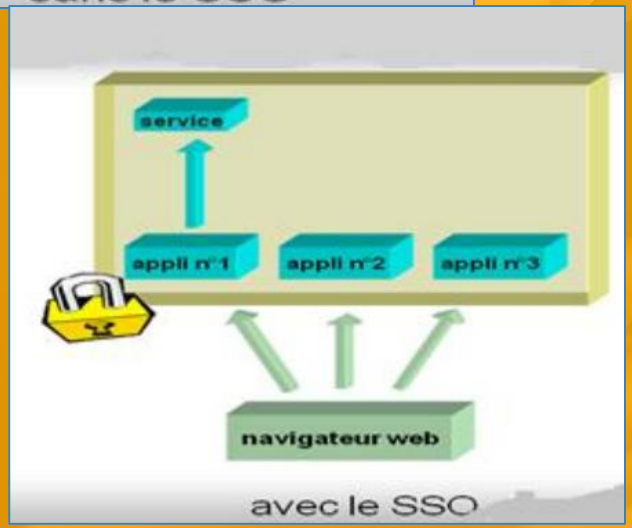


L'authentification unique (SSO) est une méthode de contrôle d'accès qui permet à un utilisateur de se connecter une fois pour accéder aux ressources de plusieurs systèmes logiciels sans être invité à se reconnecter.



Authentification unifiée des utilisateurs

Principe du SSO



Objectif d'un Système SSO

SSO, consiste à centraliser l'authentification afin de permettre à l'utilisateur d'accéder à toutes les ressources.

En unifiant les comptes en un couple unique identifiant/mot de passe.

Ce système d´authentification est l´occasion de donner de bonnes habitudes aux utilisateurs : ils ne doivent délivrer leur mot de passe qu´au seul serveur d´authentification dont la bannière de login et l´URL sont bien identifiés.

Les Avantages Du Single Sign-On

Le SSO permet une amélioration à la sécurité et aux confort des utilisateurs :

- Réduit le nombre de mot de passe pour chaque utilisateur.
- Restreint le nombre d'échange réseau pour l'authentification.
- Limite le nombre de saisies pour les authentifications.
- Diminue les appels au support informatique.

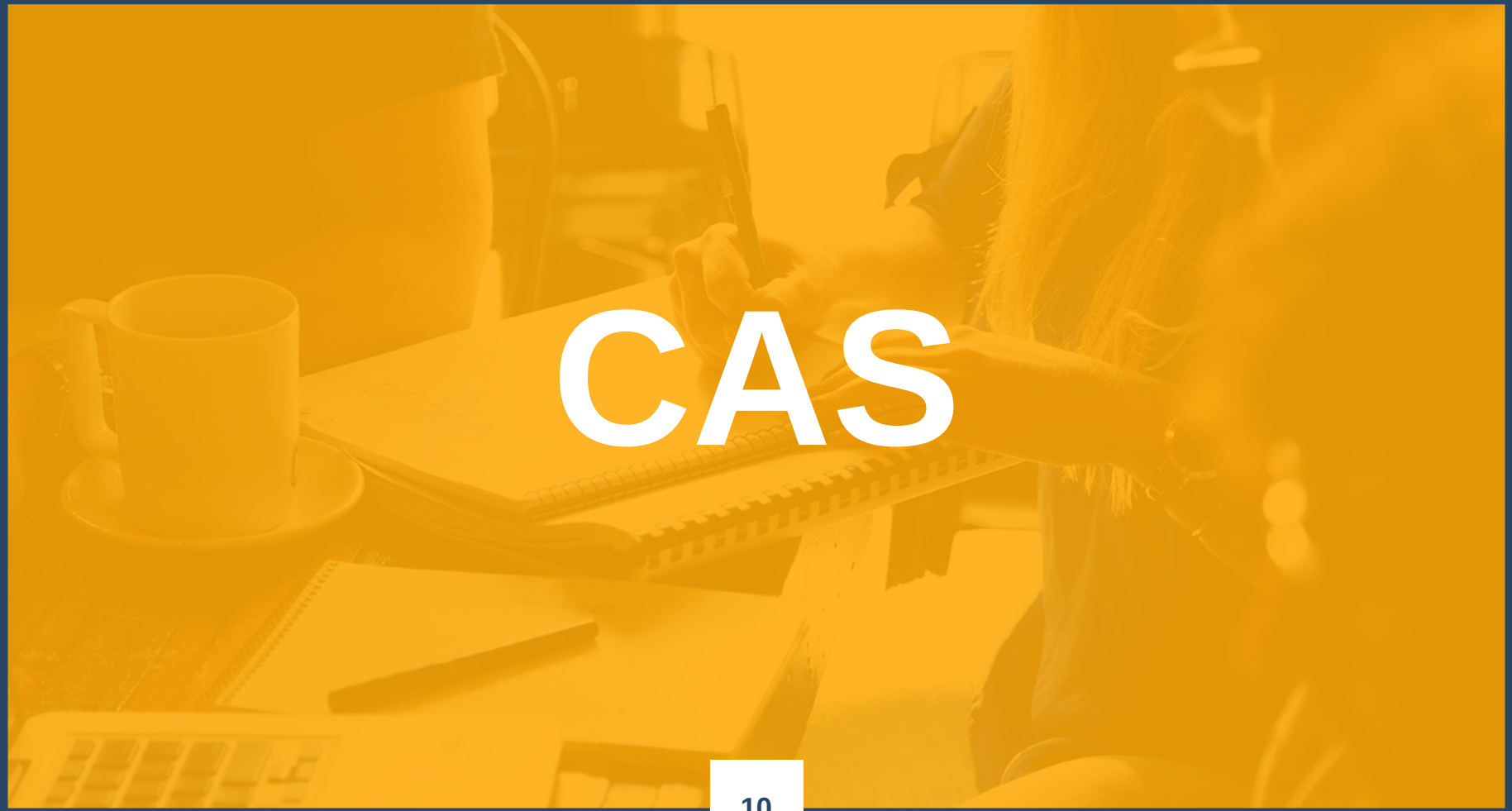
En centralisant l'authentification dans un seul système :

- Moins de risque de faille.
- Permet un système de traçabilité plus facile.
- Limite les risques d'inconsistance.

Différentes Solutions de SSO

Le tableau suivant présente quelques-unes des Sign-On (SSO) simples implémentations disponibles:

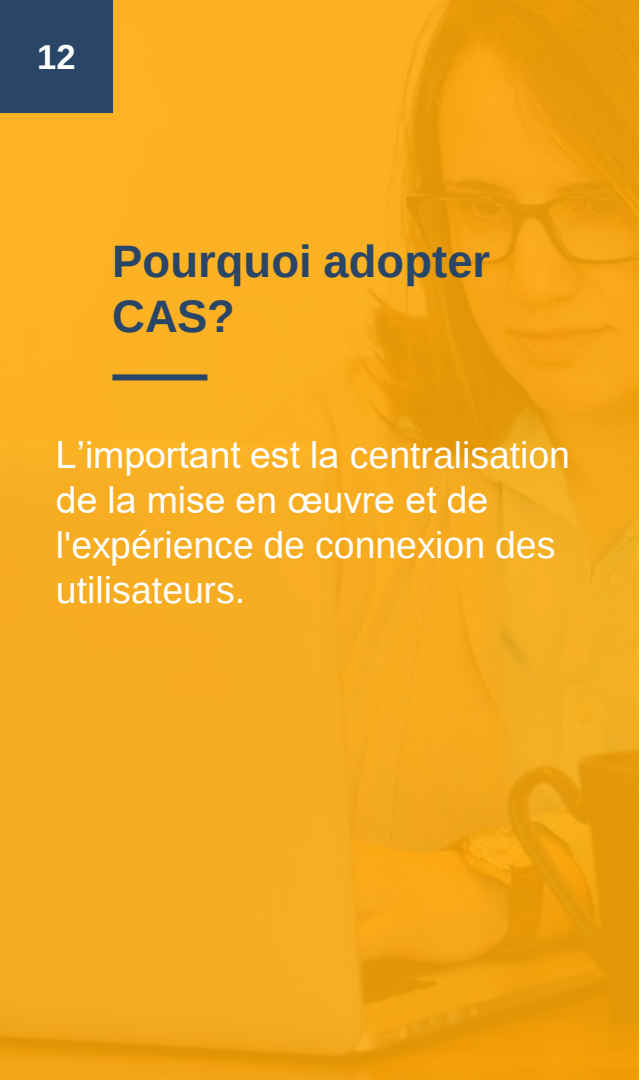
Nom du produit	Description
OpenSSO/OpenAM	Portant le nom d'oracle OpenSSO en juillet 2008 puis le nom OpenAM depuis 2010, c'est l'un des serveurs d'authentifications unique et de fédération complet de l'OpenSource.
SAML	SAML pour Security Assertion Markup Language permet entre autres la délégation d'authentification.
CAS	C'est un service d'authentification centralisé SSO pour les applications Web, inspiré de Kerberos et basé sur le protocole HTTP(S).
SHIBBOLETH	Shibboleth est développé depuis 2001 par Internet2 et désigne à la fois une norme et un produit open source. C'est une extension de SAML qui enrichit ses fonctionnalités de fédération d'identités.
OpenID	implémenté et utilisé par les sociétés clés de l'Internet (Yahoo, MySpace, Google, Microsoft...), propose un protocole ouvert pour une gestion décentralisée d'identités, mettant l'utilisateur au centre des décisions le concernant.
LIBERTY ALLIANCE	Implémenté par IBM et utilisé par Sun et Novell, utilise des jetons SAML. Ce modèle a été développé pour répondre à un besoin de gestion décentralisée des utilisateurs


CAS

Central Authentication Service (CAS):

Le service d'authentification central (CAS) est un protocole de connexion unique pour le Web. Son but est de permettre à un utilisateur de se connecter à plusieurs applications simultanément et automatiquement.

Il permet également aux applications Web non fiables d'authentifier les utilisateurs sans accéder aux informations d'identité de sécurité d'un utilisateur, telles que les mots de passe.



Pourquoi adopter CAS?

L’important est la centralisation de la mise en œuvre et de l'expérience de connexion des utilisateurs.

• Les applications participantes ne touchent pas le mot de passe de l’utilisateur final.

• Offre des fonctionnalités pour l'authentification par proxy.

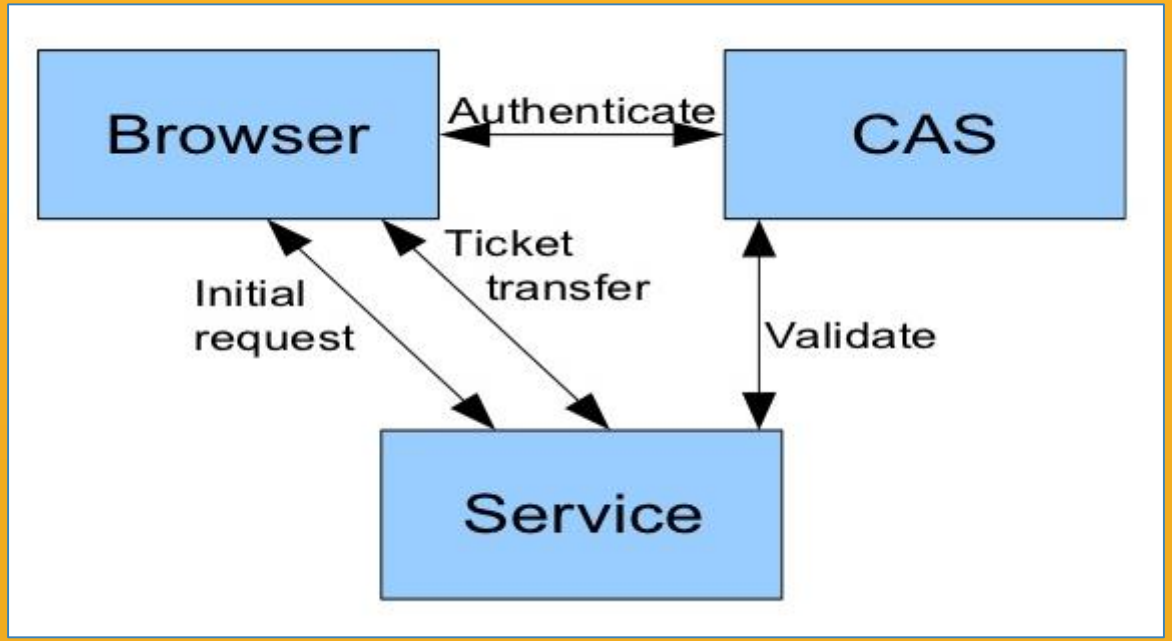
• Possibilité d'appliquer des stratégies uniformes d'authentification et d'autorisation d'entreprise sur l'ensemble du système.

• Des sessions d'audit utilisateur de bout en bout pour améliorer les rapports de sécurité et l'audit.

• Évite aux développeurs d'applications de comprendre et d'implémenter la sécurité de l'identité dans leurs applications.

• Génère généralement des économies importantes sur les coûts du service d'assistance par mot de passe.

L’architecture CAS



L’architecture CAS

Le serveur CAS

L'authentification est centralisée sur un serveur qui est le seul acteur du mécanisme CAS à avoir connaissance des mots de passe des utilisateurs.

Son rôle est double :

- authentifier les utilisateurs ;
- transmettre et certifier l'identité de la personne authentifiée (aux clients CAS).

Les clients CAS

Une application web muni d'une librairie cliente ou un serveur web utilisant le module mod_cas est alors appelé « client CAS ».

Il ne délivre les ressources qu'après s’être assuré que le navigateur qui l'accède se soit authentifié auprès du serveur CAS.

L’architecture CAS

Les navigateurs (web)

Les navigateurs doivent satisfaire les contraintes suivantes pour bénéficier de tout le confort de CAS :

- disposer d'un moteur de chiffrement leur permettant d'utiliser le protocole HTTPS ;
- savoir effectuer des redirections HTTP (accéder à une page donnée dans une entête Location lors d'une réponse 30x à une première requête HTTP) et interpréter le langage JavaScript ;
- savoir stocker des cookies, comme défini par [4]. En particulier, les cookies privés ne devront être retransmis qu'au serveur les ayant émis pour garantir la sécurité du mécanisme CAS.

Ces exigences sont satisfaites par tous les navigateurs classiquement utilisés, à savoir MicroSoft Internet Explorer (depuis 5.0), Netscape Navigator (depuis 4.7) et Mozilla.

Fonctionnement de base

Authentification d'un utilisateur

Le Ticket Granting Cookie (TGC) est le passeport de l'utilisateur auprès du serveur CAS. HTTPS).

Tous tickets utilisés dans le mécanisme CAS est opaque

TGC = (clé de session + adresse IP)

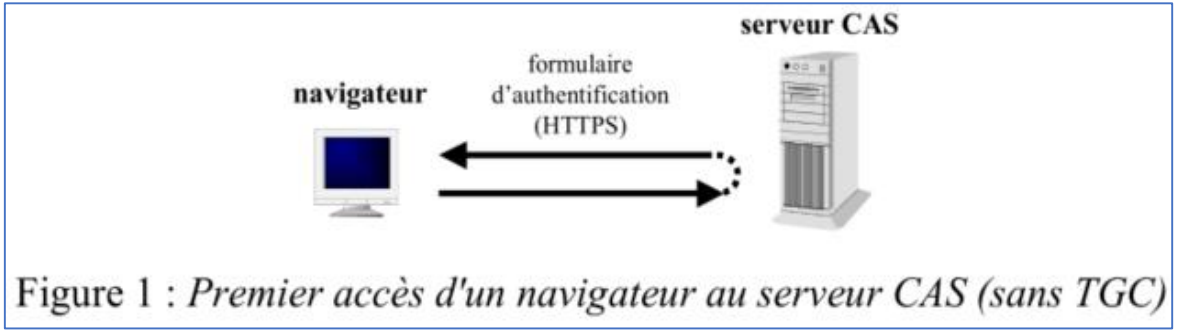


Figure 1 : *Premier accès d'un navigateur au serveur CAS (sans TGC)*

Si les informations sont correctes, le serveur renvoie au navigateur un cookie appelé TGC

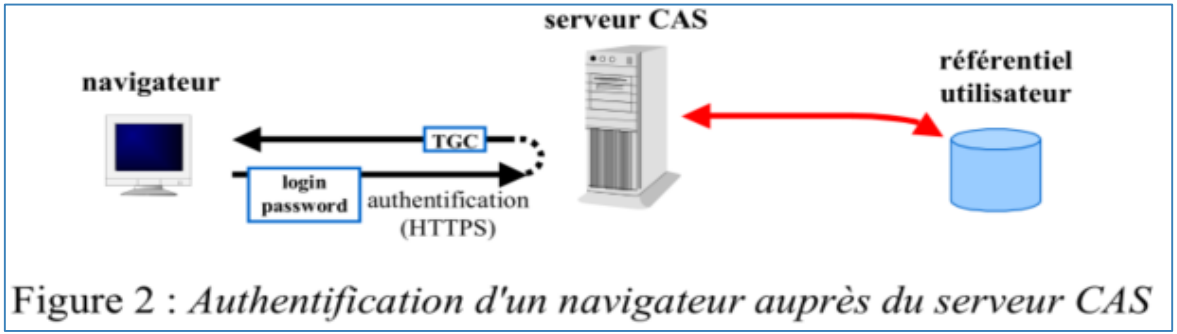


Figure 2 : *Authentification d'un navigateur auprès du serveur CAS*

Exemple

Séquence d'identifiants de session créés par le même client.

SID:ANON:www.w3.org:j6oAOxCWZh/CD723LGeXlf-01:34

SID:ANON:mc.ai.mit.edu:NRviSpoYm7mdkYB4W2471l-01:35

SID:ANON:www.w3.org:j6oAOxCWZh/CD723LGeXlf-01:36

SID:ANON:mc.ai.mit.edu:NRviSpoYm7mdkYB4W2471l-01:37

Accès à une ressource protégée après authentification

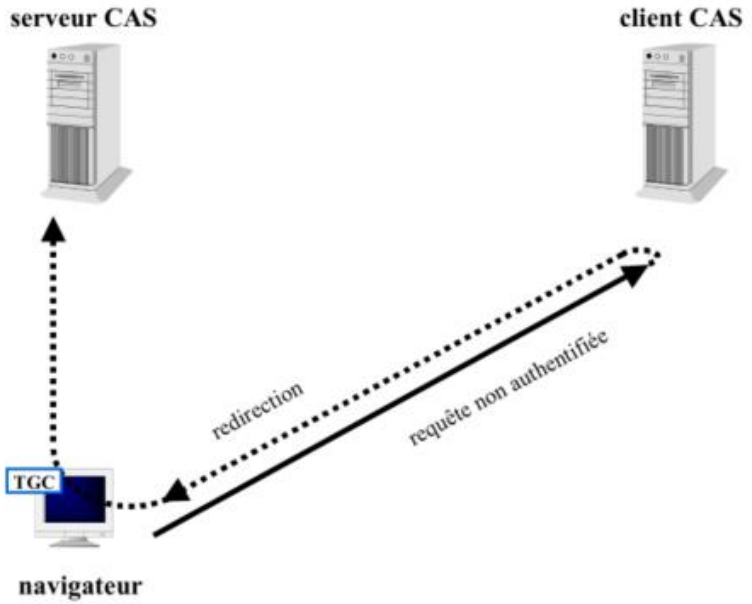


Figure 3 : *Redirection vers le serveurCAS d'un navigateur non authentifié auprès du client CAS*

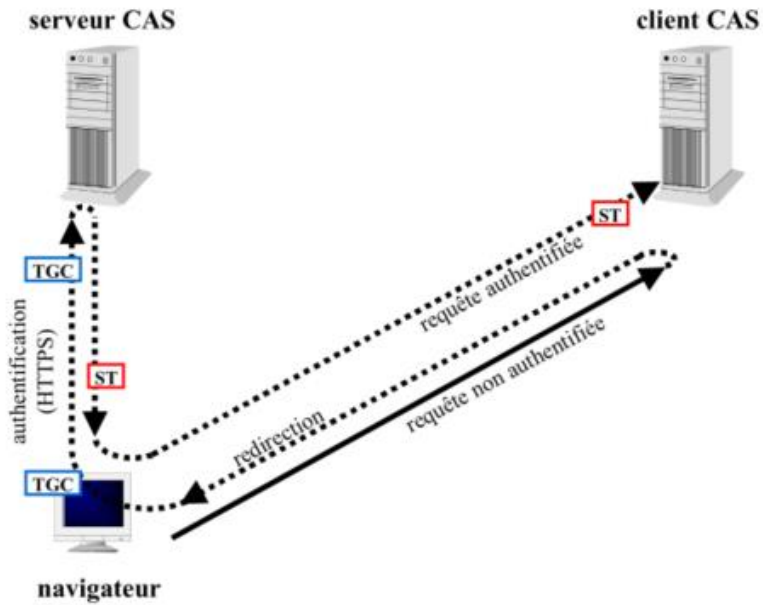


Figure 4 : *Redirection par le serveur CAS d'un navigateur vers un client CAS après authentification*

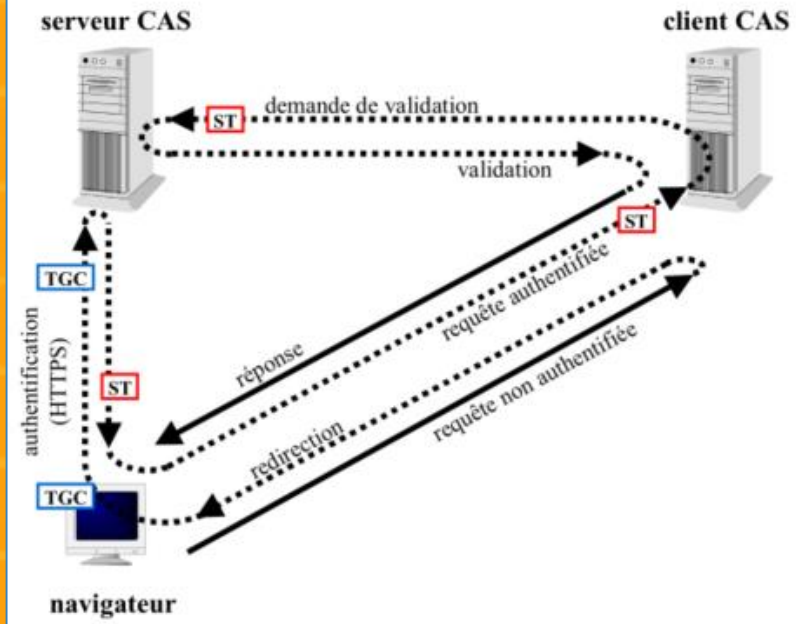


Figure 5 : *Validation d'un Service Ticket par un client CAS auprès du serveur CAS*

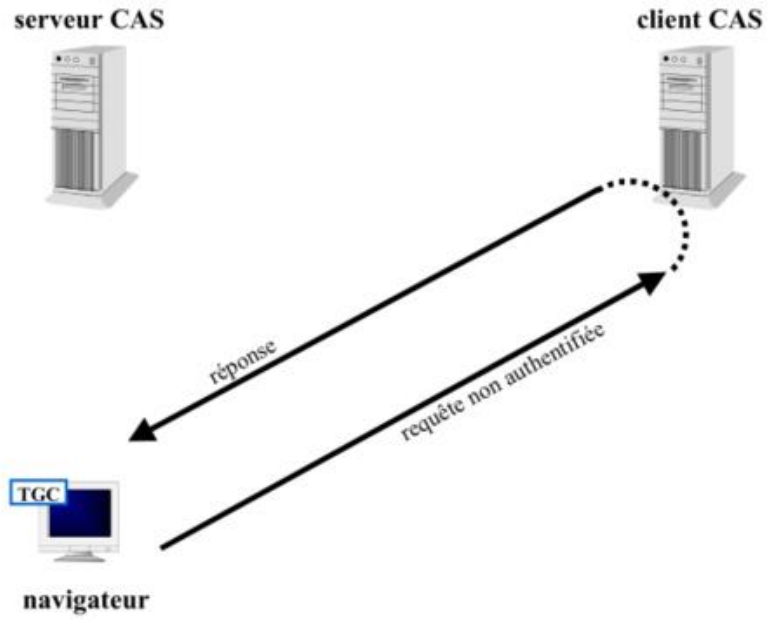


Figure 6 : *Vision par l'utilisateur du mécanisme d'authentification*

Fonctionnement 2-tiers

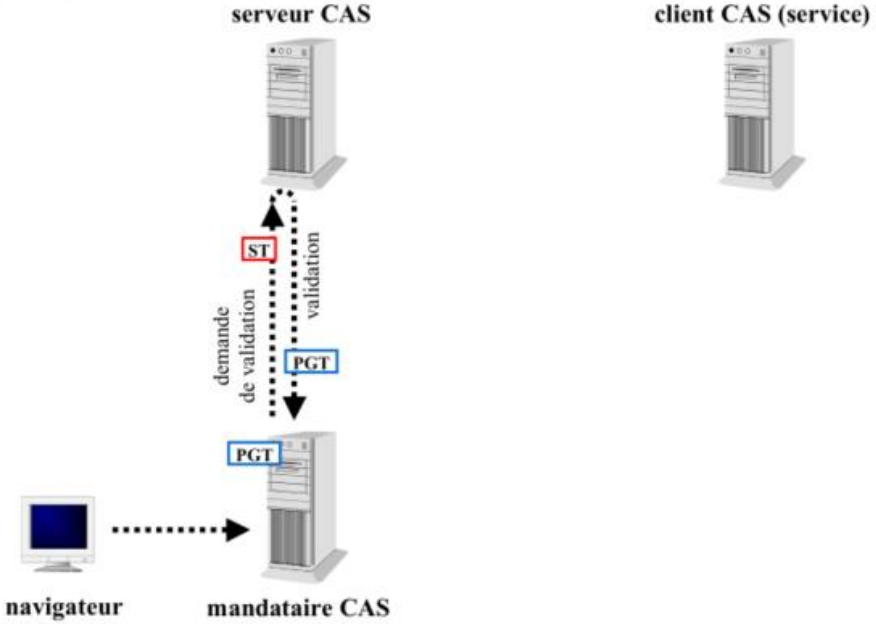


Figure 7 : *Récupération d'un PGT par un mandataire CAS auprès du serveur CAS*

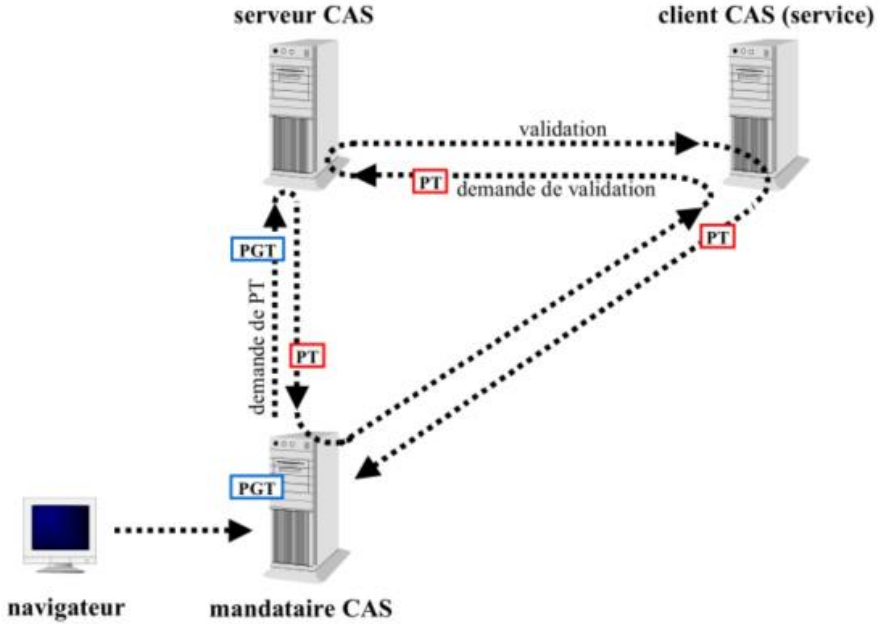


Figure 8 : *Validation d'un PT par un client CAS accédé par un mandataire CAS*

Fonctionnement n-tiers

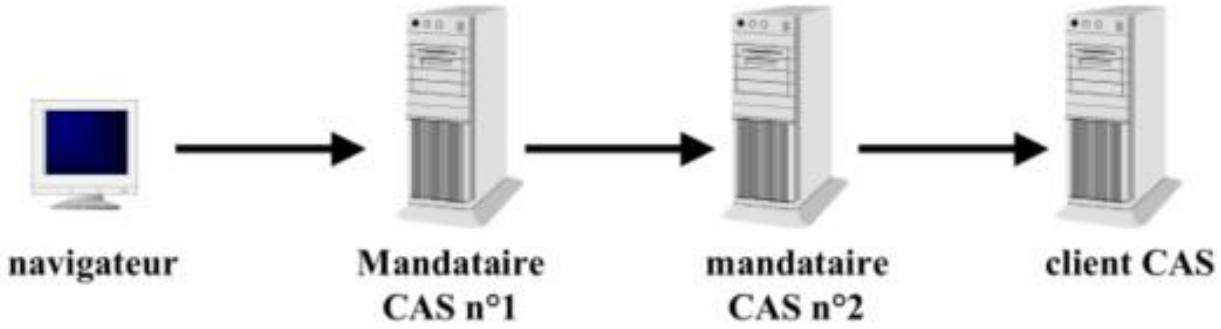
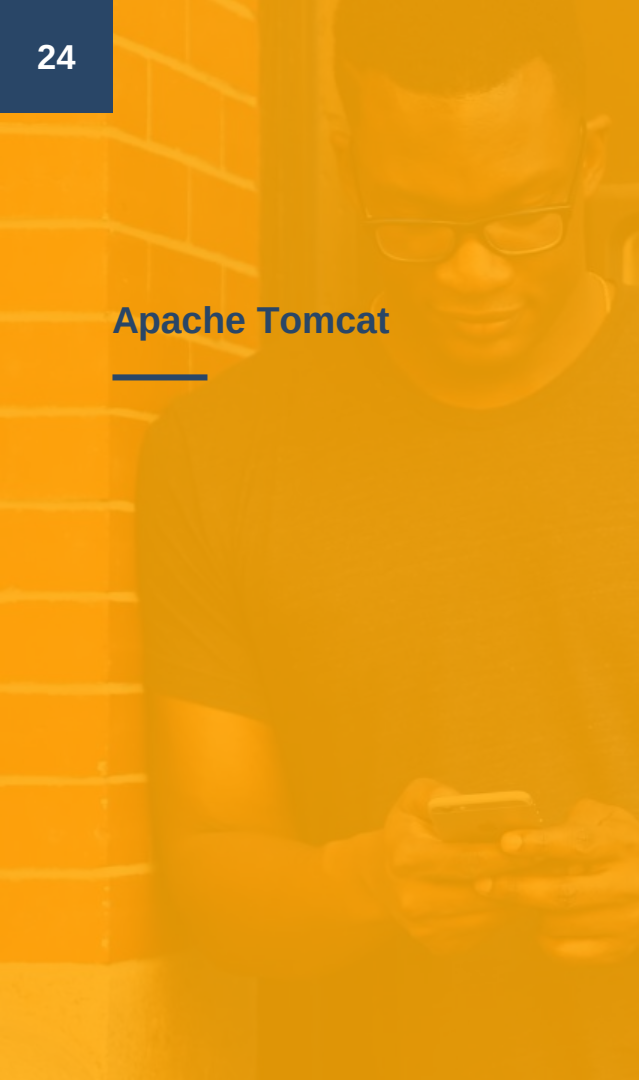


Figure 9 : *Schéma de fonctionnement n-tiers*

Implémentation

- **Pour serveur CAS :** Debian 6 comme système d'exploitation, Tomcat6, Openjdk-6,
- Maven-2.2.1, OpenLDAPv3, PhpLdapAdmin-1.2.2 et cas-server-3.4.11.
- **Pour le client :** Debian 5 comme système d'exploitation, Bind9, Zimbra 6 et cas-client-3.3.1.

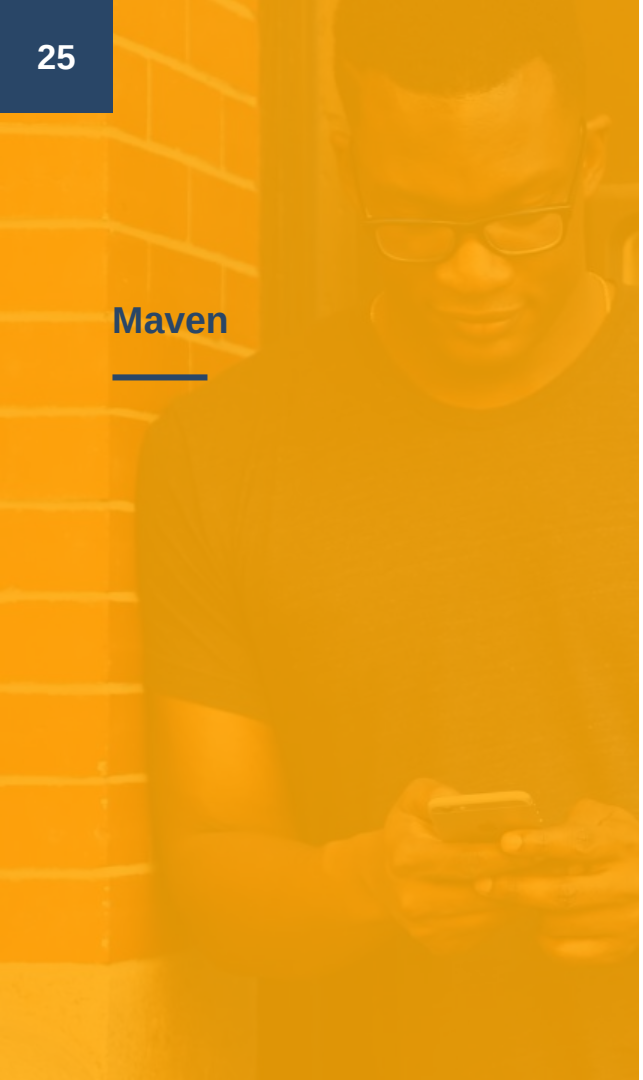
- **A) Apache Tomcat** est un conteneur libre de servlets et JSP Java EE.
- **B) Maven** est un outil open-source de gestion et d'automatisation de développement Java.
- **C) CAS**
- **D) OpenLDAP** est une implémentation libre du protocole LDAP.
  Le Lightweight Directory Access Protocol permet d'accéder aux services d'information de répertoire distribués et de les gérer sur un réseau de protocole Internet.
- **E) PhpLdapAdmin** qui est le client web permettant la gestion de l'annuaire LDAP.



Apache Tomcat

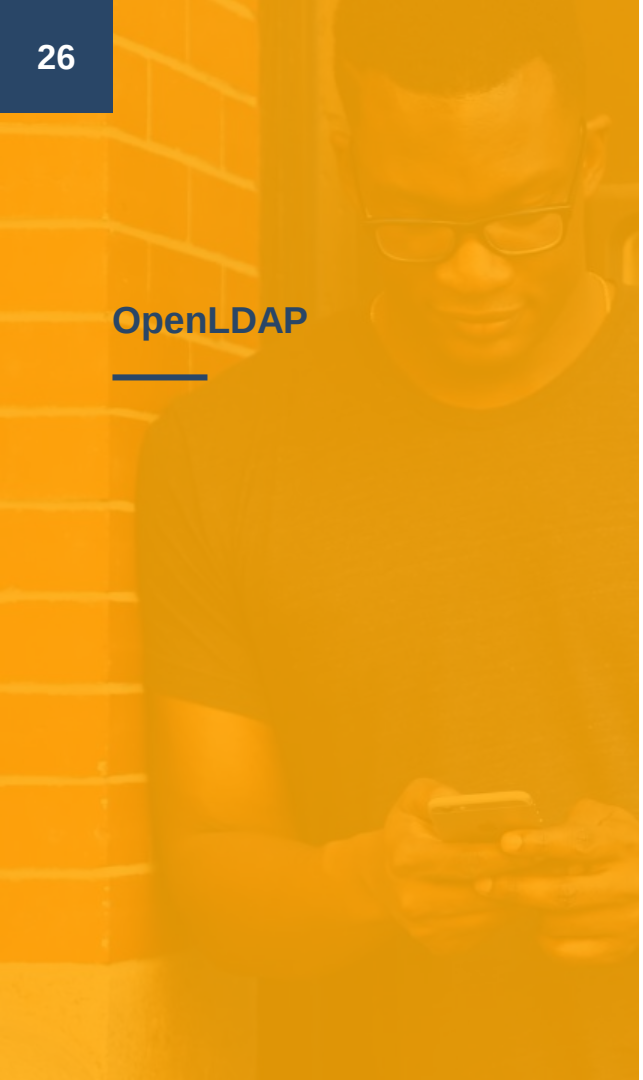
Apache Tomcat est un conteneur libre de servlets et JSP Java EE. Issu du projet Jakarta, Tomcat est un projet principal de la fondation Apache. Il implémente les spécifications des servlets et des JSP du Java Community Process1. Il est paramétrable par des fichiers XML et de propriétés, et inclut des outils pour la configuration et la gestion. Il comporte également un serveur HTTP.

Avant d’installer tomcat, nous devons vérifier si la JDK(java developement kit) est présent sur la machine



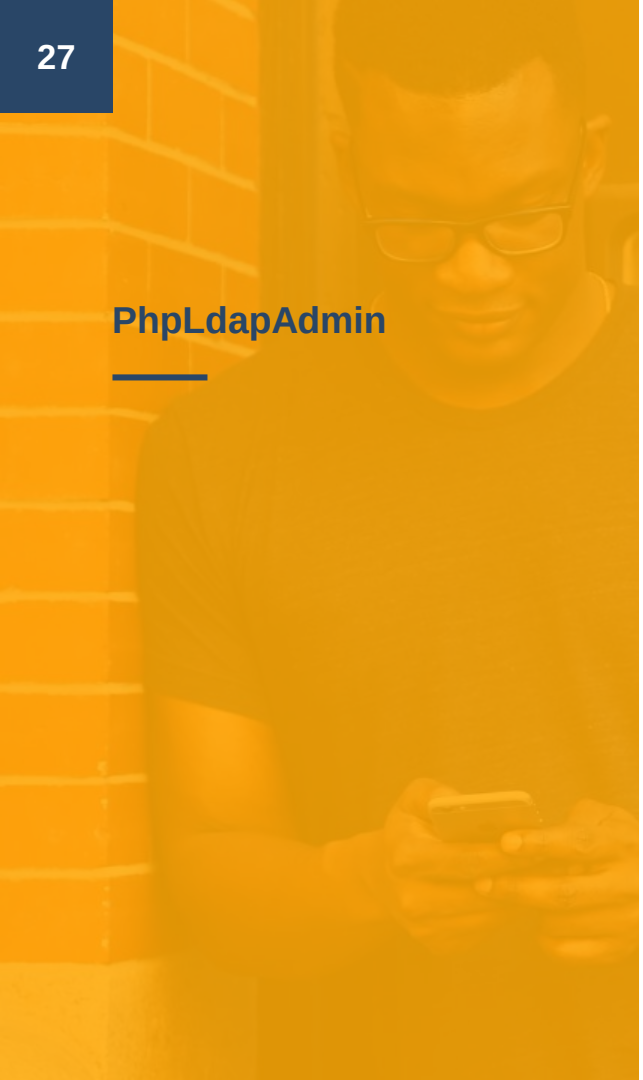
Maven

Maven est indispensable afin de mettre en place CAS. Vu que CAS est un projet développé en Java et Maven est un outil open-source de gestion et d’automatisation de développement Java. Maven est utilisé pour compiler les sources du projet CAS et retourne en sortie un fichier (.war) qui est la servlet de déploiement du serveur CAS.



OpenLDAP

OpenLDAP est une implémentation libre du protocole LDAP développée par The OpenLDAP Project. Nous utilisons la plus récente version qui est OpenLDAP v3.



PhpLdapAdmin

A fin de simplifier l’utilisation de l’annuaire LDAP, nous avons installé PhpLdapAdmin qui est le client web permettant la gestion de l’annuaire LDAP. Il offre une vue hiérarchique sous forme d’arborescences et des fonctions avancées de recherche, qui facilite la navigation et l’administration de l’annuaire.

Merci