2020

Amstutz Thomas

DIVTEC

18/11/2020

Annuaire LDAP



**Table des matières**

[1 Introduction 1](#_Toc56584614)

[2 Annuaire LDAP 2](#_Toc56584615)

[2.1 Introduction 2](#_Toc56584616)

[2.2 Description 2](#_Toc56584617)

[2.2.1 Exemple d’annuaires LDAP 2](#_Toc56584618)

[2.3 Structure 2](#_Toc56584619)

[2.4 Différences avec une base de données 4](#_Toc56584620)

[2.5 Avantages d’un annuaire LDAP 4](#_Toc56584621)

[2.6 Les infrastructures 5](#_Toc56584622)

[2.6.1 L’infrastructure client-serveur 5](#_Toc56584623)

[2.6.2 L’infrastructure N-Tier 6](#_Toc56584624)

[3 Application Web 7](#_Toc56584625)

[3.1 Informations sur la machine virtuelle 7](#_Toc56584626)

[3.2 Informations sur l’Active Directory 7](#_Toc56584627)

[3.3 Marche à suivre 8](#_Toc56584628)

[3.3.1 Installations 8](#_Toc56584629)

[3.4 Connexion 8](#_Toc56584630)

[3.5 Création d’utilisateur 10](#_Toc56584631)

[3.6 Affichage des informations 11](#_Toc56584632)

[3.6.1 Déconnexion 12](#_Toc56584633)

[4 Sources 13](#_Toc56584634)

# Introduction

Dans le cadre du module 159 sur les annuaires LDAP, nous devons réaliser un document parlant du fonctionnement d’un annuaire LDAP ainsi qu’expliquer comment les utiliser.

Une première partie de ce document se concentre sur les informations générales sur les annuaires LDAP, la deuxième montrera comment concrètement utiliser un annuaire à l’aide d’un exemple pratique.

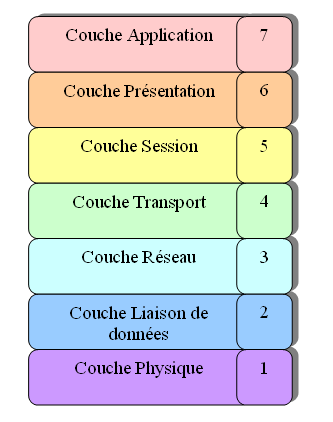
# Annuaire LDAP

## Introduction

Un annuaire LDAP centralise des données concernant des utilisateurs afin de se connecter sur plusieurs machines ou applications avec un seul compte.

Si une entreprise ne possède pas d’annuaire LDAP (peut-importe que ça soit un Active Directory ou un autre annuaire), toutes les informations de connexion serait de faible qualité et surtout très peu sécurisées (chaque utilisateur réutiliserait le même mot de passe partout).

## Description



LDAP

LDAP, pour Lightweight Directory Access Protocol (Protocole d’accès aux annuaires léger) est un protocole qui permet d’interroger et de modifier des services d’annuaire (comme Active Directory par exemple). Il repose sur TCP/IP et utilise la couche application du modèle OSI.

LDAP est surtout utilisé pour gérer des utilisateurs et des ordinateurs (comme à la DIVTEC), mais il pourrait également être utilisé pour gérer du matériel. Cette utilisation fait qu’un annuaire LDAP est surtout utilisé en lecture.

### Exemple d’annuaires LDAP

* Active Directory de Microsoft
* Apache Directory Server
* OpenLDAP

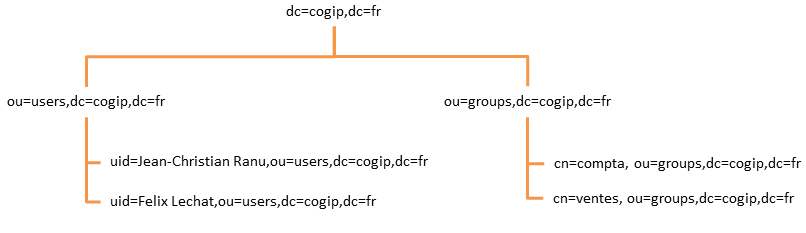
## Structure

Ce protocole se présente sous forme d’arbre qui est constitué de nœud (appelé *entrée*), il en existe trois types :

* La racine (root), qui est le premier nœud de l’arbre
* La branche (branch), entrée intermédiaire entre la racine et une feuille
* La feuille (leaf), entrée en fin d’arborescence qui ne comporte pas de descendance

Chaque entrée possède un Distinguished Name (DN), il permet de la situer dans l’arborescence de l’arbre, par exemple, le DN de l’utilisateur Felix Lechat est : "uid=Felix Lechat,ou=users,dc=cogip,de=fr"

Chaque entrée a aussi un RDN (Relative Distinguished Name). Le RDN est la partie du DN de l'élément qui est relative au DN supérieur. Le RDN d'un élément ne permet pas de l'identifier de manière unique dans l'annuaire, par exemple, le RDN de l’utilisateur Felix Lechat est : "uid=Felix Lechat"



[Hiérarchisation des données](https://openclassrooms.com/fr/courses/2257706-presentation-du-concept-dannuaire-ldap/2260321-modele-de-nommage)

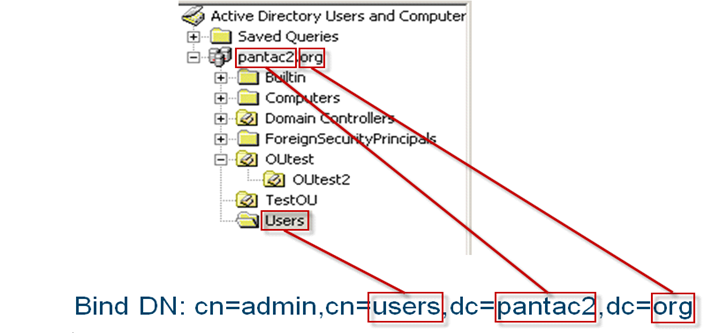
🡨 Racine

🡨 Branche

🡨 Feuille

Une entrée est constituée d'un ensemble d'attributs. Un **attribut** possède un nom, un type et une ou plusieurs valeurs. Les attributs sont définis dans des schémas et est l'une des caractéristiques de cet élément.

Dans notre exemple ci-dessus, l'arbre se découpe après la racine en deux "*ou*" (*Organisational Units*) qui constituent deux branchements : "*users*" et "*groups*", dans lesquels nous trouvons ensuite les entrées : les utilisateurs et les groupes.



Hiérarchisation des données sous Windows

## Différences avec une base de données

Tous les annuaires sont des bases de données mais toutes les bases de données ne sont pas des annuaires. Les annuaires LDAP sont un type spécial de base de données, il y a plusieurs différences entre les deux que voici :

1. **On lit plus souvent** un annuaire qu’on ne le met à jour. Contrairement à un SGBD, un annuaire n’est pas fait pour stocker des informations constamment en mouvement. Il est logique de le structurer différemment et d’organiser les données de manière arborescente (sur un SGBD, la structuration est relationnelle).
2. Un annuaire fournit**une méthode de consultation standardisée**. Le SQL est, standardisé, mais chaque SGBD (Oracle, SQLServer, MySQL, PostgreSQL ...) a sa propre couche de connexion et ses propres fonctions.
3. La norme**LDAP définit le modèle de données**, alors qu'un SGBD varie (quoi, où et comment). Avec les schémas, un annuaire fournit des modèles de données officialisés, ce qui permet d'être interopérable.

## Avantages d’un annuaire LDAP

S’il n'y a pas d'annuaire pour centraliser les utilisateurs, il faut un mot de passe pour chaque personne sur chaque ordinateur, les utilisateurs utiliseraient donc toujours le même mot de passe ; C'est pour ça que la mise en place d'un système LDAP est nécessaire

Sans LDAP

* Pas de synchronisation
* Pas de contrôle des mots de passe
* Un mot de passe par machine

Avec LDAP

* Centralisation
* Gestion avec GPO
  + Utilisateurs
  + Groupes
  + Ordinateurs

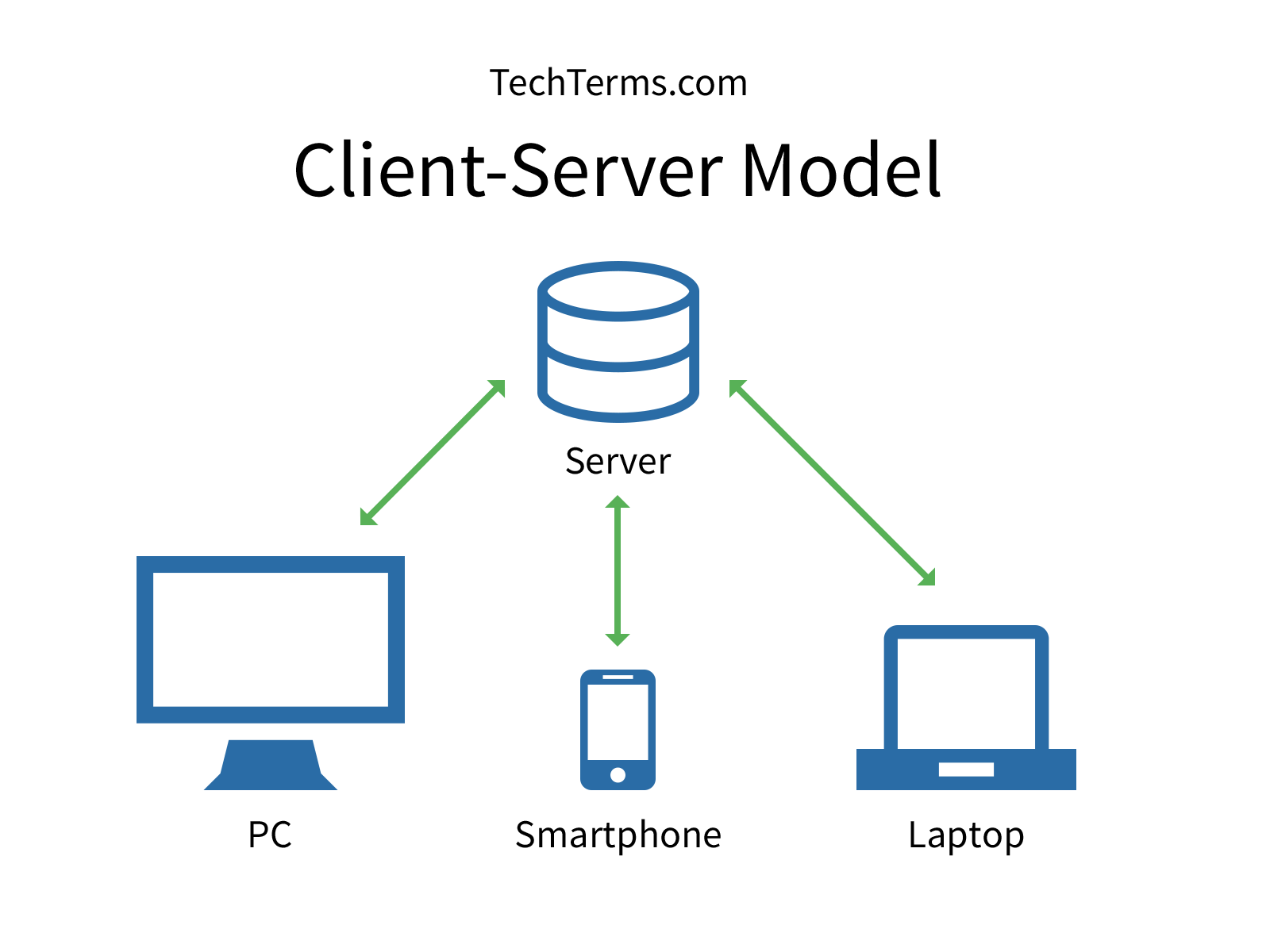
## Les infrastructures

Un annuaire LDAP peut être mis en place dans plusieurs types d’infrastructures, en voici deux :

### L’infrastructure client-serveur

C’est une infrastructure qui est utilisée en local, elle établie une connexion directe entre l’application et le serveur.

Cette infrastructure pose un problème car les applications doivent être installée, configurée et mise à jour sur chaque poste.



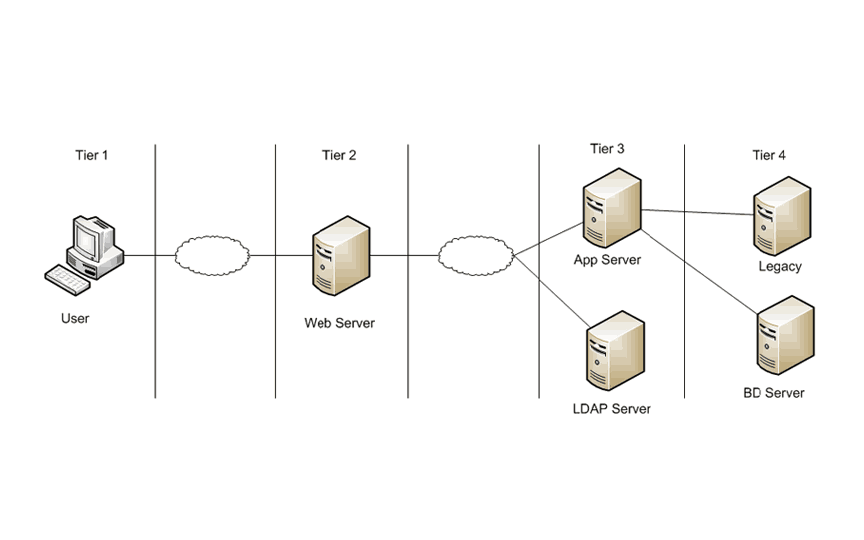
Exemple d’infrastructure client-serveur

* Avantage : Facile à mettre en place
* Désavantage : Difficile à entretenir

### L’infrastructure N-Tier

Au lieu d’installer une application sur chaque poste, elle sera centralisée sur un serveur, il fera le lien entre l’annuaire, les utilisateurs et une base de données.

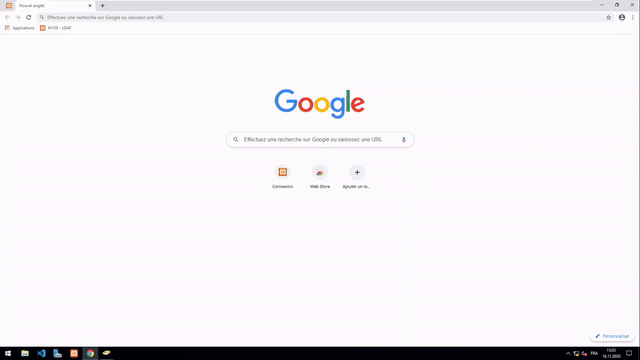
Exemple d'infrastructure N-Tier



* Avantage : Un seul endroit à configurer et à entretenir, utilise un navigateur
* Désavantage : Plus compliqué à mettre en place

# Application Web

Ce chapitre contient toutes les informations nécessaires pour créer une application web afin de se créer des utilisateurs et de se connecter avec.

Des morceaux du code sont présentés dans le document mais le meilleur moyen de comprendre le fonctionnement est d’aller voir [le code source sur Github](https://github.com/ThomasAmstutz/PHP-M159-LDAP).

## Informations sur la machine virtuelle

* Nom de la VM : M159 Win SRV2016 App N-Tier
* Nom de la machine : SRV2016-M159
* Processeur : 1 (2 coeurs)
* RAM : 4Go
* Login :
  + Nom d'utilisateur : Administrator
  + Mot de passe : Admlocal1

## Informations sur l’Active Directory

Le service Active Directory doit être installé sur le serveur, voici les paramètres que j’ai utilisé

* Nom du domaine : M159-Domain.local
* Mot de passe : Admlocal1

## Marche à suivre

### Installations

Avant de commencer à coder l’application, il faut installer la liste de logiciels qui suit :

* Installer Windows Server 2016 Standard
* Installer VM Ware Tools
* Installer XAMPP
* Installer un IDE (j'ai choisi VS Code)
* Activer Active Directory

## Connexion

Afin de se connecter avec un utilisateur, il y a plusieurs étapes qui doivent être effectuées :

1. On commence par tester si on est déjà connecté ou pas, si c’est le cas, on redirige l’utilisateur vers la page de son compte sinon on continue

// Vérifie si l'utilisateur est déjà connecté, si oui il est redirigé

if(isset($\_SESSION["loggedin"]) && $\_SESSION["loggedin"] === true){

header("location: account.php");

exit;

}

1. On ouvre une connexion vers le serveur LDAP, en cas de problème une erreur est affichée.

$ldapConnection = setupConnection();

1. On teste si l’utilisateur a juste fourni un nom d’utilisateur ou s’il a inclus le nom du domaine, dans le premier cas on rajoute le nom du domaine, dans le deuxième on ne fait rien.

function convertUsernameToLogin($username) {

// Vérifie le type du nom d'utilisateur

$usernameType = "username";

if (strpos($username, "@") !== false)

$usernameType = "mail";

$loginDomain = "@M159-Domain.local";

// Rajoute le domaine de connexion au nom d'utilisateur

if ($usernameType !== "mail")

$username = $username.$loginDomain;

return $username;

}

1. On teste si l’utilisateur existe, si c’est le cas on continue, sinon on affiche une erreur

if (alreadyExists($connection, $username)) {

// L’utilisateur existe

} else {

echo "Cet utilisateur n'existe pas.";

}

1. On utilise les informations de connexion fournies pour se relier à l’annuaire, en cas de problème une erreur est affichée.

if (@ldap\_bind($connection, $ldap\_username, $ldap\_password)) {

} else

echo "Impossible de se connecter.";

1. On enregistre quel utilisateur est connecté dans le navigateur et on passe à la page qui montre les informations du compte.

// Enregistre les données dans la variable de session

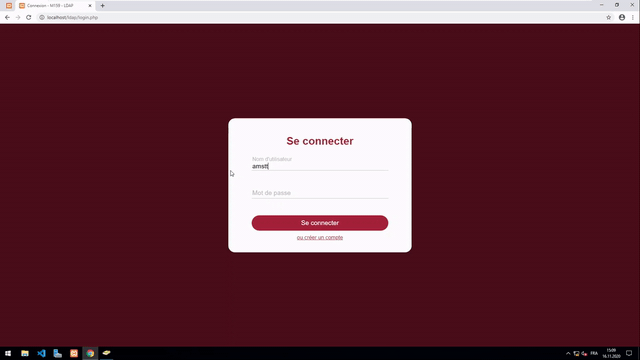
$\_SESSION["loggedin"] = true;

$\_SESSION["username"] = $ldap\_username;

header('Location: account.php');

exit;

Exemple de connexion avec un utilisateur inconnu



## Création d’utilisateur

1. On commence par ouvrir une connexion avec l’annuaire

$ldapConnection = setupConnection();

1. Ensuite on s’authentifie auprès du serveur avec un compte qui peut créer des utilisateurs

$loginDomain = '@M159-Domain.local';

$credentials = getLoginCredentials();

// Connexion avec une identité qui permet les modifications

$bind = @ldap\_bind($connection, $credentials['username'].$loginDomain, $credentials['password']);

1. On prépare les attributs du nouvel utilisateur

// Prépare les données

$info["cn"] = "$username";

$info["givenname"] = "$name";

$info["sn"] = "$lastname";

$info["sAMAccountName"] = "$username";

$info["displayname"] = "$name $lastname";

$info["userprincipalname"] = "$username$loginDomain";

$info["UserAccountControl"] = "544";

$info["objectclass"] = "user";

$info["userpassword"] = "$password";

1. On teste si l’utilisateur souhaité existe déjà

if (!alreadyExists($connection, $info["cn"])) {

// L’utilisateur n’existe pas encore

} else {

echo "Cet utilisateur existe déjà, connectez-vous.";

}

1. On ajoute l’utilisateur et vérifie s’il y a des erreurs

if(@ldap\_add($connection, "cn=".$info["cn"].",ou=utilisateurs,DC=M159-Domain,DC=local", $info)) {

echo "Utilisateur ajouté.";

} else {

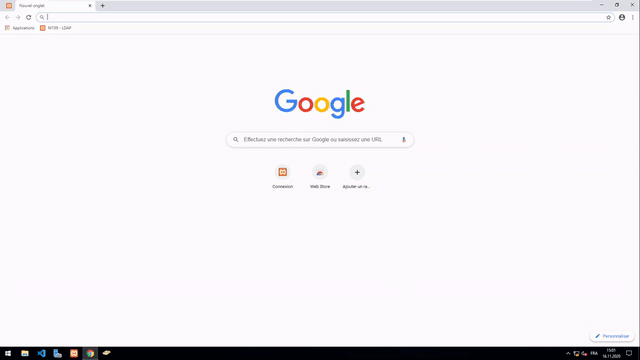
echo "Imposible de créer l'utilisateur.";

}

1. On ferme la connexion

@ldap\_close($connection);

Tentative d'ajout d'un utilisateur existant



## Affichage des informations

Lorsqu’un utilisateur est connecté, il est redirigé vers une page lui indiquant toutes les informations de son compte, il y a aussi un bouton pour se déconnecter.

1. Au chargement de la page, on teste si l’utilisateur est connecté, si c’est le cas on affiche la page, sinon on le redirige vers la page de connexion

// On vérifie si l'utilisateur est connecté, si ce n'est pas le cas il est redirigé vers la page de connexion

if(!isset($\_SESSION["loggedin"]) || $\_SESSION["loggedin"] !== true){

header("location: login.php");

exit;

}

1. On va ensuite se connecter avec un compte admin et lister tous les utilisateurs

$credentials = getLoginCredentials();

if (TRUE !== @ldap\_bind($connection, $credentials['username'] . '@M159-Domain.local', $credentials['password'])){

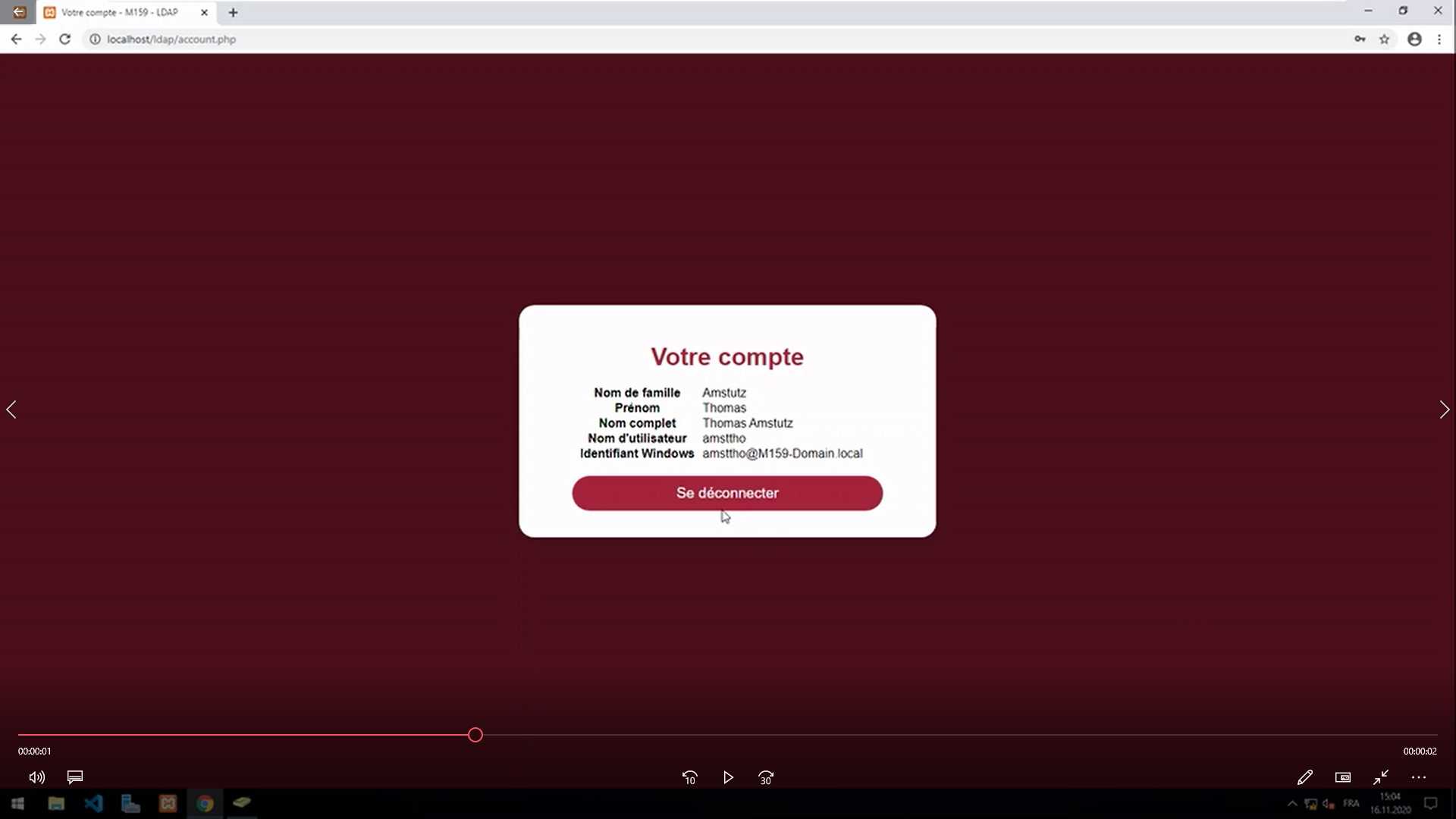
die('Impossible de se connecter au serveur LDAP. ');

}

$ldap\_base\_dn = "ou=utilisateurs,DC=M159-Domain,DC=local";

$search\_filter = "(&(objectCategory=user))";

$result = @ldap\_search($connection, $ldap\_base\_dn, $search\_filter);

1. On termine par afficher les données de l’utilisateur dans un tableau à deux colonnes

### Déconnexion

Lorsqu’un utilisateur clique sur le bouton « Se déconnecter », la variable de session est réinitialisée ce qui efface sa connexion sur ce navigateur, il est ensuite redirigé vers la page de connexion.

// Initialise la session

session\_start();

// Vide la variable de session

$\_SESSION = array();

// Détruit la session

session\_destroy();

// Redirige à la page de connexion

header("location: index.php");

exit;

# Sources

Tout ce qui m’a aidé à réaliser mon projet :

* <https://www.php.net/manual/fr/ref.ldap.php>
* <https://www.youtube.com/watch?v=AEjGhzZpGlg>
* <https://stackoverflow.com/questions/20119137/open-a-new-link-in-an-if-statement-with-php>
* <https://support.centrestack.com/hc/en-us/articles/360040212313-Configuring-the-Local-Active-Directory-Connection-for-LDAPS>
* <https://www.tutorialrepublic.com/php-tutorial/php-mysql-login-system.php>
* <https://stackoverflow.com/questions/14351602/php-ldap-get-user-attributes-including-associated-groups>