|  |  |
| --- | --- |
|  | **Partage de fichiers WebDAV** |

# WebDAV sur Apache

## Présentation

WebDAV (*Web-based Distributed Authoring and Versioning*) est une **extension du protocole HTTP**.

WebDAV permet de **simplifier la gestion de fichiers avec des serveurs distants**.   
Il permet de **récupérer**, **déposer**, **synchroniser** et **publier** **des fichiers** (et dossiers) rapidement et facilement.

L'objectif principal de WebDAV est de **rendre possible l'écriture à travers le web** et **pas seulement la lecture** de données. WebDAV permet à **plusieurs utilisateurs d'éditer le contenu d'un dossier web simultanément**.

Il saura **gérer les droits d'accès aux fichiers** (ou dossiers), en **verrouillant momentanément les fichiers et dossiers édités**.

Par exemple …

* Il est possible de **partager des documents texte** **ou des chiffriers** avec ses collègues en les téléchargeant sur le serveur WebDAV.
* Un **utilisateur peut aussi être capable de partager sa collection de musique** avec ses proches en leur donnant simplement une URL. Tout cela peut être **réalisé sans qu'ils n'installent quoi que ce soit**.

Il existe plusieurs façons de gérer les fichiers sur un serveur distant.

WebDAV présente **plusieurs avantages par rapport à d'autres solutions telles que FTP ou Samba**.

### Utilisation de WebDAV

WebDAV offre plusieurs avantages ...

* **Intégration native sur tous les principaux systèmes d'exploitation** (Windows, Mac, Linux) :   
  Il n'est pas nécessaire d'installer un logiciel tiers pour utiliser WebDAV ;
* **Prise en charge des transferts partiels** ;
* **Plus de choix pour l'authentification** :   
  Être sur HTTP signifie que **NTLM**, **Kerberos**, **LDAP**, ... sont tous possibles.

Selon sa situation, WebDAV peut être la meilleure solution pour ses besoins.

## Mise en place de WebDAV

Avant de commencer, il est préférable de créer d'abord un utilisateur avec un accès sudo.

Il est toujours possible d'exécuter des commandes en tant que root, mais cela n'est pas recommandé en raison de problèmes de sécurité.

**Création d'un utilisateur**

Pour ajouter un utilisateur ....

**>>** sudo adduser maisonneuve

**Attribution des privilèges sudo à l'utilisateur maisonneuve**

Après avoir créé un nouvel utilisateur, l'étape suivante consiste à accorder à l'utilisateur des privilèges **sudo**.

En supposant que l'usager en cours est root, il suffit d'ajouter l'utilisateur **maisonneuve** au groupe **sudo** avec la commande suivante ...

**>>** sudo usermod -aG sudo maisonneuve

Les utilisateurs du groupe sudo bénéficient de privilèges sudo.

Il est maintenant possible de se déconnecter et ouvrir une session en tant que l'utilisateur maisonneuve.

**Remarque** …  
Les prochaines commandes doivent être effectuées en tant que sudo.

**>>** sudo -i

**Étape 01 – Installation d'Apache**

Pour installer le serveur web ...

**>>** sudo apt update && apt upgrade -y && sudo apt autoremove -y

**>>** sudo apt install apache2

Le serveur Web Apache doit être installé et en cours d'exécution.

**Épuration du fichier de configuration du module de base du serveur Apache**

Pour épurer le fichier de configuration 000-default.conf …

**>>** sudo mv /etc/apache2/sites-available/000-default.conf /etc/apache2/sites-available/000-default.conf.original

**>>** sudo cat /etc/apache2/sites-available/000-default.conf.original | egrep -v -e '^[[:blank:]]\*#|^$' | sudo tee /etc/apache2/sites-available/000-default.conf

**Préparation du répertoire**

Il faut par la suite désigner un répertoire pour WebDAV.

Il faudra créer le nouveau répertoire /var/www/webdav pour cela.

Il faudra également modifier le propriétaire de www-data (l'utilisateur Apache) afin de permettre à Apache d'y écrire.

**>>** sudo mkdir --verbose /var/www/webdav

**>>** sudo chown --verbose --recursive www-data:www-data /var/www/

**Remarque** …  
L’option longue --reversive est -R comme option courte.

**Activation des modules**

Les modules WebDAV doivent être activés en en utilisant la commande a2enmod ...

**>>** sudo a2enmod dav

**>>** sudo a2enmod dav\_fs

**Rappel** …

Les modules Apache se trouvent sous **/etc/apache2/mods-available**.

Cela crée un lien symbolique depuis **/etc/apache2/mods-available** vers **/etc/apache2/mods-enabled**.

**Étape 02 – Configuration de WebDAV**

**Modification du fichier de configuration de site**

Dans un premier temps, il faut modifier le fichier de configuration dans /etc/apache2/sites-available/000-default.conf en utilisant l'éditeur de texte nano.

**>>** sudo nano /etc/apache2/sites-available/000-default.conf

Sur la première ligne, il faut ajoute la configuration de la directive DavLockDB ...  
**DavLockDB /var/www/DavLock**

Et les directives Alias ​​et Directory dans la section VirtualHost ...

**Alias ​​/webdav /var/www/webdav**

**<Directory /var/www/webdav>**

**DAV On**

**</Directory>**

Le fichier devrait ressembler à ceci après l'édition …  
**DavLockDB /var/www/DavLock**

**<VirtualHost \*:80>**

**ServerAdmin webmaster@localhost**

**DocumentRoot /var/www/html**

**Alias /webdav /var/www/webdav**

**<Directory /var/www/webdav>**

**DAV On**

**</Directory>**

**ErrorLog ${APACHE\_LOG\_DIR}/error.log**

**CustomLog ${APACHE\_LOG\_DIR}/access.log combined**

**</VirtualHost>**

**Quelques précisions** …

* La **directive DavLockDB désigne le nom de la base de données DAV Lock**.   
  Cela est le **chemin vers un fichier**.   
  Le fichier n'a pas besoin d'être créé.   
  Le **répertoire toutefois se doit d’être accessible en écriture** par le serveur Apache.
* La **directive Alias** ​​établit un **lien pour les requêtes à http://<adresse\_du\_serveur>/webdav** dans le répertoire /var/www/webdav.
* La **directive Directory** indique à Apache d'**activer** **WebDAV pour le répertoire** **/var/www/webdav**.

Pour le moment, le serveur Apache, le serveur WebDAV fonctionnera sans authentification.

En dernier lieu il faut redémarrer le serveur Apache ...

**>>** sudo systemctl restart apache2.service

**Phase de test**

WebDAV sans authentification autorise uniquement l'accès en lecture pour tous les utilisateurs.

La commande suivante va créer un fichier texte ...  
**>>** sudo echo "Ceci est un premier exemple de fichier texte" | sudo tee --append /var/www/webdav/test.txt

Un fichier texte appelé test.txt sera créé dans le répertoire /var/www/webdav.   
Il devrait contenir le texte ... **Ceci est un premier exemple de fichier texte**.

À partir de ce point, il est possible de se connecter à partir d'un hôte externe.   
Le serveur WebDAV doit être trouvé sur **http://<adresse\_du\_serveur>/webdav**.

**Remarque** …  
Pour télécharger le fichier, on clique sur le fichier à l’aide du bouton droit de la souris et on sélectionne **Enregistrer le lien sous …**

**Étape 03 - Ajout de l'authentification**

Un **serveur WebDAV sans authentification n'est pas sécurisé**.   
Dans ce qui suit, l'authentification au serveur WebDAV sera possible en utilisant le schéma d'authentification Digest.

**Authentification de base / Authentification Digest**

Il existe de nombreux **schémas d'authentification disponibles**.

Ce tableau illustre la compatibilité des différents schémas d'authentification sur différents systèmes d'exploitation. Notez que si on utilise **HTTPS**, on suppose que le **certificat SSL est valide** (non auto-signé).

|  |  |  |
| --- | --- | --- |
|  | HTTP | HTTPS |
| Authentification de base | MacOS et Linux | MacOS, Linux et Windows |
| Authentification Digest | MacOS, Linux et Windows | MacOS, Linux et Windows |

**Compatibilité WebDAV**

Avec le **protocole HTTP**, il est **préférable d'utiliser l'authentification Digest**, car elle **fonctionnera sur tous les systèmes d'exploitation**.   
Si on utilise le **protocole HTTPS**, on aura la **possibilité d'utiliser l'authentification de base**.

L'authentification Digest fonctionne sur tous les systèmes d'exploitation sans avoir besoin d'un certificat SSL.

**Authentification Digest**

Dans un premier temps il faut **générer le fichier** (appelé users.password) qui **emmagasine les mots de passe pour les utilisateurs**.   
Dans l'authentification Digest, il existe le **champ domaine qui agit comme un espace de noms pour les utilisateurs**.

Pour cet exemple, le premier utilisateur s'appellera maisonneuve.

Pour générer le fichier digest, il faut **installer les dépendances pour Apache** …

**>>** sudo apt install apache2-utils

**Ajout des utilisateurs**

On ajoute des utilisateurs par la suite.

Pour **générer le fichier de mot de passe de l'utilisateur** ...

**>>** sudo htdigest -c /etc/apache2/users.password webdav maisonneuve

Cela ajoute l'utilisateur maisonneuve au fichier de mot de passe.

La **saisie d'un mot de passe est demandée** afin de créer le mot de passe pour l'utilisateur maisonneuve.

**Remarque** …  
Pour l'ajout ultérieur d'utilisateurs, il faudra **omettre le commutateur -c car le fichier de mots de passe est déjà créé**.

Voici un autre exemple d'ajout d'un utilisateur nommé hochelaga …

**>>** sudo htdigest /etc/apache2/users.password webdav hochelaga

il faudra par la suite **permettre à Apache de lire le fichier de mot de passe**. Il faut donc en modifier le propriétaire.

**>>** sudo chown --verbose www-data:www-data /etc/apache2/users.password

Une fois le fichier de mot de passe créé, on doit apporter les **modifications qui suivent à la configuration dans /etc/apache2/sites-available/000-default.conf**.

**>>** sudo nano /etc/apache2/sites-available/000-default.conf

Ajout des lignes suivantes à la directive Directory …

**AuthType Digest**

**AuthName "webdav"**

**AuthUserFile /etc/apache2/users.password**

**Require valid-user**

La version finale devrait ressembler à ceci (avec les commentaires supprimés).

**DavLockDB /var/www/DavLock**

**<VirtualHost \*:80>**

**ServerAdmin webmaster@localhost**

**DocumentRoot /var/www/html**

**Alias /webdav /var/www/webdav**

**<Directory /var/www/webdav>**

**DAV On**

**AuthType Digest**

**AuthName "webdav"**

**AuthUserFile /etc/apache2/users.password**

**Require valid-user**

**</Directory>**

**ErrorLog ${APACHE\_LOG\_DIR}/error.log**

**CustomLog ${APACHE\_LOG\_DIR}/access.log combined**

**</VirtualHost>**

**Quelques précisions** …

* Le **module mod\_authn** contient les **définitions des directives d'authentification** ;
* La **directive AuthType** indique à Apache que pour le répertoire /var/www/webdav, il devrait y avoir une **authentification utilisant le schéma Digest**.
* L'**authentification** **Digest** nécessite une **valeur pour le paramètre realm (AuthName)** défini comme webdav.   
  Le **paramètre realm agit comme un espace de noms**.   
  Lorsque l'on a des utilisateurs portant le même nom, il est possible de les **différencier en utilisant des valeurs différentes pour le répertoire**.   
  L'utilisation de la **directive AuthName permet de définir la valeur du répertoire**.
* La **directive AuthUserFile** est utilisée afin d'indiquer l'emplacement du fichier de mot de passe.
* La directive Require indique que seuls les utilisateurs valides qui s'authentifient peuvent accéder à ce répertoire.

Enfin, il faut activer le module Digest et redémarrer le serveur pour que les paramètres prennent effet.

**>>** sudo a2enmod auth\_digest

**>>** sudo systemctl restart apache2.service

**Phase de test**

Il ne reste qu’à vérifier si l’authentification Digest fonctionne.

La commande suivante va créer un fichier texte ...

**>>** echo "Ceci est un second exemple de fichier texte" | sudo tee --append /var/www/webdav/test2.txt

Un fichier texte appelé test2.txt sera créé dans /var/www/webdav.

Il devrait contenir le texte ... Ceci est un second exemple de fichier texte.

À partir de ce point, il est possible de se connecter à partir d'un hôte externe.

Le serveur WebDAV doit être trouvé sur **http://<adresse\_du\_serveur>/webdav**.

# Annexe 01 Authentification de base / Authentification Digest

Il existe de nombreux **schémas d'authentification disponibles**.

|  |  |  |
| --- | --- | --- |
|  | HTTP | HTTPS |
| Authentification de base | MacOS et Linux | MacOS, Linux et Windows |
| Authentification Digest | MacOS, Linux et Windows | MacOS, Linux et Windows |

**Méthode Basic**

Cette méthode est la plus simple, mais également la moins sécurisée car elle **transmet le mot de passe codé en base 64 qui se décode aisément**.   
Elle **n'est recommandée qu'avec une connexion chiffrée** (protocole HTTPS).

Le serveur ne recevant pas d'en-tête d'identification correcte envoie ce genre d'en-tête HTTP …  
**WWW-Authenticate:Basic realm="Tux"**

Le serveur indique la méthode requise (Basic), suivie des paramètres.   
La méthode Basic **ne** **requiert que le paramètre realm identifiant le domaine de protection**.

Le client HTTP peut alors réessayer la requête en spécifiant l'**en-tête HTTP Authorization**.   
Celui-ci doit **contenir la méthode utilisée (Basic) suivie de la représentation en Base64 du nom de l'utilisateur et du mot de passe** séparés par deux-points ( : ).

Par exemple, pour authentifier l'utilisateur Tux avec le mot de passe qwerty.12345, le client envoie …  
**Authorization: Basic QWxhZGRpbjpvcGVuIHNlc2FtZQ==**

QWxhZGRpbjpvcGVuIHNlc2FtZQ== étant la représentation en Base64 du texte tux:qwerty12345.

## Références

<https://www.digitalocean.com/community/tutorials/how-to-configure-webdav-access-with-apache-on-ubuntu-14-04>

<https://www.server-world.info/en/note?os=Debian_10&p=httpd&f=12>

## Exercice

Question 1  
**Comment vérifier le bon fonctionnement d’HTTP WebDAV sur une station Linux en ligne de commande ?**

|  |
| --- |
|  |
|  |
|  |
|  |

Question   
**Comment vérifier le bon fonctionnement d’HTTP WebDAV sur une station Linux à l’aide d’un interface graphique ?**

|  |
| --- |
|  |
|  |
|  |
|  |

Question 3  
**Comment vérifier le fonctionnement d’HTTP WebDAV sur une station Windows ?**

|  |
| --- |
|  |
|  |
|  |
|  |

Question 4  
**Comment relier (mapper) un lecteur Windows avec un partage HTTP WebDAV ?**

|  |
| --- |
|  |
|  |
|  |
|  |