

# Serveur Subversion

Debian GNU/Linux



Matthieu Vogelweith

10 juillet 2008

# Résumé

L'objectif de cette page est de détailler l'installation d'un serveur Subversion [1].

Ce document a été rédigé en LaTeX en utilisant l'excellent Vim sous Debian GNU/Linux. Il est disponible aux formats XHTML et PDF. Les sources LaTeX sont disponibles ici : [L<sup>A</sup>T<sub>E</sub>X](#)

# Licence

Copyright ©2008 Matthieu VOGELWEITH <matthieu@vogelweith.com>.

Vous avez le droit de copier, distribuer et/ou modifier ce document selon les termes de la GNU Free Documentation License, Version 1.2 ou ultérieure publiée par la Free Software Foundation ; avec aucune section inaltérable, aucun texte de première page de couverture, et aucun texte de dernière page de couverture. Une copie de la licence est disponible dans la page [GNU Free Documentation License](#).

# Table des matières

<b>Table des matières</b>	<b>3</b>
<b>1 Présentation</b>	<b>4</b>
1.1 Subversion vs CVS . . . . .	4
<b>2 Installation de SubVersion</b>	<b>5</b>
<b>3 Administration du dépôt</b>	<b>6</b>
3.1 Création des dépôts . . . . .	6
3.2 Premier import . . . . .	6
3.3 Configuration d'Apache . . . . .	6
<b>4 Installation de WebSVN</b>	<b>8</b>
<b>5 Les clients disponibles</b>	<b>9</b>
5.1 Pour Linux . . . . .	9
5.2 Pour windows . . . . .	9
<b>6 Utilisation</b>	<b>10</b>
6.1 Principe . . . . .	10
6.2 Mise à jour de la ressource locale . . . . .	10
6.3 Tests . . . . .	10
6.4 Validation des modifications . . . . .	10
6.5 Visualisation des modifications . . . . .	10
<b>7 Références</b>	<b>11</b>

# Chapitre 1

## Présentation

- principe
- avantages
- différences par rapport aux autres

### 1.1 Subversion vs CVS

Subversion (SVN) est un système de contrôle des versions OpenSource distribué sous une licence compatible Apache/BSD. Cet outil en développement depuis plus de 3 ans propose un grand nombre de nouveautés par rapport à son aîné vieillissant qu'est CVS :

- Les répertoires, renommages et les propriétés des fichiers sont versionnés
- Validation des commit uniquement lorsqu'ils se terminent correctement dans leur globalité
- Apache comme serveur réseau, WebDAV/DeltaV comme protocole
  - Authentification
  - Autorisation d'accès basique
  - Compression à la volée
  - Parcours du référentiel
- Architecture nativement client/serveur.
- Gestion des branches et tags par copie
- Manipulation efficace des fichiers binaires
- Sorties interprétable donc utilisable par des scripts
- Les coûts sont proportionnels à la taille des changements et non des données
- Gestion des branches et des tags simplifiée

Subversion semble donc plus adapté que CVS, notamment au niveau de l'authentification (Apache donc PAM, NIS, ...) et de la gestion des versions sur l'arborescence du dépôt.

## Chapitre 2

# Installation de SubVersion

```
# aptitude install subversion libapache2-svn subversion-tools
```

- Version standalone et version module d'apache
- Plutot en module d'apache, car
- permet de bénéficier des méthodes d'authentification et de sécurité d'apache
- evite d'ouvrir un port supplémentaire sur le firewall

## Chapitre 3

# Administration du dépôt

- création du dépôt
- fonctionnement des branches et des tags
- cycle de développement habituel

### 3.1 Création des dépôts

Les dépôts doivent être créés, obligatoirement dans le système de fichier local, avec les commandes suivantes :

```
# mkdir /var/lib/svn
# svnadmin create /var/lib/svn/projet1
# chown -R www-data:www-data /var/lib/svn
# chmod -R 700 /var/lib/svn
```

Dans l'exemple ci-dessus, un dépôt a été créé : **projet1**. Il est important de bien donner les droits à l'utilisateur virtuel d'Apache (www-data) pour que le depot puisse être accessible par le Web.

Attention, maintenant que les dépôts sont créés, aucun fichier ne doit être créé, modifié ou supprimé manuellement dans le dépôt. Tout doit se faire avec les commandes fournies par Subversion.

### 3.2 Premier import

Respect de l'architecture préconisée : trunk, tags, branches.

```
# rm -rf /tmp/projet1
# mkdir -p /tmp/projet1/trunk
# mkdir -p /tmp/projet1/tags
# mkdir -p /tmp/projet1/branches
# svn import /tmp/projet1/ file:///var/lib/svn/projet1 -m "Creation initiale de l'
arborescence"
```

### 3.3 Configuration d'Apache

- config du module apache

- config du repertoire du depot
- - Attention : il faut indiquer le rep contenant les différents depots et non le rep d'un depot
- config de l'authentification
- test dans navigateur

```
# Virtual host pour la gestion du depot SVN
<VirtualHost example.org>

    # Config generale du Vhost
    Servername svn.example.org
    ServerAdmin webmaster@example.org

    # Droits d'accès au repertoire
    <Directory /var/lib/svn/>
        Options Indexes FollowSymLinks MultiViews
        AllowOverride None
        Order allow,deny
        allow from all
    </Directory>

    # Acces URL
    <Location />
        DAV svn
        # Repertoire contenant les depots SVN
        SVNParentPath /var/lib/svn
        AuthType Basic
        AuthName "Subversion Repository"

        # Authentification sur l'annuaire LDAP
        # AuthLDAPEnabled on
        # AuthLDAPAuthoritative on
        # AuthLDAPURL ldap://ldap.example.org/ou=people,dc=example,dc=org?login
        # require valid-user

        # Ajout d'une feuille de style
        # SVNIndexXSLT "/styles/svnindex.xsl"
    </Location>
</VirtualHost>
```

- ATTENTION s'il y a des errordocuments Perso !!!

## Chapitre 4

# Installation de WebSVN

- Visualisation du dépôt

```
# aptitude install websvn
```

- indiquer le chemin du répertoire parent des dépôts lors de la configuration.
- supprimer le lien de `/var/www/websvn`.
- Ajouter l'alias qui va bien - éditer la config dans le fichier `/etc/websvn/config.inc`

## Chapitre 5

# Les clients disponibles

Plusieurs clients SVN sont disponibles sur les plateformes UNIX et Windows. Dans les deux cas, il est toujours possible de visualiser le contenu du dépôt ainsi que les modifications et les commentaires de chaque fichier/dossier via un navigateur Web.

### 5.1 Pour Linux

- svn en ligne de commande
- Navigateur Web
- RapidSVN
- esvn
- Eclipse + PHPEclipse + Subclipse

### 5.2 Pour windows

- Navigateur Web
- Explorateur Windows avec TortoiseSVN
- Eclipse + PHPEclipse + Subclipse
- RapidSVN

## Chapitre 6

# Utilisation

Ce chapitre décrit rapidement le fonctionnement de Subversion pour une utilisation de tous les jours.

### 6.1 Principe

- Mise à jour de la ressource locale avec les sources du serveur subversion
- Modifications
- Tests sur le serveur de tests
- Validation des modifications sur le serveur subversion

### 6.2 Mise à jour de la ressource locale

```
$ svn update
```

### 6.3 Tests

```
http://serveur_de_test/ma_page.php
```

### 6.4 Validation des modifications

Lorsque les tests sont réussis, en fin de journée ou lorsque les modifications sont validées et considérées comme correctes, on peut mettre à jour le code sur le serveur Subversion. Cette opération est réalisée simplement par la commande **commit** :

```
$ svn commit
```

### 6.5 Visualisation des modifications

Maintenant que les modifications sont effectives sur le serveur Subversion, on peut facilement visualiser l'historique des modifications via un navigateur Web :

```
http://serveur_subversion/svn/nom_du_depot
```

## Chapitre 7

# Références

- [1] Site officiel de subversion. [subversion.tigris.org](http://subversion.tigris.org).
- [2] Site officiel du projet debian. [www.debian.org](http://www.debian.org).
- [3] Documentation de subversion. [svnbook.red-bean.com](http://svnbook.red-bean.com).