# OpenVPN

Mise en œuvre d'un réseau privé virtuel

**ANATOLE BILLET**

20 novembre 2015
Version 1.00

# OpenVPN

Mise en œuvre d'un réseau privé virtuel

## Avant de commencer…

### Prérequis :

- Un serveur sous Debian
- Un poste client sous Linux ou Windows

### Objectif :

L'objectif de ce tp est de pouvoir mettre en œuvre un VPN de type Bridged, de configurer un serveur OpenVPN et de bien configurer les clients VPN.

## Code couleur :

**-Bleu pour les commandes Debian**

**-Vert pour les noms des fichiers de configurations**

*-Italic pour les descriptions et anecdotes.*

## Sommaire

# 1. OpenVPN

Les ports utilisés par OpenVPN sont les ports TCP et UDP 1994.

Il existe 2 types de VPN le type « Routed » pour mettre en relation de machine distante par internet ou en PPP et le type « Bridged »pour mettre en relation différents réseaux.

Nous mettrons ici en œuvre la seconde option.

# 2. Construction d'une PKI

La PKI fonctionne avec une clef privé et une clef publique ainsi qu'un certificat.

OpenVPN intègre des scripts pour faciliter la gestion de ces certificats :

Dans /usr/share/easy-rsa/

Voici quelques définitions de scripts :

-**clean-all** : création et/ou effacement des clés existantes ;

**-build-ca :** création de ma certification d'autorité ;

-**build-key-server** : création de la clé et d'un certificat serveur ;

-**build-key :** création de la clé et d'un certificat client ;

Avant de lancer les scripts éditer le fichier **vars** et entrer vos propres variables

Puis chargez les variables avec la commande :

**source ./vars**

puis on copie le dossier easy-rsa dans le dossier /etc/openVPN !

## Autorité de certification

Pour exécuter un script on utilise la commande **./[nom du script]**

Pour créer l'autorité de certification on exécutera donc

**./build-ca**

**Puis**

**./build-key-server [nom du serveur]**

Puis on génère les paramètres via l'algorithme Diffie Hellman via le script :

**./build-dh**

Enfin **sur la machine cliente debian** on copie le fichier **/etc/openVPN/easy-rsa/keys/ca.crt et ca.key**

Et on exécute la commande **./build-key client1**

Avec le commun name client1