DDWD

Abderraouf METIDJI

Job 01 - Installation de debian avec une interface graphique

Pour l'installation de Debian avec interface graphique on peut se référer au guide créé lors du projet "Mes premiers pas en informatique".

Il y a dans cette documentation une explication des étapes à suivre pour installer cet OS.

Job 02 - Installation de Apache2

Afin de démarrer l'installation d'Apache2 il faut commencer par un update des paquets debian.

Pour cela nous utilisons la commande

sudo apt update

Suite à cela nous allons installer apache à l'aide de la commande suivante :

sudo apt install apache2

Afin de vérifier que Apache2 est bel et bien installé on utilise la commande :

apache2 -version

Si nous avons ufw d'activer sur notre machine (pour la gestion des pare-feu) il faut penser à autoriser 2 ports web via la commande :

sudo ufw allow 80/tcp
sudo ufw allow 443/tcp

Une fois que tout cela a été fait nous vérifions que apache2 a bel et bien été lancé via :

sudo systemctl status apache2

Si le résultat de cette commande indique "**active (running)**" c'est que le service est en marche. Autrement il suffira d'utiliser l'une des commandes suivantes afin de le démarrer :

sudo service apache2 start

/etc/init.d/apache2 start

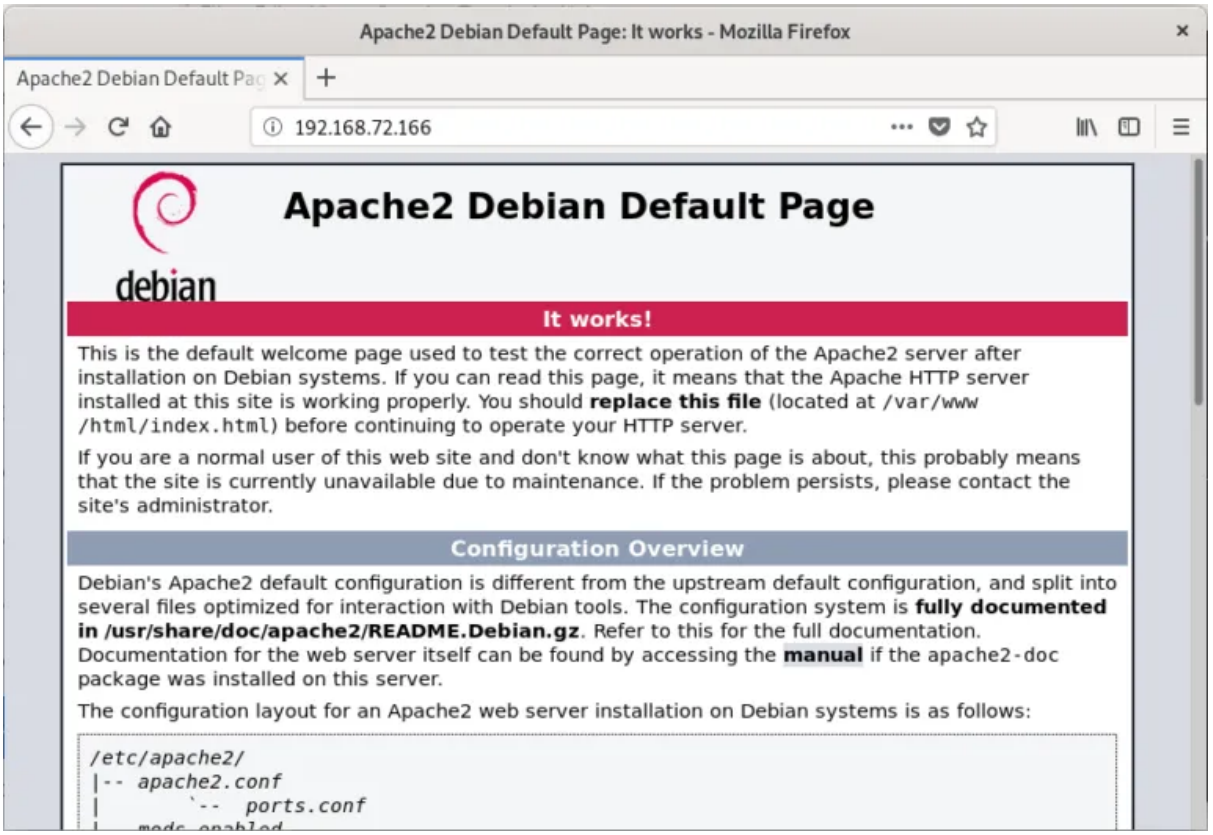
Ensuite nous utilisons :

hostname -I

Cette commande nous permet d'identifier l'adresse IP de notre serveur web apache. Pour vérifier que le serveur est bien accessible on cherche sur une page web :

http://IP_du_serveur_que_nous_venons_d'obtenir

Si on se retrouve avec une page comme celle-ci, c’est que tout fonctionne.



Job 03 - Serveurs Web (avantages & inconvénients)

Nous allons nous focaliser sur les serveurs qui possèdent les plus grandes parts de marché (selon Wikipédia).

Apache (42.5%)

Apache est un logiciel de serveur web gratuit et open-source qui alimente environ 46% des sites web à travers le monde. Le nom officiel est Serveur Apache HTTP et il est maintenu et développé par Apache Software Foundation.

Il permet aux propriétaires de sites web de servir du contenu sur le web – d'où le nom « serveur web » -. C'est l'un des serveurs web les plus anciens et les plus fiables avec une première version sortie il y a plus de 20 ans, en 1995.

Quand quelqu'un souhaite visiter un site web, il saisit un nom de domaine dans la barre d'adresse de son navigateur. Puis le serveur web fournit les fichiers demandés en agissant comme un livreur virtuel.

Avantages :

- Open-source et gratuit même pour un usage commercial.
- Logiciel fiable et stable.
- Mise à jour régulière, correctifs de sécurité réguliers.
- Flexible grâce à sa structure basée sur des modules.
- Facile à configurer, adapté aux débutants.
- Plateforme-Cross (fonctionne sur les serveurs Unix et Windows).
- Fonctionne avec les sites WordPress.
- Grande communauté et support disponible en cas de problème.

Inconvénients :

- Problèmes de performances sur les sites web avec un énorme trafic.
- Trop d'options de configuration peuvent mener à la vulnérabilité de la sécurité.

Nginx (40.1%)

Nginx est un serveur proxy inverse. Il prend en charge les protocoles suivants :

- HTTP et HTTPS
- IMAP

Le proxy inverse aide à équilibrer la charge en répartissant les requêtes et en mettant en cache certains types de contenu.

Comme Apache, Nginx a une architecture modulaire.

L'accélération de vos sites et applications est facile à réaliser avec Nginx. Nginx peut également améliorer considérablement l'architecture de votre application.

Avantages (par rapport à Apache) :

- Alors qu'Apache utilise une approche orientée processus pour gérer les requêtes, Nginx utilise une approche basée sur les événements.
- Cela le rend plus évolutif et plus apte à gérer des charges élevées ou des pics de trafic.
- Grâce à Nginx, l'Université du Texas à Austin est en mesure de fournir un temps de réponse moyen de 200 ms par application. Ils utilisent actuellement Nginx pour l'équilibrage de charge, la mise en cache et la livraison d'applications.
- Les développeurs utilisent Nginx car il consomme peu de ressources, ce qui le rend moins volatil dans un environnement d'hébergement Web.
- Il existe des exemples concrets de Nginx traitant avec succès des dizaines de millions de demandes chaque jour.
- Il peut gérer plus de 10 000 requêtes simultanées sans consommer de grandes quantités de RAM du serveur. Un compromis est un nombre réduit de fonctionnalités.

Inconvénients (par rapport à Apache) :

- La pile LAMP est presque la norme de l'industrie parmi les hébergeurs Web à faible coût
- Il y a beaucoup de soutien et d'aide disponibles
- La prise en charge de Python et Perl est intégrée à Apache, il est donc beaucoup plus facile de commencer à coder
- Ruby s'exécute plus rapidement dans Apache
- Apache dispose d'un grand nombre de modules disponibles pour l'étendre, il est donc compatible avec beaucoup plus de technologies tierces
- Nginx ne prend pas en charge .htaccess

IIS (11.9%)

Microsoft IIS est le serveur web fonctionnant sous Windows Server. IIS permet de gérer une application web avec une prise en charge avancée des langages de programmation au travers des modules CGI. IIS s'installe et s'administre via le gestionnaire de serveur comme tous les rôles Windows Server.

IIS prend en charge plusieurs techniques Web telles les CGI, les ASP, les ASP . NET et une API spécifique à IIS de nom ISAPI permettant de créer des extensions et des filtres. IIS prend aussi en charge le langage PHP en mode CGI ou ISAPI.

Avantages :

- IIS constitue un excellent choix pour la création et la gestion de sites Web commerciaux, tels que des boutiques en ligne ou des sites Web de portefeuilles promotionnels.
- IIS peut servir à la fois standard HTML pages Web et pages Web dynamiques, telles que ASP.NET applications et PHP pages.
- Lorsqu'un visiteur accède à une page d'un site statique, IIS envoie simplement le HTML et les images associées à l'utilisateur navigateur.

Inconvénients :

- IIS n’est pas aussi largement pris en charge que d’autres serveurs Web, tels qu’Apache. En tant que tel, il peut être plus difficile de trouver de l’aide et de la documentation pour IIS.
- Il n’est pas non plus aussi flexible que certains des autres serveurs Web. Il peut être difficile de le configurer pour certains types de déploiements.
- On peut l’utiliser qu’avec le système d’exploitation Windows.

LiteSpeed (2.3%)

LiteSpeed est un serveur web et un équilibreur de charge populaire à haute performance, qui peut être une meilleure option pour les sites web de commerce électronique à fort trafic si on le compare à Apache ou NGINX.

Bien qu'Apache soit l'un des serveurs web les plus répandus dans le monde, il présente certains inconvénients en termes de performances et d'évolutivité qui font qu'il ne convient pas aux sites web très chargés.

En utilisant LiteSpeed, vous pouvez gérer vos tâches avec les outils familiers à Apache, car LiteSpeed Web Server est entièrement compatible avec Apache Web Server en termes de format des fichiers de configuration. Et aussi, obtenir une solution hautement évolutive similaire à NGINX.

Avantages :

- L'un des principaux avantages de LiteSpeed est sa performance. Il peut être jusqu'à 6 fois plus rapide qu'Apache et jusqu'à 5 fois plus rapide que NGINX tout en servant des fichiers statiques et jusqu'à 50% plus rapide qu'Apache pour les sites web PHP.
- Un panneau web pour définir les paramètres et consulter les statistiques. Dans le cas d'Apache et de NGINX, vous devez tout configurer dans les fichiers de configuration, ce qui peut entraîner des erreurs ; LiteSpeed fournit une interface web pratique.
- Comparé à NGINX, LiteSpeed a des performances similaires, voire meilleures, pour la diffusion de contenu statique. Tout dépend des paramètres spécifiques.
- Il augmente la vitesse de mise en cache de vos fichiers statiques grâce à LiteSpeed Cache, ce qui améliore considérablement les performances de votre site web.

Inconvénients :

- Certaines fonctionnalités du plugin Litespeed cache qui amènent un vrai plus en termes de performance peuvent rapidement faire crasher votre site : les principaux sont liés à la minification ou à la combinaison de CSS/JS , parfois avec le CSS critique.
- Il n'est accessible gratuitement qu'aux personnes qui possèdent un hébergement utilisant la technologie LiteSpeed

Job 04 - DNS

Tout d'abord nous allons commencer par installer un service de DNS.

Ici nous utiliserons BIND (Berkeley Internet Name Domain).

sudo apt-get install bind9

La configuration principale de BIND9 est effectuée dans les fichiers suivant :

/etc/bind/named.conf
/etc/bind/named.conf.options
/etc/bind/named.conf.local

Afin de pouvoir ping le nom de domaine il faut mettre en place un forward entre l'adresse IP et le nom de domaine.

Pour cela nous allons éditer le fichier **named.conf.options** via la commande

sudo nano /etc/bind/named.conf.options

Il faudra ensuite modifier cette partie :

```
// forwarders {
//      0.0.0.0;
// };
```

On y ajoute l'adresse IP du serveur (que l'on obtient via ***hostname -I***) ainsi que le domaine en question.

```
// forwarders {
//      192.168.79.136;
//      dnsproject.prepa.com;
// };
```

Une fois cela fait, nous pourrons utiliser la commande suivante :

ping dnsproject.prepa.com

Afin de vérifier que nous pouvons bel et bien ping notre domaine.

ps : il est potentiellement nécessaire d'effectuer les modifications du Job 06 afin de pouvoir ping le serveur.

Job 05 - Domaine public

Étape 1 : choisir un bureau d'enregistrement de noms de domaine accrédité

Depuis 1998, ICANN (Internet Corporation for Assigned Names and Numbers) est responsable de l'attribution de toutes les extensions de domaine disponibles. L'organisation gère tous les domaines de premier niveau ainsi que les nouvelles extensions et vend les droits d'enregistrement aux registres, qui travaillent à leur tour avec les bureaux d'enregistrement de noms de domaine (également connus sous le nom de fournisseurs de noms de domaine). Ainsi, si vous souhaitez obtenir une adresse appropriée pour votre site Web, la première étape consiste à choisir le bureau d'enregistrement adéquat. Ce faisant, vous devez non seulement faire attention aux coûts, mais aussi vous assurer que le domaine de premier niveau souhaité est disponible. Il est important de choisir un fournisseur sérieux. Sinon, vous courez le risque que le domaine acheté ne soit pas enregistré correctement ou ne soit pas du tout disponible en réalité.

Étape 2 : trouver le bon nom de domaine, y compris le domaine de premier niveau

Comme mentionné précédemment, votre nom de domaine joue un rôle primordial dans le succès de votre présence en ligne. Une association accrocheuse et pertinente de l'extension (domaine de premier niveau) et du nom de domaine (domaine de deuxième niveau) est importante. Ainsi, les visiteurs ne trouvent pas seulement votre site via des liens, mais également lors de simples recherches sur Internet. De plus, les moteurs de recherche comme Google prennent en compte le nom de domaine dans le classement des résultats de recherche. C'est pourquoi un nom approprié peut certainement vous aider à obtenir un meilleur ranking.

En ce qui concerne les domaines de deuxième niveau, vous avez une grande liberté de conception : les noms peuvent comporter jusqu'à 63 caractères composés de lettres, chiffres, et traits d'union. Le grand défi consiste à faire correspondre le nom avec le domaine de premier niveau souhaité. En effet, les extensions de domaine les plus populaires sont souvent déjà attribuées ou réservées pour les combinaisons les plus diverses. Vous trouverez de plus amples informations et de l'aide pour trouver un nom de domaine approprié dans notre article détaillé contenant nos conseils les plus importants pour déposer un nom de domaine.

Étape 3 : vérifier la disponibilité de l'adresse avec Domain Check

Une fois que vous avez une ou, idéalement, plusieurs options en tête pour le nom de domaine de votre site Web, il est temps de vérifier la disponibilité. Presque tous les grands fournisseurs de domaines proposent un outil de vérification de domaine. En général, il suffit d'entrer le nom de domaine souhaité dans la barre de recherche de l'outil, puis de lancer la recherche.

Étape 4 : commande ou enregistrement du nom de domaine

L'outil de vérification vous permet de savoir avec certitude si le nom de domaine que vous souhaitez est disponible ou non. Mais comment acheter ce domaine Internet par la suite ?

Ici aussi, vous pouvez utiliser les outils de vérification de domaine des fournisseurs respectifs. Une fois l'adresse de vos rêves cochée, les outils vous amènent directement au processus d'achat et d'enregistrement

Étape 5 : vérifier la validité du nouveau domaine

Conformément à un règlement de l'ICANN, les fournisseurs de domaines sont tenus de vérifier l'authenticité des coordonnées des acheteurs de noms de domaine depuis le 1er septembre 2014. Cela s'applique aux domaines de premier niveau génériques, qu'ils soient classiques ou nouveaux. À la suite de l'achat d'une nouvelle adresse Web, les fournisseurs envoient donc un email à l'adresse de contact que vous avez fournie lors de l'enregistrement.

Comment enregistrer un domaine ?

Les nouveaux TLD sont très recherchés, mais l'ICANN ne les libère que progressivement. Lorsque vous utilisez l'outil de vérification, si un domaine n'a pas encore été libéré, vous ne pourrez pas acheter l'adresse souhaitée avec ce TLD. Toutefois, vous pouvez sécuriser ou réserver ce domaine auprès du fournisseur de votre choix.

Comment acheter un nom de domaine qui n’est plus disponible ?

Si le nom de domaine de vos rêves est déjà pris, vous n’avez aucun moyen de l’acheter au fournisseur, même si l’adresse n’est pas active. Vous pouvez toutefois essayer de localiser le propriétaire du domaine (via une requête WHOIS dans la base de données du registre concerné) afin de lui faire une offre d’achat. Notre article « Nom de domaine indisponible : voilà comment faire » fournit des informations sur l’ensemble du processus, de la recherche du propriétaire du domaine à l’achat du domaine et à son transfert.

Job 06 - Connexion au domaine local

Afin de faire correspondre l'adresse IP du serveur au nom de domaine "dnsproject.prepa.com", nous devons modifier le fichier host.

Pour cela nous allons faire :

sudo nano /etc/hosts

Au sein de ce fichier nous allons ajouter une nouvelle entrée qui fera correspondre l'IP du serveur au nom de domaine local.

Dans notre exemple, afin d'obtenir l'adresse IP du serveur nous utilisons :

hostname -I

Cela nous donne **192.168.79.136**

Suite à cela nous ajoutons dans le fichier la ligne suivante :

192.168.79.136 dnsproject.prepa.com

Une fois cela fait nous pourrons accéder au serveur via :

192.168.79.136 ainsi que via ***dnsproject.prepa.com***

Les deux résultats ouvriront la page Apache2 que nous avons vu précédemment avec l'indication "It Works".

Job 07 - DHCP

Dans un premier temps nous allons installer le serveur DHCP à l'aide la commande :

sudo apt-get install isc-dhcp-server

Il faut ajouter son interface réseau au fichier :

nano /etc/default/isc-dhcp-server

Comme ci dessous :

INTERFACESv4="nom de l'interface réseau, exemple eth0"

Pour trouver cette interface il suffit de faire :

ip a

Une fois l'installation effectuée et réussie nous devons modifier le fichier :

/etc/dhcp/dhcpd.conf

Voici un exemple de configuration basique :

```
# Sample /etc/dhcpd.conf
# (add your comments here)
default-lease-time 600;
max-lease-time 7200;
option subnet-mask 255.255.255.0;
option broadcast-address 192.168.1.255;
option routers 192.168.1.254;
option domain-name-servers 192.168.1.1, 192.168.1.2;
option domain-name "projectdns.prepa.com";
option ntp-servers 192.168.1.254;
```

Un autre exemple plus concret :

```
subnet 192.168.1.0 netmask 255.255.255.0 {
  range 192.168.1.10 192.168.1.100;
  range 192.168.1.150 192.168.1.200;
}
```

Suite à cela nous devons redémarrer le service dhcp via la commande

sudo service isc-dhcp-server restart

On peut également vérifier les baux (leases) qui sont en cours via

/var/lib/dhcp/dhcpd.leases

Job 08 - Gateway

Une fois que nous avons correctement configuré notre DHCP nous pouvons ajouter un gateway (une passerelle).

Le serveur joue le rôle de la passerelle dans ce cas.

Pour cela, il faut modifier :

/etc/network/interfaces

Dans ce fichier, nous allons ajouter l'adresse IP du gateway comme ci-dessous :

allow-hotplug eth0
iface eth0 inet static
  address 192.168.1.50
  broadcast 192.168.1.255
  netmask 255.255.255.0
  gateway 192.168.1.1

Une fois l'adresse ajouté, les VMs connectés au serveur et sur le même réseau local devraient normalement avoir accès à internet.

Job 09 - Pare-feu

Tout d'abord nous allons commencer par installer ufw :

sudo apt install ufw

Une fois cela fait nous allons utiliser les commandes suivantes pour rétablir les commandes par défaut :

sudo ufw default deny incoming
sudo ufw default allow outgoing

Si nous utilisons SSH, il faudra autoriser les connexions SSH via

sudo ufw allow ssh

Ainsi qu'ajouter les ports que notre SSH utilise (par défaut 2222) :

sudo ufw allow 2222/tcp

Il suffit ensuite d'activer ufw via :

sudo ufw enable

Pour autoriser les connexions http et https nous utiliserons respectivement :

sudo ufw allow http ou ***sudo ufw allow 80***
sudo ufw allow https ou ***sudo ufw allow 443***

Ensuite nous pouvons autoriser la connexion à partir d'une adresse IP spécifique :

sudo ufw allow from X.X.X.X to any port 2222

Job 10 - Dossier partagé

Pour les dossiers partagés avec un serveur Apache on peut les trouver dans :

home/www/

Afin de créer un dossier partagé à l'aide de Apache nous pouvons tout simplement ajouter un dossier au sein de :

home/www/

Par exemple :

home/www/html

Dans ce cas les différents utilisateurs du réseau local pourront naviguer sur :

http://dnsproject.prepa.com/html

Ils devraient alors avoir accès au dossier.

Si nous souhaitons rendre ce dossier disponible sur l'interface graphique de chaque VM il faut suivre les étapes suivantes.

sudo rm /var/www/html/index.html

Afin d'enlever l'index.html du dossier que nous avons créé. Il faudra ensuite :

sudo cp /path/to/file /var/ww/html/nomdefichier

Pour copier un fichier dans le dossier que l'on a créé. On peut aussi utiliser :

sudo cp -r /path/to/file /var/www/html/nomdedossier

Si l'on souhaite copier un dossier et non un fichier.

Une fois ces étapes effectuées, les personnes connectées à notre serveur web Apache et à notre réseau local devraient être en mesure d'accéder au dossier créé via l'interface graphique.

Si cela ne fonctionne pas, il sera peut-être nécessaire de modifier les autorisations du dossier home/www/.

Pour aller plus loin…

Dans un premier temps nous devons utiliser :

sudo apt install apache2 openssl

Cela permettra d'ajouter la partie openssl à notre serveur Apache.

Suite à cela nous devons activer le module ssl ainsi que le module rewrite :

sudo a2enmod ssl
sudo a2enmod rewrite

Le module a2enmod permet d'activer et de désactiver les modules dans la configuration Apache.

Une fois cela fait, nous devons activer l'override des paramètres par défaut de Apache.

Nous devons ouvrir :

sudo nano /etc/apache2/apache2.conf

Naviguer à la fin du fichier et ajouter les lignes suivantes :

<Directory /var/www/html>

AllowOverride ALL

</Directory>

Avant de générer le certificat nous allons créer un dossier dans le repo apache :

sudo mkdir /etc/apache2/certs

Nous naviguons dans ce dossier :

cd /etc/apache2/certs

Nous utilisons ensuite openssl afin de générer le certificat auto-signé :

sudo openssl req -new -newkey rsa:4096 -x509 -sha256 -days 365 -nodes -out apache.crt -keyout apache.key

Il faut ensuite remplir les informations demandées, les plus importantes étant l'adresse IP et le hostname (nom d'hôte).

Si le processus a bien fonctionné nous devrions avoir deux fichiers, ***apache.key*** et ***apache.crt***, dans le dossier ***/etc/apache2/certs***.

Suite à cela nous allons ajouter un host bloc sur le port 443 via :

sudo nano /etc/apache2/sites-enabled/000-default.conf

Ici il faudra ajouter :

```
<VirtualHost *:443>
    ServerAdmin webmaster@localhost

    DocumentRoot /var/www/html

    ErrorLog ${APACHE_LOG_DIR}/error.log

    CustomLog ${APACHE_LOG_DIR}/access.log combined

    SSLEngine on

    SSLCertificateFile /etc/apache2/certs/apache.crt

    SSLCertificateKeyFile /etc/apache2/certs/apache.key
</VirtualHost>
```

Dans la plupart des cas, il faudra ajouter une redirection vers https sur le port 80 virtual hosts :

```
        RewriteEngine on

        RewriteCond %{HTTPS} !=on

        RewriteRule ^/?(.*) https://%{SERVER_NAME}/$1 [R=301,L]
```

Il faut faire attention à ce que le blocage mis en place plus haut se trouve en dessous du port 80 dans le fichier.

Ensuite nous redémarrons Apache et nous accédons au serveur via localhost. Nous serons en mesure de voir le certificat et les informations qui le concernent.

Renseignez-vous aussi sur la différence entre les certificats SSL donnés par des organismes extérieurs et le vôtre auto-signé ?

Lorsqu'on achète un certificat SSL traditionnel, vous savez qu'il a été signé par une autorité de certification réputée. En revanche, un certificat auto-signé n'est pas signé par une autorité comme SSL ou TLS ; il est créé, mis en œuvre et signé par un développeur de logiciels tiers.

Pourquoi votre certificat apparaît-il comme non sécurisé dans votre navigateur ?

Un certificat auto-signé qui n'a pas été émis par une autorité de certification est considéré comme non fiable par défaut. Les certificats auto-signés peuvent sécuriser vos données par rapport aux oreilles indiscrètes, mais ne disent rien sur les destinataires des données que vous communiquez.