

La Plateforme

DDWS

DHCP DNS Web server Samba !



Apache

SALL Moustapha


Prépa Bachelor IT

job 1

Installation de Debian en mode graphique

job 2

installer un serveur Web Apache2.



Apache2 Debian Default Page

debian

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Debian systems. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Debian's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Debian tools. The configuration system is **fully documented in** [`/usr/share/doc/apache2/README.Debian.gz`](#). Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Debian systems is as follows:

```
/etc/apache2/
|-- apache2.conf
|   |-- ports.conf
|-- mods-enabled
|   |-- *.load
|   |-- *.conf
|-- conf-enabled
|   |-- *.conf
|-- sites-enabled
|   |-- *.conf
```

job 3

Renseignez-vous sur les différents serveurs Web existants et produisez une documentation qui contiendra votre recherche ainsi que les avantages et inconvénients de chacun des serveurs :

Apache HTTP Server (Apache) :

Avantages :

Gratuit et open source.

Très flexible et personnalisable grâce à des modules tiers.

Prise en charge de nombreuses langues de programmation (PHP, Python, etc.).

Documentation complète et communauté active.

Inconvénients :

Peut être gourmand en ressources dans certaines configurations.

La configuration peut sembler complexe pour les débutants.

Nginx :

Avantages :

Excellente performance, particulièrement pour les sites à forte charge.

Économe en ressources.

Capacité de proxy inverse pour équilibrer la charge.

Prise en charge de la réécriture d'URL et des contenus statiques.

Inconvénients :

Configuration parfois moins intuitive que celle d'Apache.

La prise en charge des langues de programmation peut nécessiter des modules tiers.

Microsoft Internet Information Services (IIS) :

Avantages :

Intégré à Windows Server pour une intégration facile.

Prise en charge des technologies Microsoft telles que ASP.NET.

Interface utilisateur graphique conviviale pour la configuration.

Inconvénients :

Disponible principalement sur les systèmes Windows.

Les licences peuvent être coûteuses pour certaines versions.

LiteSpeed Web Server :

Avantages :

Performances élevées grâce à une architecture de type événement.

Prise en charge de la réécriture d'URL et de la mise en cache.

Bonne prise en charge de PHP.

Inconvénients :

Versions gratuites limitées en fonctionnalités.

Licence payante pour les versions complètes.

Caddy :

Avantages :

Configuration automatique avec prise en charge de Let's Encrypt pour les certificats SSL.

Interface de configuration facile à utiliser.

Prise en charge des protocoles HTTP/2 et HTTP/3.

Inconvénients :

Peut manquer de certaines fonctionnalités avancées par rapport à d'autres serveurs web.

Tomcat (serveur d'application Java) :

Avantages :

Conçu pour héberger des applications Java et des servlets.

Bonne intégration avec des technologies Java telles que JSP.

Inconvénients :

N'est pas un serveur web traditionnel, mais un serveur d'application Java.

Moins adapté aux sites web statiques.

Job 4

Mettez en place un DNS sur votre serveur Linux qui fera correspondre l'adresse IP de votre serveur au nom de domaine local suivant : "dnsproject.prepa.com" Votre serveur devra donc pouvoir se ping via ce nom de domaine.

job 5

Pour obtenir un nom de domaine public, on peut suivre ces étapes suivantes :

Sélectionnez un Registrar de Domaine :

Choisissez un registrar de domaine, une entreprise ou un service qui vous permettra de rechercher, enregistrer et gérer des noms de domaine. Il existe de nombreux registrars réputés, tels que GoDaddy, Namecheap, Google Domains, etc.

Vérifiez la Disponibilité du Nom de Domaine :

Utilisez l'outil de recherche de domaine fourni par le registrar pour vérifier si le nom de domaine que vous souhaitez est disponible. Si le nom de domaine est déjà pris, vous devrez en choisir un autre ou envisager des alternatives.

Sélectionnez un Nom de Domaine :

Une fois que vous avez trouvé un nom de domaine disponible qui correspond à vos besoins, ajoutez-le à votre panier.

Ajoutez des Services Supplémentaires :

Vous pouvez également choisir d'ajouter des services supplémentaires, tels que l'hébergement web, des adresses email associées au domaine, ou la protection de la vie privée (pour masquer vos informations personnelles associées au domaine) selon les offres du registrar.

Procédez au Paiement :

Ajoutez vos informations de paiement et procédez à la transaction. Les tarifs et les conditions varient en fonction du registrar et de la durée de l'enregistrement (généralement par an).

Fournissez des Informations WHOIS :

Lors de l'achat de votre nom de domaine, vous devrez fournir des informations WHOIS, telles que le nom du propriétaire, l'adresse et les coordonnées de contact. Certaines entreprises offrent la protection de la vie privée pour masquer ces informations.

Confirmez l'Enregistrement :

Une fois la transaction confirmée, vous recevrez généralement une confirmation de l'enregistrement de votre nom de domaine.

Gérez Votre Domaine :

Utilisez le panneau de contrôle fourni par le registrar pour gérer les paramètres de votre domaine, comme la redirection, les enregistrements DNS, et d'autres configurations.

Quelles sont les spécificités que l'on peut avoir sur certaines extensions de nom de domaine ?

Voici quelques exemples de spécificités que l'on peut trouver sur certaines extensions de nom de domaine :

Géographiques (ccTLDs) : Les domaines de premier niveau géographiques, tels que .fr pour la France ou .uk pour le Royaume-Uni, sont associés à des pays ou des territoires spécifiques. Ils peuvent avoir des exigences de résidence ou d'enregistrement pour les détenteurs, limitant ainsi leur utilisation aux résidents ou aux entités liées à ce pays ou territoire.

Restrictions de Secteur d'Activité : Certaines extensions, telles que .bank ou .pharmacy, sont réservées à des secteurs d'activité spécifiques. Pour enregistrer un domaine avec ces extensions, il peut être nécessaire de prouver que vous êtes une entité légitime dans ce secteur.

Éducatives et gouvernementales : Les extensions .edu (pour les institutions éducatives) et .gov (pour les agences gouvernementales) sont généralement restreintes à des institutions et organismes gouvernementaux ou éducatifs légitimes.

Noms de Domaine Premium : Certaines extensions de domaine, notamment pour les nouveaux TLDs, proposent des noms de domaine premium qui sont disponibles à un coût supérieur en raison de leur attrait ou de leur valeur potentielle. Par exemple, un nom de domaine court ou générique peut être considéré comme premium.

Personnalisation et Créativité : Certaines extensions de domaine, telles que .io (utilisée par de nombreuses startups technologiques) ou .me (pour les sites personnels ou de blog), sont utilisées de manière créative pour former des mots ou des expressions liés au

contenu du site.

Longueur Minimale/Maximale : Certaines extensions peuvent avoir des exigences en matière de longueur minimale ou maximale des noms de domaine. Par exemple, le .muséum peut exiger un certain nombre de caractères pour promouvoir des noms de domaine informatifs.

Accents et Caractères Spéciaux : Certaines extensions de domaine permettent l'utilisation d'accents et de caractères spéciaux liés à des langues spécifiques, tels que l'extension .bébé pour les sites en français.

Certains TLDs peuvent être bloqués par certains Pays : Dans certains pays, l'accès à certaines extensions de domaine peut être restreint ou bloqué en raison de réglementations locales ou de censure.

Ces spécificités sont générales et peuvent varier en fonction des politiques spécifiques des registrars de domaines. Lors du choix d'une extension de domaine, il est important de considérer la pertinence pour le type de contenu que vous prévoyez de publier sur votre site web. De plus, certaines extensions peuvent avoir des règles d'éligibilité spécifiques, comme .edu réservé aux institutions éducatives, alors assurez-vous de respecter ces exigences si elles s'appliquent.

Job 6

Connectez votre hôte au nom de domaine local de votre serveur, pour que votre page apache soit accessible via ce même nom de domaine.

Pour ce faire: Ouvrez le fichier `/etc/hosts` en utilisant un éditeur de texte tel que nano ou vim avec des privilèges d'administrateur.

`sudo nano /etc/hosts`

Ajoutez une ligne à la fin du fichier avec l'adresse IP de votre serveur suivie du nom de domaine que vous souhaitez utiliser; Activez le module rewrite en tapant la commande `sudo a2enmod rewrite`

Redémarrez ensuite Apache pour appliquer les changements avec

`sudo service apache2 restart`

Créez un fichier de configuration pour votre site dans le répertoire sites-available.

Modifiez le fichier pour ressembler à quelque chose comme ceci :<VirtualHost *:80>

ServerAdmin webmaster@localhost

ServerName votrenomdedomaine.local

DocumentRoot /var/www/html

<Directory /var/www/html>

Options Indexes FollowSymLinks

AllowOverride All

Require all granted

</Directory>

ErrorLog \${APACHE_LOG_DIR}/error.log

CustomLog \${APACHE_LOG_DIR}/access.log combined

</VirtualHost>

Activez le site et redémarrez Apache pour appliquer les modifications :

sudo a2ensite votrenomdedomaine.conf

sudo service apache2 restart

Testez l'accès à votre site via le nom de domaine local :

Ouvrez votre navigateur web et accédez à <http://votrenomdedomaine.local>. Vous devriez voir la page d'accueil d'Apache si tout a été configuré correctement.



Apache2 Debian Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Debian systems. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Debian's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Debian tools. The configuration system is **fully documented in `/usr/share/doc/apache2/README.Debian.gz`**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Debian systems is as follows:

```
/etc/apache2/
|-- apache2.conf
|   |-- ports.conf
|-- mods-enabled
|   |-- *.load
|   |-- *.conf
|-- conf-enabled
|   |-- *.conf
|-- sites-enabled
|   |-- *.conf
```

Job 7

Mettez en place un pare-feu en utilisant `ufw` sur votre serveur principale de manière que votre hôte puisse accéder à la page apache par défaut, mais qu'il ne puisse plus ping votre serveur.

Installer `ufw` sur le serveur principal en utilisant la commande: `sudo apt-get install ufw`

Ensuite autorisez l'accès au serveur Apache en utilisant la commande `sudo ufw allow 80`

Bloquer les ping par `sudo ufw deny icmp`

Activez le pare-feu en utilisant la commande `sudo ufw enable`

vérifiez la configuration du pare-feu avec `sudo ufw status`

Tester l'accès au serveur Apache depuis l'hôte :

Ouvrez un navigateur web sur l'hôte et accédez à la page Apache par défaut du serveur. Par exemple, l'adresse IP de mon serveur est `192.168.5.133`, entrez cette adresse IP dans le navigateur.

Tester le blocage des ping avec `ping 192.168.5.133`

Cette action permettra de configurer ufw pour autoriser l'accès au serveur Apache tout en bloquant les pings.

Job 8

1. Installer Samba sur le serveur Debian :

```
sudo apt-get update
```

```
sudo apt-get install samba
```

2. Configurer un partage Samba :

Éditez le fichier de configuration de Samba avec un éditeur de texte :

```
sudo nano /etc/samba/smb.conf
```

Ajoutez à la fin du fichier la configuration de votre partage. Par exemple, ajoutez les lignes suivantes :

[Partage]

```
comment = Dossier partagé
```

```
path = /chemin/vers/votre/dossier
```

```
browsable = yes
```

```
guest ok = yes
```

```
read only = no
```

```
create mask = 0777
```

```
directory mask = 0777
```

Créer un utilisateur Samba :

```
sudo smbpasswd -a votre_utilisateur
```

Suivez les instructions pour définir un mot de passe pour l'utilisateur.

Redémarrer le service Samba :

```
sudo service smbd restart
```

Configurer le pare-feu pour autoriser le trafic Samba :

Autoriser le trafic Samba :

```
sudo ufw allow 137/udp
```

```
sudo ufw allow 138/udp
```

```
sudo ufw allow 139/tcp
```

```
sudo ufw allow 445/tcp
```

Accéder au dossier partagé depuis d'autres machines :

Sur les autres machines du réseau, utilisez l'explorateur de fichiers en interface graphique pour accéder au dossier partagé. Vous pouvez généralement le faire en entrant dans la barre d'adresse :

arduino

```
smb://192.168.5.133/Partage
```

Après cela, le dossier partagé devrait être accessible et utilisable comme tout autre dossier local.

