

Services Web (HTTP/HTT PS)

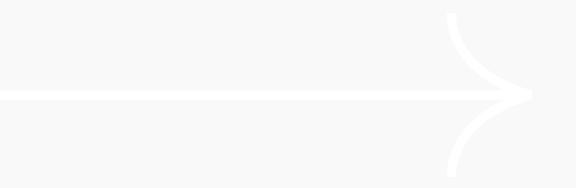
SAGALEYNI AIDARA ET IBRAHIMA
DIALLO



Introduction aux Services Web

COMPRENDRE HTTP ET HTTPS POUR L'AVENIR

Dans ce module, nous explorerons les services web, notamment le fonctionnement d'**HTTP** et la sécurisation avec **HTTPS**, essentiels dans le paysage numérique moderne et la cybersécurité.



Fonctionnement du protocole HTTP

COMPRENDRE LE MODÈLE CLIENT/SERVEUR

HTTP

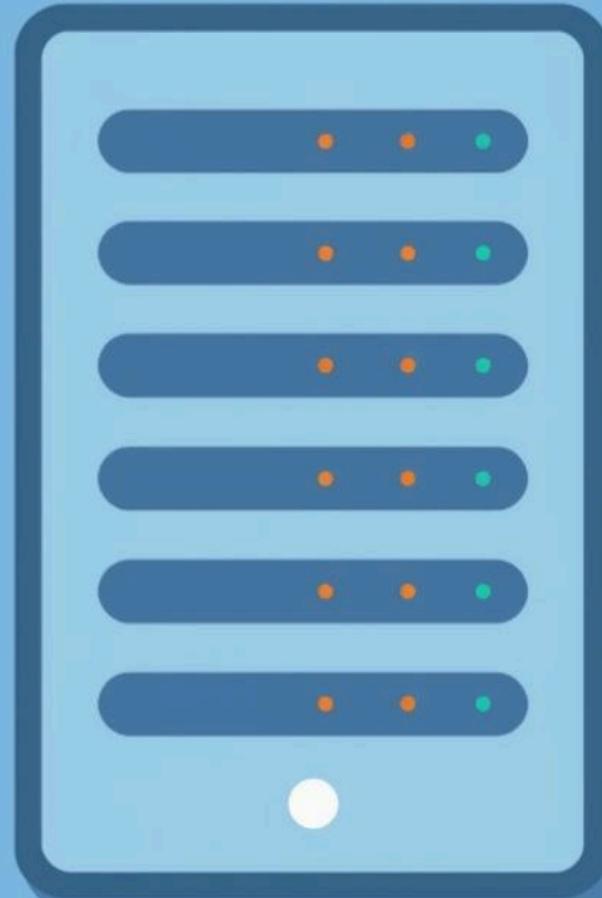
HTTP, ou **HyperText Transfer Protocol**, est le protocole essentiel qui permet la communication entre un navigateur web et un serveur, facilitant ainsi l'accès aux contenus en ligne.

Client/Serveur

Dans le modèle **client/serveur**, le client envoie une requête HTTP, et le serveur répond avec les données demandées, permettant une interaction fluide et efficace sur le web.



Serveurs Web



Les serveurs web, tels qu'**Apache HTTP Server** et **Nginx**, jouent un rôle crucial dans l'hébergement de sites internet. Apache est renommé pour sa simplicité, tandis que Nginx est apprécié pour ses performances. Choisir le bon serveur est essentiel pour une gestion efficace des requêtes.

Les certificats SSL / TLS

CHIFFREMENT ET CONFIDENTIALITÉ DES DONNÉES

Chiffrement des données

Le chiffrement assure que les données transmises entre le client et le serveur sont **protégées**, rendant difficile pour les attaquants d'accéder à des informations sensibles, comme les mots de passe.

Confidentialité

Grâce à SSL/TLS, les échanges deviennent privés et **sécurisés**, garantissant que seules les parties autorisées peuvent accéder aux informations, renforçant ainsi la **confiance** des utilisateurs envers les services en ligne.



Autorité de certification



Les autorités de certification (CA) jouent un rôle crucial dans la sécurité web. Elles délivrent des certificats SSL/TLS, garantissant la **confiance** entre le client et le serveur. Ces certificats assurent que les échanges de données sont sécurisés et intègres, renforçant ainsi la cybersécurité.

Let's Encrypt



Let's Encrypt est une **autorité de certification gratuite** qui fournit des certificats SSL valables 90 jours. Avec un renouvellement automatique, elle joue un rôle crucial dans la démocratisation de la sécurité HTTPS, permettant aux utilisateurs de sécuriser facilement leurs sites web sans frais.

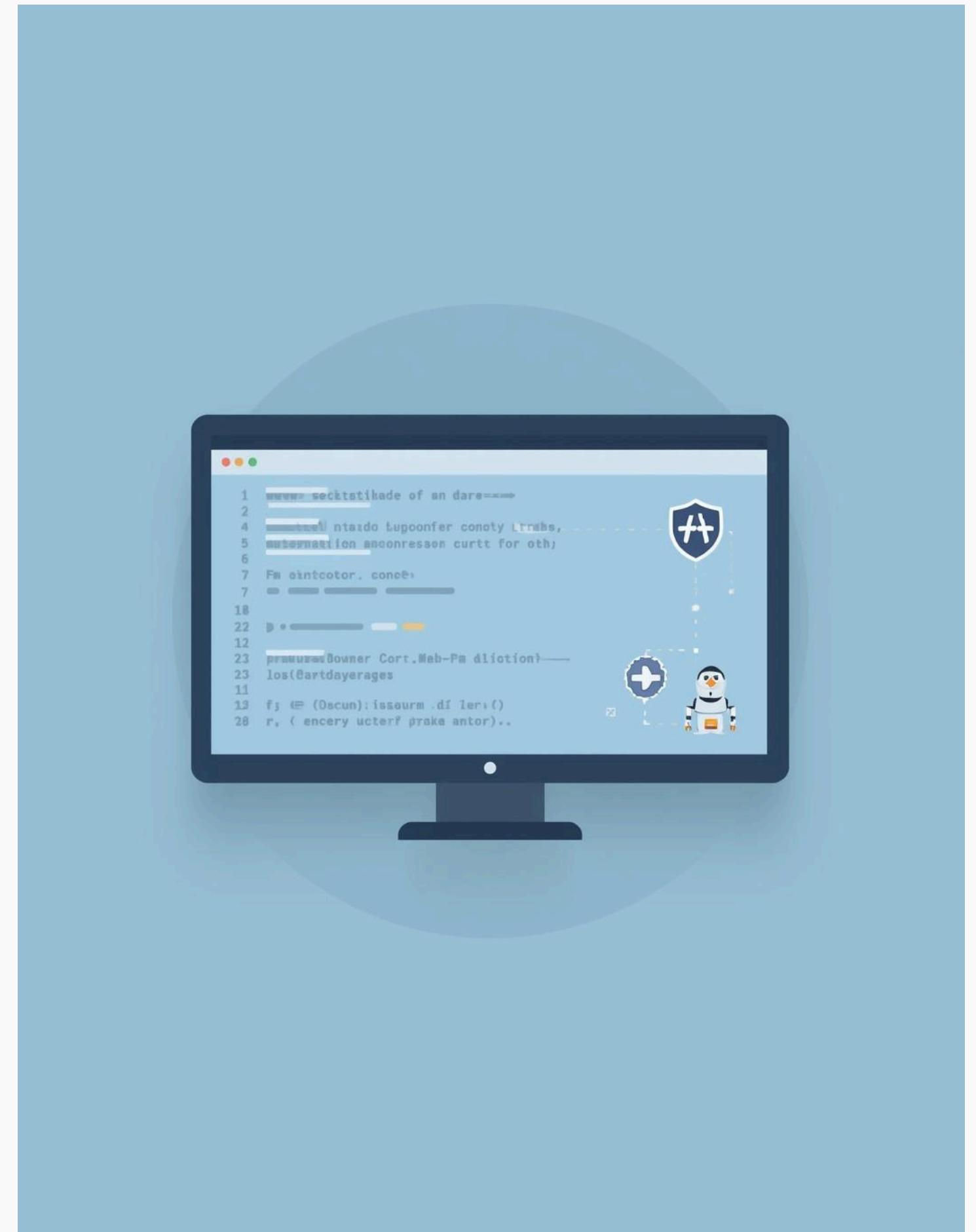
Certbot

AUTOMATISATION

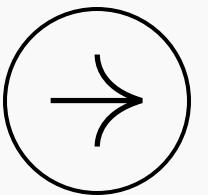
- 01 **Installation**

Pour installer ces certificats, on utilise un outil appelé :
Certbot
Certbot permet d'automatiser :
la génération du certificat la configuration HTTPS le renouvellement automatique.
- 02 **Configuration**

Certbot facilite la configuration d'HTTPS, rendant le processus d'obtention et de renouvellement de certificat rapide et simple.



HTTPS Sécurisé



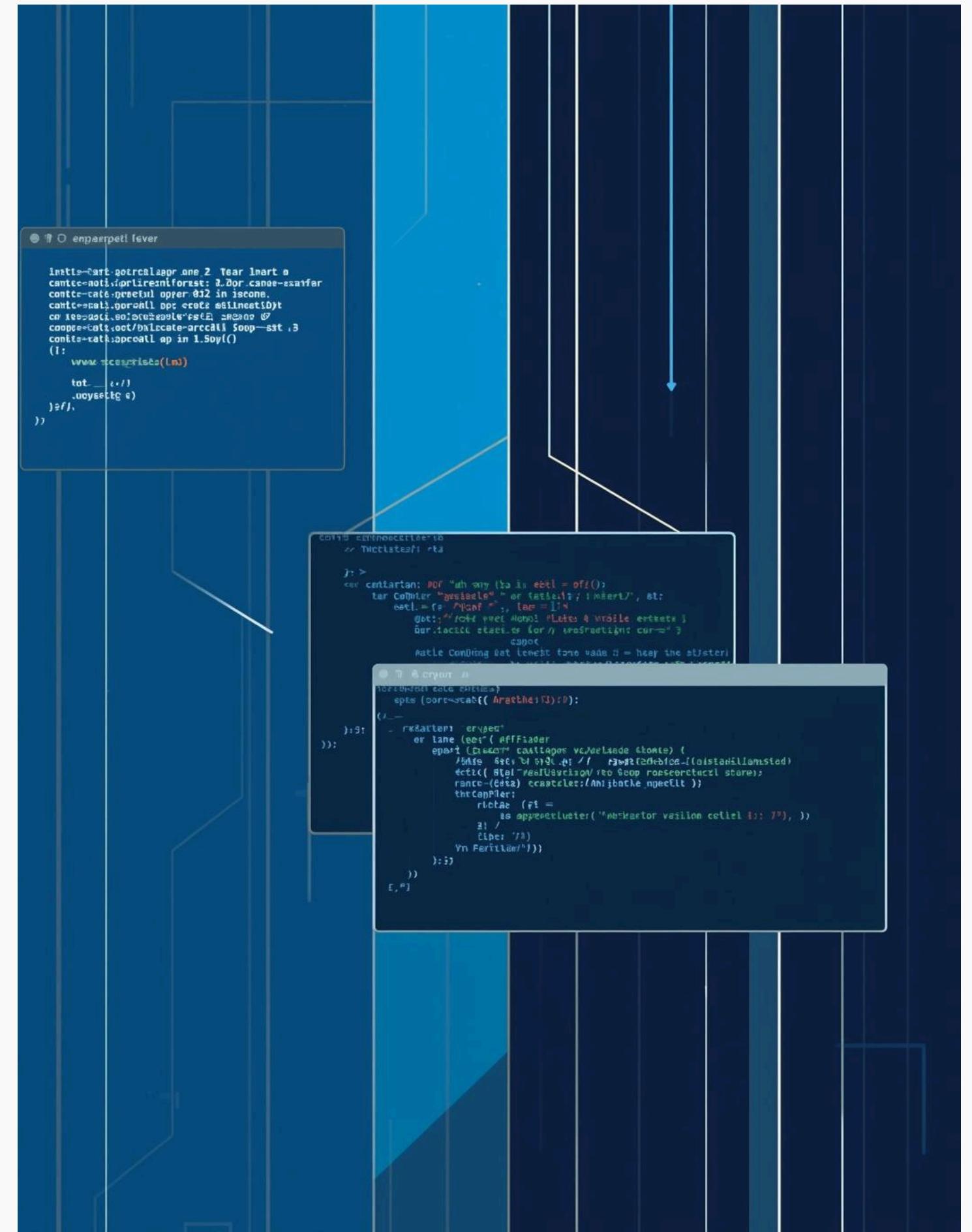
Protection des données avec SSL/TLS



Introduction au déploiement

Installation d'un serveur web sécurisé

L'objectif de cette partie est de **mettre en place** un serveur web sécurisé en utilisant Apache 2 sur un système Ubuntu. Cela garantira la protection des données échangées en ligne.



Vérification

TESTER

01

HTTP

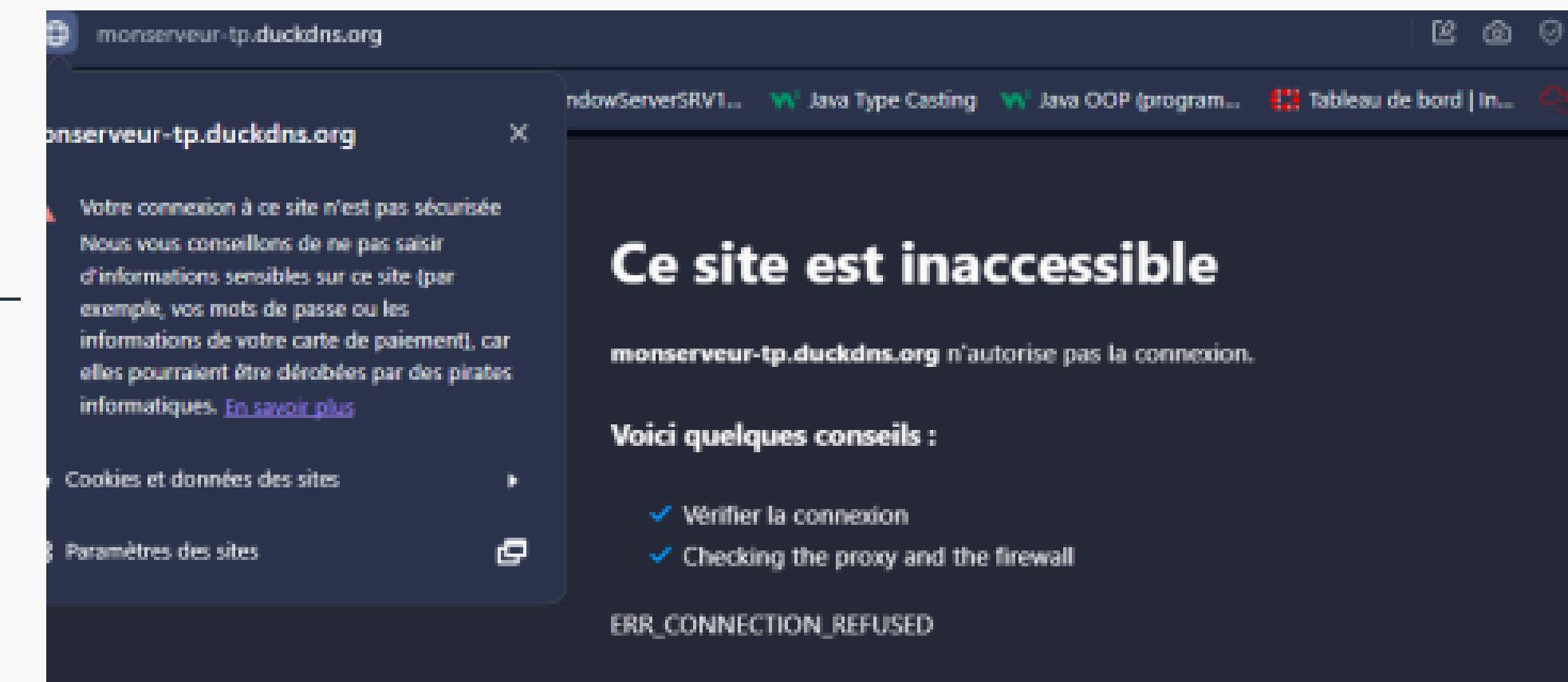
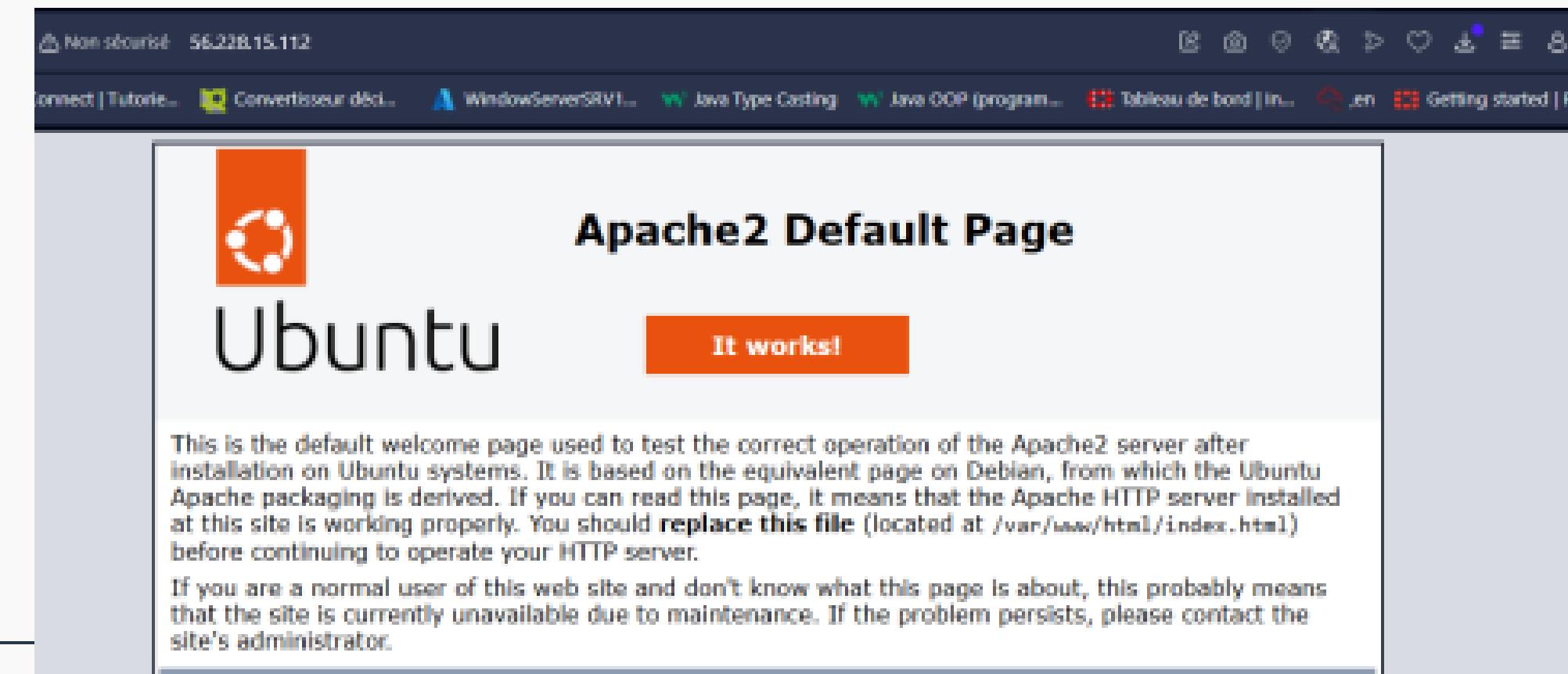
Pour vérifier le site, ouvrez un navigateur et entrez l'**IP publique de la VM** pour accéder à la page.

02

HTTPS

L'URL doit être **https://monserveur-tp.duckdns.org**, mais le certificat doit encore être ajouté pour sécuriser la connexion.

NB: Si on tape https://monserveur-tp.duckdns.org ca ne marche pas puis qu'on a pas encore ajouter le certificat



Configuration

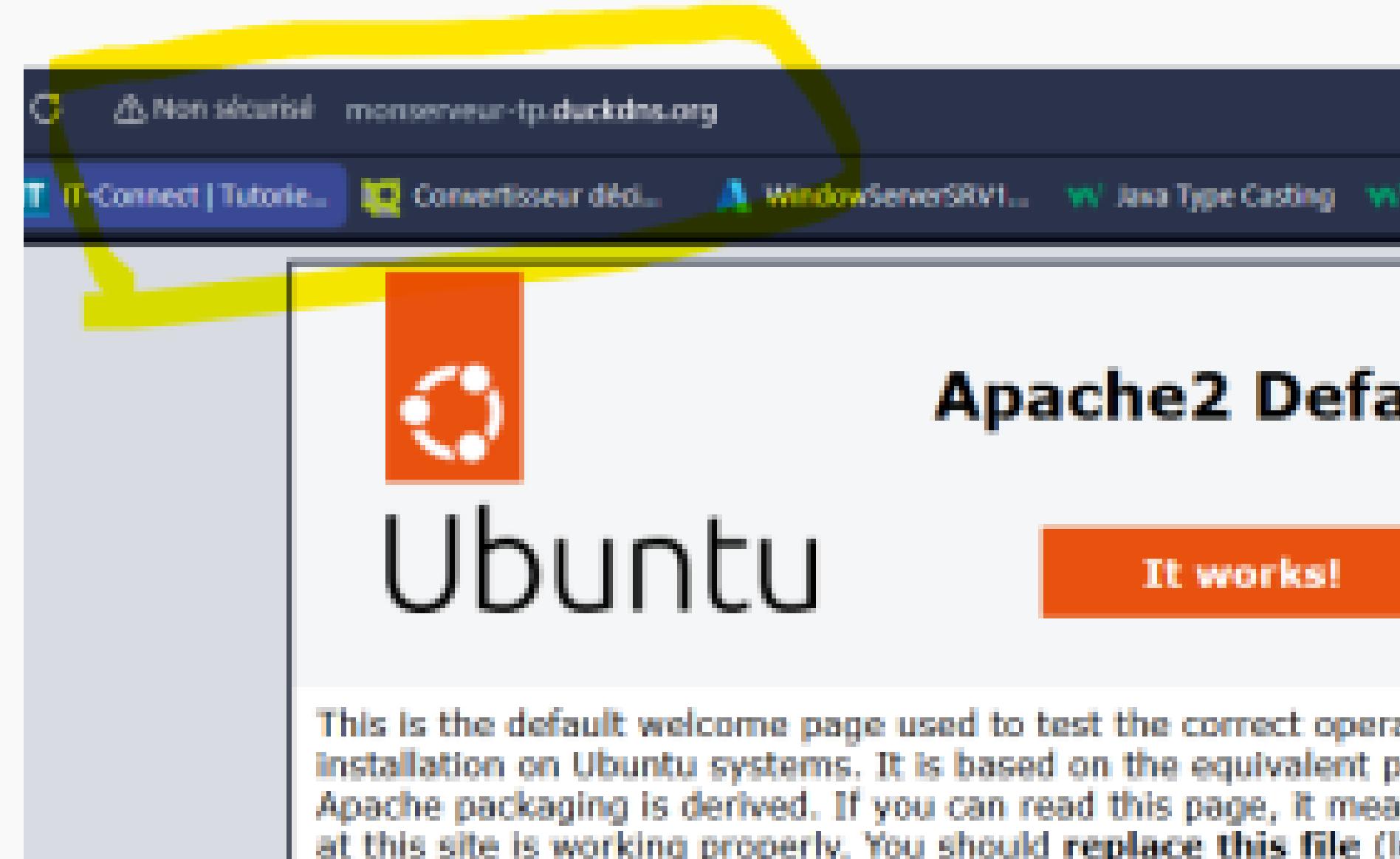
SOUS-DOMAINE

01 Sous-domaine IP

Avoir un sous-domaine gratuit pointant vers la VM AWS qui héberge Apache, pour pouvoir tester HTTP/HTTPS.

Création du sous-domaine :

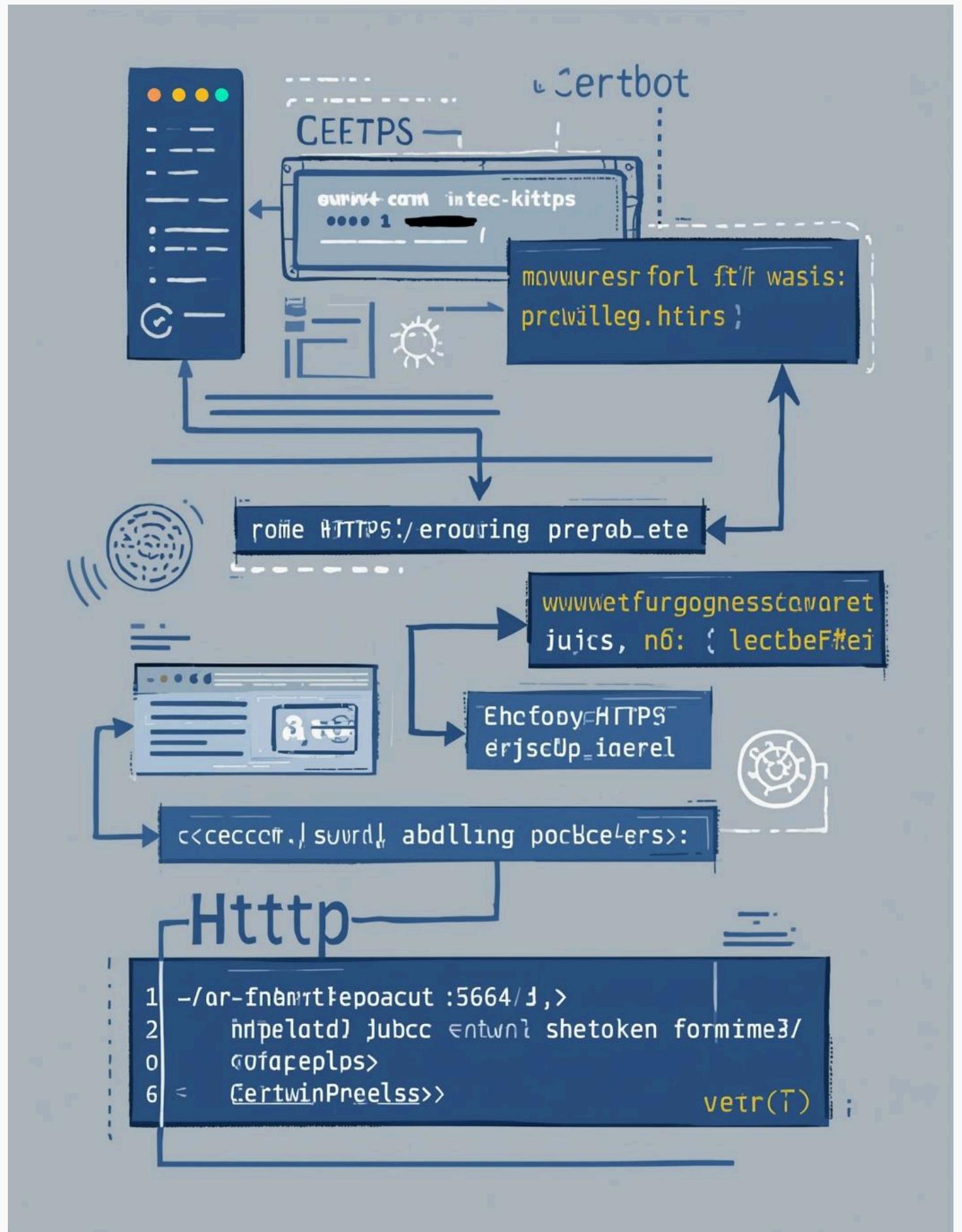
IP: 56.228.15.112 → NomDomain: monserveur-
tp.02.duckdns.org



Configuration HTTPS

Automatisation avec Certbot pour sécurité

La configuration du protocole HTTPS est simplifiée grâce à Certbot, qui permet d'automatiser l'obtention et le déploiement des certificats SSL/TLS pour un serveur sécurisé.



Installer Certbot

Configuration de Certbot sur Apache

Pour sécuriser notre serveur, nous devons **installer Certbot** et le plugin Apache. Certbot nous aidera à automatiser la gestion des certificats SSL/TLS pour HTTPS.

```
sudo apt update
sudo apt install certbot python3-certbot-apache -y
adding package lists... Done
adding dependency tree... Done
adding state information... Done
0 following additional packages will be installed:
augeas-lenses libaugeas0 python3-augeas python3-certbot python3-configparser python3-fuse python3-jcrapy python3-pexpect
python3-requests-toolbelt python3-rfc3339 python3-sopel-component python3-sopel-event python3-sopel-hookable
replaced packages:
augeas-doc python-certbot-doc python3-certbot-nginx augeas-tools python3-augparse python3-certbot-apache-doc
0 following NEW packages will be installed:
augeas-lenses certbot libaugeas0 python3-augeas python3-certbot python3-configparser python3-fuse
python3-pexpect python3-requests-toolbelt python3-rfc3339 python3-sopel-component python3-sopel-event python3-sopel-hookable
Upgraded: 16 newly installed, 0 to remove and 0 not upgraded,
and to free 1552 kB of archives.
For this operation, 7681 kB of additional disk space will be used.
t:1 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu jessie/universe augeas augeas-lenses all 1.12.0-1 [322 kB]
t:2 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu jessie/universe amd64 libaugeas0 amd64 1.12.0-1 [260 kB]
t:3 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu jessie/universe augeas python3-jcrapy all 1.10.0-1 [32.0 kB]
t:4 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu jessie/main amd64 python3-requests-toolbelt all 0.9.1-1 [38.0 kB]
t:5 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu jessie/main amd64 python3-rfc3339 all 1.1-3 [7110 B]
t:6 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu jessie-updates/universe amd64 python3-augparse all 1.21.0~ubuntu0.1 [16.4 kB]
t:7 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu jessie/universe amd64 python3-augeas all 0.5.0-1.1 [9124 B]
t:8 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu jessie/universe amd64 python3-configparser all 1.6.3-1 [34.9 kB]
t:9 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu jessie/universe amd64 python3-pexpect all 2.4-2 [33.9 kB]
t:10 http://eu-north-1.ec2.archive.ubuntu.com/ubuntu jessie/universe amd64 python3-sopel-hookable amd64 0.1.0~ubuntu0.1 [11.8 kB]
```

Obtention du certificat

Processus d'installation de Let's Encrypt

Pour sécuriser notre site, nous devons obtenir un certificat Let's Encrypt, ce qui garantit le chiffrement des données et l'authentification du serveur, améliorant ainsi la sécurité globale.

```
successfully received certificate.
Certificate is saved at: /etc/letsencrypt/live/monserveur-tcp.duckdns.org
Key is saved at:          /etc/letsencrypt/live/monserveur-tcp.duckdns.org
This certificate expires on 2026-05-25.
These files will be updated when the certificate renewals.
The bot has set up a scheduled task to automatically renew this certificate.

Deploying certificate
Successfully deployed certificate for monserveur-tcp.duckdns.org to /etc/
Congratulations! You have successfully enabled HTTPS on https://monserveur-tcp.duckdns.org

-----[REDACTED]-----
If you like Certbot, please consider supporting our work by
making a donation to ISRG / Let's Encrypt:           https://letsencrypt.org/donate
making a donation to EFF:                           https://eff.org/donate-isrg
-----[REDACTED]-----

pi@ip-172-31-20-111:~$
```

Test de sécurisation

Validation de la connexion HTTPS

Après vérification, nous constatons que le site est maintenant **sécurisé** grâce à la présence d'un cadenas dans la barre d'adresse, garantissant ainsi la **protection des données** échangées.



Apache2 Default P

It works!

This is the default welcome page used to test the correct operation of the installation on Ubuntu systems. It is based on the equivalent page on Debian packaging is derived. If you can read this page, it means that the Apache server at this site is working properly. You should replace this file (located at /var/www/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

En conclusion, nous avons vu que :

- HTTP permet la communication web mais n'est pas sécurisé.
- HTTPS ajoute une couche de chiffrement grâce aux certificats SSL/TLS.
- Apache et Nginx permettent d'héberger un site.
- Let's Encrypt facilite la sécurisation gratuitement.



Merci
beaucoup
pour votre
attention

COPIER LE LIEN SUIVANT
POUR LE GUIDE TOTAL DU
PROJET

<https://github.com/Diallo273/Services-Web-HTTP-HTTPS-.git>