

Rapport de compte rendu TP2 - Consultant Junior

Ce TP nous a permis d'explorer deux approches de déploiement d'infrastructure : le cloud public avec Microsoft Azure et le cloud privé avec Proxmox. L'objectif était de comparer ces solutions en conditions réelles.

Section 1 : Déploiement Cloud Public Microsoft Azure

Architecture déployée sur Azure

Création d'une instance Ubuntu avec 8Go de RAM, bridgé et 2 VCPUs.

On peut y voir les ressources allouées à la VM, la configuration réseau, les différentes sections de configuration de la VM (ACLs)

The screenshot displays the Azure portal interface for a virtual machine named 'myVm-maurer'. The left sidebar shows navigation options like 'Vue d'ensemble', 'Journal d'activité', 'Contrôle d'accès (IAM)', 'Étiquettes', 'Diagnostic et résolution des problèmes', 'Visualiseur de ressources', 'Se connecter', 'Mise en réseau', 'Paramètres', 'Disponibilité + mise à l'échelle', 'Sécurité', 'Sauvegarde + récupération d'urgence', 'Opérations', 'Supervision', 'Automatisation', and 'Aide'. The main area is divided into several sections:

- Bases:**
 - Groupe de res.: [TP_CLOUD](#)
 - Statut: En cours d'exécution
 - Emplacement: France Central (Zone 3)
 - Abonnement: [Azure for Students](#)
 - ID d'abonnement: e7e96a02-f6e7-4cb3-91be-808dcfd6756
 - Zone de disponibilité: 3
 - Étiquettes: [Ajouter des étiquettes](#)
- Propriétés:**
 - Machine virtuelle:**
 - Nom de l'ordinateur: myVm-maurer
 - Système d'exploitation: Linux (ubuntu 24.04)
 - Génération de machine virtuelle: V2
 - Architecture de machine virtuelle: x64
 - État de l'agent: Ready
 - Version de l'agent: 2.15.0.1
 - Mise en veille prolongée: Désactivé
 - Groupe hôte: -
 - Hôte: -
 - Groupe de placement de proximité: -
- Mise en réseau:**
 - Adresse IP publique: 40.66.42.255 (Interface réseau myVm-maurer644_z3)
 - 1 adresses IP publiques associées
 - Adresse IP publique (IPv6): -
 - Adresse IP privée: 10.0.0.4
 - Adresse IP privée (IPv6): -
 - Réseau/sous-réseau virtuel: myVm-maurer-vnet/default
 - Nom DNS: Configurer
- Taille:**
 - Taille: Standard D2s v3

Première connexion SSH

Il est nécessaire d'ajouter la clé SSH privé de la VM avec la commande pour pouvoir s'y connecter directement et plus facilement :

```
sudo chmod 600 myVm-maurer_key.pem && ssh-add myVm-maurer_key.pem
```

Après avoir déployé l'instance Azure et ajouté la clé publique nous nous connectons avec la commande.

```
ssh azureuser@<ip-public> # -i si on veut utiliser la clé privé directement
```

```
id001@id001-ThinkPad-T470:~/Téléchargements$ ssh azureuser@40.66.42.255
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.14.0-1017-azure x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

System information as of Mon Jan 12 11:51:37 UTC 2026

System load:  0.15          Processes:            130
Usage of /:   5.6% of 28.02GB Users logged in:           0
Memory usage: 3%          IPv4 address for eth0: 10.0.0.4
Swap usage:   0%
```

Installation d'Apache2

Nous installons le serveur web Apache2, nous activons le service au démarrage de la VM et nous le démarrons.

```
azureuser@myVm-maurer:~$ sudo apt install -y apache2
sudo systemctl start apache2
sudo systemctl enable apache2
sudo systemctl status apache2
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
apache2 is already the newest version (2.4.58-1ubuntu8.8).
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
Synchronizing state of apache2.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: enabled)
   Active: active (running) since Mon 2026-01-12 11:52:08 UTC; 1min 19s ago
     Docs: https://httpd.apache.org/docs/2.4/
  Main PID: 2425 (apache2)
    Tasks: 55 (limit: 9447)
   Memory: 5.4M (peak: 5.7M)
      CPU: 76ms
   CGroup: /system.slice/apache2.service
           └─2425 /usr/sbin/apache2 -k start
             └─2428 /usr/sbin/apache2 -k start
               └─2429 /usr/sbin/apache2 -k start

Jan 12 11:52:08 myVm-maurer systemd[1]: Starting apache2.service - The Apache HTTP Server...
Jan 12 11:52:08 myVm-maurer systemd[1]: Started apache2.service - The Apache HTTP Server.
azureuser@myVm-maurer:~$
```

Nous voyons que le service est correctement configuré et ne présente aucun défaut de configuration.

Règles de sécurité réseau

Dans Azure, la gestion du trafic réseau se fait à l'aide de groupes de sécurité réseau (NSG) qui permettent de définir des règles entrantes (inbound) et sortantes (outbound).

- **Règles entrantes (Inbound)** : contrôlent le trafic provenant de l'extérieur vers la VM.
- **Règles sortantes (Outbound)** : contrôlent le trafic émis par la VM vers l'extérieur.

Voici un tableau récapitulatif des règles configurées :

Sens	Protocole	Port(s)	Source/Destination	Description
Entrant	TCP	22	Any	SSH (accès à distance)
Entrant	TCP	80	Any	HTTP (serveur web)

Sens	Protocole	Port(s)	Source/Destination	Description
Entrant	TCP	443	Any	HTTPS (serveur web sécurisé)
Entrant	UDP/TCP	53	Any	DNS (résolution de noms)
Entrant	ICMP	Any	Any	ICMP (ping, diagnostic réseau)
Sortant	TCP	80	Any	HTTP (accès web sortant)
Sortant	TCP	443	Any	HTTPS (accès web sécurisé sortant)
Sortant	UDP/TCP	53	Any	DNS (résolution de noms sortante)

Différence entre inbound et outbound :

- Les règles **inbound** protègent la VM contre les accès non autorisés depuis Internet ou d'autres réseaux, en ne laissant passer que les ports/services nécessaires.
- Les règles **outbound** limitent les connexions que la VM peut initier vers l'extérieur, ce qui permet de contrôler et sécuriser les flux sortants (par exemple, empêcher la VM de communiquer avec des sites non approuvés).

Cette séparation permet de renforcer la sécurité en ne laissant passer que le trafic strictement nécessaire, aussi bien en entrée qu'en sortie.

Rechercher des règles

Source == tout

Destination == tout

Protocole == tout

Action == tout

Port == tout

Priorité ↑	Nom	Port	Protocole	Source	Destination	Action
▼	Règles des ports d'entrée (9)					
300	SSH	22	TCP	Tous	Tous	Allow
310	HTTP	80	TCP	Tous	Tous	Allow
320	HTTPS	443	TCP	Tous	Tous	Allow
330	DNS-UDP	53	UDP	Tous	Tous	Allow
340	DNS-TCP	53	TCP	Tous	Tous	Allow
350	ICMP	Tous	ICMP	Tous	Tous	Allow
65000	AllowVnetInBound	Tous	Tous	VirtualNetwork	VirtualNetwork	Allow
65001	AllowAzureLoadBalancerInBound	Tous	Tous	AzureLoadBalancer	Tous	Allow
65500	DenyAllInBound	Tous	Tous	Tous	Tous	Deny
▼	Règles de port de sortie (7)					
360	HTTP-1	80	TCP	Tous	Tous	Allow
370	HTTPS-1	443	TCP	Tous	Tous	Allow
380	DNS-TCP-1	53	TCP	Tous	Tous	Allow
390	DNS-TCP-UDP	53	UDP	Tous	Tous	Allow
65000	AllowVnetOutBound	Tous	Tous	VirtualNetwork	VirtualNetwork	Allow
65001	AllowInternetOutBound	Tous	Tous	Tous	Internet	Allow
65500	DenyAllOutBound	Tous	Tous	Tous	Tous	Deny

Configuration du nom de domaine

Nous avons créé un sous-domaine sur DuckDNS.org (api-maurer.duckdns.org) pointant vers mon IP publique 40.66.42.255. La propagation DNS a pris environ 2 minutes.

success: ip address for **api-maurer.duckdns.org** updated to **40.66.42.255**

domains 1/5

http:// .duckdns.org

domain	current ip	ipv6	changed
api-maurer	<input type="text" value="40.66.42.255"/> <input type="button" value="update ip"/>	<input type="text" value="ipv6 address"/> <input type="button" value="update ipv6"/>	0 seconds ago <input type="button" value="delete domain"/>

This site is protected by reCAPTCHA and the Google [Privacy Policy](#) and [Terms of Service](#) apply.

Nous voyons bien que le sous-domaine pointe vers l'IP publique de la VM Azure.

Mise en place du certificat SSL

Le réseau de l'IUT bloque les connexions HTTP vers des IP publiques. nous avons donc généré un certificat SSL auto-signé pour passer en HTTPS.



FortiGuard Intrusion Prevention - Access Blocked

Web Page Blocked

You have tried to access a web page that is in violation of your Internet usage policy.

Category Dynamic DNS

URL <http://api-maurer.duckdns.org/>

To have the rating of this web page re-evaluated [please click here](#).

Mise en place des certificats sur la VM Azure :

```
sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 \  
-keyout /etc/ssl/private/api-maurer.key \  
-out /etc/ssl/certs/api-maurer.crt \  
-subj "/C=FR/ST=France/L=Paris/O=MyOrg/CN=api-maurer.duckdns.org"
```

Configuration Apache pour SSL :

```
sudo nano /etc/apache2/sites-available/api-maurer-ssl.conf
```

```
<VirtualHost *:443>  
    ServerName api-maurer.duckdns.org  
    DocumentRoot /var/www/html  
  
    SSLEngine on
```

```

SSLCertificateFile /etc/ssl/certs/api-maurer.crt
SSLCertificateKeyFile /etc/ssl/private/api-maurer.key

<Directory /var/www/html>
    Options Indexes FollowSymLinks
    AllowOverride All
    Require all granted
</Directory>

ErrorLog ${APACHE_LOG_DIR}/ssl_error.log
CustomLog ${APACHE_LOG_DIR}/ssl_access.log combined
</VirtualHost>

```

Activation des modules SSL et du site sur apache2 :

```

sudo a2enmod ssl
sudo a2ensite api-maurer-ssl

```

```

azureuser@myVm-maurer:~$ sudo a2enmod ssl
sudo a2ensite api-maurer-ssl
Considering dependency mime for ssl:
Module mime already enabled
Considering dependency socache_shmcb for ssl:
Enabling module socache_shmcb.
Enabling module ssl.
See /usr/share/doc/apache2/README.Debian.gz on how to configure SSL and create self-signed certificates.
To activate the new configuration, you need to run:
    systemctl restart apache2
Enabling site api-maurer-ssl.
To activate the new configuration, you need to run:
    systemctl reload apache2
azureuser@myVm-maurer:~$ sudo systemctl reload apache2

```

Vérifications et tests d'accès

Pour valider la configuration, plusieurs vérifications ont été réalisées :

- **Test CURL en HTTPS** : Un test avec `curl` a permis de vérifier l'accessibilité du site en HTTPS, confirmant la bonne installation du certificat SSL.

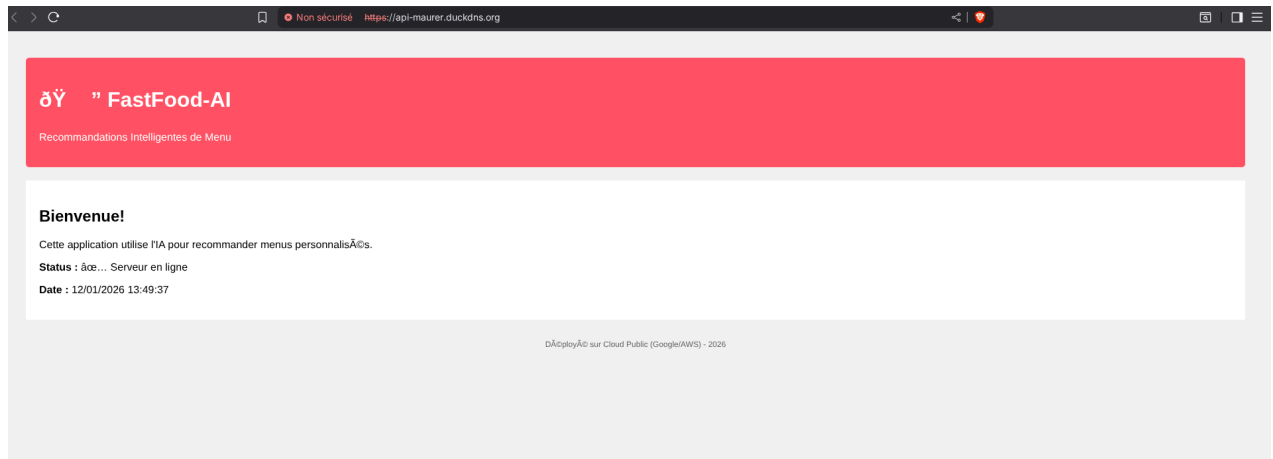
```

azureuser@myVm-maurer:~$ curl -k https://api-maurer.duckdns.org
<!DOCTYPE html>
<html>
<head>
  <title>MAURER - FastFood-AI | Menu Recommendations</title>
  <style>
    body { font-family: Arial; margin: 40px; background: #f0f0f0; }
    .header { background: #ff6b6b; color: white; padding: 20px; border-radius: 5px; }
    .content { background: white; padding: 20px; margin-top: 20px; }
    .footer { text-align: center; color: #666; margin-top: 30px; font-size: 12px; }
  </style>
</head>
<body>
  <div class="header">
    <h1>🍔 FastFood-AI</h1>
    <p>Recommandations Intelligentes de Menu</p>
  </div>
  <div class="content">
    <h2>Bienvenue!</h2>
    <p>Cette application utilise l'IA pour recommander menus personnalisés.</p>
    <p><strong>Status :</strong> ✅ Serveur en ligne</p>
    <p><strong>Date :</strong> <script>document.write(new Date().toLocaleString('fr-FR'));</script></p>
  </div>
  <div class="footer">
    <p>Déployé sur Cloud Public (Google/AWS) - 2026</p>
  </div>
</body>
</html>
azureuser@myVm-maurer:~$

```

- **Accès externe** : Un accès depuis un poste extérieur a été effectué pour s'assurer que le site est bien

accessible publiquement.



- **Temps de réponse** : Les mesures montrent des temps de réponse relativement longs, principalement dus aux ressources limitées de la VM et à l'utilisation d'un certificat auto-signé.

```
id001@id001-ThinkPad-T470:~/Téléchargements$ curl -w "Temps réponse : %{time_total}s\n" https://api-maurer.duckdns.org/
curl: (60) SSL certificate problem: self-signed certificate
More details here: https://curl.se/docs/sslcerts.html

curl failed to verify the legitimacy of the server and therefore could not
establish a secure connection to it. To learn more about this situation and
how to fix it, please visit the web page mentioned above.
Temps réponse : 11.730657s
id001@id001-ThinkPad-T470:~/Téléchargements$
```

- **Logs Apache** : L'analyse des logs HTTP confirme la réception correcte des requêtes et l'absence d'erreurs.

```
azureuser@myVm-maurer:~$ sudo tail -5 /var/log/apache2/access.log
192.42.116.192 - - [12/Jan/2026:13:16:47 +0000] "GET / HXXP/1.1" 400 482 "-" "-"
192.42.116.192 - - [12/Jan/2026:13:16:48 +0000] "GET /favicon.ico HTTP/1.1" 404 435 "-" "4Loo98b"
192.42.116.192 - - [12/Jan/2026:13:16:49 +0000] "GET /robots.txt HTTP/1.1" 404 435 "-" "jJvi0qxkHIFH"
192.42.116.192 - - [12/Jan/2026:13:16:53 +0000] "HEAD / HTTP/1.1" 200 253 "-" "yHC3R4TzvdYK9y"
192.42.116.192 - - [12/Jan/2026:13:16:54 +0000] "DELETE / HTTP/1.1" 405 501 "-" "ffYiaVaK6DA8kys"
azureuser@myVm-maurer:~$
```

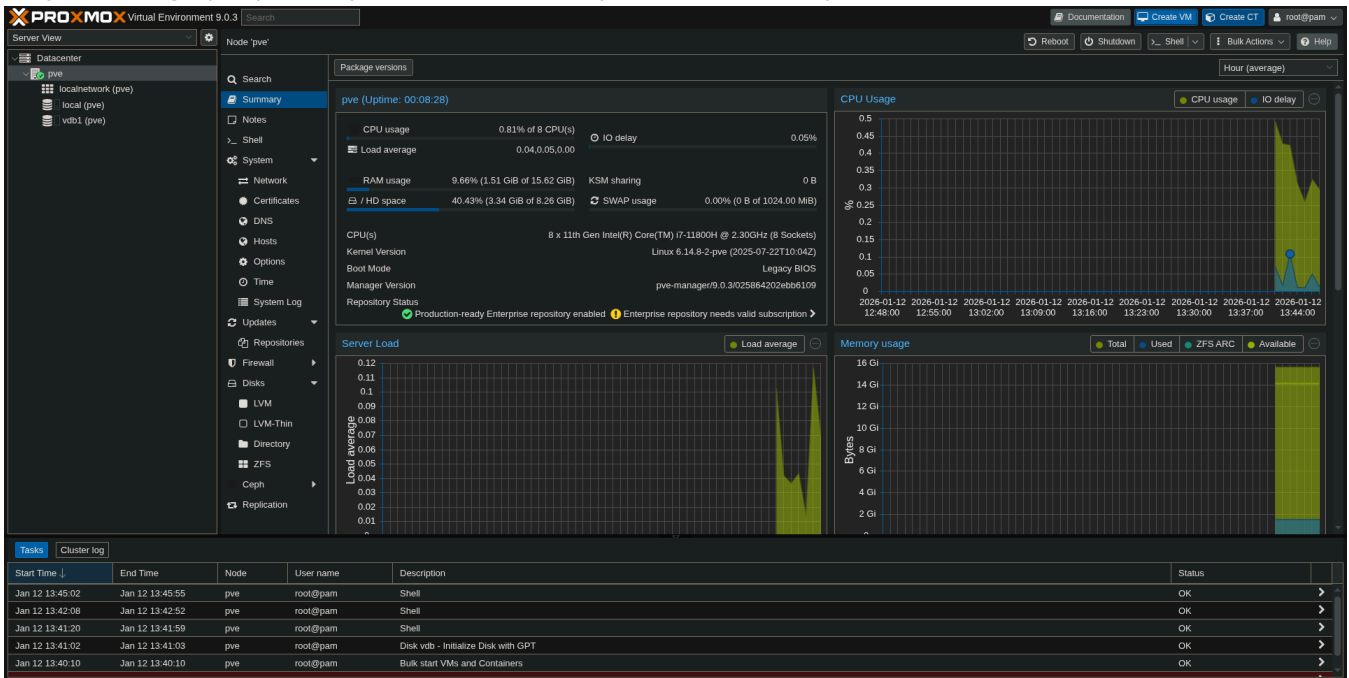
Ces vérifications attestent du bon fonctionnement du service web déployé sur Azure, malgré quelques limitations de performance liées à la configuration matérielle.

Section 2 : Déploiement On-premise avec Proxmox

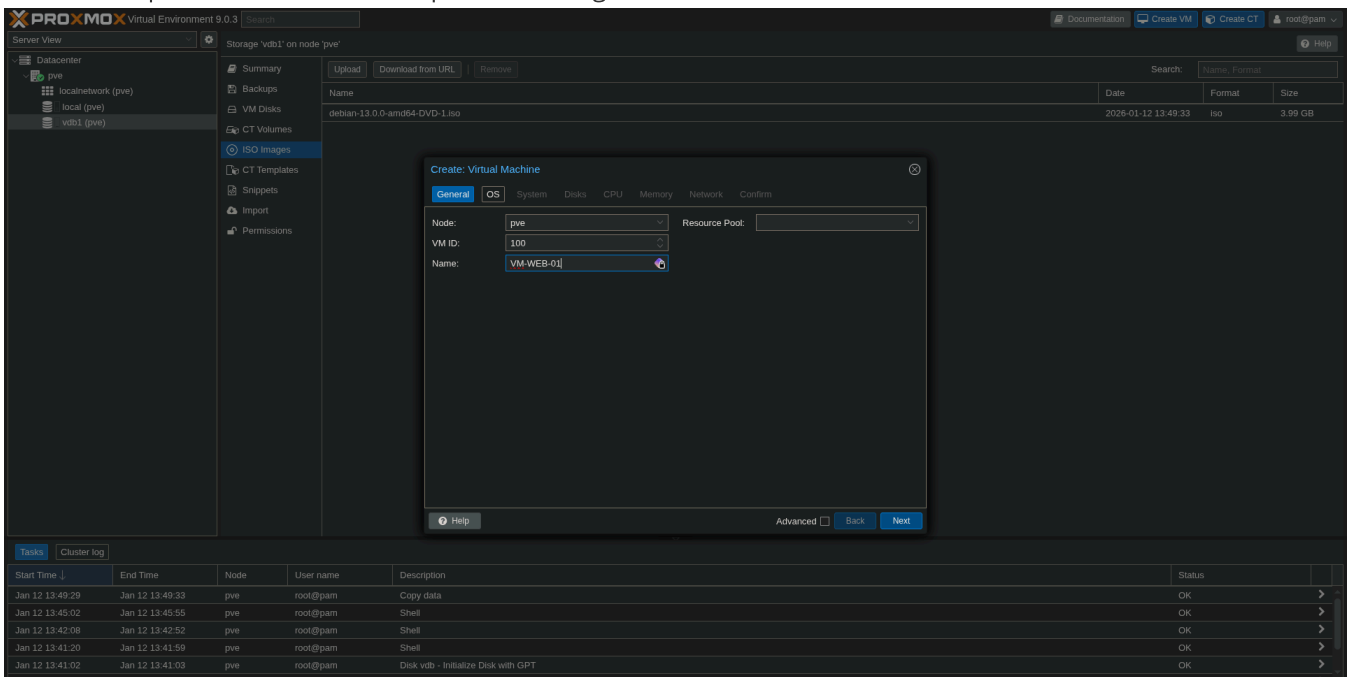
Nous avons installé Proxmox 9.0 sur une machine disposant de 16 Go de RAM et 8 vCPU. Le disque système fait 10 Go et celui pour les VMs 100 Go.

Étapes réalisées : upload de l'ISO Debian, création de la VM, connexion SSH, passage root, nettoyage du sources.list, mise à jour des paquets, installation et activation d'Apache2. L'ensemble a pris environ 10 minutes, sans difficulté particulière.

On y voit des graphiques de performance en temps réel du serveur proxmox.



Voici une capture d'écran d'une étape de la configuration réalisée d'une VM Debian sur Proxmox :



Voici les étapes d'installation et d'activation du service Apache2 sur la VM Debian :

```
root@VM-WEB-01:~# apt install apache2
Installation de :
  apache2

Installation de dépendances :
  apache2-bin          libaprutil1t64  librtmp1
  apache2-data         libcurl4t64     libsasl2-2
  apache2-utils        libldap-common  libsasl2-modules
  libapr1t64           libldap2         libsasl2-modules-db
  libaprutil1-dbd-sqlite3 liblua5.4-0     libssh2-1t64
  libaprutil1-ldap     libnghttp3-9    ssl-cert

Paquets suggérés :
  apache2-doc          libsasl2-modules-gssapi-mit
  apache2-suexec-pristine | libsasl2-modules-gssapi-heimdal
  | apache2-suexec-custom libsasl2-modules-ldap
  ufw                  libsasl2-modules-otp
  www-browser          libsasl2-modules-sql

Sommaire :
  Mise à niveau de : 0. Installation de : 19Supprimé : 0. Non mis à jour : 0
  Taille du téléchargement : 3 529 kB
  Espace nécessaire : 11,6 MB / 7 427 MB disponible

Continuer ? [O/n] Y
```

Modification de la page d'accueil :

```
sudo nano /var/www/html/index.html
```

```
<!DOCTYPE html>
<html>
<head>
  <title>MAURER - FastFood-AI | Menu Recommendations</title>
  <style>
    body { font-family: Arial; margin: 40px; background: #f0f0f0; }
    .header { background: #ff6b6b; color: white; padding: 20px; border-radius: 5px; }
    .content { background: white; padding: 20px; margin-top: 20px; }
    .footer { text-align: center; color: #666; margin-top: 30px; font-size: 12px; }
  </style>
</head>
<body>
  <div class="header">
    <h1>🍔 FastFood-AI</h1>
    <p>Recommandations Intelligentes de Menu</p>
  </div>
  <div class="content">
    <h2>Bienvenue!</h2>
    <p>Cette application utilise l'IA pour recommander menus personnalisés.</p>
    <p><strong>Status :</strong> ✅ Serveur en ligne</p>
  </div>
</body>
</html>
```

```

<p><strong>Date :</strong> <script>document.write(new Date().toLocaleString('fr-
FR'));</script></p>
</div>
<div class="footer">
  <p>Déployé sur Cloud Public (Google/AWS) - 2026</p>
</div>
</body>
</html>

```

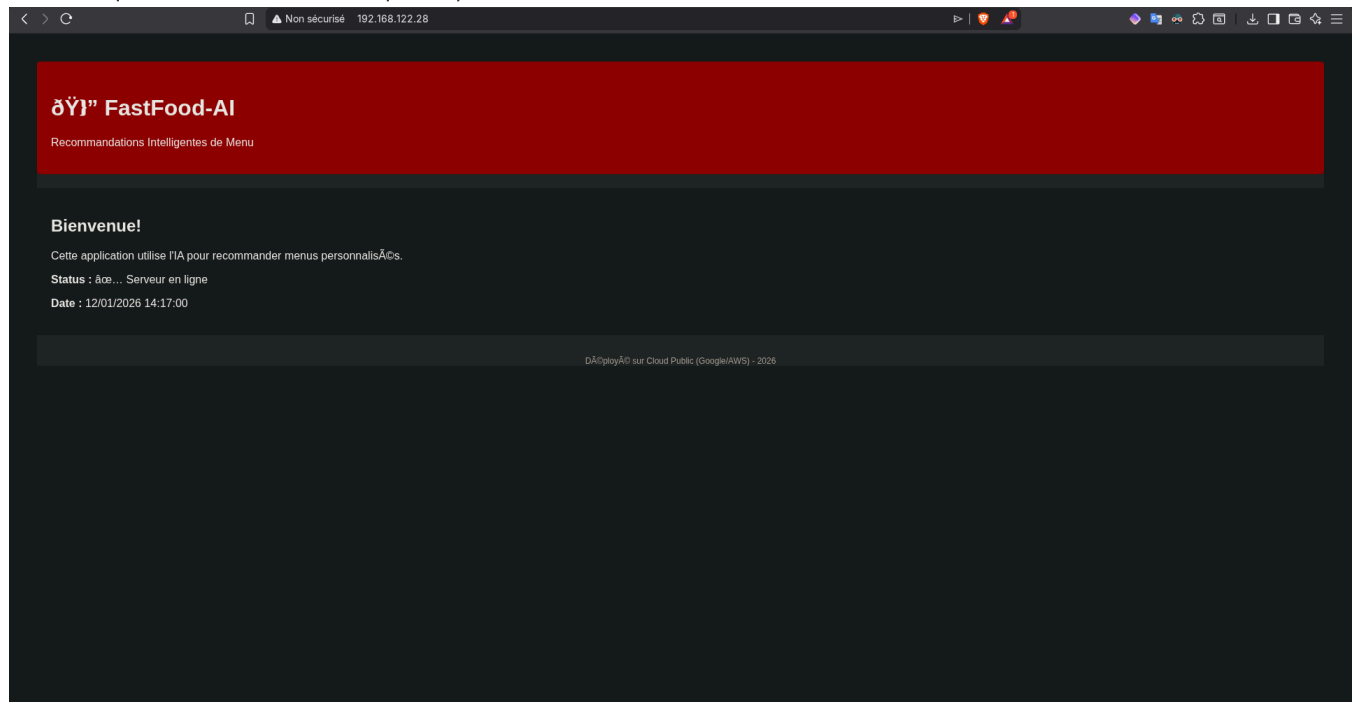
Activation du service Apache2 :

```

root@VM-WEB-01:~# systemctl enable --now apache2
Synchronizing state of apache2.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable apache2

```

Accès depuis l'extérieur via l'IP publique (IP du serveur Proxmox) :



Les temps de réponse sont meilleurs (0.002446s) et la stabilité d'Apache est comparable au cloud public.

Section 3 : Résultats des performances

Performances sur Proxmox (Cloud Privé)

Voici l'utilisation des ressources pour la VM Debian déployée sur Proxmox :



- **Temps de réponse mesuré** : 0.002446 secondes pour une requête HTTP simple.
- **Stabilité** : Le service Apache2 est resté stable durant toute la période de test, sans interruption ni surcharge.
- **Utilisation des ressources** : La VM consomme peu de CPU et de mémoire, ce qui laisse de la marge pour d'autres services ou une montée en charge.

Performances sur Azure (Cloud Public)

- **Temps de réponse mesuré** : Les temps de réponse sont plus élevés (voir Section 1), principalement à cause des ressources limitées de la VM et du certificat SSL auto-signé.
- **Stabilité** : Apache2 est stable, mais la latence est plus variable selon la charge du cloud public et la configuration réseau.

Synthèse comparative

- **Cloud privé (Proxmox)** : Excellente réactivité, contrôle total sur les ressources, stabilité au rendez-vous.
- **Cloud public (Azure)** : Facilité de déploiement, mais performances dépendantes du type de VM choisi et de la configuration réseau.

En conclusion, pour des besoins de performance immédiate et un contrôle précis des ressources, le cloud privé offre de meilleurs résultats. Le cloud public reste pertinent pour sa flexibilité et sa simplicité de gestion, au prix d'une latence parfois supérieure.

Section 4 : Comparaison des deux solutions

Critère	Cloud Public (Azure)	Cloud Privé (Proxmox)	Gagnant
Temps de déploiement	30 min	30 min	Égalité
Complexité	Simple	Moyenne	Cloud Public
Contrôle infrastructure	Limité	Total	Cloud Privé
Scalabilité	Rapide	Lente	Cloud Public
Documentation	Support (Microsoft)	Forum communautaire	Cloud Public
Coût initial	0€	~50€ matériel	Cloud Public
Coût mensuel	20-50€	5-10€	Cloud Privé
Configuration réseau	Moyenne	Simple	Cloud Privé
Sauvegardes	Automatisables	Manuelles	Égalité
Mise en production	Rapide	Plus longue	Cloud Public

Analyse des coûts

Pour le cloud public (Azure), il n'y a pas d'investissement matériel initial à prévoir. Le coût mensuel pour une machine virtuelle Ubuntu, incluant le stockage, les transferts et la gestion du DNS, s'élève à environ 32€. Sur une période de cinq ans, cela représente un total de 1 920€.

En ce qui concerne le cloud privé (Proxmox), il faut compter un investissement de départ, si on prend un exemple environ 3 200€ pour l'achat du serveur, du switch et de l'onduleur. À cela s'ajoutent des coûts mensuels estimés à 92€, couvrant l'électricité, l'accès à Internet et la maintenance, soit un total de 8 720€ sur cinq ans.

Ainsi, pour une petite infrastructure, le cloud public s'avère plus économique sur la durée de cinq ans.

Recommandation pour une startup

Pour une jeune entreprise comme FastFood-AI, le cloud public est plus adapté car :

- Aucun investissement matériel de départ
- Déploiement rapide pour tester le marché
- Capacité à augmenter les ressources selon la demande
- Pas besoin d'équipe système dédiée
- Haute disponibilité garantie

Le cloud privé devient intéressant quand l'entreprise grandit et a besoin de plus de contrôle sur ses données, ou si elle dispose déjà d'un datacenter.

Lien du repository GitHub : <https://github.com/LOIC-only-one/Cloud-TP-BUT3/tree/main/rapport/tp2>