

Année 2024-2025

Annexe Technique



Déploiement



Benoît ROBART

Table des matières – Dossier Déploiement

Sommaire

1. Introduction et Objectifs	page 3
• 1.1. Déploiement dans le Cloud du fournisseur Infomaniak	page 3
• 1.2. Architecture cible : nom de domaine, IP, service attendu	page 3
2. DNS, domaine et IP publique	page 4
• 2.1. Liaison domaine/IP dans Infomaniak	page 4
• 2.2. Validation DNS avec <code>dig</code> et <code>nslookup</code>	page 5
• 2.3. Propagation DNS : tests, délais, résolution	page 6 - 7
3. Installation de Jitsi Meet	page 7
• 3.1. Préparation système et clés GPG	page 7
• 3.2. Dépôt officiel et installation Jitsi	page 7 - 8
• 3.3. Vérification des services : <code>jicofo</code> , <code>prosody</code> , <code>videobridge</code>	page 8
4. Configuration du nom de domaine dans Prosody	page 9
• 4.1. Fichier <code>visio.workeezconnect.fr.cfg.lua</code>	page 9 - 10
• 4.2. Modules supplémentaires Prosody (ex: <code>speakerstats</code>)	page 10
• 4.3. Création utilisateur XMPP	page 10
5. Résolution des erreurs fréquentes	page 10
• 5.1. Conflits <code>localhost</code> vs nom de domaine réel	page 10 - 11
• 5.2. Erreur <code>localhost.cfg.lua</code> manquant	page 11
• 5.3. Correction <code>dpkg</code> et configuration Apache	
6. Remplacement d'Apache par NGINX	page 12
• 6.1. Suppression Apache & installation NGINX	page 12
• 6.2. Configuration de NGINX pour Jitsi	page 12 - 13
• 6.3. Activation HTTPS avec Certbot	page 13 - 15
• 6.4. Génération du certificat SSL (Let's Encrypt)	page 15
7. Tests fonctionnels de déploiement	page 15
• 7.1. Accès HTTPS depuis navigateur	page 15 - 16
• 7.2. Vérifications : ports, pare-feu, DNS, <code>curl</code> , <code>telnet</code> , <code>openssl</code>	page 16
• 7.3. Vérification de la chaîne SSL complète	page 17 - 19

1. Introduction et Objectifs :

1/ Lien avec le projet global.

Déploiement dans le Cloud du fournisseur Infomaniak

<https://www.infomaniak.com/fr/support/faq/2025/lier-un-nom-de-domaine-a-un-hebergement-web-infomaniak>

Nom de domaine fixé : <https://visio.workeezconnect.fr/>

Adresse IP fixée : 37.156.46.238 // chaque nouvelle instance donne un nouvel IP.

Nous garderons celle-ci car l'on va shut off l'instance lorsque que l'on arrête son accessibilité.

Explication théorique claire :

Élément

Rôle

DNS (A record)

Associe visio.workeezconnect.fr → 37.156.46.238

Certificat SSL (Let's Encrypt)

Prouve que le serveur web contrôle réellement ce domaine

ACME challenge

Mécanisme utilisé pour prouver que tu possèdes le domaine

1.2. Architecture cible : nom de domaine, IP, service attendu

Avant même l'installation du certificat il faut lier domaine et l'IP : Interface Infomaniak

Interface Infomaniak : DNS Zones Create Zone

The screenshot shows the 'Create Record Set' form in the Infomaniak DNS interface. The form is titled 'Create Record Set' and has a close button (X) in the top right corner. It contains the following fields and options:

- Type ***: A dropdown menu with 'A - Address record' selected.
- Name ***: A text input field containing 'visio.' with a green checkmark icon on the right. Below it, a small text says 'DNS name for the record set, ending in '.''.
- Description**: A text input field containing 'visio-conférence device'.
- TTL ***: A text input field containing '300' with a green checkmark icon on the right. Below it, a small text says 'Time To Live in seconds.'.
- Records**: A section with a 'Record' label and a text input field containing '84.234.29.126' with a green checkmark icon on the right. Below it, a small text says 'Records for the record set.'.
- Buttons**: At the bottom, there are three buttons: 'Cancel' (with an X icon), 'Add Record' (with a plus icon), and 'Submit' (with a checkmark icon).

2. DNS, domaine et IP publique

2.1. Liaison domaine/IP dans Infomaniak

workeezconnect.fr.

Create Record Set

OverviewRecord Sets

Details

Attributes

IDbf01ffa4-f9d8-4d93-a067-7b4021738474

Descriptionvisio-conférence

TypePrimary

StatusActive

ActionNone

Emailadmin@workeezconnect.fr

Serial1751537163

Time To Live3600

Version-

Attributes

Modification Times

Associations

Created At2025-07-03T09:56:50.000000

Updated At2025-07-03T10:08:08.000000

Transferred At-

Pool ID794ccc2c-d751-44fe-b57f-8894c9f5c842

Project ID7646a67b63144f89bb5938cff2ccba60

Masters

Project / DNS / Zones

Back

workeezconnect.fr.

Create Record Set

OverviewRecord Sets

Q Click here for filters or full text search.

Displaying 3 items

Name ^	Type	Records	Status
> visio.workeezconnect.fr.	A - Address record	84.234.29.126	Active <div>Update</div>
> workeezconnect.fr.	NS - Name server	ns1.pub2.infomaniak.cloud., ns2.pub2.infomaniak.cloud.	Active
> workeezconnect.fr.	SOA - Start of authority record	ns1.pub2.infomaniak.cloud. admin.workeezconnect.fr. 1751537163 3534 600 86400 3600	Active

Displaying 3 items

workeezconnect.fr.

Create Record Set

OverviewRecord Sets

Details

Attributes

IDbf01ffa4-f9d8-4d93-a067-7b4021738474

Descriptionvisio-conférence

TypePrimary

StatusActive

ActionNone

Emailadmin@workeezconnect.fr

Serial1751537163

Time To Live3600

Version-

Attributes

Modification Times

Associations

Created At2025-07-03T09:56:50.000000

Updated At2025-07-03T10:08:08.000000

Transferred At-

Pool ID794ccc2c-d751-44fe-b57f-8894c9f5c842

Project ID7646a67b63144f89bb5938cff2ccba60

Masters

2.2. Validation DNS avec dig et nslookup

Détails des créations :

workeezconnect.fr. Create Record Set

Overview **Record Sets**

Q Click here for filters or full text search. X

Displaying 3 items

Name	Type	Records	Status
visio.workeezconnect.fr.	A - Address record	84.234.29.126	Active Update
Notes None		ID 3941a16d-eda6-4e93-ab9b-2976d187aa8f	
Description visio-conférence			
workeezconnect.fr.	NS - Name server	ns1.pub2.infomaniak.cloud., ns2.pub2.infomaniak.cloud.	Active
Notes None		ID 69267247-8ff5-4f60-8371-cbfc161cc354	
Description None			
workeezconnect.fr.	SOA - Start of authority record	ns1.pub2.infomaniak.cloud. admin.workeezconnect.fr. 1751537163 3534 600 86400 3600	Active
Notes None		ID b7a0e55a-3048-4630-bd50-30f50f55fd1e	
Description None			

Displaying 3 items

Tests avec : dig workeezconnect.fr NS +short

- **dig** : commande DNS utilisée pour interroger les serveurs DNS.
- **workeezconnect.fr** : le nom de domaine que tu veux interroger.
- **NS** : signifie *Name Server*, c'est-à-dire que tu demandes les serveurs de noms responsables de ce domaine.
- **+short** : format de sortie simplifié (juste la réponse, pas les détails).

```
test@KUS-F-STAGE:/mnt/c/Windows/System32$ dig +trace visio.workeezconnect.fr
;; communications error to 10.255.255.254#53: timed out
;; communications error to 10.255.255.254#53: timed out
;; communications error to 10.255.255.254#53: timed out

; <<>> DiG 9.18.30-Ubuntu0.24.04.2-Ubuntu <<>> +trace visio.workeezconnect.fr
;; global options: +cmd
;; no servers could be reached
test@KUS-F-STAGE:/mnt/c/Windows/System32$ dig visio.workeezconnect.fr @ns1.pub2.infomaniak.cloud

; <<>> DiG 9.18.30-Ubuntu0.24.04.2-Ubuntu <<>> visio.workeezconnect.fr @ns1.pub2.infomaniak.cloud
;; global options: +cmd
;; Got answer:
;; ->HEADER<- opcode: QUERY, status: NXDOMAIN, id: 6766
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;visio.workeezconnect.fr.      IN      A

;; Query time: 0 msec
;; SERVER: 83.166.143.128#53(ns1.pub2.infomaniak.cloud) (UDP)
;; WHEN: Thu Jul 03 13:17:46 CEST 2025
;; MSG SIZE rcvd: 52

test@KUS-F-STAGE:/mnt/c/Windows/System32$ dig NS workeezconnect.fr +short
marjory.ns.cloudflare.com.
ajay.ns.cloudflare.com.
test@KUS-F-STAGE:/mnt/c/Windows/System32$
```

2.3. Propagation DNS : tests, délais, résolution

Problèmes : les tests de connexions et d'identifications ont échoué, les serveurs sont des **cloudflares** non des **infomaniak**. Il faudra changer via l'accès client permettant la création de nom de domaines pour switcher les serveurs.

Actions requises :

Action	Où ?	Objectif
Modifier les serveurs de noms NS	Chez le registrar du domaine (OVH, Infomaniak, Gandi, etc.)	Remplacer *.ns.cloudflare.com → ns1.pub2.infomaniak.cloud
Attendre propagation DNS	Internet (1h à 24h)	Reflète les bons serveurs
Tester avec dig ou nslookup	Ton terminal	Confirmer propagation

Vérification après propagation : avec ces commandes et les réponses attendues

1. **dig NS workeezconnect.fr +short**
Doit répondre avec :
ns1.pub2.infomaniak.cloud.
ns2.pub2.infomaniak.cloud.
2. **dig visio.workeezconnect.fr +short**
Doit renvoyer :
84.234.29.126

Délai de propagation mondial réaliste.

Niveau de cache / infrastructure	Délai estimé max
Serveurs racines & TLD (.fr)	15 à 30 min
Résolveurs publics (Google 8.8.8.8, etc.)	15 à 60 min
CDN (Cloudflare, Akamai...)	1 à 2 h
FAI (France, Europe, Afrique...)	2 à 24 h
Pays lointains (zones isolées, Asie, etc.)	24 à 48 h max

Test : Réussi

```
ubuntu@jjitsi-tercium:~$ dig +short visio.workeezconnect.fr
84.234.29.126
```

2nd Test : Réussi

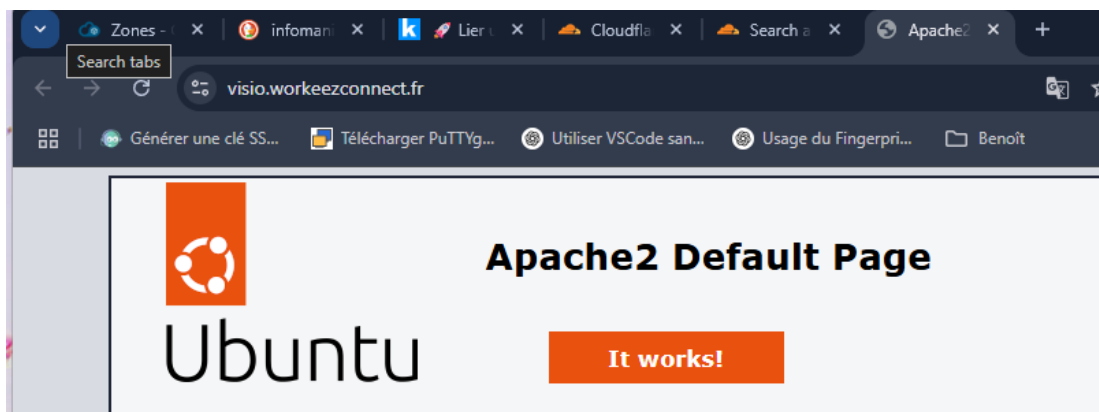
```
ubuntu@jitsi-tercium:~$ curl http://visio.workeezconnect.fr
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <!--
    Modified from the Debian original for Ubuntu
    Last updated: 2022-03-22
    See: https://launchpad.net/bugs/1966084
  -->
```

Cependant : le test avec le navigateur accède au site avec une non secure alert.

Relance donc de la création du certificat :

cd : sudo certbot --apache -d visio.workeezconnect.fr

Création réussie :



3. Installation de Jitsi Meet

3.1. Préparation système et clés GPG

Relions maintenant le serveur Jitsi à notre instance domaine + Adresse IP

Se connecter à son dépôt via le dossier où se situent les clefs privée et publique :

ssh -i Tercium-instance_key/tercium-instance_key ubuntu@37.156.46.238

3.2. Dépôt officiel et installation Jitsi

Ajouter le dépôt de [Jitsi Meet](#) :

Préambule : Le dépôt fourni par Jitsi utilise le chiffrement, donc vérifier que vous avez bien le paquet apt-transport-https.

cd : sudo apt install apt-transport-https ca-certificates curl gnupg -y

► apt-transport-https

Permet à apt (le gestionnaire de paquets) de télécharger des paquets depuis des dépôts HTTPS (sécurisés via TLS). Sans cela, apt ne pourra pas accéder à <https://download.jitsi.org>.

► **ca-certificates**

Installe les autorités de certification (CA) racines utilisées pour vérifier la validité des certificats SSL/TLS lors des connexions sécurisées. Indispensable pour vérifier le certificat du dépôt Jitsi, de Let's Encrypt, etc.

► **curl**

Outil de ligne de commande pour faire des requêtes HTTP(S), utilisé ici pour récupérer la clé GPG de Jitsi.

► **gnupg**

Contient gpg (GNU Privacy Guard), utilisé pour décrypter et gérer les clés de signature GPG, comme celle du dépôt Jitsi.

► **-y**

Flag qui force l'acceptation automatique de l'installation sans demander confirmation à chaque paquet. Très utilisé en script pour éviter l'interaction manuelle.

Importer la clé de manière sécurisée

```
curl https://download.jitsi.org/jitsi-key.gpg.key | gpg --dearmor | sudo tee /usr/share/keyrings/jitsi-keyring.gpg > /dev/null
```

Ajouter le dépôt en l'associant à la clé

```
echo "deb [signed-by=/usr/share/keyrings/jitsi-keyring.gpg] https://download.jitsi.org stable/" | sudo tee /etc/apt/sources.list.d/jitsi-stable.list > /dev/null
```

Mettre à jour les paquets

```
sudo apt update
```

Installation de Jitsi Meet

```
Sudo apt install jitsi-meet -y
```

Associer l'IP et le nom de domaine : Déjà réaliser !

"Si votre nom de domaine pointe déjà via DNS public vers l'adresse IP de l'instance, l'étape /etc/hosts est facultative."

3.3. Vérification des services : jicofo, prosody, videobridge

Vérifier l'état des services des serveurs

```
systemctl status jitsi-videobridge2
```

```
systemctl status jicofo
```

```
systemctl status prosody
```

Analyse DNS complète avec dig.

```
dig +short visio.workeezconnect.fr
```

Vérification DNS + HTTP avec curl.

```
curl -i http://visio.workeezconnect.fr
```

Vérification croisée : IP publique actuelle du serveur.

```
curl -4 ifconfig.me
```


4. Configuration du nom de domaine dans Prosody

4.1. Fichier visio.workeezconnect.fr.cfg.lua

Fichier à éditer : `sudo nano /etc/prosody/conf.d/visio.workeezconnect.fr.cfg.lua`

Code de base avec GNU nano 7.2 :

```
VirtualHost "visio.workeezconnect.fr"
  authentication = "internal_hashed"
  modules_enabled = {
    "bosh";
    "pubsub";
    "ping";
    "speakerstats";
    "conference_duration";
    "muc_lobby_rooms";
  }
```

☑ Ne rien ajouter après ce point-virgule dans la table !

Component "conference.visio.workeezconnect.fr" "muc"

Component "auth.visio.workeezconnect.fr"
authentication = "internal_hashed"

Component "focus.visio.workeezconnect.fr"
component_secret = "FOCUS_SECRET" **Il faudra ajouter en sus la possibilité d'inviter**

Redémarrage Prosody :

`sudo systemctl restart prosody`

Vérifier qu'il est actif :

`sudo systemctl status prosody`

Vérifier si le port 5280 est ouvert :

`sudo ss -tln | grep 5280`

Tester l'accès HTTP BOSH :

`curl http://localhost:5280/http-bind/`

Activer si nécessaire le Firewall sur le port (UFW doit être actif) :

`sudo ufw allow 5280/tcp`

Vérification de la configuration

`sudo prosodyctl check config`

4.2. Modules supplémentaires Prosody (ex: speakerstats)

1/ Il manque des modules :

modulemanager: Unable to load module 'conference_duration'

modulemanager: Unable to load module 'speakerstats'

modulemanager: Unable to load module 'muc_lobby_rooms'

On les installe avec : `sudo apt install prosody-modules`

On redémarre avec : `sudo systemctl restart prosody`

2/ Bug car il demande un localhost (qui pose un problème lors des connexions avec l'instance). L'erreur indique que le script post-installation de jitsi-meet-prosody échoue parce qu'il attend un fichier /etc/prosody/conf.avail/localhost.cfg.lua qui n'existe pas.

Création de ce fichier : localhost.cfg.lua
sudo nano /etc/prosody/conf.avail/localhost.cfg.lua

Contenu minimal du fichier (pour débloquer dpkg) :
VirtualHost "localhost"
authentication = "anonymous"

On relance la configuration des paquets cassés
sudo dpkg --configure -a

On vérifie l'installation
sudo apt install -f
sudo systemctl restart prosody jicofo jitsi-videobridge2 nginx

4.3. Création utilisateur XMPP

Créer un utilisateur XMPP :
sudo prosodyctl register admin visio.workeezconnect.fr motdepasse

5. Résolution des erreurs fréquentes

Cas Particulier source de problèmes :

5.1. Conflits localhost vs nom de domaine réel

Conflit localhost vs nom de domaine public

C'est le problème principal qui a perturbé Certbot / Let's Encrypt / Apache :

▣ Cause :

Jitsi installe par défaut un vhost Apache localhost.conf :

```
apache
CopyEdit
<VirtualHost *:80>
    ServerName localhost
    DocumentRoot "/usr/share/jitsi-meet"
</VirtualHost>
```

Simultanément, ton domaine réel (visio.workeezconnect.fr) est servi par :

- /etc/apache2/sites-enabled/000-default.conf (HTTP).
- /etc/apache2/sites-enabled/000-default-le-ssl.conf (HTTPS)

Certbot ou acme.sh essaie de **créer un challenge via le nom de domaine réel**, mais Apache répond par défaut avec la conf localhost, car elle est mal priorisée.

▣ Symptôme :

```
bash
CopyEdit
curl http://localhost/.well-known/acme-challenge/test.txt
```

donne une **redirection 301** → **HTTPS sur localhost**, ce qui empêche la validation http-01 par Let's Encrypt.

5.2. Erreur localhost.cfg.lua manquant

▣ Solutions recommandées :

1. **Désactiver localhost.conf** s'il n'est plus nécessaire :

```
bash
CopyEdit
sudo a2dissite localhost.conf
sudo systemctl reload apache2
```

2. **S'assurer que visio.workeezconnect.fr est le ServerName prioritaire** dans les fichiers 000-default.conf et 000-default-le-ssl.conf.
3. **Contrôler l'ordre de priorité via :**

```
bash
CopyEdit
sudo apachectl -S
```

Et **supprimer tout vhost résiduel localhost** qui interfère.

5.3. Correction dpkg et configuration Apache

✓ Résumé logique :

Test	Commande	Interprétation
IP DNS	<code>dig +short visio.workeezconnect.fr</code>	Résultat doit être l'IP publique du serveur
IP publique (instance)	<code>curl -4 ifconfig.me</code>	Doit matcher celle obtenue par dig
Résolution + Apache OK	<code>curl -I http://visio.workeezconnect.fr</code>	HTTP 200/301/302 = OK
Conf Apache (ServerName)	<code>sudo apachectl -S</code>	Doit afficher visio.workeezconnect.fr en vhost

6. Remplacement d'Apache par NGINX

On va installer Nginx fortement recommandé pour la sécurisation de échanges, cela implique arrêter Apache car il y aurait conflit sur le port 80 !

6.1. Suppression Apache & installation NGINX

Installer Nginx

```
sudo apt install -y nginx
```

Démarrer le service

```
sudo systemctl start nginx
```

Le rendre actif à chaque démarrage

```
sudo systemctl enable nginx
```

Arrêt d'Apache2

```
sudo systemctl stop apache2
```

```
sudo systemctl disable apache2
```

Supprimer d'Apache2

```
sudo apt purge apache2 apache2-utils apache2-bin apache2.2-common -y
```

Puis, s'assurer que le FQDN Fully Qualified Domain Name : nom de domaine complet incluant tous les niveaux (ex. auth.visio.workeezconnect.fr) est reconnu

```
echo "127.0.1.1 visio.workeezconnect.fr" | sudo tee -a /etc/hosts
```

redémarrer Nginx

```
sudo systemctl start nginx
```

```
sudo systemctl enable nginx
```

Vérifions l'état de Nginx et celui du port 80

```
sudo systemctl status nginx
```

```
sudo ss -ltn -i :80
```

6.2. Configuration de NGINX pour Jitsi

Vérification les fichiers de configuration NGINX pour jitsi

```
cat /etc/nginx/sites-available/visio.workeezconnect.fr.conf | grep ssl
```

```
cat: /etc/nginx/sites-available/visio.workeezconnect.fr.conf: No such file or directory
```

Il faudra donc le crée :

```
sudo nano /etc/nginx/sites-available/visio.workeezconnect.fr.conf
```

```
server {  
    listen 80;  
    server_name visio.workeezconnect.fr;  
  
    location / {  
        proxy_pass http://localhost:8000; # ou 5000 selon ton backend  
        proxy_http_version 1.1;  
        proxy_set_header Upgrade $http_upgrade;  
        proxy_set_header Connection "upgrade";  
    }  
}
```

```

    proxy_set_header Host $host;
    proxy_set_header X-Real-IP $remote_addr;
}

location /http-bind {
    proxy_pass http://localhost:5280/http-bind;
    proxy_set_header Host $host;
    proxy_http_version 1.1;
    proxy_set_header X-Forwarded-For $remote_addr;
}
}

```

Créer le lien symbolique

```
sudo ln -s /etc/nginx/sites-available/visio.workeezconnect.fr.conf /etc/nginx/sites-enabled/
```

Vérifions la configuration

```
sudo nginx -t
```

Redémarrer Nginx

```
sudo systemctl restart nginx
```

6.3. Activation HTTPS avec Certbot + Let's Encrypt

Insérons dans ce fichier un bloc sécurisé avec HTTPS + Cerbot et vérifions que le certificat est présent

```
sudo ls /etc/letsencrypt/live/visio.workeezconnect.fr/
```

Non il faut installer le plugin Certbot pour Nginx

```
sudo apt install python3-certbot-nginx -y
```

Puis Relancer la commande Certbot

```
sudo certbot --nginx -d visio.workeezconnect.fr
```

Ici je stop l'installation et je debug car il y a eu trop de manipulations générant des bugs récurrents.

1-Corriger les erreurs d'installation des paquets Jitsi (dpkg) et assurer une base fonctionnelle propre.

cd: hostnamectl

```

ubuntu@visio:~$ hostnamectl
  Static hostname: visio.workeezconnect.fr
        Icon name: computer-vm
        Chassis: vm
        Machine ID: 2627f45bbb694092bcc3d701d9e436a1
        Boot ID: ca67ac6e94c049c7b488d6a4eb4815a1
        Virtualization: kvm
  Operating System: Ubuntu 24.04.2 LTS
        Kernel: Linux 6.8.0-60-generic
        Architecture: x86_64
  Hardware Vendor: OpenStack Foundation
  Hardware Model: OpenStack Nova
  Firmware Version: 1.16.3-debian-1.16.3-2~bpo12+1
        Firmware Date: Tue 2014-04-01
        Firmware Age: 11y 3month 3d
ubuntu@visio:~$

```

2-Corriger /etc/hosts

cd: `sudo nano /etc/hosts`

```
GNU nano 7.2 /etc/hosts
127.0.0.1    localhost
::1         localhost ip6-localhost ip6-loopback
127.0.1.1    visio.workeezconnect.fr auth.visio.workeezconnect.fr guest.visio.workeezconnect.fr focus.visio.workeezconnect.fr conference.visio.workeezconnect.fr
```

3-Redémarrer Prosody

cd: `sudo systemctl restart prosody`

4-Redémarrer Jicofo et JVB

cd: `sudo systemctl restart jicofo`

cd: `sudo systemctl restart jitsi-videobridge2`

5- Vérifier l'état

cd: `sudo systemctl status prosody jicofo jitsi-videobridge2` **ok active (running)**

Nettoyer le cache DNS interne en powershell: **cd :** `ipconfig /flushdns`

PRÉREQUIS

1. Nom de domaine pointant vers l'IP publique : `visio.workeezconnect.fr` → `37.156.46.238`.
2. Accès SSH root ou sudo via ta clé (Tercium-instance_key).
3. Résolution DNS locale correcte (dig, ping, etc.).
4. Port TCP 443 (HTTPS) et 80 (HTTP) ouverts dans les règles de pare-feu GCP/Infomaniak.
5. ufw désactivé ou configuré (on vérifiera ça).
6. Pas d'Apache2 actif pour éviter conflit avec NGINX.

ÉTAPES POUR INSTALLER JITSY + NGINX AVEC CONFIG LOCALE

Installation des paquets requis :

```
sudo apt update && sudo apt upgrade -y
sudo apt install curl gnupg2 software-properties-common apt-transport-https -y
```

Ajout du dépôt Jitsi officiel :

Ajout la clef GPG :

```
curl https://download.jitsi.org/jitsi-key.gpg.key | sudo gpg --dearmor -o /usr/share/keyrings/jitsi-keyring.gpg
```

Ajoute la clé GPG :

```
echo 'deb [signed-by=/usr/share/keyrings/jitsi-keyring.gpg] https://download.jitsi.org stable/' | sudo tee /etc/apt/sources.list.d/jitsi-stable.list
```

`sudo apt update`

Installation de Jitsi Meet :

cd : `sudo apt install -y jitsi-meet`

Pendant l'installation : Possibilités

- FQDN : visio.workeezconnect.fr
- Choisir "configurer SSL plus tard (manuel)"

6.4 Génération du certificat SSL (Let's Encrypt)

```
sudo apt install certbot python3-certbot-nginx -y
sudo certbot --nginx -d visio.workeezconnect.fr
```

Valide automatiquement le NGINX + active HTTPS.

7. Tests fonctionnels de déploiement

Vérifier que Nginx est bien configuré pour Jitsi

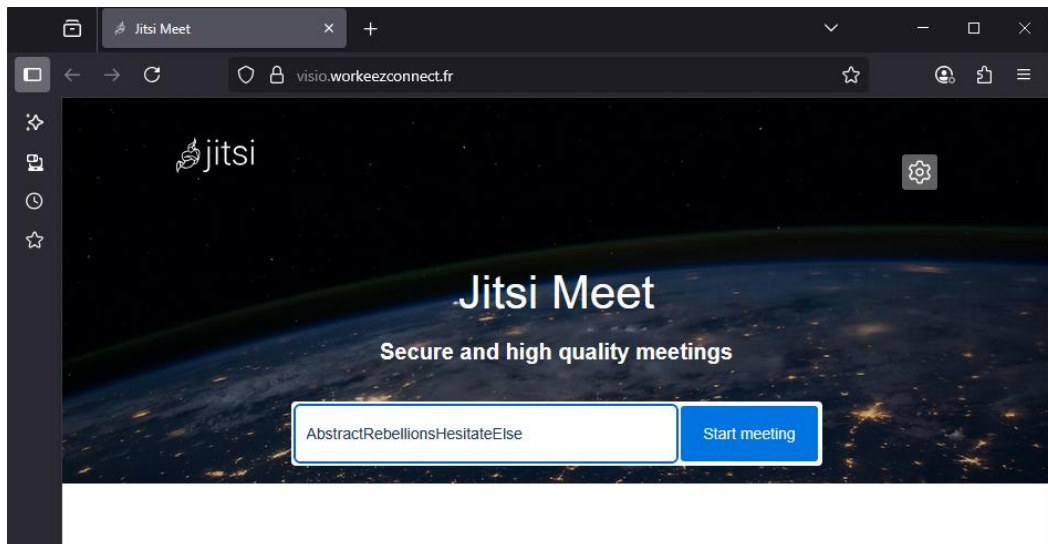
cd : `sudo systemctl status nginx`

```
ubuntu@jitsi-tercium:~$ sudo systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/usr/lib/systemd/system/nginx.service; enabled; preset: enabled)
   Active: active (running) since Mon 2025-07-07 12:13:02 UTC; 12min ago
     Docs: man:nginx(8)
  Main PID: 48836 (nginx)
    Tasks: 5 (limit: 9434)
   Memory: 9.1M (peak: 11.2M)
      CPU: 2.475s
   CGroup: /system.slice/nginx.service
           └─48836 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;
              └─61651 "nginx: worker process"
                 └─61652 "nginx: worker process"
                    └─61653 "nginx: worker process"
                       └─61654 "nginx: worker process"

Jul 07 12:16:33 jitsi-tercium nginx[54327]: 2025/07/07 12:16:33 [notice] 54327#54327:
Jul 07 12:16:33 jitsi-tercium systemd[1]: Reloaded nginx.service - A high performance web server and a reverse proxy server
Jul 07 12:21:40 jitsi-tercium systemd[1]: Reloading nginx.service - A high performance web server and a reverse proxy server
Jul 07 12:21:40 jitsi-tercium nginx[59285]: 2025/07/07 12:21:40 [warn] 59285#59285: *1012 [notice] 59285#59285:
Jul 07 12:21:40 jitsi-tercium nginx[59285]: 2025/07/07 12:21:40 [notice] 59285#59285:
Jul 07 12:21:40 jitsi-tercium systemd[1]: Reloaded nginx.service - A high performance web server and a reverse proxy server
Jul 07 12:21:52 jitsi-tercium systemd[1]: Reloading nginx.service - A high performance web server and a reverse proxy server
Jul 07 12:21:52 jitsi-tercium nginx[61646]: 2025/07/07 12:21:52 [warn] 61646#61646: *1012 [notice] 61646#61646:
Jul 07 12:21:52 jitsi-tercium nginx[61646]: 2025/07/07 12:21:52 [notice] 61646#61646:
Jul 07 12:21:52 jitsi-tercium systemd[1]: Reloaded nginx.service - A high performance web server and a reverse proxy server
lines 1-25/25 (END)
```

7.1. Accès HTTPS depuis navigateur

Tester la connexion HTTPS depuis un navigateur : Accède à <https://visio.workeezconnect.fr>



7.2. Vérifications : ports, pare-feu, DNS, curl, telnet, openssl

Vérifier les ports nécessaires ouverts : **il faut installer ufw**
cd : `sudo apt update && sudo apt install ufw`

cd : `sudo ufw allow 80,443/tcp`
`sudo ufw allow 10000/udp`
`sudo ufw enable`
`sudo ufw status`

```
test@KUS-F-STAGE:/mnt/c/Windows/System32$ sudo ufw allow 80,443/tcp
Rules updated
Rules updated (v6)
test@KUS-F-STAGE:/mnt/c/Windows/System32$ sudo ufw allow 1000
Rules updated
Rules updated (v6)
test@KUS-F-STAGE:/mnt/c/Windows/System32$ sudo ufw enable
Firewall is active and enabled on system startup
test@KUS-F-STAGE:/mnt/c/Windows/System32$ sudo ufw status
Status: active

To                Action            From
--                -
80,443/tcp        ALLOW            Anywhere
1000              ALLOW            Anywhere
80,443/tcp (v6)   ALLOW            Anywhere (v6)
1000 (v6)         ALLOW            Anywhere (v6)

test@KUS-F-STAGE:/mnt/c/Windows/System32$ |
```

1. Vérifier les modules Prosody :

cd : `sudo cat /etc/prosody/conf.avail/visio.workeezconnect.fr.cfg.lua | grep modules_enabled -A 20`

2. Test de fonctionnement des modules : relance puis vérification de son état

cd : `sudo systemctl restart prosody`

Puis

cd : `sudo systemctl status prosody`

```
ubuntu@jitsi-tercium:~$ sudo systemctl restart prosody
ubuntu@jitsi-tercium:~$ sudo systemctl status prosody
● prosody.service - Prosody XMPP Server
   Loaded: loaded (/usr/lib/systemd/system/prosody.service; enabled; >
   Active: active (running) since Mon 2025-07-07 13:10:28 UTC; 12s ago
     Docs: https://prosody.im/doc
    Main PID: 86388 (lua5.4)
      Tasks: 1 (limit: 9434)
     Memory: 12.9M (peak: 13.6M)
        CPU: 709ms
      CGroup: /system.slice/prosody.service
             └─86388 lua5.4 /usr/bin/prosody -F

Jul 07 13:10:28 jitsi-tercium systemd[1]: Started prosody.service - Pro>
lines 1-12/12 (END)
```

Ne pouvant contrôler Jitsi du côté d'Infomaniak, car nous ne pouvons pas voir l'installation. Cependant la section DNS via Zones donnera un tableau permettant de réaliser des vérifications.

1. Vérifier la résolution DNS (commande `dig`) :

cd : `dig visio.workeezconnect.fr @8.8.8.8 +short`

Réponse attendue et reçue : 37.156.48.238

```
ubuntu@jitsi-tercium:~$ dig visio.workeezconnect.fr @8.8.8.8 +short
37.156.46.238
```

2. Tester la connectivité HTTP(S) (commande `curl`) :

cd : `curl -I https://visio.workeezconnect.fr`

```
ubuntu@jitsi-tercium:~$ curl -I https://visio.workeezconnect.fr
HTTP/2 200
server: nginx/1.24.0 (Ubuntu)
date: Mon, 07 Jul 2025 13:33:00 GMT
content-type: text/html
vary: Accept-Encoding
strict-transport-security: max-age=63072000
```

3. S'il y avait échec avec HTTPS, tester HTTP:

cd : `curl -I http://visio.workeezconnect.fr`

```
ubuntu@jitsi-tercium:~$ curl -I http://visio.workeezconnect.fr
HTTP/1.1 301 Moved Permanently
Server: nginx/1.24.0 (Ubuntu)
Date: Mon, 07 Jul 2025 13:34:56 GMT
Content-Type: text/html
Content-Length: 178
Connection: keep-alive
Location: https://visio.workeezconnect.fr/
```

4. Tester le port 443 avec telnet ou nc

cd : `nc -zv visio.workeezconnect.fr 443`

```
ubuntu@jitsi-tercium:~$ nc -zv visio.workeezconnect.fr 443
Connection to visio.workeezconnect.fr (37.156.46.238) 443 port [tcp/https] succeeded!
```

7.3. Vérification de la chaîne SSL complète

Tester la chaîne de certificat SSL (Certbot utilisé)

```
cd : echo | openssl s_client -connect visio.workeezconnect.fr:443 -servername visio.workeezconnect.fr
```

```
ubuntu@jitsi-tercium:~$ echo | openssl s_client -connect visio.workeezconnect.fr:443
-sservername visio.workeezconnect.fr
CONNECTED(00000003)
depth=2 C = US, O = Internet Security Research Group, CN = ISRG Root X1
verify return:1
depth=1 C = US, O = Let's Encrypt, CN = E6
verify return:1
depth=0 CN = visio.workeezconnect.fr
verify return:1
```

- ISRG Root X1 → racine de Let's Encrypt
- Let's Encrypt E6 → intermédiaire
- visio.workeezconnect.fr → certificat de ton instance

Vérification SSL :

```
New, TLSv1.3, Cipher is TLS_AES_256_GCM_SHA384
Server public key is 256 bit
Secure Renegotiation IS NOT supported
Compression: NONE
Expansion: NONE
No ALPN negotiated
Early data was not sent
Verify return code: 0 (ok)
```

Verify return code: 0 (ok)

Vérification TLS :

```
Post-Handshake New Session Ticket arrived:
SSL-Session:
  Protocol  : TLSv1.3
  Cipher    : TLS_AES_256_GCM_SHA384
  Session-ID: 1133B848101D1A2A3E76A77AE8E076BEE87764C960945D561BE09F481529A70C
  Session-ID-ctx:
  Resumption PSK: DF4416E07C370CB47A55B42865FE58FF06F62D0D4A0202376E0BFE1037F14504E
  FBC9E0FC28A5FABA592EAF4B0E7E582
  PSK identity: None
  PSK identity hint: None
  SRP username: None
  TLS session ticket lifetime hint: 86400 (seconds)
  TLS session ticket:
  0000 - 0a 19 f2 1a 1f b3 53 bb-db c6 83 ee 0a 98 28 bf .....S.....(.
  0010 - 14 24 cf 0d ce 72 a4 11-e6 58 16 cc 2c 72 40 dc .$....r...X...r@.

  Start Time: 1751895664
  Timeout    : 7200 (sec)
  Verify return code: 0 (ok)
  Extended master secret: no
  Max Early Data: 0
```

Conclusion :

Élément vérifié	État
Certificat SSL bien installé	✓ OK
Chaîne complète avec intermédiaire & racine	✓ OK
Nom de domaine valide (CN = visio...)	✓ OK
Protocole TLS récent (1.3)	✓ OK