Vorteks

make IT work, make IT free

# Sécurisation de Jitsi en SaaS

OPEN SOURCE EXPERIENCE

# Sommaire

# Présentation

# Worteks (\vɔʁ.tɛks\)

## Service

Infrastructures hétérogènes et complexes, cloud, mail, authentification, sécurité

- Études, audit et consulting
- Expertise technique
- Support technique
- Formation
- R&D et innovation

## Édition

**V'Sweet** — Portail d'applications collaboratif

**V'Opla** — Plateforme mutualisée de développement

**V'IDaaS** — Gestion des identités et des accès

## Partenaires

Red Hat Advanced Business Partner Reseller

BlueMind

ONLYOFFICE

Collabora Online

# Présentation

## Soisik FROGER

Architecte logiciel @ Worteks

20 ans d'XP dans le développement et l'intégration de solutions open-source dans les domaines de

- La gestion d'identité : SSO, LDAP
- Les outils collaboratifs : chat, visio, cloud, mail
- Les applications métiers : spécifique, GED/BPM, décisionnel

.

soisik.froger@worteks.com
@soisikf@framapiaf.org
https://fr.linkedin.com/in/soisik-froger-712548116

# Worteks (\vɔʁ.tɛks\)

https://www.worteks.com/rejoindre/

# Vue d’ensemble

# Besoins métiers

- Doter les agents d’un outil de planification et d’accès à des visio-conférences répondant aux exigences HDS
  - Sécurisation : données, accès, disponibilité
  - Externalisation (SaaS) : cloud et logiciels souverains
  - Auditabilité : journalisation et reporting métiers
  - Deploiement automatisé (ansible), scalabilité
- Cas d’usage types
  - Echange confraternel entre un médecin conseil et un médecin libéral sur un sujet médical
  - A terme, échange entre un médecin conseil et un assuré

# Un outil sécurisé

- Habilitation via l’import automatisé de référentiels
  - Identifiants des agents et groupes
- Identification opérée depuis le SI client
  - Connexion via le SSO du client accessible en interne/VPN
  - Récupération des attributs utilisateurs via le SSO
- Accueil des invités sur le créneau réservé
  - Portail d'accueil
  - Attente de l’arrivée du modérateur dans une salle d'attente
  - Filtrage des arrivées par les modérateurs
  - Deconnexion automatique des invités à la fin de la réunion

# Un outil sécurisé et souverain

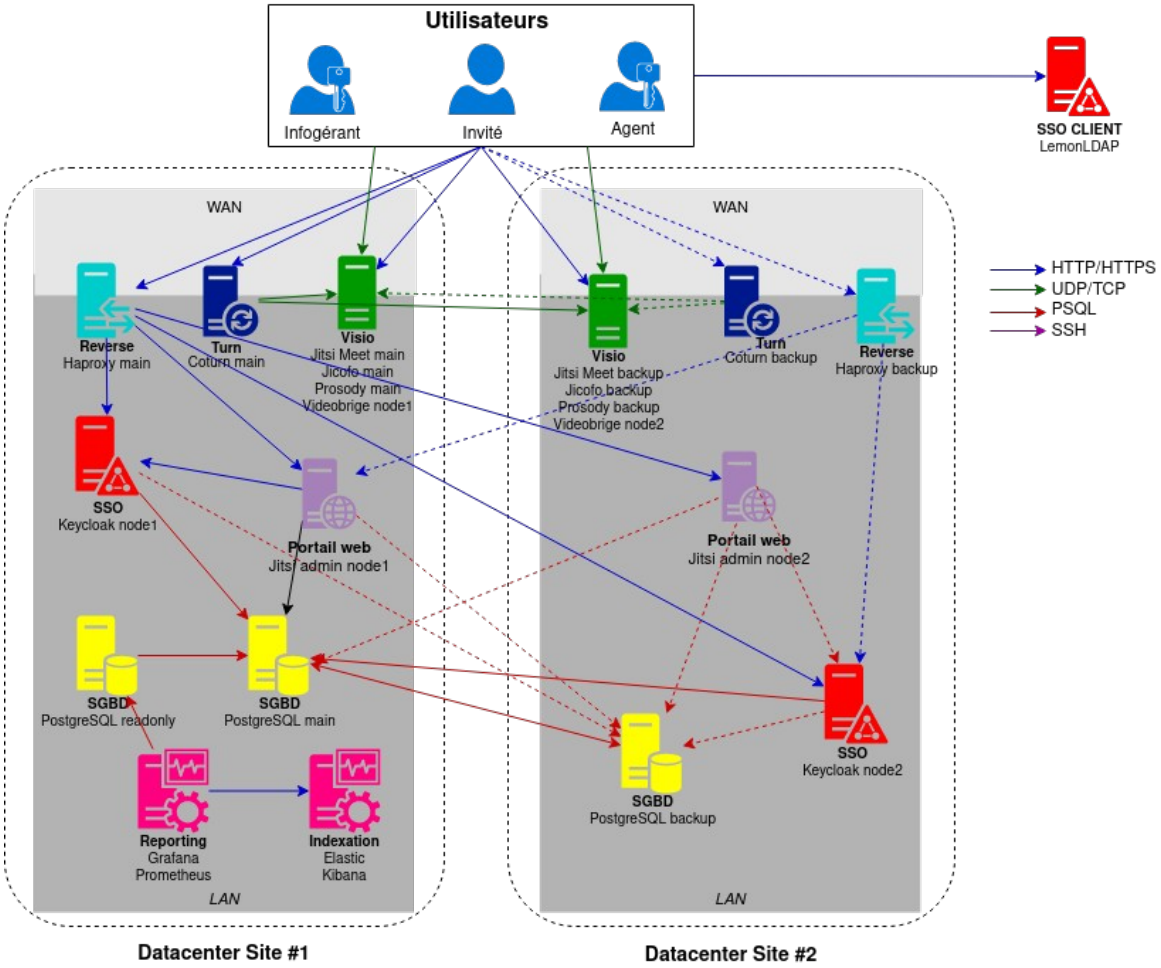
- Architecture pensée pour la haute-disponibilité et la scalabilité sur deux sites géographiques
- Intégration de composants opensource : Jitsi, Jitsi-Admin, Keycloak, Grafana/ELK, PostgreSQL, CoTurn, OpenLDAP, Pacemaker, Corosync, PCS
- Services et données hébergés en France
- Infogérance externalisée à des prestataires

# Un outil sécurisé, souverain et pilote

- Projet pilote pour ce client dans son évaluation et son appropriation du cloud et de l’open-source
  - Problématiques sur la maîtrise des couts, les limites des outils vs exigences métiers ou contraintes techniques, la MCO
- Hébergement externalisé auprès d'un opérateur de cloud Français
  - Problématiques sur le SMTP, cloisonnement, réseaux, DNS, certificats ...

# Architecture technique

# Schéma d’architecture technique

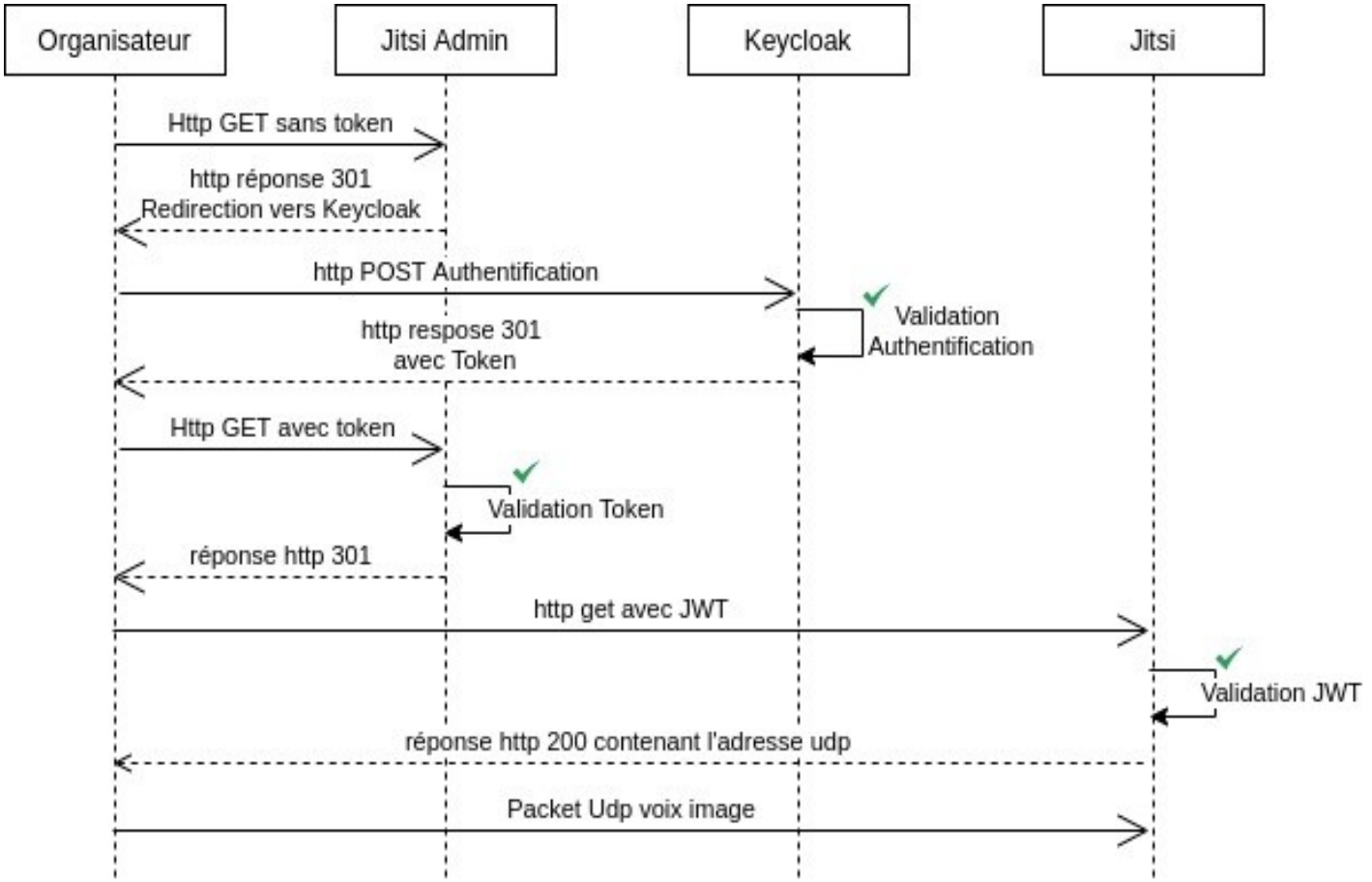


# Sécurisation
# Json Web Token

# Modes d’authentification sur Jitsi

- Par défaut : pas d'authentification
- Authentifier les utilisateurs à l’ouverture d’une réunion
  - Authentification depuis un annuaire LDAP (Cyrus SASL)
    - Ce n'est pas du SSO
  - Authentification SSO avec headers (mode dit shibboleth)
    - N'est plus supporté dans les dernières versions
  - Authentification par jitsi JWT token
    - Authentification depuis un SSO compatible
    - Intégration avec des applications
      - **Génération à la volée de liens sécurisé vers des conférences**
      - Envoi de liens sécurisés dans les invitations

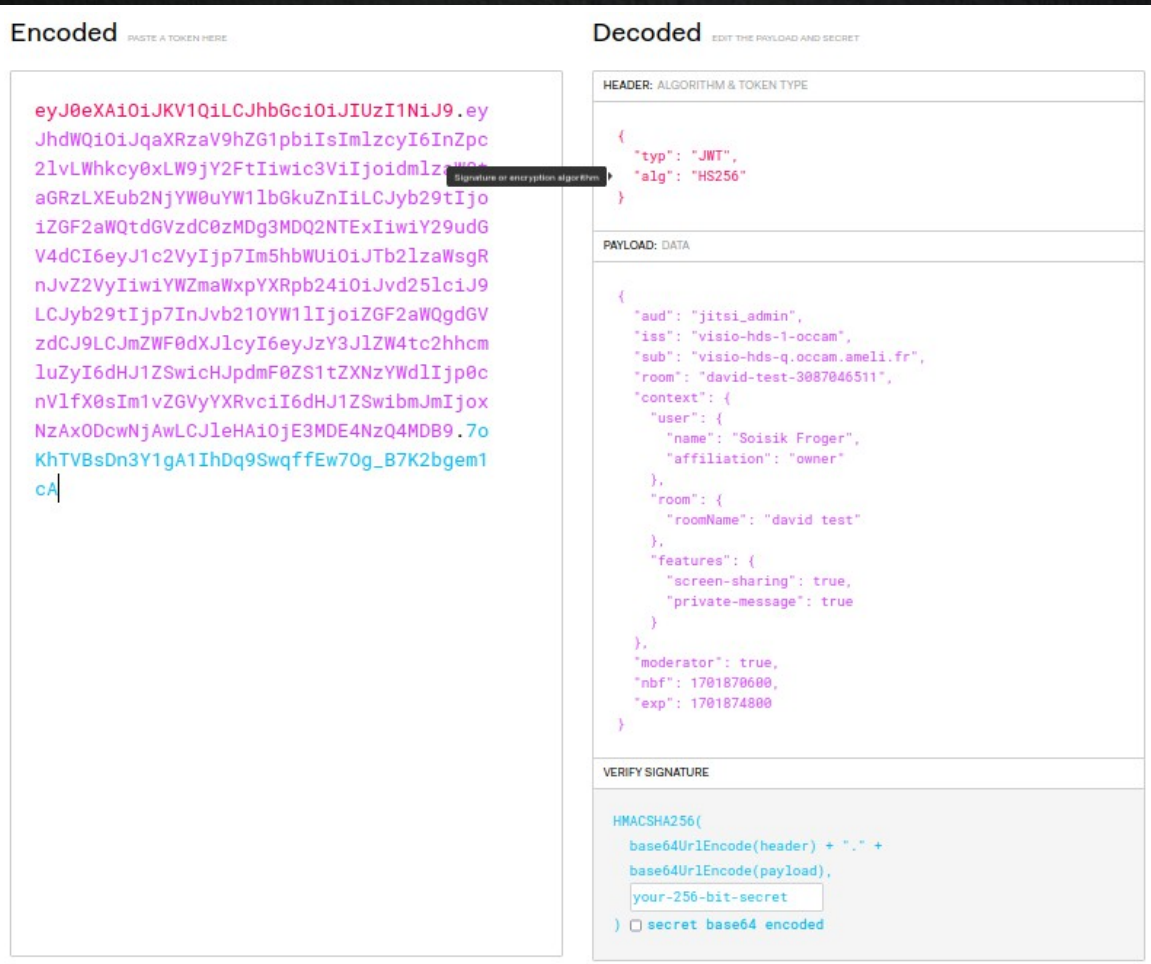
# Flux d'authentification



# Qu’est-ce qu’un JSON web token ?

- Standard RFC 7519
- Format Json
- Défini un mode de transmission d'information sécurisée entre deux applications
- Lisible par tous mais inaltérable
- Décomposé en trois sections : entête, payload, signature (via un secret ou une clé asymétrique)

# Contenu du JWT : modérateur



# Contenu du JWT : invité

**Encoded** PASTE A TOKEN HERE


```
eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.ey
JhdWQiOiJqaXRzaV9hZG1pbiIsImlzcyI6InZpc
2lvLWhkcy0xLW9jY2FtIiwic3ViIjoidmlzaW8t
aGRzLXEub2NjYW0uYW1lbGkuZnIiLCJyb29tIjo
iZGF2aWQtdGVzdC0zMDg3MDQ2NTExIiwiY29udG
V4dCI6eyJ1c2VyIjp7Im5hbWUiOiJzb2lzaWsuZ
nJvZ2VyQHdvcnRla3MuY29tIiwiYWZmaWxpYXRp
b24iOiJtZW1iZXIifSwicm9vbSI6eyJyb29tTmF
tZSI6ImRhdmlkIHRlc3QifSwiZmVhdHVyZXMiOn
sic2NyZWVuLXNoYXJpbmciOnRydWUsInByaXZhd
GUtbWVzc2FnZSI6dHJ1ZX19LCJtb2RlcmF0b3Ii
OmZhbHNlLCJuYmYiOjE3MDE4NzA2MDAsImV4cCI
6MTcwMTg3NDgwMH0.4OWv1CZgWWi1C4_e5UuiuZ
EPzUF887zi1Ylxq8IL04E
```


**Decoded** EDIT THE PAYLOAD AND SECRET

**HEADER:** ALGORITHM & TOKEN TYPE


```
{
  "typ": "JWT",
  "alg": "HS256"
}
```


**PAYLOAD:** DATA


```
{
  "aud": "jitsi_admin",
  "iss": "visio-hds-1-occam",
  "sub": "visio-hds-q.occam.ameli.fr",
  "room": "david-test-3087046511",
  "context": {
    "user": {
      "name": "soisik.froger@worteks.com",
      "affiliation": "member"
    },
    "room": {
      "roomName": "david test"
    },
    "features": {
      "screen-sharing": true,
      "private-message": true
    }
  },
  "moderator": false,
  "nbf": 1701870600,
  "exp": 1701874800
}
```


**VERIFY SIGNATURE**

```
HMACSHA256(
  base64UrlEncode(header) + "." +
  base64UrlEncode(payload),
  your-256-bit-secret
) ☐ secret base64 encoded
```

# Démo

# Configuration

# jitsi-meet-tokens

- Prérequis :
  - VMs ubuntu/debian
  - Installation de Jitsi-meet
  - Installation de Jitsi-admin
- Installer le module jitsi-meet-tokens
  - Un nom d’application et une chaine aléatoire (le secret partagé) est demandé lors de l'installation

# Modules communautaires

- Installler les modules communautaires Jitsi
  - Jitsi-token-moderation
    - Lit l'attribut « moderator » du JWT pour déterminer si l'utilisateur est modérateur de la réunion
  - token_affiliation
    - Lit l'attribut « affiliation » du JWT pour déterminer si l'utilisateur est propriétaire de la réunion (dépendance de token_owner_party)
  - token_owner_party
    - Fermer la conférence 10 minutes après le départ du dernier propriétaire

# Configuration Jitsi

Dans /etc/prosody/conf.d/visio.domain.io.cfg.lua

```
VirtualHost "visio.domain.io"
    authentication = "token"
    app_id="visio"
    app_secret="some_secret"
    allow_empty_token = false
```

`allow_empty_token = false` impose la fourniture d'un token pour l'accès à une réunion (y compris pour les invités)

# Configuration Jitsi

Dans le virtualHost conference.visio.domain.io :

```
Component "conference.visio.domain.io" "muc"
modules_enabled = {
        ....
        "token_verification";
        "token_moderation";
        "token_affiliation";
        "token_owner_party";
        "muc_max_occupants";
```

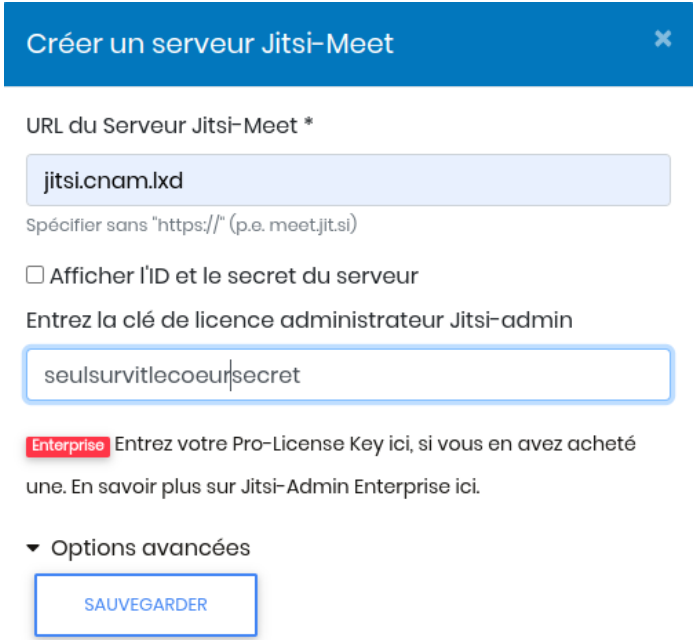
# Configuration Jitsi

Dans le fichier /etc/jitsi/jicofo/sip-communicator.properties

```
org.jitsi.jicofo.auth.URL=EXT_JWT:visio.domain.io
org.jitsi.jicofo.DISABLE_AUTO_OWNER=true
org.jitsi.jicofo.auth.DISABLE_AUTOLOGIN=true
```

# Configuration Jitsi-Admin

- Déclarer le serveur et le secret partagé depuis l'interface avec un profil admin



# Adaptations

# Adaptations Jitsi-admin

- Jitsi-admin a fait l'objet de très nombreuses adaptations
  - Anonymisation de l'adresse de l’organisateur dans les invitations envoyées par mail
  - Format du token : attributs affiliation, exp, nbf, mail du modérateur si non anonyme
  - Forcer l’authentification via le SSO pour les modérateurs
  - Traitement des réponses reçues par mail (relai, alerte...)

# Adaptations : Jitsi / Prosodie

- Modules prosody
  - jitsi-meet-tokens
  - lobby_rooms
    - Activation du « lobby » à l'arrivée du premier participant
    - Le modérateur by-pass le « lobby »
- Jitsi front
  - Thème graphique
  - Déconnexion des invités en fin de réunion
  - Affichage du contact (mailto:) du modérateur en cas de souci de connexion

Worteks

*make IT work, make IT free*

# Merci


**info@worteks.com**

**@worteks_com@mastodon.social**

**linkedin.com/company/worteks**