

BTS Services informatiques aux organisations - SISR**Session 2026****E4 – Support et mise à disposition de services informatiques****Coefficient 4****DESCRIPTION DE LA REALISATION PROFESSIONNELLE****NOM et prénom du candidat : Aoudani Noeh****Contexte de la réalisation professionnelle**

L'entreprise, Les Ateliers De Joigny, gérait jusqu'à présent son inventaire informatique et les demandes d'assistance de manière informelle, à l'aide de tableurs et d'échanges par courriel. Cette organisation entraînait un manque de visibilité sur l'état du parc informatique, une difficulté à suivre les interventions et des délais de résolution des incidents trop longs.

Cette évolution vise à centraliser la gestion des actifs informatiques (postes de travail, logiciels, licences) et mettre à disposition des utilisateurs un portail de support unique pour le traitement des demandes d'assistance. Elle nécessite une solution capable de structurer le suivi du parc, d'automatiser l'inventaire et d'améliorer la gestion des tickets.

La solution retenue est GLPI, pour sa robustesse, son caractère Open Source et sa capacité à s'interfacer avec un outil d'inventaire automatique. Cette réalisation professionnelle couvre l'installation du serveur GLPI, la configuration du module de helpdesk et l'importation du parc informatique.

Intitulé de la réalisation professionnelle*Mise en œuvre d'une solution de gestion de parc et de helpdesk (GLPI)***Période de réalisation : Décembre 2025****Lieu : Joigny****Modalité : Individuelle****Principale(s) activité(s) concernée(s) :**

Gérer le patrimoine informatique
Répondre aux incidents et aux demandes d'assistance et d'évolution
Mettre à disposition des utilisateurs un service informatique

Conditions de réalisation

- **Ressources initiales** : 1 VM Debian 12 (2 vCPU, 4 GB RAM, 40 GB disque), serveur MariaDB existant ou local.
- **Résultats attendus** : Interface GLPI accessible en HTTPS, configuration des entités et profils, remontée d'inventaire réussie.
- **Durée** : 5 à 7 jours.

Modalités d'accès à cette réalisation professionnelle.*<https://site.jjba.fr>. Compte d'accès : aucun. Mot de passe : BTSSiosisr***Partie 1 – Procédure de mise en œuvre.****Préparation de l'environnement**

Dans un premier temps, un environnement conteneurisé a été mis en place afin de déployer l'application GLPI de manière isolée et reproductible.

Le projet est stocké dans le répertoire suivant :

```
/opt/docker/glpi
```

Ce répertoire contient le fichier docker-compose.yml ainsi que les volumes nécessaires au fonctionnement de l'application.

```
root@docker:~# tree /opt/docker/glpi
/opt/docker/glpi
├── docker-compose.yml

1 directory, 1 file
root@docker:~# _
```

Création du réseau Docker

Deux réseaux Docker sont utilisés dans cette infrastructure :

- Un réseau interne nommé `internal`, permettant la communication entre le conteneur GLPI et la base de données
- Un réseau externe nommé `proxy`, utilisé par le reverse proxy Traefik pour exposer les services

Le réseau `proxy` étant déjà existant, il est déclaré comme externe dans la configuration.

```
root@docker:~# docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
0c92eceac869       bridge             bridge              local
686d0177b223       glpi_internal      bridge              local
2346d32f76fd       host               host                local
985ce14e3353       none               null                local
29d53f569ab9       proxy              bridge              local
c64b65384764       traefik_dmz_network macvlan             local
ec3b6a57f3b1       traefik_srv_network macvlan             local
7bf8c570a0c6       zabbix-docker_default bridge              local
7054001cf8c6       zabbix_proxy      bridge              local
root@docker:~# _
```

Déploiement des conteneurs avec Docker Compose

Le déploiement de GLPI repose sur deux conteneurs :

- Un conteneur MariaDB pour la base de données
- Un conteneur GLPI pour l'application web

Extrait du fichier docker-compose.yml

```
version: '3.8'

services:
  db:
    image: mariadb:10.6
    container_name: glpi-db
    restart: always
    environment:
      MYSQL_ROOT_PASSWORD: rootpassword
      MYSQL_DATABASE: glpi
      MYSQL_USER: glpi
      MYSQL_PASSWORD: glpipassword
    volumes:
      - db_data:/var/lib/mysql
    networks:
      - internal

  glpi:
    image: diouxx/glpi
    container_name: glpi-app
    restart: always
    depends_on:
      - db
    environment:
      TIMEZONE: Europe/Paris
    volumes:
      - glpi_data:/var/www/html
    networks:
      - internal
      - proxy
```

Explication

Le service db utilise l'image MariaDB et stocke les données de GLPI.

Le service glpi correspond à l'application web accessible via un navigateur.

La directive depends_on permet de s'assurer que la base de données est démarrée avant GLPI.

Les volumes garantissent la persistance des données.

Lancement des services

```

root@docker:~# docker ps
CONTAINER ID   IMAGE                                COMMAND                  CREATED        STATUS        PORTS
12c983ae69c2   diouxx/glpi                          "/opt/glpi-start.sh"   19 hours ago  Up 19 hours  80/tcp, 443/tcp
581b34807a27   mariadb:10.6                         "docker-entrypoint.s..." 19 hours ago  Up 19 hours  3306/tcp
1e27df280436   grafana/grafana:10.4.2              "/run.sh"              22 hours ago  Up 21 hours  3000/tcp
ff54a99f4d52   zabbix/zabbix-web-nginx-mysql:ubun... "docker-entrypoint.sh"  23 hours ago  Up 23 hours (healthy)  8080/tcp, 8443/tcp
7971b9c93cd2   zabbix/zabbix-server-mysql:ubuntu-7... "/usr/bin/docker-ent..." 23 hours ago  Up 23 hours  10051/tcp
9096e2f51a60   mysql:8.0                            "docker-entrypoint.s..." 23 hours ago  Up 23 hours (healthy)  3306/tcp, 33060/tcp
a3d975d5f4bb   wordpress:latest                    "docker-entrypoint.s..." 8 weeks ago   Up 2 days    80/tcp
cf8d0759f049   mariadb:10.11                       "docker-entrypoint.s..." 8 weeks ago   Up 2 days    3306/tcp
f53a6d4b84ff   nextcloud:latest                    "/entrypoint.sh apac..." 8 weeks ago   Up 2 days    80/tcp
4dec58e79f01   traefik:v2.11                       "/entrypoint.sh --ap..." 8 weeks ago   Up 2 days    0.0.0.0:80->80/tcp, [::]:80->80/tcp
root@docker:~#

```

docker-compose up -d

Intégration avec Traefik

L'application GLPI est exposée via le reverse proxy Traefik déjà présent dans l'infrastructure.

L'accès se fait via le nom de domaine :

<https://glpi.jjba.fr>

```

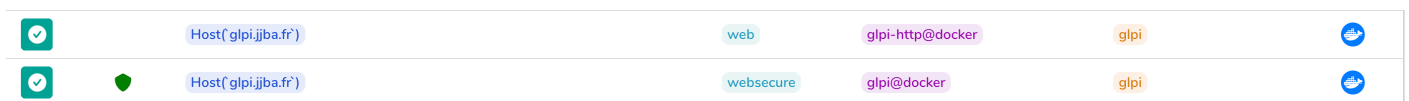
labels:
- "traefik.enable=true"
- "traefik.http.routers.glpi.rule=Host(`glpi.jjba.fr`)"
- "traefik.http.routers.glpi.entrypoints=websecure"
- "traefik.http.routers.glpi.tls=true"
- "traefik.http.routers.glpi.tls.certresolver=myresolver"
- "traefik.http.routers.glpi-http.rule=Host(`glpi.jjba.fr`)"
- "traefik.http.routers.glpi-http.entrypoints=web"
- "traefik.http.routers.glpi-http.middlewares=redirect-to-https"
- "traefik.http.services.glpi.loadbalancer.server.port=80"
- "traefik.docker.network=proxy"

```

La configuration est réalisée à l'aide des labels présents dans le fichier docker-compose.yml.

Extrait de configuration Traefik

Explication



Traefik détecte automatiquement le conteneur GLPI grâce aux labels Docker.

Le domaine `glpi.jjba.fr` est associé au service GLPI.

Le protocole HTTPS est activé via Let's Encrypt.

Une redirection automatique du HTTP vers HTTPS est mise en place.



Installation de GLPI

Une fois les conteneurs démarrés, l'installation de GLPI est réalisée via l'interface web.

Étapes

- Accès à l'URL de GLPI
- Choix de la langue
- Acceptation de la licence
- Configuration de la base de données :
 - serveur : db
 - utilisateur : glpi
 - mot de passe : glpipassword
- Finalisation de l'installation

Configuration LDAP

GLPI est connecté à l'Active Directory afin de permettre l'authentification centralisée des utilisateurs.

Paramètres utilisés












Nom	<input type="text" value="AD"/>		
Serveur par défaut	<input type="text" value="Oui"/>	Activé	<input type="text" value="Non"/>
Serveur	<input type="text" value="10.89.10.5"/>	Port (par défaut 389)	<input type="text" value="389"/>
Commentaires	<input type="text"/>		
Filtre de connexion	<input type="text" value="(& (&(objectClass=user)(objectCategory=person)(!(userAccountControl:1.2.840.113556.1.4.803:=2))))"/>		
BaseDN	<input type="text" value="OU=User,OU=Utilisateurs,OU=JJBA,DC=jjba,DC=lan"/>		
Utiliser bind ?	<input type="text" value="Oui"/>		
DN du compte (pour les connexions non anonymes)	<input type="text" value="glpi@jjba.lan"/>		
Mot de passe du compte (pour les connexions non anonymes)	<input type="text"/> <input type="checkbox"/> Effacer		
Champ de l'identifiant	<input type="text" value="samaccountname"/>	Champ de synchronisation ?	<input type="text" value="objectguid"/>

Étapes

- Accès au menu Configuration → Authentification → LDAP
- Ajout d'un serveur LDAP
- Test de la connexion
- Import des utilisateurs

Test LDAP Serveur : AD



<input type="checkbox"/>	IDENTIFIANT	NOM DE FAMILLE	E-MAILS
<input type="checkbox"/>	 glpi		
<input type="checkbox"/>	 post-only		
<input type="checkbox"/>	 tech		
<input type="checkbox"/>	 normal		
	 glpi-system	Support	
<input type="checkbox"/>	 Noeh	Aoudani	noehaoudani@gmail.com
<input type="checkbox"/>	 veeam		
<input type="checkbox"/>	 noeh		noeh@jjba.fr
<input type="checkbox"/>	 liana		liana@jjba.fr
<input type="checkbox"/>	 boudew	boudew	boudew@jjba.fr
<input type="checkbox"/>	 Loulou	Hallal	eloann@jjba.fr

Tests de bon fonctionnement

Plusieurs tests ont été réalisés afin de valider le bon fonctionnement de la solution GLPI mise en place.

Test de déploiement des conteneurs

Une vérification a été effectuée afin de s'assurer que les conteneurs Docker sont correctement démarrés.

La commande suivante a été utilisée :

```
docker ps
```

Résultat attendu :

- conteneur glpi-app en état "Up"
- conteneur glpi-db en état "Up"

Ce test confirme que l'application et la base de données sont opérationnelles.

Test d'accès via navigateur

Un test d'accès à GLPI a été réalisé via un navigateur web en utilisant l'URL suivante :

```
https://glpi.jjba.fr
```

Résultat attendu :

- affichage de l'interface GLPI
- présence du cadenas HTTPS dans la barre d'adresse

Ce test valide :

- le bon fonctionnement du reverse proxy Traefik
- la génération du certificat SSL via Let's Encrypt
- l'accessibilité de l'application depuis le réseau

Test de routage Traefik

Une vérification a été effectuée via le dashboard Traefik afin de s'assurer que le service GLPI est correctement détecté.

Résultat attendu :

- présence d'un router associé à glpi.jjba.fr
- service actif et fonctionnel

Ce test confirme que le routage dynamique fonctionne correctement.

Test de connexion LDAP

Un test de connexion à l'Active Directory a été réalisé depuis GLPI.

Résultat obtenu :

- connexion au serveur LDAP réussie
- authentification du compte de liaison validée
- récupération des entrées utilisateurs confirmée

Ce test valide :

- la communication avec l'Active Directory
- la validité des paramètres LDAP

Test d'import des utilisateurs

Une recherche LDAP a été effectuée depuis GLPI, permettant de récupérer les utilisateurs présents dans l'annuaire.

Résultat obtenu :

- utilisateurs détectés dans l'Active Directory
- import manuel des utilisateurs dans GLPI

Ce test confirme :

- la bonne configuration du filtre LDAP
- la capacité de GLPI à exploiter l'annuaire

Test de connexion utilisateur

Un test de connexion a été réalisé avec un utilisateur issu de l'Active Directory.

Étapes :

- connexion à GLPI avec un compte AD
- accès à l'interface utilisateur

Résultat attendu :

- authentification réussie
- accès à l'interface GLPI

Ce test valide l'intégration LDAP et l'authentification centralisée.

Test de création de ticket

Un test fonctionnel a été réalisé afin de vérifier le bon fonctionnement du système de gestion de tickets.

Étapes :

- connexion avec un utilisateur
- création d'un ticket
- enregistrement du ticket

Résultat attendu :

- ticket créé avec succès
- ticket visible dans l'interface GLPI

Ce test confirme que la fonctionnalité principale de l'application est opérationnelle.

Preuves et résultats

Les différents tests réalisés permettent de valider que :

- les conteneurs Docker sont fonctionnels
- l'application GLPI est accessible en HTTPS
- le routage via Traefik est opérationnel
- l'intégration LDAP fonctionne correctement
- les utilisateurs peuvent se connecter
- les tickets peuvent être créés et suivis

Validation de la solution

Les objectifs définis en début de projet ont été atteints.

La solution mise en place permet :

- de centraliser les demandes utilisateurs
- d'assurer un suivi des tickets
- de s'appuyer sur l'Active Directory pour l'authentification
- de proposer un accès sécurisé via HTTPS

L'infrastructure déployée est fonctionnelle, cohérente et adaptée aux besoins de l'entreprise.

Partie 3 – Veille technologique.

Analyse des résultats

La solution mise en place repose sur une infrastructure conteneurisée utilisant Docker, avec une exposition des services via le reverse proxy Traefik.

Les différents tests réalisés ont permis de valider le bon fonctionnement global du système :

- les conteneurs GLPI et MariaDB sont correctement déployés et fonctionnels
- l'application GLPI est accessible via un nom de domaine dédié
- l'accès est sécurisé en HTTPS grâce à l'intégration avec Traefik et Let's Encrypt
- le routage des requêtes est correctement assuré
- l'intégration avec l'Active Directory via LDAP est opérationnelle
- les utilisateurs peuvent s'authentifier avec leurs comptes existants
- les tickets peuvent être créés et suivis dans l'interface

La solution répond donc au besoin initial de centralisation et de suivi des demandes utilisateurs.

Avantages de la solution

La mise en place de GLPI dans un environnement Docker présente plusieurs avantages :

- déploiement rapide et reproductible
- isolation des services grâce à la conteneurisation
- facilité de maintenance et de mise à jour
- intégration simple avec Traefik pour la gestion des accès
- sécurisation automatique des échanges via HTTPS
- authentification centralisée avec l'Active Directory

Cette architecture permet également d'ajouter facilement de nouveaux services dans l'infrastructure existante.

Limites identifiées

Malgré les résultats satisfaisants, certaines limites ont été identifiées :

- absence de configuration des notifications par mail
- import des utilisateurs LDAP non automatique (nécessite une action manuelle)
- absence de système de sauvegarde des données
- configuration initiale de GLPI et LDAP pouvant être complexe
- absence de supervision avancée de l'application

Ces éléments peuvent impacter l'exploitation à long terme de la solution.

Propositions d'amélioration

Plusieurs axes d'amélioration peuvent être envisagés afin d'optimiser la solution mise en place.

Notifications par mail

La configuration d'un serveur SMTP permettrait :

- l'envoi automatique de notifications lors de la création ou modification des tickets
- une meilleure réactivité du support
- une amélioration de la communication avec les utilisateurs

Automatisation LDAP

Il serait possible de :

- automatiser l'import des utilisateurs depuis l'Active Directory
- synchroniser régulièrement les comptes

Cela permettrait de limiter les actions manuelles et d'assurer une meilleure cohérence des données.

Sauvegardes

La mise en place de sauvegardes régulières permettrait de sécuriser les données :

- sauvegarde des volumes Docker
- export de la base de données MariaDB

Sécurisation

Des améliorations de sécurité peuvent être apportées :

- restriction des accès à GLPI
- mise en place de règles de pare-feu
- utilisation de LDAP sécurisé (LDAPS)

Supervision

L'ajout d'un outil de supervision permettrait de surveiller l'état des services :

- suivi des conteneurs Docker
- alertes en cas de panne
- analyse des performances

Veille technologique

Une veille technologique a été réalisée autour des outils de gestion de tickets et des solutions de conteneurisation.

Plusieurs alternatives à GLPI ont été étudiées :

- OTRS : solution de gestion de tickets plus orientée entreprise
- Zammad : interface moderne avec fonctionnalités avancées
- Freshdesk : solution cloud externalisée

Concernant le reverse proxy, des alternatives à Traefik existent :

- Nginx Proxy Manager : interface graphique simple à utiliser
- HAProxy : solution performante mais plus complexe

Cette veille a permis de confirmer le choix de GLPI et Traefik, notamment pour leur intégration avec Docker et leur flexibilité.

Conclusion

La solution mise en place répond aux besoins de l'entreprise en proposant un système de gestion de tickets fonctionnel, sécurisé et intégré à l'infrastructure existante.

L'utilisation de Docker et Traefik permet de simplifier le déploiement et la gestion des services, tout en garantissant une bonne évolutivité.

Des améliorations peuvent être apportées, notamment en matière de notifications, de sauvegardes et de supervision, afin de rendre la solution encore plus complète et professionnelle.