Mode Opératoire

SSO Active Directory des services Docker (Authentik)

Code :	MO-PLT-025
Version :	1.3
Date :	21 septembre 2026
Auteur :	Cédric LEGRAND
Classification :	USAGE INTERNE — Équipe BTS SIO

Historique des révisions

Version	Date	Modifications
1.0	15/09/2026	Création initiale — SSO AD des services Docker (chantier des 14–15/09/2026).
1.1	20/09/2026	UniFi Network basculé derrière la porte (forwardAuth, login local conservé) ; accès direct IP macvlan restreint par règles flottantes OPNsense.
1.2	20/09/2026	Bind LDAP de la source AD basculé sur le compte de service en lecture seule `svc-authentik-ldap` (MO-PLT-026).
1.3	21/09/2026	Accès de secours IP restreints au VLAN PLT (pare-feu OPNsense + Proxmox, ports web retirés du réseau plat) — chantier blocage étudiants (MO-NET-009).

1 Objet

Ce mode opératoire décrit l'exploitation du **SSO Active Directory** des services Docker de l'infrastructure BTS SIO, mis en production les 14 et 15 septembre 2026 : comprendre l'architecture en place, **ajouter ou retirer un administrateur**, **brancher un nouveau service** sur la porte unique Authentik et **dépanner** les situations courantes.

Depuis ce chantier, les six administrateurs de la plateforme ouvrent chaque service avec leur **compte Active Directory** et obtiennent automatiquement les droits d'administration ; tout compte hors du groupe `GG-AdminsPlateforme` est refusé à la porte, avant même d'atteindre le service.

2 Champ d'application

Public concerné	Administrateurs de l'infrastructure BTS SIO (membres de `GG-AdminsPlateforme`)
Fournisseur d'identité	Authentik 2026.2.2 — CT 200 `docker-srv`, `https://auth.docker.bts.sio`
Annuaire	Active Directory `bts.sio` (DC1, `ldaps://10.0.112.2`) ; bind en lecture seule par le compte de service `svc-authentik-ldap` depuis le 20/09/2026 (MO-PLT-026)
Groupe habilité	`GG-AdminsPlateforme` (`CN=GG-AdminsPlateforme,OU=Profs,DC=bts,DC=sio`)
Comptes de secours	Coffre Vaultwarden (items cités dans ce document)

3 Architecture mise en place (15/09/2026)

3.1 La porte unique

Authentik joue le rôle de fournisseur d'identité (IdP) pour toute la plateforme Docker. Il synchronise ses utilisateurs depuis l'annuaire AD via la source LDAP « Active Directory BTS SIO » (slug `ad-bts-sio`), dont le filtre a été resserré au groupe seul :

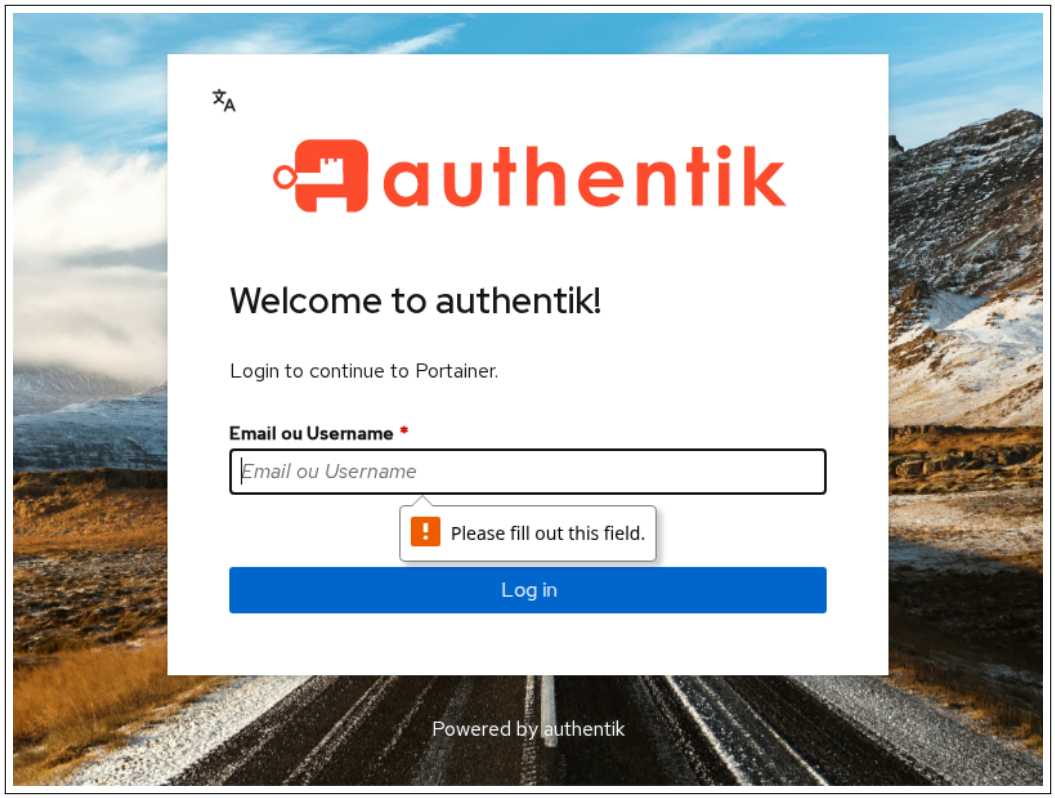

Figure 1 – La porte Authentik, commune à tous les services protégés : compte AD puis mot de passe AD, une seule saisie par session.

```
(&(objectCategory=person)(objectClass=user)
  (memberOf=CN=GG-AdminsPlateforme,OU=Profs,DC=bts,DC=sio))
```

avec `delete_not_found_objects=true` : un compte qui sort du groupe est **supprimé d'Authentik** à la synchronisation suivante. Seuls les six comptes du groupe existent donc dans Authentik (aux côtés des comptes internes `akadmin` et `ak-outpost-...`).

Une **policy partagée** « membre GG-AdminsPlateforme » est liée à chaque application, en défense en profondeur : si un compte AD hors groupe était un jour synchronisé (filtre élargi), elle le refuserait (« You don't have permission »). Expression :

```
return request.user.ak_groups.filter(name="GG-AdminsPlateforme").exists()
```

3.2 Modes de raccordement des services

Deux techniques coexistent, selon ce que sait faire le produit :

— **OIDC natif** : le service parle OpenID Connect et dialogue directement avec Authentik (mapping groupe → rôle dans le produit) ;

— **forwardAuth** : le reverse proxy Traefik interroge l'*outpost embarqué* Authentik (middleware `authentik@docker`) avant de laisser passer la requête — pour les produits sans OIDC.

Service	Mode	Particularité
Grafana	OIDC natif	Rôle *Admin* automatique (mapping de groupe) ; compte local `admin` conservé (secours).
Portainer	OIDC natif	6 comptes pré-créés `Role=1`, OAuth-only (sans mot de passe) ; login local `admin` derrière **Use internal authentication**.
Netbox	OIDC natif	Élévation superuser manuelle après le 1er login (`t6_elevate_user.sh`, voir note ci-dessous) ; formulaire local `/login/` conservé.
Prometheus	forwardAuth	Middleware `authentik@docker` sur le routeur `prometheus`.
cAdvisor	forwardAuth	Idem, routeur `cadvisor`.
Uptime Kuma	forwardAuth	Routeur `uptimekuma` ; **login local désactivé** (`disableAuth`).
Pi-hole	forwardAuth	Routeur `pihole` ; **mot de passe local supprimé**.
Traefik (dashboard)	forwardAuth	Routeur `dashboard`.
Homer	public	Aucune authentification (portail de liens en lecture seule).
UniFi Network	forwardAuth	Middleware `authentik@docker` sur le routeur `unifi` (basculé le 20/09/2026) ; login local conservé (double saisie, limite produit Ubiquiti).
Vaultwarden	hors SSO	Compte propre au coffre (inchangé).
Wazuh	hors SSO	Comptes propres au SIEM (inchangé).

Netbox : élévation manuelle des superusers

Le mapping groupe → superuser (`REMOTE_AUTH_SUPERUSER_GROUPS`) n'est pas honoré par le pipeline social-auth de Netbox v4.5.9 : après le premier login OIDC d'un administrateur, l'élever depuis le CT 200 par `bash /opt/docker/netbox/t6_elevate_user.sh <username>` (détails section 4, étape 3). Ne jamais pré-créer les comptes.

Sondes Uptime Kuma

Les sondes HTTP de Kuma vers les services protégés suivent la redirection vers la porte et aboutissent sur la page de login Authentik (HTTP 200) : elles vérifient désormais *la porte SSO*, pas le backend nu. C'est voulu ; affiner avec `skip_path_regex` si une sonde doit atteindre le backend.

4 Prérequis

Prérequis

- **Poste d'administration connecté au VPN WireGuard (ou au réseau du lycée)**
- **Compte AD membre de `GG-AdminsPlateforme`**
- **Accès au coffre Vaultwarden (comptes de secours et secrets OIDC)**
- **Pour les opérations Authentik avancées : accès SSH au CT 200 (`docker exec authentik-server ...`) ou token API (item « Authentik API (token akadmin) »)**
- **Pour la gestion du groupe AD : console *Utilisateurs et ordinateurs Active Directory* ou PowerShell avec le module ActiveDirectory**

5 Ajouter ou retirer un administrateur

Toute l'habilitation tient en un seul endroit : le **groupe AD** `GG-AdminsPlateforme`. Aucun compte n'est à créer dans Authentik ni dans les services (sauf élévation Netbox, étape 3).

1 Ajouter (ou retirer) le compte dans le groupe AD

Console *Utilisateurs et ordinateurs Active Directory* : ouvrir `GG-AdminsPlateforme` (dans `OU=Profs`), onglet Membres, ajouter ou retirer le compte. En PowerShell :

```
Add-ADGroupMember -Identity 'GG-AdminsPlateforme' -Members
    <samaccountname>
Remove-ADGroupMember -Identity 'GG-AdminsPlateforme' -Members
    <samaccountname> -Confirm:$false
```

Compte hors Protected Users

Un compte membre du groupe AD Protected Users ne peut pas s'authentifier auprès d'Authentik (celui-ci utilise un *simple bind* LDAP, que Protected Users interdit : erreur LDAP 49, `data 52f`). Ne jamais ajouter d'administrateur de la plateforme à Protected Users — voir la section Dépannage.

2 Synchroniser Authentik

La synchronisation LDAP est périodique : l'ajout ou le retrait est pris en compte au plus tard au cycle suivant, sans intervention. Pour l'appliquer immédiatement, forcer la synchronisation depuis le CT 200 (pas d'endpoint API pour cela en 2026.2) :

```
docker exec authentik-server ak shell -c \
```

```
"from authentik.sources.ldap.tasks import ldap_sync; \
 ldap_sync.send('f0215b49-da55-4ff4-a377-4937911cdf8a')"
```

(l'uuid est le `pk` de la source LDAP `ad-bts-sio`). Attendre 60 à 90 secondes (tâche de fond). Vérifier dans l'interface Authentik, Directory → Users, que le compte apparaît (ajout) ou a disparu (retrait — purge automatique via `delete_not_found_objects`).

3 Effets par service

— **Grafana : rien à faire. Au premier login OIDC le compte est créé avec le rôle *Admin* (membre du groupe) — *Viewer* sinon, cas qui ne se produit pas avec le filtre actuel. Au retrait du groupe, le compte Authentik est purgé et l'utilisateur ne peut plus se connecter.**

— **Portainer : pour un *ajout*, pré-créer le compte avec le rôle administrateur (Settings → Users, ou API `POST /api/users` avec `{"Username": "<compte>", "Role": 1}` — sans mot de passe, refusé en mode OAuth). Sans pré-création, le compte est auto-créé sans droits d'administration. Au retrait, supprimer ou rétrograder le compte Portainer.**

— **Netbox : pour un *ajout*, laisser l'utilisateur se connecter une première fois (le compte est auto-créé sans privilèges), puis l'élever superuser depuis le CT 200 :**

```
bash /opt/docker/netbox/t6_elevate_user.sh <username>
```

Ne jamais pré-créer le compte dans Netbox : social-auth associe l'identité par l'attribut `sub` (inconnu avant le premier login) et suffixerait le nom pré-existant, créant un doublon. Au

***retrait*, penser à retirer aussi le superuser Netbox : l'élévation est un état figé en base, non resynchronisé.**

— **Prometheus, cAdvisor, Kuma, Pi-hole, Traefik : rien à faire — la porte forwardAuth seule décide (pas de comptes locaux, ou désactivés).**

6 Brancher un nouveau service

Choisir le mode selon le produit : s'il sait parler OIDC (OAuth2/OpenID Connect dans ses réglages d'authentification), préférer l'**OIDC natif** (variante B) qui transporte les groupes et permet le mapping de rôle ; sinon, le placer derrière la **porte forwardAuth** (variante A).

Les opérations Authentik se font par l'API (`https://auth.docker.bts.sio/api/v3/`, en-tête `Authorization: Bearer <token>` — item coffre « Authentik API (token akadmin) ») ou par l'interface d'administration (**Applications**, **Providers**, **Outposts**) ; les deux sont équivalentes.

6.1 Variante A — Porte forwardAuth (middleware Traefik)

4 **Créer le provider proxy et l'application**

Provider de type *Proxy*, mode `forward_single`, `external_host` = FQDN public du service (sans chemin) ; flows `default-provider-authorization-explicit-consent` et `default-provider-invalidation-flow`. Puis l'application liée (slug = nom du service, `policy_engine_mode=any`). En 2026.2, `invalidation_flow` est exigé sur le provider (il n'existe pas sur l'application).

```
POST /api/v3/providers/proxy/
```

```
{"name": "<service>", "authorization_flow": "<uuid flow>",
 "invalidation_flow": "<uuid invalidation>",
 "mode": "forward_single", "external_host":
    "https://<service>.docker.bts.sio"}
POST /api/v3/core/applications/
{"name": "<Service>", "slug": "<service>", "provider": <pk>,
 "launch_url": "https://<service>.docker.bts.sio", "policy_engine_mode":
    "any"}
```

Lier la policy partagée « membre GG-AdminsPlateforme » (pk `48b53a25-8cb4-4027-8421-3b2013c4346c`) à l'application (`POST /api/v3/policies/bindings/`, `order=0`, `enabled`).

5 Attacher le provider à l'outpost embarqué

L'outpost embarqué (« authentik Embedded Outpost », pk `a343c0b6-f568-4667-80d8-ba0da7db5da9`) ne sert que les providers qui lui sont explicitement attachés. Relire la liste actuelle puis la réécrire complète avec le nouveau pk — un PATCH écrase la liste :

```
GET   /api/v3/outposts/instances/a343c0b6-f568-4667-80d8-ba0da7db5da9/
PATCH /api/v3/outposts/instances/a343c0b6-f568-4667-80d8-ba0da7db5da9/
      {"providers": [<pks existants...>, <pk nouveau>]}
```

(la route est `outposts/instances/` en 2026.2 — `outposts/outposts/` renvoie 404.) Prise en compte sous 5 minutes au plus (cycle de rafraîchissement) ; la preuve est dans les logs : `docker logs authentik-server` doit afficher « `Loaded application` » avec le nom du nouveau service. Ne tester qu'après cette ligne.

6 Brancher le routeur Traefik et tester

Ajouter le label au routeur du service (dans le compose de sa stack), puis recréer :

```
- "traefik.http.routers.<routeur>.middlewares=authentik@docker"
docker compose config --quiet && docker compose up -d
```

Vérifier : `curl -sI https://<service>.docker.bts.sio` doit renvoyer `302` avec un en-tête `Location` pointant vers `auth.docker.bts.sio` (`/application/o/authorize/?...`), puis un login AD complet au navigateur doit aboutir au service. Tester aussi un compte hors groupe (refus à la porte). Si le produit possède un login local propre, décider ensuite de sa désactivation (section Rollback pour le chemin inverse).

6.2 Variante B — OIDC natif

7 Créer le provider OAuth2 et l'application

Provider *OAuth2/OpenID* (`providers/oauth2/`), `client_type=confidential`, `redirect_uris` en matching strict (l'URL de callback exacte du produit, ni plus ni moins), `sub_mode=hashed_user_id`, `issuer_mode=per_provider`, `invalidation_flow` renseigné (exigé par l'API 2026.2 à la création). Attacher les quatre property mappings : `openid` et `profile` (intégrés), `groups-custom` (pk `4863d720-d48c-48bf-8b46-ae5898fab834` — renvoie la liste des groupes, sans mapping intégré « groups » en 2026.2) et `email-with-fallback`

(pk `280070cf-3ccd-478c-9858-55179eb1c01f` — repli `<user>@bts.sio` quand l'attribut `mail` AD est vide, ce qui est le cas de nos comptes).
Puis l'application liée + binding de la policy partagée (comme variante A, étape 1). Relever le `client_id` et le `client_secret` affichés à la création : les verser immédiatement au coffre (item « OIDC <service> (Authentik) »).

8 Configurer le produit

Renseigner côté produit : URLs `/application/o/authorize/`, `/application/o/token/`, `/application/o/userinfo/` de `auth.docker.bts.sio`, `client_id`/`client_secret`, scopes `openid profile email groups`, identifiant = `preferred_username`, et le mapping groupe → rôle du produit (ex. Grafana : `role_attribute_path` sur le claim `groups`).

9 Confiance TLS interne (point critique)

Le produit contacte Authentik *en serveur à serveur* : le conteneur doit faire confiance à la racine interne « BTS SIO Root CA » (MO-SEC-005/006), faute de quoi le login échoue sur une erreur `x509` *après* la saisie des identifiants. Racine disponible sur le CT 200 (`/opt/docker/homer/assets/BTS-SIO-Root-CA.crt`) ; trois patrons éprouvés :

- **Produit en Go (Portainer…) : monter la racine sous son nom hashé dans le répertoire du bundle — `./BTS-SIO-Root-CA.crt:/etc/ssl/certs/70d93f95.0:ro`**

(le hash s'obtient par `openssl x509 -hash -in BTS-SIO-Root-CA.crt`). Ne jamais poser `SSL_CERT_FILE` seul (remplace tout le bundle).

- **Produit en Python/requests (Netbox…) : `requests` ne lit que la variable d'environnement `REQUESTS_CA_BUNDLE` — monter un bundle combiné (bundle système + racine concaténée) par-dessus `/etc/ssl/certs/ca-certificates.crt` et poser `REQUESTS_CA_BUNDLE` sur ce chemin.**
- **Produit avec réglage dédié (Grafana…) : utiliser sa variable (ex. `GF_AUTH_GENERIC_OAUTH_TLS_CLIENT_CA`).**

10 Tester (positif, négatif, secours)

1. **Positif : login AD d'un membre du groupe au navigateur → accès avec le rôle attendu (vérifier le rôle dans le produit, pas seulement l'accès).**
2. **Négatif : compte AD hors groupe → refus « Invalid password » à la porte (le compte n'existe pas dans Authentik), aucune session créée côté produit.**
3. **Secours : le compte local administrateur du produit (break-glass) doit rester fonctionnel — le vérifier avant de généraliser.**

7 Vérification

Vérification

Après toute opération sur le SSO :

- ☐ `https://homer.docker.bts.sio` **répond 200 (public) et les services protégés répondent 302 vers** `auth.docker.bts.sio` **sans session**
- ☐ **Un login AD (membre du groupe) aboutit sur chaque service concerné, avec le rôle attendu**
- ☐ **Un compte hors groupe est refusé à la porte (« Invalid password »), sans session**
- ☐ **Les comptes de secours locaux (Grafana, Portainer, Netbox) et** `akadmin` **restent fonctionnels**
- ☐ `docker ps` **sur le CT 200 : tous les conteneurs *healthy***
- ☐ **Tout secret nouvellement créé (client secret OIDC, mot de passe régénéré) est versé au coffre Vaultwarden**

8 Rollback

11 **Retirer un service de la porte forwardAuth**

Retirer le label `middlewares=authentik@docker` **du routeur et recréer la stack. Pour un service dont le login local a été désactivé (Kuma, Pi-hole), réactiver aussi l'authentification locale, sinon le service devient injoignable ou ouvert :**

```
# Pi-hole : reposer un mot de passe (a chaud, sans restart)
docker exec pihole pihole-FTL --config webserver.api.password "<nouveau>"
# Kuma : depuis le CT 200, client uptime-kuma-api sur
    http://localhost:3001
#   login() sans identifiants (auto-login), puis
#   set_settings(disableAuth=False, ...valeurs actuelles...), puis
    change_password(...)
```

> **Pour un retour complet en arrière, retirer enfin le pk du provider de la liste de l'outpost (PATCH liste complète, étape 2 de la variante A).**

9 Dépannage

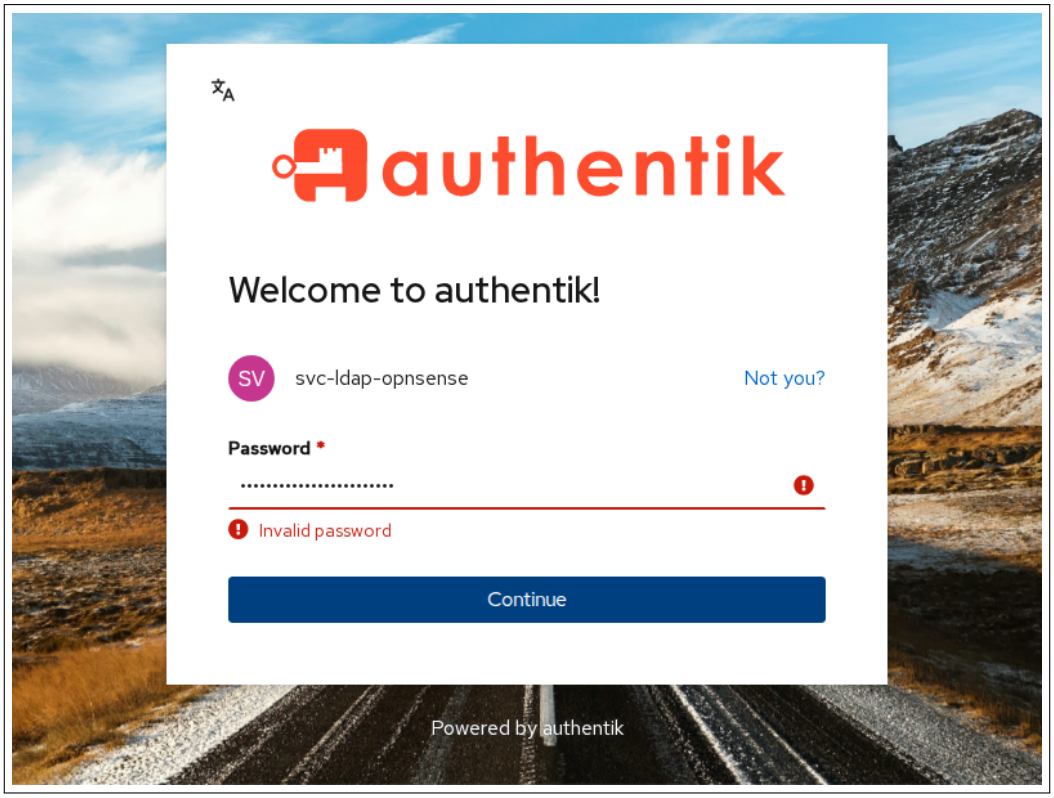

Figure 2 – Le refus générique « Invalid password » d'Authentik : il signifie que le compte est inconnu (hors groupe, donc filtré par la source LDAP) *ou* que le bind LDAP a échoué — pas nécessairement que le mot de passe est faux.

Problème	**Solution**
« Invalid password » alors que le mot de passe est correct	Vérifier d'abord l'appartenance du compte au groupe AD **Protected Users** : il bloque le *simple bind* LDAP utilisé par Authentik (erreur 49, `data 52f`, indistinguable d'un mot de passe faux côté client). Retirer le compte de Protected Users. Si le compte n'y est pas : vérifier qu'il est bien membre de `GG-AdminsPlateforme` et qu'une synchronisation LDAP a eu lieu (étape 2 de la section 4).
Page « You don't have permission » après login	Le compte existe dans Authentik mais la policy groupe l'a refusé : il n'est pas (ou plus) membre de `GG-AdminsPlateforme`, ou le filtre LDAP a été élargi. Corriger l'appartenance au groupe et resynchroniser.
Boucle de redirection OIDC (le service renvoie sans fin vers Authentik)	La `redirect_uri` déclarée par le produit ne correspond pas *exactement* à celle du provider (matching strict : schéma, hôte, chemin, slash final). Corriger l'un des deux côtés. Vérifier aussi l'horloge du serveur (validité du jeton) et, pour Grafana, que `GF_SERVER_ROOT_URL` est renseigné (sinon l'URL de callback part en `localhost`).

Problème	Solution
502/504 sur un service forwardAuth	Le provider n'est pas attaché à l'outpost embarqué, ou l'outpost n'a pas encore rafraîchi (jusqu'à 5 minutes). Contrôler la liste `providers` de l'outpost et attendre la ligne « `Loaded application` » dans `docker logs authentik-server`. Vérifier aussi que l'option `authentik_host=https://auth.docker.bts.sio` est présente dans la configuration de l'outpost (sans elle, les redirections pointent sur `0.0.0.0:9000`).
Erreur `x509: certificate signed by unknown authority` dans les logs du service	Le conteneur ne connaît pas la racine interne « BTS SIO Root CA ». Appliquer le patron TLS adapté (section 5, étape « Confiance TLS interne »).
Plus aucun accès administrateur (lockout)	Voies de secours conservées : `akadmin` sur `https://auth.docker.bts.sio` (item « Authentik SSO » du coffre) ; comptes locaux `admin` de Grafana (formulaire `/login`), Portainer (bouton **Use internal authentication**) et Netbox (formulaire `/login/`) — tous au coffre.
Le poste d'administration ne résout plus les `*.docker.bts.sio`	Panne connue de `systemd-resolved` sur le lien VPN (validation DNSSEC en échec) : `resolvectl reset-server-features` puis `resolvectl flush-caches`, sans privilège.

10 Limites connues

— **Accès de secours IP — traités le 21/09/2026 (MO-NET-009)** : les accès directs par IP n'écoutent plus que sur le VLAN PLT (`http://172.31.250.20:3001` pour Kuma, `http://172.31.250.20:8053/admin/` pour Pi-hole, etc.) — invisibles du réseau plat et des réseaux étudiants, joignables par les admins (VLAN admin, VPN collègues, Wi-Fi profs pour le portail 443). La double écoute transitoire du port 443 sur `10.0.112.20` est filtrée par le pare-feu Proxmox en attendant la bascule DNS finale.

— **UniFi Network — double saisie** : basculé derrière la porte le 20/09/2026 (forwardAuth, routeur `unifi`), mais Ubiquiti ne délègue pas l'authentification d'administration : la porte AD est suivie du **login local** du contrôleur (compte `admin@bts.sio`, coffre). L'accès direct `https://10.0.112.228:8443` est restreint aux sources de l'alias OPNsense `UniFi_Console_Admin_Sources` (VLAN admin `10.0.112.0/24` + VPN collègues `10.10.30.0/24`) par deux règles flottantes (PASS puis BLOCK journalisé) — voir MO-NET-006.

— **Protected Users désormais vide** : le retrait du compte concerné a été validé par le demandeur le 14/09/2026. Si un administrateur de la plateforme y est réintégré (GPO, réflexe de bonne pratique), son SSO cassera sans message explicite (voir Dépannage) — ré-évaluer la stratégie dans ce cas.

— **Élévation Netbox figée** : le superuser Netbox n'est pas resynchronisé depuis l'AD ; un retrait du groupe ne retire pas le superuser (action manuelle, section 4). Une synchronisation dynamique passerait par un handler personnalisé du pipeline social-auth (hors périmètre actuel).

— **Consentement mémorisé** : Authentik ne redemande le consentement qu'à la première connexion de chaque couple utilisateur/application — son absence ensuite est normale.

— **Sondes Kuma** : elles vérifient la porte SSO et non le backend nu (voir la note de la section 3).

11 Mises à jour des conteneurs

Le raccordement SSO **survit aux mises à jour d'images** (`docker compose pull && docker compose up -d`) sans aucun reparamétrage : toute la configuration vit hors des images, dans les volumes de données, les fichiers montés depuis l'hôte et les compose eux-mêmes.

— **Authentik** : providers, applications, policies, mappings, filtre LDAP et configuration de l'outpost sont en base PostgreSQL (volume).

— **Grafana** : paramètres OIDC dans le compose (`GF_AUTH_GENERIC_OAUTH_*`), secrets dans le `.env`, comptes et rôles dans le volume `grafana_data`.

— **Portainer** : `OAuthSettings` et les six comptes administrateurs dans la base interne (volume `portainer_data`).

— **Netbox** : bloc OIDC dans le `configuration.py` monté depuis l'hôte, bundle CA monté, comptes en PostgreSQL. **Point de vigilance** : lors d'une montée de version majeure de Netbox, consulter les notes de version pour d'éventuelles nouvelles clés de configuration obligatoires et réconcilier notre fichier monté avec le `configuration.py` d'origine de la nouvelle image.

— **Traefik** : middleware `authentik@docker` et labels des routeurs dans les fichiers compose.

— **Kuma** : `disableAuth` en base (volume) ; **Pi-hole** : haché de mot de passe vide persisté dans `pihole.toml`.

— **Active Directory** : le groupe `GG-AdminsPlateforme` vit dans l'annuaire, indépendamment de Docker.

Deux interdictions et un rappel :

— ne jamais utiliser `docker compose down -v` (l'option `-v` détruit les volumes, donc toute la configuration SSO) ;

— ne pas recréer les comptes Netbox à la main avant leur premier login (doublons suffixés, voir section 3) ;

— les certificats ADCS ont une validité d'un an : leur renouvellement relève des MO-SEC-005/006, pas de ce document.

12 Voir aussi

— **MO-PLT-010** : Accéder au portail des services — point d'entrée et exceptions au SSO
— **MO-SEC-005** : Certificats ADCS des services — émission des certificats TLS (dont ceux d'Authentik et des services)
— **MO-SEC-006** : Architecture PKI interne et HTTPS des services — la chaîne de confiance « BTS SIO Root CA »
— **MO-NET-008** : Architecture du Wi-Fi pédagogique — l'autre usage du couple AD/SSO (802.1X)

Références

— Authentik 2026.2 — documentation officielle (`goauthentik.io/docs`) : providers OAuth2 et proxy, property mappings, outposts
— Spécification du chantier (dépôt git) : `docs/superpowers/specs/2026-09-14-sso-ad-services-docker-design.md`
— Traefik v2.11 — middleware *ForwardAuth*
— Documentation produits : Grafana 13 (*Generic OAuth*), Portainer CE 2.41 (*OAuth settings*), Netbox v4.5 (*social-auth OIDC*), Pi-hole v6 (`pihole-FTL -config`), Uptime Kuma 1.x