Projet Laser-Campus Numérique

DOSSIER DE CONCEPTION

Modernisation Infrastructure IT

INFORMATIONS DOCUMENT

Version : 1.1
Date : 01/02/2026
Auteurs : Alexandru Nasui, Benson François
Statut : En conception

1. INTRODUCTION

1.1 Contexte

Laser-Campus Numérique souhaite moderniser son infrastructure IT pour améliorer la gestion des utilisateurs et l'accès aux ressources numériques.

1.2 Objectifs du projet

- Centraliser l'authentification (LDAP/Kerberos)
- Automatiser les accès aux ressources (NFS)
- Virtualiser les postes de travail (VDI)

1.2 Périmètre

- Projet 1 : Gestion centralisée utilisateurs
- Projet 2 : Solution VDI automatisée

1.4 Contraintes techniques imposées

- Hyperviseur : Proxmox VE 9.X bare metal
- Services : OpenLDAP, Kerberos, NFS
- Frontend : Flask
- Web server : Nginx (HTTPS)
- IaC : OpenTofu, Ansible

2. ARCHITECTURE GLOBALE

2.1 Vue d'ensemble
Description des composants principaux :

- **Couche authentification** : OpenLDAP/NIS + Kerberos (VM1-AUTH) pour la gestion centralisée des identités et l'authentification sécurisée

- **Couche stockage** : Serveur NFS (VM2-STORAGE) avec espaces privés quotés et partage commun

- **Couche présentation** : Flask + Nginx (VM3-WEB) pour l'interface web HTTPS et la création autonome de comptes.

- **Couche virtualisation** : Proxmox VE bare metal pour l'hébergement et le provisionnement automatisé des VMs

- **Couche DNS** : DNS dynamique (VM4-DNS) pour l'enregistrement automatique des ressources

3. ARCHITECTURE LOGIQUE

3.1 Composants fonctionnels

3.1.1 OpenLDAP/NIS

- Annuaire central des utilisateurs
- Gestion des groupes et droits d'accès
- Stockage des identifiants et profils
- Port : 636 (LDAPS)

3.1.2 Kerberos

- Authentification sécurisée via tickets
- Coordination avec LDAP pour l'émission de tickets
- Chiffrement AES-256
- Port : 88

3.1.3 Serveur NFS

- Stockage privé par utilisateur avec quotas
- Stockage partagé commun
- Intégration Kerberos
- S'appuie sur LDAP pour les droits d'accès

3.1.4 Application Flask

- Interface utilisateur Web avec formulaire d'authentification
- Création autonome de comptes
- Provisionnement de VMs (Projet 2)
- Mode invité avec accès limité

3.1.5 Nginx

- Reverse proxy HTTPS
- Point d'entrée unique sécurisé
- Gestion des certificats SSL
- Redirection HTTP→HTTPS

3.1.6 DNS Dynamique

- Enregistrement automatique des VMs créées
- Résolution de noms pour les services

3.2 Flux d'authentification

- Scénario utilisateur distant (BYOD) :
  - ⇨ Utilisateur → Interface Web Flask (HTTPS via Nginx)
  - ⇨ Flask → LDAP (vérification identité)
  - ⇨ LDAP + Kerberos → Émission ticket
  - ⇨ Accès → Partages NFS selon droits LDAP

- Scénario utilisateur présentiel (Linux Mint) :
  - ⇨ Session locale → Authentification LDAP/NIS
  - ⇨ Montage automatique NFS
  - ⇨ Accès ressources selon profil utilisateur

- Scénario utilisateur invité :
  - ⇨ Interface Web → Mode invité
  - ⇨ Accès limité aux ressources communes uniquement
  - ⇨ Pas de stockage privé persistant

3.3 Flux de données

- ⇨ Internet → Routeur (192.168.100.254)
- ⇨ Routeur → Nginx (192.168.100.30:443)
- ⇨ Nginx → Flask
- ⇨ Flask → LDAP (192.168.100.10) pour authentification
- ⇨ Flask → NFS (192.168.100.20) pour accès données
- ⇨ Linux Mint (192.168.100.201-250) → LDAP auth → NFS auto-mount

4. ARCHITECTURE PHYSIQUE

4.1 Infrastructure Proxmox

Configuration serveur hôte (bare metal) :

- CPU : 8+ cœurs (support multiples VMs)
- RAM: 32+ Go
- Stockage : 500 Go minimum (SSD recommandé)
- Réseau : 2 interfaces (gestion + production)
- IP : 192.168.100.250

4.2 Machines virtuelles

VM	Service	IP	vCPU	RAM	Disque	OS
VM1-AUTH	OpenLDAP/NIS + Kerberos	192.168.100.221	2	4 Go	20 Go	Debian/Ubuntu
VM2-STORAGE	NFS + Sauvegardes	192.168.100.220	2	4 Go	100+ Go	Debian/Ubuntu
VM3-WEB	Flask + Nginx (HTTPS)	192.168.100.210	2	4 Go	20 Go	Debian/Ubuntu
VM4-DNS	DNS dynamique	192.168.100.200	1	2 Go	10 Go	Debian/Ubuntu
VM-USERS	Debian Cinnamon (VDI)	192.168.100.1-100	2	4 Go	30 Go	Debian

Note : VM-USERS créées à la demande via OpenTofu/Ansible

4.3 Plan d'adressage

Réseau de production : 192.168.100.0/24

Équipement	IP	Rôle
Proxmox Host	192.168.100.250	Hyperviseur
VM1-AUTH	192.168.100.221	LDAP/Kerberos
VM2-STORAGE	192.168.100.220	NFS
VM3-WEB	192.168.100.210	Flask/Nginx
VM4-DNS	192.168.100.200	DNS dynamique

- Pool VMs utilisateurs 192.168.100.1-100 VDI auto-créées
- Postes Linux Mint 192.168.100.101-199 Clients présentiel
- Passerelle/Routeur 192.168.100.254 Gateway
- DNS primaire : 192.168.100.200

4.4 Réseau et connectivité

- Bridge principal : vmbr0
  - Connecté à l'interface physique
  - Toutes les VMs attachées à ce bridge
  - Mode : NAT ou Bridge selon accès Internet

- Segmentation optionnelle (VLANs) :
  - VLAN 10 : Management (Proxmox)
  - VLAN 100 : Production (VMs services)
  - VLAN 200 Utilisateurs (VDI + Linux Mint)

- Stockage Proxmox :
  - Disques des VMs : /var/lib/vz
  - Templates VM (Debian Cinnamon) : /var/lib/vz/template
  - Backups Proxmox: /var/lib/vz/dump

- Stockage NFS (VM2-STORAGE) :
  - Espaces privés avec quotas : /home/users/*
  - Espace partagé : /shared/common
  - Sauvegardes automatisées : /backups/*

5. ARCHITECTURE RÉSEAU

5.1 Matrice des flux

Source	Destination	Port	Protocole	Explications
Client BYOD	Nginx (VM3-WEB)	443	HTTPS	Interface Web sécurisée
Nginx	Flask	5000	HTTP	Communication interne
Flask	OpenLDAP (VM1-AUTH)	636	LDAPS	Vérification identités
Flask	Kerberos (VM1-AUTH)	88	Kerberos	Obtention tickets
Clients	NFS (VM2-STORAGE)	2049	NFS	Accès stockage
Linux Mint	OpenLDAP	636	LDAPS	Authentification locale
Flask	API Proxmox	8006	HTTPS	Provisionnement VMs
Toutes VMs	DNS (VM4-DNS)	53	DNS	Résolution noms
VMs	Routeur	Any	Any	Accès Internet

5.2 Règles de firewall

- VM1-AUTH (LDAP/Kerberos) :
  - Entrant : 636 (LDAPS), 88 (Kerberos), 22 (SSH admin)
  - Sortant : DNS (53), NTP (123)

- VM2-STORAGE (NFS) :
  - Entrant : 2049 (NFS), 111 (RPC), 22 (SSH admin)
  - Sortant : DNS (53), LDAP (636)

- VM3-WEB (Flask/Nginx) :
  - Entrant : 443 (HTTPS public), 22 (SSH admin)
  - Sortant : LDAPS (636), NFS (2049), API Proxmox (8006), DNS (53)

- VM4-DNS :
  - Entrant : 53 (DNS), 22 (SSH admin)
  - Sortant : DNS upstream (53)

- Proxmox Host :
  - Entrant : 8006 (Web UI), 22 (SSH admin)
  - Sortant : Repositories apt, NTP

6. SÉCURITÉ

6.3 Gestion des droits

- Groupes LDAP suggérés :
  - users : Utilisateurs standard (accès privé + commun)
  - guests : Invités (accès commun lecture seule)
  - admins : Administrateurs (accès complet)
  - vdi-users : Utilisateurs autorisés à provisionner des VMs

- Principe du moindre privilège :
  - Séparation des rôles par groupes LDAP
  - Quotas individuels sur espaces privés
  - ACLs NFS basées sur les groupes LDAP

6.5 Sauvegardes

- Stratégie (VM2-STORAGE) :
  - Quotidien : données utilisateurs (/home/users/*)
  - Hebdomadaire : configurations système
  - Rétention : 7 jours (quotidiennes) + 4 semaines (hebdomadaires)
  - Stockage : /backups/* sur VM2-STORAGE
  - Backups Proxmox : /var/lib/vz/dump

7. AUTOMATISATION (Projet 2)

7.2 Processus de provisionnement

- Workflow création VM (VDI) :
  - ⇨ Utilisateur authentifié → Demande VM via Flask
  - ⇨ Flask → API Proxmox (192.168.100.1:8006)
  - ⇨ OpenTofu → Création VM Debian Cinnamon depuis template
  - ⇨ Attribution IP dynamique (pool 192.168.100.100-200)
  - ⇨ Ansible → Configuration post-installation (montage NFS, configuration utilisateur)
  - ⇨ DNS dynamique → Enregistrement automatique
  - ⇨ Notification email → Utilisateur informé

8. DIMENSIONNEMENT

8.1 Capacité

- Nombre utilisateurs : 50
- VMs simultanées : ~20 (pool 192.168.100.100-200 = 101 IPs disponibles)
- Stockage NFS : 500 Go (100+ Go pour VM2-STORAGE)
- Postes Linux Mint présentiel : 50 (192.168.100.201-250)

8.2 Performances

- Ressources totales requises (hors VMs utilisateurs) :
  - vCPU : 9 (2+2+2+2+1)
  - RAM : 18 Go
  - Stockage : 170 Go

- Avec 20 VMs utilisateurs actives :
  - vCPU : 49 (9+40)
  - RAM : 98 Go (18+80)
  - Stockage : 770 Go (170+600)

La configuration Proxmox recommandée couvre largement ces besoins

9. RISQUES ET CONTRAINTES

- Risques identifiés :
  - Point unique de défaillance : Proxmox en nœud unique
  - Mitigation : Sauvegardes régulières, documentation complète pour reconstruction rapide
  - Saturation réseau : 20 + Vms sur un seul bridge
    - Mitigation : Monitoring bande passante, segmentation VLAN optionnelle
  - Capacité stockage NFS : Croissance données utilisateur
    - Mitigation : Quotas stricts, politique de rétention, monitoring espace disque
  - Compromission VM3-WEB : Exposition Internet
    - Mitigation : HTTPS obligatoire, certificats valides, WAF optionnel, mises à jour régulières
  - Complexité Kerberos : Configuration sensible
    - Mitigation : Documentation détaillée, tests authentification multi-scénarios