Compte Rendu - SAE 5.01

Lernould Adam - Leroy Rémi - Picamal Julien - Hault Théo

Sommaire :

Dossier Technique de Référence : Infrastructure de Salle de TP Virtuelle

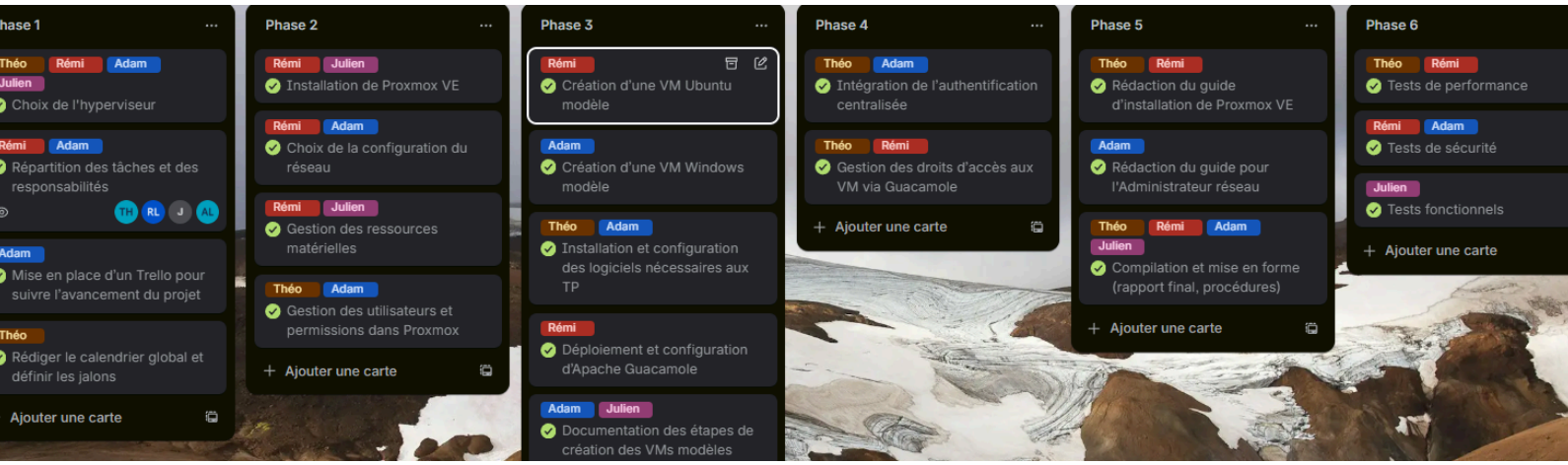
1. Contexte et Objectifs du Projet

L'objectif de ce projet est la conception, le déploiement et la sécurisation d'une infrastructure de virtualisation complète destinée à héberger des environnements de Travaux Pratiques (TP) pour une classe d'étudiants.

L'infrastructure doit garantir :

- **Performance :** Utilisation optimale des ressources matérielles via la paravirtualisation.
- **Isolation :** Ségrégation stricte des données et des ressources entre les utilisateurs (Multi-tenant).
- **Accessibilité :** Accès distant sécurisé aux environnements de travail (RDP/Guacamole).
- **Supervision :** Monitoring proactif de l'état de santé du système et des services.
- **Gestion des Données :** Stockage centralisé, sécurisé et contingenté (Quotas).

Planning :



2. Infrastructure de Virtualisation (Couche Hôte)

Le socle technique repose sur l'hyperviseur de type 1 (Bare-Metal) **Proxmox VE**.

2.1. Installation et Configuration de l'Hyperviseur

Procédure d'installation :

1. Téléchargement de l'ISO Proxmox VE (PVE) depuis le site officiel.
2. Création d'un support bootable (Clé USB avec Rufus/Ventoy).
3. Installation sur le serveur physique (sélection du disque cible, configuration locale).

Configuration Réseau (Pont) : Pour permettre aux VMs de communiquer avec le réseau physique, un pont Linux (`vmbr0`) a été configuré via le fichier `/etc/network/interfaces`. Ce pont agit comme un switch virtuel reliant les interfaces virtuelles des VMs à l'interface physique du serveur.

- ➤ **Adresse IP Statique :** Attribution d'une IP fixe au serveur pour l'administration.
- ➤ **Gateway/DNS :** Configuration de la passerelle du laboratoire pour l'accès Internet.

Stratégie de Stockage :

- ➤ **`local` (Directory) :** Stockage fichier classique réservé aux images ISO d'installation et aux fichiers de sauvegarde (VZDump).
- ➤ **`local-lvm` (LVM-Thin) :** Volume logique réservé aux disques virtuels des VMs. Le choix du *Thin Provisioning* permet d'allouer de l'espace disque à la demande, optimisant l'espace réel consommé (seul l'espace écrit est consommé).
- ➤ **`ExStorage` :** Volume logique réservé principalement aux Backups.

2.2. Ingénierie des Modèles de Machines Virtuelles (Templates)

Pour industrialiser le déploiement des postes de travail étudiants, des images "Master" (Templates) ont été créées et optimisées.

Optimisation des Performances (Paravirtualisation)

L'émulation matérielle par défaut (Disque IDE, Réseau E1000) génère une surcharge CPU (Overhead). Nous avons basculé vers la paravirtualisation **VirtIO** pour des performances quasi-natives. Cette technologie permet à l'OS invité de communiquer directement avec l'hyperviseur.

Procédure d'installation des pilotes VirtIO sur Windows :

1. **Prérequis :** Téléchargement et upload de l'ISO `virtio-win.iso` sur Proxmox.
2. **Configuration VM :**
   - ➤ Disque Dur : Bus **SCSI** (avec contrôleur VirtIO SCSI).
   - ➤ Carte Réseau : Modèle **VirtIO (paravirtualized)**.
3. **Installation des Drivers (In-Guest) :**
   - ➤ Montage de l'ISO `virtio-win.iso` dans le lecteur CD virtuel.
   - ➤ Ouverture du *Gestionnaire de Périphériques*.
   - ➤ Mise à jour manuelle des périphériques inconnus (Triangle jaune) en pointant vers la racine du CD :.
     - ➥ *Contrôleur SCSI* ➔ Pilote **viostor** (I/O Disque).
     - ➥ *Périphérique Système de Base* ➔ Pilote **VirtIO Serial** (Communication Agent).

Intégration de l'Agent Invité (QEMU Guest Agent)

Ce service est fondamental pour l'orchestration des VMs par Proxmox.

Mise en œuvre :

1. **Installation :** Exécution du package `qemu-ga-x86_64.msi` (présent sur l'ISO VirtIO) à l'intérieur de la VM.
2. **Activation :** Dans Proxmox, menu *Options* de la VM > Activer "QEMU Guest Agent".
3. **Fonctionnalités activées :**
   - ➤ Remontée des adresses IP de la VM dans l'interface Proxmox (Onglet *Summary*).
   - ➤ Exécution propre des commandes d'arrêt (Shutdown ACPI) via le canal VirtIO Serial.

Création du Master Windows 10/11 (Sysprep)

Pour éviter les conflits d'identité (SID dupliqués) lors du déploiement massif.

Procédure de préparation :

1. Installation de l'OS, mises à jour Windows Update et logiciels pédagogiques (7Zip, VSCode).
2. Activation du Bureau à distance (RDP) et autorisation du groupe "Utilisateurs du domaine".
3. **Généralisation (Sysprep) :**
   - ➤ Lancement d'une invite de commande en Administrateur.
   - ➤ Exécution : `%windir%\system32\sysprep\sysprep.exe /generalize /oobe /shutdown`.
   - ➤ *Explication :* `/generalize` supprime le SID unique, `/oobe` force l'assistant de bienvenue au prochain démarrage, `/shutdown` éteint la VM pour la conversion.
4. **Conversion :** Clic droit sur la VM éteinte > *Convert to Template*.

Création du Master Ubuntu 22.04

Préparation d'une image Linux prête pour l'intégration AD.

Procédure :

1. **Installation des paquets :** `apt install realmd sssd sssd-tools adcli libpam-sss`.
2. **Configuration Réseau (Netplan) :** Modification de `/etc/netplan/50-cloud-init.yaml` pour ajouter `dhcp4-overrides: use-dns: false`. Cela force la machine à ignorer les DNS du DHCP et prépare le terrain pour l'utilisation des DNS de l'AD.

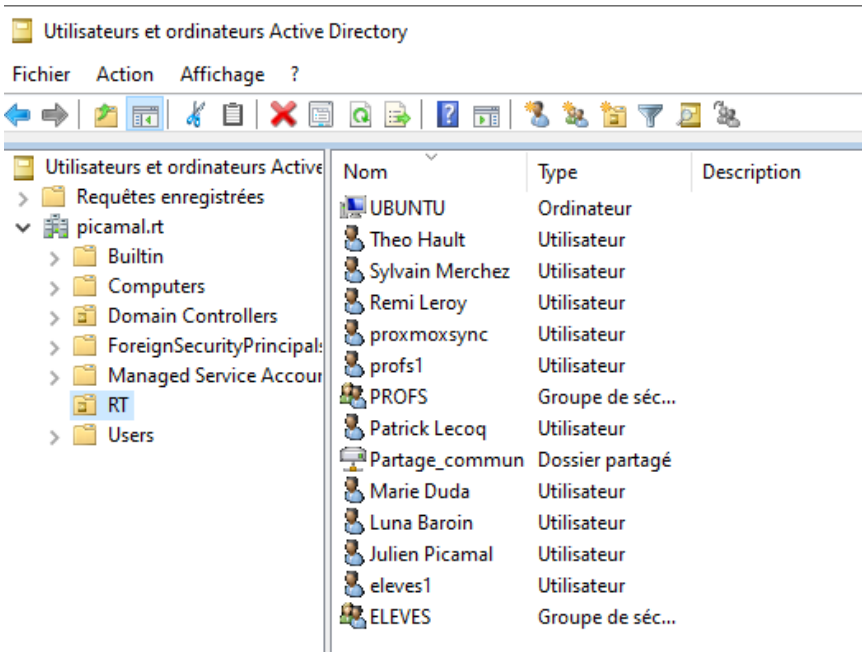
```
 GNU nano 7.2                  /etc/netplan/50-cloud-init.yaml
network:
  version: 2
  ethernets:
    ens18:
      dhcp4: true
      dhcp4-overrides:
        use-dns: false
      nameservers:
        addresses: [192.168.1.203, 8.8.8.8]
        search: [picamal.rt]
```

3. Infrastructure d'Identité : Active Directory (Windows Server)

La gestion centralisée des identités repose sur un Contrôleur de Domaine (DC) sous Windows Server 2022.

3.1. Architecture Logique

- **Domaine :** `picamal.rt`
- **Services :** Installation des rôles AD DS (Annuaire) et DNS.
- **Groupes de Sécurité :** Création des groupes globaux `PROFS` et `ELEVES` pour l'attribution des droits.
- **Organisation (OU) :** Création d'une structure hiérarchique :
  - `OU=RT` (Racine) > `PROFS` et `ELEVES`.



3.2. Intégration des Postes Clients

Intégration des Postes Clients Windows

La jonction au domaine nécessite une configuration réseau rigoureuse.

Procédure de Jonction :

1. **DNS Statique :** Configuration de la carte réseau cliente avec l'IP du Contrôleur de Domaine (`192.168.1.203`) comme DNS Préféré.
2. **Désactivation IPv6 :** Décochage du protocole IPv6 pour forcer la résolution de noms via le DNS local IPv4.
3. **Synchronisation NTP :** Vérification et correction manuelle de l'heure pour respecter la tolérance Kerberos (max 5 minutes d'écart).
4. **Jonction :** Via *Paramètres* > *Système* > *Renommer ce PC (avancé)* > *Domaine : picamal.rt*.

Intégration des Postes Clients Linux (Ubuntu)

Intégration hétérogène réalisée via la suite **SSSD** (System Security Services Daemon).

Procédure détaillée :

1. **DNS :** Application de la configuration Netplan pointant vers le serveur AD.
2. **Jonction :** Exécution de la commande `realm join -U Administrateur picamal.rt --verbose`.
    - ➤ *Action technique :* Authentification Kerberos, création de l'objet ordinateur dans l'AD et génération du fichier `/etc/krb5.keytab`.

```
root@ubuntu-client:/home/test# realm discover picamal.rt
picamal.rt
  type: kerberos
  realm-name: PICAMAL.RT
  domain-name: picamal.rt
  configured: kerberos-member
  server-software: active-directory
  client-software: sssd
  required-package: sssd-tools
  required-package: sssd
  required-package: libnss-sss
  required-package: libpam-sss
  required-package: adcli
  required-package: samba-common-bin
  login-formats: %U
  login-policy: allow-realm-logins
```

3. **Adaptation SSSD (`/etc/sssd/sssd.conf`) :**
    - ➤ `use_fully_qualified_names = False` : Permet la connexion avec le login court (`mduda`).
    - ➤ `ad_gpo_access_control = permissive` : Désactive le filtrage strict des GPO Windows (résolvant l'erreur `su: permission denied`).
4. **Création du Home Directory (PAM) :**
    - ➤ Commande : `pam-auth-update --enable mkhomedir`.
    - ➤ *Action :* Active le module PAM qui crée automatiquement le répertoire `/home/%u` à la volée lors de la première connexion.

```
 GNU nano 7.2                    /etc/sssd/sssd.c

[sssd]
domains = picamal.rt
config_file_version = 2
services = nss, pam

[domain/picamal.rt]
default_shell = /bin/bash
krb5_store_password_if_offline = True
cache_credentials = True
krb5_realm = PICAMAL.RT
realmd_tags = manages-system joined-with-adcli
id_provider = ad
fallback_homedir = /home/%u
ad_domain = picamal.rt
use_fully_qualified_names = False
ldap_id_mapping = True
access_provider = ad
ad_gpo_access_control = permissive
```

3.3. Automatisation de l'Onboarding Utilisateurs (PowerShell)

Afin d'éviter les tâches répétitives et les erreurs de saisie lors de la rentrée d'une nouvelle promotion, nous avons développé un script PowerShell permettant le provisioning en masse des comptes utilisateurs.

Fonctionnement du script : Le programme ingère un fichier source `.csv` standardisé contenant les attributs clés (Nom, Prénom, Groupe d'appartenance : Élèves ou Profs) et exécute la logique suivante pour chaque entrée :

1. **Vérification d'existence :** Contrôle préalable pour éviter les doublons dans l'Active Directory.
2. **Création du compte :** Génération de l'objet utilisateur avec les attributs adéquats.
3. **Affectation de Groupe :** Ajout automatique au groupe de sécurité (ELEVES ou PROFS).
4. **Génération du Répertoire Personnel :**

- ➤ Création physique du dossier sur le serveur de fichiers.
- ➤ Attribution des ACLs NTFS (Droit exclusif pour l'utilisateur).
- ➤ Configuration de l'attribut AD `HomeDirectory` pour monter automatiquement le lecteur réseau (Z:) à la connexion.

Script : **Script PowerShell**

4. Portail Web de Gestion des Machines Virtuelles

4.1. Objectifs

- ➤ Automatiser la gestion des VMs étudiants.
- ➤ Garantir l'isolation par ACL Proxmox.
- ➤ Gérer automatiquement les connexions RDP via Apache Guacamole.
- ➤ Remplacer toute utilisation du compte root.

4.2. Architecture Fonctionnelle

Composants :

A. Frontend Flask :

- ➤ Authentification utilisateur (normalisation automatique : `user@picamal.rt`).
- ➤ Dashboard VM (démarrage, arrêt, suppression).
- ➤ Formulaire de création VM (ISO ou template).

Script Flask : Script Flask

B. Proxmox VE API (proxmoxer) :

- ➤ Toutes les actions respectent strictement les ACL de l'utilisateur :
  - ➥ Création / clonage VM
  - ➥ Gestion alimentation
  - ➥ Suppression
  - ➥ Attribution au pool étudiant

C. Guacamole (backend MySQL) :

- ➤ Provisioning automatique :
  - ➥ création de compte utilisateur,
  - ➥ création de connexion RDP (basée sur IP détectée via guest agent),
  - ➥ suppression dynamique lors de l'arrêt/suppression VM.

4.3. Gestion des Droits Proxmox (ACLs)

L'objectif de cette configuration est de sécuriser l'utilisation du portail. Chaque utilisateur doit pouvoir créer, démarrer et gérer uniquement ses propres VMs.

Rôles et droits attribués :

- ➤ **Rôle UserVMAdminPool :**
  - ➥ *Chemin :* `/pool/<user-pool>`
  - ➥ *Droits :* `VM.Allocate`, `VM.Clone`, `VM.PowerMgmt`, `VM.Config.*`, `VM.Audit`, `Pool.Allocate`.
  - ➥ *Fonction :* Gestion complète des VMs personnelles.
- ➤ **Rôle PVETemplateUser :**
  - ➥ *Chemins :* `/vms/401` et `/vms/402` (Templates).
  - ➥ *Droits :* `VM.Clone`, `VM.Audit`.
  - ➥ *Restrictions :* Impossible de modifier ou démarrer le template.
- ➤ **Rôle PVEDatastoreUser :**
  - ➥ *Chemin :* `/storage/local-lvm`.
  - ➥ *Droits :* `Datastore.AllocateSpace`.
  - ➥ *Fonction :* Nécessaire pour la création de disques VM.
- ➤ **Rôle PVESDNUser :**
  - ➥ *Chemin :* `/sdn/zones/localnetwork/vmbr0`.
  - ➥ *Droits :* `SDN.Use`, `SDN.Audit`.
  - ➥ *Fonction :* Autorise la connexion des VMs au réseau.

4.4. Flux Utilisateurs

1. **Authentification :** Le login est transformé en `<username>@picamal.rt`.
2. **Création de VM :**
   - ➤ ISO ➔ VM personnalisée.
   - ➤ Template ➔ VM standardisée (récupération d'un VMID libre ➔ ajout au pool utilisateur).
3. **Démarrage de VM :**
   - ➤ Détection IP via guest-agent (boucle).
   - ➤ Création automatique d'une connexion Guacamole RDP.
4. **Arrêt / Suppression :**
   - ➤ Extinction ➔ suppression des connexions Guacamole.
   - ➤ Suppression ➔ purge + nettoyage Guacamole.

4.5. Gestion des Pools Proxmox

Chaque utilisateur reçoit un pool dédié : `<user>-picamal-rt`, garantissant une isolation totale et une gouvernance simplifiée.

5. Services de Fichiers Avancés (Bonus)

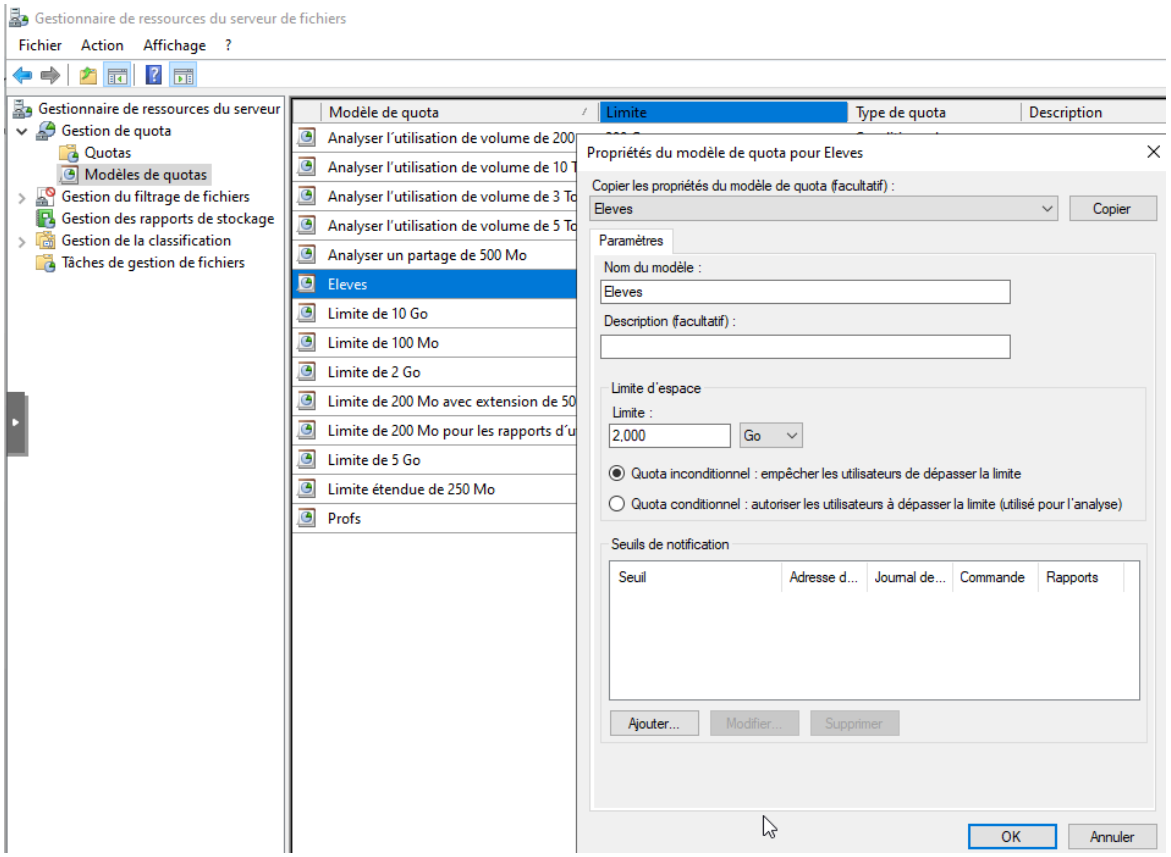
Mise en place d'un serveur de fichiers sécurisé avec gestion des quotas et isolation stricte.

5.1. Gestion des Quotas (FSRM)

Déploiement du rôle **Gestionnaire de ressources du serveur de fichiers (FSRM)** pour contrôler l'espace disque.

Procédure Technique :

1. **Installation :** Ajout du rôle FSRM via le Gestionnaire de serveur.
2. **Création du Modèle :** Dans la console FSRM, création du modèle "Quota Élèves" avec une limite stricte (**Hard Quota**) de **2 Go**.
3. **Déploiement Automatique :** Application du quota sur le dossier racine `C:\Partage_Perso` avec l'option **"Appliquer automatiquement le modèle aux sous-dossiers"**.
    - ➥ *Résultat :* Chaque nouveau dossier utilisateur créé hérite dynamiquement de sa propre limite de 2 Go.

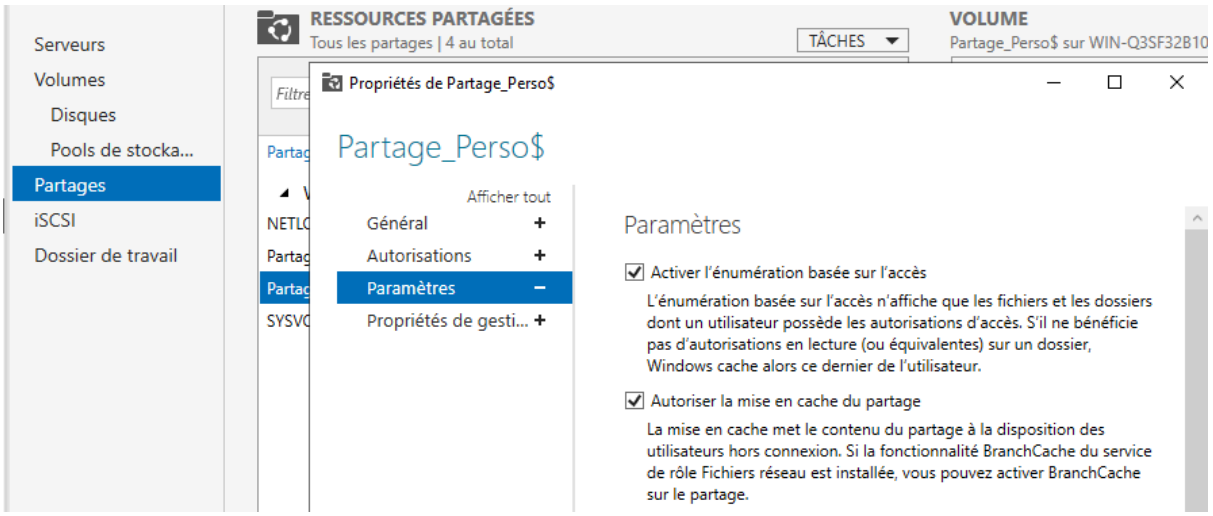


5.2. Confidentialité (Access-Based Enumeration - ABE)

Pour masquer les dossiers auxquels l'utilisateur n'a pas accès.

Procédure :

1. Dans les propriétés du partage SMB `Partage_Perso$`.
2. Activation de l'option **"Activer l'énumération basée sur l'accès"**.
3. *Résultat :* Le serveur filtre la vue explorateur en temps réel selon les ACLs NTFS. L'isolation visuelle est totale.

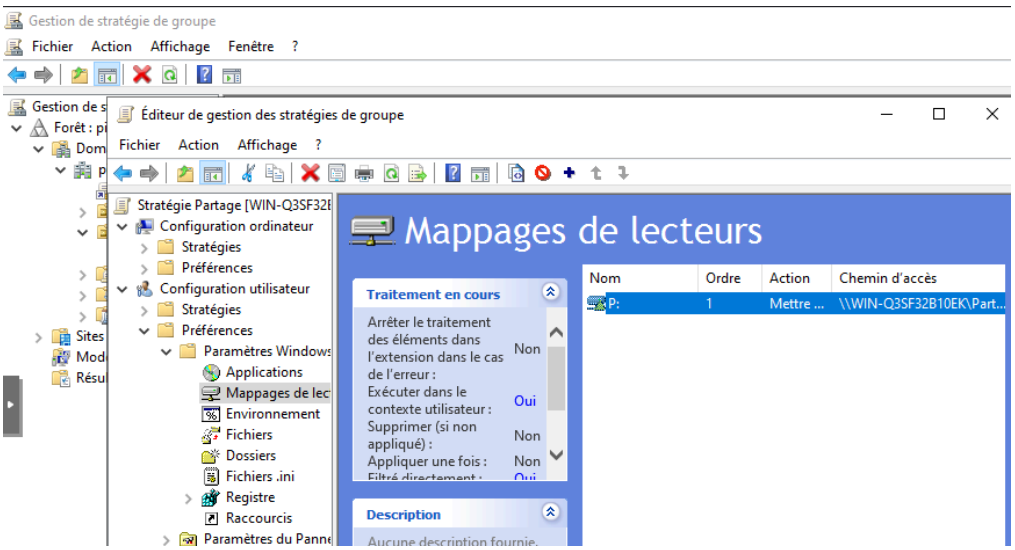


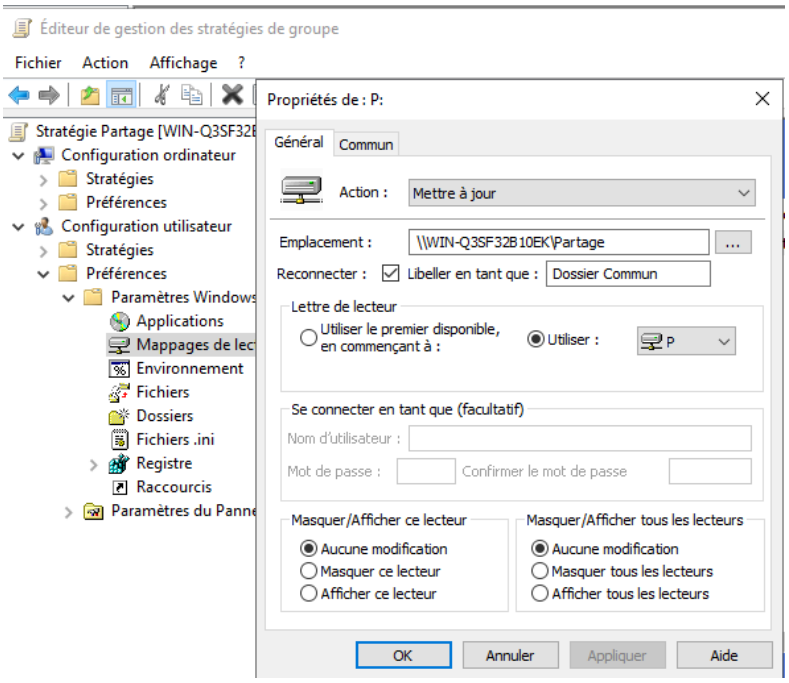
5.3. Architecture des Partages et Droits NTFS

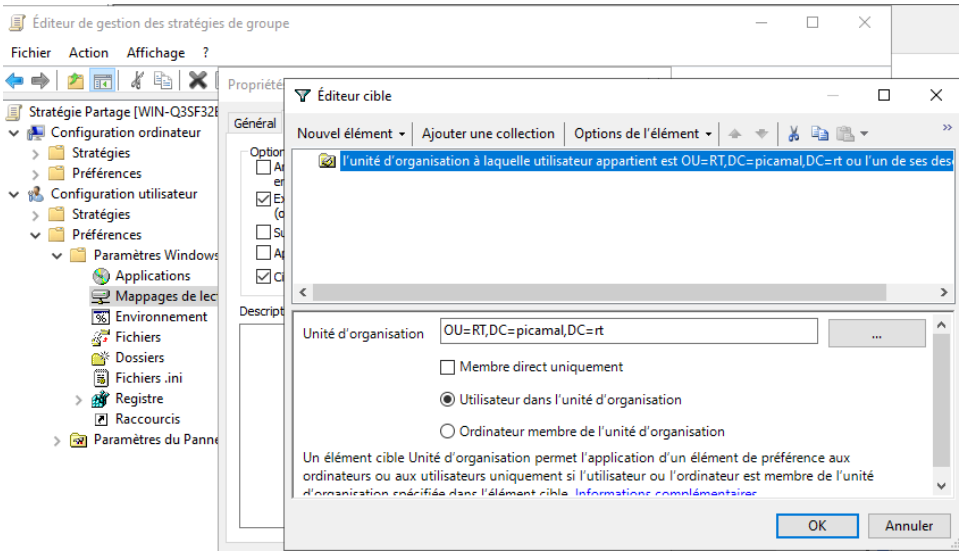
Application du principe du moindre privilège.

A. Dossier Commun (`P:`)

- **Partage :** `\\Serveur\Partage`.
- **Sécurité NTFS :**
  - `PROFS` : *Modification* (Lecture/Écriture/Suppression).
  - `ELEVES` : *Lecture et exécution* (Interdiction stricte de modification).
- **Déploiement (GPO) :** Création d'une GPO "Mappage Lecteur P" ciblée sur l'OU `RT` via *Item-Level Targeting*.

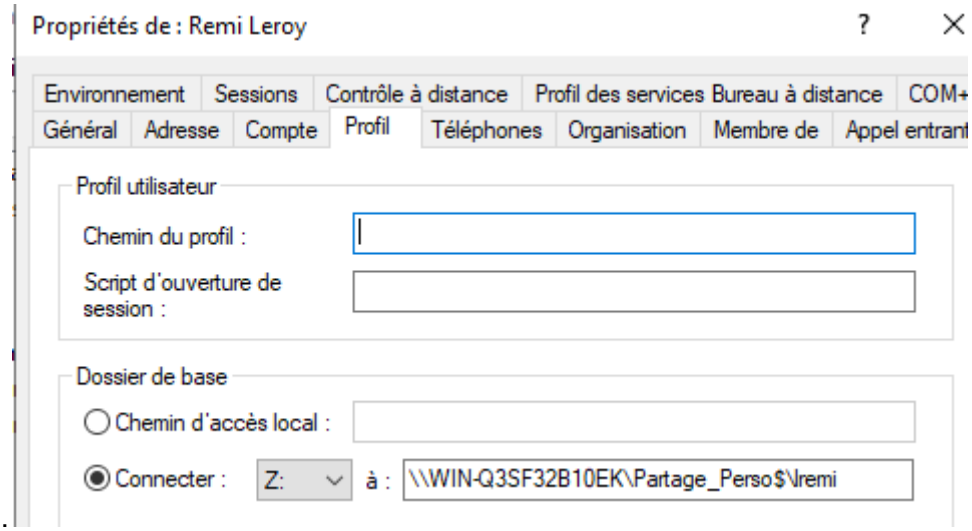






B. Dossiers Personnels (`Z:`) :

- **Partage :** `\\Serveur\Partage_Perso$` (Caché).
- **Sécurité NTFS (Isolation) :**
  - Rupture de l'héritage à la racine.
  - Racine : Droit *Création de dossier* pour "Utilisateurs du domaine" (sur "Ce dossier seulement").
  - Sous-dossiers : Droit *Contrôle total* pour le groupe spécial **CREATEUR PROPRIETAIRE**.
- **Déploiement :** Configuration automatique dans le profil AD (`HomeDirectory` connectant `Z:` à `\\Serveur\Partage_Perso$\%username%`)



6. Supervision et Métrologie (Bonus Majeur)

Mise en place d'une stack de monitoring pour la maintenance prédictive.

6.1. Architecture

- **Serveur de Collecte :** VM Debian dédiée hébergeant **Prometheus** (Base de données de séries temporelles).
- **Interface de Visualisation : Grafana** connecté à Prometheus.

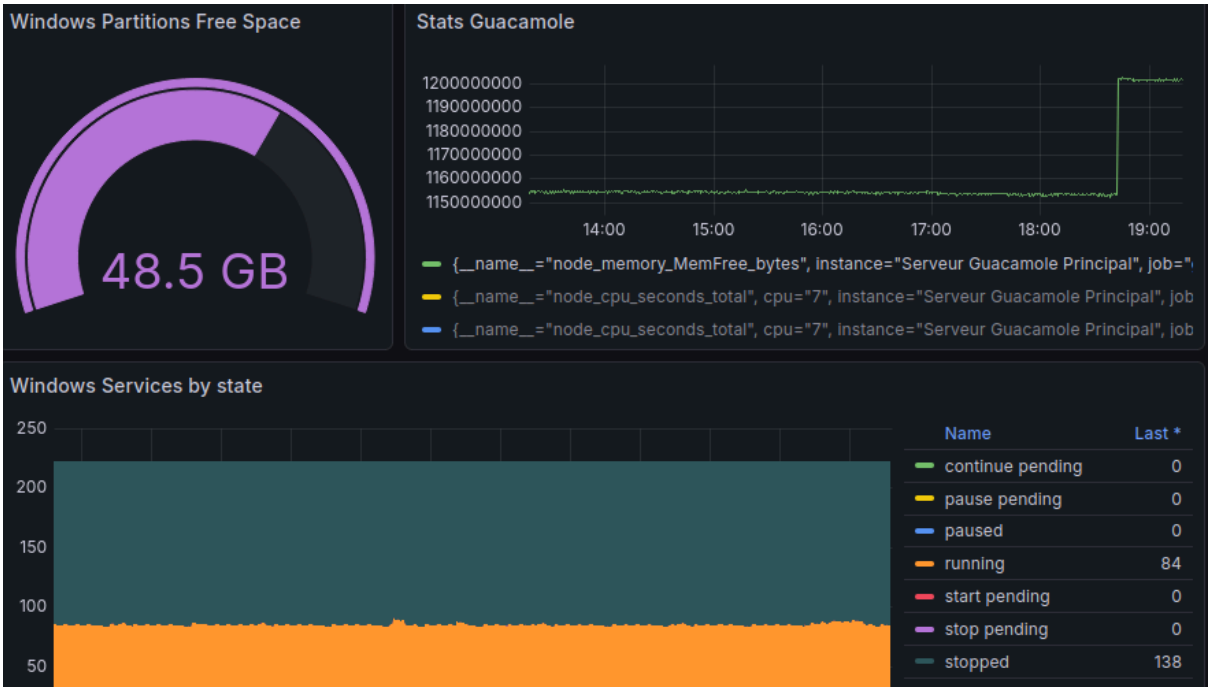
6.2. Instrumentation (Agents)

- **Node Exporter (Linux) :** Installé sur les serveurs Linux (port 9100) pour remonter les métriques système (CPU, RAM, I/O).
- **Windows Exporter (Windows) :** Installé sur le Contrôleur de Domaine (port 9182) pour les métriques WMI.

6.3. Visualisation

Création de Dashboards Grafana utilisant le langage de requête **PromQL**.

- **Exemple de métrique CPU :** `100 - (avg by (instance) (irate(node_cpu_seconds_total{mode="idle"}[1m])) * 100)`.
- **Objectif :** Détection en temps réel des goulots d'étranglement (CPU saturation, Memory leaks).



7. Analyse des Difficultés et Solutions

Cette section documente les incidents techniques rencontrés et la méthodologie de résolution appliquée.

7.1. Dysfonctionnement GPO (Mappage P:)

- **Symptôme :** Le lecteur réseau commun ne remontait pas. `gpupdate` retournait "Absence de connectivité LDAP".
- **Analyse Technique :**
  1. **Heure :** Désynchronisation de 12 minutes entre Client et Serveur. Kerberos, utilisant des tickets horodatés pour éviter les attaques par rejeu, rejetait l'authentification (tolérance max 5 min).
  2. **Réseau :** Windows classifiait le réseau en "Public" (non-confiance), activant le pare-feu sur les ports RPC/SMB nécessaires aux GPO.
  3. **Trust Relationship :** Rupture du canal sécurisé suite aux changements d'IP post-jonction.
- **Solution :** Resynchronisation NTP manuelle, passage du profil réseau en "Privé" (`Set-NetConnectionProfile`) et ré-intégration complète au domaine (Reset du Trust).

7.2. Échec d'Authentification Linux (SSSD)

- **Symptôme :** La commande `id user` retournait "utilisateur inexistant" après la jonction.
- **Diagnostic :** Les logs SSSD (`systemctl status sssd`) indiquaient une erreur `Invalid Credentials`.

- **Cause Technique :** Conflit de nom d'hôte. Le nom Linux (`test-standard`) ne correspondait pas à l'objet ordinateur créé dans l'AD (`UBUNTU-CLIENT`), rendant le fichier clé Kerberos (`keytab`) invalide.
- **Solution :**
    1. Nettoyage complet (`realm leave`, suppression des caches `/var/lib/sss/db`).
    2. Standardisation du nom d'hôte (`hostnamectl set-hostname ubuntu-client`).
    3. Nouvelle jonction propre au domaine.

8. Développement Durable / Green IT

Mutualisation des Ressources : La virtualisation permet de remplacer 30 postes physiques par un seul serveur robuste, réduisant drastiquement la consommation électrique et l'empreinte carbone de fabrication.

Prolongation de la durée de vie : L'accès via un simple navigateur (Guacamole) permet aux étudiants d'utiliser des PC portables anciens ou peu puissants (Thin Clients) pour accéder à des environnements performants.

Recyclage de serveur : Dans une logique de sobriété numérique, nous avons privilégié le réemploi d'une station de travail existante pour héberger l'hyperviseur. Ce choix permet d'éviter l'empreinte carbone liée à la fabrication d'un nouveau serveur (phase la plus polluante du cycle de vie d'un équipement informatique).

9. Summary - SAE 5.01 Technical Report: Virtual Lab Infrastructure

This document details the design, deployment, and security of a comprehensive virtualization infrastructure intended to host practical work (TP) environments for students. The core technical stack is built on the **Proxmox VE** hypervisor (Type 1).

Key components and objectives:

- **Virtualization Infrastructure:**
    - Uses **Proxmox VE** with a host bridge (`vmbr0`) for network communication.
    - Storage strategy includes `local` (ISO), `local-lvm` (VM disks) with Thin Provisioning and ExStorage used mainly for backups.
    - Performance is optimized via **Paravirtualization (VirtIO)** drivers for Windows and Linux.
    - **QEMU Guest Agent** is integrated for IP address reporting and clean shutdown commands.

    - Master images (Templates) for Windows (using **Sysprep** to generalize the SID) and Ubuntu are prepared for mass deployment.
- **Identity Infrastructure (Active Directory):**
    - Centralized identity management is based on **Windows Server 2022** (Domain: `picamal.rt`).
    - Client integration for Windows is rigorous (Static DNS, NTP synchronization, Kerberos tolerance).
    - Heterogeneous Linux integration is achieved using **SSSD**, configured for short login names and permissive GPO access.
    - User onboarding is automated via a **PowerShell script** that creates AD accounts, assigns groups (PROFS/ELEVES), and sets up the personal Home Directory with NTFS ACLs.
- **Web Portal for VM Management (Flask/Proxmox/Guacamole):**
    - The portal automates VM lifecycle management (creation, power, deletion) and strictly enforces isolation via **Proxmox ACLs**.
    - **Apache Guacamole** automatically handles RDP connections to the VMs, provisioned dynamically upon VM startup and cleaned up upon shutdown/deletion.
    - Each user gets a dedicated Proxmox pool (`<user>-picamal-rt`).
- **Advanced File Services (Bonus):**
    - **FSRM (File Server Resource Manager)** is used to enforce strict **Hard Quotas (2 GB)** on student personal folders.
    - **Access-Based Enumeration (ABE)** is enabled for confidentiality, hiding folders the user cannot access.
    - NTFS permissions adhere to the principle of least privilege, with separate configurations for the **Common Share (P:)** and **Personal Folders (Z:)**.
- **Supervision (Bonus):**
    - A monitoring stack is deployed with **Prometheus** (time-series database) and **Grafana** (visualization).
    - **Node Exporter** (Linux) and **Windows Exporter** (Windows) are used to collect system metrics (CPU, RAM, I/O).
- **Difficulties and Solutions (Feedback):**
    - **GPO Dysfunction (P: Drive Mapping):** Solved by manually resynchronizing NTP, changing the network profile to "Private", and performing a domain re-integration (Reset Trust) due to Kerberos time tolerance and Windows firewall issues.
    - **Linux Authentication Failure (SSSD):** Resolved by performing a clean re-join after standardizing the Linux hostname, which initially conflicted with the computer object in AD, invalidating the Kerberos keytab file.
- **Green IT / Sustainable Development:** The project promotes resource mutualization through virtualization (replacing 30 physical PCs with one server), prolonging hardware life via web access (Guacamole Thin Clients), and prioritizing the reuse of an existing workstation.