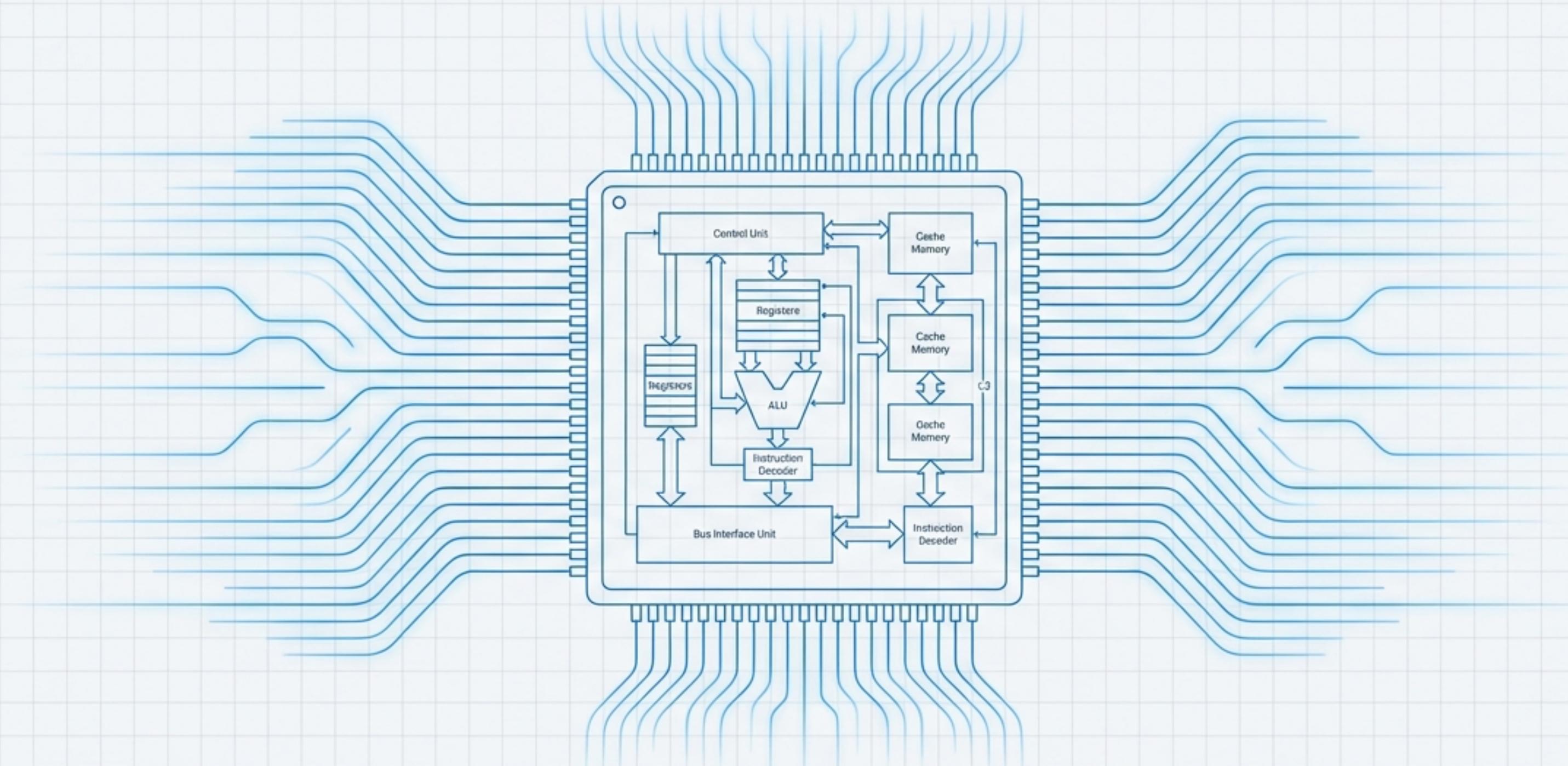


Projet d'Administration Système Multi-Plateforme

Sprint 2 : La Naissance du Cœur Opérationnel



L'Équipe du Sprint 2 : Nouveaux Rôles, Même Mission



Romain

Rôle : Scrum Master
Contributions : Script Principal & Module 1 (Utilisation Machine)



Sami

Rôle : Product Owner
Contributions : Module 2 (Gestion Utilisateurs : Création & Modification)



Georges

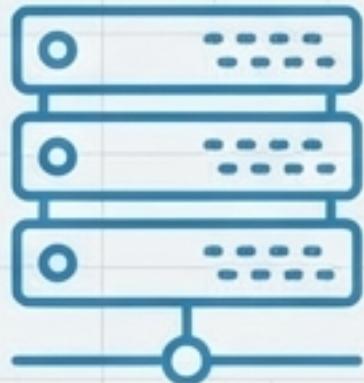
Rôle : Technicien
Contributions : Module 3 (Informations Machine & Journalisation)



Renaud

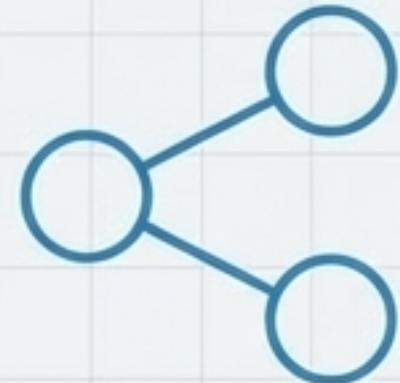
Rôle : Technicien
Contributions : Module 2 (Gestion Utilisateurs : Suppression & Infos)

Sprint 1 en Bref : Les Fondations



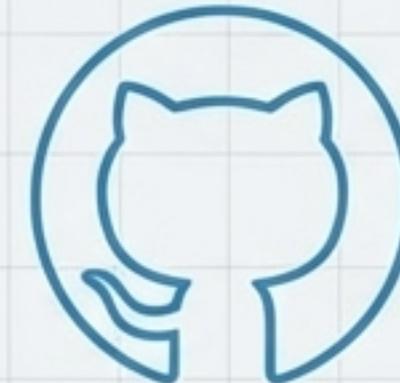
Infrastructure Déployée

4 VMs sur Proxmox opérationnelles : SRVLX01 (Debian 12), SRVWIN01 (Windows Server 2022), CLILIN01 (Ubuntu 24), CLIWIN01 (Windows 10).



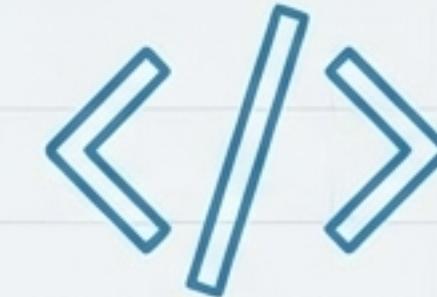
Réseau Configuré

Un sous-réseau unifié (`172.16.40.0/24`) garantissant la communication entre toutes les machines.



Dépôt Initialisé

Structure du projet sur GitHub établie avec les dossiers `/Ressources` et `/Script`, et un `README` V1.

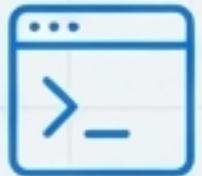


Squelette Crée

Création d'un script Bash de base servant de structure évolutive pour le projet. Il définit l'architecture du code, le menu principal et les logiques de navigation, assurant une base saine et maintenable.

La Mission du Sprint 2 : Donner Vie au Script d'Administration

Nos objectifs étaient clairs et centrés sur la création du premier prototype fonctionnel de notre outil.



Création des Scripts Bash: Développer l'ensemble des fonctionnalités logiques pour l'administration Linux.



Implémentation de la Connexion SSH: Établir et valider une connexion sécurisée et fonctionnelle vers les machines cibles depuis notre serveur de contrôle Proxmox.

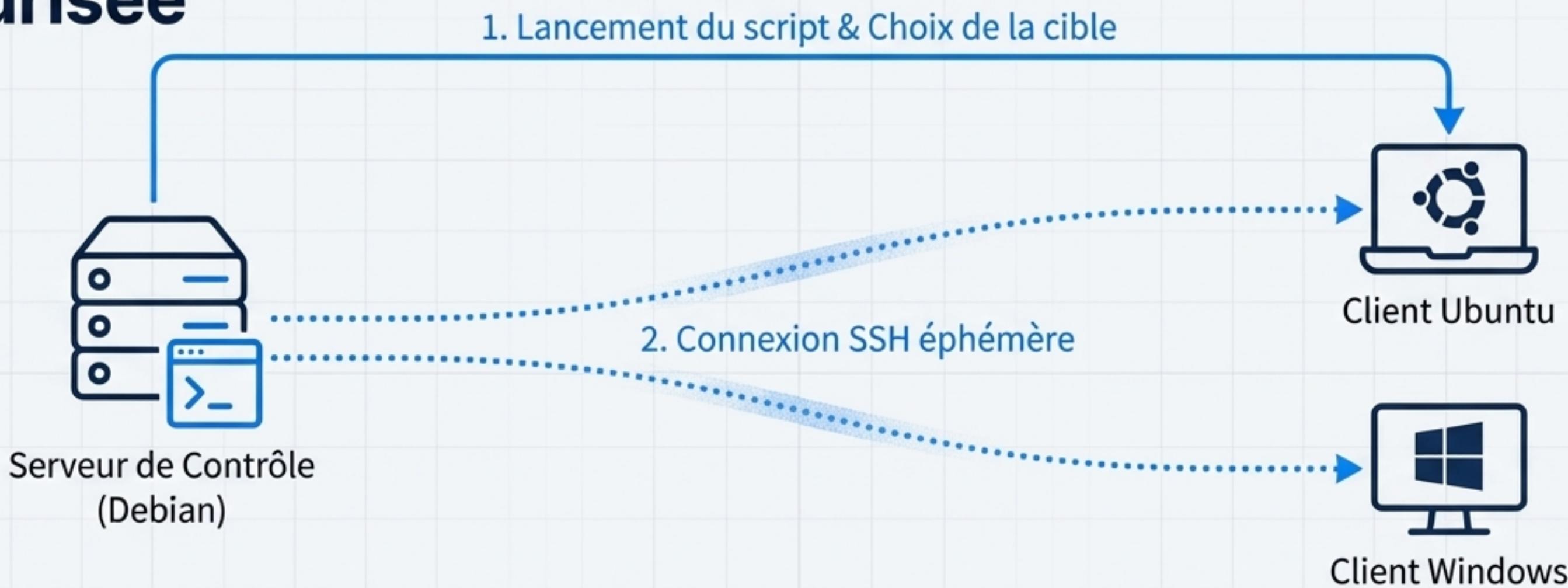


Tests en Environnement Réel: Déployer et tester l'intégralité du script sur l'infrastructure Proxmox pour identifier les points de friction.



Débogage et Validation: Itérer sur les scripts pour corriger les erreurs et valider la robustesse de la solution.

Notre Architecture : Une Gestion Centralisée et Sécurisée

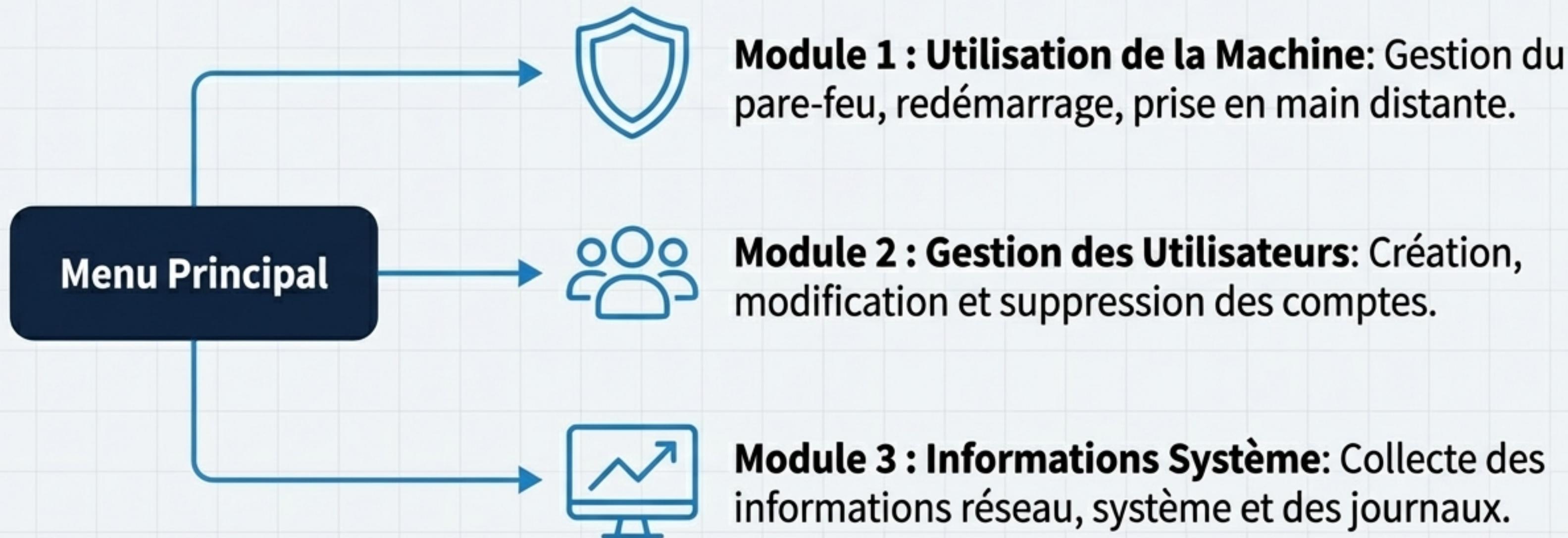


Points Clés

- Le script principal, lancé depuis le serveur Debian, orchestre toutes les actions.
- Une session SSH est établie uniquement pour la durée de l'interaction avec la machine cliente, puis est immédiatement fermée.
- Cette approche de connexions à usage unique maximise la sécurité en réduisant drastiquement la surface d'attaque et minimise l'utilisation des ressources.

Au Cœur du Code : Une Structure Modulaire

Pour un code lisible et évolutif, le script principal orchestre l'appel de sous-scripts dédiés à chaque fonctionnalité. Il est basé sur une écriture simple avec des fonctions, des boucles `while` et des conditions `case`.



Démonstration : Le Cœur Opérationnel en Action

1. Lancement depuis le Serveur de Contrôle

L'opérateur exécute le script principal sur le serveur central Debian.

```
user@SRVLX01:~$ ./script_principal.sh

ADMINISTRATION SYSTÈME LINUX

1. Gestion des Utilisateurs
2. Informations Système
3. ...
q. Quitter

Votre choix : _
```

2. Connexion & Exécution

Une connexion SSH est établie avec la cible. Le menu des options s'affiche, prêt à recevoir les commandes.

Les Défis Rencontrés : De la Théorie à la Réalité



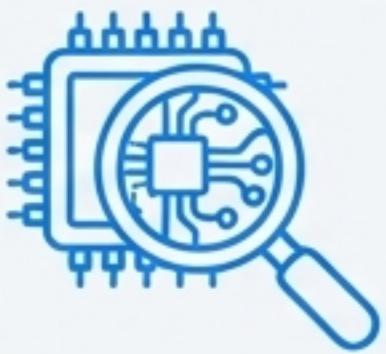
Gestion des Privilèges `sudo`

Comment éléver les droits à distance pour des actions critiques sans stocker de mot de passe en clair ?



Stabilité de la Connexion SSH

Comment gérer un `reboot` distant et s'assurer de pouvoir reprendre la main sur la machine ? Comment revenir au script précédent sans quitter la session ?



Accès aux Informations Bas Niveau

Comment lire des données système spécifiques (ex : BIOS) sans nécessiter de privilèges élevés ou d'adaptation complexe ?

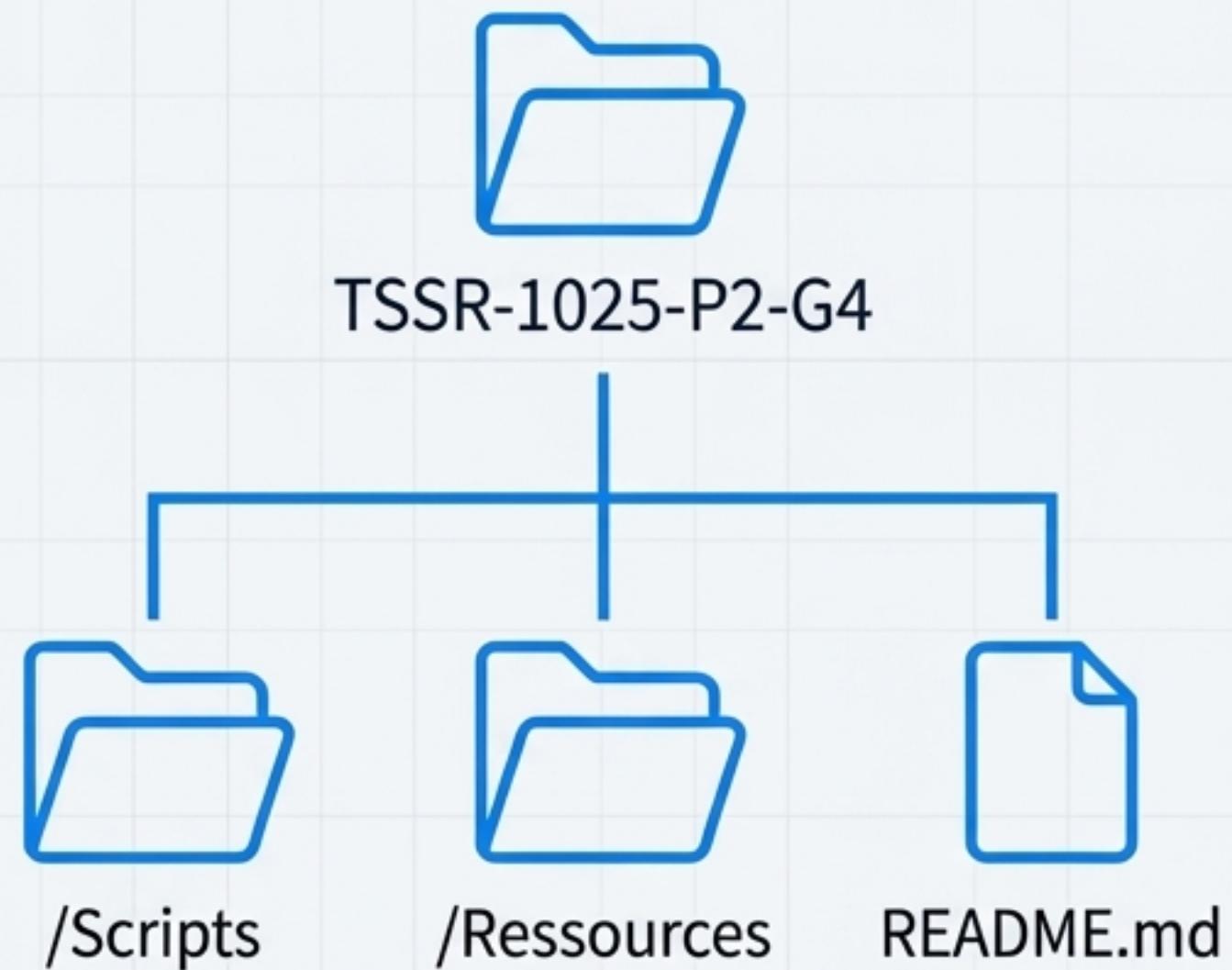


Synchronisation du Travail d'Équipe

Comment coordonner efficacement le développement de 4 personnes pour éviter les conflits de code ('problème de synchro github') ?

Notre Atelier : Un Workflow Structuré sur GitHub

Une organisation rigoureuse a été la clé pour surmonter nos défis techniques et collaboratifs.



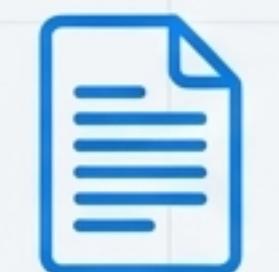
Arborescence Claire & Logique

Séparation des scripts, des ressources et de la documentation pour une navigation intuitive.



Développement Isolé et Sécurisé

Utilisation systématique de *feature-branches* pour développer chaque module sans conflit.



Documentation Centralisée

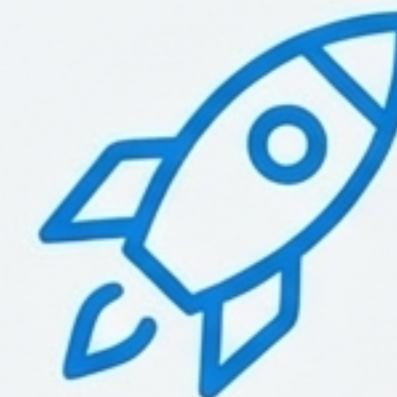
Un `README` à jour, et des guides `INSTALL` et `USER_GUIDE` en préparation pour assurer la pérennité du projet.

Mission Accomplie & Cap sur le Sprint 3



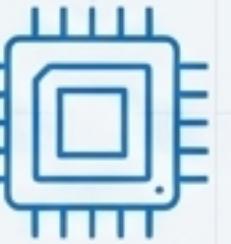
Bilan du Sprint 2

- ✓ Scripts Bash 100% fonctionnels pour les modules 1, 2 et 3. (Quelques erreurs d'esthétique et d'affichage à revoir).
- ✓ Déploiement et connexion SSH sur l'environnement Proxmox validés.
- ✓ Défis techniques surmontés et compétences de l'équipe renforcées.



Objectifs du Sprint 3

- 🚀 Finalisation et optimisation du script Bash (gestion des erreurs, affichage).
- 🚀 Initialisation du script PowerShell pour la gestion du client Windows.
- 🚀 Préparation de la documentation technique pour assurer la pérennité du projet.



Projet d'Administration Système Multi-Plateforme

Une réalisation de l'équipe :

Georges
Renaud
Romain
Sami