



Projet UF

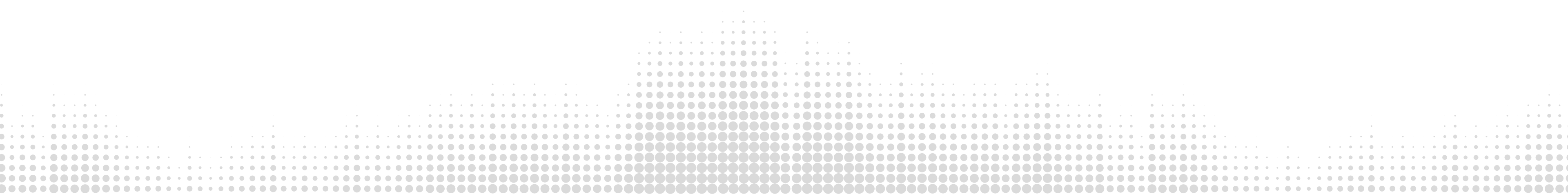
Infrastructure & Système d'information

sommaire



Ce que cette présentation couvre

- 01 présentation du projet
- 02 les étapes pour mettre en œuvre la solution
- 03 les technologies utilisées
- 04 les contraintes du projet





présentation du projet

vue globale

Ce projet consiste à assurer la sécurité des données d'une entreprise spécialisée dans le conditionnement des semi-conducteurs et les services de test pour les marchés des PC, des communications et des circuits intégrés grand public. Ils ont de nombreux clients indirects tels que IBM, Cisco et autres.

Pour arriver à cet objectif il faudrait mettre en place un système de sauvegarde qui serait placé dans les salles de serveurs de production. Ce qui permet d'assurer une plus grande sécurité de ses données, à travers leur pérennisation.

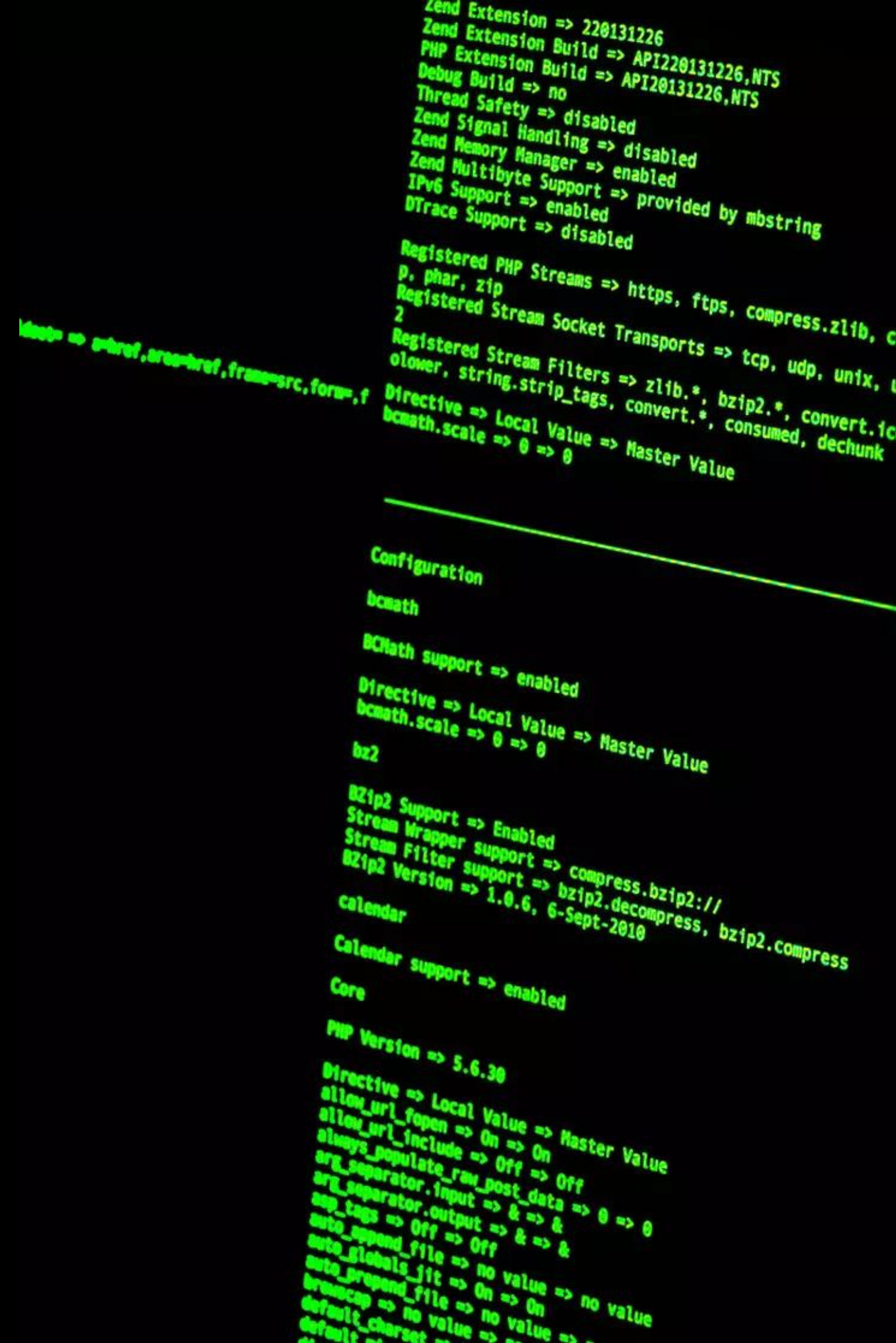
Dans ce cadre nous aurons besoin d'un serveur de stockage ou envoyer les données a partir du serveur client de l'entreprise.

Rédaction du script de sauvegarde

En informatique, un script désigne un programme (ou un bout de programme) chargé d'exécuter une action prédéfinie. Il s'agit d'une suite de commandes simples et souvent peu structurées qui permettent l'automatisation de certaines tâches successives dans un ordre donné. Un script peut donc par exemple ouvrir un répertoire et crypter des fichiers qui s'y trouvent, ou modifier à la volée la taille d'une image à l'ouverture d'une page.

Pour la réalisation de notre projet il nous faut un script pour réaliser la sauvegarde et l'envoyer au serveur de stockage.

Notre script utilisera une connexion sécurisée ssh et une copie des données en utilisons RSYNC.



```
1 #!/usr/bin/env bash
2
3 #the source path
4 src="/"
5
6 #the destination path
7 dest="mero@192.168.217.140:/home/mero/backup"
8
9
10 # RSYNC COMMAND
11 rsync -aAXv ssh --delete --exclude=/dev/* --exclude=/proc/* --exclude=/sys/* --exclude=/tmp/* --exclude=/run/* --
  exclude=/mnt/* --exclude=/media/* --exclude="swapfile" --exclude="lost+found" --exclude=".cache" --
  exclude="Downloads" --exclude=".VirtualBoxVMs" --exclude=".ecryptfs" "$src" "$dest"
12
```

Configuration CRONTAB pour l'automatisation

Crontab est un outil qui permet de lancer des applications de façon régulière, pratique pour un serveur pour y lancer des scripts de sauvegardes.

Pour être autorisé à utiliser la commande crontab, il faut que l'utilisateur soit présent dans le groupe cron.

Les fichiers /etc/cron.allow et /etc/cron.deny permettent de définir les droits d'utilisation sur crontab.



```
GNU nano 4.8 /tmp/crontab.wELiCX/crontab
0 6 * * 1,6 /home/ubuntu/Desktop/backup.sh
```

every Monday and Friday at 6am


les technologies utilisées



Secure Shell est à la fois un programme informatique et un protocole de communication sécurisé. Le protocole de connexion impose un échange de clés de chiffrement en début de connexion. Par la suite, tous les segments TCP sont authentifiés et chiffrés.



rsync remote synchronization, est un logiciel libre de synchronisation de fichiers. Il travaille de manière unidirectionnelle c'est-à-dire qu'il synchronise, copie ou actualise les données d'une source (locale ou distante) vers une destination (locale ou distante) en ne transférant que les octets des fichiers qui ont été modifiés.



les contraintes du projet

Connexion SSH automatisée avec une clé

pour que notre script soit opérationnel sans aucune intervention il faut s'authentifier au serveur pour ne pas avoir à taper son mot de passe à chaque fois .
donc il faut exporter la clé publique sur le serveur.

Contrôle de l'accès à la commande crontab

pour pouvoir utiliser crontab sans avoir à se connecter en temps que superutilisateur.
il faut ajouter l'utilisateur au fichier `/etc/cron.allow` ,
de cette manière l'utilisateur aura le droit de créer,
modifier, afficher ou supprimer des fichiers crontab