Pour Helia	→ Ajouter lcredit=-1 dans → sudo nano /etc/pam.d/common-password (pour forcer un char minuscule dans le password) → **(to check)** sleep dans le cron (pour le forcer à lancer cron au boot)

Pourquoi Debian ?

→ Plus facile à installer et à configurer, plus adapté pour les projets perso et pour les débutants.

Différence entre Debian et CentOS

→ Debian est plus simple à utiliser et à update que CentOS.
→ Debian est plus ergonomique et il est compatible avec plus de librairies.
→ Debian est plus customizable
→ Pour les entreprises ou projets pros, CentOS est plus stable et a un support technique

Qu'est ce qu'une VM ?

Ressource qui utilise un logiciel au lieu d'un ordinateur physique afin de faire tourner des programmes ou applications. Une VM a son propre operating system et ses propres fonctions. Une VM est utilisée pour tester des applications dans un environnement contrôlé et safe.
→ Une VM utilise un logiciel qui simule un disque dur virtuel sur la machine host.
(un ordinateur simule dans un ordinateur)

Que faire si le correcteur n'accepte pas une des réponses ?

-Se lever, attraper une chaise et le menacer.
-S'il persiste, le frapper jusqu'à ce qu'il ne se relève plus puis jeter le corps dans la Seine.

Difference entre aptitude et APT (Advanced Packaging Tool)

→ Aptitude = haut niveau package manager (plus intelligent, automatise l'installation)
→ APT = bas niveau qui peut être utilisé par les hauts niveaux (ne fait que ce qu'on lui dit de faire)

Qu'est ce que AppArmor ?

Système de sécurité qui permet au système admin de restreindre les actions que les processes peuvent faire, inclus par défaut dans Debian.
→sudo aa-status pour voir si ca marche

Règles du Mot de Passe

→ sudo nano /etc/login.defs (Password aging controls)
→ sudo nano /etc/pam.d/common-password

retry = nombre de retries
minlen = longueur minimal du psswd
maxrepeat = nombre maximum de repetition de char a la suite
lcredit = nombre maximum de majuscule (-1 pour en forcer au moins 1)
ucredit = nombre maximum de minuscule (-1 pour en forcer au moins 1)
dcredit = nombre maximum de chiffre (-1 pour en forcer au moins 1)
difok = nombre minimum de char differents de l'ancien psswd
usercheck = verifier si le username est dans le psswd (0 pour interdit)
reject_username = interdiction d'avoir le username dans le psswd (double protection)
enforce_for_root = obligation de faire un psswd selon les regles

UFW = Uncomplicated Firewall, interface qui modifie les firewall sans compromettre la sécurité. Permet d'ouvrir et de fermer les ports tout en protégeant SSH.

SSH = Secure Shell, mécanisme d'authentification entre les clients et le host afin de permettre une communication cryptée.

Cron = ligne de commande qui se produit à intervalles réguliers ou à un moment spécifique de la journée. (Exemple : reboot un serveur tous les jours à x heure).
→ cd /usr/local/bin - pour montrer notre script monitoring.sh
→ sudo crontab -u root -e - pour éditer cronjob
→changer la dernière ligne de cron tab en */1 * * * * sleep 30s && script path pour lancer cron toutes les 30s
→ supprimer la dernière ligne pour arrêter le cronjob

Pour créer un nouvel utilisateur etc:

-sudo adduser username
-sudo adduser username sudo
-sudo adduser username user42

Pour check tout les machins chelous la:

-lsblk (check les partitions)
-sudo aa-status (check AppArmor)
-getent group sudo (check les utilisateurs sudo)
-getent group user42 (check les utilisateurs user42)
-sudo service ssh status (check le statut de ssh)
-ssh username@localhost -p 4242 (se connecter a sa vm depuis le terminal)
-vim /etc/sudoers.d/<nomdufichier> (check le fichier de configuration de sudo)
-vim /etc/login.defs (check les règles de mot de passe)
-vim /etc/pam.d/common-password check la police de mot de passe)
-sudo crontab -l (affiche ton fichier crontab)

Commandes utiles:

pour changer hostname:

-sudo vim /etc/hostname

pour trouver les logs de sudo:
-cd /var/log/sudo/ ensuite ca depend dans quel fichiers vous avez rangé vos logs

pour ajouter et supprimer le port 8080 dans UFW:
-sudo ufw allow 8080 (autorise)
-sudo ufw status (check)
-sudo ufw deny 8080 (interdit)

Monitoring.sh

```
#!/bin/bash
architecture=$(uname -a) // imprime les infos du systeme -a = all
physicap_p=$(grep "physical id" /proc/cpuinfo | sort | uniq | wc -l) // cherche dans cpuinfo | trie par ordre alphabétiquement | supprime les doubles | compte le nombre de lignes
virtual_p=$(grep "^processor" /proc/cpuinfo | wc -l) // cherche dans cpuinfo | prend les resultats commencant avec processor
free_ram=$(free -m | awk '$1 == "Mem:" {print $2}') // affiche la RAM | awk éditeur de texte 2eme colonne
used_ram=$(free -m | awk '$1 == "Mem:" {print $3}')
percent_ram=$(free | awk '$1 == "Mem:" {printf("%.2f"), $3/$2*100}') // 3eme colonne / 2eme colonne * 100
total_disk=$(df -BG | grep '^/dev/' | grep -v '/boot$' | awk '{ft += $2} END {print ft}') // imprime les files systems sur le système en taille G (gigabytes) | prends les lignes commencant par /dev | exclut /boott$ | ft += → additionne toutes les infos de la 2eme colonne
used_disk=$(df -BM | grep '^/dev/' | grep -v '/boot$' | awk '{ut += $3} END {print ut}') // imprime les files systèmes en taille M (Megabytes)
percent_disk=$(df -BM | grep '^/dev/' | grep -v '/boot$' | awk '{ut += $3} {ft+= $2} END {printf("%d"), ut/ft*100}') // pourcentage entre ut et ft
cpul=$(top -bn1 | grep '^%Cpu' | cut -c 9- | xargs | awk '{printf("%.1f%%"), $1 + $3}') //usage du CPU | filtre en le 9eme et le dernier char | xargs = concatene en une seule ligne | pourcentage
lb=$(who -b | awk '$1 == "system" {print $3 " " $4}') // date et jour quand le systeme a ete boot
lvmu=$(if [ $(lsblk | grep "lvm" | wc -l) -eq 0 ]; then echo no; else echo yes; fi) // variable assigne a yes si le systeme a des LVM, sinon no. | lsblk → liste des appareils sur le systeme | eq 0 → si le resultat de la commande est egal a 0, alors echo no
ctcp=$(ss -neopt state established | wc -l) // montre les nombres des connections TCP deja etablis
ulog=$(users | wc -w) // users | word count
ip=$(hostname -I) // montre le nom du host name -I → les adresses IP
mac=$(ip link show | grep "ether" | awk '{print $2}') // liste les interfaces network
cmds=$(journalctl _COMM=sudo | grep COMMAND | wc -l) // liste toutes les commandes sudo du systemes
wall "   #Architecture: $architecture
         #CPU physical: $physical_p
```

```
#vCPU: $virtual_p
#Memory Usage: $used_ram/${free_ram}MB ($percent_ram%)
#Disk Usage: $udisk/${fdisk}Gb ($pdisk%)
#CPU load: $cpul
#Last boot: $lb
#LVM use: $lvmu
#Connections TCP: $ctcp ESTABLISHED
#User log: $ulog
#Network: IP $ip ($mac)
#Sudo: $cmds cmd"
```