|  |  |
| --- | --- |
|  | Évaluation finale pratique - 2 |

|  |  |
| --- | --- |
|  |  |
| Automne 2022 | **Évaluation Finale Pratique – Partie 2 de 2** |
|  | Installation et configuration d’un serveur Linux. |

**NOTE : 75/100**

|  |  |  |
| --- | --- | --- |
|  |  | Table des matiÈres |

[Table des matiÈres 1](#_Toc121737095)

[Consignes générales 2](#_Toc121737096)

[Répertoires et fichiers 3](#_Toc121737097)

[Utilisateurs et groupes 4](#_Toc121737098)

[Permissions 5](#_Toc121737099)

[admin1 5](#_Toc121737100)

[Les développeurs (membres du groupe prog) 5](#_Toc121737101)

[Les techniciens (membres du groupe admin) 6](#_Toc121737102)

[Les clients 7](#_Toc121737103)

[script rapportssysteme.sh 8](#_Toc121737104)

[Questions 9](#_Toc121737105)

[Site web 10](#_Toc121737106)

|  |  |  |
| --- | --- | --- |
|  |  | Consignes générales |

L’évaluation finale pratique se déroule en classe et est d’une durée déterminée.

Elle se déroule en 2 parties :

**Partie 1** : installation d’un serveur Linux sur une machine virtuelle, configuration du réseau et du pare-feu et quelques installations logicielles.

**Partie 2** : création et configurations de répertoires et fichiers, utilisateurs et groupes, permissions, scripts etc.

**Ce document présente la partie 2**, d’une durée maximale de 5 périodes.

Vous travaillez sur le serveur Linux créé à la partie 1. Travaillez avec une connexion par PuTTY pour profiter des fonctionnalités de copier-coller.

Vous devez remettre ce document sur Léa dans la section travaux, avec les sections demandées remplies.

Le travail de cette étape consiste à réaliser les actions ci-dessous, en suivant des consignes précises :

* La majorité des configurations sont faites à partir de scripts, ce sont ces scripts qui sont corrigés
* Assumez que TOUS LES SCRIPTS SERONT EXÉCUTÉS AVEC PRIVILÈGES ADMINISTRATEUR (sudo)
* Tous les scripts doivent fonctionner indépendamment du répertoire courant

Le travail consiste à :

* Créer une structure de répertoires et de fichiers
* Créer des utilisateurs et des groupes
* Assigner des permissions
* Créer des scripts
* Créer un site web Nginx
* Répondre aux questions dans ce document

|  |  |  |
| --- | --- | --- |
|  |  | Répertoires et fichiers |

**10/10**

Écrivez un script (documents.sh) qui permet la création de la structure de répertoires et fichiers présentée plus bas et qui respecte les consignes :

* Le script doit d’abord vérifier s’il y a déjà un répertoire nommé Documents à la racine.
* Si un répertoire nommé Documents existe déjà, il doit être supprimé ainsi que tous ses sous-répertoires et fichiers.
* Après la suppression ou s’il n’y a pas déjà de répertoire Documents, la structure de répertoires et fichiers doit être créée.
* Pour les fichiers, créer simplement des fichiers vides.
* Copiez le contenu du script dans l’espace prévu à cet effet.

Structure de répertoires et fichiers à créer (les documents en surbrillance sont des répertoires, les documents avec une extension sont des fichiers) :

|  |
| --- |
| /Documents  ├── Clients  ├── Developpeurs  | ├── Images  | | └── logo.png  | └── Source  | └── index.html  └── Techniciens  ├── Rapports  | └── specifications.txt  └── Scripts |

Exécutez le script documents.sh, puis copiez le contenu du script documents.sh ci-dessous.

|  |
| --- |
| #!/bin/bash  if test -d /Documents  then  rm -r -f /Documents  fi  mkdir /Documents  mkdir /Documents/Clients  mkdir /Documents/Developpeurs  mkdir /Documents/Developpeurs/Images  mkdir /Documents/Developpeurs/Source  touch /Documents/Developpeurs/Images/logo.png  touch /Documents/Developpeurs/Source/index.html  mkdir /Documents/Techniciens  mkdir /Documents/Techniciens/Rapports  mkdir /Documents/Techniciens/Scripts  touch /Documents/Techniciens/Rapports/specifications.txt |

|  |  |  |
| --- | --- | --- |
|  |  | Utilisateurs et groupes |

**18/20**

Écrivez un script (utilisateurs.sh) qui permet la création de groupes et d’utilisateurs avec leurs configurations et qui respecte les consignes :

* Créer un groupe **prog**
* Créer un groupe **admin**
* Créer un groupe **systeme**
* Créer deux utilisateurs **prog1** et **prog2** membres du groupe secondaire **prog**
* Créer trois utilisateurs **admin1** **admin2** et **admin3** membres du groupe secondaire **admin** et dont le groupe principal est **systeme**
* Créer 20 utilisateurs nommés **client1**, **client2**, **client3** jusqu’à **client20**. La création de ces utilisateurs DOIT utiliser une **boucle for**.
* Tous les utilisateurs doivent utiliser bash comme shell.
* Les développeurs (prog1 et prog2) et les techniciens (admin1admin2 et admin3) doivent avoir un répertoire personnel
* Les clients ne doivent pas avoir de répertoire personnel.
* Tous les utilisateurs doivent avoir **DDDddd444** comme mot de passe

Exécutez le script utilisateurs.sh, puis copiez le contenu du script utilisateurs.sh ci-dessous.

|  |
| --- |
| !/bin/bash  groupadd prog  groupadd admin  groupadd systeme  useradd -s /bin/bash -m -G prog prog1  useradd -s /bin/bash -m -G prog prog2  useradd -s /bin/bash -m -G admin -g systeme admin1  useradd -s /bin/bash -m -G admin -g systeme admin2  useradd -s /bin/bash -m -G admin -g systeme admin3  nbclient="1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20"  for nbclient in $nbclient  do  useradd -s /bin/bash client$nbclient  su -c "echo -e 'DDDddd444\nDDDddd444' | passwd prog$nbclient"  su -c "echo -e 'DDDddd444\nDDDddd444' | passwd admin$nbclient"  su -c "echo -e 'DDDddd444\nDDDddd444' | passwd client$nbclient"  done |

Les mots de passe de prog et admin ne vont pas dans la boucle for pour les clients

|  |  |  |
| --- | --- | --- |
|  |  | Permissions |

**26/30**

Vous devez concevoir et écrire les commandes pour réaliser une sécurité de fichiers en suivant les consignes données ci-dessous.

Les commandes seront exécutées à partir d’un script (permissions.sh) que vous devez copier dans l’espace prévu à cet effet plus bas.

Vous pouvez assigner les **utilisateurs propriétaires** et **groupes propriétaires** des répertoires et fichiers en fonction de votre conception.

Vous assignez ensuite les **permissions** (autorisations) Linux de base et/ou avancées (étendues).

admin1

**admin1** est administrateur du système. Il doit avoir **toutes les autorisations** sur le répertoire **Documents** et tous ses fichiers et sous-répertoires.

Les développeurs (membres du groupe prog)

Le membres du **groupe prog** doivent pouvoir accéder au répertoire **Documents**(voir son contenu), mais pas y ajouter supprimer ou renommer des fichiers ou répertoires. C’est simplement pour pouvoir accéder au dossier **Developpeurs**.

Les membres du **groupe prog** doivent avoir **plein accès**(toutes les permissions) au répertoire **Developpeurs** et tous les sous-répertoires et fichiers dans ce répertoire. Cela inclus lire et modifier le contenu de fichiers, ainsi que créer supprimer et renommer de répertoires.

Les membres du **groupe prog** ne doivent avoir **AUCUN accès** au répertoire **Techniciens**.

Pour les membres du **groupe prog**, **aucune sécurité supplémentaire** n’est requise pour les **fichiers** à l’intérieur des répertoires mentionnés ci-dessus.

Les techniciens (membres du groupe admin)

Le membres du **groupe admin** doivent pouvoir accéder au répertoire **Documents**(voir son contenu), mais pas y ajouter supprimer ou renommer des fichiers ou répertoires (sauf admin1 voir plus bas). C’est simplement pour pouvoir accéder au dossier **Techniciens**.

Les membres du **groupe admin** doivent avoir **plein accès**(toutes les permissions) au répertoire **Techniciens** et tous les sous-répertoires et fichiers dans ce répertoire. Cela inclus lire et modifier le contenu de fichiers, ainsi que créer supprimer et renommer de répertoires.

Les membres du **groupe admin** doivent avoir **plein accès**(toutes les permissions) au répertoire **Clients** et tous les sous-répertoires et fichiers dans ce répertoire. Cela inclus lire et modifier le contenu de fichiers, ainsi que créer supprimer et renommer de répertoires.

Les membres du **groupe admin** (sauf admin1) ne doivent avoir **AUCUN accès** au répertoire **Developpeurs**.

Pour les membres du **groupe admin**, **aucune sécurité supplémentaire** n’est requise pour les **fichiers** à l’intérieur des répertoires mentionnés ci-dessus.

Les clients

Les clients doivent pouvoir consulter le contenu du répertoire **Clients**. Ils ne peuvent pas ajouter, supprimer ou modifier des répertoires ou fichiers dans le répertoire **Clients**. On accepte qu’ils aient droit de lecture dans le répertoire **Documents** pour accéder au répertoire **Clients**.

Ils ne doivent avoir **aucun accès** aux répertoires **Techniciens** et **Developpeurs**.

NOTE : il n’y aura pas d’autre type d’utilisateur sur le système que ceux mentionnés, vous pouvez donc utiliser ***tous les autres*** utilisateurs pour compléter la configuration de permissions.

Exécutez le script permissions.sh, puis copiez le contenu du script permissions.sh ci-dessous.

|  |
| --- |
| #!/bin/bash  #admin1  chown admin1 -R /Documents  chmod u=rwx -R /Documents  chmod o=rxt /Documents  #admin  chgrp admin /Documents  chmod g=rx /Documents  chmod g=rwx -R /Documents/Techniciens  chgrp admin -R /Documents/Techniciens  chmod g=rwx -R /Documents/Clients  chgrp admin -R /Documents/Clients  chmod o= /Documents/Developpeurs  #prog  chgrp prog -R /Documents/Developpeurs  chmod g=rwx -R /Documents/Developpeurs  chmod o= /Documents/Techniciens |

Pour admin =rx, pas de t

Manque quelques -R pour certains chgrp

|  |  |  |
| --- | --- | --- |
|  |  | script rapportssysteme.sh |

**18/20**

Écrivez un script (rapportssysteme.sh) qui permet de recueillir des informations à propos du système, à partir d’un menu proposé à l’utilisateur. Toutes les informations seront écrites dans des fichiers et non affichées à l’écran. Le script doit être copié dans l’espace prévu à cet effet plus bas et répondre aux directives ci-dessous :

* Afficher un menu avec une boucle **while**
* Choix du menu :
  + D-Taille du disque dur
  + P-Liste des pilotes
  + S-Taille des secteurs
  + Q-Quitter
* Les options doivent réaliser :
  + Lorsque l’utilisateur appuie sur **d** ou **D** : afficher la taille totale du disque dur (redirigé dans /Documents/Techniciens/Rapports/tailledisque.txt)
  + Lorsque l’utilisateur appuie sur **p** ou **P** : Afficher la liste des pilotes chargés par le système (redirigé dans /Documents/Techniciens/Rapports/pilotes.txt)
  + Lorsque l’utilisateur appuie sur **s** ou **S** : Afficher la taille des secteurs (physical sectors) sur le disque dur (redirigé dans /Documents/Techniciens/Rapports/secteurs.txt)
* Le menu doit respecter la séquence suivante :
  + Vider l’écran
  + Afficher le menu
  + Attendre que l’utilisateur fasse une sélection
  + Lorsqu’une option est sélectionnée, effectuer l’action, puis afficher un message indiquant à l’utilisateur que le rapport a été produit, et finalement attendre qu’il appuie sur une touche
  + Lorsque l’utilisateur appuie sur une touche après une des actions du menu, vider l’écran et réafficher le menu
* Quitter le script lorsque l’utilisateur appuie sur **q** ou **Q**

Copiez le contenu du script rapportssysteme.sh ci-dessous.

|  |
| --- |
| #!/bin/bash  touch /Documents/Techniciens/Rapports/tailledisque.txt  touch /Documents/Techniciens/Rapports/pilotes.txt  touch /Documents/Techniciens/Rapports/secteurs.txt  ftext() {  echo "D-Taille du disque dur"  echo "P-Liste des pilotes"  echo "S-Taille des secteurs"  echo "Q-Quitter"  }  clear  i=0  while test $i -eq 0  do  ftext  read enter  clear  case $enter in  q|Q)  ((i++))  ;;  d|D)  df -a > /Documents/Techniciens/Rapports/tailledisque.txt  ;;  s|S)  fdisk -l > /Documents/Techniciens/Rapports/secteurs.txt  ;;  p|P)  lsmod > /Documents/Techniciens/Rapports/pilotes.txt  ;;  esac  echo ""    done |

**Manque les read et clear après chaque option**

|  |  |  |
| --- | --- | --- |
|  |  | Questions |

**1/5**

En présumant une installation par défaut, si l’utilisateur **admin2** a une session ouverte et qu’il créé un nouveau fichier dans le répertoire **Rapports** quels seront :

* L’utilisateur propriétaire ?
* Le groupe propriétaire ?
* Les permissions sur le fichier (donner la liste telle qu’affichée par ls -l) ?

|  |
| --- |
| Admin2 est l’utilisateur principal et admin le groupe. Seul l’utilisateur principale peut modifier lire et executer ce ficher les autres peuvent que le lire |

Groupe = système

Permissions : -rw-rw-r--

**0/5**

On ajoute la permission suivante : **SGID** sur le répertoire **Images**.

Ensuite, si l’utilisateur **prog1** créer un fichier dans **Images**, quels seront :

* L’utilisateur propriétaire
* Le groupe propriétaire
* Les permissions sur le fichier (donner la liste telle qu’affichée par ls -l) ?

|  |
| --- |
| Le groupe et l’utilisateur principale peuvent tout faire et les autres peuvent que lire et executer |

prog1

prog

-rw-rw-r--

|  |  |  |
| --- | --- | --- |
|  |  | Site web |

**2/10**

Configurez un site web qui Nginx qui répond aux consignes suivantes :

* Répertoire du site : **/var/www/examfinal**
* Nom du fichier html par défaut : **index.html**
* Accès au site : par port, sur le **port 8888**

Faites toutes les configurations, répertoires, fichiers, configurations Nginx ET les configurations requises pour le pare-feu lorsqu’il est démarré, puis tester un accès à partir d’un navigateur pour en vérifier l’exécution. (le contenu de la page web n’a pas d’importance, faites simplement une page html simple pour pouvoir tester).

Entrez les informations désignées dans les espaces ci-dessous.

Copiez-ci-dessous le résultat de la commande **sudo tree /var/www** :

|  |
| --- |
| html/Index.html |

Copiez ci-dessous le résultat de la commande **sudo tree /etc/nginx** :

|  |
| --- |
|  |

Copiez ci-dessous le contenu du fichier de configuration Nginx du site web :

|  |
| --- |
| server {  listen 8888;  index index.html;  root /var/www/ examfinal /html;  } |

Copiez ci-dessous les commandes qui permettront au pare-feu d’autoriser l’accès au site web par le réseau :

|  |
| --- |
| Sudo ufw allow 8888 |

Remettre ce document rempli sur Léa.