

## Examen

L'examen est sur 20 points et comprend 4 pages. Les exercices sont indépendants. Aucun document n'est autorisé. Les réponses doivent être soigneusement justifiées. L'utilisation de téléphones portables ou calculatrices est interdite. Le barème est donné uniquement à titre indicatif. Un aide mémoire HTML et PHP est donné en page 4.

**Les instructions suivantes doivent être respectées scrupuleusement, sous peine de non-correction :**

- Reporter sur **toutes** les copies le numéro d'anonymat imprimé sur la copie principale. Cette copie **doit** être rendue, même blanche.
- Chaque copie doit être numérotée et mentionner le nombre total de copies rendues (1/3, 2/3, ...). Si une seule copie est rendue, la numéroté 1/1. Les copies non numérotées **ne seront pas corrigées**.
- Toutes les réponses doivent être rédigées **au stylo**, sur des copies. Les parties rédigées au crayon de papier ou sur les feuilles de brouillons **ne seront pas corrigées**.
- Les copies doivent être anonymées, mais uniquement après vérification d'identité par un surveillant et signature de la liste d'émargement.

### 1 Questions de cours (3 points)

1. (1 point) Quel est le rôle d'un routeur ?

**Réponse:** Un routeur retransmet les paquets IP (datagrammes) qu'il reçoit vers leur destination finale.

2. (1 point) À quoi sert la commande `ls` ?

**Réponse:** La commande `ls` sert à lister des fichiers et répertoires.

3. (1 point) HTTP est un protocole réseau *texte* où les données transitent sans chiffrement (encryption). Citer un problème de sécurité que cela engendre (citer brièvement un problème parmi ceux qui existent).

**Réponse:** Problème de confidentialité : un utilisateur avec les privilèges suffisants, sur l'une des machines entre le client et le serveur peut lire le contenu des messages envoyés entre le client et le serveur. Problème d'authenticité : un utilisateur avec les privilèges suffisants sur une machine entre le client et le serveur peut se faire passer pour le client ou pour le serveur.

### Problème (17 points)

On souhaite traiter différents aspects du *déploiement* (installation) d'un site Web. Le problème aborde donc dans ce cadre unique des questions de Système, Réseau et Programmation HTML/PHP. Vous pouvez cependant faire les exercices de manière indépendante et dans l'ordre que vous souhaitez, en les numérotant clairement en début d'exercice.

- On considère une machine Unix faisant office de serveur Web. La machine a les caractéristiques suivantes :
- un utilisateur `lambda`, appartenant au groupe `webdev` et dont le répertoire utilisateur est `/home/lambda`. Les permissions dans `/home/lambda` (y compris pour le répertoire lui-même) sont :
    - pour les fichiers** : droits en lecture et écriture uniquement pour `lambda`, aucun droit pour son groupe, aucun droit pour les autres ;
    - pour les répertoires** : droits en exécution, lecture et écriture pour `lambda`, aucun droit pour son groupe, aucun droit pour les autres.
  - un utilisateur `www`, appartenant au groupe `webdev` et dont le répertoire utilisateur est `/var/www`. Le répertoire `/var/www` est *vide*. **Initialement**, les permissions pour ce dernier sont : droits en lecture, écriture et exécution pour l'utilisateur `www`, aucune permission pour son groupe, aucune permission pour les autres.
  - comme dans tous les systèmes UNIX, un utilisateur `root` (l'administrateur) qui peut modifier les droits de tous les fichiers et répertoires, quel que soit leur propriétaire.
  - la machine possède une seule interface réseau, dont l'adresse IP est 129.175.28.179.
  - le nom de domaine associé à cette machine est `www.monsite.org`.
  - le répertoire racine du serveur web est `/var/www/` et le serveur web est un processus lancé par l'utilisateur `www`.

## 2 Chemins et permissions Unix, expressions régulières (4 points)

1. (1.5 point) On souhaite que l'utilisateur `lambda` puisse déposer librement dans `/var/www` des fichiers et répertoires. Quelles doivent être les permissions en octal du répertoire `/var/www` pour que cela soit possible ? (on demande uniquement les permissions pas la commande pour les modifier). Quel(s) utilisateur(s) peut (peuvent) modifier les permissions de ce répertoire ?

**Réponse:** Les permissions doivent être 770 au moins. Les utilisateurs pouvant effectuer ce changement sont `www` et `root`.

2. (2.5 points) Le répertoire utilisateur de `lambda` contient un répertoire `site` dont le contenu est partiellement listé ci-après :

```
img répertoire contenant les fichiers :
    deuxpoints.jpg  img004.jpg  img009.jpg
    img000.jpg      img005.jpg  logo.jp
    img001.jpg      img006.jpg  slash.jp
    img002.jpg      img007.jpg  vide.jpg
    img003.jpg      img008.jpg
```

```
index.html
horloge.php
fuseaux_horaires.txt
...
```

Dans la liste ci-dessus, le « ... » signifie qu'il y a d'autres fichiers, dont on ne donne pas les noms. Il n'y a pas d'autre répertoire que « `img` ». Donner la ligne de commande Unix permettant de réaliser chacune des actions suivantes (et respectant, le cas échéant les contraintes supplémentaires demandées). On suppose que l'utilisateur `lambda` a tous les droits nécessaires pour effectuer ces commandes et que le répertoire courant est son répertoire utilisateur. Vous pouvez utiliser des expressions régulières simples ou *glob* lorsque c'est nécessaire.

- Création du répertoire `img` dans le répertoire de l'utilisateur `www` ;
- Copie de tous les fichiers « `.jpg` » du répertoire `img` vers le répertoire `/var/www/img` (utiliser une **unique** expression régulière qui sélectionne tout ces fichiers) ;
- Copier dans `/var/www/` tous les fichiers du répertoire `site` dont l'extension est `.html` et dont le nom, hors extension, contient au moins un « `x` » ou un chiffre ;

- Copier dans `/var/www/` tous les fichiers textes du répertoire `site` (extension « `.txt` ») dont le nom est constitué de 3, 4, ou 5 chiffres (par exemple `000.txt`, `12.txt`, `9.txt` sont des fichiers possibles);
- Copier tous les fichiers dont l'extension est `php` ou `txt` et dont le nom ne contient aucun chiffre.

**Réponse:**

- `mkdir /var/www/img`
- `cp site/img/*.jpg /var/www/img`
- `cp *[x0-9]*.html /var/www`
- `cp site/[0-9][0-9][0-9]?([0-9]?([0-9])).txt /var/www`
- `cp site/*([0-9]).@(php|txt)`

### 3 TCP/IP, Réseau (5 points)

On suppose que l'on se trouve sur une machine autre que la machine `129.175.28.179`, (par exemple chez soi, en utilisant son ordinateur personnel et sa connexion Internet. On ne connaît pas, *a priori* l'adresse IP à laquelle on se trouve).

- (3 points) En faisant pointer son navigateur web sur `http://www.monsite.org`, on obtient un message d'erreur disant que le site ne peut être atteint. On décide alors de diagnostiquer le problème au moyen de la commande « `ping` ». Dans chacun des 3 cas suivants, dire où se situe le problème et justifier (on suppose que le problème ne vient pas du navigateur lui même) :

**1<sup>er</sup> cas** échec de `ping www.monsite.org`, échec de `ping 129.175.28.179`;

**2<sup>ème</sup> cas** échec de `ping www.monsite.org`, succès de `ping 129.175.28.179`;

**3<sup>ème</sup> cas** succès de `ping www.monsite.org` (et donc succès de `ping 129.175.28.179`).

- (2 points) On considère maintenant le premier cas de la question 1. (échec de `ping www.monsite.org`, échec de `ping 129.175.28.179`) avec les informations supplémentaires suivantes :
  - La machine sur laquelle on se trouve (celle depuis laquelle on exécute le `ping`) a l'IP `192.168.0.4`.
  - La connexion passe par un routeur ayant deux interfaces réseau, l'interface A pour IP `192.168.0.1` (qui correspond au réseau local), et l'interface B ayant pour IP `91.121.200.14` (qui est celle connectée à Internet).

Destination	Interface
<code>192.168.0.4</code>	B
<code>192.168.0.*</code>	A
<code>*.*.*</code>	B

La table de routage du routeur est la suivante :

On rappelle que lors de l'exécution de la commande `ping`, un paquet est envoyé à l'adresse de destination. La machine `pingée` renvoie un paquet de confirmation à la machine source. Si ce paquet de confirmation n'est pas reçu, la commande renvoie un échec. En supposant que le paquet initial arrive bien à la machine cible, expliquer **précisément** ce qui arrive au paquet de confirmation lorsqu'il atteint le routeur (1.5 point). Comment corriger le problème? (0.5 point).

### 4 Programmation HTML et PHP (8 points)

Toutes les fonctions qui pourraient être nécessaire sont listées dans l'aide mémoire de la page (cela ne signifie pas que vous devez toutes les utiliser coûte que coûte!).

On souhaite réaliser une horloge universelle en PHP, c'est à dire une page Web sur laquelle on peut spécifier un fuseau horaire et qui affiche l'heure actuelle ainsi que la date, pour ce fuseau. On suppose que tout le code est écrit dans le fichier `horloge.php` se trouvant à la racine du serveur web.

- (1.5 point) Écrire une fonction PHP `charge_fuseaux` qui ne prend pas d'arguments, charge le fichier `fuseaux_horaires.txt` et renvoie un tableau correspondant au contenu de ce fichier. Le fichier est une liste de lignes contenant un nom de ville et un décalage positif ou négatif par rapport au temps universel en heures, séparé par un caractère « `|` » :

```
London|0
Paris|1
Berlin|1
New York|-5
Adelaide|9.5
...
```

(la dernière ligne signifie un décalage de 9h30 entre la ville d'Adélaïde et le temps universel). Le fichier n'est *a priori* pas ordonné (ni par ordre alphabétique de Ville, ni par décalage). La fonction renvoie un tableau au format :

```
("London" => 0, "Amsterdam" => 1, "Paris" => 1, ...)
```

2. (1.5 point) Écrire une fonction PHP `date_fuseau` prenant trois arguments :
  - Un entier positif (représentant le nombre de secondes écoulées la date origine du 1/1/1970, tel que renvoyé par la fonction `time()`);
  - Une chaîne de caractères représentant un nom de ville, passé par référence;
  - Un tableau tel que celui renvoyé par la fonction `charge_fuseau`, passé par référence.la fonction renvoie la date correspondante, au format « `jj/mm/aaaa hh:mm:ss` », sous forme d'une chaîne de caractères (cf. la documentation de la fonction `date` de l'aide mémoire).
3. (2 points) Écrire une fonction `affichage` qui prend en argument une chaîne de caractères au « `jj/mm/aaaa hh:mm:ss` » et produit du code HTML réalisant l'affichage de la date. Cet affichage se fait sous la forme d'une *table* HTML contenant une seule ligne et autant de colonnes que de caractères dans la chaîne. Chaque case de la table contient une image correspondant au caractère, en utilisant les fichiers du répertoire `img`. On suppose que la chaîne donnée en entrée est valide, elle ne contient donc que des chiffres (`img000.jpg` à `img009.jpg`) ou les caractères « / » (`slash.jpg`), « : » (`deuxpoints.jpg`) ou un espace (`vide.jpg`).
4. (2 points) Écrire une fonction `affiche_form` prenant en premier argument par référence un tableau comme celui renvoyé par `charge_fuseau`, comme deuxième argument un booléen et qui affiche un formulaire HTML contenant une liste déroulante de toutes les villes et un bouton de validation. Lorsque le bouton est pressé, la page `horloge.php` est rechargée, et reçoit en paramètre POST le nom de la ville sélectionnée. Si le booléen passé en argument vaut vrai, les entrées de la liste déroulante sont triées par ordre alphabétique de villes. Si le booléen vaut faux, les villes sont triées par ordre croissant de décalage. **NB** on demande **uniquement** la partie de la page correspondant au formulaire, pas la page en entier.
5. (1 point) Écrire un fragment de code PHP qui :
  - initialise une session PHP.
  - si la variable de session "`fuseaux`" n'est définie, appelle la fonction `charge_fuseaux` et stocke le tableau résultant dans cette variableOù doit on placer ce fragment de code dans le fichier `horloge.php` pour qu'il soit fonctionnel?