CONCOURS : TECHNICIEN RECHERCHE ET FORMATION EXTERNE
BAP E : INFORMATIQUE, STATISTIQUE ET CALCUL SCIENTIFIQUE
Emploi Type : TECH D'EXPLOITATION ET DE MAINTENANCE

SUJET DE L'EPREUVE PROFESSIONNELLE

Durée 90 mn – coefficient 4

Il vous est rappelé que **votre identité ne doit figurer que dans la partie basse de la bande en-tête de la copie mise à votre disposition**. Toute mention d'identité ou tout signe distinctif porté sur toute autre partie de la copie (ou des copies) mènera à l'annulation de votre épreuve.

Ce dossier constitue le sujet de l'épreuve et le document sur lequel vous devez formuler vos réponses. Il comporte 10 pages y compris celle-ci, numérotées de 1 à 10.
Assurez-vous **immédiatement** que votre exemplaire est complet, s'il ne l'est pas, demandez en un autre au surveillant de salle.
Ne pas dégrafer et répondre au stylo sur ce document qui sert de copie-réponse.

L'usage de la calculatrice n'est pas autorisé.
L'usage de tous documents, autres que ceux qui vous seront remis lors de l'épreuve, et l'utilisation de tout matériel électronique est interdit.
Les réponses doivent être faites sur la copie, aucun document complémentaire ne sera accepté ni corrigé.
Les téléphones portables doivent être éteints.

NOM patronymique (nom de naissance):……..

Nom d'usage :……

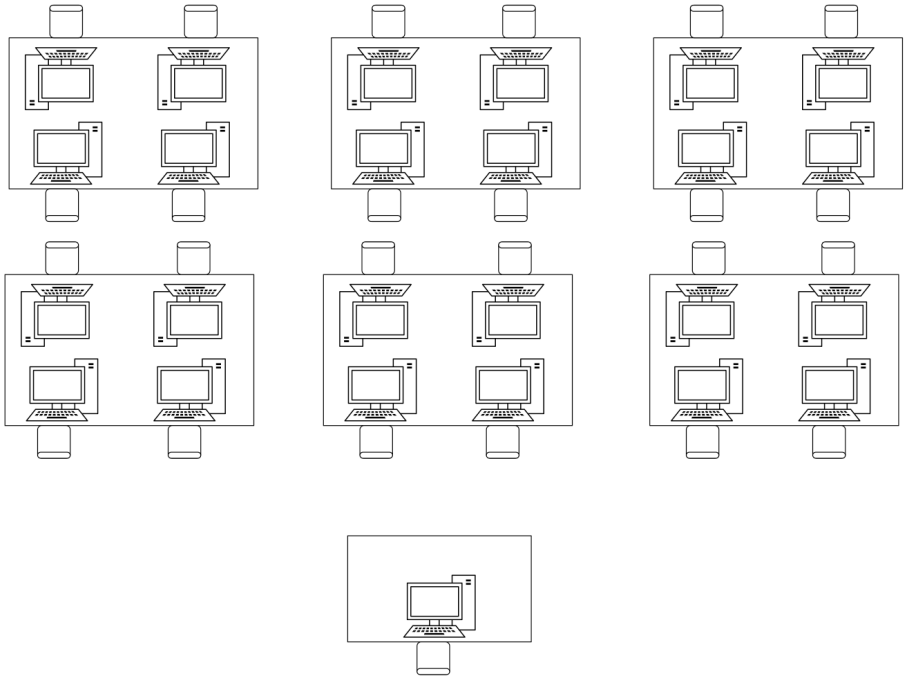
Prénom :…….

Q1 : Expliquez à un non technicien comment https est sécurisé ?

Q2 : Expliquez les pratiques que vous préconiseriez à des utilisateurs administratifs pour le choix et la gestion de leur mot de passe ?

Q3 - Quelles sont les mesures à prendre lorsque l'on suspecte la compromission d'un ordinateur ?

Q4 : Vous devez réinstaller complètement (système d'exploitation et applications) des postes clients d'une salle de travaux pratiques (définie comme sur le schéma ci-dessous) comportant des postes de travail récents (processeur 4 cœurs, 8 Go de RAM). De nombreux logiciels (CAO, calcul, logiciels bureautique, …) ont été demandés par les différents responsables de formation qui utiliseront cette salle. Vous installerez également les logiciels habituels de sécurité (antivirus) et utilitaires (compression, navigateurs, Virtual box, …). Tous les logiciels sont à installer sous Windows 7 Pro. Un enseignant souhaite également pouvoir faire des cours d'initiation à Linux et demande l'installation de la dernière version d'Ubuntu Dekstop. Les postes doivent être intégrés à un domaine Active Directory sur lequel vous disposez des droits d'administration.



Décrivez précisément le matériel utilisé, les outils de déploiement et la procédure en précisant l'ordre des opérations que vous réalisez pour automatiser ce déploiement répondant à toutes les demandes. L'installation de la salle devra s'effectuer dans les meilleurs délais.

Q5 - GREP (General Regular Expression Parser, analyseur général d'expression régulière) est un utilitaire qui sélectionne toutes les lignes qui satisfont une expression régulière (ou rationnelle).

A l'aide du man grep dont un extrait est donné en annexe, vous devez donner la ligne de commande répondant aux questions. Pour éviter toute confusion dans l'interprétation des symboles spéciaux, il est indispensable de "verrouiller" le pattern (motif) en plaçant l'expression entre guillemets " ".

A titre d'exemple, pour afficher les lignes contenant le mot "martin" du fichier passwd :
grep "martin" /etc/passwd

Toutes les questions s'appliquent au fichier passwd situé sous /etc.

5.1. Chercher toutes les lignes commençant par un f ou un F

5.2. Chercher toutes les lignes finissant par "false"

5.3. Chercher toutes les lignes contenant au moins un chiffre

5.4. Chercher toutes les lignes commençant par une majuscule

5.5. Chercher toutes les lignes commençant par b p ou g

_____

5.6. Chercher toutes les lignes contenant un k et qui soient précédées de n'importe quelle lettre minuscule ou majuscule

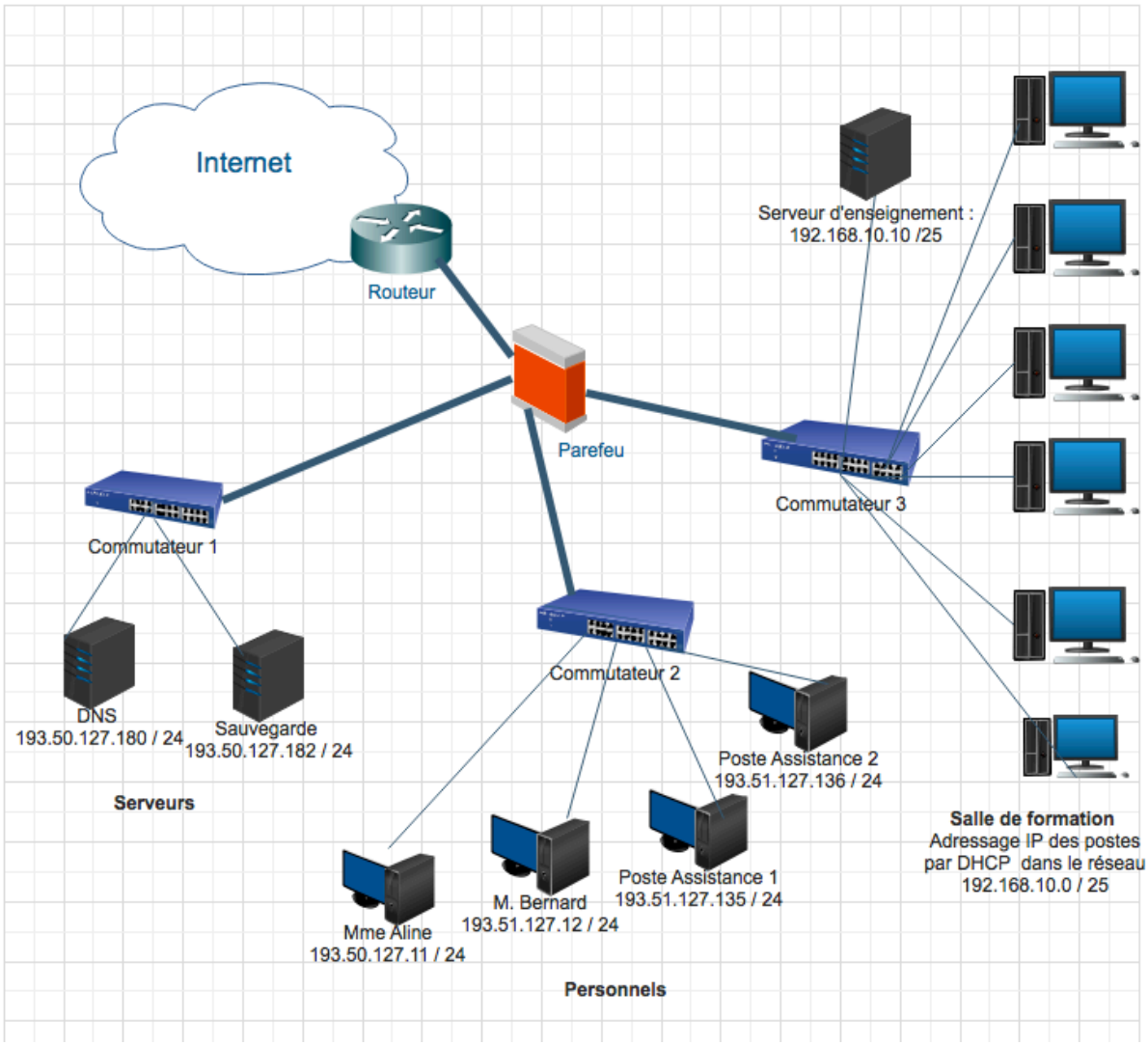
_____

5.7. Chercher toutes les lignes dont au moins un mot a pour seconde lettre un o

_____

Q6 : Vous êtes affecté-e au service d'assistance. M. Bernard a laissé le message suivant : "Je n'arrive plus à accéder à la page web du ministère de l'enseignement supérieur. Il faut me dépanner."

Dans la base documentaire du service, vous disposez du schéma ci-dessous :



Détaillez précisément les vérifications que vous faites et les actions que vous entreprenez pour diagnostiquer le problème.

ANNEXE : MAN de GREP

GREP(1) GREP(1)

NAME

grep, egrep, fgrep - print lines matching a pattern

SYNOPSIS

grep [options] PATTERN [FILE...]
grep [options] [-e PATTERN | -f FILE] [FILE...]

DESCRIPTION

Grep searches the named input FILEs (or standard input if no files are named, or the file name - is given) for lines containing a match to the given PATTERN. By default, grep prints the matching lines.

In addition, two variant programs egrep and fgrep are available. Egrep is the same as grep -E. Fgrep is the same as grep -F.

REGULAR EXPRESSIONS

A regular expression is a pattern that describes a set of strings. Regular expressions are constructed analogously to arithmetic expressions, by using various operators to combine smaller expressions.

Grep understands two different versions of regular expression syntax: "basic" and "extended." In GNU grep, there is no difference in available functionality using either syntax. In other implementations, basic regular expressions are less powerful. The following description applies to extended regular expressions; differences for basic regular expressions are summarized afterwards.

The fundamental building blocks are the regular expressions that match a single character. Most characters, including all letters and digits, are regular expressions that match themselves. Any metacharacter with special meaning may be quoted by preceding it with a backslash.

A bracket expression is a list of characters enclosed by [and]. It matches any single character in that list; if the first character of the list is the caret ^ then it matches any character not in the list. For example, the regular expression [0123456789] matches any single digit.

Within a bracket expression, a range expression consists of two characters separated by a hyphen. It matches any single character that sorts between the two characters, inclusive, using the locale's collating sequence and character set. For example, in the default C locale, [a-d] is equivalent to [abcd]. Many locales sort characters in dictionary order, and in these locales [a-d] is typically not equivalent to

[abcd]; it might be equivalent to [aBbCcDd], for example. To obtain the traditional interpretation of bracket expressions, you can use the C locale by setting the LC_ALL environment variable to the value C.

Finally, certain named classes of characters are predefined within bracket expressions, as follows. Their names are self explanatory, and they are [:alnum:], [:alpha:], [:cntrl:], [:digit:], [:graph:], [:lower:], [:print:], [:punct:], [:space:], [:upper:], and [:xdigit:]. For example, [[:alnum:]] means [0-9A-Za-z], except the latter form depends upon the C locale and the ASCII character encoding, whereas the former is independent of locale and character set. (Note that the brackets in these class names are part of the symbolic names, and must be included in addition to the brackets delimiting the bracket list.) Most metacharacters lose their special meaning inside lists. To include a literal] place it first in the list. Similarly, to include a literal ^ place it anywhere but first. Finally, to include a literal - place it last.

The period . matches any single character. The symbol \w is a synonym for [[:alnum:]] and \W is a synonym for [^[:alnum]].

The caret ^ and the dollar sign $ are metacharacters that respectively match the empty string at the beginning and end of a line. The symbols \< and \> respectively match the empty string at the beginning and end of a word. The symbol \b matches the empty string at the edge of a word, and \B matches the empty string provided it's not at the edge of a word.

A regular expression may be followed by one of several repetition operators:

? The preceding item is optional and matched at most once.
*** The preceding item will be matched zero or more times.**
+ The preceding item will be matched one or more times.
{n} The preceding item is matched exactly n times.
{n,} The preceding item is matched n or more times.
{n,m} The preceding item is matched at least n times, but not more than m times.

Two regular expressions may be concatenated; the resulting regular expression matches any string formed by concatenating two substrings that respectively match the concatenated subexpressions.

Two regular expressions may be joined by the infix operator |; the resulting regular expression matches any string matching either subexpression.

Repetition takes precedence over concatenation, which in turn takes precedence over alternation. A whole subexpression may be enclosed in parentheses to override these precedence rules.

The backreference \n, where n is a single digit, matches the substring previously matched by the nth parenthesized subexpression of the regular expression.

In basic regular expressions the metacharacters ?, +, {, |, (, and) lose their special meaning; instead use the backslashed versions \?, \+, \{, \|, \(, and \).

Traditional egrep did not support the { metacharacter, and some egrep implementations support \{ instead, so portable scripts should avoid { in egrep patterns and should use [{] to match a literal {.

GNU egrep attempts to support traditional usage by assuming that { is not special if it would be the start of an invalid interval specification. For example, the shell command egrep searches for the two-character string instead of reporting a syntax error in the regular expression. POSIX.2 allows this behavior as an extension, but portable scripts should avoid it.