

Examen d'ASR2

L3 IF

31 Mai 2018

- Pas de connexion avec une machine autre que la vôtre et celles mentionnées dans le sujet.
- En particulier : pas de navigateur, messagerie instantanée, utilisation de téléphone ou smartphone. (Attention, un script surveille les programmes que vous lancez sur votre machine.)
- Vous devez utiliser le compte fourni. Pas de connexion à un autre compte.
- Vous avez le droit de consulter les pages **man** et les documents fournis (une sous-partie du cours, et les programmes exemples).
- Ne changez pas le mot de passe qui vous a été assigné. Je serai dans l'incapacité de récupérer votre programme.
- Notez votre login sur toutes vos copies. Numérotez vos copies.
- Le barème est donné à titre indicatif.

0.1 Non-identifiant (-2 points)

Oubliez d'indiquer votre login sur l'une de vos copies.

1 “Théorie” (8 points)

1.1 Appels système

Expliquez quel est l'intérêt des appels système, et comment ils se déroulent.

1.2 En-têtes

Quels champs apparaissent dans l'en-tête d'une trame ethernet ? Quels champs apparaissent dans l'en-tête d'un paquet IPv4 ?

1.3 TCP

Expliquez comment se passe l'établissement d'une connexion TCP (c'est-à-dire, expliquez quels différents messages sont envoyés entre l'initiateur de la connexion et le receveur).

1.4 Adresses

Comment un ordinateur connecté à un réseau fait pour connaître la correspondance entre les adresses MAC et les adresses IP ?

1.5 TCP/UDP

Quelles sont les différences entre UDP et TCP ? Quelles sont leurs utilisations ?

1.6 C'est quoi ça ?

Que fait le code suivant ? À quoi sert-il ? Décrivez ce qui se passe lors de son execution.

```
if (!fork ())
    if (fork ())
        exit (0);
    else {
        //quelquechose
    }
//autrechose
```

1.7 Switch

Expliquez le mécanisme utilisé par un switch (ou commutateur réseau) pour savoir sur quel port envoyer une trame ethernet.

1.8 Variables de condition

Expliquez à quoi servent les variables de condition.

2 Pratique (12 points)

Expliquez sur votre copie, pour les questions suivantes, lesquelles vous avez réussi à faire, et lesquelles vous avez tenté de faire sans succès. Dans ce deuxième cas, vous pouvez expliquer quelle aurait été votre méthode, ceci afin de gratter des points.

Si vous avez une idée de comment procéder, mais sans réussir, expliquez sur votre copie quelle aurait été votre méthodologie.

2.1 Des chiffres et la lettre C (I)

Le programme `primes.c` compte le nombre de nombres premiers entre 2 et le nombre donné en argument. Deux moyens de le faire y sont implémentés : `countPrime` utilise `isPrime`, qui est le test de Miller-Rabin pour vérifier la

primalité d'un nombre, et `countNonFact` utilise `isFact`, qui implémente un algorithme trivial de factorisation pour décider si un nombre n'est pas premier. Or, l'une ou l'autre des méthodes peut être plus rapide, selon le nombre dont on cherche à vérifier la primalité.

Parallélisez le code (avec `pthread`) pour lancer les deux algorithmes en même temps (`isNonFact` et `isPrime`), et arrêter le second algorithme dès qu'un des deux a réussi à décider si un nombre est premier ou non.

Notez que votre code pourra être plus lent : ce n'est pas important pour cette question.

Laissez, à la racine de votre compte, la version parallélisée sous le nom « `primes_pthread.c` ».

2.2 Des chiffres et la lettre C (II)

Optimisez le code, du moyen que vous voulez, pour aller le plus loin possible. Vous pouvez aussi bien utiliser des astuces mathématiques¹ que paralléliser le code. Votre code sera lancé sur une machine avec 96 cœurs, et sera noté selon sa vitesse.

Laissez, à la racine de votre compte, la version la plus rapide possible sous le nom « `primes_rapide.c` ».

2.3 Gros fichier

Le fichier `gros.bz2` est un GROS fichier (128Go) comprimé en bzip2. Cherchez la position x où se trouve votre identifiant (`z4678xx`) dans ce fichier, et allez récupérer votre code (un nombre décimal à 8 chiffres) en position $x + 2^{35} = x + 34359738368$. N'essayez pas de décompresser le fichier sur le disque, il n'y a de toutes façons pas assez de place!

2.4 ICMP

Envoyez à la machine 163.172.27.59 un message ICMP contenant le mot de passe qui vous a été assigné².

Vous pouvez vérifier si cela a fonctionné en vous connectant à la même machine, sur le port 4567, en TCP.

2.5 Coffre fort

Le programme `coffretfort` vous demande votre identifiant challenge, et vous donnera une clef de validation pour la question. Problème : ce programme est aussi protégé par un code secret : une chaîne de caractères. Piratez-le, et donnez la clef de validation.

Valeur test : Pour l'identifiant `z456082`, la clef doit être `70DDF9BD`

1. Oui, implémenter un crible quadratique, un NFS ou un ECM est autorisé... mais ne vous aidera pas beaucoup.

2. Je sais, c'est moche, mais c'est comme ça.

2.6 Coffre fort (bonus)

Trouvez un/le code secret de la question précédente.