ECOLE POLYTECHNIQUE DE MONTREAL

Département de génie informatique et génie logiciel

Cours INF4410: Systèmes répartis et infonuagique (Hiver 2017)
3 crédits (3-1.5-4.5)

EXAMEN FINAL

DATE: Jeudi le 20 avril 2017

HEURE: 9h30 à 12h00

DUREE: 2H30

NOTE: Toute documentation permise, calculatrice non programmable permise

Ce questionnaire comprend 4 questions pour 20 points

Question 1 (5 points)

a) Le serveur DNS d'un fournisseur de service Internet peut servir des requêtes DNS soit de manière itérative ou de manière récursive. De manière itérative, le serveur traite la requête avec son CPU pendant 2ms, lit en plus son disque dans 20% des cas pendant 20ms, et retourne dans 40% des cas la réponse demandée et dans 60% des cas une redirection vers un serveur plus haut dans la hiérarchie. En recevant une redirection, le client doit aller chercher la réponse sur un autre serveur, ce qui lui prend 10ms. Si le serveur est configuré pour fonctionner de manière récursive, il prendra aussi 2ms de CPU, et en plus 20ms de disque dans 20% des cas. Ensuite, il retournera la réponse demandée dans 90% des cas et dans 10% des cas devra faire une requête et attendre la réponse avant de la retourner, ce qui lui demande en plus 1ms de CPU et 20ms d'attente. Quel sera le délai moyen vu par un client pour obtenir la réponse demandée dans chaque cas (récursif ou itératif)? Quel est le nombre de requêtes par seconde que peut soutenir le serveur, s'il n'utilise qu'un seul thread, de manière itérative? De manière récursive? **(2 points)**

b) Sur un serveur DNS en autorité, la base de donnée entre entièrement en mémoire et l'ingénieur responsable a réussi à faire traiter chaque requête en 3us (1us recevoir le paquet UDP de requête, 1us chercher la réponse en mémoire, 1us renvoyer la réponse). Chaque paquet UDP (requête ou réponse) occupe en moyenne 100 octets. Pour simplifier le calcul, on suppose que les transferts peuvent se faire à la bande passante indiquée, on néglige les autres effets comme la latence d'envoi pour les paquets. Le serveur est connecté à deux réseaux qui peuvent chacun soutenir 100 mégabits/s par seconde simultanément dans chaque direction. Un usager malicieux a réussi à prendre le contrôle d'un certain nombre de processeurs dans des caméras IP et les utilise pour effectuer une attaque en déni de service sur le serveur DNS. Chaque caméra IP est capable de générer 100000 paquets de requête par seconde et est connectée par un réseau à 1mégabit/s. Les requêtes sont émises avec de fausses adresses de retour et ne reviennent jamais aux caméras. Combien de caméras sont requises pour saturer ce serveur DNS, en tenant compte de la capacité de leur processeur et de leur réseau respectifs? **(2 points)**

c) Pour les requêtes de recherche avec le protocole LDAP, il est possible de spécifier une limite sur la longueur de la réponse ainsi que sur le traitement requis. Pourquoi de tels paramètres ont-ils été incorporés à ce protocole? Dans quelle situation est-ce utile? **(1 point)**

Question 2 (5 points)

a) Un client A effectue deux requêtes auprès d'un serveur de temps B. La première requête part de A à 9h00m40.000 et arrive à B à 9h00m20.010, puis la réponse part de B à 9h00m20.060 et arrive à A à 9h00m40.100. La seconde requête part de A à 9h01m50.000 et arrive à B à 9h01m30.030, puis la réponse part de B à 9h01m30.130 et arrive à A à 9h01m50.140. Quelles sont les valeurs de décalage et d'incertitude calculées pour chaque requête? Laquelle valeur (i.e. de quelle requête) devrait-on utiliser? Peut-on combiner les informations des requêtes pour avoir un meilleur estimé du décalage? **(2 points)**

b) Dans le cadre du travail pratique 2, vous avez implémenté un service de calcul sécurisé (pas de réponse malicieuse) et un service non sécurisé (réponses possiblement malicieuses). Quelle était

votre stratégie pour détecter et laisser tomber les réponses malicieuses? Quel était le surcoût de cette détection (temps CPU pour calcul sécurisé versus calcul non-sécurisé, mais avec très peu de réponses malicieuses finalement)? Quelle était la probabilité de ne pas détecter une réponse malicieuse si les réponses malicieuses sont générées indépendamment sur chaque serveur de calcul? Si elles sont générées de manière concertée? **(2 points)**

c) Nous avons vu deux algorithmes qui peuvent être utilisés pour une élection dans un système réparti, l'élection hiérarchique et l'algorithme de Paxos. L'élection hiérarchique est a priori beaucoup plus simple. Quels sont donc les avantages de l'algorithme de Paxos? Donnez un exemple où seul l'algorithme de Paxos fonctionnerait correctement? **(1 point)**

Question 3 (5 points)

a) Lesquelles des transactions T, U et V pourraient être validées si une validation en reculant était utilisée pour vérifier la cohérence des transactions? Une validation en avançant? **(2 points)**

T: Début
U: Début
V: Début
T: Read(e)
T: Read(a)
T: Read(d)
U: Read(a)
U: Read(e)
T: Write(a,21)
T: Write(b,12)
T: Compléter
U: Read(b)
V: Read(b)
U: Write(c,3)
U: Write(f,8)
U: Compléter
V: Read(c)
V: Write(d,12)
V: Write(e,1)
V: Compléter

b) Supposons que les opérations listées en a) pour les transactions T, U et V sont en fait des transactions réparties. Un premier serveur X contient les variables a et b, un second serveur Y contient les variables c et d, alors que le troisième serveur Z contient les variables e et f. Le système utilise un protocole de fin de transaction atomique à deux phases. En supposant que les trois transactions puissent être validées avec les opérations telles que listées, que retrouverait-on dans le journal de chacun des serveurs (X, Y et Z)? Expliquez qu'est-ce qui est écrit à quel moment. **(2 points)**

c) Pour les transactions locales, les algorithmes de contrôle optimiste de la concurrence, validation en avançant et validation en reculant, peuvent être intéressantes. Sont-elles applicables pour des transactions réparties? Expliquez. **(1 point)**

Question 4 (5 points)

a) Un système transactionnel en ligne, pour des ventes aux enchères, utilise de la redondance à plusieurs niveaux. Il est connecté à 2 fournisseurs Internet et peut fonctionner tant que l'un des deux est opérationnel (i.e. chaque serveur est connecté aux deux réseaux). Il y a ensuite 3 serveurs de façade alors qu'il suffit d'un seul pour fonctionner. Les serveurs de façade font des requêtes à 2 serveurs de base de données redondants alors qu'un seul suffit. Deux unités de disque RAID sont utilisées. Chaque unité de disque RAID fonctionne si au moins 3 de ses 4 disques sont fonctionnels. Deux configurations sont possibles. Dans la première, chaque unité RAID est connectée à un seul des deux serveurs de base de données; un serveur ne fonctionnera que si son unité RAID fonctionne. Dans la seconde configuration, chaque serveur de base de données peut accéder aux deux unités RAID; un serveur ne fonctionnera que si au moins une des 2 unités est fonctionnelle. Si la probabilité de panne est de .1 pour un fournisseur Internet, .2 pour un serveur de façade, .15 pour un serveur de base de données et .25 pour un disque, et que la probabilité de panne est négligeable pour les autres composantes, quelle sera la probabilité que le service soit disponible aux clients pour chacune des deux configurations? **(2 points)**

b) Dans le cadre du travail pratique 3, vous avez utilisé des gabarits Heat afin de spécifier le comportement d'un service. Que doit-on mettre dans le gabarit afin de répartir la charge entre plusieurs serveurs? Que doit-on ajouter au gabarit Heat afin d'ajuster selon la charge le nombre de serveurs entre lesquels le travail est réparti? Donnez le type des principales ressources requises et leur fonction. **(2 points)**

c) Le système Google Wide Profiling permet de comparer la performance fournie par les différentes versions d'un logiciel, par différents types de matériel, ou même de calculer le coût global (temps CPU) pour une fonction donnée. Comment cela fonctionne-t-il? Est-ce que le surcoût est important pour obtenir ces informations? Sont-elles fiables? Expliquez. **(1 point)**

Le professeur: Michel Dagenais