Vrai ou faux:

1. Le terme “paravirtualisation” désigne le fait de masquer la virtualisation au système invité (à la VM).

> *Faux: c'est le contraire. La paravirtualisation est une technique pour améliorer la performance de la virtualisation, et qui requiert la coopération du système d'exploitation de la machine virtuelle.*

2. Dans certains cas, la virtualisation par traduction binaire peut être plus rapide qu'une plateforme non virtualisée.

> *Dans certains cas, c'est vrai. Des instructions priviliégiées qui pourraient normalement prendre plusieurs cycles à s'exécuter peuvent être remplacées (traduites) par de simples accès plus rapides en mémoire. Cela étant dit, ce n'est pas le cas général et une plateforme virtualisée est globalement moins performante.*

3. Il est possible de rouler une machine virtuelle Mac OS X sur un système Linux en faisant de la paravirtualisation.

> *La paravirtualisation en tant que telle n'est pas une technique de virtualisation mais signifie simplement une coopération de la part du système d'exploitation virtualisé.*

4. Il est possible de rouler une machine virtuelle Mac OS X sur un système Linux en utilisant LXC (conteneur).

> *Faux: les conteneurs ne permettent pas de choisir le système d'exploitation de la machine virtuelle.*

5. L'avantage avec Windows Azure est un accès direct aux ressources matérielles qu'on exploite.

> *Faux: le point essentiel de l'infonuagique est que l'utilisateur n'a plus à gérer les ressources matérielles. Si on a besoin d'un accès direct au matériel physique, on ne devrait pas utiliser un plateforme infonuagique.*

6. NFS utilise TCP pour la communication sur le réseau.

> *Faux: NFS utilise UDP (les fonctions RPC sur un serveur NFS sont idempotentes puisque la connexion est supposée non fiable).*

7. Le Protocol Buffer de Google est un algorithme d'exclusion mutuelle répartie qui utilise un *buffer* pour garantir l'ordre d'obtention du verrou.

> *Faux: c'est un protocole de sérialisation utilisé pour faire du RPC (ex: sérialisation des paramètres à envoyer lors de l'invocation d'une fonction RPC).*

8. Lorsqu'on fait une recherche sur Google, le moteur de recherche crée une opération Map-Reduce qui permet de scanner Internet au complet en parallèle pour effectuer la recherche.

> *Faux: on ne scanne pas Internet au complet à chaque fois qu'un utilisateur lance une recherche*

sur Google (Google = millions de requêtes par seconde).

Questions à développement:

9. Décrivez l'architecture d'un service de diffusion de films similaire à Netflix. Utilisez les services offerts par un fournisseur d'infonuagique (comme Amazon).

Il n'y a pas nécessairement une unique réponse. On pourrait suggérer:
- Service de stockage pour les films et séries.
- Base de données non-relationnelle pour enregistrer de l'information sur les clients comme leurs préférences, leur historique, où ils sont rendus dans une certaine série, etc.
- Potentiellement une base de données non-relationnelle en graphe (GraphDB) pour les algorithmes qui proposent aux utilisateurs des films à voir selon leur historique.
- Un service de calcul qui répond aux requêtes.
- Un service de calcul pour parcourir la base de données en graphe et exécute l'algorithme de suggestions de films

10. Que signifie le terme surengagement des ressources?

Une ressource est surengagée lorsqu'elle est promise à plus "d'utilisateurs" qu'elle ne peut gérer; donc lorsqu'elle a plus ***d'engagements*** *qu'elle ne peut gérer. Par exemple, si plusieurs machines virtuelles se battent pour obtenir le CPU et chacune d'entre elles pense y avoir un accès exclusif, le CPU est dit être surengagé. Si l'on a deux machines virtuelles et chacune d'entre elles pense avoir 8GB de mémoire (pour un total de 16GB), alors que le système hôte n'a que 8GB de mémoire physique au total, alors la mémoire physique est surengagée.*

Note externe au cours*: Le surengagement des ressources n'est pas spécifique à l'infonuagique. Notamment, la mémoire virtuelle (revoir le cours des systèmes d'exploitation) permet de surengager la mémoire physique d'un système.*

11. Expliquez comment une machine virtuelle peut être migrée sans être complètement arrêtée.

La migration se fait en plusieurs passes (en plusieurs rounds) en copiant à chaque fois uniquement les pages mémoires qui ont été modifiées. Une migration pourrait se produire comme suit:
- À la première passe, on copie l'espace mémoire total de la VM de la machine physique source vers la machine destination.
- Lors de la seconde passe, on copie seulement les pages mémoires qui ont été modifiées par la VM depuis la première round.
- Lors de la troisième round, s'il y a eu peu de pages qui ont été modifiées depuis la passe précédente, on met la VM en pause et on transfère uniquemenet ces pages-ci. Ensuite, une fois que l'entièreté de l'espace mémoire a été transféré, on démarre la VM sur la machine destination. Ce mécanisme est transparent à la VM, puisque son état est inchangé lors de la migration. Seule une "anomalie" pourrait être perçue lors de la migration, qui est généralement très rapide (peu de pages à copier lors de la dernière passe et la migration est souvent faite au sein d'un même parc, voire même au sein d'un même rack).

Note externe au cours*: ceci est un algorithme simplifié, en réalité il pourrait y avoir plus que 3 round et le comportement pourrait varier d'une implémentation à l'autre.*

12. En quoi la réplication améliore-t-elle la disponibilité?

Disponibilité = performance + tolérance aux pannes. La réplication améliore la performance (balancement des requêtes) ainsi que la tolérance aux pannes (la même information se trouve à plusieurs endroits).

13. La réplication active (cohérence forte) permet-elle au client d'avoir un meilleur ou un pire débit d'écriture?

La réplication active entraîne généralement une pire latence d'écriture. Le client émet une seule écriture, mais celle-ci cause en réalité plusieurs écritures du côté du serveur (une écriture sur chacune des réplicas). On a donc plusieurs écritures sur le chemin critique de la requête.

***Note générale**: attention, une pire latence ne veut pas toujours dire **un pire débit**:*
"Never underestimate the bandwidth of a station wagon full of tapes hurtling down the highway." -Tanenbaum, Andrew S. (1989)

14. Pourquoi le service Dropbox utilise-t-il S3 au lieu de gérer le service de stockage sur disque?

Ceci pourrait-être expliqué par plusieurs raisons:
- Raisons de fiabilité et de coût: Amazon est une fournisseur infonagique fiable et reconnu. Dropbox en tire avantage et n'a pas à réimplémenter et maintenir un service de stockage au complet. Amazon se charge des pannes, bugs, mises-à-jour, etc. De plus, Dropbox ont sûrement un "prix d'ami" avec Amazon pour rentabiliser une utilisation volumineuse de leur service S3.
- Raisons historiques: Dropbox a commencé comme une "startup" pour qui il était plus facile de transférer la responsabilité de stockage chez Amazon. Ce changement fondamental devient de plus en plus difficile (et de plus en plus coûteux) avec le temps.

15. Comment se fait-il que charger une page web peut prendre plusieurs secondes, alors qu'il est possible de jouer un jeu en ligne, avec rendu graphique 3D et un univers de jeu immense, en temps réel.

Plusieurs raisons peuvent entrer en compte:
- Le jeu utilise très probablement UDP (pas besoin de retransmettre un paquet perdu puisque l'information qu'il contient est probablement expirée) alors qu'on utilise TCP pour le site web
- Le serveur du jeu envoie le strict minimum au client (un évènement ponctuel qui s'est produit, le position d'un joueur, etc) et le calcul ainsi que le rendu se font chez le client
- La page web pourrait se trouver sur un serveur distant (et le contenu de la page web n'est pas en cache chez le serveur) alors que certaines compagnies de jeu en ligne ont plusieurs serveurs répartis géographiquement (CDN)
- Ouvrir une page web requiert aussi une résolution initiale de nom DNS
- Le contenu de certaines pages web pourrait être "éparpillé" à travers plusieurs serveurs (un serveur pour le contenu HTML, un service de stockage pour les images, etc.)
- Ouvrir une page web requiert d'établir une connexion initiale (coûteux) alors qu'un jeu garde la connexion ouverte et ne fait qu'envoyer et recevoir très peu de données
- D'autres raisons peuvent être acceptées

16. Quelqu'un vous dit que lire un octet d'un service de stockage de fichiers est plus rapide que lire 8 octetes car la latence pour lire un unique octet est moindre. A-t-il raison?

S'il s'agissait réellement de lire un unique octet, alors oui il a raison. Ce qu'il faut savoir, c'est que demander à "lire juste un octet" est pratiquement impossible. Les lectures se font en blocs pour justement éviter de déranger le serveur et/ou le disque et/ou le réseau pour des petites requêtes. Lorsque l'on demande à lire "juste un octet", c'est un bloc bien plus gros qui est lu. De plus, la lecture d'un octet d'un fichier requiert d'avoir du métadata sur ce fichier (où il est enregistré sur le disque, etc.) ce qui pourrait causer aussi des requêtes de lectures en plus.

Plusieurs optimisations sont faites sur les entrées et sorties et prédire comment le système va réellement se comporter est généralement très difficile, à moins d'avoir une connaissance approfondie de son fonctionnement.

En résumé, demander à lire un octet ou 8 octets (ou potentiellement plus) d'un service de stockage (ou même d'un disque) prend pratiquement le même délai.

17. On vous demande de choisir entre deux configurations pour le prochain serveur de fichiers de votre entreprise. La première configuration consiste en un ordinateur avec 4 disques en miroir. Chaque disque contient l'ensemble des données et un seul disque suffit donc. Le second système est constitué de deux ordinateurs, qui sont deux serveurs redondants, chacun étant connecté à deux disques en miroir et un seul disque suffit donc pour un serveur. La probabilité qu'un ordinateur soit opérationnel (hormis les disques) est de 0.95. La probabilité qu'un disque soit opérationnel est de 0.85. Quelle est la probabilité que le service soit disponible, pour chacune des deux configurations?

Dans la première configuration, le système de disque est disponible sauf si les 4 sont indisponibles $1 - (1 - .85)\ 4 = 0.99949375$. Le service sera disponible si les disques et l'ordinateur le sont, $0.99949375 \times 0.95 = 0.949519063$.
Dans le second cas, sur un serveur,
les disques seront disponibles sauf si les 2 sont indisponibles $1 - (1 - .85)\ 2 = 0.9775$.
Un serveur sera disponible si l'ordinateur et les disques le sont $0.9775 \times 0.95 = 0.928625$.
Le service sera disponible sauf si les deux serveurs redondants sont indisponibles $1 - (1 - 0.928625)\ 2 = 0.994905609$. Cette deuxième configuration est donc nettement mieux pour réduire le temps d'indisponibilité. Elle est toutefois légèrement plus coûteuse puisqu'elle utilise deux ordinateurs plutôt qu'un, tout en conservant le même nombre de disques.

18. Les interblocages posent un problème sérieux dans les systèmes de base de données répartis. i) Comment peut-on détecter un interblocage dans un tel système? ii) Est-il possible de le faire sans simultanément arrêter tous les processus impliqués? iii) Doit-on constamment vérifier la présence d'interblocages, ou peut-on le faire seulement à la demande (selon quel critère)? iv) A défaut de faire une détection précise des interblocages, quel mécanisme peut-on utiliser pour s'assurer de ne pas laisser des transactions bloquées indéfiniment?

i) Pour détecter les interblocages, il est possible de construire un graphe de dépendance entre les verrous, ceux qui les possèdent et ceux qui les attendent, et de vérifier l'existence d'un cycle dans le graphe. ii) A défaut de tout arrêter pour avoir un cliché exact du graphe réparti de dépendance, il est possible de le construire de manière incrémentale un graphe approximatif en interrogeant les processus impliqués les uns après les autres. Si un interblocage est

présent, les dépendances en cause sont figées et ceci apparaîtra dans le graphe obtenu, même s'il est construit incrémentalement. Cependant, si une transaction est annulée au milieu de la construction de notre graphe, il est possible que nous détections de manière erronnée un interblocage. iii) La vérification des interblocages ne se fait pas sans arrêt puisque ce serait trop coûteux. Elle peut se faire à intervalle régulier ou seulement lorsqu'il est détecté qu'une transaction reste anormalement longtemps en attente d'un verrou. iv) Dans la plupart des cas, la détection des interblocages n'est pas faite directement. Chaque transaction est simplement annulée si elle reste bloquée trop longtemps.

19. Vous avez besoin d'un service d'exclusion mutuelle pour votre système réparti. On vous propose un système symétrique réparti. Pour obtenir un verrou, un système demande la permission de chaque autre système. Celui qui détient le verrou attend d'en avoir terminé avant de donner la permission. Celui qui est en demande de verrou attend d'avoir obtenu et fini du verrou avant de donner la permission. Est-ce que ce système est sûr, vivace et respecte l'ordre? Expliquez. Est-ce que ce système est efficace?

Ce système est sûr puisque la permission de chacun ne peut être obtenue tant qu'un autre processus détient le verrou. Ce système respecte l'ordre puisque chaque demande est traitée dans l'ordre par chaque processus. Le système est vivace puisque chacun passe dans l'ordre et que le système ne peut normalement bloquer; ainsi chacun verra éventuellement sa demande satisfaite. Ce système n'est pas efficace puisqu'il ne fonctionne que si les n processus répartis sont opérationnels, et puisque l'obtention de chaque verrou demande au moins n messages, contre 2 pour un serveur central.

20. Donnez un exemple de pannes transitoire, intermittente et permanente.

Panne transitoire: *exemple.*
Panne intermittente:
- un bris dans le service de refroidissement. Les ordinateurs chauffent, s'éteignent, puis redémarrent lorsqu'ils refroidissent. Le cycle répète, ce qui cause des pannes intermittentes.
- un bug rare qui cause une requête à échouer
Panne permanente:
- une erreure de segmentation
- le bris d'un disque

21. Une entreprise opère 9 ordinateurs identiques, 3 pour chacun de trois départements très semblables. Chacun des 3 ordinateurs pour un département est affecté à un service particulier (serveur LDAP, serveur de fichiers, serveur de compilation) et ces trois serveurs ont un taux d'occupation moyen de 10%, 20% et 30% respectivement. Un administrateur de système, inspiré par le cours INF4410, propose de consolider ces serveurs en les virtualisant, ce qui permettrait de réduire le nombre d'ordinateurs requis. Le surcoût de la virtualisation est de 20%, ce qui prenait 1 seconde en prendrait 1.2. Deux solutions sont envisagées, conserver un ordinateur par département qui roule les trois machines virtuelles de ce département, ou avoir un nuage de 3 ordinateurs physiques sur lesquels seraient répartis les 9 machines virtuelles. Que deviendrait le taux d'utilisation moyen dans chaque cas, en supposant que la charge sur chaque serveur était assez uniforme dans le temps? Quelle solution vous semble la plus intéressante? Pourquoi?

Sur 100 secondes, les trois serveurs en prenaient respectivement 10, 20 et 30, soit un total de 60. Avec le surcoût de la virtualisation, ceci deviendrait 60 × 1.2 = 72, soit un taux

d'occupation de 72%. Il n'y a pas de différence de taux d'utilisation entre les deux cas, trois fois plus de charge sur trois ordinateurs au lieu d'un seul. La solution du nuage est plus intéressante car elle permet une certaine tolérance aux pannes. Par contre, elle peut être un peu plus difficile à mettre en oeuvre, et ne sépare pas tout à fait aussi bien les activités des trois départements au niveau de la sécurité informatique.

22. Sur le nuage de la compagnie Amazon, un service de répartiteur existe qui envoie les requêtes reçues à tour de rôle à un des serveurs disponibles. Le répartiteur reçoit de l'information sur les différents serveurs (taux d'utilisation des serveurs, et temps de réponse aux requêtes). Le répartiteur peut aussi décider d'instancier des serveurs supplémentaires ou de retirer des serveurs instanciés. Quel critère est-ce que le répartiteur utilise pour choisir le prochain serveur auquel envoyer une requête reçue? Quel critère utilise-t-il pour décider d'activer une instance supplémentaire de serveur? Pour retirer une instance de serveur?

Le répartiteur envoie la requête reçue au serveur qui répond le plus rapidement, ou les distribue à tour de rôle lorsque la différence de temps n'est pas importante. Lorsque le taux d'utilisation des serveurs est trop élevé, de nouvelles instances sont ajoutées. Lorsque le taux d'utilisation est trop faible, des instances sont retirées.

23. Les solutions de virtualisation comme KVM offrent la virtualisation complète ou la paravirtualisation. Quels sont les avantages et limitations de ces deux alternatives? Dans quelle situation choisirait-on de préférence chacune?

La paravirtualisation est plus rapide puisque l'opération demandée est directement déléguée à la machine physique, plutôt que d'émuler le comportement d'un dispositif physique particulier, comme une carte réseau, avant de déléguer le travail à la machine physique. Toutefois, la paravirtualisation n'est possible que lorsqu'on peut facilement configurer de cette manière la machine virtuelle. Lorsqu'on a plein contrôle sur la configuration de la machine virtuelle, la paravirtualisation est préférable puisqu'elle est plus rapide et ne requiert pas de support matériel. Par contre, s'il n'est pas possible de modifier la machine virtualisée, par exemple parce qu'elle est fournie par un client qui ne sait pas que son ordinateur est virtualisé ou qui n'est pas intéressé ou capable de modifier sa configuration, alors la virtualisation complète sera nécessaire.