Hébergement d'applications Web dans le cloud AWS

Bonnes pratiques

Septembre 2012

Matt Tavis, Philip Fitzsimons

Résumé

L'hébergement Web évolutif et à haute disponibilité peut être complexe et onéreux. Les architectures Web évolutives classiques ont non seulement dû mettre en œuvre des solutions complexes pour garantir des hauts niveaux de fiabilité, mais ont également eu besoin de prévisions de trafic précises pour fournir un service client optimal. Des périodes de pics de trafic denses et des variations imprévues des modèles de trafic entraînent des faibles taux d'utilisation de matériel onéreux, une augmentation considérable des frais d'entretien du matériel inactif et une utilisation non rentable du capital consenti dans du matériel sous-utilisé.

Amazon Web Services fournit une infrastructure fiable, évolutive, sûre et ultra performante pour les applications Web les plus exigeantes et dont le coût informatique correspond aux modèles de trafic du client en temps réel.

Public ciblé

Ce livre blanc est destiné aux gestionnaires et aux architectes de systèmes informatiques qui comptent sur le cloud pour les aider à obtenir le niveau d'évolutivité nécessaire pour répondre à leurs besoins informatiques à la demande.

Présentation de l'hébergement Web traditionnel

L'hébergement Web évolutif étant confronté à des problèmes d'espace bien connus, l'illustration ci-après qui représente un modèle d'hébergement Web traditionnel ne contient en principe aucune surprise. Nous le présentons à des fins de comparaison avec une architecture similaire mise en œuvre dans le cloud. En effet, c'est le déploiement du cloud qui constitue le sujet de ce document.

Pare-feu extérieur
Solution matérielle ou logicielle pour ouvrir des ports standard (80, 443)

Équilibreur de charge Web
Solution matérielle ou logicielle permettant de répartir le trafic sur différents serveurs Web

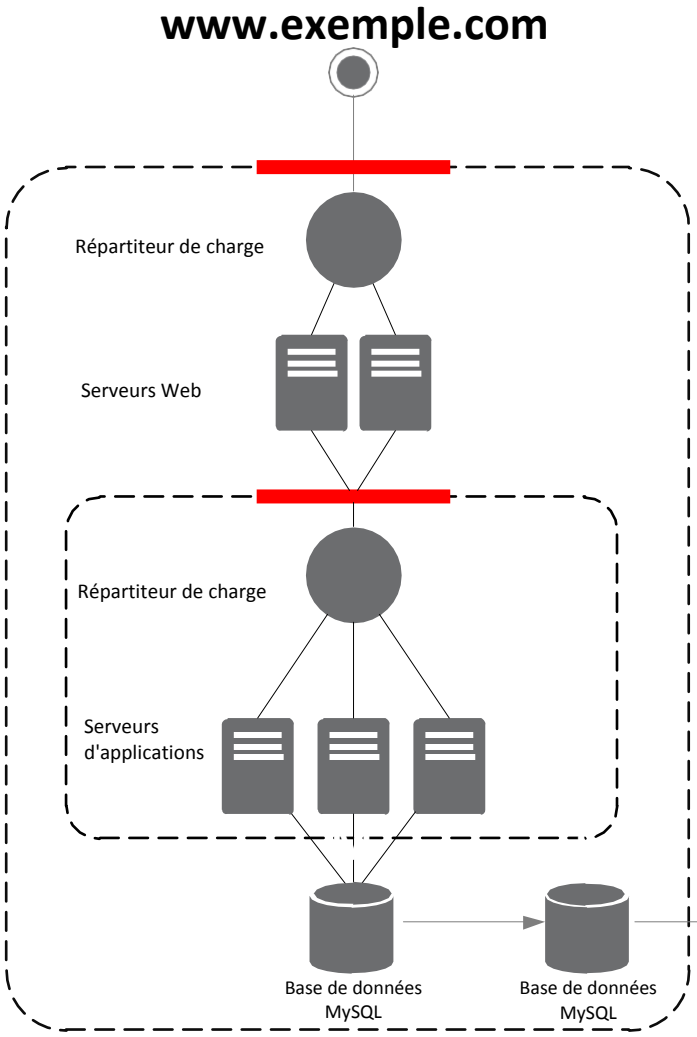
Couche Web
Flotte de machines gérant des requêtes HTTP

Pare-feu principal limitant l'accès à l'application liée à partir de la couche Web

Équilibreur de charge d'applications
Solution matérielle ou logicielle permettant de répartir le trafic sur différents serveurs d'applications

Couche serveur d'applications
Flotte de machines gérant des charges de travail spécifiques aux applications. Les machines du serveur de mise en cache peuvent être implémentées au niveau de cette couche

Couche données
Machines du serveur de base de données avec une base de données principale et une base de données locale s'exécutant séparément, stockage réseau pour les objets statiques



Sauvegardes sur bandes
Sauvegardes périodiques stockées sur bandes et généralement gérées par des tiers sur leur propre site

Illustration 1 Architecture d'applications Web traditionnelle

Cette architecture traditionnelle d'hébergement Web implémente un modèle commun d'application Web à trois niveaux qui sépare l'architecture en couche de présentation, couche d'application et couche de persistance. L'évolutivité est fournie en ajoutant des hôtes à l'une des couches présentation, persistance ou application. L'architecture possède aussi des fonctions de disponibilité, de basculement et de performance intégrées. Les sections suivantes décrivent les raisons pour lesquelles une telle architecture devrait et pourrait être déployée dans le cloud Amazon Web Services ainsi que la manière dont ce déploiement peut être réalisé.

Hébergement d'applications Web dans le cloud à l'aide d'Amazon Web Services

L'architecture traditionnelle d'hébergement Web (illustration 1) peut être facilement transférée vers les services de cloud fournis par les produits AWS uniquement avec un nombre limité de modifications, mais la première question à poser concerne la valeur du déplacement d'une solution classique d'hébergement d'applications Web vers le cloud AWS. Si vous décidez que le cloud vous convient, vous aurez besoin d'une architecture adaptée. Cette section vous aide à évaluer une solution de cloud AWS. Elle compare le déploiement de votre application Web dans le cloud à un déploiement sur site, présente une architecture de cloud AWS pour héberger votre application et décrit les principaux composants de cette solution.

Comment AWS peut résoudre des problèmes communs d'hébergement d'applications Web ?

Si vous êtes responsable de l'exécution d'une application Web, vous êtes confronté à différents problèmes d'architecture et d'infrastructure pour lesquels AWS peut fournir des solutions transparentes et économiques. Vous trouverez ci-après quelques avantages parmi d'autres qu'offre AWS par rapport à un modèle d'hébergement traditionnel :

Alternative économique aux flottes surdimensionnées devant gérer des pics

Dans le modèle d'hébergement traditionnel, des serveurs doivent être mis en service pour gérer la capacité des pics, mais en dehors de ces périodes de pic, des cycles inutilisés sont gaspillés. Les applications Web hébergées par AWS peuvent tirer parti de la mise en service à la demande de serveurs supplémentaires, ce qui permet d'ajuster constamment la capacité et les coûts aux modèles de trafic réels.

Le graphique ci-après, par exemple, illustre une application Web avec un pic d'utilisation entre 9 heures et 15 heures et une utilisation minimale durant le reste de la journée. Une approche de type dimensionnement automatique basée sur des tendances de trafic réelles qui ne met des ressources en service qu'en cas de nécessité, réduit le gaspillage de capacité et diminue les coûts de plus de 50 %.

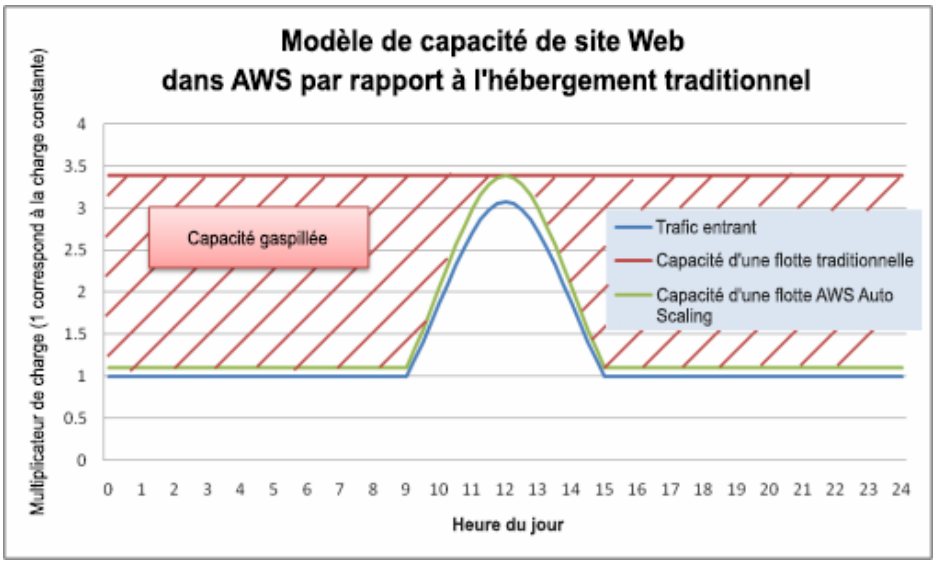


Illustration 2 - Exemple de gaspillage de capacité dans un modèle d'hébergement classique

Solution évolutive de gestion des pics de trafic imprévus

Une conséquence encore plus grave de la lenteur de mise en service qui caractérise le modèle d'hébergement traditionnel est l'incapacité à réagir à temps aux pics de trafic imprévus. Il existe de nombreux cas de blocages d'applications Web pour cause de pic de trafic imprévu une fois que l'existence du site a été annoncée dans les médias populaires. Cette même capacité à la demande qui aide à adapter l'évolutivité des applications Web aux pics de trafic réguliers peut aussi gérer une charge imprévue. Des nouveaux hôtes peuvent être lancés et être opérationnels en quelques minutes, puis être mis hors service dès que le trafic redevient normal.

Solution à la demande pour environnements de test, de charge, bêta et de préproduction

Les coûts de matériel liés à la création d'un environnement d'hébergement traditionnel pour une application Web de production ne se limitent pas à la flotte de production. Il faut bien souvent aussi créer des flottes de préproduction, bêta et de test pour assurer la qualité de l'application Web à chaque étape du cycle de développement. Bien que diverses optimisations puissent être réalisées pour assurer la plus haute utilisation possible de ce matériel de test, ces flottes parallèles ne sont pas toujours exploitées de manière optimale : beaucoup de matériels onéreux restent inactifs pendant de longues périodes. Dans le cloud AWS, vous pouvez mettre en service des flottes de test seulement lorsque vous en avez besoin. Vous pouvez même simuler le trafic utilisateur sur le cloud AWS pendant le test de la charge. Vous pouvez également utiliser ces flottes en tant qu'environnement intermédiaire pour une nouvelle version de produit, ce qui permet un basculement rapide de la version de production actuelle vers une nouvelle version de l'application avec peu ou pas d'indisponibilité de service.

Architecture de cloud AWS pour hébergement Web

Vous trouverez ci-dessous une autre représentation de l'architecture d'application Web classique, dans laquelle nous examinons la manière dont elle peut tirer parti de l'infrastructure du cloud computing AWS :

Route 53
Fournit des services DNS pour simplifier la gestion de domaine et la prise en charge des zones APEX (http://example.com)

Elastic Load Balancer
ELB pour répartir le trafic vers des groupes Auto Scaling de serveur Web

Pare-feu extérieur déplacé vers chaque instance du serveur Web via des groupes de sécurité
Couche Web Auto Scaling
Groupe d'instances EC2 gérant des requêtes HTTP.
Pare-feu principal déplacé vers chaque instance principale

Équilibreur de charge du serveur d'applications
Équilibreur de charge logiciel (par exemple, HAProxy) sur des instances EC2 pour répartir le trafic sur le cluster de serveurs d'applications

Couche application Auto Scaling
Groupe d'instances EC2 exécutant l'application réelle. Les instances appartiennent au groupe Auto Scaling

ElastiCache
Fournit des services de mise en cache pour les applications, en supprimant la charge depuis la couche base de données

Couche Base de données
La base de données MySQL RDS crée une architecture à haut niveau de disponibilité avec des déploiements multi-AZ. Des réplicas en lecture seule peuvent aussi être utilisés pour dimensionner des applications de lecture intensive

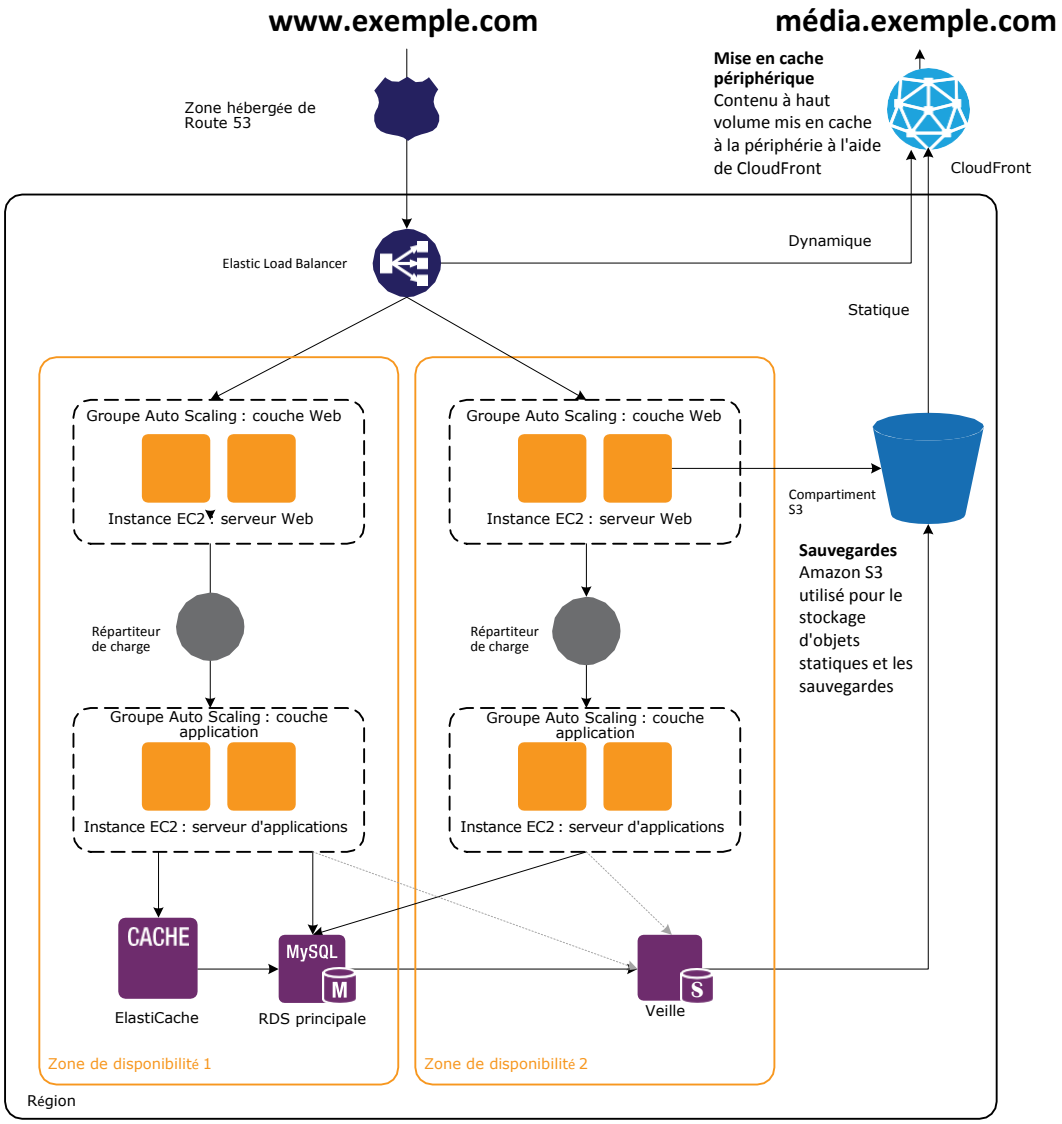


Illustration 3 - Exemple d'architecture d'hébergement Web sur AWS

Principaux composants d'une architecture d'hébergement Web AWS

Les sections suivantes décrivent certains des principaux composants d'une architecture d'hébergement Web déployée dans le cloud AWS et décrivent de quelle manière ils se distinguent d'une architecture d'hébergement Web traditionnelle.

Diffusion de contenu

La mise en cache en périphérie s'avère également pratique dans l'infrastructure de cloud computing d'Amazon Web Service. Toutes les solutions utilisées dans votre infrastructure d'application Web fonctionnent en principe aussi parfaitement dans le cloud AWS. Néanmoins, l'utilisation d'AWS vous offre en outre la possibilité de profiter du service Amazon CloudFront[1] pour la mise en cache à la périphérie de votre site Web.

Amazon CloudFront permet de diffuser l'intégralité de votre site Web (y compris les contenus dynamiques, statiques et en streaming) à partir d'un réseau mondial d'emplacements périphériques. Les requêtes ciblant votre contenu sont acheminées automatiquement vers l'emplacement périphérique le plus proche, de sorte que le contenu puisse être diffusé de manière optimale. Amazon CloudFront est optimisé pour fonctionner avec d'autres Amazon Web Services, tels qu'Amazon Simple Storage Service[2] (Amazon S3) et Amazon Elastic Compute Cloud[3] (Amazon EC2). Amazon CloudFront fonctionne aussi sans problème avec n'importe quel serveur d'origine tiers sur lequel vous stockez les versions définitives et originales de vos fichiers.

À l'instar d'autres Amazon Web Services, l'utilisation d'Amazon CloudFront ne nécessite aucun contrat ou engagement mensuel ; vous ne payez que pour le volume de contenu réellement traité par le service.

Gestion de DNS public

Pour déplacer une application Web vers le cloud AWS, des modifications doivent être apportées au DNS afin de tirer parti des différentes zones de disponibilité fournies par AWS. Pour vous aider à gérer le routage DNS, AWS fournit Amazon Route 53[4], un service Web DNS à haut niveau de disponibilité et évolutif. Les requêtes concernant votre domaine sont automatiquement acheminées vers le serveur DNS le plus proche et, par conséquent, les réponses sont fournies avec les meilleures performances possibles. Route 53 résout les requêtes pour votre nom de domaine (par exemple, www.exemple.com) dans votre service Elastic Load Balancer, ainsi que votre enregistrement de zone apex (exemple.com).

Sécurité de l'hôte

Contrairement à un modèle d'hébergement Web traditionnel, le filtrage du trafic entrant n'est pas limité à la périphérie du réseau, mais est appliqué au niveau de l'hôte. Le service Amazon Elastic Compute Cloud (EC2) propose une fonctionnalité appelée *groupes de sécurité*. Un groupe de sécurité est analogue à un pare-feu réseau entrant, pour lequel vous spécifiez les protocoles, les ports et les plages d'adresses IP source qui permettent d'atteindre vos instances EC2. Chaque instance EC2 peut se voir assigner un ou plusieurs groupes de sécurité, où chacun achemine le trafic approprié vers chaque l'instance. Les groupes de sécurité peuvent être configurés de manière à ce que seuls des sous-réseaux ou des adresses IP spécifiques puissent accéder à une instance EC2 ou ils peuvent référencer d'autres groupes de sécurité afin de limiter l'accès aux instances EC2 appartenant à des groupes spécifiques.

---

1 Amazon CloudFront – http://aws.amazon.com/cloudfront/
2 Amazon S3 – http://aws.amazon.com/s3
3 Amazon EC2 – http://aws.amazon.com/ec2/
4 Amazon Route 53 – http://aws.amazon.com/route53/

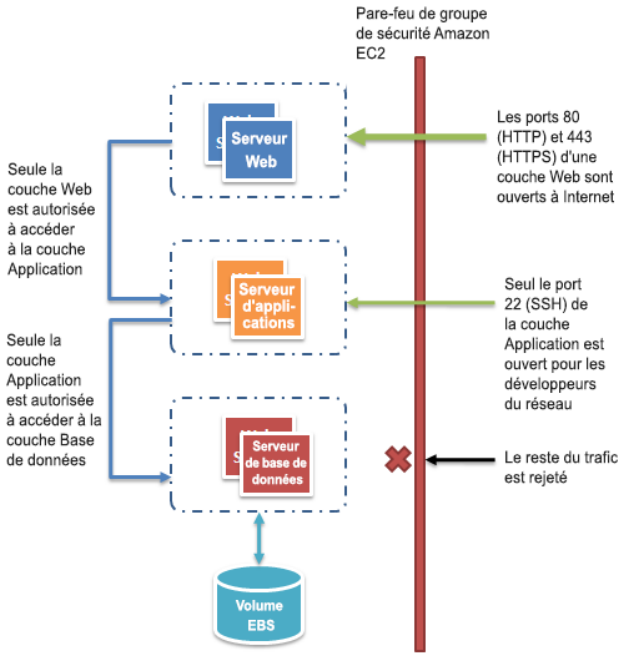


Illustration 4 : groupes de sécurité dans une application Web

Dans le cas de l'exemple d'architecture d'hébergement Web AWS ci-dessus, le groupe de sécurité pour le cluster de serveurs Web peut restreindre l'accès à tous les hôtes via TCP sur les ports 80 et 443 (HTTP et HTTPS) et depuis les instances du groupe de sécurité du serveur d'application sur le port 22 (SSH) pour une gestion directe de l'hôte. En revanche, le groupe de sécurité du cluster du serveur d'applications peut autoriser l'accès du groupe de sécurité du serveur Web afin de gérer les requêtes Web ainsi que depuis le sous-réseau de votre organisation via TCP sur le port 22 (SSH) pour une gestion directe de l'hôte. Dans le modèle illustré ci-dessus, vos ingénieurs de support pourraient se connecter directement aux serveurs d'applications à partir du réseau d'entreprise, puis accéder à d'autres clusters depuis les zones de serveur d'applications. Pour en savoir plus sur la sécurité, visitez le site du Centre de sécurité AWS[5]. Le centre comprend des bulletins de sécurité, des informations de certification et des livres blancs sur la sécurité qui décrivent les capacités d'AWS en termes de sécurité.

Équilibrage de la charge entre les clusters

Les équilibreurs de charge matériels sont des composants habituels des réseaux utilisés dans les architectures d'hébergement Web traditionnelles. AWS offre cette fonctionnalité par le biais de son service Elastic Load Balancing[6], une solution de répartition des charges configurable qui prend en charge la vérification de l'état des hôtes, la distribution du trafic vers les instances EC2 à travers plusieurs zones de disponibilité ainsi que l'ajout et le retrait dynamiques d'hôtes EC2 dans la rotation de répartition de charge. Le service Elastic Load Balancing peut également augmenter et réduire dynamiquement la capacité d'équilibrage de charge pour l'adapter aux demandes de trafic tout en fournissant un point d'entrée prévisible à l'aide d'un CNAME permanent. Le service Elastic Load Balancing prend également en charge les sessions persistantes afin de gérer les besoins de routage plus élaborés. Si votre application nécessite des fonctionnalités d'équilibrage de charge plus avancées, vous pouvez exécuter un package d'équilibrage de charge logicielle (tel que Zeus, HAProxy ou nginx) sur les instances EC2. Vous pouvez ensuite attribuer des adresses IP élastiques[7] à ces instances EC2 d'équilibrage de charge afin de minimiser les modifications apportées au DNS.

[5] Centre de sécurité AWS – http://aws.amazon.com/security

[6] Amazon Elastic Load Balancing – http://aws.amazon.com/elasticloadbalancing

Recherche d'autres hôtes et services

Dans l'architecture d'hébergement Web traditionnelle, la plupart de vos hôtes possède des adresses IP statiques tandis que dans le cloud, leurs adresses IP sont généralement dynamiques. Même si chaque instance EC2 peut avoir des entrées DNS publiques et privées et qu'elle peut être adressée à partir d'Internet, les entrées DNS et les adresses IP sont attribuées dynamiquement au lancement de l'instance et ne peuvent pas être attribuées manuellement. Des adresses IP statiques (adresses IP Elastic selon la terminologie AWS) peuvent être attribuées à des instances en cours d'exécution après leur lancement. Vous devez utiliser des adresses IP Elastic pour les instances et les services qui nécessitent des points de terminaison fiables, comme les bases de données principales, les serveurs de fichiers centraux et les équilibreurs de charge hébergés sur EC2.

Les rôles de serveur susceptibles d'évoluer facilement, tels que les serveurs Web, devraient pouvoir être détectés au niveau de leurs points de terminaison dynamiques en enregistrant leur adresse IP dans un référentiel central. Comme la plupart des architectures d'applications Web possède un serveur de base de données activé en permanence, ce dernier constitue un référentiel commun pour la découverte d'informations. Dans les cas où l'adressage doit être fiable, une adresse IP Elastic provenant d'un groupe d'adresses peut être allouée par un script d'action d'amorçage à des instances au moment de leur lancement.

Via ce modèle, des hôtes nouvellement ajoutés peuvent demander la liste des points de terminaison requis pour assurer la communication à partir de la base de données pendant la phase d'action d'amorçage. L'emplacement de la base de données peut être fourni sous forme de données utilisateur [8] transmises à chaque instance au moment de leur lancement. Vous pouvez aussi utiliser le service Amazon SimpleDB pour stocker et conserver des informations de configuration. Amazon SimpleDB est un service à haute disponibilité qui est accessible sur un point de terminaison bien connu.

Mise en cache dans l'application Web

Les caches d'application en mémoire peuvent réduire la charge des services et améliorer les performances ainsi que l'évolutivité dans la couche base de données en mettant en cache des informations fréquemment utilisées. Amazon ElastiCache[9] est un service Web qui facilite le déploiement, l'utilisation et le dimensionnement d'un cache en mémoire dans le cloud. Le cache en mémoire que vous créez peut être configuré pour évoluer automatiquement en fonction de la charge et pour remplacer automatiquement des nœuds défaillants. Amazon ElastiCache offre une conformité de protocole avec Memcached, ce qui simplifie la migration à partir de votre solution sur site actuelle.

Configuration de base de données, sauvegarde et basculement

La plupart des applications Web contient des formes de persistance, généralement sous la forme d'une base de données relationnelle ou d'une base de données NoSQL. AWS offre à la fois une infrastructure de base de données relationnelle et NoSQL. Vous pouvez également déployer votre propre logiciel de base de données sur une instance Amazon EC2. Le tableau suivant résume ces options, qui sont ensuite décrites en détail.

---

[7] Les adresses IP Elastic sont des adresses IP statiques conçues pour le cloud computing dynamique, qui peuvent être transférées d'une instance à l'autre [8] Données utilisateur – http://docs.amazonwebservices.com/AWSEC2/latest/APIReference/index.html?ApiReference-ItemType- RunInstancesType.html

[9] Amazon ElastiCache – http://aws.amazon.com/elasticache

	Solutions de base de données relationnelle	Solutions NoSQL
Service de base de données gérée	Amazon Relational Database Service (RDS) – MySQL, Oracle, SQL Server	Amazon DynamoDB
Auto-gestion	Hébergement d'un système de gestion de base de données relationnelle sur une instance	Hébergement d'une solution NoSQL sur une instance Amazon EC2

Amazon Relational Database Service (RDS)

Amazon RDS vous donne accès aux capacités d'un moteur de base de données MySQL, Oracle ou Microsoft SQL Server classique. Vous pouvez utiliser avec Amazon RDS du code, des applications et des outils qui vous sont déjà familiers. Amazon RDS applique automatiquement les correctifs logiciels à la base de données, sauvegarde votre base de données et stocke les sauvegardes pendant une période de rétention définie par l'utilisateur. La restauration à un instant dans le passé est également prise en charge. Vous profitez de la flexibilité de pouvoir dimensionner les ressources de calcul ou la capacité de stockage associées à votre instance de base de données relationnelle par un simple appel API.

En outre, les déploiements multi-AZ Amazon RDS augmentent la disponibilité de votre base de données et protègent celle-ci des interruptions imprévues. Les réplicas en lecture d'Amazon RDS fournissent des réplicas en lecture seule de votre base de données qui facilitent le dimensionnement au-delà de la capacité de déploiement d'une seule base de données pour les charges de travail de base de données à lecture intensive. À l'instar de tous les Amazon Web Services, aucun investissement initial n'est requis et vous ne payez que pour les ressources que vous utilisez.

Hébergement d'une base de données relationnelle (RDBMS) sur une instance Amazon EC2

En plus de l'offre Amazon RDS gérée, vous pouvez installer votre choix de SGBDR (notamment MySQL, Oracle, SQL Server ou DB2) sur une instance EC2 et le gérer vous-même. Les clients AWS qui hébergent une base de données sur Amazon EC2 exploitent avec succès une variété de modèles maître/esclave et de réplication, avec mise en miroir de copies en lecture seule et expédition de journaux pour des esclaves passifs toujours prêts.

Lorsque vous gérez vous-même votre logiciel de base de données directement sur Amazon EC2, vous devez également prendre en compte la disponibilité du stockage tolérant aux pannes et du stockage permanent. À cette fin, nous recommandons que les bases de données s'exécutant sur Amazon EC2 utilisent des volumes Amazon Elastic Block Storage (Amazon EBS), qui sont semblables à un stockage NAS (Network Attached Storage). Pour les instances EC2 exécutant une base de données, toutes les données de base de données et tous les journaux doivent être placés sur des volumes Amazon EBS, qui demeurent disponibles même en cas de défaillance de l'hôte de la base de données. Cette configuration permet un scénario de basculement simple, qui lance une nouvelle instance EC2 en cas de défaillance de l'hôte et attache les volumes Amazon EBS existants à cette nouvelle instance. La base de données peut ensuite être récupérée là où elle a été arrêtée.

Les volumes Amazon EBS fournissent la redondance à l'intérieur de la zone de disponibilité, dont la disponibilité est supérieure à celle de simples disques. Si les performances d'un seul volume Amazon EBS ne suffisent pas aux besoins de votre base de données, des volumes peuvent être agrégés par bandes afin d'accroître les performances d'IOPS pour votre base de données. Dans le cas des charges de travail exigeantes, vous pouvez aussi utiliser des IOPS provisionnées EBS, où vous spécifiez l'IOPS requise. Si vous utilisez Amazon RDS, le service gère son propre stockage et vous pouvez vous concentrer sur la gestion de vos propres données.

Solutions NoSQL

Outre la prise en charge de bases de données relationnelles, AWS propose aussi Amazon DynamoDB, un service de base de données NoSQL entièrement géré, offrant des performances exceptionnelles et prévisibles en termes de rapidité et d'évolutivité. À l'aide d'AWS Management Console ou de l'API Amazon DynamoDB, vous pouvez augmenter ou diminuer les capacités, en évitant les temps d'arrêt ou la dégradation des performances. Comme Amazon DynamoDB gère les charges administratives liées au fonctionnement et au dimensionnement des bases de données distribuées vers AWS, vous n'avez plus à vous soucier de la mise en service, du paramétrage, de la configuration et de la réplication du matériel, des correctifs logiciels ou du dimensionnement des clusters.

Amazon SimpleDB fournit un service de base de données non relationnelle léger, tolérant aux pannes et à haute disponibilité avec des possibilités d'interrogation et d'indexation de données sans nécessiter de schéma fixe. SimpleDB peut constituer une solution de remplacement des bases de données très efficace dans les scénarios d'accès aux données qui nécessitent une grande table de schéma flexible à haut niveau d'indexation.

En outre, Amazon EC2 peut aussi héberger de nombreuses autres technologies émergentes du mouvement NoSQL, comme Cassandra, CouchDB et MemcacheDB.

Stockage et sauvegarde des données et des ressources

Le cloud AWS fournit de nombreuses possibilités en termes de stockage, d'accès et de sauvegarde de vos données d'applications Web et ressources. Amazon Simple Storage Service (Amazon S3) fournit un magasin d'objets à redondance et haut niveau de disponibilité. Amazon S3 constitue une solution de stockage idéale pour des objets relativement statiques ou à évolution lente, tels que les images, les vidéos et autres médias statiques. Amazon S3 prend aussi en charge la mise en cache à la périphérie et le streaming de ces ressources via des interactions avec le service Amazon CloudFront.

Pour le système de fichiers attaché tel que le stockage, il est possible d'attacher à des instances EC2 des volumes Amazon Elastic Block Storage, qui peuvent se comporter comme des disques à monter pour exécuter des instances EC2. Amazon EBS est parfait pour des données qui doivent être accessibles sous forme de stockage de bloc et qui doivent perdurer au-delà de l'exécution en cours de l'instance, comme les partitions de base de données et les journaux d'application.

Outre la durée de vie indépendante de l'instance EC2, des instantanés des volumes Amazon EBS peuvent être capturés et stockés dans Amazon S3. Comme les instantanés EBS ne sauvegardent que les modifications ultérieures à l'instantané précédent, des instantanés plus fréquents peuvent réduire leurs durées. Vous pouvez aussi utiliser un instantané Amazon EBS comme base pour répliquer des données entre plusieurs volumes Amazon EBS et attacher ces volumes à d'autres instances en cours d'exécution.

Les volumes Amazon EBS peuvent atteindre jusqu'à 1 To et plusieurs volumes Amazon EBS peuvent être agrégés par bandes pour des volumes encore plus grands ou pour accroître les performances d'E/S. Pour maximiser les performances de vos applications gourmandes en E/S, vous pouvez utiliser des volumes d'IOPS provisionnées. Les volumes d'IOPS provisionnées sont conçus pour satisfaire les besoins des charges de travail gourmandes en E/S, notamment les charges de travail de base de données qui sont sensibles aux performances de stockage et à l'homogénéité du débit d'E/S à accès aléatoire. Vous spécifiez la vitesse des opérations d'E/S par seconde lorsque vous créez le volume et Amazon EBS alloue ce débit pour la durée de vie du volume. Amazon EBS prend actuellement en charge jusqu'à 1 000 IOPS par volume. Vous pouvez agréger plusieurs volumes entre eux pour fournir des milliers d'IOPS par instance à votre application.

Auto Scaling de la flotte

Une des principales différences entre l'architecture du cloud AWS et le modèle d'hébergement traditionnel réside dans le fait qu'AWS peut dimensionner dynamiquement la flotte d'applications Web à la demande, pour gérer les variations de trafic. Le modèle d'hébergement traditionnel utilise généralement des modèles de prévision de trafic pour dimensionner anticipativement des hôtes en fonction du trafic prévu. Dans AWS, des instances peuvent être provisionnées à la volée en fonction d'un ensemble de déclencheurs pour augmenter ou diminuer le dimensionnement de la flotte. Amazon Auto Scaling peut créer des groupes de serveurs dont la capacité augmente ou diminue selon la demande. Auto Scaling fonctionne aussi directement avec Amazon CloudWatch pour les indicateurs et avec le service Elastic Load Balancing pour l'ajout ou la suppression d'hôtes afin de distribuer la charge. Par exemple, si les serveurs Web indiquent que l'utilisation de l'UC est supérieure à 80 % sur une période donnée, un serveur Web supplémentaire peut rapidement être déployé, puis être ajouté automatiquement au service Elastic Load Balancer pour être immédiatement intégré dans la rotation d'équilibrage de charge.

Comme illustré dans le modèle d'architecture d'hébergement Web AWS, plusieurs groupes Auto Scaling peuvent être créés pour différentes couches de l'architecture afin de permettre à chaque couche d'évoluer indépendamment. Par exemple, le groupe Auto Scaling du serveur Web peut déclencher l'évolution, en réponse à des modifications au niveau des E/S du réseau, tandis que le groupe Auto Scaling du serveur d'applications peut évoluer en fonction de l'utilisation de l'UC. Vous pouvez définir des minima et des maxima pour assurer une disponibilité permanente (24 heures sur 24, 7 jours sur 7) et pour limiter l'utilisation au sein d'un groupe.

Les déclencheurs Auto Scaling peuvent être définis pour accroître et réduire la flotte totale d'une couche donnée afin d'adapter l'utilisation des ressources à la demande réelle. En plus du service Auto Scaling, les flottes EC2 peuvent facilement être mises à l'échelle directement via l'API EC2, ce qui permet d'exécuter, d'arrêter et d'inspecter des instances.

Basculement avec AWS

Les zones de disponibilité, qui facilitent l'accès à des emplacements de déploiement redondants, constituent un autre atout d'AWS par rapport à l'hébergement Web traditionnel. Les zones de disponibilité sont des emplacements distincts physiquement, conçues pour être isolées des défaillances dans d'autres zones de disponibilité. Elles fournissent une connectivité réseau économique à faible latence à d'autres zones de disponibilité de la même région. Comme le montre le schéma de l'architecture d'hébergement Web d'AWS dans ce document, nous vous recommandons de déployer des hôtes EC2 sur plusieurs zones de disponibilité afin d'accroître la tolérance aux pannes de votre application Web. Il est impératif de veiller à ce que la migration des points d'accès uniques sur différentes zones de disponibilité soit possible en cas de défaillance. Par exemple, une base de données subordonnée doit être configurée dans une seconde zone de disponibilité afin de préserver la persistance et la haute disponibilité des données, même dans l'éventualité peu probable où une défaillance surviendrait.

Même si certaines modifications architecturales sont nécessaires pour déplacer une application Web existante vers le cloud AWS, les améliorations significatives en termes d'évolutivité, de fiabilité et de rentabilité que génère l'utilisation d'AWS en valent bien la peine. Ce sont ces améliorations que nous allons aborder dans la section suivante.

Principaux avantages de l'hébergement Web via AWS

Il existe plusieurs différences clés entre le modèle d'hébergement d'applications Web traditionnel et celui du cloud AWS. La section précédente a mis en évidence la plupart des éléments clé à prendre en compte pour déployer une application Web dans le cloud. La présente section décrit certaines modifications architecturales essentielles à appliquer pour déplacer une application quelconque dans le cloud.

Finis les composants de réseau physiques

Vous ne pouvez pas déployer des composants de réseau physiques dans AWS. Par exemple, les pare-feux, routeurs et équilibreurs de charge pour vos applications AWS ne peuvent plus résider sur des composants physiques, mais doivent être remplacés par des solutions logicielles. Il existe un large éventail de solutions logicielles professionnelles pour équilibrer les charges (notamment Zeus, HAProxy, nginx et Pound) ou établir une connexion VPN (par exemple, OpenVPN, OpenSwan et Vyatta). Cette différence ne limite en rien ce qui peut être exécuté sur le cloud AWS, mais constitue un changement architectural de votre application si vous utilisez actuellement de tels périphériques.

Pare-feux omniprésents

Là où vous n'aviez auparavant qu'une simple DMZ avant d'ouvrir des communications entre vos hôtes dans un modèle d'hébergement traditionnel, AWS applique un modèle plus sûr dans lequel chaque hôte est verrouillé. Une des étapes de la planification d'un déploiement AWS consiste à analyser le trafic entre les hôtes. Cette analyse oriente les décisions à prendre au sujet de ce que les ports doivent précisément ouvrir. Des groupes de sécurité peuvent être créés au sein d'Amazon EC2 pour chaque type d'hôte de votre architecture et un large éventail de modèles de sécurité simples et à niveaux peut être prévu pour garantir un accès minimum entre les hôtes dans votre architecture.

Multiples centres de données disponibles

Dans une région AWS, les zones de disponibilité doivent être considérées comme des centres de données multiples. Les instances EC2 des différentes zones de disponibilité sont séparées tant au niveau logique que physique et fournissent un modèle facile à utiliser pour déployer votre application dans des centres de données tout en bénéficiant d'un haut niveau de disponibilité et de fiabilité.

Hôtes éphémères et dynamiques

Le plus gros changement dans la conception architecturale de votre application AWS réside probablement dans le fait que les hôtes Amazon EC2 doivent être considérés comme éphémères et dynamiques. Toute application destinée au cloud AWS ne doit pas supposer qu'un hôte sera disponible en permanence et doit être conçue en tenant compte du fait que toutes les données qui ne figurent pas dans un volume Amazon EBS seront perdues en cas de défaillance d'une instance EC2. De plus, lorsqu'un nouvel hôte est mis à disposition, aucune hypothèse ne doit être faite au sujet de son adresse IP ou de son emplacement au sein d'une zone de disponibilité. Votre modèle de configuration doit être souple et votre approche à propos de l'action d'amorçage d'un hôte doit tenir compte de la nature dynamique du cloud. Ces techniques sont essentielles pour concevoir et exécuter une application à haut niveau d'évolutivité et de tolérance aux pannes.

Réflexions finales

Si la migration d'une application Web vers le cloud AWS nécessite la prise en compte de nombreux aspects conceptuels et architecturaux, les avantages générés par une infrastructure rentable, tolérante aux pannes et qui évolue au rythme de votre activité dépassent de loin les efforts à consentir.

Autres ressources de mise en route

1. Guide de démarrage – Hébergement d'applications Web AWS pour Linux
   http://docs.amazonwebservices.com/gettingstarted/latest/wah-linux/
2. Guide de démarrage – Hébergement d'applications Web AWS pour Windows
   http://docs.amazonwebservices.com/gettingstarted/latest/wah/
3. Séries vidéos de démarrage – Applications Web Linux dans le cloud AWS
   http://aws.amazon.com/web-applications/gsg-webapps-linux/
4. Séries vidéos de démarrage – Applications Web .NET dans le cloud AWS
   http://aws.amazon.com/web-applications/gsg-webapps-windows/

Modifications depuis la dernière version (mai 2010)

- Mise à jour de plusieurs sections pour plus de clarté
- Mise à jour de schémas pour l'utilisation des icônes AWS
- Ajout de la section Gestion de DNS public pour Route 53
- Mise à jour de la section relative à la recherche d'autres hôtes et services pour plus de clarté
- Mise à jour de la section Configuration de base de données, sauvegarde et basculement pour DynamoDB et pour plus de clarté
- Extension de la section Stockage et sauvegarde des données et des ressources pour couvrir les volumes d'IOPS provisionnées d'EBS