Alojamiento de aplicación web en la nube de AWS

Septiembre de 2017

Avisos

Este documento se suministra únicamente con fines informativos. Representa la oferta actual de productos y prácticas de AWS a partir de la fecha de publicación de este documento. Dichas prácticas y productos pueden modificarse sin previo aviso. Los clientes son responsables de realizar sus propias evaluaciones independientes de la información contenida en este documento y de cualquier uso de los productos o servicios de AWS, cada uno de los cuales se ofrece “tal cual”, sin garantía de ningún tipo, ya sea explícita o implícita. Este documento no genera ninguna garantía, declaración, compromiso contractual, condición ni certeza por parte de AWS, sus filiales, proveedores o licenciantes. Las responsabilidades y obligaciones de AWS con respecto a sus clientes se controlan mediante los acuerdos de AWS y este documento no forma parte ni modifica ningún acuerdo entre AWS y sus clientes.

Contenido

Resumen

El alojamiento web de alta disponibilidad y escalable puede ser una propuesta compleja y costosa. Las arquitecturas web escalables tradicionales, no solo son necesarias para implementar soluciones complejas que garanticen altos niveles de fiabilidad, también requieren un pronóstico preciso del tráfico para ofrecer al cliente un servicio de calidad. Periodos pico de tráfico denso y oscilaciones imprevisibles en los patrones de tráfico dar lugar a bajos porcentajes de uso de costoso hardware. Esto se traduce en un gran costo de funcionamiento para mantener el hardware inactivo y un uso poco eficiente del capital del hardware infrautilizado.

Amazon Web Services (AWS) proporciona una infraestructura fiable, escalable, segura y de gran desempeño para las aplicaciones web más exigentes. Esta infraestructura iguala los costos de TI con los patronos de tráfico de los clientes en tiempo real.

Este documento técnico ha sido diseñado para los arquitectos de sistema y administradores de TI que consultan a la nube para que les ayude a alcanzar la escalabilidad para satisfacer sus necesidades computacionales bajo demanda.

Información general sobre el alojamiento web tradicional

El alojamiento web escalable es tradicionalmente un asunto problemático. La figura 1 muestra una arquitectura tradicional de alojamiento web que implementa un modelo común de aplicación web de tres capas. En este modelo, la arquitectura se divide en capas de presentación, aplicación y persistencia. La escalabilidad se proporciona al añadir hosts en estas capas. La arquitectura también cuenta con características de desempeño, conmutación por error y disponibilidad integradas. La arquitectura de alojamiento web tradicional es fácilmente transferida a la nube de AWS con solo algunas modificaciones.

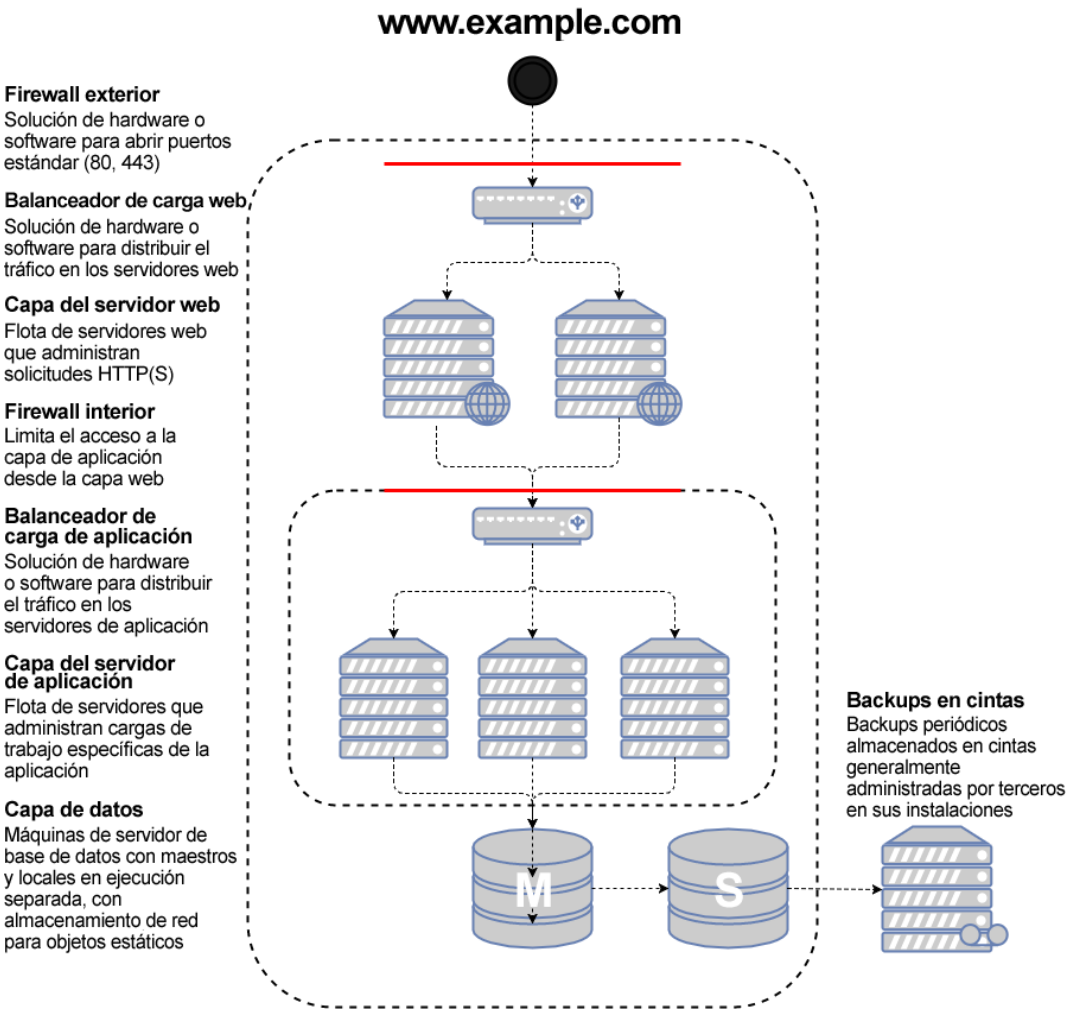

Figura 1. Una arquitectura de alojamiento web tradicional

En las secciones siguientes, analizamos por qué y cómo esta arquitectura debe y puede implementarse en la nube de AWS.

Alojamiento de aplicación web en la nube con AWS

La primera pregunta que debería hacer es cuál es el valor de mover a la nube de AWS una solución de alojamiento de aplicación web clásica. Si decide que la nube es correcta para sus necesidades, necesitará una arquitectura adecuada. Esta sección le ayuda a evaluar una solución en la nube de AWS. Compara la implementación de su aplicación web en la nube con una implementación en las instalaciones físicas, presenta una arquitectura de la nube de AWS para alojar su aplicación y comenta los componentes clave de esta solución.

¿Cómo puede AWS solucionar problemas comunes de alojamiento de aplicación web?

Si es la primera vez que se encarga de ejecutar una aplicación web, se enfrenta a una variedad de problemas de infraestructura y arquitectura para los cuales AWS puede proporcionar soluciones óptimas y rentables. A continuación, se indican algunas de las ventajas de utilizar AWS, frente a un modelo de alojamiento tradicional.

Una alternativa rentable que las grandes flotas necesitaban para administrar los picos de demanda

En el modelo de alojamiento tradicional, tiene que aprovisionar servidores para administrar la capacidad en picos de demanda. Los ciclos que no se utilizan se pierden fuera de los periodos de máxima actividad. Las aplicaciones web alojadas en AWS pueden aprovechar el aprovisionamiento bajo demanda de servidores adicionales, por lo que puede ajustar constantemente la capacidad y los costos a los patrones de tráfico reales.

Por ejemplo, el siguiente gráfico muestra una aplicación web con un uso máximo de 9:00 h a 15:00 h, y menos uso en el resto del día. Un enfoque de escalado automático basado en las tendencias de tráfico reales, que aprovisiona los recursos solo cuando es necesario, se traduciría en un menor desperdicio de capacidad y una reducción en el costo superior al 50 por ciento.

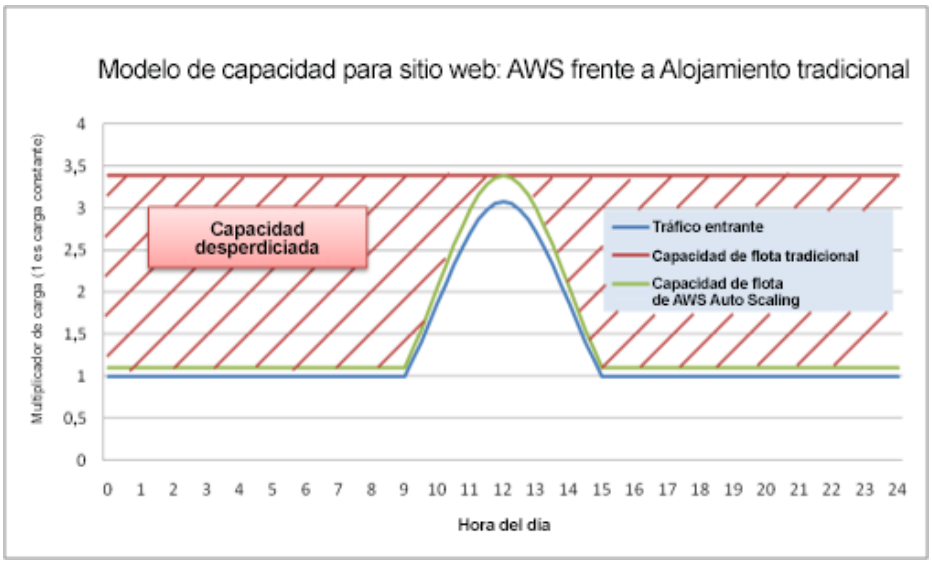

Figura 2. Un ejemplo de capacidad desperdiciada en un modelo de alojamiento clásico

Una solución escalable para la administración de picos de tráfico inesperados

Una consecuencia aún más grave del aprovisionamiento lento típico de los modelos de alojamiento tradicionales es la incapacidad para responder a tiempo a los picos de tráfico inesperados. Existen muchas historias sobre aplicaciones web cuya capacidad se desbordar debido a un pico de tráfico inesperado cuando el sitio en cuestión se menciona en medios de comunicación de masas. La misma capacidad bajo demanda que ayuda a aplicaciones web a escalar para regular los picos de tráfico normal, también puede utilizarse con una carga inesperada. Pueden prepararse y lanzarse nuevos hosts en cuestión de minutos, y desconectarlos igual de rápido cuando el tráfico vuelva a ser normal.

Una solución bajo demanda para entornos de prueba, carga, beta y preproducción

Los costos de hardware de crear un entorno de alojamiento tradicional para una aplicación web de producción no se detienen con la flota de producción. A menudo debe crear flotas de prueba, beta y preproducción para garantizar la calidad de la aplicación web en cada fase del ciclo de desarrollo. Aunque puede realizar varias optimizaciones para garantizar el mayor uso posible de este hardware de prueba, estas flotas paralelas no siempre se utilizan de forma óptima: mucho hardware costoso queda sin uso durante largos períodos de tiempo.

En la nube de AWS puede aprovisionar flotas de prueba según sea necesario. Además, puede simular el tráfico de usuario en la nube de AWS durante las pruebas de carga. También puede utilizar estas flotas en paralelo como un entorno de ensayo para un nuevo lanzamiento de producción. Esto permite un cambio rápido de la producción actual a una nueva versión de la aplicación con poca o ninguna interrupción del servicio.

Una arquitectura de la nube de AWS para el alojamiento web

La siguiente figura ofrece otra perspectiva de una arquitectura clásica de aplicación web y de qué forma puede aprovechar la infraestructura de computación en la nube de AWS.

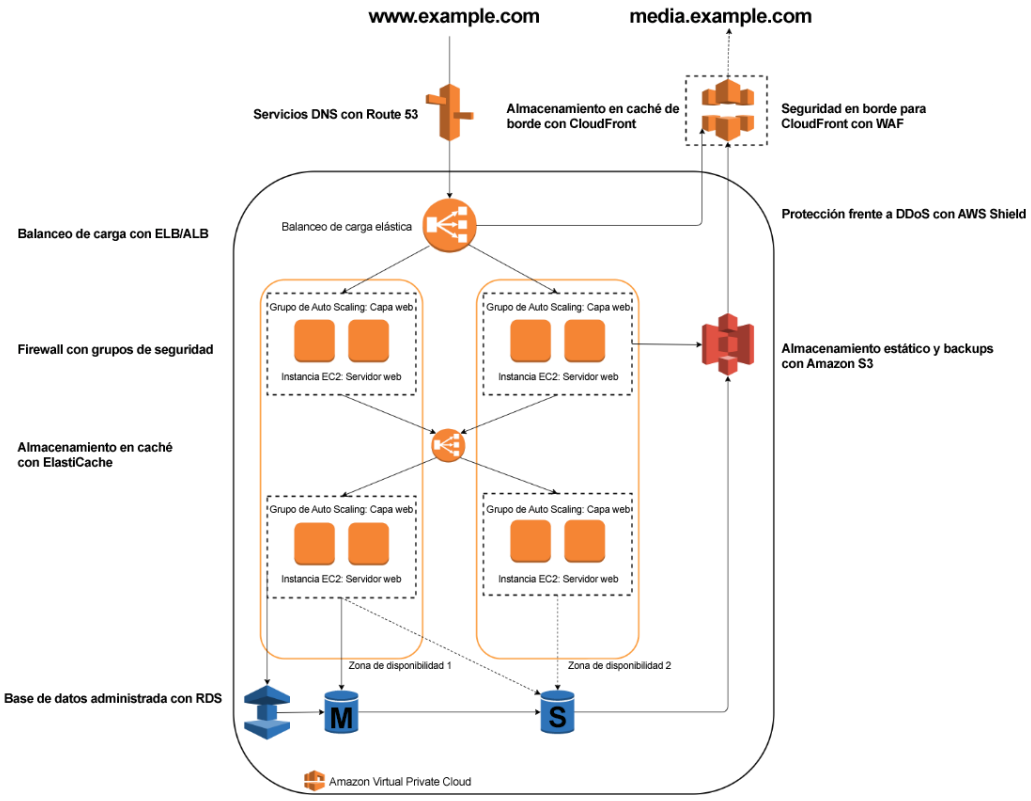

Figura 3. Un ejemplo de una arquitectura de alojamiento web en AWS

1. **Balanceo de carga con Elastic Load Balancing (ELB)/Application Load Balancer (ALB)**: le permite distribuir la carga en varias zonas de disponibilidad y grupos de Auto Scaling Amazon EC2 para redundancia y desacoplamiento de servicios.

2. **Firewalls con grupos de seguridad**: traslada seguridad al nivel de la instancia, para proporcionar un firewall a nivel de host y con estado, para los servidores de aplicaciones y web.
3. **Almacenamiento en caché con Amazon ElastiCache**: proporciona servicios de almacenamiento en caché con Redis o Memcached para eliminar carga desde la aplicación y la base de datos, y reducir la latencia para solicitudes frecuentes.
4. **Bases de datos administradas con Amazon RDS**: crea una arquitectura de base de datos de alta disponibilidad, para varias zonas de disponibilidad y con seis posibles motores de base de datos.
5. **Servicios DNS con Amazon Route 53**: proporciona servicios de DNS para simplificar la administración de dominios.
6. **Caché de borde con Amazon CloudFront**: almacena contenido de gran volumen en la caché de borde, para reducir la latencia cara a los clientes.
7. **Seguridad de borde para Amazon CloudFront con AWS WAF**: filtra el tráfico malicioso, incluida la inyección de XSS y SQL, mediante reglas definidas por el usuario.
8. **Protección DDoS con AWS Shield**: protege de forma automática su infraestructura frente a los ataques DDoS de capa de transporte y red más comunes.
9. **Almacenamiento estático y backups con Amazon S3**: habilita el almacenamiento de objetos basado en HTTP simple para los backups y los recursos estáticos como imágenes y vídeo.

Componentes principales de una arquitectura de alojamiento web AWS

En las siguientes secciones se describen algunos de los componentes clave de una arquitectura de alojamiento web implementada en la nube de AWS, y se explican las diferencias frente a una arquitectura de alojamiento web tradicional.

Administración de redes

En un entorno en la nube como AWS, la capacidad para segmentar su red frente a las de otros clientes le permite una arquitectura más escalable y segura.

Aunque los grupos de seguridad proporcionan seguridad a nivel de host (consulte la sección Seguridad de Host), la nube virtual privada Amazon Virtual Private Cloud (Amazon VPC) le permite lanzar recursos en una red virtual y aislada lógicamente, definida por usted.[1]

Amazon VPC es un servicio gratuito que le ofrece control total sobre los detalles de su configuración de red en AWS. Entre otras cosas, este control incluye la creación de subredes de cara al público para los servidores web, así como subredes privadas sin acceso a Internet para sus bases de datos. Además, Amazon VPC le permite crear arquitecturas híbridas mediante el uso de redes privadas virtuales de hardware (VPN), y utilizar la nube de AWS como una extensión de su propio centro de datos.

Amazon VPC también incluye soporte de IPv6, además de soporte de IPv4 tradicional para su red.

Entrega de contenido

El almacenamiento en caché de borde sigue siendo aplicable en la infraestructura de computación en la nube de AWS. Todas las soluciones existentes en su infraestructura de aplicación web deberían funcionar bien en la nube de AWS. Otra opción, sin embargo, es utilizar Amazon CloudFront para el almacenamiento en caché de borde de su sitio web.[2]

Puede usar CloudFront para entregar su sitio web, incluido el contenido dinámico, estático y de streaming, a través de una red global de ubicaciones de borde. CloudFront dirige de forma automática las solicitudes de contenido a la ubicación de borde más cercana para que el contenido se entregue con el mejor desempeño posible. CloudFront está optimizado para trabajar con otros servicios de AWS, como Amazon Simple Storage Service[3] (Amazon S3) y Amazon Elastic Compute Cloud[4] (Amazon EC2). CloudFront también funciona a la perfección con servidores de origen distintos de los de AWS, que almacenen las versiones originales y definitivas de sus archivos.

Al igual que ocurre con otros servicios de AWS, no es necesario firmar ningún contrato ni aceptar ningún compromiso mensual para utilizar CloudFront: pagará exactamente por el contenido que distribuya a través del servicio.

Administración de DNS público

Trasladar una aplicación web a la nube de AWS requiere algunos cambios de DNS para aprovechar las distintas zonas de disponibilidad que ofrece AWS. Para ayudarle a administrar el enrutamiento de DNS, AWS ofrece Amazon Route 53,[5] un servicio web DNS escalable y de alta disponibilidad. Amazon Route 53 enruta automáticamente las consultas de dominio al servidor DNS más cercano. Como resultado, las consultas se responden con el mejor desempeño posible. Amazon Route 53 resuelve las solicitudes de su nombre de dominio (por ejemplo, www.example.com) a su Classic Load Balancer, además de su registro de ápex de zona (example.com).

Seguridad del host

A diferencia de un modelo de alojamiento web tradicional, el filtrado de tráfico de red entrante no debería reducirse al borde; también debería aplicarse al nivel del host. Amazon EC2 proporciona una característica llamada grupos de seguridad. Un grupo de seguridad es análogo a un firewall de red entrante, en el que se pueden especificar los protocolos, los puertos y los rangos IP de origen que están autorizados a obtener acceso a las instancias EC2. Puede asignar uno o varios grupos de seguridad para cada instancia EC2. Cada grupo de seguridad enruta el tráfico adecuado a cada instancia. Los grupos de seguridad se pueden configurar para que solo direcciones IP o subredes específicas tengan acceso a una instancia EC2. O bien pueden hacer referencia a otros grupos de seguridad para limitar el acceso a las instancias EC2 que se encuentran en grupos específicos.

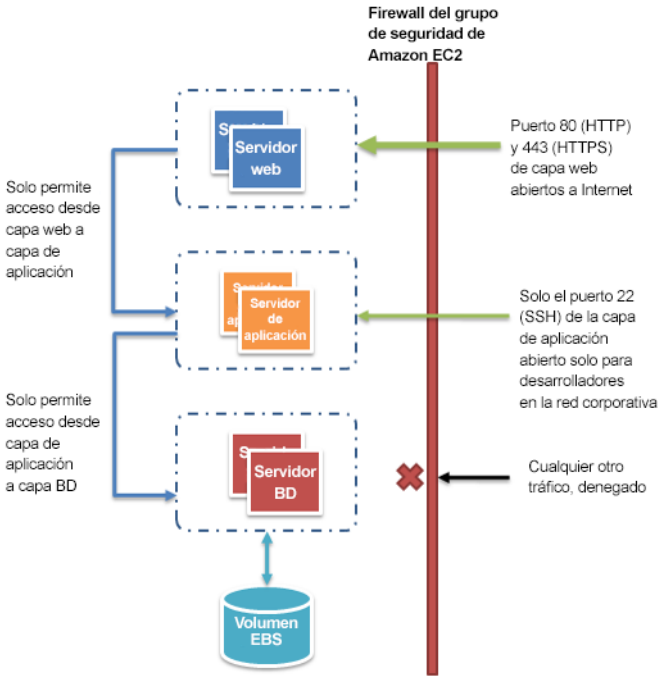

Figura 4. Grupos de seguridad en una aplicación web

En el ejemplo de arquitectura de alojamiento web de AWS de la figura 4, el grupo de seguridad para el clúster de servidor web permitiría el acceso a cualquier host solo a través de TCP en los puertos 80 y 443 (HTTP y HTTPS) y desde las instancias en el grupo de seguridad del servidor de aplicaciones en el puerto 22 (SSH) para administración directa de host. El grupo de seguridad del servidor de aplicaciones, por otro lado, permitiría el acceso desde el grupo de seguridad del servidor web para la administración de las solicitudes web y desde la subred de su organización sobre TCP en el puerto 22 (SSH) para la administración directa de host. En este modelo, los ingenieros de soporte podrían iniciar sesión directamente en los servidores de aplicaciones desde la red corporativa y, a continuación, obtener acceso a los demás clústeres desde los cuadros del servidor de la aplicación. Para un análisis más profundo de la seguridad, visite el Centro de seguridad de AWS.[6] El centro contiene boletines de seguridad, información de certificación, y documentos técnicos de seguridad que explican las capacidades de seguridad de AWS.

Balanceo de carga en clústeres

Los balanceadores de carga por hardware son dispositivos de red comunes que se utilizan en las arquitecturas de aplicaciones web tradicionales. AWS ofrece esta capacidad a través del servicio Elastic Load Balancing (ELB).[7] ELB es una

solución de balanceo de carga configurable que admite comprobaciones de estado en hosts, distribución de tráfico a las instancias EC2 en varias zonas de disponibilidad e incorporación y eliminación dinámicas de hosts de Amazon EC2 desde la rotación de balanceo de cargas. ELB también puede crecer dinámicamente y reducir la capacidad de balanceo de carga para ajustarse a las demandas de tráfico, al mismo tiempo que ofrece un punto de entrada predecible a través de un CNAME persistente. ELB también admite sesiones sticky (persistentes) para abordar las necesidades de enrutamiento más avanzadas. Si su aplicación necesita capacidades de balanceo de carga más avanzadas, puede ejecutar un paquete de balanceo de carga mediante software (por ejemplo, Zeus, HAProxy o NGINX Plus) en instancias EC2. A continuación, puede asignar direcciones IP elásticas a esas instancias EC2 de balanceo de carga para minimizar los cambios de DNS.[8]

Búsqueda de otros hosts y servicios

En la arquitectura de alojamiento web tradicional, la mayoría de sus hosts tienen direcciones IP estáticas. En la nube, la mayoría de sus hosts tendrán direcciones IP dinámicas. Aunque cada instancia EC2 puede tener entradas de DNS públicas y privadas y disponibles a través de Internet, las entradas de DNS y las direcciones IP se asignan de forma dinámica al lanzar la instancia. No se pueden asignar manualmente. Las direcciones IP estáticas (direcciones IP elásticas en la terminología de AWS) se pueden asignar a instancias en ejecución una vez que se hayan lanzado. Debe utilizar las direcciones IP elásticas para las instancias y servicios que requieren puntos de enlace coherentes, como, por ejemplo, las bases de datos maestras, servidores centrales de archivos y balanceadores de carga alojados en EC2.

Los roles de servidor que pueden ampliarse y reducirse, como por ejemplo los servidores web, deberían poderse detectar en sus puntos de enlace dinámico mediante el registro de su dirección IP en un repositorio central. Ya que la mayoría de las arquitecturas de aplicaciones web tienen un servidor de base de datos que está siempre activado, dicho servidor es un repositorio común para detectar información. Para situaciones en las que se necesitan direcciones coherentes, las instancias pueden ser direcciones IP elásticas asignadas a partir de un grupo de direcciones por un proceso de arranque cuando se lance la instancia.

Al usar este modelo, los hosts añadidos recientemente pueden solicitar la lista de puntos de enlace necesarios para las comunicaciones de la base de datos como parte de un proceso de arranque. La ubicación de la base de datos se

puede proporcionar como dato de usuario[9] que se transfiere a cada instancia cuando esta se lanza. Si lo prefiere, puede utilizar Amazon SimpleDB para almacenar y mantener información de configuración.[10] SimpleDB es un servicio de alta disponibilidad al que puede acceder desde un punto de enlace conocido.

Caché dentro de la aplicación web

El almacenamiento en la caché de la memoria de la aplicación puede reducir la carga sobre los servicios y mejorar el desempeño y la escalabilidad en la capa de base de datos mediante el almacenamiento en caché de la información más utilizada. Amazon ElastiCache[11] es un servicio web que facilita la implementación, el funcionamiento y el escalado del almacenamiento en la caché de la memoria en la nube. Puede configurar el almacenamiento en la caché de la memoria para que escale automáticamente según la carga y para sustituir automáticamente los nodos que hayan fallado. ElastiCache es compatible con el protocolo de Memcached y Redis, lo que simplifica la migración de su solución en las instalaciones físicas actual.

Configuración de bases de datos, backup, y conmutación por error

Muchas aplicaciones web contienen algún tipo de persistencia, por lo general en forma de base de datos relacional o NoSQL. AWS ofrece infraestructura de bases de datos relacionales y NoSQL. Si lo prefiere, puede implementar su propio software de base de datos en una instancia EC2. En la tabla siguiente se resumen estas opciones, y se explican más detalladamente en esta sección.

	Soluciones de bases de datos relacionales	**Soluciones NoSQL**
Servicio de bases de datos administrado	Amazon RDS: MySQL, Oracle, SQL Server, MariaDB, PostgreSQL, Amazon Aurora	Amazon DynamoDB
Autoadministrado	Alojamiento de un DBMS relacional en una instancia EC2	Alojamiento de una solución NoSQL en una instancia EC2

Amazon RDS

Amazon Relational Database Service (Amazon RDS) le proporciona acceso a las capacidades de motores de base de datos familiares como MySQL, PostgreSQL, Oracle y Microsoft SQL Server.[12] El código, las aplicaciones y las herramientas que ya utiliza se pueden utilizar con Amazon RDS. Amazon RDS aplica parches automáticamente en el software de la base de datos y realiza y almacena backups durante un periodo de retención definido por el usuario. También

admite la recuperación a un momento dado. Podrá beneficiarse de la flexibilidad que supone poder escalar los recursos de computación o la capacidad de almacenamiento asociada con su instancia de base de datos relacional mediante una única llamada a la API.

Además la implementación de varias zonas de disponibilidad (Multi-AZ) de Amazon RDS aumenta la disponibilidad de su base de datos y protege su base de datos frente a interrupciones no planificadas. Las réplicas de lectura de Amazon RDS ofrecen réplicas de solo lectura de su base de datos, por lo que puede escalar por encima de la capacidad de una única implementación de base de datos para cargas de trabajo de bases de datos intensivas en lectura. Al igual que con todos los servicios de AWS, no se requieren inversiones iniciales, y solo paga por los recursos que utiliza.

Alojamiento de un sistema de gestión de bases de datos relacionales (RDBMS) en una instancia de Amazon EC2

Además de la oferta de Amazon RDS administrada, puede instalar su la RDBMS que prefiera (MySQL, Oracle, SQL Server o DB2) en una instancia EC2 y administrarla usted mismo. Los clientes de AWS que alojan una base de datos en Amazon EC2 utilizan sin problemas una gran variedad de modelos de replicación y maestro/esclavo, como la duplicación para copias de solo lectura y el envío de registros para esclavos pasivos siempre dispuestos.

Si decide administrar su propio software de base de datos directamente en Amazon EC2, también debe tener en cuenta la disponibilidad del almacenamiento persistente y tolerante a errores. Para ello, le recomendamos que las bases de datos que se ejecuten en Amazon EC2 utilicen volúmenes de Amazon Elastic Block Store (Amazon EBS)[13], que son similares al almacenamiento asociado a la red. Para las instancias EC2 que ejecuten una base de datos, debe colocar todos los datos y registros de la base de datos en los volúmenes de EBS. Estos seguirán estando disponibles incluso si falla el host de la base de datos. Esta configuración permite un sencillo escenario de conmutación por error, en el que se puede lanzar una nueva instancia EC2 si falla un host y los volúmenes de EBS existentes se pueden asociar a la nueva instancia. La base de datos se puede retomar entonces desde donde se haya quedado.

Los volúmenes de EBS ofrecen automáticamente redundancia dentro de la zona de disponibilidad, lo que mejora su disponibilidad en comparación con los simples discos. Si el desempeño de un volumen de EBS individual no es

suficiente para las necesidades de sus bases de datos, se pueden distribuir los volúmenes para aumentar el desempeño de IOPS de su base de datos. Para cargas de trabajo exigentes también puede utilizar IOPS provisionadas de EBS, en las que puede especificarse el IOPS necesario. Si utiliza Amazon RDS, el servicio administra su propio almacenamiento para que pueda centrarse en la administración de sus datos.

Soluciones NoSQL

Además de admitir bases de datos relacionales, AWS también ofrece Amazon DynamoDB, [14] un servicio de bases de datos NoSQL totalmente administrado que ofrece un desempeño rápido y previsible, así como una perfecta escalabilidad. Con la consola de administración de AWS o la API de DynamoDB, puede aumentar o disminuir la capacidad sin que repercuta en el desempeño o el tiempo de inactividad. Ya que DynamoDB deja en manos de AWS la administración de las cargas administrativas de funcionamiento y ajuste de la escala de las bases de datos distribuidas, no tiene que preocuparse del aprovisionamiento, la instalación y la configuración del hardware, ni tampoco de las tareas de replicación, aplicación de parches de software o escalado de clústeres.

Amazon SimpleDB proporciona un servicio de base de datos no relacionales ligero, de alta disponibilidad y con tolerancia a errores, que ofrece la consulta e indexación de datos sin la necesidad de un esquema fijo. SimpleDB puede ser un sustituto muy eficaz de bases de datos en situaciones de acceso a los datos que requieran una tabla de esquema muy indexada y flexible.

Además, puede usar Amazon EC2 para alojar muchas otras tecnologías emergentes del movimiento NoSQL, como Cassandra, CouchDB y MongoDB.

Almacenamiento y backup de datos y recursos

Existen numerosas opciones dentro de la nube de AWS para almacenar, acceder y realizar un backup de los datos y recursos de su aplicación web. Amazon S3 proporciona un almacenamiento de objetos redundante y de alta disponibilidad. Amazon S3 es una gran solución de almacenamiento para objetos en cierto modo estáticos o que cambian lentamente, como, por ejemplo, imágenes, vídeos y otros recursos estáticos. Amazon S3 también admite el almacenamiento en caché de borde y el soporte de estos recursos mediante la interacción con CloudFront.

Para el almacenamiento como sistema de archivos adjuntos, las instancias EC2 pueden tener volúmenes de EBS asociados. Dichos volúmenes actúan como

discos montables para ejecutar las instancias EC2. Amazon EBS es ideal para datos a los que se deba acceder como almacenamiento por bloques y que requieran persistencia más allá de la vida útil de la instancia en ejecución, como, por ejemplo, las particiones de base de datos y los registros de la aplicación.

Además de disponer de una vida útil independiente de la instancia EC2, puede capturar instantáneas de los volúmenes de EBS y almacenarlos en Amazon S3. Debido a que las instantáneas de EBS solo realizan backups a partir de la instantánea anterior, si toma instantáneas con más frecuencia puede reducir el tiempo necesario para tomar cada instantánea. También puede utilizar una instantánea de EBS como referencia para replicar datos en varios volúmenes de EBS y asociar esos volúmenes a otras instancias en ejecución.

Los volúmenes de EBS pueden tener un tamaño de hasta 16 TB y se pueden distribuir varios volúmenes de EBS en volúmenes incluso más grandes o para un mayor desempeño de E/S. Para maximizar el desempeño de sus aplicaciones intensivas en E/S, puede utilizar los volúmenes de IOPS provisionadas. En el caso de los volúmenes de IOPS aprovisionadas, están diseñados para satisfacer las necesidades de las cargas de trabajo intensivas de E/S, en especial de las cargas de trabajo de bases de datos que son sensibles al desempeño del almacenamiento y a la coherencia en el rendimiento de E/S de acceso aleatorio. Especifique una velocidad de IOPS al crear el volumen y Amazon EBS aprovisionará esa velocidad durante toda la vida útil del volumen. Amazon EBS admite actualmente hasta 20 000 IOPS por volumen. También puede distribuir varios volúmenes juntos para ofrecer miles de IOPS por instancia a su aplicación.

Escalado automático de la flota

Una de las principales diferencias entre la arquitectura de nube de AWS y el modelo de alojamiento tradicional es que AWS puede escalar automáticamente bajo demanda la flota de la aplicación web, para administrar los cambios que se produzcan en el tráfico. En el modelo de alojamiento tradicional, los modelos de previsión de tráfico se utilizan generalmente para aprovisionar los hosts antes del tráfico proyectado. En AWS, las instancias se pueden aprovisionar sobre la marcha, de acuerdo a un conjunto de disparadores que permiten aumentar la flota y disminuirla de nuevo. El servicio Auto Scaling puede crear grupos de capacidad de servidores que pueden aumentarse o reducirse bajo demanda.[15] Auto Scaling también funciona directamente con Amazon CloudWatch para datos de métricas y con Elastic Load Balancing para añadir y eliminar hosts para distribución de carga. Por ejemplo, si los servidores web indican un uso de

la CPU superior al 80% durante un periodo de tiempo, se podría implementar rápidamente un servidor web adicional y, a continuación, agregarse automáticamente al balanceador de carga para su inmediata inclusión en la rotación de balanceo de carga.

Como aparece en el modelo de arquitectura de alojamiento web de AWS, puede crear varios grupos de Auto Scaling para diferentes capas de la arquitectura, de modo que cada capa se pueda escalar de forma independiente. Por ejemplo, el grupo de Auto Scaling del servidor web podría disparar la disminución y el aumento de escala en respuesta a los cambios que se produzcan en E/S de red, mientras que el grupo de Auto Scaling del servidor de aplicaciones podría aumentar o disminuir de escala en respuesta al uso de la CPU. Puede establecer mínimos y máximos para ayudar a garantizar la disponibilidad ininterrumpida y para limitar el uso dentro de un grupo.

Los disparadores de Auto Scaling se pueden establecer para aumentar y para reducir el tamaño de la flota total de una determinada capa de manera que el uso de recursos coincida con la demanda real. Además del servicio Auto Scaling, puede escalar las flotas de Amazon EC2 directamente a través de la API de Amazon EC2, que permite lanzar, terminar e inspeccionar instancias.

Características de seguridad adicionales

El número y la sofisticación de los ataques de denegación de servicio distribuido (DDoS) están aumentando. Estos ataques siempre han sido difíciles de repeler. A menudo acaban siendo costosos tanto en tiempo de mitigación como en esfuerzo empleado, además de los costos de oportunidad de las visitas a su sitio web perdidas durante el ataque. Hay una serie de factores y servicios de AWS que pueden ayudarle a defenderse contra dichos ataques. El primero es la escala de la red de AWS. La infraestructura de AWS es muy grande, y usted se aprovechará de nuestra escala para optimizar su defensa. Varios servicios, entre otros Elastic Load Balancing, Amazon CloudFront y Amazon Route 53, son eficaces en el escalado de su aplicación web en respuesta a un gran aumento de tráfico.

Dos servicios en particular le ayudarán con su estrategia de defensa. AWS Shield es un servicio de protección de DDoS administrado que le ayuda a protegerse frente a distintas formas de vectores de ataque DDoS.[16] La oferta estándar de AWS Shield es gratuita y se activa automáticamente a través de su cuenta. Esta oferta estándar ayuda en la defensa frente a los ataques más comunes a la capa de transporte y a la de red. Además de este nivel, la oferta avanzada otorga un nivel

de protección superior de su aplicación web al proporcionarle visibilidad casi en tiempo real del ataque en curso, así como integración en niveles superiores con los servicios mencionados anteriormente. Además, obtendrá acceso al Equipo de respuesta frente a DDoS de AWS (DRT) para ayudar a mitigar los ataques sofisticados y a gran escala contra sus recursos.

AWS WAF (firewall de aplicación web) está diseñado para proteger sus aplicaciones web de ataques que pueden comprometer la disponibilidad o la seguridad, o consumir demasiados recursos.[17] AWS WAF funciona en línea con CloudFront o Application Load Balancer, junto con sus reglas personalizadas, para ayudar en la defensa frente a ataques como, por ejemplo, scripting entre sitios, inyección SQL y DDoS. Al igual que ocurre con la mayoría de servicios de AWS, AWS WAF incluye una API completa que puede ayudar a automatizar la creación y la edición de reglas para su WAF a medida que cambian sus necesidades de seguridad.

Conmutación por error con AWS

Otra ventaja clave de AWS frente al alojamiento web tradicional son las zonas de disponibilidad, que le ofrecen un fácil acceso a las ubicaciones de implementación redundantes. Las zonas de disponibilidad son ubicaciones físicamente distintas, diseñadas de forma que queden aisladas en caso de error en otras zonas de disponibilidad. Proporcionan conectividad de red económica y de baja latencia con otras zonas de disponibilidad dentro de la misma región AWS. Como indica el diagrama de la arquitectura de alojamiento web de AWS de la figura 3, le recomendamos que implemente hosts de EC2 en varias zonas de disponibilidad para que su aplicación web sea más tolerante a errores. Es importante garantizar que existe aprovisionamiento para migrar puntos de acceso individuales entre zonas de disponibilidad, en el caso de que se produzca algún error. Por ejemplo, debe configurar un esclavo de base de datos en una segunda zona de disponibilidad para que la persistencia de datos permanezca constante y con alta disponibilidad, incluso durante una improbable situación de error. Puede hacer esto en Amazon EC2 o Amazon RDS con solo hacer clic en un botón.

Aunque suelen ser necesarios algunos cambios de arquitectura al mover una aplicación web existente a la nube de AWS, existen importantes mejoras en términos de escalabilidad, fiabilidad y rentabilidad que hacen que utilizar la nube de AWS merezca el esfuerzo. En la siguiente sección, se explican las mejoras.

Consideraciones clave sobre el uso de AWS para el alojamiento web

Existen algunas diferencias clave entre la nube de AWS y un modelo de alojamiento de tradicional. En la sección anterior se resaltan muchas de las áreas clave que debe tener en cuenta al implementar una aplicación web en la nube. En esta sección se señalan algunos de los cambios de arquitectura clave que debe tener en cuenta cuando incorpore cualquier aplicación a la nube.

Ya no hay dispositivos de red físicos

No se pueden implementar dispositivos de red físicos en AWS. Por ejemplo, los firewalls, los routers o los balanceadores de carga de sus aplicaciones de AWS ya no puede residir en dispositivos físicos; deben sustituirse por soluciones de software. Existe una amplia variedad de soluciones de software de calidad empresarial, ya sea para el balanceo de carga (p. ej., Zeus, HAProxy, NGINX Plus y Pound) o para establecer una conexión de VPN (p. ej., OpenVPN, OpenSwan y Vyatta). Esto no debe entenderse como una limitación de qué se puede ejecutar en la nube de AWS, sino como un cambio de arquitectura para su aplicación si hoy por hoy utiliza estos dispositivos.

Firewalls en todas partes

Donde una vez hubo una sencilla DMZ, y a continuación, comunicaciones abiertas entre sus hosts, en un modelo de alojamiento tradicional, AWS aplica hoy un modelo más seguro, en el que cada host queda bloqueado. Uno de los pasos que hay que realizar en la planificación de una implementación de AWS, es el análisis de tráfico entre hosts. Este análisis guiará las decisiones acerca de qué puertos se deben abrir exactamente. Puede crear grupos de seguridad dentro de Amazon EC2 para cada tipo de host de su arquitectura. Además, puede crear de forma sencilla una gran variedad de modelos de seguridad por niveles para habilitar el mínimo acceso entre los hosts de su arquitectura. El uso de listas de control de acceso a la red en Amazon VPC puede ayudar a bloquear su red a nivel de subred.

Tenga en cuenta la disponibilidad de varios centros de datos

Las zonas de disponibilidad de una región de AWS son como varios centros de datos distintos. Las instancias EC2 en diferentes zonas de disponibilidad están separadas lógicamente y físicamente, y proporcionan un modelo fácil de usar

para implementar la aplicación en los distintos centros y de ese modo obtener una gran disponibilidad y fiabilidad. Amazon VPC como servicio regional le permite aprovechar las zonas de disponibilidad manteniendo todos sus recursos en la misma red lógica.

Tratar los hosts como efímeros y dinámicos

Probablemente el cambio más importantes en cómo diseñar su aplicación de AWS es que los hosts de Amazon EC2 deben considerarse efímeros y dinámicos. Cualquier aplicación creada para la nube de AWS no debe asumir que un host siempre esté disponible y se debe diseñar sabiendo que los datos que no se encuentren en un volumen de EBS se perderán si falla una instancia EC2. Además, cuando aparece un nuevo host, no se deben hacer suposiciones sobre la dirección IP o ubicación dentro de una zona de disponibilidad del host. El modelo de configuración debe ser flexible y su enfoque de proceso de arranque en host debe tener en cuenta la naturaleza dinámica de la nube. Estas técnicas son fundamentales para crear y ejecutar una aplicación muy escalable y tolerante a fallos.

Considere una arquitectura sin servidor

Este documento técnico se centra principalmente en una arquitectura web más tradicional. Sin embargo, los servicios más recientes como AWS Lambda[18] y Amazon API Gateway[19] le permiten crear una aplicación web sin servidor que omite el uso de máquinas virtuales para realizar la computación. En estos casos, el código se ejecuta solicitud por solicitud, y usted solo paga por el número de solicitudes y la longitud de las solicitudes. Puede obtener más información acerca de las arquitecturas sin servidor aquí.

Conclusiones

Debe considerar muchos aspectos tanto arquitectónicos como conceptuales si piensa migrar su aplicación web a la nube de AWS. Los beneficios de tener una infraestructura rentable, muy escalable y tolerante a errores que crece con su empresa superan en gran medida los esfuerzos necesarios para migrar a la nube de AWS.

Colaboradores

Las siguientes personas y organizaciones han participado en la redacción de este documento:

- Jack Hemion, Arquitecto asociado de soluciones, AWS
- Matt Tavis, Arquitecto principal de soluciones, AWS
- Philip Fitzsimons, Director de buena arquitectura, AWS

Documentación adicional

- Guía de introducción - Alojamiento de aplicación web AWS para Linux
- Guía de introducción - Alojamiento de aplicación web AWS para Windows
- Serie de vídeos de introducción: Aplicaciones web Linux en la nube de AWS
- Serie de vídeos de introducción: .Aplicaciones web .NET en la nube de AWS

Revisiones del documento

Fecha	Descripción
Julio de 2017	Se han añadido y actualizado varias secciones para nuevos servicios. Diagramas actualizados para mayor claridad y para servicios adicionales. Se ha añadido VPC como método de red estándar en AWS dentro de "Administración de redes". Se ha añadido la sección sobre protección y mitigación de ataques DDoS en "Características de seguridad adicionales". Se ha añadido una pequeña sección sobre arquitecturas sin servidor para alojamiento web.
Septiembre de 2012	Se han actualizado varias secciones para mayor claridad. Se han actualizado diagramas para utilizar iconos AWS. Se ha añadido la sección "Administración de DNS público" para ofrecer información sobre Amazon Route 53. Se ha actualizado la sección "Búsqueda de otros hosts y servicios" para mayor claridad. Se ha actualizado la sección "Configuración de bases de datos, backup, y conmutación por error" para mayor claridad y se ha añadido DynamoDB. Se ha ampliado la sección "Almacenamiento y backup de datos y recursos" para cubrir los volúmenes de IOPS provisionadas de EBS.
Mayo de 2010	Publicación inicial

Notes

[1] https://aws.amazon.com/vpc/

[2] https://aws.amazon.com/cloudfront/

[3] https://aws.amazon.com/s3/

[4] https://aws.amazon.com/ec2/

[5] https://aws.amazon.com/route53/

[6] https://aws.amazon.com/security/

[7] https://aws.amazon.com/elasticloadbalancing/

[8] Las direcciones IP elásticas son direcciones IP estáticas diseñadas para la computación en la nube dinámica, que puede mover de una instancia a otra.

[9] http://docs.aws.amazon.com/AWSEC2/latest/APIReference/Welcome.html

[10] https://aws.amazon.com/simpledb/

[11] https://aws.amazon.com/elasticache/

[12] https://aws.amazon.com/rds/

[13] https://aws.amazon.com/ebs/

[14] https://aws.amazon.com/dynamodb/

[15] https://aws.amazon.com/autoscaling/

[16] https://aws.amazon.com/shield/

[17] https://aws.amazon.com/waf/

[18] https://aws.amazon.com/lambda/

[19] https://aws.amazon.com/api-gateway/