Disponibilidad

La disponibilidad es el acceso a la información y a los sistemas por personas autorizadas en el momento que así lo requieran. Los controles de seguridad utilizados para protegerlo, y los canales de comunicación protegidos que se utilizan para acceder a ella deben estar funcionando correctamente. La alta disponibilidad debe estar disponible en todo momento, evitando interrupciones del servicio debido a cortes de energía, fallos de hardware, y actualizaciones del sistema.

Garantizar la disponibilidad implica también la prevención de ataque de denegación de servicio. Para poder manejar con mayor facilidad la seguridad de la información, las empresas o negocios se pueden ayudar con un sistema de gestión que permita conocer, administrar y minimizar los posibles riesgos que atenten contra la seguridad de la información del negocio.

Introducción a la disponibilidad en Cloud Foundry

Cloud Foundry usado correctamente puede convertirse en la plataforma central para el desarrollo de aplicaciones así como las actividades de su implementación. Por tanto, la disponibilidad de la plataforma es fundamental para la continuidad del servicio. Los fallos se pueden solucionar de tres formas principalmente:

- Diseño para mayor resistencia y alta disponibilidad.
- Emplear mecanismos de respaldo y restauración.
- Ejecutar pruebas de verificación de la plataforma de manera constante.

Consideraciones de alta disponibilidad

A los usuarios finales no le importa hasta qué punto sea o no resistente su sistema, solo les preocupa que su aplicación, servicio o funcionalidad esté disponible siempre. La alta disponibilidad se mide con la percepción de los usuarios finales, que al fin y al cabo son los que experimentan percepciones negativas cuando el servicio al que intentan acceder no está disponible.

Tener alta disponibilidad puede ser considerado como el grado en que un componente está disponible para un usuario cuando éste lo necesite. Para conseguir una alta disponibilidad, básicamente, se necesita tener una configuración redundante de cada uno de los componentes que haya en la infraestructura, de modo que se evite cualquier punto que provoque fallos graves, evitando así que el sistema caiga.

Comprender los posibles errores, como pueden ser a nivel de red, host o cluster, por ejemplo, puede ayudar a configurar y elegir una estrategia para conseguir la alta disponibilidad ya que permite evaluar cada uno de los posibles riesgos frente al coste de evitar dichos riesgos.

Por ejemplo, una máquina tiene una probabilidad alta de tener un fallo y por lo tanto es razonable tener una réplica; en cambio, la probabilidad de que un dentro de datos por completo quede desconectado no es tan alta y los costes asociados a la replicación, en

comparación con el riesgo, son demasiado altos. Es esencial comprender qué puede fallar, el impacto de que esto ocurra así como el coste de prevenirlo antes de buscar la alta disponibilidad.

Una vez estudiados estos aspectos, dentro de un sistema distribuido como Cloud Foundry, se debe centrar el interés en la interconexión de los componentes para que cuando un componente individual deje de funcionar correctamente, pueda ser omitido sin perder la funcionalidad general del sistema. Cloud Foundry, junto con BOSH, promueve la capacidad de recuperación mediante la recuperación automática, que es la capacidad de superar el fallo en la aplicación, proceso o componente mediante el tratamiento y corrección de errores incorporado.

Cloud Foundry encara la alta disponibilidad desde arriba hacia abajo, comenzando con la disponibilidad de la aplicación y luego comprobando cada uno de los componentes del sistema que conforman la infraestructura.

Disponibilidad del centro de datos

Cabe destacar que antes de pensar en la alta disponibilidad de Cloud Foundry, es necesario que si centro de datos esté configurado correctamente con un nivel adecuado de alta disponibilidad.

Ampliación de la capacidad de recuperación de Cloud Foundry

Cloud Foundry va más allá de la simple estrategia de replicación, sino que logra la resistencia incorporada de cuatro formas clave:

- Reiniciar procesos fallidos del sistema.
- Recrear máquinas virtuales no disponibles.
- Implementación dinámica de nuevas instancias de aplicaciones si una aplicación deja de responder.
- Distribución de las aplicaciones para forzar la separación de la infraestructura subyacente.

Cloud Foundry proporciona recuperación automática de aplicaciones, procesos y máquinas virtuales. Los cuatro niveles de alta disponibilidad nombrados anteriormente proporcionan resistencia dentro de los límites de una sola instalación de Cloud Foundry. Si se experimenta un fallo más amplio en el centro de datos debido a problemas en la infraestructura, una única implementación de Cloud Foundry podría quedar temporalmente inutilizable.

Las interrupciones del centro de datos son extremadamente raras, pero si necesita un nivel adicional de resistencia para mitigar cualquier posible fallo del centro de datos, es posible ejecutar múltiples implementaciones de Cloud Foundry en diferentes centros de datos.

Consistencia de datos a través de servicios

Uno de los mayores retos con la ejecución de cualquier tecnología de capa de aplicación en dos centros de datos de forma activa en ambos, es la capa de datos que sustenta la aplicación. Los servicios de respaldo de datos deben mantener la coherencia entre los dos centros de

datos. Esta preocupación se explica por la teoría de que es imposible que un sistema distribuido proporcione simultáneamente garantías de consistencia, disponibilidad y tolerancia de partición. En cualquier momento, un sistema distribuido puede garantizar solo dos de los tres requisitos.

Mantener la consistencia de los datos en diferentes centros de datos aumenta el desafío debido a la mayor latencia cuando se intenta propagar los cambios de datos en dos ubicaciones separadas.

- Si adopta un modelo de escritura simultánea en sus centros de datos, en el que devuelve una confirmación de escritura después de que se complete, la experiencia del usuario final podrá ser muy lenta. La latencia es un problema de disponibilidad y cualquier latencia de red adicional solo agravará este problema.
- Si adopta un modelo de escritura diferida en el que devuelve inmediatamente la confirmación de una escritura exitosa y luego intenta propagar el cambio después, corre el riesgo de que los dos almacenes de datos no queden sincronizados, dejando datos inconsistentes.

La solución aquí puede ser utilizar una capa de almacenamiento en caché local como Cassandra. El uso de una capa de datos distribuidos que garantiza la consistencia eventual en una red WAN le permite conservar los cambios localmente, permitiendo una respuesta rápida y alta disponibilidad al usuario final. Si ocurre algún conflicto, se usan varios algoritmos de resolución de dichos conflictos para garantizar que el sistema sea finalmente coherente. Aunque ningún sistema consistente es infalible al cien por cien, se proporciona una solución sólida al problema de la consistencia planteado.

Copias de seguridad

Hay veces que es posible que necesite restaurar completamente su entorno. Esto podría deberse a cualquiera de estas situaciones:

- Creación de una copia de una implementación existente para crear un entorno completamente nuevo.
- Motivos de mantenimiento, como cambiar o actualizar su red, servidor o capa de almacenamiento.
- Recuperación de un ataque malicioso en todo el centro de datos que provoca un fallo catastrófico.

Hay varios proyectos que existen para respaldar y restaurar Cloud Foundry, como por ejemplo:

- cf-converger de Engineer Better
- cfops de Pivotal Services

Para restaurar su entorno, primero debe crear una copia de seguridad de la capa de datos. Al pensar en Cloud Foundry desde una perspectiva de recuperación ante desastres, debe pensar en el sistema distribuido como un conjunto de discos persistentes que contienen lo siguiente:

- El CCDB.

- La base de datos de la UAA.
- El BBS.
- El DB BOSH.
- El blobstore o servidor NFS.
- Configuración.
- Cualquier otra capa de persistencia centrada en la aplicación.

La capa de persistencia es Cloud Foundry desde una perspectiva de recuperación de desastres; Todo lo demás, todos los procesos en ejecución, pueden volver a conectarse fácilmente.

Disponibilidad de Cloud Foundry

La copia de seguridad de Cloud Foundry suspenderá las escrituras mientras dure la copia de seguridad. Esto es importante porque no puede escribir simultáneamente en una base de datos y mantener la integridad de la copia de seguridad de la base de datos. La suspensión de las escrituras hará que su base de datos se convierta en de solo lectura durante la copia de seguridad. Las aplicaciones no funcionarán durante este tiempo.

Recuperar BOSH es el primer paso para recuperar cualquier entorno de Cloud Foundry que se encuentre offline. Por lo tanto, es vital que tome instantáneas regulares de la base de datos BOSH. Es mejor usar una base de datos externa como MySQL en clúster o AWS-RDS y un blobstore externo como Amazon S3. Sin embargo, si utiliza la base de datos interna de BOSH, hay algunos pasos adicionales necesarios para realizar una copia de seguridad y restaurar la base de datos interna y el blobstore de BOSH.

Puede usar los siguientes pasos para hacer una copia de seguridad de la base de datos interna y el blobstore de BOSH:

1. Usando SSH, conéctese a BOSH: $ ssh -i key vcap@boship
2. Conviértete en root: $ su -(usa tus credenciales de VM)
3. Ejecutar $ monit summary para ver todos los procesos BOSH
4. Ejecutar $ monit stop all para detener limpiamente todos los procesos BOSH
5. Tome una instantánea del volumen de disco persistente BOSH

Estos son los pasos para restaurar la base de datos interna y el blobstore de BOSH:

1. Usando su manifiesto bosh.yml original , reconstruya BOSH. Esto crea un nuevo disco persistente vacío.
2. Repita los pasos 1 a 4 del procedimiento de copia de seguridad. Deberá detener todos los procesos antes de desconectar el disco persistente.
3. Desacoplar el disco persistente (eliminándolo).
4. Cree un nuevo volumen a partir de su instantánea de volumen de disco y luego adjunte manualmente el nuevo volumen a la máquina virtual BOSH.
5. Iniciar todos los procesos de nuevo. BOSH ahora tendrá el mismo UOS BOSH (porque está almacenado en la base de datos).

Una vez que haya restaurado con éxito BOSH, BOSH debería restaurar con éxito la implementación de Cloud Foundry, asumiendo que ha utilizado una base de datos externa y accesible y blobstore para los otros componentes.