¿Qué es Docker?

Docker es una plataforma de software que le permite crear, probar e implementar aplicaciones rápidamente. Docker empaqueta software en unidades estandarizadas llamadas contenedores que incluyen todo lo necesario para que el software se ejecute, incluidas bibliotecas, herramientas de sistema, código y tiempo de ejecución. Con Docker, puede implementar y ajustar la escala de aplicaciones rápidamente en cualquier entorno con la certeza de saber que su código se ejecutará.

Docker es un proyecto de código abierto para automatizar la implementación de aplicaciones como contenedores portátiles y autosuficientes que se pueden ejecutar en la nube o localmente. Docker es también una empresa que promueve e impulsa esta tecnología, en colaboración con proveedores de la nube, Linux y Windows, incluido Microsoft.

os contenedores de Docker se pueden ejecutar en cualquier lugar, localmente en el centro de recursos de cliente, en un proveedor de servicios externo o en la nube, en Azure. Los contenedores de imágenes de Docker también se pueden ejecutar de forma nativa en Linux y Windows. Sin embargo, las imágenes de Windows solo pueden ejecutarse en hosts de Windows y las imágenes de Linux pueden ejecutarse en hosts de Linux y hosts de Windows (con una máquina virtual Linux de Hyper-V, hasta el momento), donde host significa un servidor o una máquina virtual.

Los desarrolladores pueden usar entornos de desarrollo en Windows, Linux o macOS. En el equipo de desarrollo, el desarrollador ejecuta un host de Docker en que se implementan imágenes de Docker, incluidas la aplicación y sus dependencias. Los desarrolladores que trabajan en Linux o en Mac usan un host de Docker basado en Linux y solo pueden crear imágenes para contenedores de Linux. (Los desarrolladores que trabajan en Mac pueden editar código o ejecutar la interfaz de la línea de comandos, o CLI, de Docker en macOS, pero en el momento de redactar este artículo, los contenedores no se ejecutan directamente en macOS). Los desarrolladores que trabajan en Windows pueden crear imágenes para contenedores de Windows o Linux.

Para hospedar contenedores en entornos de desarrollo y proporcionar herramientas de desarrollador adicionales, Docker entrega Docker Desktop para Windows o para macOS. Estos productos instalan la máquina virtual necesaria (el host de Docker) para hospedar los contenedores.

Para ejecutar contenedores de Windows, hay dos tipos de tiempos de ejecución:

Los contenedores de Windows Server ofrecen aislamiento de aplicaciones a través de tecnología de aislamiento de proceso y de espacio de nombres. Un contenedor de Windows Server comparte el kernel con el host de contenedor y con todos los contenedores que se ejecutan en el host.

Los contenedores de Hyper-V amplían el aislamiento que ofrecen los contenedores de Windows Server mediante la ejecución de cada contenedor en una máquina virtual altamente optimizada. En esta configuración, el kernel del host del contenedor no se comparte con los contenedores de Hyper-V, lo que proporciona un mejor aislamiento.

Las imágenes de estos contenedores se crean y funcionan de la misma manera. La diferencia radica en cómo se crea el contenedor desde la imagen ejecutando un contenedor de Hyper-V que requiere un parámetro adicional. Para más información, vea Contenedores de Hyper-V.

Comparación de los contenedores de Docker con las máquinas virtuales

En la figura 1-3 se muestra una comparación entre las máquinas virtuales y los contenedores de Docker.

Diagrama que muestra una comparación de los entornos de máquinas virtuales y contenedores.

Figura 1-3. Comparación de las máquinas virtuales tradicionales con los contenedores de Docker

Tal y como se muestra en el diagrama anterior, en el caso de las máquinas virtuales, hay tres capas base en el servidor host. De abajo hacia arriba: infraestructura, sistema operativo host y un hipervisor. Encima de todo eso, cada máquina virtual tiene su propio sistema operativo y todas las bibliotecas necesarias. Por otro lado, para Docker, el servidor host solo tiene la infraestructura y el sistema operativo. Además, el motor de contenedor mantiene los contenedores aislados, pero les permite compartir los servicios del sistema operativo de base única.

Dado que los contenedores requieren muchos menos recursos (por ejemplo, no necesitan un sistema operativo completo), se inician rápidamente y son fáciles de implementar. Esto permite tener una mayor densidad, lo que significa que se pueden ejecutar más servicios en la misma unidad de hardware, reduciendo así los costos.

Como efecto secundario de la ejecución en el mismo kernel, obtiene menos aislamiento que las máquinas virtuales.

El objetivo principal de una imagen consiste en garantizar el mismo entorno (dependencias) entre las distintas implementaciones. Esto significa que puede depurarlo en su equipo y, después, implementarlo en otra máquina con el mismo entorno garantizado.

Una imagen de contenedor es una manera de empaquetar una aplicación o un servicio e implementarlo de forma confiable y reproducible. Podría decir que Docker no solo es una tecnología, sino también una filosofía y un proceso.

Al usar Docker, no escuchará a los desarrolladores decir "Si funciona en mi máquina, ¿por qué no en producción?". Pueden decir simplemente "Se ejecuta en Docker", porque la aplicación de Docker empaquetada puede ejecutarse en cualquier entorno de Docker compatible, y se ejecuta de la forma prevista en todos los destinos de implementación (como desarrollo, control de calidad, almacenamiento provisional y producción).

Una analogía simple

Quizás una analogía simple puede ayudar a entender el concepto básico de Docker.

Vamos a remontarnos a la década de 1950 por un momento. No había ningún procesador de texto y las fotocopiadoras se usaban en todas partes (o casi).

Imagine que es responsable de enviar lotes de cartas, según proceda, para enviarlas por correo a los clientes en papel y sobres reales, que se entregarán físicamente en la dirección postal de cada cliente (entonces no existía el correo electrónico).

En algún momento, se da cuenta de que las cartas no son más que una composición de un conjunto grande de párrafos, que se eligen y ordenan según proceda, según el propósito de la carga, por lo que diseña un sistema para emitir cartas rápidamente, esperando conseguir una increíble mejora.

El sistema es simple:

Comience con un conjunto de hojas transparentes que contienen un párrafo.

Para emitir un conjunto de cartas, elija las hojas con los párrafos necesarios, apílelas y alinéelas para que queden y se lean bien.

Por último, colóquelas en la fotocopiadora y presione inicio para producir tantas cartas como sean necesarias.

Por tanto, para simplificar, esa es la idea principal de Docker.

En Docker, cada capa es el conjunto resultante de los cambios que se producen en el sistema de archivos después de ejecutar un comando, como instalar un programa.

Por lo tanto, al "Examinar" el sistema de archivos después de que se ha copiado la capa, verá todos los archivos, incluida la capa cuando se instaló el programa.

Puede pensar en una imagen como un disco duro de solo lectura auxiliar listo para instalarse en un "equipo" donde el sistema operativo ya está instalado.

De forma similar, puede pensar en un contenedor como el "equipo" con el disco duro de imagen instalado. El contenedor, como un equipo, se puede apagar o desactivar.