Escalamiento en Azure con Máquinas Virtuales, Sacale Sets y Service Plans

Diego León
Noviembre de 2022

Escuela Colombiana de Ingeniería
Ingeniería de Sistemas
Arquitecturas de Software

Parte 1 - Escalabilidad vertical

Preguntas

1. ¿Cuántos y cuáles recursos crea Azure junto con la VM?
    - Azure crea 5 recursos junto al recurso de la Máquina virtual. Los recursos que crea son: Virtual Network, Public IP address, Network Security Group, Network Interface y OS Disk.
2. ¿Brevemente describa para qué sirve cada recurso?
    - Virtual Network: Permite que recursos de Azure como lo son las máquinas virtuales, se puedan comunicarse de forma segura entre usuarios, con Internet y con las redes locales. Es similar a una red tradicional.
    - Public IP address: Dirección IP cuyo conjunto de números identifica, de manera lógica y jerárquica, a una interfaz en la red de un dispositivo que utilice el protocolo o que corresponde al nivel de red del modelo TCP/IP, de tal manera que sea visible con internet y algunos servicios públicos de azure.
    - Network Security Group: Se utiliza para filtrar el tráfico de red desde y para los recursos de Azure en una red virtual de Azure. Un grupo de seguridad se basa principalmente en un conjunto de reglas de seguridad que permiten o niegan el tráfico de red entrante o el tráfico de red saliente.
    - Network Interface: Componente que permite que las VM de Azure se comunique con Internet y demás recursos locales.
    - OS Disk: Almacenamiento del OS de la máquina virtual creada.

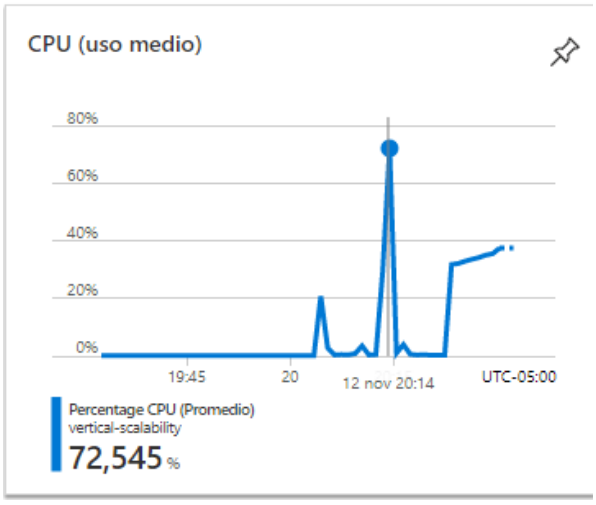
3. ¿Al cerrar la conexión ssh con la VM, por qué se cae la aplicación que ejecutamos con el comando npm FibonacciApp.js? ¿Por qué debemos crear un Inbound port rule antes de acceder al servicio?
    - A la hora de iniciar la conexión con la máquina virtual por medio de ssh y ejecutar el comando npm para la ejecución de la aplicación se está realizando y asociando a un usuario en específico que es por el que entramos con ssh. Así, a la hora de cerrar la conexión con ese usuario también se va a cerrar los procesos que tuviera el mismo en la máquina como la ejecución de la aplicación.
    - Se debe crear para indicarle al grupo de seguridad de la máquina que abra ese puerto para poder acceder al mismo. De lo contrario, no se podría acceder al mismo y no se podría ejecutar la aplicación.
4. Adjunte la tabla de tiempos e intérprete por qué la función tarda tanto tiempo.

N	B1ls (s)	B2ms (s)
1000000	19.51	20.05
1010000	19.68	20.07
1020000	20.17	20.37
1030000	20.57	20.75
1040000	20.93	21.11
1050000	21.49	21.62
1060000	21.80	22.50
1070000	22.35	23.03
1080000	23.14	22.74
1090000	22.99	23.18

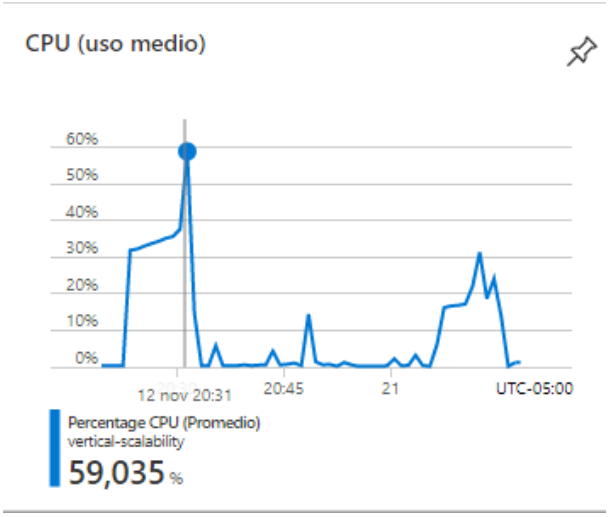
La función tarda demasiado tiempo debido a que el código con el cual se implementa Fibonacci no es óptimo. El problema se podría resolver haciendo uso de recursividad y guardar valores ya implementados para partir de los mismos y no tener que hacerlos de nuevo.

5. Adjunte imágen del consumo de CPU de la VM e intérprete por qué la función consume esa cantidad de CPU.

- B1ls



- B2ms



La cantidad de CPU consumida es bastante grande por lo mencionado anteriormente, el algoritmo con el cual se hace Fibonacci no es óptimo y no es recurrente, por lo que realiza para cada nuevo valor una nueva secuencia para llegar al fibonacci del número ingresado.

6. Adjunte la imagen del resumen de la ejecución de Postman. Intérprete:

   - B1ls

	executed	failed
iterations	10	0
requests	10	0
test-scripts	10	0
prerequest-scripts	0	0
assertions	0	0
total run duration: 4m 11.5s		
total data received: 2.09MB (approx)		
average response time: 25.1s [min: 19.1s, max: 37.9s, s.d.: 7.3s]		

   - B2ms

	executed	failed
iterations	10	0
requests	10	0
test-scripts	10	0
prerequest-scripts	0	0
assertions	0	0
total run duration: 3m 12.3s		
total data received: 2.09MB (approx)		
average response time: 19.2s [min: 19s, max: 19.6s, s.d.: 190ms]		

- Tiempos de ejecución de cada petición.
- Debido a que realizamos el cambio de tamaño para permitir más peticiones recurrentes se puede visualizar que el tiempo de ejecución total y el promedio es menor cuando tenemos B2ms que cuando lo hicimos teniendo B1ls
- Si hubo fallos documentalos y explique.
- No hubo fallos en las iteraciones.

7. ¿Cuál es la diferencia entre los tamaños B2ms y B1ls (no solo busque especificaciones de infraestructura)?
   - B1ls

Name	Memory	vCPUs	Linux Pay As You Go cost
B1ls	Search	Search	Search
B1ls	0.5 Gib	1	0.00624 hourly

   - B2ms

Name	Memory	vCPUs	Linux Pay As You Go cost
B2m	Search	Search	Search
B2ms	8 Gib	2	0.0998 hourly

   Las GiB de almacenamiento temporal(SSD) son 4 para B1ls y 16 para B2ms lo que sería cuatro veces más almacenamiento. Asimismo, el rendimiento base de CPU de la máquina virtual con B1ls es del 5% mientras que con B2ms es del 60%.

8. ¿Aumentar el tamaño de la VM es una buena solución en este escenario?, ¿Qué pasa con la FibonacciApp cuando cambiamos el tamaño de la VM?
   - Aumentar el tamaño de la máquina virtual puede dar solución a mejorar en un mínimo porcentaje los tiempos de ejecución de Fibonacci, más sin embargo, para

encontrar una solución que radique el problema de raíz es revisar el algoritmo con el cual se está ejecutando la secuencia.

- Cuando cambiamos de tamaño en la máquina virtual se reduce el tiempo de ejecución de FibonacciApp.

9. ¿Qué pasa con la infraestructura cuando cambia el tamaño de la VM? ¿Qué efectos negativos implica?
    - El cambio de tamaño de la máquina virtual implica la necesidad de reiniciar la máquina para poder efectuar el cambio. Por lo que, la máquina estará sin servicio durante un corto lapso de tiempo mientras se realiza el respectivo cambio de tamaño.
10. ¿Hubo mejoras en el consumo de CPU o en los tiempos de respuesta? Si/No ¿Por qué?
    - Sí, debido a que como muestro en la comparativa de tamaños al usar B2ms se tiene un rendimiento base de CPU del 60% muchísimo más que cuando usamos B1ls.
11. Aumente la cantidad de ejecuciones paralelas del comando de postman a 4. ¿El comportamiento del sistema es porcentualmente mejor?
    - No hubo ningún cambio en el tiempo de ejecución de FibonacciApp por lo que no se puede apreciar ninguna mejora en el compartiempo del sistema. Debido a que los tiempos son bastantes similares.

Parte 2 - Escalabilidad horizontal

Preguntas

1. ¿Cuáles son los tipos de balanceadores de carga en Azure y en qué se diferencian?, ¿Qué es SKU, qué tipos hay y en qué se diferencian?, ¿Por qué el balanceador de carga necesita una IP pública?

   1.1 ¿Cuáles son los tipos de balanceadores de carga en Azure y en qué se diferencian?

   - Balanceador de carga interno: Este balanceador de carga se encarga de equilibrar la carga de tráfico de una red privada
   - Balanceador de carga público: Este balanceador de carga se encarga de equilibrar la carga de tráfico de redes públicas, específicamente de la carga proveniente de internet.
   - Balanceador de carga de puerta de enlace: Es un balanceador que se adapta a escenarios de alto rendimiento y alta disponibilidad con dispositivos virtuales de red (NVA) de terceros. Con las capacidades de Gateway Load Balancer, puede implementar, escalar y administrar NVA fácilmente.

   1.2 ¿Qué es SKU, qué tipos hay y en qué se diferencian?

   - Representa una Unidad de Mantenimiento de Stock (SKU) adquirible bajo un producto. Representan las diferentes formas del producto.

   - Azure Load Balancer tiene 3 SKU: Básico, Estándar y Puerta de enlace. Cada SKU está diseñada para un escenario específico y tiene diferencias en cuanto a escala, características y precios.

	Standard Load Balancer	Versión Básico de Load Balancer
Escenario	Equipado para el tráfico de la capa de red de equilibrio de carga cuando se necesitan un alto rendimiento y una latencia muy baja. Enruta el tráfico dentro y entre regiones, y a zonas de disponibilidad para lograr una alta resistencia.	Equipado para aplicaciones a pequeña escala que no necesitan alta disponibilidad ni redundancia. No es compatible con las zonas de disponibilidad.
Tipo de back-end	Basado en IP, basado en NIC	Basado en NIC
Protocolo	TCP, UDP	TCP, UDP
Puntos de conexión del grupo de back-end	Todas las máquinas virtuales o conjuntos de escalado de máquinas virtuales de una red virtual individual.	Máquinas virtuales en un único conjunto de disponibilidad o conjunto de escalado de máquinas virtuales.
Sondeos de estado	TCP, HTTP, HTTPS	TCP, HTTP
Comportamiento del sondeo de mantenimiento	Las conexiones TCP permanecen activas en el sondeo de la instancia y en todos los sondeos.	Las conexiones TCP permanecen activas en un sondeo de instancia. Todas las conexiones TCP terminan cuando todos los sondeos están inactivos.
Zonas de disponibilidad	Servidores front-end con redundancia de zona y zonales para el tráfico de entrada y salida.	No disponible
Diagnóstico	Métricas multidimensionales de Azure Monitor	No compatible
Puertos de alta disponibilidad	Disponibles para el equilibrador de carga interno	No disponible
Seguro de forma predeterminada	Cerrado a los flujos de entrada, a menos que lo permita un grupo de seguridad de red. Se permite el tráfico interno desde la red virtual al equilibrador de carga interno.	Abrir de forma predeterminada. Grupo de seguridad de red opcional.
Reglas de salida	Configuración declarativa de NAT de salida	No disponible
Restablecimiento de TCP en tiempo de espera de inactividad	Disponible en cualquier regla	No disponible
Varios servidores front-end	Entrada y salida	Solo de entrada

1.3 ¿Por qué el balanceador de carga necesita una IP pública?

- Una IP pública asociada a un equilibrador de carga actúa como configuración de IP del front-end orientada a Internet. El front-end se usa para acceder a los recursos del grupo de back-end. La IP del front-end se puede usar para que los miembros del grupo de back-end salgan a Internet.

2. ¿Cuál es el propósito del Backend Pool?

- El almacenamiento de las direcciones IP de las máquinas virtuales conectadas al balanceador de carga, además de esto, el componente define el grupo de recursos que brindarán tráfico para una Load Balancing Rule determinada.

3. ¿Cuál es el propósito del Health Probe?

    - Supervisión de el estado de la aplicación, este se utiliza para detectar fallos de una aplicación en un endpoint del backend; Las respuestas de Health Probe determinan qué instancias del backend pool recibirán nuevos flujos.

4. ¿Cuál es el propósito de la Load Balancing Rule? ¿Qué tipos de sesión persistente existen, por qué esto es importante y cómo puede afectar la escalabilidad del sistema?.

    - El propósito del Load Balancing Rule es definir cómo se deberá distribuir el tráfico de las máquinas virtuales dentro del pool de backend.

    ¿Qué tipos de sesión persistente existen, por qué esto es importante y cómo puede afectar la escalabilidad del sistema?

    - Ninguno(hash-based): Peticiones recurrentes de un mismo cliente podrían ser atendidas por máquinas diferentes del backend.
    - IP del cliente: Todas las peticiones que procedan de una misma IP de origen serán atendidas por la misma máquina del backend.
    - IP y protocolo del cliente: Todas las peticiones que procedan de una misma IP y puerto de origen serán atendidas por la misma máquina del backend.

5. ¿Qué es una Virtual Network? ¿Qué es una Subnet? ¿Para qué sirven los address space y address range?

    - Virtual Network: Permite que recursos de Azure como lo son las máquinas virtuales, se puedan comunicarse de forma segura entre usuarios, con Internet y con las redes locales. Es similar a una red tradicional.
    - Subnet: Es la segmentación de una red física o red virtual, donde cada segmentación contará con su propio rango de direcciones IP.
    - Address space: Cuando se crea una virtual network se especifica el rango de direcciones IP que no se superponen unas con otras.
    - Address range: Determina el número de direcciones que se tienen o se pueden tener en un address space dependiendo de la cantidad de recursos que se necesiten en la red virtual.

6. ¿Qué son las Availability Zone y por qué seleccionamos 3 diferentes zonas?. ¿Qué significa que una IP sea zone-redundant?

    - Availability Zone: Son zonas que buscan garantizar una alta disponibilidad replicando sus aplicaciones y datos con el fin de protegerlos de puntos de fallo, estas zonas se encuentran dentro de una región. Se seleccionan 3 zonas de disponibilidad para garantizar una mejor disponibilidad y tolerancia a fallos dentro del sistema.
    - Ip Zone-Redundant: Separa física y lógicamente el gateway dentro de una región con el fin de permitir mejorar la conectividad de la red privada y disminuir fallos a nivel de zona de disponibilidad.

7. ¿Cuál es el propósito del Network Security Group?

    - Permitir o denegar el tráfico de red entrante o el tráfico de red saliente de varios tipos de recursos de Azure. Para cada regla, puede especificar un origen y destino, un puerto y un protocolo.

Lista de referencias

Microsoft Azure: Tamaños de las máquinas virtuales ampliables serie B
URL https://learn.microsoft.com/es-es/azure/virtual-machines/sizes-b-series-burstable

Azure Instances: Azure Virtual Machines Comparison
URL https://azure-instances.info

Microsoft Azure: ¿Qué es Azure Load Balancer?
URL https://learn.microsoft.com/es-es/azure/load-balancer/load-balancer-overview

Microsoft Azure Learn: Administración de una IP pública con un equilibrador de carga
URL https://learn.microsoft.com/es-es/azure/virtual-network/ip-services/configure-public-ip-load-balancer

Microsoft Azure Learn: SKU de Azure Load Balancer
URL https://learn.microsoft.com/es-es/azure/load-balancer/skus

Microsoft Azure Learn: Grupos de seguridad de red
URL https://learn.microsoft.com/es-es/azure/virtual-network/network-security-groups-overview