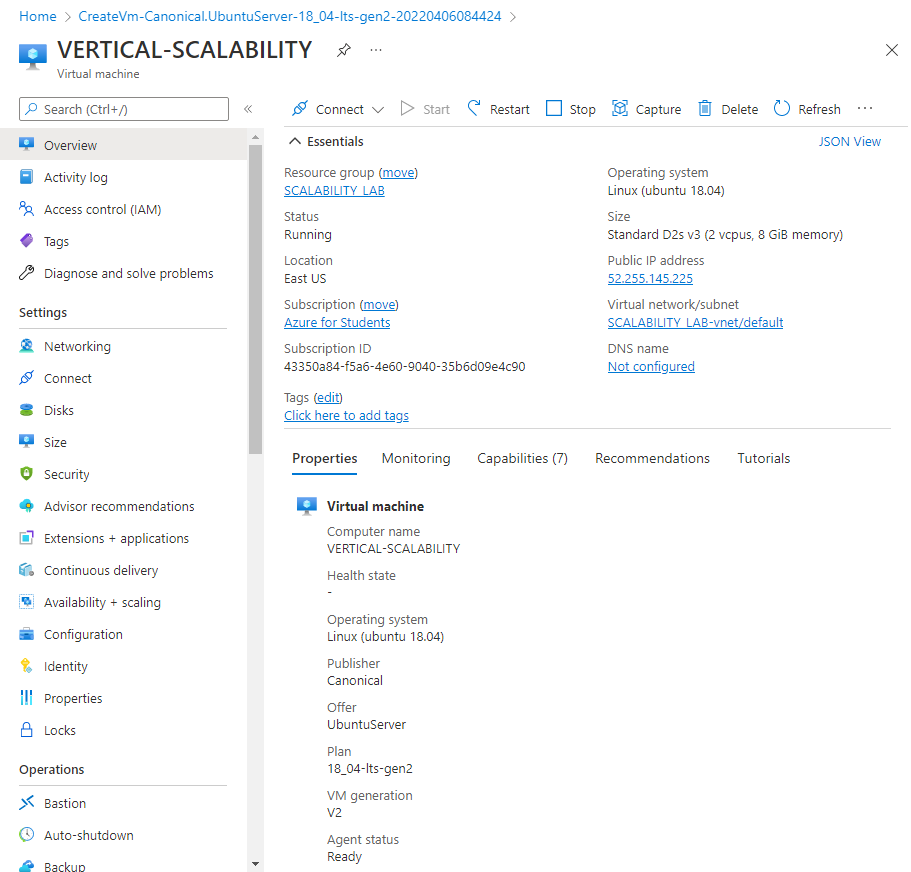
# Laboratorio 9

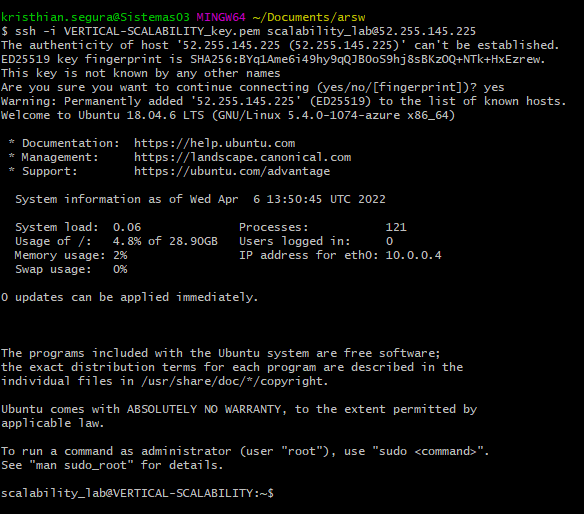
**Nombres:** Kristhian Segura, Eduardo Ospina

### **Parte 1 - Escalabilidad vertical**

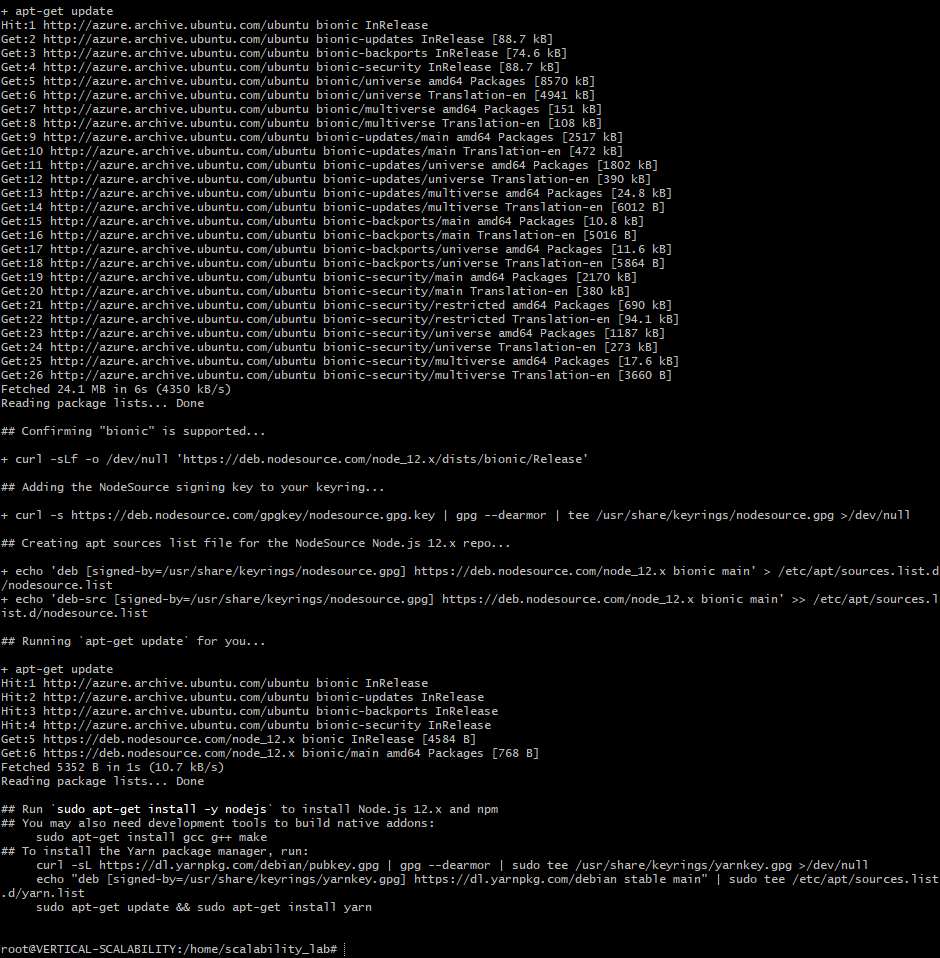
Creación del recurso

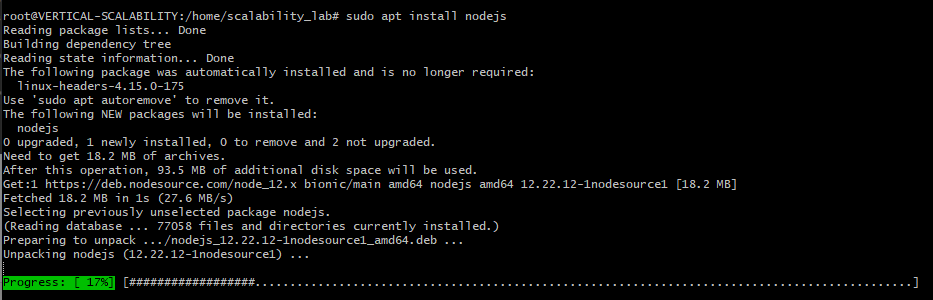


Conexión al recurso por medio del protocolo SSH

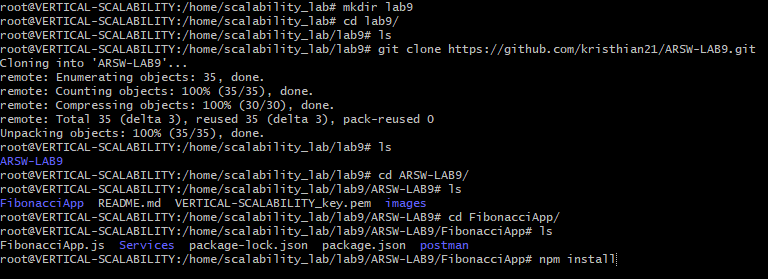


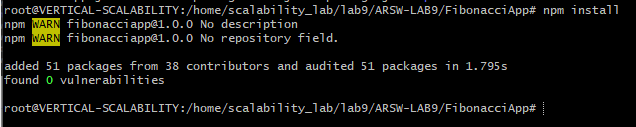
Instalación de NodeJs



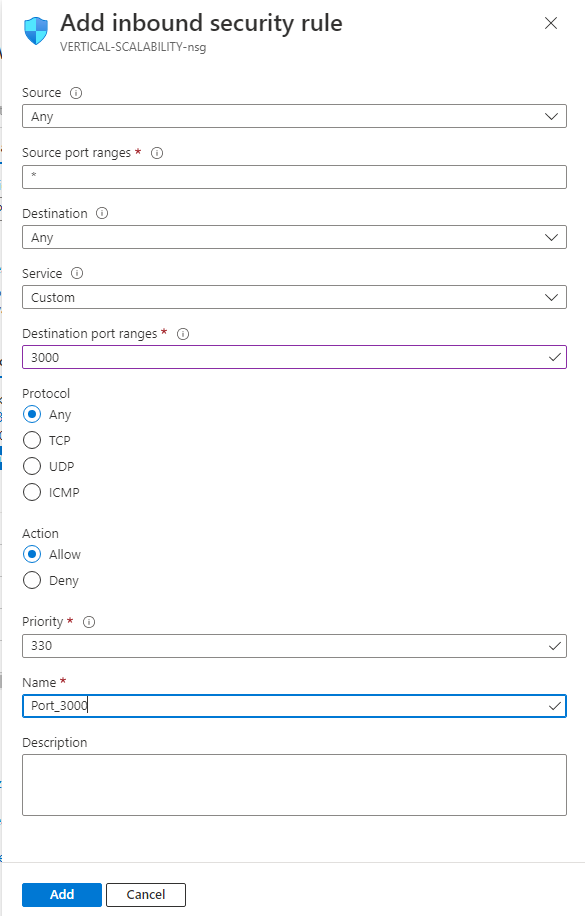




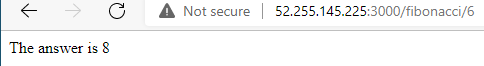




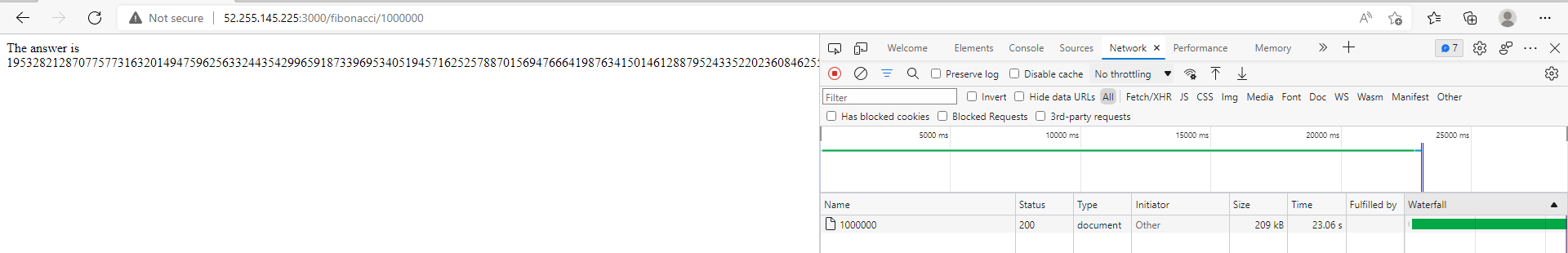




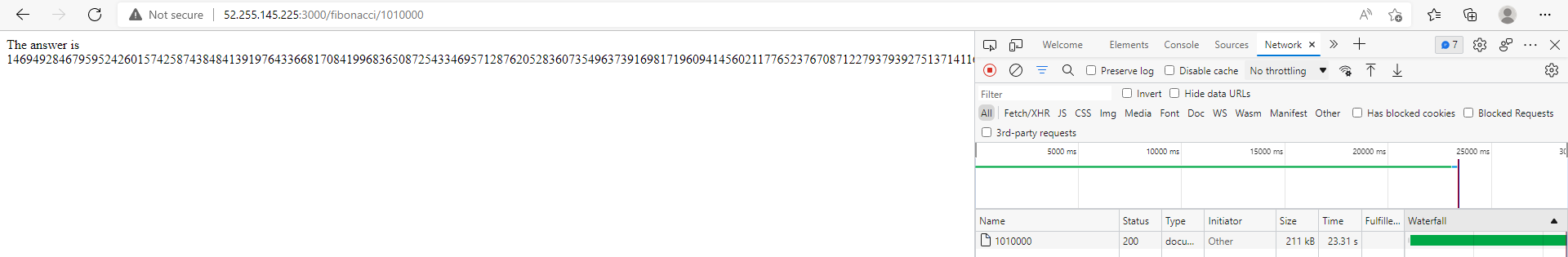




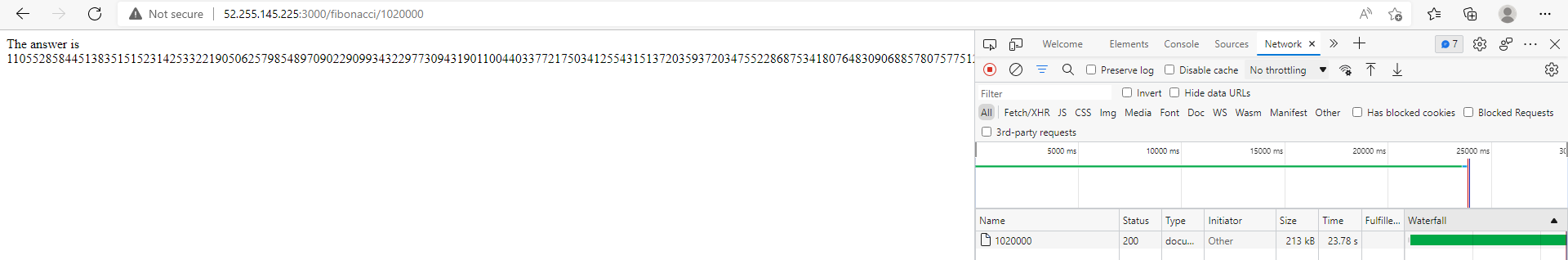
1000000



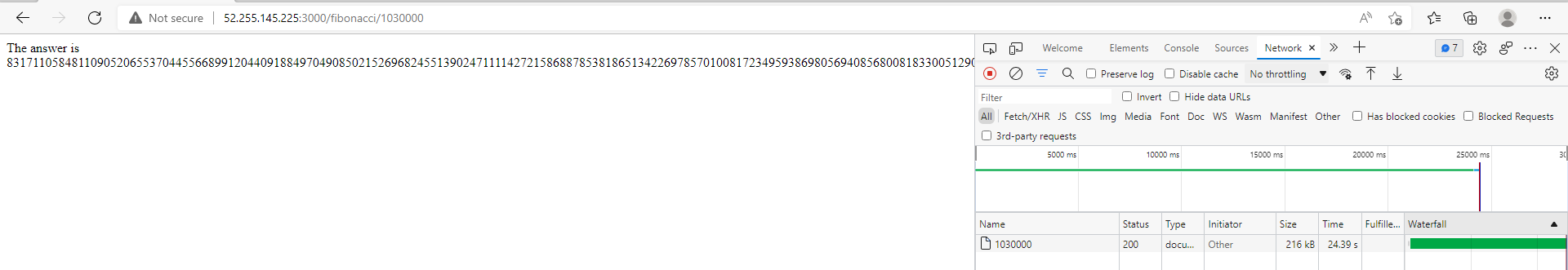
1010000



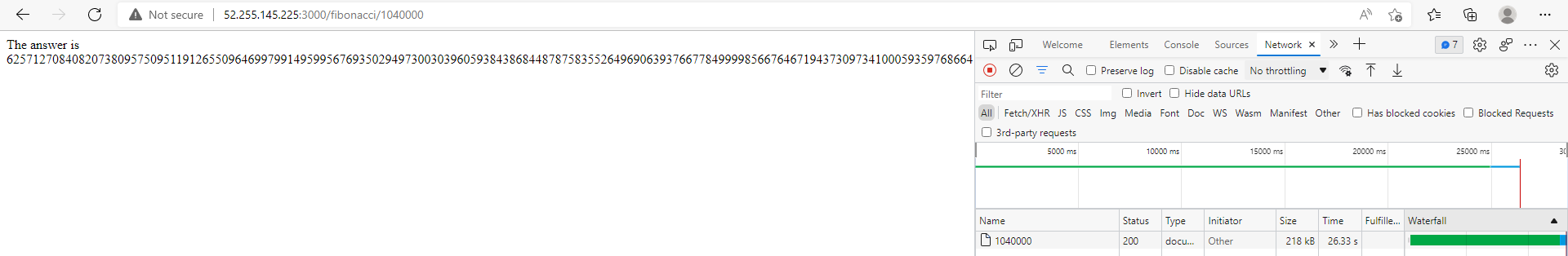
1020000



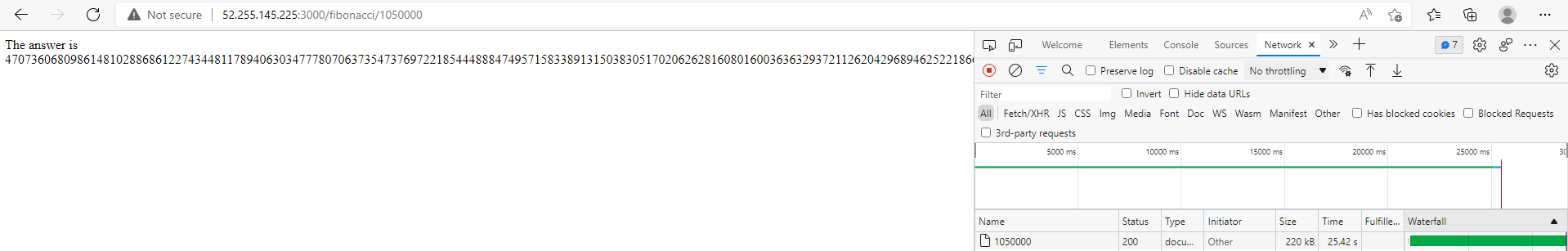
1030000



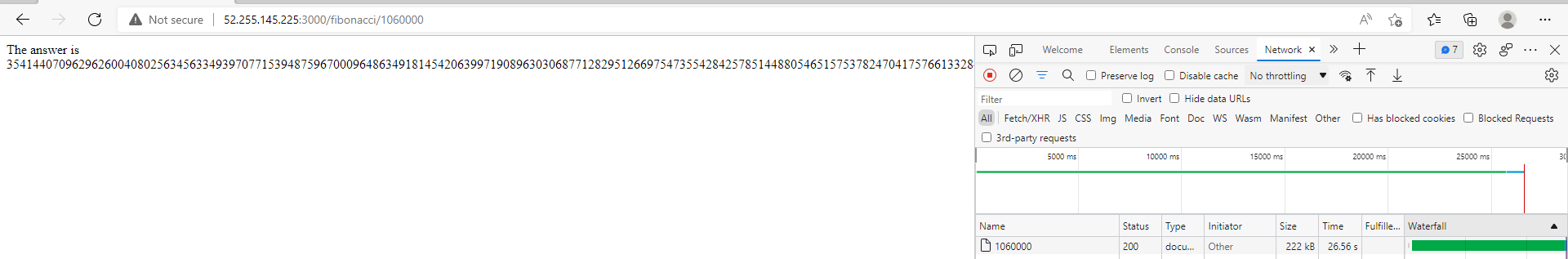
1040000



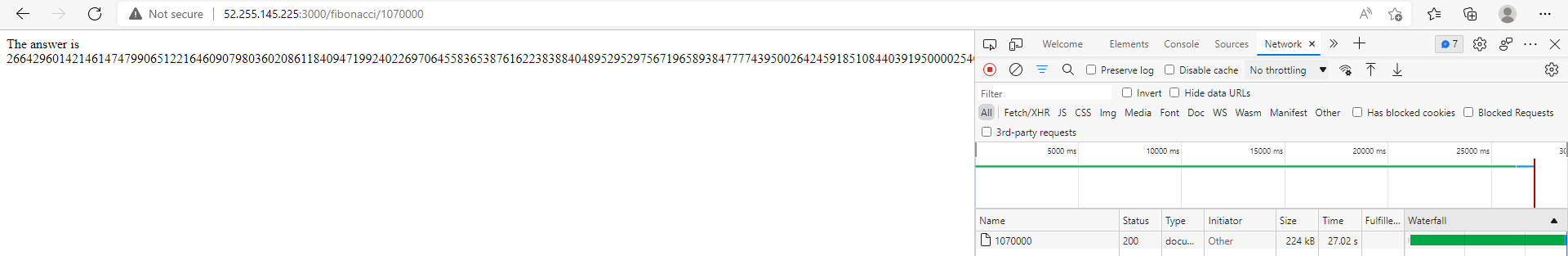
1050000



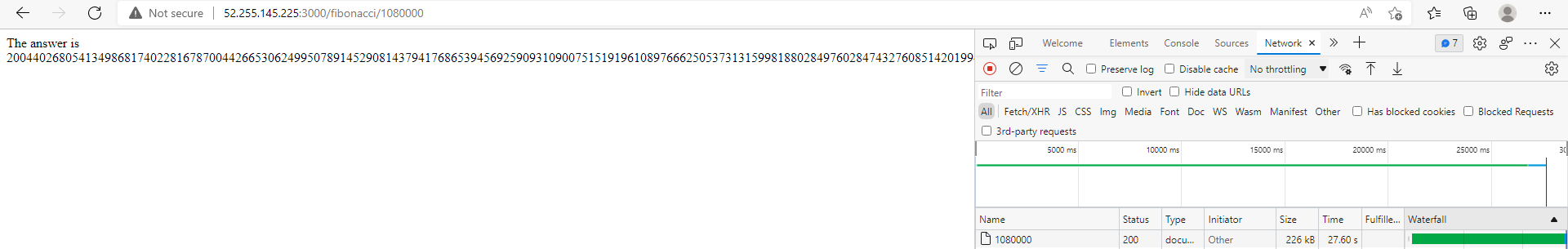
1060000



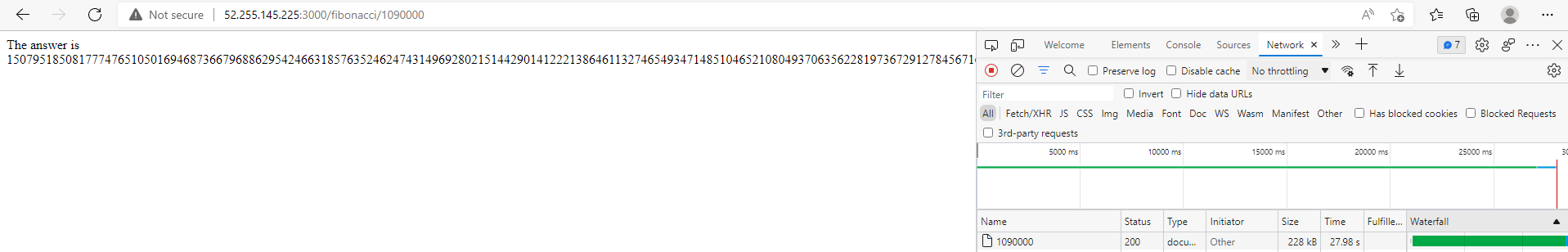
1070000



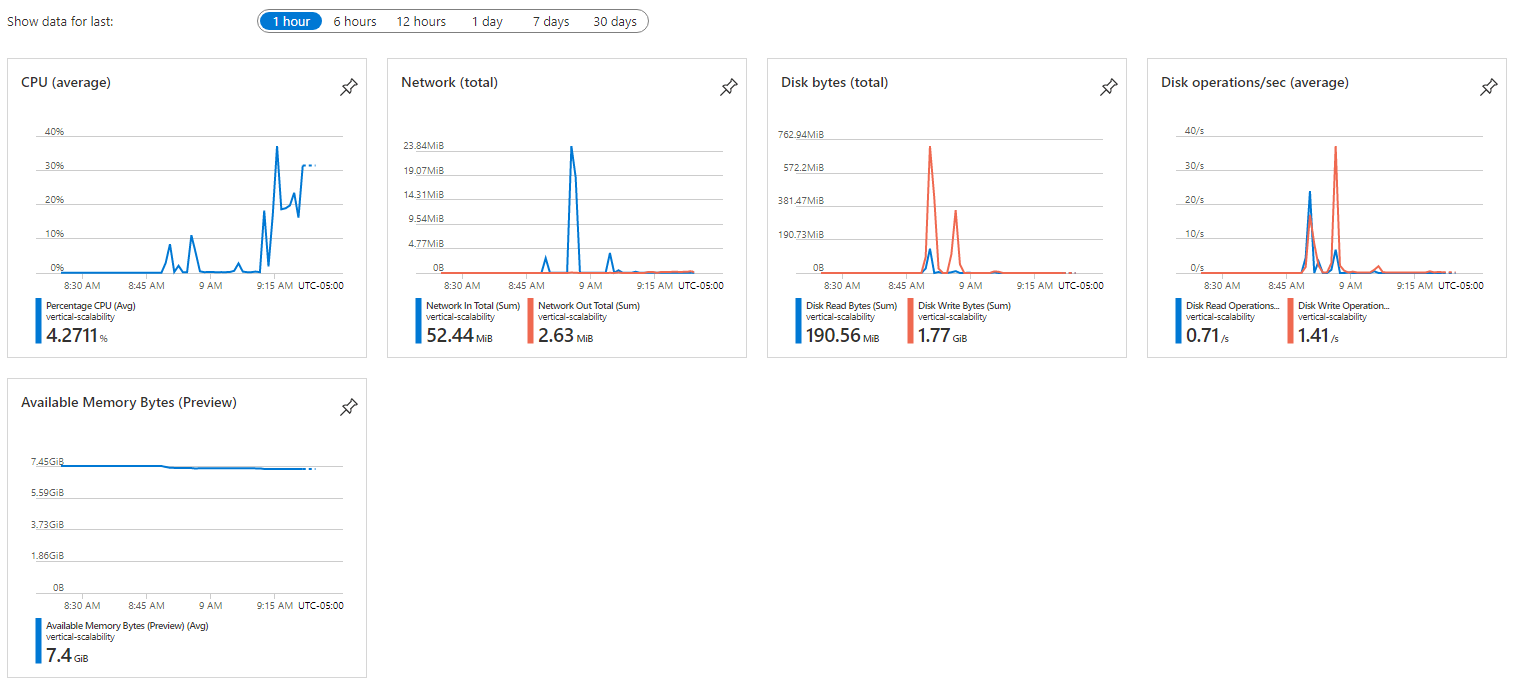
1080000



1090000



8



9

Texto

Descripción generada automáticamente con confianza media

M1 host

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

M2

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Una captura de pantalla de un celular

Descripción generada automáticamente

10

Calendario

Descripción generada automáticamente con confianza media

Repetición 7

1000000

Una captura de pantalla de una red social

Descripción generada automáticamente

1010000

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

1020000

Una captura de pantalla de una red social

Descripción generada automáticamente

1030000

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

1040000

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

1050000

Una captura de pantalla de una red social

Descripción generada automáticamente

1060000

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

1070000

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

1080000

Una captura de pantalla de una red social

Descripción generada automáticamente

1090000

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza media

Repetición 8

Captura de pantalla de computadora

Descripción generada automáticamente

Repetición 9

Una captura de pantalla de un celular

Descripción generada automáticamente

13

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

**Preguntas**

**¿Cuántos y cuáles recursos crea Azure junto con la VM?**

* Adicionalmente crea una disco, una clave de SSH, una interfaz de red, una red virtual, una dirección IP publica y un grupo de seguridad

**¿Brevemente describa para qué sirve cada recurso?**

* El disco en el cual se encuentra el sistema operativo de la VM
* Una clave SSH para conectarse de manera segura por medio del protocolo SSH a la máquina virtual
* Interfaz de red para poder conectarse a internet
* Una red virtual que conecta a los diferentes recursos dentro del mismo grupo de recursos
* Una dirección IP publica para poder comunicarse con otros equipos fuera de la red virtual
* Un grupo de seguridad para poder establecer políticas y reglas de seguridad

**¿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?**

* En el momento en el que se inicia una conexión por el protocolo SSH con una VM, se inicia un proceso para este servicio, y este proceso está asociado al usuario que realizó la conexión por lo cual al cerrar la conexión se termina este proceso
* Hay que crear un Inbound Port porque la aplicación corre en el puerto 3000, por esto hay que habilitar este puerto

**Adjunte tabla de tiempos e interprete por qué la función tarda tanto tiempo.**

|  |  |  |
| --- | --- | --- |
| **N** | **Standard\_B1ls (s)** | **Standars\_B2ms (s)** |
| 1000000 | 23.06 | 16.56 |
| 1010000 | 23.31 | 16.59 |
| 1020000 | 23.78 | 17.07 |
| 1030000 | 24.39 | 17.41 |
| 1040000 | 26.33 | 17.60 |
| 1050000 | 26.42 | 18.09 |
| 1060000 | 26.56 | 18.32 |
| 1070000 | 27.02 | 18.71 |
| 1080000 | 27.60 | 19.10 |
| 1090000 | 27.98 | 19.46 |

**Adjunte imagen del consumo de CPU de la VM e interprete por qué la función consume esa cantidad de CPU.**

**B1ls**

Captura de pantalla de computadora

Descripción generada automáticamente

**B2ms**Captura de pantalla de computadora

Descripción generada automáticamente

* Se hace mucho uso de la CPU dado que se están realizando cálculos redundantes y no se hace uso de técnicas como la programación dinámica para poder mitigar esto, adicionalmente no se está implementando concurrencia para poder hacer que se hagan los cálculos de una manera más eficiente

**Adjunte la imagen del resumen de la ejecución de Postman. Interprete:**

Tiempos de ejecución de cada petición.

**B1ls**

Una captura de pantalla de un celular

Descripción generada automáticamente

**B2ms**

Una captura de pantalla de un celular

Descripción generada automáticamente

* Se mejora considerablemente el tiempo promedio de respuesta con el tamaño B2ms
* Si hubo fallos documentelos y explique.

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

**Especificaciones**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Size** | **vCPU** | **Memory Gib** | **Temp storage(SSD) Gib** | **Base CPU perf of VM** | **Max CPU perf of VM** | **Cost/Month** |
| Standard B1ls | 1 | 0.5 | 4 | 5% | 100% | 3,80 US$ |
| Standard B2ms | 2 | 8 | 16 | 60% | 200% | 60,74 US$ |

* Ambos tamaños son ideales para cargas de trabajo que no requieran continuo rendimiento de la CPU, pequeñas bases de datos y entornos de desarrollo y pruebas

**¿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?**

* En este caso sí, las peticiones que se hacen a la aplicación FibonacciApp se responden considerablemente más rápido

**¿Qué pasa con la infraestructura cuando cambia el tamaño de la VM? ¿Qué efectos negativos implica?**

* Cuando se cambia el tamaño de la máquina se reinicia la máquina virtual, lo cual implica que se debe realizar de nuevo la conexión SSH, por esto, durante este tiempo la máquina no podrá prestar el servicio web

**¿Hubo mejora en el consumo de CPU o en los tiempos de respuesta? Si/No ¿Por qué?**

* Si, dado que la máquina virtual dispone de más recursos para realizar sus cálculos y atender a las peticiones.

**Aumente la cantidad de ejecuciones paralelas del comando de postman a 4. ¿El comportamiento del sistema es porcentualmente mejor?**

Una captura de pantalla de un celular

Descripción generada automáticamente con confianza media

Una captura de pantalla de un celular

Descripción generada automáticamente con confianza media

Una captura de pantalla de un celular

Descripción generada automáticamente con confianza media

Una captura de pantalla de un celular

Descripción generada automáticamente con confianza media

### **Parte 2 - Escalabilidad horizontal**

#### Crear el Balanceador de Carga

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

#### Crear las máquinas virtuales (Nodos)

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Prueba Load Balancer

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Una captura de pantalla de un celular

Descripción generada automáticamente con confianza media

Interfaz de usuario gráfica

Descripción generada automáticamente

Captura de pantalla de un celular

Descripción generada automáticamente

VM1

Gráfico, Gráfico de líneas

Descripción generada automáticamente

VM2

Gráfico, Gráfico de líneas

Descripción generada automáticamente con confianza media

VM3

Gráfico, Gráfico de líneas

Descripción generada automáticamente

VM4

Gráfico, Gráfico de líneas

Descripción generada automáticamente

**Preguntas**

**¿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?**

* Tipos de balanceadores
  + **Balanceador de carga interno( Privado ):** Este balanceador de carga se encarga de equilibrar la carga de trafico de una red privada ( se utilizan únicamente direcciones ip privadas en la interfaz).
  + **Balanceador de carga público:** Este balanceador de carga se encarga de equilibrar la carga de trafico de redes públicas, específicamente de la carga proveniente de internet, la dirección ip pública y el puerto son asignados.
  + **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. Encadenar un balanceador de carga de puerta de enlace a su punto final público solo requiere un clic.
* SKU( Stock Keeping Unit) Azure
  + Son uno de los elementos fundamentales para llevar el control y gestionar el stock en la bodega. SKU es el número de referencia único de un producto, según aparece registrado en el sistema de la empresa
* Tipos de SKU
  + **Estándar:** Productos estándar los cuales se pueden vender de manera individual o en paquetes
  + **Ensamblaje:** Aquellos productos que se deben ensamblar antes de un envió, todos los SKU deberán encontrarse dentro de una misma instalación
  + **Virtual:** Aquellos productos que son virtuales, es decir que no necesitan de una instalación física evitando así un nivel de inventario
  + **Componente:** Productos incluidos en paquetes, ensamblajes y colecciones, los cuales no se pueden vender de manera individual

**¿Cuál es el propósito del *Backend Pool*?**

* El pool de backend define el grupo de recursos que servirá el tráfico para una regla de equilibrio de carga determinada

**¿Cuál es el propósito del *Health Probe*?**

* Se utiliza para determinar el estado de salud de las instancias en el pool de backend. Durante la creación del equilibrador de carga, configure una sonda de salud para que el equilibrador de carga la utilice. Esta sonda de salud determinará si una instancia está sana y puede recibir tráfico

**¿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 como se deberá distribuir el tráfico de las máquinas virtuales dentro del pool de backend.
* En azure exiten 3 tipos de sesiones de persistencia:
* Ninguno(hash-based): Peticiones recurrentes de un mismo cliente podrían ser atendidas por máquinas diferentes del backend (No importa el estado de una petición).
* IP del cliente: : Todas las peticiones que procedan de una misma IP de origen serás 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, pero si vienen de la misma IP pero con un puerto de origen diferente podrían ser atendidas por otra máquina del backend.

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

* **Red Virtual:** Es una red que permite la interconexión de dispositivos y máquinas virtuales mediante software, independientemente de un ubicación física.
* **Subnet:** Es una segmentación de una red física o red virtual, estas segmentaciones contaran con su propio rango de direcciones ip según como se realice la partición de la dirección ip original.
* **Address space:** Sirve para especificar un rango de direcciones ip las cuales no se superponen unas con otras.
* **Address range:** Determina el número de direcciones que se tienen o se pueden tener en un address space y dependiendo de la cantidad de recursos que se necesiten en la red virtual, el rango aumentará o disminuirá.

**¿Qué son las *Availability Zone* y por qué seleccionamos 3 diferentes zonas?.**

* **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 y cada una de ellas se compone de uno o más centros de datos con refrigeración y redes independientes.
* Se seleccionaron 3 zonas de disponibilidad para garantizar una mejor disponibilidad y tolerancia a fallos dentro del sistema.

**¿Qué significa que una IP sea *zone-redundant*?**

* **IP zone-redundant:** Cuando utilizamos una ip zone-redundant azure separa física y lógicamente el gateway dentro de una región, lo cual permite mejorar la conectividad de la red privada y disminuye fallos a nivel de zona de disponibilidad.

**¿Cuál es el propósito del *Network Security Group*?**

* Es poder lograr filtrar el tráfico desde y hacia los recursos de una red virtual de Azure, un grupo de seguridad nos permite definir diferentes reglas de entrada y salida para que permitan o nieguen el tráfico de datos entre los recursos de Azure (Similar a las funcionalidades de un Firewall)

**Informe de newman 1 (Punto 2)**

* **Tiempos de respuesta:**

Los tiempos de respuesta de la máquina con un tamaño de B2ms están entre 46 segundos y 1 minuto. Por otro lado aplicando el balanceador de carga sobre tres máquinas con tamaño de B1ls son de entre 19 y 25 segundos

* **Cantidad de peticiones respondidas con éxito:**

La cantidad de peticiones respondidas con éxito en la máquina de tamaño B2ms fueron 23 de 40. Por otro lado aplicando el balanceador de carga sobre tres máquinas con tamaño de B1ls se respondieron con éxito 34 de 40 peticiones.

Para este caso, considerando los anteriores aspectos es considerablemente mejor aplicar escalabilidad horizontal con un balanceador de carga.