



Elasticsearch en Ubuntu 18 (VM Azure)

VM en Azure

22/10/2020

# Tabla de contenido

[Tabla de contenido 1](#_Toc54254950)

[Historial de Revisiones 2](#_Toc54254951)

[Introducción 3](#_Toc54254952)

[Manual 3](#_Toc54254953)

# Historial de Revisiones

|  |  |  |  |
| --- | --- | --- | --- |
| Fecha | Versión | Descripción | Autor |
| 22/10/2020 | 1.0 | Documento original | Brayan Quirino |

# Introducción

Pasos que seguir para levantar los servicios de Elasticseacrh en una máquina virtual de Azure, crear un índice, insertar datos y consultarlos con POSTMAN.

# Manual

1. Crear una cuenta en Azure; <https://azure.microsoft.com/en-us/free/?ref=portal>
2. Inicializar un VM[[1]](#footnote-1) con la distribución Ubuntu
3. Instalar Java en nuestra VM

sudo add-apt-repository ppa:webupd8team/java

sudo apt-get update

sudo apt-get install oracle-java8-installer

1. Instalar Elasticsearch

Descargamos la clave publica

wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add -

Creamos el repositorio

echo "deb https://artifacts.elastic.co/packages/6.x/apt stable main" | sudo tee -a /etc/apt/sources.list.d/elastic-6.x.list

Actualizamos el repositorio

sudo apt-get update && sudo apt-get install elasticsearch

Comando para iniciar Elasticsearch cuando la maquina inicie

sudo update-rc.d elasticsearch defaults 95 10

Comandos para arrancar, parar y revisar el servicio:

sudo -i service elasticsearch start

sudo -i service elasticsearch stop

sudo -i service elasticsearch status

1. Si durante los pasos anteriores existe un problema, quizá sea por la versión de Java, entonces deberás ejecutar los siguientes comandos, una vez terminado **repite el paso 4**:

sudo apt-get install default-jre

sudo apt-get install default-jdk

sudo apt-get install python-software-properties

sudo add-apt-repository ppa:webupd8team/java

sudo apt-get update

1. Una vez el servicio haya arrancado **crearemos nuestro primer index.** Es muy importante que el nombre del incide solo lleve minúsculas, para borrar el index simplemente cambiamos PUT por DELETE:

Curl -X PUT “localhost:9200/yourindex?pretty”

1. Insertamos nuestro primer **documento** en nuestro index.

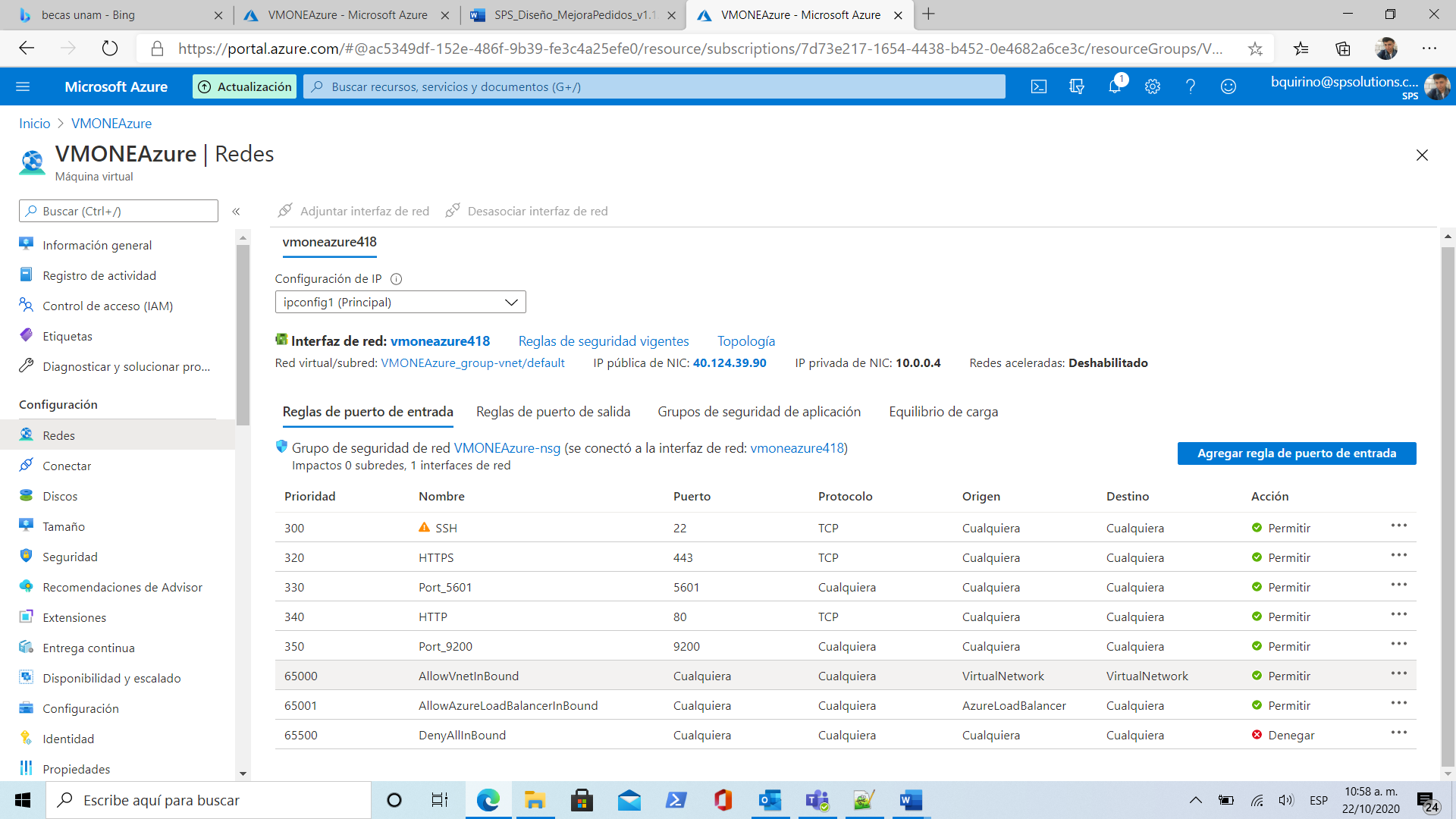
Tenemos dos formas de insertar un nuevo documento, la primera es usando el método PUT y especificando el ID de nuestro documento y la segunda es usando el método POST que por si solo nos generara un ID:

curl -X PUT "localhost:9200/marvelpeliculastest/\_doc/1?pretty" -H 'Content-Type: application/json' -d'{"idPelicula": "123abc-456def-789ghi-951jkl","nombre": "Iron Man 2","sinopsis": "Iron man con más trajes xD","fechaEstreno": "2010-05-01T00:00:00.000Z","duracionMinutos": 124,"clasificacion": ["Action","Adventure","Sci-Fi"],"idiomas": ["ESP","ENG","FRN"],"personajes": {"value": [{"idPersonaje": 1,"nombre": "Iron Man","papel": "protagonista","muereEnBatalla": false,"esHeroe": true,"objetoDePoder": "traje","debilidad": "Egocentrismo"}]},"valoraciones": {"value": [{"idValoracion": 1,"idEmpresaValoro": 5,"valoracion": 8.5,"comentarios": "Palomera"},{"idValoracion": 2,"idEmpresaValoro": 2,"valoracion": 6.7,"comentarios": "Palomera x2"}]}}'

curl -X POST "localhost:9200/marvelpeliculastest/\_doc?pretty" -H 'Content-Type: application/json' -d'{"idPelicula": "123abc-456def-789ghi-951jkl","nombre": "Iron Man 2","sinopsis": "Iron man con más trajes xD","fechaEstreno": "2010-05-01T00:00:00.000Z","duracionMinutos": 124,"clasificacion": ["Action","Adventure","Sci-Fi"],"idiomas": ["ESP","ENG","FRN"],"personajes": {"value": [{"idPersonaje": 1,"nombre": "Iron Man","papel": "protagonista","muereEnBatalla": false,"esHeroe": true,"objetoDePoder": "traje","debilidad": "Egocentrismo"}]},"valoraciones": {"value": [{"idValoracion": 1,"idEmpresaValoro": 5,"valoracion": 8.5,"comentarios": "Palomera"},{"idValoracion": 2,"idEmpresaValoro": 2,"valoracion": 6.7,"comentarios": "Palomera x2"}]}}'

1. Modificar los puertos disponibles de nuestra VM.

Yendo a la sección de Redes de nuestra maquina virtual **en Azure** agregaremos nuevas reglas de entrada:



Agregando el puerto 9200 para Elasticsearch.

1. Después editaremos el archivo elasticsearch.yml:

sudo nano /etc/elasticsearch/elasticsearch.yml

Cambiamos:

network.host: 0.0.0.0

discovery.seed\_hosts: []

Enseguida reiniciamos el servicio:

sudo service elasticsearch restart

Para checar que nuestro servicio está correcto ejecutamos:

Curl -X GET “localhost:9200”

O también podemos checarlo en nuestro buscador haciendo uso del DNS o la IP:

<http://tu_DNS_o_IP:9200/tu_Index>

1. Instalar Kibana (**Opcional por ahora**), ejecutamos los siguientes comandos:

wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add -

echo "deb https://artifacts.elastic.co/packages/6.x/apt stable main" | sudo tee -a /etc/apt/sources.list.d/elastic-6.x.list

sudo apt-get update

sudo apt-get install kibana

sudo update-rc.d kibana defaults 95 10

sudo -i service kibana start

sudo -i service kibana stop

sudo /bin/systemctl daemon-reload

sudo /bin/systemctl enable kibana.service

sudo systemctl start kibana.service

sudo systemctl stop kibana.service

1. Editaremos el archivo kibana.yml:

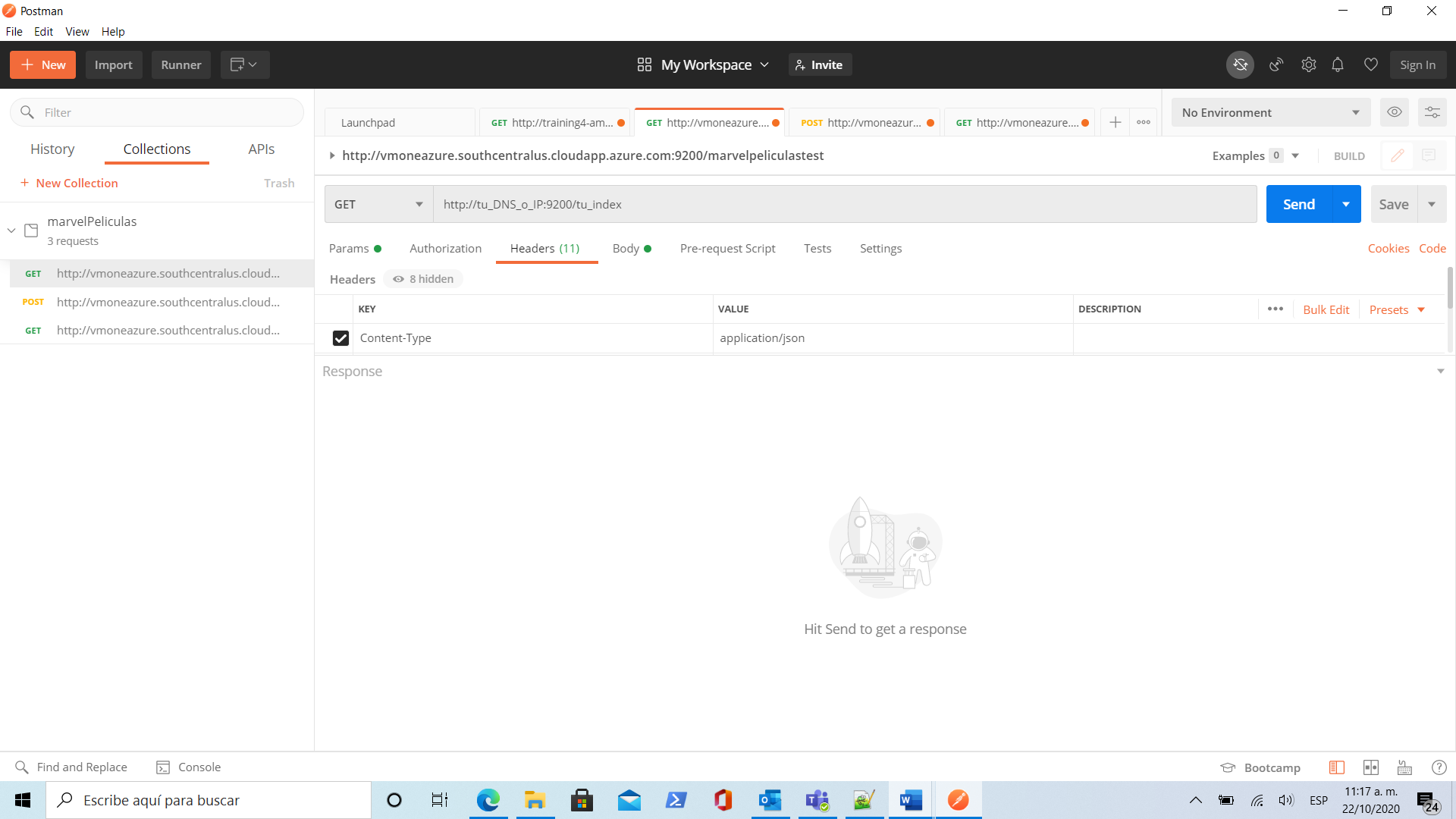
sudo nano /etc/kibana/kibana.yml

Cambiamos:

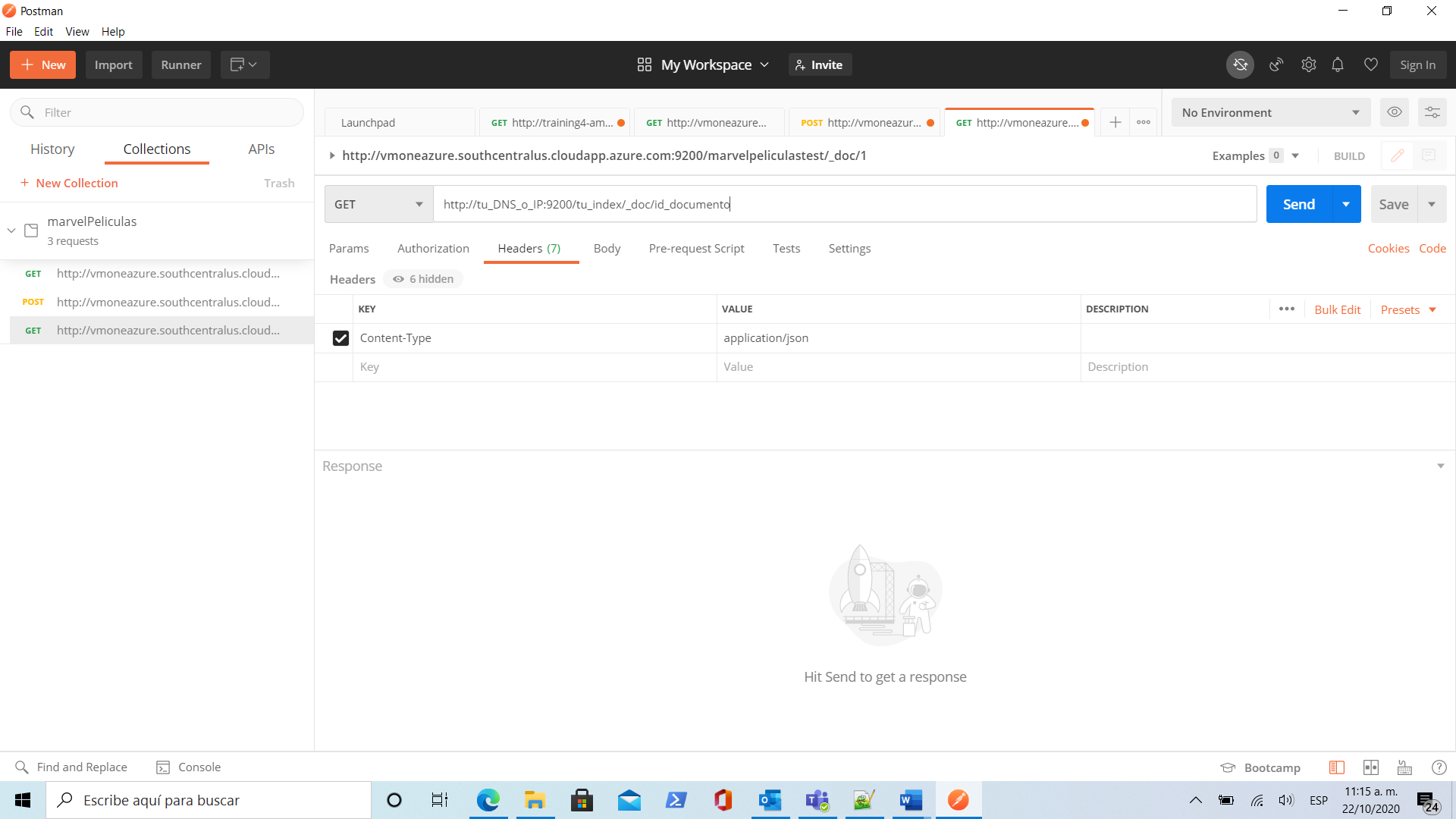
server.host: “0.0.0.0”

1. Una vez el servicio esté levantado podremos realizar operaciones con nuestro índex, haciendo uso de POSTMAN ejecutaremos las siguientes consultas:

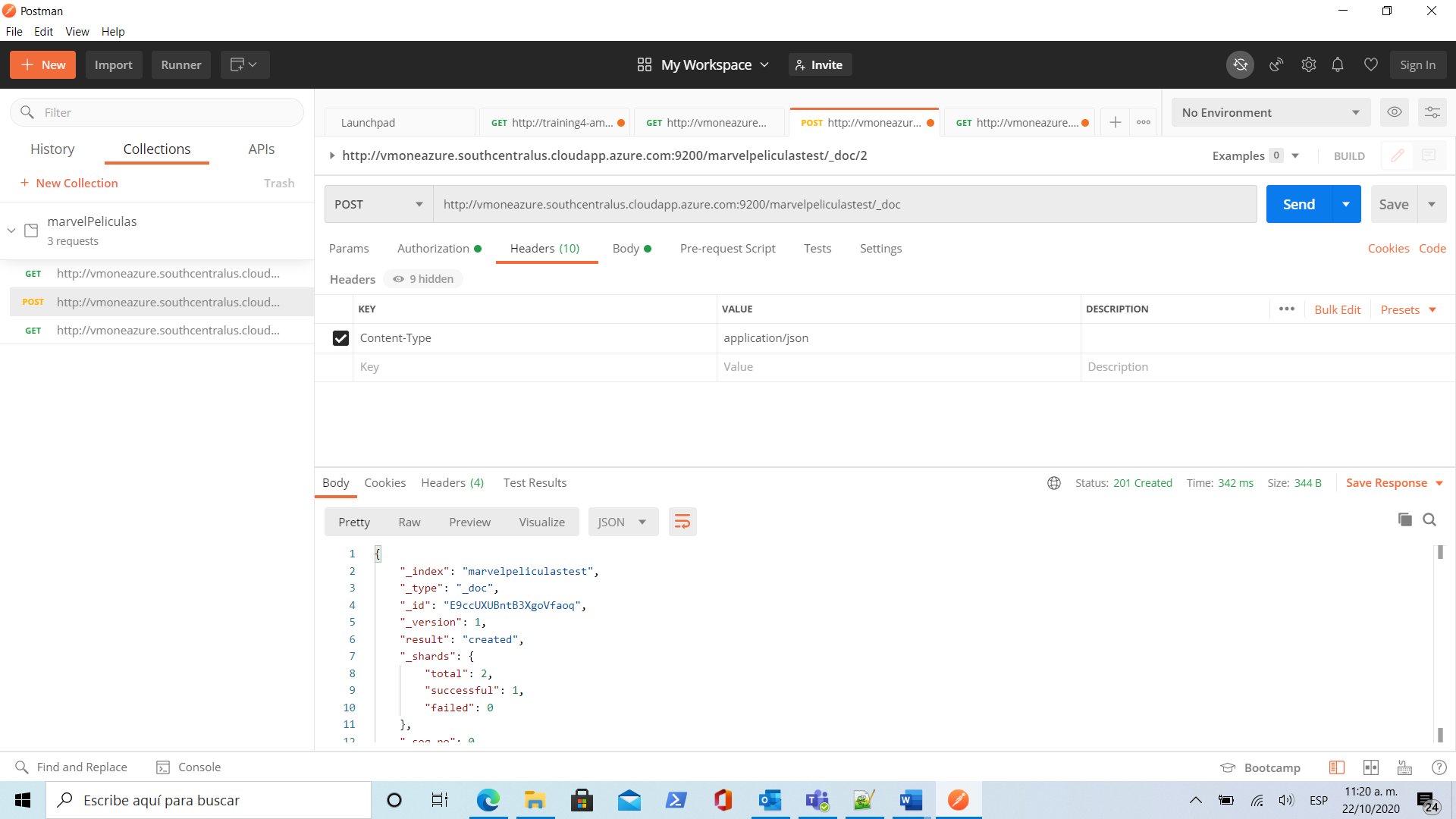
Obtener todo el Índice:



Obtener un documento especifico:



Insertar un nuevo documento con POST, en este ejemplo use un DNS e índex específicos:



1. Virtual Machine [↑](#footnote-ref-1)