



Instituto Politécnico Nacional.
Escuela Superior De Cómputo.



Materia:

Desarrollo de Sistemas Distribuidos.

Tarea 4: Implementación de un Token-Ring.

Profesor:

Pineda Guerrero Carlos.

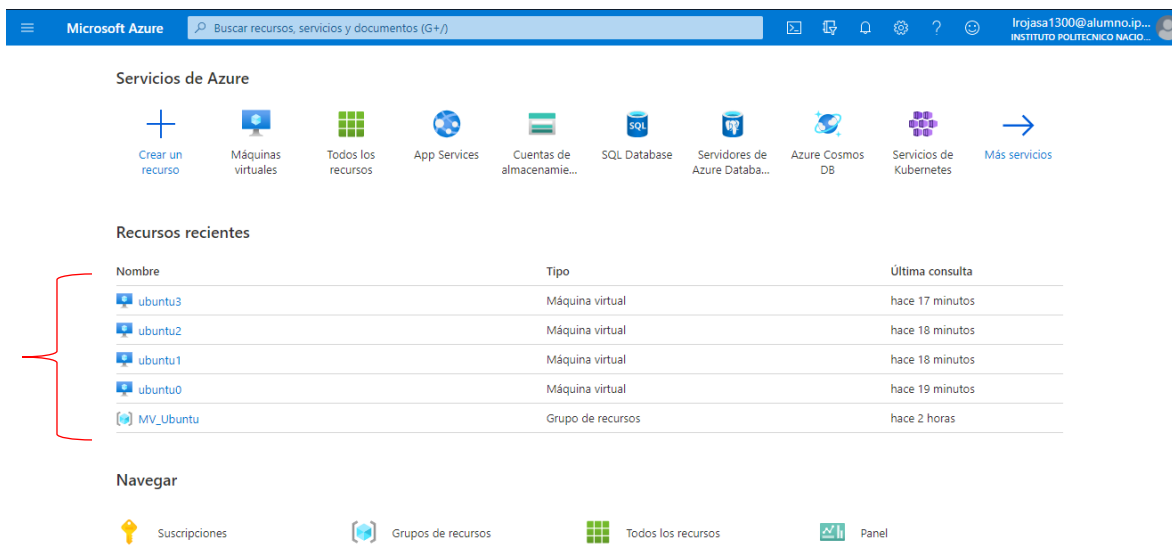
Alumno:

Luis Enrique Rojas Alvarado.

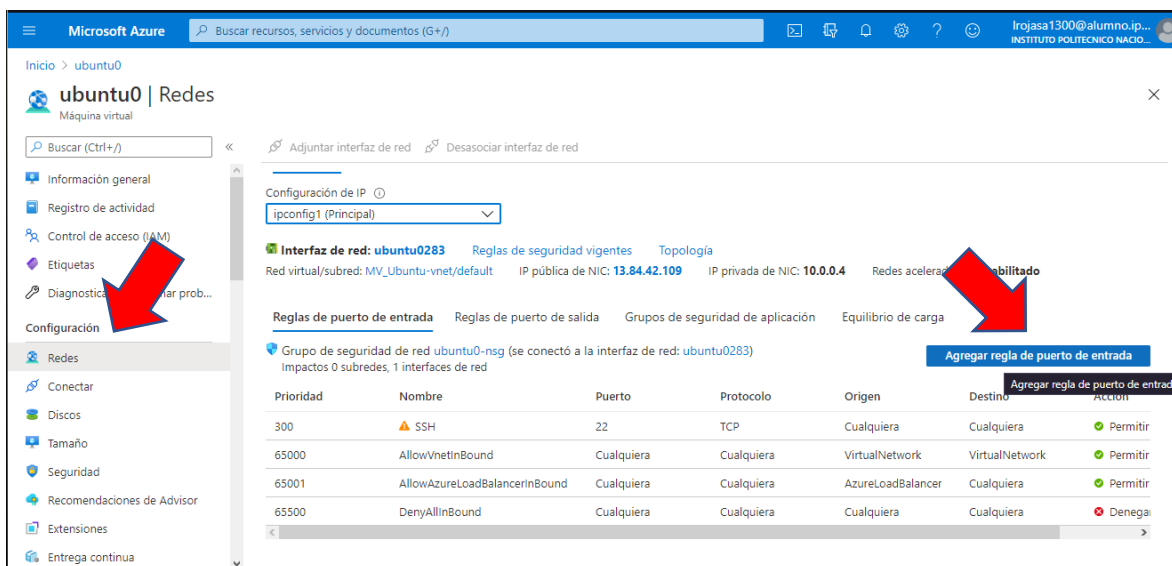
Grupo:

4CM5

Creamos las máquinas virtuales de acuerdo con las indicaciones del profesor de clases anteriores.



Una vez creadas las máquinas virtuales tenemos que hacer que se pueda recibir conexiones a través del puerto 50000 (que es con el que vamos a trabajar). Para esto, nos vamos a la sección de redes dentro de la configuración de la máquina virtual y damos click en el botón azul “agregar regla de puerto de entrada”.



Y configurar el puerto 50000 como entrada. Seleccionar también el protocolo TCP y como nombre le colocamos Port_50000 (Este procedimiento para todas las máquinas virtuales).

Agregar regla de seguridad de entrada ✕
ubuntu0-nsg

[Básica](#)

Intervalos de puertos de destino * ⓘ
50000 ✓

Protocolo *
☐ Any ☒ TCP ☐ UDP ☐ ICMP

Acción *
☒ Permitir ☐ Denegar

Prioridad * ⓘ
310

Nombre *
Port_50000 ✓

Descripción

Agregar

Iniciamos PUTTY con las direcciones IP de cada una de las máquinas virtuales que acabamos de crear:

PuTTY Configuration ? ✕

Category:

- Session
 - Logging
- Terminal
 - Keyboard
 - Bell
 - Features
- Window
 - Appearance
 - Behaviour
 - Translation
 - Selection
 - Colours
- Connection
 - Data
 - Proxy
 - Telnet
 - Rlogin
 - SSH
 - Serial

Basic options for your PuTTY session

Specify the destination you want to connect to

Host Name (or IP address)	Port
13.84.42.109	22

Connection type:
☐ Raw ☐ Telnet ☐ Rlogin ☒ SSH ☐ Serial

Load, save or delete a stored session

Saved Sessions

Default Settings

Load

Save

Delete

Close window on exit:
☐ Always ☐ Never ☒ Only on clean exit

About

Help

Open

Cancel

Accediendo a cada una de las máquinas virtuales con usuario y contraseña que configuramos al inicio:

```
ubuntu0@ubuntu0: ~  
login as: ubuntu0  
ubuntu0@13.84.42.109's password:  
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 5.4.0-1031-azure x86_64)  
  
* Documentation:  https://help.ubuntu.com  
* Management:    https://landscape.canonical.com  
* Support:        https://ubuntu.com/advantage  
  
System information as of Wed Oct 28 01:38:35 UTC 2020  
  
System load:  0.0          Processes:      108  
Usage of /:   4.4% of 28.90GB Users logged in: 0  
Memory usage: 19%         IP address for eth0: 10.0.0.4  
Swap usage:   0%  
  
0 packages can be updated.  
0 updates are security updates.  
  
The programs included with the Ubuntu system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by  
applicable law.  
  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
ubuntu0@ubuntu0:~$
```

```
ubuntu1@ubuntu1: ~  
login as: ubuntu1  
ubuntu1@104.214.96.215's password:  
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 5.4.0-1031-azure x86_64)  
  
* Documentation:  https://help.ubuntu.com  
* Management:    https://landscape.canonical.com  
* Support:        https://ubuntu.com/advantage  
  
System information as of Wed Oct 28 01:41:30 UTC 2020  
  
System load:  0.01         Processes:      109  
Usage of /:   4.4% of 28.90GB Users logged in: 0  
Memory usage: 20%         IP address for eth0: 10.0.0.5  
Swap usage:   0%  
  
0 packages can be updated.  
0 updates are security updates.  
  
The programs included with the Ubuntu system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by  
applicable law.  
  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
ubuntu1@ubuntu1:~$
```

```
ubuntu2@ubuntu2: ~  
login as: ubuntu2  
ubuntu2@52.171.32.185's password:  
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 5.4.0-1031-azure x86_64)  
  
* Documentation:  https://help.ubuntu.com  
* Management:    https://landscape.canonical.com  
* Support:        https://ubuntu.com/advantage  
  
System information as of Wed Oct 28 01:45:17 UTC 2020  
  
System load:  0.12         Processes:      110  
Usage of /:   4.4% of 28.90GB Users logged in: 0  
Memory usage: 30%         IP address for eth0: 10.0.0.6  
Swap usage:   0%  
  
* Introducing self-healing high availability clustering for MicroK8s!  
Super simple, hardened and opinionated Kubernetes for production.  
  
https://microk8s.io/high-availability  
  
0 packages can be updated.  
0 updates are security updates.  
  
The programs included with the Ubuntu system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by  
applicable law.  
  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
ubuntu2@ubuntu2:~$
```

```
ubuntu3@ubuntu3: ~  
login as: ubuntu3  
ubuntu3@40.84.232.39's password:  
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 5.4.0-1031-azure x86_64)  
  
* Documentation:  https://help.ubuntu.com  
* Management:    https://landscape.canonical.com  
* Support:        https://ubuntu.com/advantage  
  
System information as of Wed Oct 28 01:44:13 UTC 2020  
  
System load:  0.1          Processes:      109  
Usage of /:   4.4% of 28.90GB Users logged in: 0  
Memory usage: 20%         IP address for eth0: 10.0.0.7  
Swap usage:   0%  
  
* Introducing self-healing high availability clustering for MicroK8s!  
Super simple, hardened and opinionated Kubernetes for production.  
  
https://microk8s.io/high-availability  
  
0 packages can be updated.  
0 updates are security updates.  
  
The programs included with the Ubuntu system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by  
applicable law.  
  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
ubuntu3@ubuntu3:~$
```

Instalamos jdk 8 para todas las máquinas virtuales con `sudo apt install openjdk-8-jdk`. Y para comprobar usamos `java -version`

```
ubuntu0@ubuntu0: ~  
ubuntu0@ubuntu0:~$ java -version  
openjdk version "1.8.0_272"  
OpenJDK Runtime Environment (build 1.8.0_272-8u272-b10-0ubuntu1~18.04-b10)  
OpenJDK 64-Bit Server VM (build 25.272-b10, mixed mode)  
ubuntu0@ubuntu0:~$  
  
ubuntu2@ubuntu2: ~  
ubuntu2@ubuntu2:~$ java -version  
openjdk version "1.8.0_272"  
OpenJDK Runtime Environment (build 1.8.0_272-8u272-b10-0ubuntu1~18.04-b10)  
OpenJDK 64-Bit Server VM (build 25.272-b10, mixed mode)  
ubuntu2@ubuntu2:~$  
  
ubuntu1@ubuntu1: ~  
ubuntu1@ubuntu1:~$ java -version  
openjdk version "1.8.0_272"  
OpenJDK Runtime Environment (build 1.8.0_272-8u272-b10-0ubuntu1~18.04-b10)  
OpenJDK 64-Bit Server VM (build 25.272-b10, mixed mode)  
ubuntu1@ubuntu1:~$  
  
ubuntu3@ubuntu3: ~  
ubuntu3@ubuntu3:~$ java -version  
openjdk version "1.8.0_272"  
OpenJDK Runtime Environment (build 1.8.0_272-8u272-b10-0ubuntu1~18.04-b10)  
OpenJDK 64-Bit Server VM (build 25.272-b10, mixed mode)  
ubuntu3@ubuntu3:~$
```

Configuramos las variables de entorno en todas las máquinas.

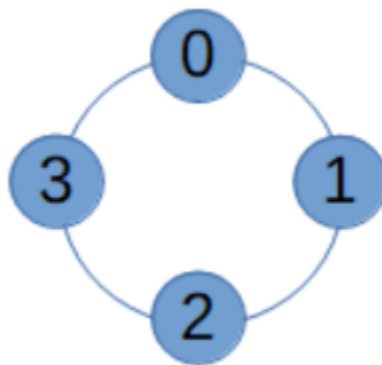
```
ubuntu0@ubuntu0: ~  
ubuntu0@ubuntu0:~$ java -version  
openjdk version "1.8.0_272"  
OpenJDK Runtime Environment (build 1.8.0_272-8u272-b10-0ubuntu1~18.04-b10)  
OpenJDK 64-Bit Server VM (build 25.272-b10, mixed mode)  
ubuntu0@ubuntu0:~$ sudo nano /etc/environment  
ubuntu0@ubuntu0:~$ source /etc/environment  
ubuntu0@ubuntu0:~$ echo $JAVA_HOME  
/usr/lib/jvm/java-8-openjdk-amd64/jre/bin/  
ubuntu0@ubuntu0:~$  
  
ubuntu2@ubuntu2: ~  
ubuntu2@ubuntu2:~$ echo $JAVA_HOME  
ubuntu2@ubuntu2:~$ java -version  
openjdk version "1.8.0_272"  
OpenJDK Runtime Environment (build 1.8.0_272-8u272-b10-0ubuntu1~18.04-b10)  
OpenJDK 64-Bit Server VM (build 25.272-b10, mixed mode)  
ubuntu2@ubuntu2:~$ javac -version  
javac 1.8.0_272  
ubuntu2@ubuntu2:~$ sudo update-alternatives --config java  
There is only one alternative in link group java (providing /usr/bin/java)  
: /usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java  
Nothing to configure.  
ubuntu2@ubuntu2:~$ sudo nano /etc/environment  
ubuntu2@ubuntu2:~$ echo $JAVA_HOME  
ubuntu2@ubuntu2:~$ sudo nano /etc/environment  
ubuntu2@ubuntu2:~$ source /etc/environment  
ubuntu2@ubuntu2:~$ echo $JAVA_HOME  
/usr/lib/jvm/java-8-openjdk-amd64/jre/bin/  
ubuntu2@ubuntu2:~$ sudo nano /etc/environment  
ubuntu2@ubuntu2:~$  
  
ubuntu1@ubuntu1: ~  
ubuntu1@ubuntu1:~$ java -version  
openjdk version "1.8.0_272"  
OpenJDK Runtime Environment (build 1.8.0_272-8u272-b10-0ubuntu1~18.04-b10)  
OpenJDK 64-Bit Server VM (build 25.272-b10, mixed mode)  
ubuntu1@ubuntu1:~$ sudo nano /etc/environment  
ubuntu1@ubuntu1:~$ source /etc/environment  
ubuntu1@ubuntu1:~$ echo $JAVA_HOME  
/usr/lib/jvm/java-8-openjdk-amd64/jre/bin/  
ubuntu1@ubuntu1:~$  
  
ubuntu3@ubuntu3: ~  
ubuntu3@ubuntu3:~$ java -version  
openjdk version "1.8.0_272"  
OpenJDK Runtime Environment (build 1.8.0_272-8u272-b10-0ubuntu1~18.04-b10)  
OpenJDK 64-Bit Server VM (build 25.272-b10, mixed mode)  
ubuntu3@ubuntu3:~$ sudo nano /etc/environment  
sudo: command not found  
ubuntu3@ubuntu3:~$ sudo nano /etc/environment  
-bash: source/etc/environment: No such file or directory  
ubuntu3@ubuntu3:~$ source /etc/environment  
ubuntu3@ubuntu3:~$ echo $JAVA_HOME  
/usr/lib/jvm/java-8-openjdk-amd64/jre/bin/  
ubuntu3@ubuntu3:~$
```

Obteniendo el archivo TokenRing.java. Usando wget para adquirirlo desde mi github personal con el comando:

Wget <https://raw.githubusercontent.com/Wicho1313/Desarrollo-de-Sistemas-Distribuidos/main/Tarea4/TokenRing.java>

```
ubuntu0@ubuntu0: ~  
ubuntu0@ubuntu0:~$ wget https://raw.githubusercontent.com/Wicho1313/Desarrollo-de-Sistemas-Distribuidos/main/Tarea4/TokenRing.java  
--2020-10-28 02:45:11-- https://raw.githubusercontent.com/Wicho1313/Desarrollo-de-Sistemas-Distribuidos/main/Tarea4/TokenRing.java  
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 151.101.48.133  
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|151.101.48.133|:443... connected.  
HTTP request sent, awaiting response... 200 OK  
Length: 3505 (3.4K) [text/plain]  
Saving to: 'TokenRing.java.1'  
  
TokenRing.java.1 100%[=====] 3.42K --.-KB/s in 0s  
  
2020-10-28 02:45:11 (50.4 MB/s) - 'TokenRing.java.1' saved [3505/3505]  
ubuntu0@ubuntu0:~$  
  
ubuntu2@ubuntu2: ~  
ubuntu2@ubuntu2:~$ wget https://raw.githubusercontent.com/Wicho1313/Desarrollo-de-Sistemas-Distribuidos/main/Tarea4/TokenRing.java  
--2020-10-28 02:45:10-- https://raw.githubusercontent.com/Wicho1313/Desarrollo-de-Sistemas-Distribuidos/main/Tarea4/TokenRing.java  
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 151.101.48.133  
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|151.101.48.133|:443... connected.  
HTTP request sent, awaiting response... 200 OK  
Length: 3505 (3.4K) [text/plain]  
Saving to: 'TokenRing.java'  
  
TokenRing.java 100%[=====] 3.42K --.-KB/s in 0s  
  
2020-10-28 02:45:10 (45.0 MB/s) - 'TokenRing.java' saved [3505/3505]  
ubuntu2@ubuntu2:~$  
  
ubuntu1@ubuntu1: ~  
ubuntu1@ubuntu1:~$ wget https://raw.githubusercontent.com/Wicho1313/Desarrollo-de-Sistemas-Distribuidos/main/Tarea4/TokenRing.java  
--2020-10-28 02:45:12-- https://raw.githubusercontent.com/Wicho1313/Desarrollo-de-Sistemas-Distribuidos/main/Tarea4/TokenRing.java  
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 151.101.48.133  
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|151.101.48.133|:443... connected.  
HTTP request sent, awaiting response... 200 OK  
Length: 3505 (3.4K) [text/plain]  
Saving to: 'TokenRing.java'  
  
TokenRing.java 100%[=====] 3.42K --.-KB/s in 0s  
  
2020-10-28 02:45:12 (47.3 MB/s) - 'TokenRing.java' saved [3505/3505]  
ubuntu1@ubuntu1:~$  
  
ubuntu3@ubuntu3: ~  
ubuntu3@ubuntu3:~$ wget https://raw.githubusercontent.com/Wicho1313/Desarrollo-de-Sistemas-Distribuidos/main/Tarea4/TokenRing.java  
--2020-10-28 02:45:07-- https://raw.githubusercontent.com/Wicho1313/Desarrollo-de-Sistemas-Distribuidos/main/Tarea4/TokenRing.java  
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 151.101.48.133  
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|151.101.48.133|:443... connected.  
HTTP request sent, awaiting response... 200 OK  
Length: 3505 (3.4K) [text/plain]  
Saving to: 'TokenRing.java'  
  
TokenRing.java 100%[=====] 3.42K --.-KB/s in 0s  
  
2020-10-28 02:45:07 (43.2 MB/s) - 'TokenRing.java' saved [3505/3505]  
ubuntu3@ubuntu3:~$
```

Se debe compilar y ejecutar en cada una de las máquinas virtuales agregando el número de nodo y la dirección IP del siguiente nodo como se muestra en la imagen:



Compilación y ejecución en las 4 máquinas virtuales:

```
ubuntu0@ubuntu0: ~  
ubuntu0@ubuntu0:~$ javac TokenRing.java  
ubuntu0@ubuntu0:~$ ls  
'TokenRing$Worker.class'  TokenRing.class  TokenRing.java  
ubuntu0@ubuntu0:~$  
  
ubuntu1@ubuntu1: ~  
ubuntu1@ubuntu1:~$ javac TokenRing.java  
ubuntu1@ubuntu1:~$ ls  
'TokenRing$Worker.class'  TokenRing.class  TokenRing.java  
ubuntu1@ubuntu1:~$  
  
ubuntu2@ubuntu2: ~  
ubuntu2@ubuntu2:~$ javac TokenRing.java  
ubuntu2@ubuntu2:~$ ls  
'TokenRing$Worker.class'  TokenRing.class  TokenRing.java  
ubuntu2@ubuntu2:~$  
  
ubuntu3@ubuntu3: ~  
ubuntu3@ubuntu3:~$ javac TokenRing.java  
ubuntu3@ubuntu3:~$ ls  
'TokenRing$Worker.class'  TokenRing.class  TokenRing.java  
ubuntu3@ubuntu3:~$
```

Dirección IP del nodo 0: 13.84.42.109

Dirección IP del nodo 1: 104.214.96.215

Dirección IP del nodo 2: 52.171.32.185

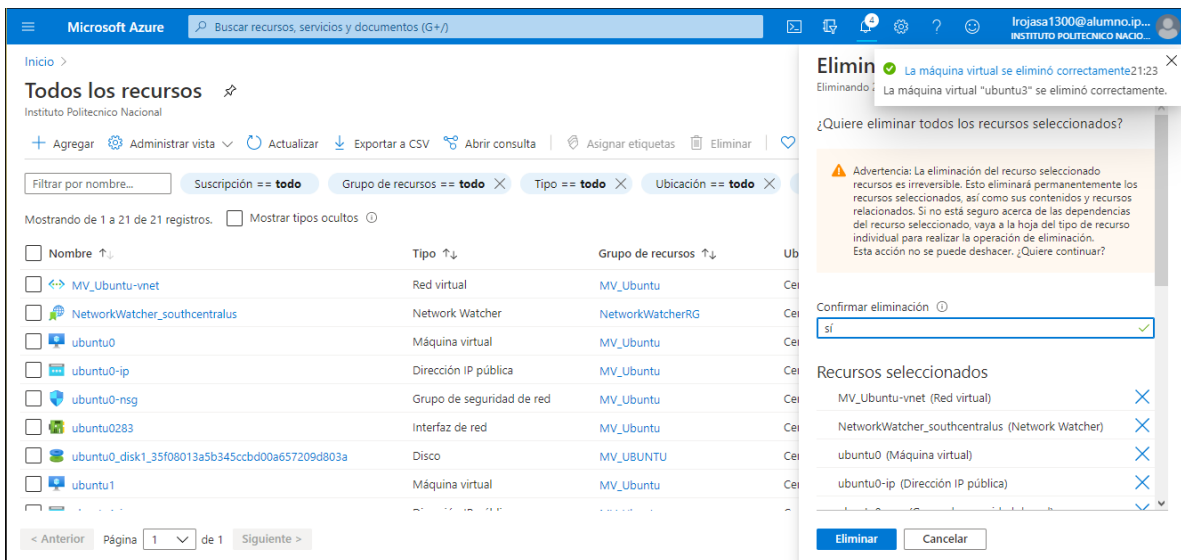
Dirección IP del nodo 3: 40.84.232.39

Ejecutando en todas las máquinas:

```
ubuntu0@ubuntu0: ~  
ubuntu0@ubuntu0:~$ java TokenRing 0 104.214.96.215  
  
ubuntu1@ubuntu1: ~  
ubuntu1@ubuntu1:~$ java TokenRing 1 52.171.32.185  
  
ubuntu2@ubuntu2: ~  
ubuntu2@ubuntu2:~$ java TokenRing 2 40.84.232.39  
  
ubuntu3@ubuntu3: ~  
ubuntu3@ubuntu3:~$ java TokenRing 3 13.84.42.109
```

The image shows four terminal windows from different virtual machines (ubuntu0, ubuntu1, ubuntu2, ubuntu3). Each window displays a list of token values. The tokens are being passed from one VM to the next in a sequence, demonstrating a ring topology. For example, ubuntu0 passes tokens to ubuntu1, which passes them to ubuntu2, and so on.

Al finalizar eliminamos todos los recursos de las máquinas virtuales para ahorrar saldo en la cuenta de Azure for students.



Cabe destacar que el programa puede ejecutarse en cualquier nodo, simplemente se queda a la escucha de la ejecución del siguiente nodo iniciando el token cuando es la primera vez, se incrementa y lo manda al siguiente, este volverá a incrementar y lo manda al siguiente nodo, así se repite infinitamente en la topología de anillo.

No funciona si no están conectadas entre sí, y cuando se ejecuta espera a que se conecte el siguiente nodo y no se sigue de largo, e iniciará, hasta que el último nodo se conecte al primero.