

# Tarea 4 - Token Ring

Pulido Bejarano Raymundo

28 de Octubre del 2020

## 1. Introducción

En esta Tarea se implementara el *Algoritmo Token Ring*, este algoritmo nos ofrece una solución al problema de *exclusión mutua* en un sistema distribuido, gestionando el acceso a los recursos.

Debido a esto y como se comento anteriormente, en esta tarea implementaremos el dicho algoritmo en el lenguaje de programación *Java*, deniendo como base la siguiente Topologia de nodos.

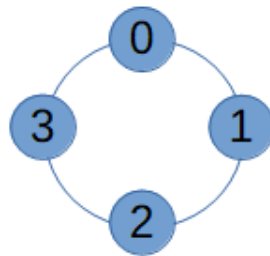


Figura 1: Topologia del Sistema

Tomando en cuenta la topologia, se crearan Maquinas Virtuales en la Nube de Microsoft Azure, una por nodo, las cuales se interconectaran para formar la topologia, y ejecutar el Algoritmo.

## 2. Desarrollo

En primera instancia se crearon las Maquinas virtuales en la nube Azure, se les asigno un usuario y una contraseña, a su vez se permitio la comunicacion

con ellas por el puerto 22 - *SSH*, y por el puerto 50000 - Puerto designado para efectuar la comunicacion entre las Maquinas Virtuales.

## 2.1. Capturas de Pantalla - Maquinas Virtuales

Se mostraran la implementacion de las maquinas virtuales en Azure




 <b>Máquina virtual</b>		 <b>Redes</b>	
Nombre del equipo	Nodo0	Dirección IP pública	40.84.138.95
Sistema operativo	Linux (ubuntu 18.04)	Dirección IP pública (IPv6) -	
Publicador	Canonical	Dirección IP privada	10.0.0.4
Oferta	UbuntuServer	Dirección IP privada (IPv6) -	
Plan	18.04-LTS	Red virtual/subred	SisDis-vnet/default
Generación de VM	V1	Nombre DNS	Configurar
Estado del agente	Ready		
Versión del agente	2.2.49.2		
Host	-		
Grupo con ubicación por proximidad	N/A		
Estado de ubicación	N/D		
		 <b>Tamaño</b>	
		Tamaño	B1s estándar
		vCPU	1
		RAM	1 GiB

Figura 2: Maquina Virtual Nodo0

Esta Maquina Virtual Nombrada como Nodo0, tiene asociada la dirección IP Publica = 40.84.139.95, por la que nos podremos comunicar por el protocolo ssh o entre las maquinas virtuales para realizar la implementacion de la aplicación.




 <b>Máquina virtual</b>		 <b>Redes</b>	
Nombre del equipo	Nodo1	Dirección IP pública	104.210.214.164
Sistema operativo	Linux (ubuntu 18.04)	Dirección IP pública (IPv6) -	
Publicador	Canonical	Dirección IP privada	10.0.0.5
Oferta	UbuntuServer	Dirección IP privada (IPv6) -	
Plan	18.04-LTS	Red virtual/subred	SisDis-vnet/default
Generación de VM	V1	Nombre DNS	Configurar
Estado del agente	Ready		
Versión del agente	2.2.49.2		
Host	-		
Grupo con ubicación por proximidad	N/A		
Estado de ubicación	N/D		
		 <b>Tamaño</b>	
		Tamaño	B1s estándar
		vCPU	1
		RAM	1 GiB

Figura 3: Maquina Virtual Nodo1

Esta Maquina Virtual Nombrada como Nodo1, tiene asociada la dirección IP Publica = 104.210.214.164, por la que nos podremos comunicar por el protocolo ssh o entre las maquinas virtuales para realizar la implementacion de la aplicación.

Máquina virtual		Redes	
Nombre del equipo	Nodo2	Dirección IP pública	13.65.171.127
Sistema operativo	Linux (ubuntu 18.04)	Dirección IP pública (IPv6) -	
Publicador	Canonical	Dirección IP privada	10.0.0.6
Oferta	UbuntuServer	Dirección IP privada (IPv6) -	
Plan	18.04-LTS	Red virtual/subred	SisDis-vnet/default
Generación de VM	V1	Nombre DNS	Configurar
Estado del agente	Ready		
Versión del agente	2.2.49.2		
Host	-		
Grupo con ubicación por proximidad	N/A		
Estado de ubicación	N/D		
		Tamaño	
		Tamaño	B1s estándar
		vCPU	1
		RAM	1 GiB

Figura 4: Maquina Virtual Nodo2

Esta Maquina Virtual Nombrada como Nodo2, tiene asociada la dirección IP Publica = 13.65.171.127, por la que nos podremos comunicar por el protocolo ssh o entre las maquinas virtuales para realizar la implementacion de la aplicación.

Máquina virtual		Redes	
Nombre del equipo	Nodo3	Dirección IP pública	40.84.236.189
Sistema operativo	Linux (ubuntu 18.04)	Dirección IP pública (IPv6) -	
Publicador	Canonical	Dirección IP privada	10.0.0.7
Oferta	UbuntuServer	Dirección IP privada (IPv6) -	
Plan	18.04-LTS	Red virtual/subred	SisDis-vnet/default
Generación de VM	V1	Nombre DNS	Configurar
Estado del agente	Ready		
Versión del agente	2.2.49.2		
Host	-		
Grupo con ubicación por proximidad	N/A		
Estado de ubicación	N/D		
		Tamaño	
		Tamaño	B1s estándar
		vCPU	1
		RAM	1 GiB

Figura 5: Maquina Virtual Nodo3

Esta Maquina Virtual Nombrada como Nodo3, tiene asociada la dirección IP Publica = 140.84.236.189, por la que nos podremos comunicar por el protocolo ssh o entre las maquinas virtuales para realizar la implementacion de la aplicación.

## 2.2. Preparaciones

Una vez creadas las maquinas virtuales, se les debio hacer una preparacion para poder ejecutar la JVM, como tambien enviarles la aplicacion a ejecutar para lo que se realizo lo siguiente.

Como primera instancia se accedio a todas las Maquinas Virtuales y se creo una carpeta llamada *Ray*, para alojar el codigo a ejecutar, esto con el siguiente comando.

```
ssh Ray@<DireccionIP>
```

Una vez esto, ingresar la contraseña de la maquina virtual, una vez dentro crear el directorio y salir.

Una vez realizado esto en todas las maquinas virtuales, se les copio el codigo fuentea cada una de las maquinas de la siguiente forma.

```
sshpass -p <Contraseña> scp *.java Ray@<DireccionIP>:~/Ray
```

## 2.3. Capturas de Pantalla - Ejecucion

Una vez teniendo todo preparado, se procedio a realizar la ejecucion del programa, dando como resultados lo siguiente:

```
Ray@Nodo0:~/Ray$ java Token 0 104.210.214.164
1
5
9
13
17
21
25
29
33
37
41
45
49
53
57
61
65
69
73
77
81
85
89
```

Figura 6: Ejecución Nodo 0

Se puede mostrar la ejecucion del nodo 0 y conectandose al siguiente nodo, usando su Ip, donde tambien empieza el flujo con el token en 1, y haciendo un flujo de 4 en 4 debido al numero de nodos en la topologia, se muestra un flujo sin fin, ya que este algoritmo esta destinado ah correr siempre cordinando el acceso a los recursos.

```
Ray@Nodo1:~/Ray$ java Token 1 13.65.171.127
2
6
10
14
18
22
26
30
34
38
42
46
50
54
58
62
66
70
74
78
82
86
90
```

Figura 7: Ejecución Nodo 1

Se puede mostrar la ejecucion del nodo 1 y conectandose al siguiente nodo, usando su Ip, donde tambien empieza el flujo mostrando el token en

2, y haciendo un flujo de 4 en 4 debido al numero de nodos en la topologia, se muestra un flujo sin fin, ya que este algoritmo esta destinado ah correr siempre cordinando el acceso a los recursos.

```
Ray@Nodo2:~/Ray$ java Token 2 40.84.236.189
3
7
11
15
19
23
27
31
35
39
43
47
51
55
59
63
67
71
75
79
83
87
91
```

Figura 8: Ejecución Nodo 2

Se puede mostrar la ejecucion del nodo 2 y conectandose al siguiente nodo, usando su Ip, donde tambien empieza el flujo mostrando el token en 3, y haciendo un flujo de 4 en 4 debido al numero de nodos en la topologia, se muestra un flujo sin fin, ya que este algoritmo esta destinado ah correr siempre cordinando el acceso a los recursos.

```
Ray@Nodo3:~/Ray$ java Token 3 40.84.138.95
4
8
12
16
20
24
28
32
36
40
44
48
52
56
60
64
68
72
76
80
84
88
```

Figura 9: Ejecución Nodo 3

Se puede mostrar la ejecución del nodo 3 y conectándose al siguiente nodo, usando su Ip, donde también empieza el flujo mostrando el token en 4, y haciendo un flujo de 4 en 4 debido al número de nodos en la topología, se muestra un flujo sin fin, ya que este algoritmo está destinado a correr siempre coordinando el acceso a los recursos.