



INSTITUTO POLITÉCNICO NACIONAL
ESCUELA SUPERIOR DE CÓMPUTO

"Tarea 4. Implementación de un token-ring"

Alumno:

Lara Cázares Jaime Arturo

Materia:

Desarrollo de sistemas distribuidos

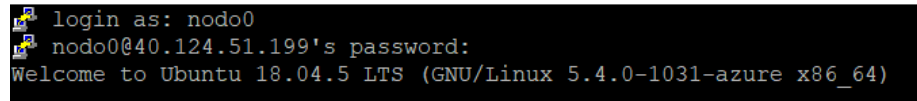
Grupo:

4CM3

Profesor: Pineda Guerrero Carlos

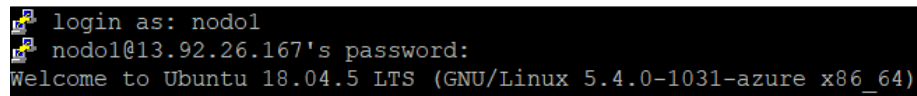
1. Capturas de pantallas.

Se comienza accediendo con ayuda de **Putty** a las maquinas virtuales. La Figuras 1, 2, 3 y 4 muestran el acceso a cada maquina virtual.



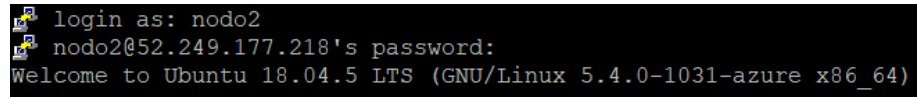
```
login as: nodo0
nodo0@40.124.51.199's password:
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 5.4.0-1031-azure x86_64)
```

Figura 1: Acceso a la maquina virtual 0 (nodo 0).



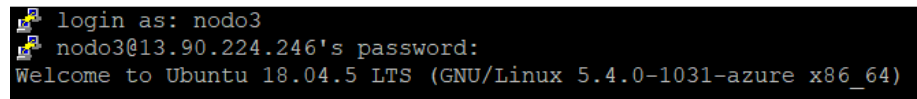
```
login as: nodo1
nodo1@13.92.26.167's password:
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 5.4.0-1031-azure x86_64)
```

Figura 2: Acceso a la maquina virtual 1 (nodo 1).



```
login as: nodo2
nodo2@52.249.177.218's password:
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 5.4.0-1031-azure x86_64)
```

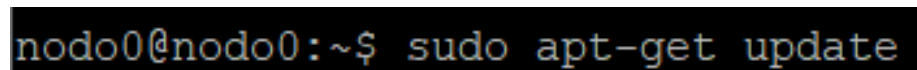
Figura 3: Acceso a la maquina virtual 2 (nodo 2).



```
login as: nodo3
nodo3@13.90.224.246's password:
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 5.4.0-1031-azure x86_64)
```

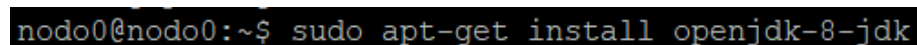
Figura 4: Acceso a la maquina virtual 3 (nodo 3).

Una vez teniendo las 4 maquinas virtuales creadas se procede a instalar el JDK, para ello en cada una se hace una actualización, Figura 6, y después se instala como se muestra en la Figura ???. Este proceso se realiza en todas las maquinas virtuales.



```
nodo0@nodo0:~$ sudo apt-get update
```

Figura 5: Actualización.



```
nodo0@nodo0:~$ sudo apt-get install openjdk-8-jdk
```

Figura 6: Instalación del JDK.

Aprovechando la portabilidad de Java, se compila en una maquina personal (ver la Figura 7) y con ayuda de un repositorio en *Github* se tiene facilidad de acceso al programa.

```
D:\Documentos\ESCOM\Distribuidos\Desarrollo-de-sistemas-distribuidos\Tarea4>javac Token.java
D:\Documentos\ESCOM\Distribuidos\Desarrollo-de-sistemas-distribuidos\Tarea4>
```

Figura 7: Compilación del programa Token.

Se utiliza el comando git para poder obtener el programa directamente de mi repositorio personal. En la Figura 8 se aprecian las terminales que clonaron el repositorio y están listas para correr el programa Token.

```
git clone https://github.com/artLara/Desarrollo-de-sistemas-distribuidos.git
```

The figure displays four terminal windows arranged in a 2x2 grid, showing the process of cloning a repository and compiling a Java program. The top-left window shows the cloning of the repository 'Desarrollo-de-sistemas-distribuidos' from GitHub. The top-right window shows the cloning of the repository 'Desarrollo-de-sistemas-distribuidos' from GitHub. The bottom-left window shows the cloning of the repository 'Desarrollo-de-sistemas-distribuidos' from GitHub. The bottom-right window shows the cloning of the repository 'Desarrollo-de-sistemas-distribuidos' from GitHub. All windows show the same output, indicating that the repository was successfully cloned and the program was compiled.

Figura 8: Obtención del programa clonando el repositorio github.

Se prosigue a la ejecución del programa Token en la maquina virtual del **nodo 0**, ver en la Figura 9 que el nodo 0 esta en espera de los demás nodos, pasando como primer parámetro el **id del nodo** y como segundo la **dirección de IP**. Las Figuras 10, 11, 12 y 13 muestran las ejecuciones del programa en los distintos nodos.

```
nodo0@nodo0:~/Desarrollo-de-sistemas-distribuidos/Tarea4$ java Token 0 13.92.26.167
Iniciado nodo: 0
Direccion IP del siguiente nodo: 13.92.26.167
nodo0@nodo0:~/Desarrollo-de-sistemas-distribuidos/Tarea4$ java Token 0 13.92.26.167
Iniciado nodo: 0
Direccion IP del siguiente nodo: 13.92.26.167
```

Figura 9: Nodo en 0 en espera de los demás nodos.

```
nodo0@nodo0:~/Desarrollo-de-sistemas-distribuidos/Tarea4$ java Token 0 13.92.26.167
Iniciado nodo: 0
Direccion IP del siguiente nodo: 13.92.26.167
nodo0@nodo0:~/Desarrollo-de-sistemas-distribuidos/Tarea4$ java Token 0 13.92.26.167
Iniciado nodo: 0
Direccion IP del siguiente nodo: 13.92.26.167
Token=1
Token=5
Token=9
Token=13
Token=17
Token=21
Token=25
Token=29
Token=33
Token=37
Token=41
Token=45
Token=49
Token=53
Token=57
Token=61
Token=65
Token=69
```

Figura 10: Ejecución del programa Token en el nodo 0.

```
nodo1@nodo1:~/Desarrollo-de-sistemas-distribuidos/Tarea4$ java Token 1 52.249.177.218
Iniciado nodo: 1
Direccion IP del siguiente nodo: 52.249.177.218
Token=2
Token=6
Token=10
Token=14
Token=18
Token=22
Token=26
Token=30
Token=34
Token=38
Token=42
Token=46
Token=50
Token=54
Token=58
Token=62
Token=66
Token=70
Token=74
Token=78
```

Figura 11: Ejecución del programa Token en el nodo 1.

```
nodo2@nodo2:~/Desarrollo-de-sistemas-distribuidos/Tarea4$ java Token 2 13.90.224.246
Iniciado nodo: 2
Direccion IP del siguiente nodo: 13.90.224.246
Token=3
Token=7
Token=11
Token=15
Token=19
Token=23
Token=27
Token=31
Token=35
Token=39
Token=43
Token=47
Token=51
Token=55
Token=59
Token=63
Token=67
Token=71
Token=75
Token=79
Token=83
```

Figura 12: Ejecución del programa Token en el nodo 2.

```
nodo3@nodo3:~/Desarrollo-de-sistemas-distribuidos/Tarea4$ java Token 3 40.124.51.199
Iniciado nodo: 3
Direccion IP del siguiente nodo: 40.124.51.199
Token=4
Token=8
Token=12
Token=16
Token=20
Token=24
Token=28
Token=32
Token=36
Token=40
Token=44
Token=48
Token=52
Token=56
Token=60
Token=64
Token=68
Token=72
Token=76
Token=80
Token=84
```

Figura 13: Ejecución del programa Token en el nodo 3.

Por último en la Figura 14 se observa la captura de todas las terminales ejecutando algoritmo *Token Ring*.

The figure displays four terminal windows arranged in a 2x2 grid, each showing the execution of a Java program named 'Token' on a different node. The windows are titled 'nodo0@nodo0', 'nodo1@nodo1', 'nodo2@nodo2', and 'nodo3@nodo3'. Each window shows the following sequence of output:

- Initial command: `java Token 0 13.92.26.167` (for nodo0), `java Token 1 52.249.177.218` (for nodo1), `java Token 2 13.90.224.246` (for nodo2), and `java Token 3 40.124.51.199` (for nodo3).
- Initial message: `Iniciado nodo: 0` (or 1, 2, 3 respectively).
- Direction message: `Direccion IP del siguiente nodo: 13.92.26.167` (or the next node's IP respectively).
- Token messages: A series of `Token=` followed by a number, indicating the token's value at each step. For example, nodo0 shows `Token=5`, `Token=9`, `Token=13`, etc., up to `Token=65`. nodo1 shows `Token=6` through `Token=78`. nodo2 shows `Token=7` through `Token=79`. nodo3 shows `Token=8` through `Token=84`.

Figura 14: Captura de todas las terminales de la ejecución del programa *Token Ring*.