



Instituto Politécnico Nacional Escuela Superior de Cómputo

Desarrollo de Sistemas Distribuidos

Actividad: Replicación de un servidor en la nube

Curso impartido por el profesor: Pineda Guerrero Carlos

Grupo: 4CV1

21/01

Alumno: Adrian González Pardo



Ultima fecha modificado: 8 de enero de 2021

1. Desarrollo

Para esta tarea fue necesario hacer uso de dos maquinas virtuales con Linux y la maquina local con Windows (Winbugs), la cual debe tener instalados los archivos de las aplicaciones contenidas de Putty como *putty.exe*, y *psftp.exe* las cuales pueden ser trabajadas desde las variables de entorno para ser ejecutadas en cmd o en Powershell de forma más sencilla, sobre todo *psftp.exe* que necesita al igual que ssh un usuario y una dirección IP o de host.

2. Códigos y scripts para funcionamiento

2.1. Código modificado Cliente2.java

```
1 public static void main(String[] args) throws Exception{
2     /* Aniadir estas lineas */
3     if(args.length<1){
4         System.out.println("Error\n\tUsage: java Cliente2 <IP>");
5         System.exit(1);
6     }
7     Socket conexion = null;
8     /* Aniadir esto */
9     System.out.println("Conectando a: "+args[0]);
10    for(;;){
11        try{
12            conexion = new Socket(args[0],50000); /* Modificacion aqui */
13            break;
14        }catch (Exception e){
15            Thread.sleep(100);
16        }
17    } /* Resto de codigo */
18    System.out.println("Saliendo...");
19 }
```

Para el lado de las maquinas virtuales se hizo uso de dos scripts los cuales permiten instalar de forma rápida y sencilla todo lo que necesitaremos, pero para ejecutar estos scripts es necesario ya haber mandado todos los archivos con los que va a trabajar la maquina virtual.

2.2. Script para VM 1

```
1 #!/usr/bin/env bash
2 # @author Adrian Gonzalez Pardo
3 if [ $# -ne 1 ];then
4     echo -e "Error\n\tUsage: $0 <IP_Second>"
5     exit 1
6 fi
7 echo "Instalacion de archivos en la VM"
8 sudo apt update && sudo apt install openjdk-8-jdk-headless build-essential -y
9 echo "Modificacion del archivo"
10 sed -i -e 's/ServerSocket servidor = new ServerSocket(50000);/ServerSocket
    servidor = new ServerSocket(50001);/g' Servidor2.java
11 echo "Compilacion de archivos java"
12 javac Servidor2.java && javac SimpleProxyServer.java
13 echo "Corriendo binarios"
14 java Servidor2&
15 java SimpleProxyServer $1 50000 50000 50001&
16 echo "Fin"
```

2.3. Script para VM 2

```
1 #!/usr/bin/env bash
2 # @author Adrian Gonzalez Pardo
3 echo "Instalacion de archivos en la VM"
4 sudo apt update && sudo apt install openjdk-8-jdk-headless build-essential -y
5 echo "Compilacion de archivos java"
```

```

6 javac Servidor2.java && javac SimpleProxyServer.java
7 echo "Corriendo binarios"
8 java Servidor2&
9 echo "Fin"

```

3. Capturas

3.1. VM

The screenshot shows the 'Casillas de selección' (Selection boxes) for creating a VM in the Azure portal. The options are as follows:

- Subscription:** Azure para estudiantes
- Region:** Distribuidos
- Image:** azureClient2
- Redundancy:** (US) Centro-Sur de EE. UU.
- Image Gallery:** No se requiere redundancia de la infraestructura
- Image Version:** Ubuntu Server 18.04 LTS - Gen1
- Size:** Standard_B1s - 1 vcpu, 1 GiB de memoria (MXN 176.11/mes)

Figura 1: Casillas de selección para la creación de la VM

The screenshot shows the 'Cuenta de administrador' (Administrator account) section of the VM creation page. The options are as follows:

- Tipo de autenticación:** Contraseña (selected)
- Nombre de usuario:** adrian
- Contraseña:** [Redacted]
- Confirmar contraseña:** [Redacted]

Figura 2: Casillas de selección para la creación de la VM

The screenshot shows the 'Discos' (Disks) section of the VM creation page. The options are as follows:

- Tipo de disco del sistema operativo:** HDD estándar
- Tipo de cifrado:** (Predeterminado) Cifrado en reposo con una clave administrada por la pl...

Figura 3: Casillas de selección para la creación de la VM

Configure las opciones de supervisión y administración de la VM.

Azure Security Center

Azure Security Center proporciona características unificadas de administración de la seguridad y protección contra amenazas en todas las cargas de trabajo de nube híbrida. [Más información](#)

✔ La suscripción está protegida por el plan básico de Azure Security Center.

Supervisión

- Diagnósticos de arranque ⓘ
- ☐ Habilitar con la cuenta de almacenamiento administrada (recomendado)
 - ☐ Habilitar con la cuenta de almacenamiento personalizada
 - ☒ Deshabilitar

Figura 4: Casillas de selección para la creación de la VM

| Prioridad | Nombre | Puerto | Protocolo | Origen | Destino | Acción | |
|-----------|---------------------------|------------|------------|-------------------|----------------|------------|-----|
| 300 | SSH | 22 | TCP | Cualquiera | Cualquiera | ✔ Permitir | ... |
| 310 | Port_50000 | 50000 | Cualquiera | Cualquiera | Cualquiera | ✔ Permitir | ... |
| 65000 | AllowVnetInBound | Cualquiera | Cualquiera | VirtualNetwork | VirtualNetwork | ✔ Permitir | ... |
| 65001 | AllowAzureLoadBalancer... | Cualquiera | Cualquiera | AzureLoadBalancer | Cualquiera | ✔ Permitir | ... |
| 65500 | DenyAllInBound | Cualquiera | Cualquiera | Cualquiera | Cualquiera | ✖ Denegar | ... |

Figura 5: Configuración de puertos para funcionamiento de la aplicación

3.2. Apartado de Winbugs

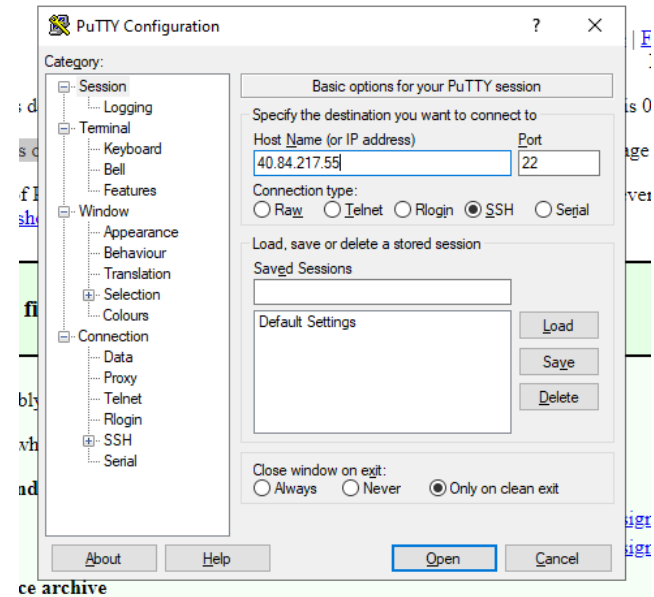


Figura 6: Conexión desde putty

```

PS C:\Users\Laura Pardo\Documents\distribuidos> psftp adrian@40.84.217.55
Using username "adrian".
adrian@40.84.217.55's password:
Remote working directory is /home/adrian
/psftp> put script1.sh
local:script1.sh => remote:/home/adrian/script1.sh
/psftp> put Servidor2.java
local:Servidor2.java => remote:/home/adrian/Servidor2.java
/psftp> put Simple*
local: unable to open Simple*
/psftp> put SimpleProxyServer.java
local:SimpleProxyServer.java => remote:/home/adrian/SimpleProxyServer.java
/psftp>

```

Figura 7: Conexion de psftp desde Powershell para enviar archivos java y sh
Para enviar archivos desde psftp solo es necesario hacer un put <archivo>

```

adrian@adrianvml: ~
adrian@adrianvml:~$ ls
adrian@adrianvml:~$ ls
Servidor2.java  SimpleProxyServer.java  script1.sh
adrian@adrianvml:~$

```

Figura 8: ls antes y despues de ejecutar el psftp en la maquina virtual

```

adrian@adrianvml: ~
* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:        https://ubuntu.com/advantage

System information as of Sat Jan  9 02:35:15 UTC 2021

System load:  0.01          Processes:      111
Usage of /:   6.3% of 28.90GB Users logged in:  1
Memory usage: 23%          IP address for eth0: 10.0.0.5
Swap usage:   0%

20 packages can be updated.
17 of these updates are security updates.
To see these additional updates run: apt list --upgradable

New release '20.04.1 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Sat Jan  9 02:33:37 2021 from 187.189.91.62
adrian@adrianvml:~$ ls
Servidor2.java  SimpleProxyServer.java  script1.sh
adrian@adrianvml:~$

```

Figura 9: Conexion a maquina 1 desde Putty

```
adrian@adrianvm2: ~  
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 5.4.0-1035-azure x86_64)  
  
* Documentation:  https://help.ubuntu.com  
* Management:    https://landscape.canonical.com  
* Support:       https://ubuntu.com/advantage  
  
System information as of Sat Jan  9 02:37:16 UTC 2021  
  
System load:  0.0          Processes:      110  
Usage of /:   6.3% of 28.90GB Users logged in:   0  
Memory usage: 23%         IP address for eth0: 10.0.0.6  
Swap usage:   0%  
  
20 packages can be updated.  
17 of these updates are security updates.  
To see these additional updates run: apt list --upgradable  
  
New release '20.04.1 LTS' available.  
Run 'do-release-upgrade' to upgrade to it.  
  
Last login: Sat Jan  9 02:33:27 2021 from 187.189.91.62  
adrian@adrianvm2:~$
```

Figura 10: Conexion a maquina 2 desde Putty

Despues de realizar esto es necesario primero ejecutar el script2.sh en la VM2

```
adrian@adrianvm2: ~  
adrian@adrianvm2:~$ bash script2.sh  
Instalacion de archivos en la VM  
Hit:1 http://azure.archive.ubuntu.com/ubuntu bionic InRelease  
Hit:2 http://azure.archive.ubuntu.com/ubuntu bionic-updates InRelease  
Hit:3 http://azure.archive.ubuntu.com/ubuntu bionic-backports InRelease  
Hit:4 http://security.ubuntu.com/ubuntu bionic-security InRelease  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
11 packages can be upgraded. Run 'apt list --upgradable' to see them.  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
build-essential is already the newest version (12.4ubuntu1).  
openjdk-8-jdk-headless is already the newest version (8u275-b01-0ubuntu1~18.04).  
The following package was automatically installed and is no longer required:  
  linux-headers-4.15.0-129  
Use 'sudo apt autoremove' to remove it.  
0 upgraded, 0 newly installed, 0 to remove and 11 not upgraded.  
Compilacion de archivos java  
Corriendo binarios  
Fin  
adrian@adrianvm2:~$
```

Figura 11: Ejecución del script 2

Despues se sigue con el script1.sh y la IP de la VM2 para su correcto funcionamiento

```
ne
tcError
tc      Usage: script1.sh <IP_Second>
tc
tc      adrian@adrianvml:~$ bash script1.sh 40.124.31.8
Instalacion de archivos en la VM
Hit:1 http://azure.archive.ubuntu.com/ubuntu bionic InRelease
Hit:2 http://azure.archive.ubuntu.com/ubuntu bionic-updates InRelease
Hit:3 http://azure.archive.ubuntu.com/ubuntu bionic-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu bionic-security InRelease
Reading package lists... Done
Building dependency tree
Reading state information... Done
11 packages can be upgraded. Run 'apt list --upgradable' to see them.
Reading package lists... Done
Building dependency tree
Reading state information... Done
build-essential is already the newest version (12.4ubuntu1).
openjdk-8-jdk-headless is already the newest version (8u275-b01-0ubuntu1~18.04).
The following package was automatically installed and is no longer required:
  linux-headers-4.15.0-129
Use 'sudo apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 11 not upgraded.
Modificación del archivo
Compilacion de archivos java
```

Figura 12: Ejecución del script 1

Finalmente ejecutaremos nuestro cliente de tal modo en que pondremos la IP de la VM1

```
PS C:\Users\Laura Pardo\Documents\distribuidos> java Cliente2
Error
      Usage: java Cliente2 <IP>
PS C:\Users\Laura Pardo\Documents\distribuidos> java Cliente2 40.84.217.55
Conectando a: 40.84.217.55
HOLA
Saliendo...
PS C:\Users\Laura Pardo\Documents\distribuidos>
```

Figura 13: Ejecución del cliente

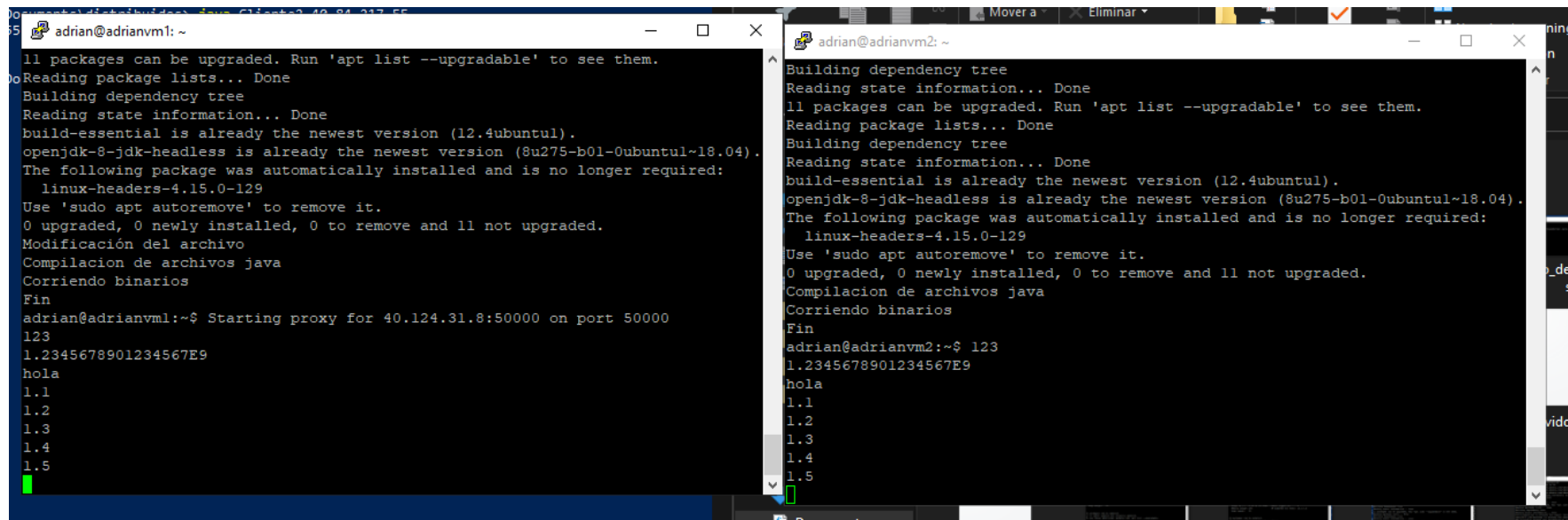
The image shows two terminal windows side-by-side. The left window, titled 'adrian@adrianvm1: ~', displays the output of an 'apt update' command. It reports that 11 packages can be upgraded and lists several already installed versions like 'build-essential (12.4ubuntu1)' and 'openjdk-8-jdk-headless (8u275-b01-0ubuntu1~18.04)'. It also notes that 'linux-headers-4.15.0-129' is no longer required. The window concludes with '0 upgraded, 0 newly installed, 0 to remove and 11 not upgraded.' followed by a list of IP addresses (123, 1.2345678901234567E9) and a 'hola' message. The right window, titled 'adrian@adrianvm2: ~', shows identical output for an 'apt update' command. It also reports 11 packages can be upgraded and lists the same already installed versions. It notes that 'linux-headers-4.15.0-129' is no longer required. The window concludes with '0 upgraded, 0 newly installed, 0 to remove and 11 not upgraded.' followed by the same list of IP addresses and a 'hola' message. Both windows have a dark background and white text. The left window has a green cursor at the bottom, and the right window has a green cursor at the bottom.

Figura 14: Vistazo a los servidores o VMs de tal modo en que vemos como se replica la información.

4. Conclusiones

Finalmente como podemos ver el hacer uso de servidores los cuales puedan replicar la información a través de un proxy o de otro tipo de medio es de suma importancia ya que así se puede mantener un control de logs pensando en servicios web para algunos servicios que vemos generalmente como Google que tiene múltiples servidores para atender las peticiones de los usuarios, o pensando en algún Webservice que funcione con este tipo de metodología, por otro lado el hacer uso de un sistema que no es Linux muchas veces no permite la creación o automatización de varios procesos mediante scripting, por lo que es necesario recurrir en algunas ocasiones a herramientas terceras como Putty, y añadir estas herramientas luego puede ser tan doloroso que en Linux simplemente todo podría funcionar con unas líneas como *sshpas* para evitar la parte del password, *ssh* para la conexión al servidor o para la ejecución de algún comando, y *scp* para mandar archivos a la máquina o viceversa.

Para el caso de ser implementado vía Linux los comandos o el script a ejecutar sería el siguiente:

```
1 #!/usr/bin/env bash
2 # @author Adrian Gonzalez Pardo
3 if [ $# -ne 2 ];then
4     echo -e "Error\n\tUsage: <IP-1> <IP-2>"
5     exit 1
6 fi
7 pass="adrianPardo_99"
8 user="adrian"
9 vms=( $1 $2 )
10 for i in "${vms[@]";do
11     echo "Conectando a la vm con IP: ${i}"
12     v=$(sshpas -p ${pass} ssh -o "StrictHostKeyChecking no" ${user}@${i} "sudo
13         apt update && sudo apt install openjdk-8-jdk-headless build-essential cmake
14         -y")
15     echo "Enviando archivos java de la VM con IP ${i}"
16     sshpas -p ${pass} scp script1.sh script2.sh Servidor2.java SimpleProxyServer.
17         java ${user}@${i}:~/
18 done
19 echo "VMS ${vms[0]} modificando puerto de aplicacion"
20 sshpas -p ${pass} ssh ${user}@${vms[0]} "sed -i -e 's/ServerSocket servidor =
21     new ServerSocket(50000);/ServerSocket servidor = new ServerSocket(50001);/g'
22     Servidor2.java"
23 echo "Compilando todos los archivos java"
24 for i in "${vms[@]";do
25     sshpas -p ${pass} ssh ${user}@${i} "javac Servidor2.java && javac
26         SimpleProxyServer.java"
27 done
28 echo -e "Siguiente paso: Ejecutar en VM2 con IP ${vms[1]} :\n\tjava Servidor2&
29     ny posteriormente en VM1 con IP ${vms[0]} :\n\tjava Servidor2& \ny
30     finalmente en la misma VM1:\n\tjava SimpleProxyServer ${vms[1]} 50000 50000
31     50001&"
```