



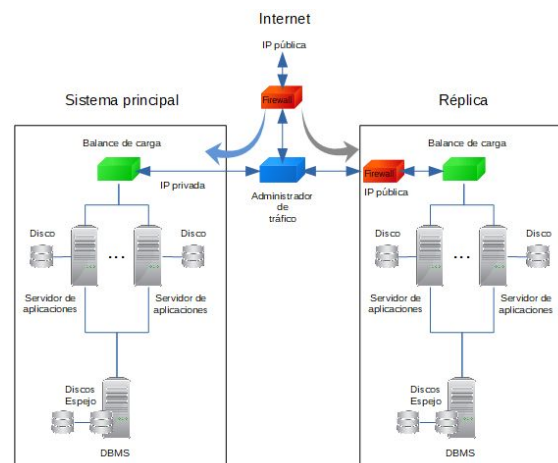
INSTITUTO POLITÉCNICO NACIONAL



ESCUELA SUPERIOR DE CÓMPUTO

Tarea 10:

Replicación de un servidor en la nube



PRESENTA

Pulido Bejarano Raymundo

CIUDAD DE MÉXICO 11 de enero del 2021

Descripción de la Practica

Esta tarea número 10, consiste en implementar un sistema de respaldo para las peticiones y los datos, realizados por el Cliente, siendo más específico, consistía en montar dos máquinas virtuales en la nube de Azure, donde a la Máquina Virtual 1, fungiría como sistema principal y la Máquina Virtual 2 como réplica de lo ocurrido en la MV1, para esto haciendo uso de un proxy que toma las peticiones hechas a la MV1 y se las envía a la MV1 como a la MV2, esto permitiéndonos tener ambos sistemas con sus datos consistentes en caso de ser necesario recurrir a la replica en caso de caída de la MV1 el profeso nos suministró los siguientes programas: “Servidor2.java”, “Cliente2.java” y “SimpleProxyServer.java”, como las instrucciones para su implementación.

Captura de pantalla de cada paso:

Paso 1:

The image displays two screenshots of the Azure portal interface, showing the configuration details for two virtual machines, MV1 and MV2.

MV1 (Máquina virtual):

- Grupo de recur... (cambiar) : SisDis
- Sistema operativo : Linux (ubuntu 18.04)
- Tamaño : B1s estándar (1 vcpu, 1 GiB de memoria)
- Estado : En ejecución
- Dirección IP pública : 40.124.46.173
- Ubicación : Centro-Sur de EE. UU.
- Red virtual/subred : SisDis-vnet/default
- Suscripción (cambiar) : Azure para estudiantes
- Nombre DNS : Configurar
- Id. de suscripción : 746ae8d5-1a0f-4832-bec4-8c42cca66b78
- Etiquetas (cambiar) : Haga clic aquí para agregar etiquetas.

MV2 (Máquina virtual):

- Grupo de recur... (cambiar) : SisDis
- Sistema operativo : Linux (ubuntu 18.04)
- Tamaño : B1s estándar (1 vcpu, 1 GiB de memoria)
- Estado : En ejecución
- Dirección IP pública : 70.37.106.115
- Ubicación : Centro-Sur de EE. UU.
- Red virtual/subred : SisDis-vnet/default
- Suscripción (cambiar) : Azure para estudiantes
- Nombre DNS : Configurar
- Id. de suscripción : 746ae8d5-1a0f-4832-bec4-8c42cca66b78
- Etiquetas (cambiar) : Haga clic aquí para agregar etiquetas.

Paso 2:

MV1 | Redes

Máquina virtual

Adjuntar interfaz de redDesasociar interfaz de red

mv1506

Configuración de IP

ipconfig1 (Principal)

Interfaz de red: mv1506

Reglas de seguridad vigentes

Topología

Red virtual/subred: SisDis-vnet/default

IP pública de NIC: 40.124.46.173

IP privada de NIC: 10.0.0.4

Redes aceleradas: Deshabilitado

Reglas de puerto de entrada

Reglas de puerto de salida

Grupos de seguridad de aplicación

Equilibrio de carga

Grupo de seguridad de red MV1-nsg (se conectó a la interfaz de red: mv1506)

Impactos 0 subredes, 1 interfaces de red

Agregar regla de puerto de entrada

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

Paso 3:

MV2 | Redes

Máquina virtual

Adjuntar interfaz de redDesasociar interfaz de red

mv2551

Configuración de IP

ipconfig1 (Principal)

Interfaz de red: mv2551

Reglas de seguridad vigentes

Topología

Red virtual/subred: SisDis-vnet/default

IP pública de NIC: 70.37.106.115

IP privada de NIC: 10.0.0.5

Redes aceleradas: Deshabilitado

Reglas de puerto de entrada

Reglas de puerto de salida

Grupos de seguridad de aplicación

Equilibrio de carga

Grupo de seguridad de red MV2-nsg (se conectó a la interfaz de red: mv2551)

Impactos 0 subredes, 1 interfaces de red

Agregar regla de puerto de entrada

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

Paso 4:

Ray@MV1: ~


Ray@MV1:~\$

Paso 5:

```
Ray@MV1: ~  
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/bin/serialver to provide /usr/bin/serialver (serialver) in auto mode  
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/bin/jfr to provide /usr/bin/jfr (jfr) in auto mode  
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/bin/wsgen to provide /usr/bin/wsgen (wsgen) in auto mode  
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/bin/jcmd to provide /usr/bin/jcmd (jcmd) in auto mode  
update-alternatives: using /usr/lib/jvm/java-8-openjdk-amd64/bin/jmap to provide /usr/bin/jmap (jmap) in auto mode  
Processing triggers for libc-bin (2.27-3ubuntu1.3) ...  
Processing triggers for systemd (237-3ubuntu10.43) ...  
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...  
Processing triggers for ca-certificates (20201027ubuntu0.18.04.1) ...  
Updating certificates in /etc/ssl/certs...  
0 added, 0 removed; done.  
Running hooks in /etc/ca-certificates/update.d...  
  
done.  
done.  
Processing triggers for ureadahead (0.100.0-21) ...  
Ray@MV1:~$
```

Paso 6:

```
psftp> put Servidor2.java
local:Servidor2.java => remote:/home/Ray/Servidor2.java
psftp> put SimpleProxyServer.java
local:SimpleProxyServer.java => remote:/home/Ray/SimpleProxyServer.java
```

 Ray@MV1: ~


```
Ray@MV1:~$ ls
Servidor2.java  SimpleProxyServer.java
Ray@MV1:~$
```

Paso 7:

```
public static void main(String[] args) throws Exception
{
    ServerSocket servidor = new ServerSocket(50001);


    for (;;)
    {
        Socket conexion = servidor.accept();
        Worker w = new Worker(conexion);
        w.start();
    }
}
Ray@MV1:~$
```

Paso 8:

 Ray@MV1: ~

```
Ray@MV1:~$ javac Servidor2.java
Ray@MV1:~$ javac SimpleProxyServer.java
```

Paso 9:

 Ray@MV2: ~

```
Ray@MV2:~$
```

Paso 10:

```
Ray@MV2: ~  
Ray@MV2:~$ sudo apt update;sudo apt install openjdk-8-jdk-headless
```

Paso 11:

```
Ray@MV2: ~  
Ray@MV2:~$ ls  
Servidor2.java  
Ray@MV2:~$
```

```
psftp> put Servidor2.java  
local:Servidor2.java => remote:/home/Ray/Servidor2.java
```

Paso 12:

```
public static void main(String[] args) throws Exception  
{  
    ServerSocket servidor = new ServerSocket(50000);  
  
    for (;;)   
    {  
        Socket conexion = servidor.accept();  
        Worker w = new Worker(conexion);  
        w.start();  
    }  
}  
}Ray@MV2:~$
```

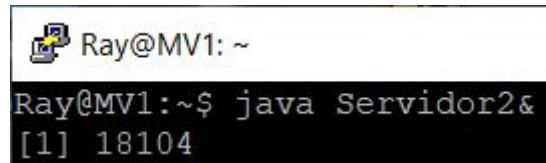
Paso 13:

```
Ray@MV2: ~  
Ray@MV2:~$ javac Servidor2.java  
Ray@MV2:~$
```

Paso 14:

```
Ray@MV2:~$ java Servidor2&  
[1] 16700
```

Paso 15:



```
Ray@MV1: ~  
Ray@MV1:~$ java Servidor2&  
[1] 18104
```

Paso 16:

```
Ray@MV1:~$ java SimpleProxyServer 70.37.106.115 50000 50000 50001&  
[2] 18341  
Ray@MV1:~$ Starting proxy for 70.37.106.115:50000 on port 50000
```


Paso 17:

```
Cliente2.java X
C: > Users > ray_e > Desktop > Cliente2.java

21 public static void main(String[] args) throws Exception
22 {
23     Socket conexion = null;
24
25     for(;;)
26     try
27     {
28         conexion = new Socket("40.124.46.173",50000);
29         break;
30     }
31     catch (Exception e)
32     {}
33     Thread.sleep(100);
34 }
35
```

```
Símbolo del sistema

C:\Users\ray_e\Desktop>javac Cliente2.java

C:\Users\ray_e\Desktop>java Cliente2
HOLA

C:\Users\ray_e\Desktop>_
```

```
[2] 18341 Ray@MV2:~$ 123
Ray@MV1:~$ Starting pro 1.2345678901234567E9
123 hola
1.2345678901234567E9 1.1
hola 1.2
1.1 1.3
1.2 1.4
1.3 1.5
1.4 
1.5 

```


Conclusión

Para cerrar este reporte, no me queda más que decir que la implementación de replicación es complicada de primer instancia cuando la lees, ya que pesar en como realizarla digamos no es algo de primer acercamiento, pero ver los resultados y que tan sencillo puede ser la implementación una vez que se explica, es impresionante, la replicación es sumamente útil y necesaria para cualquier sistema de un tamaño considerable o de una prioridad mediana en adelante, y todos los ingenieros deberían de conocer tácticas como estas sencillas pero muy útiles y necesarias.