

UNIVERSIDAD AUTÓNOMA



PRÁCTICAS DE SISTEMAS INFORMÁTICOS II

PRÁCTICA 3

Memoria

Autores:

Adrián FERNÁNDEZ

Santiago GONZÁLEZ-CARVAJAL

Pareja 1
Grupo 2401

29 de abril de 2019

Índice

1. Ejercicio 1	2
2. Ejercicio 2	6
3. Ejercicio 3	9
4. Ejercicio 4	12
5. Ejercicio 5	16
6. Ejercicio 6	19
7. Ejercicio 7	23
8. Ejercicio 8	26
9. Ejercicio 9	30

1. Ejercicio 1

Ejercicio 1: Preparar 3 máquinas virtuales desde cero (a partir de la VM en moodle) con acceso SSH entre ellas. Esta tarea es necesaria para la correcta gestión del *cluster* que definiremos en el próximo apartado. Las VMs las denominaremos:

- si2srv01: Dirección IP 10.X.Y.1, 768MB RAM
- si2srv02: Dirección IP 10.X.Y.2, 512MB RAM
- si2srv03: Dirección IP 10.X.Y.3, 512MB RAM

RECUERDE RANDOMIZAR LAS DIRECCIONES MAC DE CADA COPIA ANTES DE INTENTAR USAR EL NODO.

En la primera máquina (10.X.Y.1), generaremos el par de claves con DSA. A continuación importaremos la clave pública en cada uno de los otros dos nodos (10.X.Y.2 y 10.X.Y.3). Probaremos a acceder por SSH desde .1 a .2 y .3, comprobando que no requiere la introducción de la clave. Obtener una evidencia del inicio remoto de sesión mediante la salida detallada (`ssh -v si2@10.X.Y.2` y `ssh -v si2@10.X.Y.3`). Anote dicha salida en la memoria de prácticas.

Revisar y comentar la salida del mandato ssh.

Salida del acceso mediante SSH a .2:

```
si2@si2srv01:~$ ssh -v si2@10.1.1.2
OpenSSH_5.3p1 Debian-3ubuntu7, OpenSSL 0.9.8k 25 Mar 2009
debug1: Reading configuration data /etc/ssh/ssh_config
debug1: Applying options for *
debug1: Connecting to 10.1.1.2 [10.1.1.2] port 22.
debug1: Connection established.
debug1: identity file /home/si2/.ssh/identity type -1
debug1: identity file /home/si2/.ssh/id_rsa type -1
debug1: identity file /home/si2/.ssh/id_dsa type 2
debug1: Checking blacklist file /usr/share/ssh/blacklist.DSA-1024
debug1: Checking blacklist file /etc/ssh/blacklist.DSA-1024
debug1: Remote protocol version 2.0, remote software version OpenSSH_5
.3p1 Debian-3ubuntu7
debug1: match: OpenSSH_5.3p1 Debian-3ubuntu7 pat OpenSSH*
debug1: Enabling compatibility mode for protocol 2.0
debug1: Local version string SSH-2.0-OpenSSH_5.3p1 Debian-3ubuntu7
debug1: SSH2_MSG_KEXINIT sent
debug1: SSH2_MSG_KEXINIT received
debug1: kex: server->client aes128-ctr hmac-md5 none
debug1: kex: client->server aes128-ctr hmac-md5 none
debug1: SSH2_MSG_KEX_DH_GEX_REQUEST(1024<1024<8192) sent
debug1: expecting SSH2_MSG_KEX_DH_GEX_GROUP
debug1: SSH2_MSG_KEX_DH_GEX_INIT sent
debug1: expecting SSH2_MSG_KEX_DH_GEX_REPLY
debug1: Host '10.1.1.2' is known and matches the RSA host key.
debug1: Found key in /home/si2/.ssh/known_hosts:1
debug1: ssh_rsa_verify: signature correct
```

```

debug1: SSH2.MSG.NEWKEYS sent
debug1: expecting SSH2.MSG.NEWKEYS
debug1: SSH2.MSG.NEWKEYS received
debug1: SSH2.MSG.SERVICE.REQUEST sent
debug1: SSH2.MSG.SERVICE.ACCEPT received
debug1: Authentications that can continue: publickey,password
debug1: Next authentication method: publickey
debug1: Trying private key: /home/si2/.ssh/identity
debug1: Trying private key: /home/si2/.ssh/id_rsa
debug1: Offering public key: /home/si2/.ssh/id_dsa
debug1: Server accepts key: pkalg ssh-dss blen 433
debug1: read PEM private key done: type DSA
debug1: Authentication succeeded (publickey).
debug1: channel 0: new [client-session]
debug1: Requesting no-more-sessions@openssh.com
debug1: Entering interactive session.
debug1: Sending environment.
debug1: Sending env LC_PAPER = es_ES.UTF-8
debug1: Sending env LC_ADDRESS = es_ES.UTF-8
debug1: Sending env LC_MONETARY = es_ES.UTF-8
debug1: Sending env LC_NUMERIC = es_ES.UTF-8
debug1: Sending env LC_TELEPHONE = es_ES.UTF-8
debug1: Sending env LC_IDENTIFICATION = es_ES.UTF-8
debug1: Sending env LANG = C
debug1: Sending env LC_MEASUREMENT = es_ES.UTF-8
debug1: Sending env LC_TIME = es_ES.UTF-8
debug1: Sending env LC_NAME = es_ES.UTF-8
Linux si2srv02 2.6.32-33-generic #72-Ubuntu SMP Fri Jul 29 21:08:37 UTC
      2011 i686 GNU/Linux
Ubuntu 10.04.3 LTS

Welcome to Ubuntu!
 * Documentation:  https://help.ubuntu.com/
New release 'precise' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Tue Apr 16 06:51:08 2019 from 10.1.1.1
Loading es
perl: warning: Setting locale failed.
perl: warning: Please check that your locale settings:
    LANGUAGE = (unset),
    LC_ALL = (unset),
    LC_TIME = "es_ES.UTF-8",
    LC_MONETARY = "es_ES.UTF-8",
    LC_ADDRESS = "es_ES.UTF-8",
    LC_TELEPHONE = "es_ES.UTF-8",
    LC_NAME = "es_ES.UTF-8",
    LC_MEASUREMENT = "es_ES.UTF-8",
    LC_IDENTIFICATION = "es_ES.UTF-8",
    LC_NUMERIC = "es_ES.UTF-8",
    LC_PAPER = "es_ES.UTF-8",
    LANG = "en_US.UTF-8"
are supported and installed on your system.
perl: warning: Falling back to the standard locale ("C").

```

Salida del acceso mediante SSH a .3:

```
si2@si2srv01:~$ ssh -v si2@10.1.1.3
OpenSSH_5.3p1 Debian-3ubuntu7, OpenSSL 0.9.8k 25 Mar 2009
debug1: Reading configuration data /etc/ssh/ssh_config
debug1: Applying options for *
debug1: Connecting to 10.1.1.3 [10.1.1.3] port 22.
debug1: Connection established.
debug1: identity file /home/si2/.ssh/identity type -1
debug1: identity file /home/si2/.ssh/id_rsa type -1
debug1: identity file /home/si2/.ssh/id_dsa type 2
debug1: Checking blacklist file /usr/share/ssh/blacklist.DSA-1024
debug1: Checking blacklist file /etc/ssh/blacklist.DSA-1024
debug1: Remote protocol version 2.0, remote software version OpenSSH_5
.3p1 Debian-3ubuntu7
debug1: match: OpenSSH_5.3p1 Debian-3ubuntu7 pat OpenSSH*
debug1: Enabling compatibility mode for protocol 2.0
debug1: Local version string SSH-2.0-OpenSSH_5.3p1 Debian-3ubuntu7
debug1: SSH2_MSG_KEXINIT sent
debug1: SSH2_MSG_KEXINIT received
debug1: kex: server->client aes128-ctr hmac-md5 none
debug1: kex: client->server aes128-ctr hmac-md5 none
debug1: SSH2_MSG_KEX_DH_GEX_REQUEST(1024<1024<8192) sent
debug1: expecting SSH2_MSG_KEX_DH_GEX_GROUP
debug1: SSH2_MSG_KEX_DH_GEX_INIT sent
debug1: expecting SSH2_MSG_KEX_DH_GEX_REPLY
debug1: Host '10.1.1.3' is known and matches the RSA host key.
debug1: Found key in /home/si2/.ssh/known_hosts:2
debug1: ssh_rsa_verify: signature correct
debug1: SSH2_MSG_NEWKEYS sent
debug1: expecting SSH2_MSG_NEWKEYS
debug1: SSH2_MSG_NEWKEYS received
debug1: SSH2_MSG_SERVICE_REQUEST sent
debug1: SSH2_MSG_SERVICE_ACCEPT received
debug1: Authentications that can continue: publickey,password
debug1: Next authentication method: publickey
debug1: Trying private key: /home/si2/.ssh/identity
debug1: Trying private key: /home/si2/.ssh/id_rsa
debug1: Offering public key: /home/si2/.ssh/id_dsa
debug1: Server accepts key: pkalg ssh-dss blen 433
debug1: read PEM private key done: type DSA
debug1: Authentication succeeded (publickey).
debug1: channel 0: new [client-session]
debug1: Requesting no-more-sessions@openssh.com
debug1: Entering interactive session.
debug1: Sending environment.
debug1: Sending env LC_PAPER = es_ES.UTF-8
debug1: Sending env LC_ADDRESS = es_ES.UTF-8
debug1: Sending env LC_MONETARY = es_ES.UTF-8
debug1: Sending env LC_NUMERIC = es_ES.UTF-8
debug1: Sending env LC_TELEPHONE = es_ES.UTF-8
debug1: Sending env LC_IDENTIFICATION = es_ES.UTF-8
debug1: Sending env LANG = C
debug1: Sending env LC_MEASUREMENT = es_ES.UTF-8
debug1: Sending env LC_TIME = es_ES.UTF-8
debug1: Sending env LC_NAME = es_ES.UTF-8
Linux si2srv03 2.6.32-33-generic #72-Ubuntu SMP Fri Jul 29 21:08:37 UTC
2011 i686 GNU/Linux
```

```
Ubuntu 10.04.3 LTS
```

```
Welcome to Ubuntu!
```

```
* Documentation:  https://help.ubuntu.com/
```

```
New release 'precise' available.
```

```
Run 'do-release-upgrade' to upgrade to it.
```

```
Last login: Tue Apr 16 06:48:23 2019 from 10.10.1.40
```

```
Loading es
```

```
perl: warning: Setting locale failed.
```

```
perl: warning: Please check that your locale settings:
```

```
    LANGUAGE = (unset),
```

```
    LC_ALL = (unset),
```

```
    LC_TIME = "es_ES.UTF-8",
```

```
    LC_MONETARY = "es_ES.UTF-8",
```

```
    LC_ADDRESS = "es_ES.UTF-8",
```

```
    LC_TELEPHONE = "es_ES.UTF-8",
```

```
    LC_NAME = "es_ES.UTF-8",
```

```
    LC_MEASUREMENT = "es_ES.UTF-8",
```

```
    LC_IDENTIFICATION = "es_ES.UTF-8",
```

```
    LC_NUMERIC = "es_ES.UTF-8",
```

```
    LC_PAPER = "es_ES.UTF-8",
```

```
    LANG = "en_US.UTF-8"
```

```
are supported and installed on your system.
```

```
perl: warning: Falling back to the standard locale ("C").
```

Explicación de resultados:

Como podemos observar, ambas salidas nos muestran el intercambio y comprobación de claves (que hemos creado anteriormente) con éxito. Además, una vez producida la autenticación mediante las claves, podemos comprobar que se realiza el envío de variables de configuración como LC_PAPER y LANG.

2. Ejercicio 2

Ejercicio 2: Realizar los pasos del apartado 4 con el fin de obtener una configuración válida del *cluster* SI2Cluster, con la topología indicada de 1 DAS y 2 nodos SSH de instancias. Inicie el *cluster*. Liste las instancias del *cluster* y verifique que los *pids* de los procesos Java (JVM) correspondientes están efectivamente corriendo en cada una de las dos máquinas virtuales. Adjunte evidencias a la memoria de la práctica.

Evidencias:

Una vez creados los nodos 1 y 2, realizamos las siguientes acciones:

```
si2@si2srv01:~$ asadmin --user admin --passwordfile /opt/SI2/passwordfile list-nodes
localhost-domain1 CONFIG localhost
Node01 SSH 10.1.1.2
Node02 SSH 10.1.1.3
Command list-nodes executed successfully.
```

Figura 1: Listado de nodos.

```
si2@si2srv01:~$ asadmin --user admin --passwordfile /opt/SI2/passwordfile ping-node-ssh Node01
Successfully made SSH connection to node Node01 (10.1.1.2)
Command ping-node-ssh executed successfully.
si2@si2srv01:~$ asadmin --user admin --passwordfile /opt/SI2/passwordfile ping-node-ssh Node02
Successfully made SSH connection to node Node02 (10.1.1.3)
Command ping-node-ssh executed successfully.
```

Figura 2: Ping a los nodos 1 y 2.

Nodes (3)					
<div><div></div><div>New</div><div>Delete</div><div>Delete and Uninstall</div></div>					
Select	Name	Node Host	Type	Instances	Action
<input type="checkbox"/>	Node01	10.1.1.2	SSH		Ping
<input type="checkbox"/>	Node02	10.1.1.3	SSH		Ping
<input type="checkbox"/>	localhost-domain1	localhost	CONFIG		

Figura 3: Consulta de nodos en la consola de administración.

Tras crear el cluster **SI2Cluster**:

```
si2@si2srv01:~$ asadmin --user admin --passwordfile /opt/SI2/passwordfile list-instances -l
Name      Host      Port  Pid  Cluster      State
Instance01 10.1.1.2  24848 --   SI2Cluster   not running
Instance02 10.1.1.3  24848 --   SI2Cluster   not running
Command list-instances executed successfully.
```

Figura 4: Listado de instancias antes de iniciar el cluster.

```
si2@si2srv01:~$ asadmin --user admin --passwordfile /opt/SI2/passwordfile start-cluster SI2Cluster
Command start-cluster executed successfully.
```

Figura 5: Resultado de iniciar el cluster.

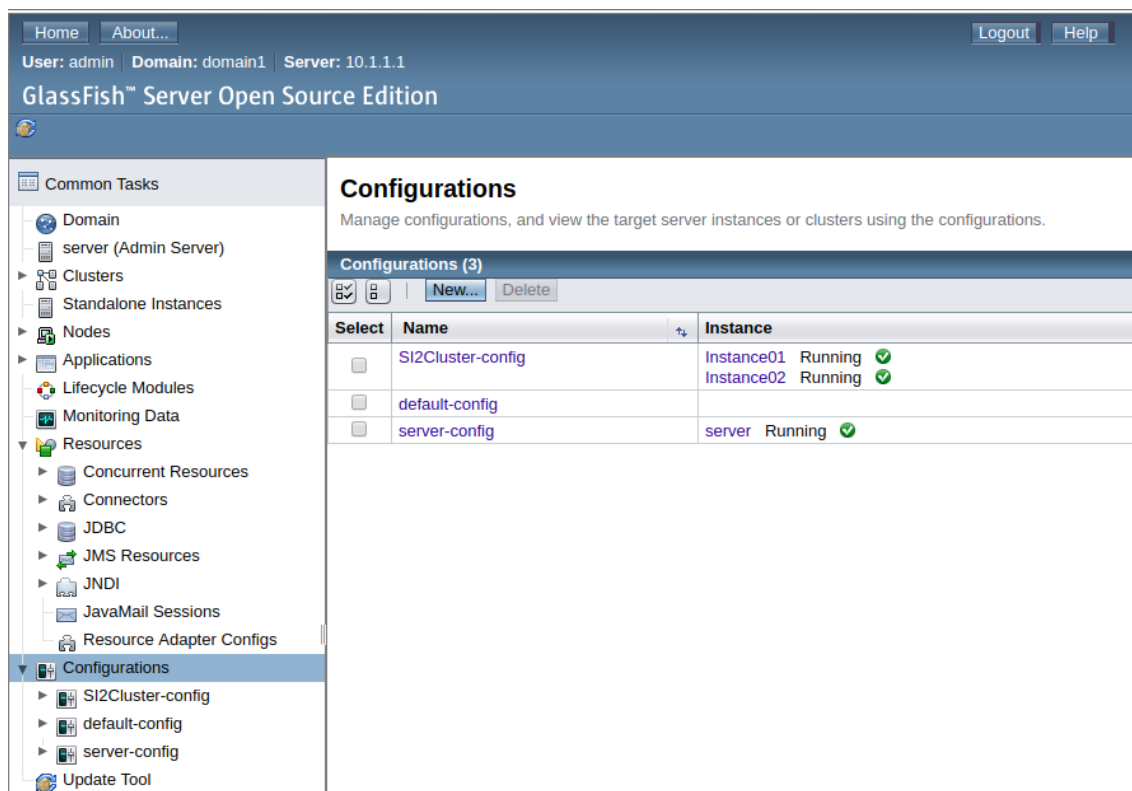


Figura 6: Consulta de configuraciones del sistema.

```
si2@si2srv01:~$ asadmin --user admin --passwordfile /opt/SI2/passwordfile list-instances -l
Name      Host      Port  Pid  Cluster  State
Instance01 10.1.1.2  24848 1818 SI2Cluster running
Instance02 10.1.1.3  24848 1829 SI2Cluster running
Command list-instances executed successfully.
```

Figura 7: Listado de instancias después de iniciar el cluster.

Comprobamos que los pids de los procesos Java (JVM) correspondientes están corriendo en cada una de las dos máquinas virtuales:

```
si2@si2srv02:~$ ps -aefl | grep java
0 S si2      1818      1  9  80    0 - 135767 futex_08:24 ?        00:00:19 /usr/lib/jvm/java-8-oracle/bin/java -cp /opt/glassfish4/glassfish/modules/glassfish.jar -XX:+UnlockDiagnosticVMOptions -XX:NewRatio=2 -XX:MaxPermSize=96m -Xmx128m -Xms128m -server -javaagent:/opt/glassfish4/glassfish/lib/monitor/flashlight-agent.jar -Djavax.net.ssl.trustStore=/opt/glassfish4/Node01/Instance01/config/cacerts.jks -Djdk.corba.allowOutputStreamSubclass=true -Dfelix.fileinstall.dir=/opt/glassfish4/glassfish/modules/autostart/ -Dorg.glassfish.additionalOSGiBundlesToStart=org.apache.felix.shell,org.apache.felix.gogo.runtime,org.apache.felix.gogo.shell,org.apache.felix.gogo.command,org.apache.felix.fileinstall -Dcom.sun.aas.installRoot=/opt/glassfish4/glassfish -Dfelix.fileinstall.poll=5000 -Djava.security.policy=/opt/glassfish4/Node01/Instance01/config/server.policy -Djava.endorsed.dirs=/opt/glassfish4/glassfish/modules/endorsed:/opt/glassfish4/glassfish/lib/endorsed -Dfelix.fileinstall.bundles.startTransient=true -Dosgi.shell.telnet.maxconn=1 -Dfelix.fileinstall.log.level=3 -Dcom.sun.enterprise.config.config_environment_factory_class=com.sun.enterprise.config.serverbeans.AppserverConfigEnvironmentFactory -Djavax.net.ssl.keyStore=/opt/glassfish4/Node01/Instance01/config/keystore.jks -Djava.security.auth.login.config=/opt/glassfish4/Node01/Instance01/config/login.conf -Dfelix.fileinstall.disableConfigSave=false -Dfelix.fileinstall.bundles.new.start=true -Dcom.sun.aas.instanceRoot=/opt/glassfish4/Node01/Instance01 -Dosgi.shell.telnet.port=26666 -Dgosh.args=-no shutdown -c noop=true -Dcom.sun.enterprise.security.httpsOutboundKeyAlias=s1as -Dosgi.shell.telnet.ip=127.0.0.1 -DANTLR_USE_DIRECT_CLASS_LOADING=true -Djava.awt.headless=true -Djava.ext.dirs=/usr/lib/jvm/java-8-oracle/lib/ext:/usr/lib/jvm/java-8-oracle/jre/lib/ext:/opt/glassfish4/Node01/Instance01/lib/ext -Djdbc.drivers=org.apache.derby.jdbc.ClientDriver -Djava.library.path=/opt/glassfish4/glassfish/lib:/usr/java/packages/lib/i386:/lib:/usr/lib com.sun.enterprise.glassfish.bootstrap.ASMain -upgrade false -read-stdin true -asadmin-args --host,,si2srv01,,--port,,4848,,--secure=false,,--terse=false,,--echo=false,,--interactive=false,,start-local-instance,,--verbose=false,,--watchdog=false,,--debug=false,,--nodedir,,/opt/glassfish4,,--node,,Node01,,Instance01 -i nstancename Instance01 -type INSTANCE -verbose false -instancedir /opt/glassfish4/Node01/Instance01 -asadmin-classpath /opt/glassfish4/glassfish/modules/admin-cli.jar -debug false -asadmin-classname com.sun.enterprise.admin.cli.AdminMain
0 S si2      2003  1997  0  80    0 - 465 pipe_w 08:27 pts/0    00:00:00 grep java
```

Figura 8: Procesos Java corriendo en la máquina 2.

```
si2@si2srv03:~$ ps -aefl | grep java
0 S si2      1829      1  7  80    0 - 136018 futex_08:24 ?        00:00:20 /usr/lib/jvm/java-8-oracle/bin/java -cp /opt/glassfish4/glassfish/modules/glassfish.jar -XX:+UnlockDiagnosticVMOptions -XX:NewRatio=2 -XX:MaxPermSize=96m -Xmx128m -Xms128m -server -javaagent:/opt/glassfish4/glassfish/lib/monitor/flashlight-agent.jar -Djavax.net.ssl.trustStore=/opt/glassfish4/Node02/Instance02/config/cacerts.jks -Djdk.corba.allowOutputStreamSubclass=true -Dfelix.fileinstall.dir=/opt/glassfish4/glassfish/modules/autostart/ -Dorg.glassfish.additionalOSGiBundlesToStart=org.apache.felix.shell,org.apache.felix.gogo.runtime,org.apache.felix.gogo.shell,org.apache.felix.gogo.command,org.apache.felix.fileinstall -Dcom.sun.aas.installRoot=/opt/glassfish4/glassfish -Dfelix.fileinstall.poll=5000 -Djava.security.policy=/opt/glassfish4/Node02/Instance02/config/server.policy -Djava.endorsed.dirs=/opt/glassfish4/glassfish/modules/endorsed:/opt/glassfish4/glassfish/lib/endorsed -Dfelix.fileinstall.bundles.startTransient=true -Dosgi.shell.telnet.maxconn=1 -Dfelix.fileinstall.log.level=3 -Dcom.sun.enterprise.config.config_environment_factory_class=com.sun.enterprise.config.serverbeans.AppserverConfigEnvironmentFactory -Djavax.net.ssl.keyStore=/opt/glassfish4/Node02/Instance02/config/keystore.jks -Djava.security.auth.login.config=/opt/glassfish4/Node02/Instance02/config/login.conf -Dfelix.fileinstall.disableConfigSave=false -Dfelix.fileinstall.bundles.new.start=true -Dcom.sun.aas.instanceRoot=/opt/glassfish4/Node02/Instance02 -Dosgi.shell.telnet.port=26666 -Dgosh.args=-no shutdown -c noop=true -Dcom.sun.enterprise.security.httpsOutboundKeyAlias=s1as -Dosgi.shell.telnet.ip=127.0.0.1 -DANTLR_USE_DIRECT_CLASS_LOADING=true -Djava.awt.headless=true -Djava.ext.dirs=/usr/lib/jvm/java-8-oracle/lib/ext:/usr/lib/jvm/java-8-oracle/jre/lib/ext:/opt/glassfish4/Node02/Instance02/lib/ext -Djdbc.drivers=org.apache.derby.jdbc.ClientDriver -Djava.library.path=/opt/glassfish4/glassfish/lib:/usr/java/packages/lib/i386:/lib:/usr/lib com.sun.enterprise.glassfish.bootstrap.ASMain -upgrade false -read-stdin true -asadmin-args --host,,si2srv01,,--port,,4848,,--secure=false,,--terse=false,,--echo=false,,--interactive=false,,start-local-instance,,--verbose=false,,--watchdog=false,,--debug=false,,--nodedir,,/opt/glassfish4,,--node,,Node02,,Instance02 -i nstancename Instance02 -type INSTANCE -verbose false -instancedir /opt/glassfish4/Node02/Instance02 -asadmin-classpath /opt/glassfish4/glassfish/modules/admin-cli.jar -debug false -asadmin-classname com.sun.enterprise.admin.cli.AdminMain
0 R si2      2024  2017  0  80    0 - 465 -      08:28 pts/0    00:00:00 grep java
```

Figura 9: Procesos Java corriendo en la máquina 3.

3. Ejercicio 3

Ejercicio 3: Pruebe a realizar un pago **individualmente** en cada instancia. Para ello, identifique los puertos en los que están siendo ejecutados cada una de las dos instancias (IPs 10.X.Y.2 y 10.X.Y.3 respectivamente). Puede realizar esa comprobación directamente desde la *consola de administración*, opción *Applications*, acción *Launch*, observando los *Web Application Links* generados.

Realice un único pago en cada nodo. Verifique que el pago se ha anotado correctamente el nombre de la instancia y la dirección IP. Anote sus observaciones (puertos de cada instancia) y evidencias (captura de pantalla de la tabla de pagos).

Evidencias:

Consulta previa a la realización de pagos:

Web Application Links

If the server or listener is not running, the link may not work. In this event, check the status of the server instance. After launching the web application, use the browser's Back button to return to this screen.

Application Name: P3

Links:

[Instance01] <http://10.1.1.2:28080/P3>
[Instance01] <https://10.1.1.2:28181/P3>
[Instance02] <http://10.1.1.3:28080/P3>
[Instance02] <https://10.1.1.3:28181/P3>

Figura 10: Listado de links a instancias con puertos disponibles.

Table Name									
pago	idautorizacion	idtransaccion	codrespuesta	importe	idcomercio	instancia	ip	numerotarjeta	fecha
tarjeta									

Figura 11: Estado inicial de la base de datos.

Procedemos a realizar un pago en la instancia 1:

Sistema de Pago con tarjeta × +

← → ↺ 🏠 ⓘ 10.1.1.2:28080/P3/comienzapago

Pago con tarjeta

Numero de visa:

Titular:

Fecha Emisión:

Fecha Caducidad:

CVV2:

Id Transacción: 11
Id Comercion: 12
Importe: 13.0

Prácticas de Sistemas Informáticos II

Figura 12: Comienzo de pago en la instancia 1.

Sistema de Pago con tarjeta × +

← → ↺ 🏠 ⓘ 10.1.1.2:28080/P3/procesapago

Pago con tarjeta

Pago realizado con éxito. A continuación se muestra el comprobante del mismo:

idTransaccion: 11
idComercio: 12
importe: 13.0
codRespuesta: 000
idAutorizacion: 1

[Volver al comercio](#)

Prácticas de Sistemas Informáticos II

Figura 13: Finalización de pago en la instancia 1.

Table Name	idautorizacion	idtransaccion	codrespuesta	importe	idcomercio	instancia	ip	numerotarjeta	fecha
pago	1	11	000	13	12	Instance01	10.1.1.2	1111 2222 3333 4444	16/04/19 10:54

Figura 14: Estado de la base de datos tras el primer pago.

Procedemos a realizar un pago en la instancia 2:

Sistema de Pago con tarjeta X +

10.1.1.3:28080/P3/comienzapago

Pago con tarjeta

Numero de visa: 1111 2222 3333 4444

Titular: Jose Garcia

Fecha Emisión: 11/09

Fecha Caducidad: 11/20

CVV2: 123

Pagar

Id Transacción: 21
Id Comercion: 22
Importe: 23.0

Prácticas de Sistemas Informáticos II

Figura 15: Comienzo de pago en la instancia 2.

Sistema de Pago con tarjeta X +

10.1.1.3:28080/P3/procesapago

Pago con tarjeta

Pago realizado con éxito. A continuación se muestra el comprobante del mismo:

idTransaccion: 21
idComercio: 22
importe: 23.0
codRespuesta: 000
idAutorizacion: 2

[Volver al comercio](#)

Prácticas de Sistemas Informáticos II

Figura 16: Finalización de pago en la instancia 2.

TablesViewsIndexesSequencesCode

Table Name

pago

tarjeta

ColumnsIndexesConstraintsTriggersDataInformation

Figura 17: Estado de la base de datos tras el segundo pago.

4. Ejercicio 4

Ejercicio 4: : Probar la influencia de jvmRoute en la afinidad de sesión.

1. Eliminar todas las cookies del navegador.
2. Sin la propiedad jvmRoute, acceder a la aplicación P3 a través de la URL del balanceador: `http://10.X.Y.1/P3`.
3. Completar el pago con datos de tarjeta correctos.
4. Repetir los pagos hasta que uno falle debido a la falta de afinidad de sesión.
5. Mostrar la cookie “JSESSIONID” correspondiente a la URL del balanceador donde se vea:

Name: JSESSIONID
Content: YYYYYYYYYYYYYYYYYYYY
Domain: 10.X.Y.1
Path: /P3

6. Añadir la propiedad “jvmRoute” al cluster y rearrancar el cluster.
7. Eliminar todas las cookies del navegador.
8. Acceso a la aplicación P3 a través de la URL del balanceador: `http://10.X.Y.1/P3`.
9. Completar el pago con datos de tarjeta correctos. Se pueden repetir los pagos y no fallarán.
10. Mostrar la cookie “JSESSIONID” correspondiente a la URL del balanceador donde se vea:

Name: JSESSIONID
Content: ZZZZZZZZZZZZZZZZZZZZZZZZZ
Domain: 10.X.Y.1
Path: /P3

Mostrar las pantallas y comentar: las diferencias en el contenido de las cookies respecto a jvmRoute, y cómo esta diferencia afecta a la afinidad y por qué.

¿Se podría, en general, usar el valor `$com.sun.aas.hostName` para la propiedad jvmRoute, en lugar de `$com.sun.aas.instanceName`?

Evidencias:

Sin la propiedad jvmRoute, acceso a la aplicación P3 a través de la URL del balanceador:

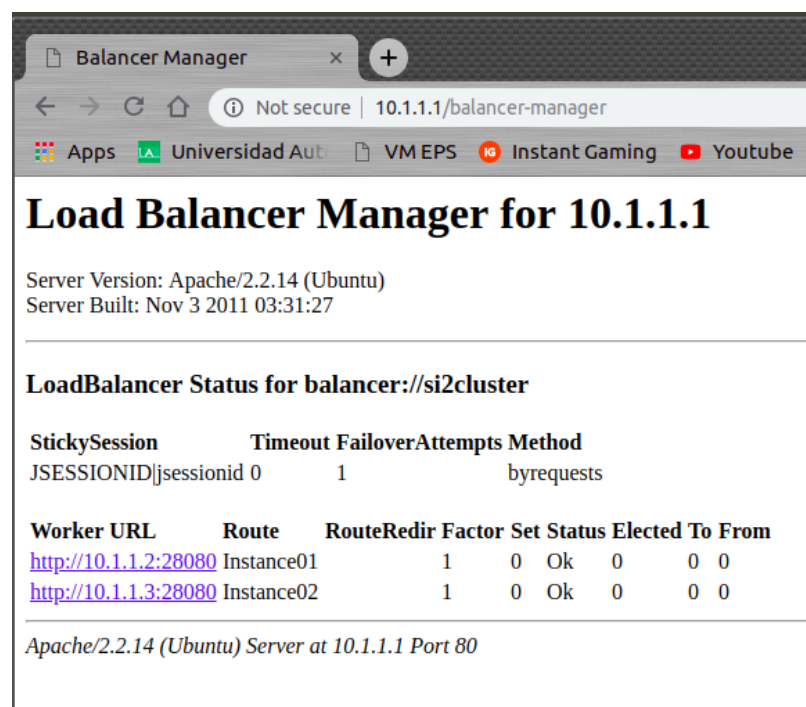


Figura 18: URL de la aplicación reflejada en el balanceador.

JSESSIONID
Name
JSESSIONID
Content
bfacd1214ff518deda20918cef59
Domain
10.1.1.1
Path
/P3

Figura 19: Cookie “JSESSIONID” antes de jvmRoute.

Añadida la propiedad “jvmRoute” al cluster:



Figura 20: Acceso a la aplicación a través de la URL del balanceador tras añadir jvmRoute.

JSESSIONID

Name

JSESSIONID

Content

c01f36a1c5dfabeb443a3d28cffd.Instance01

Domain

10.1.1.1

Path

/P3

Figura 21: Cookie “JSESSIONID” con jvmRoute.

Evidencia de los pagos realizados:

	idautorizacion	idtransaccion	codrespuesta	importe	idcomercio	instancia	ip	numerotarjeta	fecha
1	3	1	000	13	12	Instance01	10.1.1.2	1111 2222 3333 4444	17/04/19 08:54
2	4	2	000	2	1	Instance01	10.1.1.2	1111 2222 3333 4444	17/04/19 08:57

Figura 22: Estado de la base de datos tras realizar dos pagos correctos al activar la cookie.

Cuestiones:

Comentar las diferencias en el contenido de las cookies respecto a jvmRoute, y cómo esta diferencia afecta a la afinidad y por qué.

La diferencia de las cookies es que al añadir jvmRoute aparece la instancia que se ha hecho responsable de comenzar y procesar el pago, mientras que cuando no hemos añadido jvmRoute no aparece. Antes de añadir jvmRoute, el pago falla a la primera, lo cual es debido a la falta de afinidad de sesión porque el balanceador no sabe que instancia se está encargando de las peticiones del usuario en cuestión. Mientras que, al añadir jvmRoute, el balanceador sabe que instancia se encarga de las peticiones del usuario en cuestión y, por lo tanto, manda las peticiones a la instancia indicada con el objetivo de mantener la afinidad de sesión.

¿Se podría, en general, usar el valor `$com.sun.aas.hostName` para la propiedad jvmRoute, en lugar de `$com.sun.aas.instanceName`?

No, en general no se podría porque el `hostName` es global a las dos instancias y por lo tanto no las diferencia (lo cual es el objetivo de utilizar `instanceName`), y por ello no garantiza mantener la afinidad de sesión.

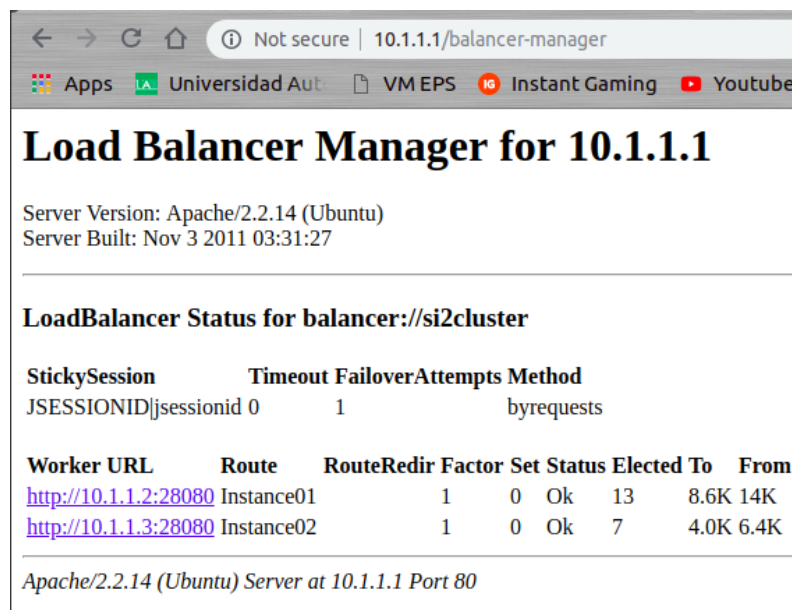
5. Ejercicio 5

Ejercicio 5: Probar el balanceo de carga y la afinidad de sesión, realizando un pago directamente contra la dirección del *cluster* `http://10.X.Y.1/P3` desde distintos ordenadores. Comprobar que las peticiones se reparten entre ambos nodos del *cluster*, y que se mantiene la sesión iniciada por cada usuario sobre el mismo nodo.

Comentad la información mostrada en la página del Load Balancer Manager.

Evidencias:

Consultamos el balanceador de carga antes de realizar los pagos:



StickySession	Timeout	FailoverAttempts	Method
JSESSIONID jsessionid 0	1		byrequests

Worker URL	Route	RouteRedir	Factor	Set	Status	Elected	To	From
http://10.1.1.2:28080	Instance01		1	0	Ok	13	8.6K	14K
http://10.1.1.3:28080	Instance02		1	0	Ok	7	4.0K	6.4K

Apache/2.2.14 (Ubuntu) Server at 10.1.1.1 Port 80

Figura 23: Balanceador de carga.

Realizamos los pagos desde distintos ordenadores:

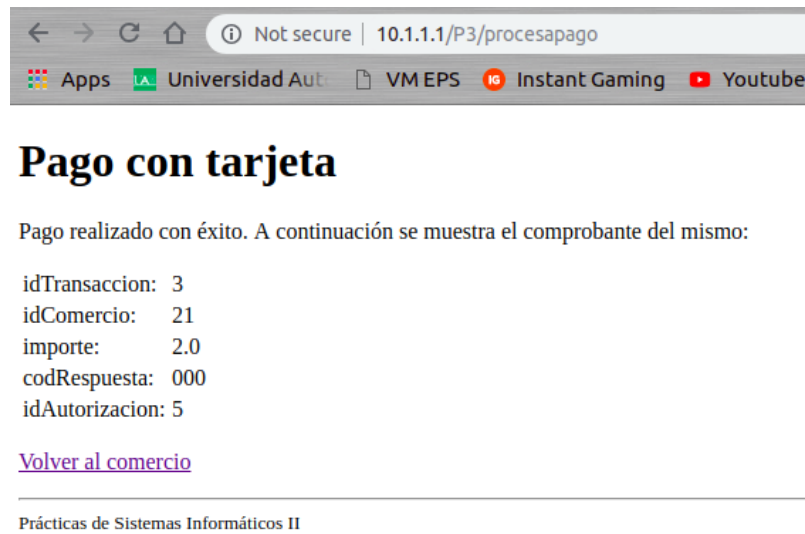


Figura 24: Pago realizado desde el ordenador 1.

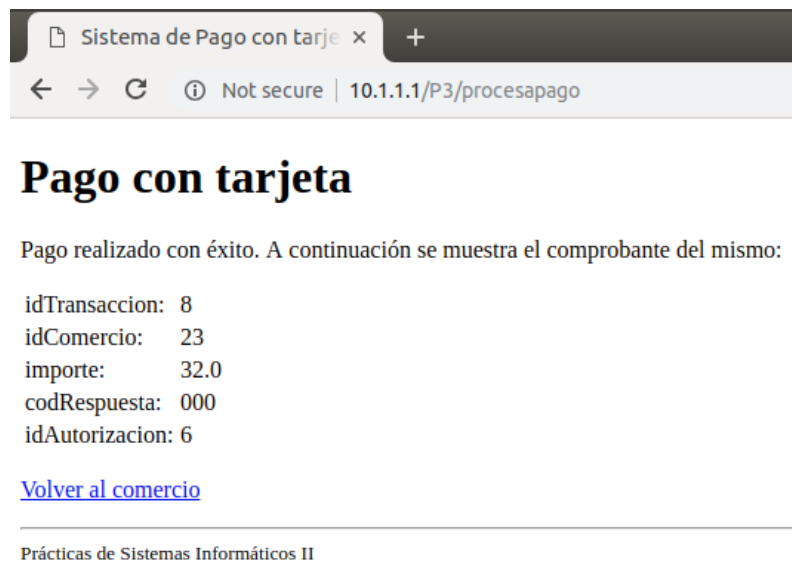


Figura 25: Pago realizado desde el ordenador 2.

Tras realizar los pagos:

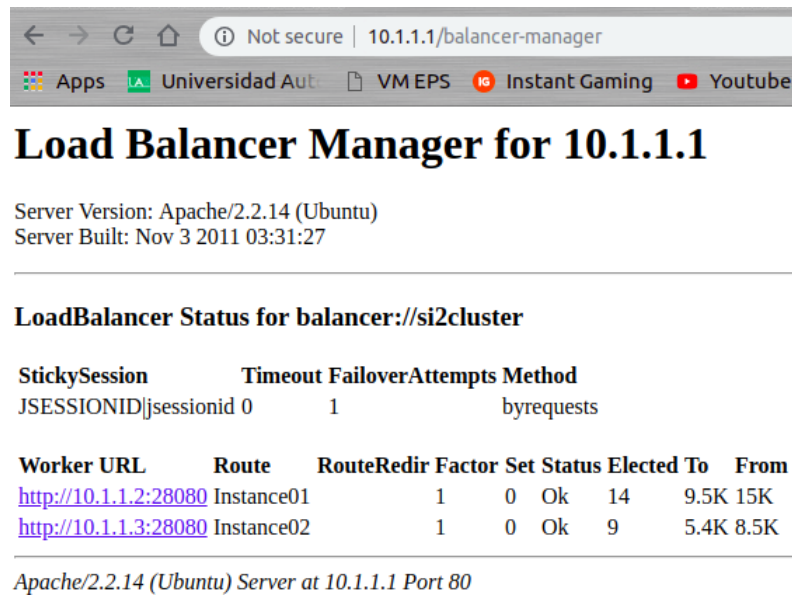


Figura 26: Balanceador de carga.

	idautorizacion	idtransaccion	codrespuesta	importe	idcomercio	instancia	ip	numerotarjeta	fecha
1	5	3	000	2	21	Instance01	10.1.1.2	1111 2222 3333 4444	17/04/19 10:23
2	6	8	000	32	23	Instance02	10.1.1.3	4579 1165 7919 4384	17/04/19 10:23

Figura 27: Base de datos con los pagos divididos en las dos instancias.

Comentarios:

Como podemos apreciar en la página del balanceador, cada uno de los pagos ha sido procesado por una instancia distinta (esto lo podemos ver comparando el valor de la columna Elected previo a realizar los pagos con el valor posterior), además, nos aparece que ambas instancias están operativas (Status: Ok), y otra información que puede ser relevante en distintos contextos.

6. Ejercicio 6

Ejercicio 6: Comprobación del proceso de fail-over. Parar la instancia del cluster que haya tenido menos elecciones hasta el momento. Para ello, identificaremos el **pid** (identificador del proceso java) de la instancia usando las herramientas descritas en esta práctica o el mandato ‘ps -aef — grep java’. Realizaremos un **kill -9 pid** en el nodo correspondiente. Vuelva a realizar peticiones y compruebe (accediendo a la página /balancer-manager y revisando el contenido de la base de datos) que el anterior nodo ha sido marcado como “erróneo” y que todas las peticiones se dirijan al nuevo servidor. Adjunte la secuencia de comandos y evidencias obtenidas en la memoria de la práctica.

Evidencias:

Consultamos la instancia que ha tenido menos elecciones:

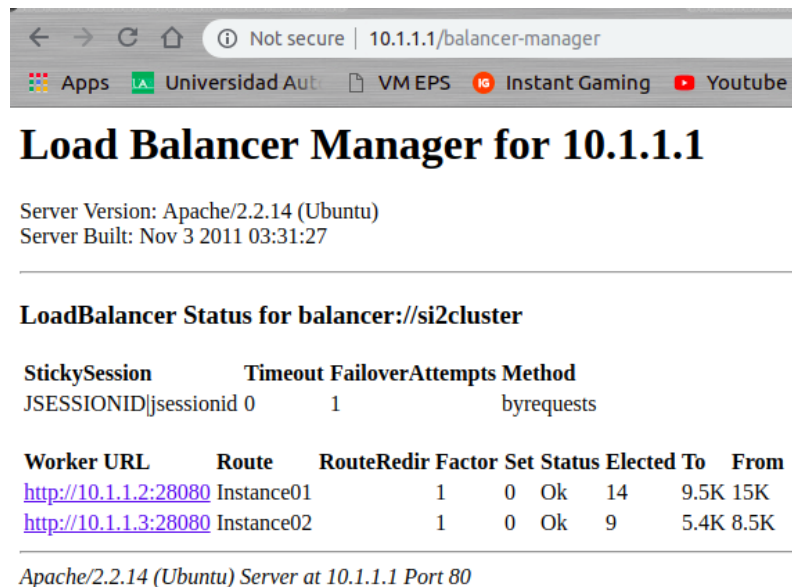


Figura 28: Balanceador de carga del cluster.

Obtenemos el pid del proceso de la instancia 2 y hacemos lo terminamos:

```
si2@si2srv03:~$ ps -aef | grep java
si2      1326      1  3 10:16 ?        00:00:24 /usr/lib/jvm/java-8-oracle/bin/java -cp /opt/glassfish4/glassfish/modules/glassfish.jar -XX:+UnlockDiagnosticVMOptions -XX:NewRatio=2 -XX:MaxPermSize=96m -Xmx128m -Xms128m -server -javaagent:/opt/glassfish4/glassfish/lib/monitor/flashlight-agent.jar -Djavax.net.ssl.trustStore=/opt/glassfish4/Node02/Instance02/config/cacerts.jks -Djdk.corba.allowOutputStreamSubclass=true -Dfelix.fileinstall.dir=/opt/glassfish4/glassfish/modules/autostart/ -Dorg.glassfish.additionalOSGiBundlesToStart=org.apache.felix.shell,org.apache.felix.gogo.runtime,org.apache.felix.gogo.shell,org.apache.felix.gogo.command,org.apache.felix.fileinstall -Dcom.sun.aas.installRoot=/opt/glassfish4/glassfish -Dfelix.fileinstall.poll=5000 -Djava.security.policy=/opt/glassfish4/Node02/Instance02/config/server.policy -Djava.endorsed.dirs=/opt/glassfish4/glassfish/modules/endorsed:/opt/glassfish4/glassfish/lib/endorsed -Dfelix.fileinstall.bundles.startTransient=true -Dosgi.shell.telnet.maxconn=1 -Dfelix.fileinstall.log.level=3 -Dcom.sun.enterprise.config.config_environment_factory_class=com.sun.enterprise.config.serverbeans.AppserverConfigEnvironmentFactory -Djavax.net.ssl.keyStore=/opt/glassfish4/Node02/Instance02/config/keystore.jks -Djava.security.auth.login.config=/opt/glassfish4/Node02/Instance02/config/login.conf -Dfelix.fileinstall.disableConfigSave=false -Dfelix.fileinstall.bundles.new.start=true -Dcom.sun.aas.instanceRoot=/opt/glassfish4/Node02/Instance02 -Dosgi.shell.telnet.port=26666 -Dgosh.args=--noshutdown -c noop=true -Dcom.sun.enterprise.security.httpOutboundKeyAlias=sias -Dosgi.shell.telnet.ip=127.0.0.1 -DANTLR_USE_DIRECT_CLASS_LOADING=true -Djava.awt.headless=true -Djava.ext.dirs=/usr/lib/jvm/java-8-oracle/lib/ext:/usr/lib/jvm/java-8-oracle/jre/lib/ext:/opt/glassfish4/Node02/Instance02/lib/ext -Djdbc.drivers=org.apache.derby.jdbc.ClientDriver -Djava.library.path=/opt/glassfish4/glassfish/lib:/usr/java/packages/lib/i386:/lib:/usr/lib com.sun.enterprise.glassfish.bootstrap.ASMain -upgrade false -read-stdin true -asadmin-args --host,,si2srv01,,--port,,4848,,--secure=false,,--terse=false,,--echo=false,,--interactive=false,,start-local-instance,,--verbose=false,,--watchdog=false,,--debug=false,,--nodedir,,/opt/glassfish4,,--node,,Node02,,Instance02 -instancename Instance02 -type INSTANCE -verbose false -instancedir /opt/glassfish4/Node02/Instance02 -asadmin-classpath /opt/glassfish4/glassfish/modules/admin-cli.jar -debug false -asadmin-classname com.sun.enterprise.admin.cli.AdminMain
si2      1542  1536   0 10:28 pts/0    00:00:00 grep java
```

Figura 29: Resultado de consulta de pid de la instancia 2.

```
si2@si2srv03:~$ kill -9 1326
```

Figura 30: Paramos la instancia 2.

Load Balancer Manager for 10.1.1.1

Server Version: Apache/2.2.14 (Ubuntu)
Server Built: Nov 3 2011 03:31:27

LoadBalancer Status for balancer://si2cluster

StickySession	Timeout	FailoverAttempts	Method
JSESSIONIDjsessionid 0	1		byrequests

Worker URL	Route	RouteRedir	Factor	Set	Status	Elected	To	From
http://10.1.1.2:28080	Instance01		1	0	Ok	16	11K	18K
http://10.1.1.3:28080	Instance02		1	0	Err	10	5.4K	8.5K

Apache/2.2.14 (Ubuntu) Server at 10.1.1.1 Port 80

Figura 31: Comprobación de estado de error (es necesario hacer un pago para que se actualice el status).

Realizamos pagos y comprobamos que se procesan en la instancia 1:

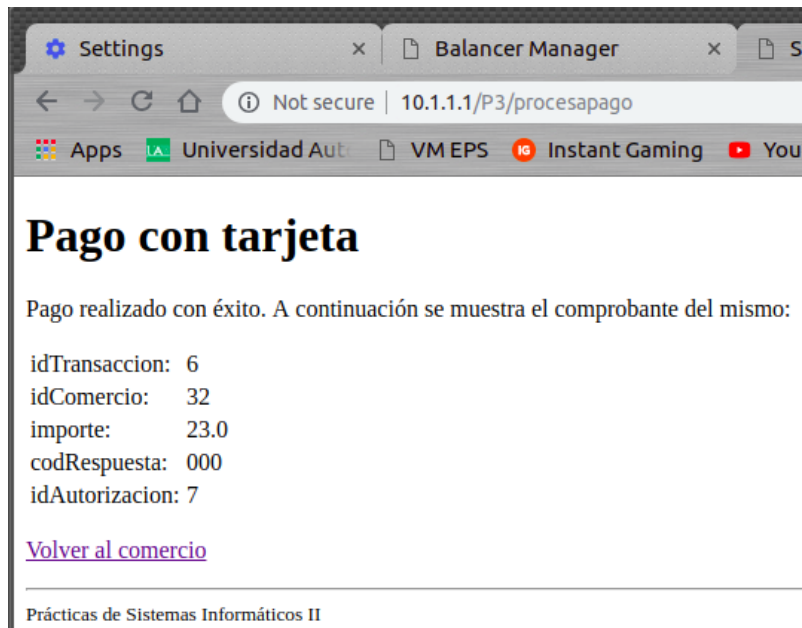


Figura 32: Pago realizado desde un ordenador.

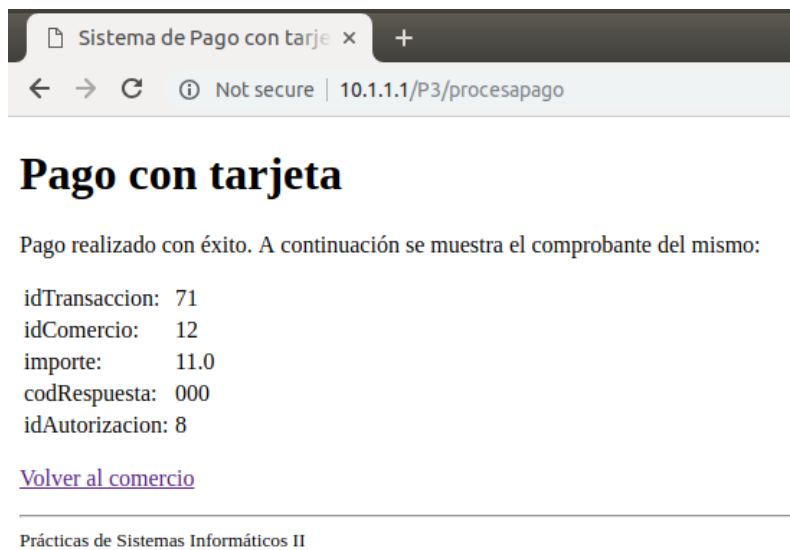


Figura 33: Pago realizado desde el otro ordenador.

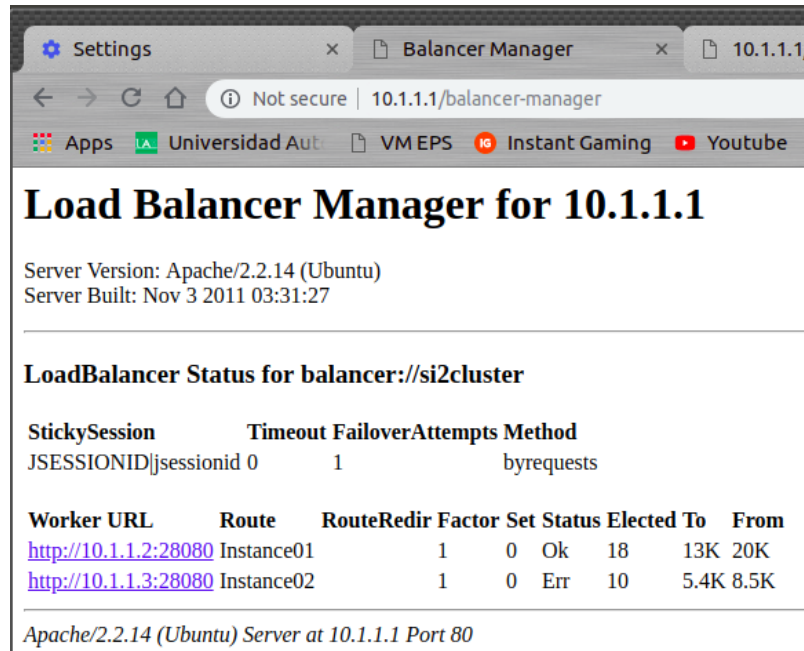


Figura 34: Comprobación de que ambos pagos han sido procesados por la instancia 1.

	idautorizacion	idtransaccion	codrespuesta	importe	idcomercio	instancia	ip	numerotarjeta	fecha
1	7	6	000	23	32	Instance01	10.1.1.2	1111 2222 3333 4444	17/04/19 10:34
2	8	71	000	11	12	Instance01	10.1.1.2	4579 1165 7919 4384	17/04/19 10:34

Figura 35: Base de datos con los pagos procesados por la instancia 1.

7. Ejercicio 7

Ejercicio 7: Comprobación del proceso de fail-back. Inicie manualmente la instancia detenida en el comando anterior. Verificar la activación de la instancia en el gestor del balanceador. Incluir todas las evidencias en la memoria de prácticas y comentar qué sucede con los nuevos pagos. **Consulte los apéndices para información detallada de comandos de gestión individual de las instancias.**

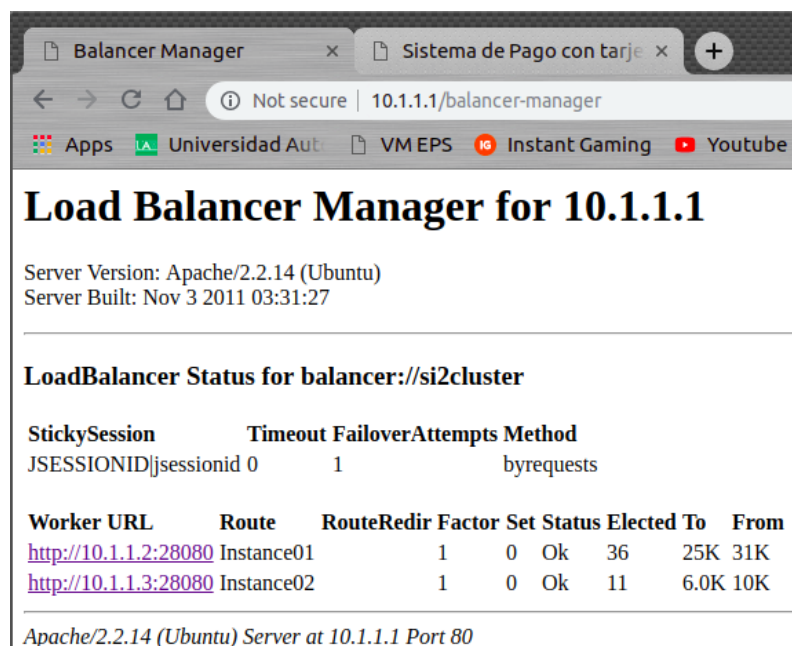
Evidencias:

Volvemos a iniciar la instancia 2:

```
si2@si2srv01:~$ asadmin start-instance Instance02
Enter admin user name> admin
Enter admin password for user "admin">
Waiting for Instance02 to start .....
Successfully started the instance: Instance02
instance Location: /opt/glassfish4/Node02/Instance02
Log File: /opt/glassfish4/Node02/Instance02/logs/server.log
Admin Port: 24848
Command start-local-instance executed successfully.
The instance, Instance02, was started on host 10.1.1.3
Command start-instance executed successfully.
```

Figura 36: Resultado de start-instance.

Estado del balanceador antes de realizar los pagos y tras el pago de verificación (para que se actualice Status a Ok):



Load Balancer Manager for 10.1.1.1

Server Version: Apache/2.2.14 (Ubuntu)
Server Built: Nov 3 2011 03:31:27

LoadBalancer Status for balancer://si2cluster

StickySession	Timeout	FailoverAttempts	Method
JSESSIONID jsessionid 0	1		byrequests

Worker URL	Route	RouteRedir	Factor	Set	Status	Elected	To	From
http://10.1.1.2:28080	Instance01		1	0	Ok	36	25K	31K
http://10.1.1.3:28080	Instance02		1	0	Ok	11	6.0K	10K

Apache/2.2.14 (Ubuntu) Server at 10.1.1.1 Port 80

Figura 37: Balanceador de carga del cluster.

Realizamos pagos y comprobamos que se procesan en ambas instancias:

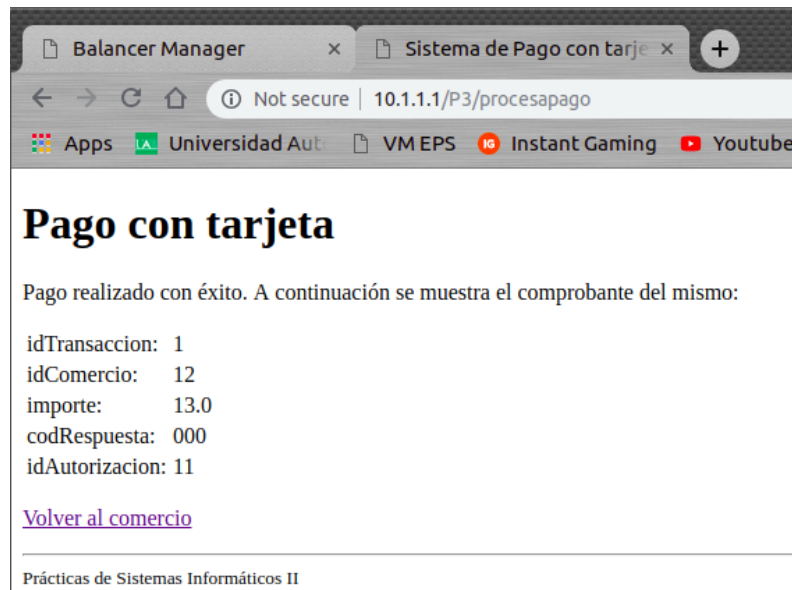


Figura 38: Pago realizado desde un ordenador.

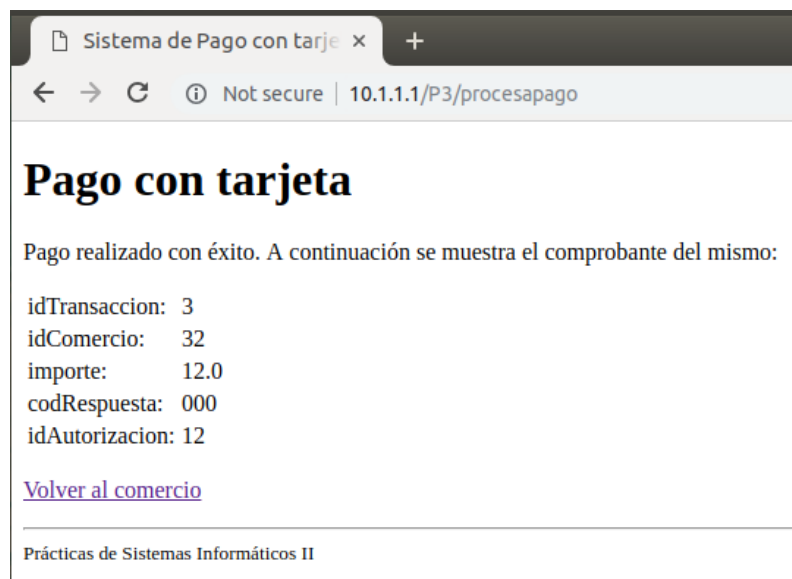


Figura 39: Pago realizado desde el otro ordenador.

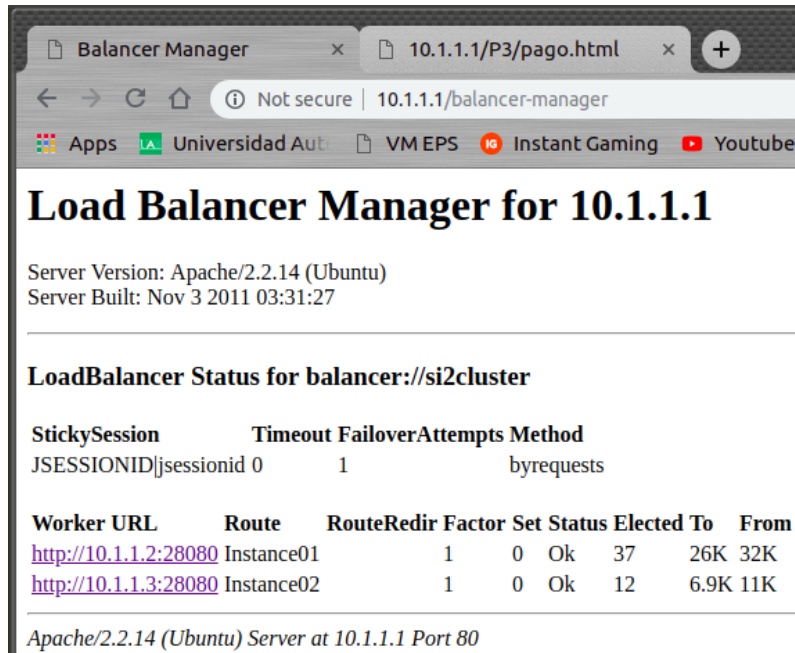


Figura 40: Comprobación de que los pagos han sido procesados en distintas instancias.

	idautorizacion	idtransaccion	codrespuesta	importe	idcomercio	instancia	ip	numerotarjeta	fecha
1	11	1	000	13	12	Instance01	10.1.1.2	1111 2222 3333 4444	17/04/19 11:00
2	12	3	000	12	32	Instance02	10.1.1.3	4579 1165 7919 4384	17/04/19 11:00

Figura 41: Base de datos con pagos realizados en las distintas instancias.

Comentarios:

Como se puede observar en las capturas de pantalla, los pagos vuelven a repartirse entre ambas instancias del cluster (el sistema se ha recuperado tras el error).

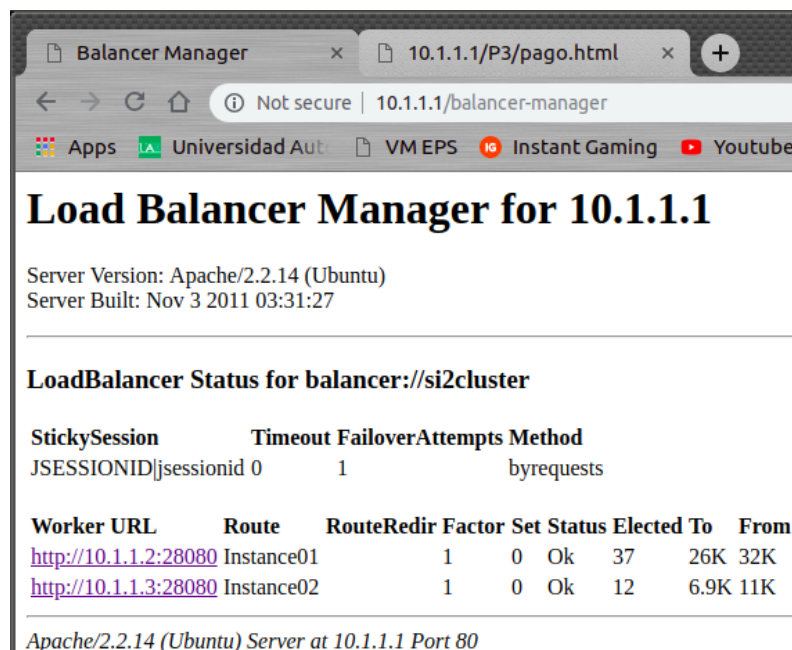
8. Ejercicio 8

Ejercicio 8: Fallo en el transcurso de una sesión.

- Desde un navegador, comenzar una petición de pago introduciendo los valores del mismo en la pantalla inicial y realizando la llamada al *servlet* Comienza-Pago.
- Al presentarse la pantalla de "Pago con tarjeta", leer la instancia del servidor que ha procesado la petición y detenerla. Se puede encontrar la instancia que ha procesado la petición revisando la cookie de sesión (tiene la instancia como sufijo), el balancer-manager o el server.log de cada instancia.
- Completar los datos de la tarjeta de modo que el pago fuera válido, y enviar la petición.
- Observar la instancia del *cluster* que procesa el pago, y razonar las causas por las que se rechaza la petición.

Evidencias:

Estado inicial del balanceador del cluster:



Balancer Manager x 10.1.1.1/P3/pago.html x +

Not secure | 10.1.1.1/balancer-manager

Apps Universidad Aut VM EPS Instant Gaming Youtube

Load Balancer Manager for 10.1.1.1

Server Version: Apache/2.2.14 (Ubuntu)
Server Built: Nov 3 2011 03:31:27

LoadBalancer Status for balancer://si2cluster

StickySession	Timeout	FailoverAttempts	Method
JSESSIONIDjsessionid 0	1		byrequests

Worker URL	Route	RouteRedir	Factor	Set	Status	Elected	To	From
http://10.1.1.2:28080	Instance01		1	0	Ok	37	26K	32K
http://10.1.1.3:28080	Instance02		1	0	Ok	12	6.9K	11K

Apache/2.2.14 (Ubuntu) Server at 10.1.1.1 Port 80

Figura 42: Balanceador de carga del cluster.

Comenzamos el pago:

The screenshot shows a web browser with two tabs: 'Balancer Manager' and 'Sistema de Pago con tarjeta'. The address bar shows '10.1.1.1/P3/comienzapago'. The page title is 'Pago con tarjeta'. The form contains the following fields:

- Numero de visa:
- Titular:
- Fecha Emisión:
- Fecha Caducidad:
- CVV2:
-

Below the form, the transaction details are displayed:

Id Transacción: 1
Id Comercion: 12
Importe: 13.0

At the bottom, the footer reads 'Prácticas de Sistemas Informáticos II'.

Figura 43: Pantalla de pago con tarjeta.

Encontramos la instancia que se encarga del pago y la detenemos:

The screenshot shows a web browser with two tabs: 'Balancer Manager' and 'Sistema de Pago con tarjeta'. The address bar shows '10.1.1.1/balancer-manager'. The page title is 'Load Balancer Manager for 10.1.1.1'. The page content includes:

Server Version: Apache/2.2.14 (Ubuntu)
Server Built: Nov 3 2011 03:31:27

LoadBalancer Status for balancer://si2cluster

StickySession	Timeout	FailoverAttempts	Method
JSESSIONID jsessionid 0	1		byrequests

Worker URL	Route	RouteRedir Factor	Set Status	Elected To	From
http://10.1.1.2:28080	Instance01	1	0 Ok	38	27K 33K
http://10.1.1.3:28080	Instance02	1	0 Ok	12	6.9K 11K

Apache/2.2.14 (Ubuntu) Server at 10.1.1.1 Port 80

Figura 44: Balanceador de carga del cluster.

```
si2@si2srv01:~$ asadmin stop-instance Instance01
Enter admin user name> admin
Enter admin password for user "admin">
The instance, Instance01, is stopped.
Command stop-instance executed successfully.
```

Figura 45: Resultado de stop-instance.

Continuamos el pago tras detener la instancia:

The screenshot shows a web browser window with two tabs: 'Balancer Manager' and 'Sistema de Pago con tarjeta'. The address bar shows '10.1.1.1/P3/comienzapago'. The page title is 'Pago con tarjeta'. The form contains the following fields and values:

Numero de visa:	1111 2222 3333 4444
Titular:	Jose Garcia
Fecha Emisión:	11/09
Fecha Caducidad:	11/20
CVV2:	123

Below the fields is a 'Pagar' button. Underneath the form, the transaction details are displayed:

Id Transacción: 1
Id Comercion: 12
Importe: 13.0

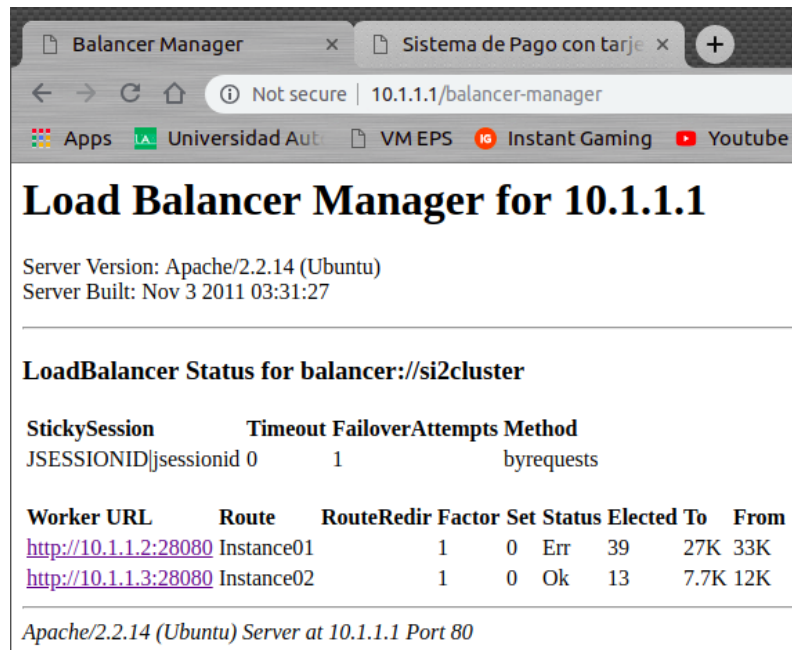
At the bottom, it says 'Prácticas de Sistemas Informáticos II'.

Figura 46: Formulario de pago con tarjeta válido.

The screenshot shows the same web browser window, but the address bar now shows '10.1.1.1/P3/procesapago'. The page title is still 'Pago con tarjeta'. The main content area displays the message 'Pago incorrecto'. At the bottom, it says 'Prácticas de Sistemas Informáticos II'.

Figura 47: Error en el pago.

Estado final del cluster:



Load Balancer Manager for 10.1.1.1

Server Version: Apache/2.2.14 (Ubuntu)
Server Built: Nov 3 2011 03:31:27

LoadBalancer Status for balancer://si2cluster

StickySession	Timeout	FailoverAttempts	Method
JSESSIONIDjsessionid 0	1		byrequests

Worker URL	Route	RouteRedir	Factor	Set	Status	Elected To	From
http://10.1.1.2:28080	Instance01		1	0	Err	39	27K 33K
http://10.1.1.3:28080	Instance02		1	0	Ok	13	7.7K 12K

Apache/2.2.14 (Ubuntu) Server at 10.1.1.1 Port 80

Figura 48: Balanceador de carga del cluster.

Comentarios:

La petición se rechaza porque la instancia que estaba procesando el pago (tenía la información de ComienzaPago y la sesión) es detenida. Esto hace que se pierda toda la información necesaria para procesar el pago con éxito (se pierde toda la sesión).

9. Ejercicio 9

Ejercicio 9: Modificar el script de pruebas JMeter desarrollado durante la P2. (P2.jmx) Habilitar un ciclo de 1000 pruebas en un solo hilo contra la IP del cluster y nueva URL de la aplicación: **http://10.X.Y.1/P3**

Eliminar posibles pagos previos al ciclo de pruebas. Verificar el porcentaje de pagos realizados por cada instancia, así como (posibles) pagos correctos e incorrectos. ¿Qué algoritmo de reparto parece haber seguido el balanceador? Comente todas sus conclusiones en la memoria de prácticas.

Evidencias:

Estado inicial del balanceador de carga del cluster:

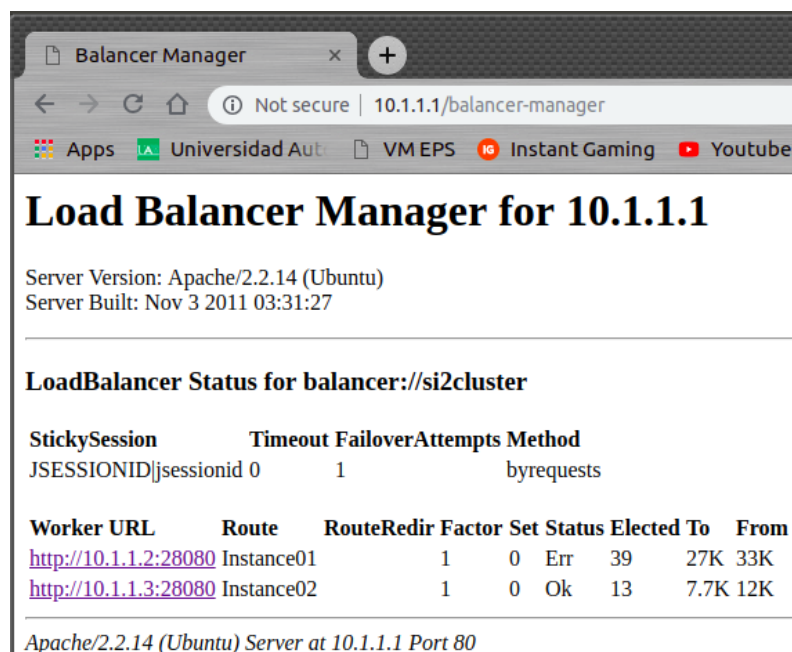


Figura 49: Balanceador de carga del cluster.

Resultado de las pruebas:

Label	# Samples	Average	Median	90% Line	95% Line	99% Line	Min	Maximum	Error %	Throughput	Received kB/sec	Sent KB/sec
P3	1000	25	19	30	35	51	5	3971	0.00%	39.2/sec	53.80	0.00
TOTAL	1000	25	19	30	35	51	5	3971	0.00%	39.2/sec	53.80	0.00

Figura 50: Resultados de las pruebas con JMeter.

#	count
1	1000

Figura 51: Número de pagos en la base de datos (todos los pagos se han llevado a cabo correctamente).

	idautorizacion	idtransaccion	codrespuesta	importe	idcomercio	instancia	ip	numerotarjeta	fecha
1	13	1	000	468	1	Instance02	10.1.1.3	4579 1165 7919 4384	17/04/19 11:26
2	14	6	000	306	1	Instance02	10.1.1.3	6496 3318 3339 9724	17/04/19 11:26
3	15	11	000	413	1	Instance01	10.1.1.2	1725 7286 0489 3263	17/04/19 11:26
4	16	16	000	21	1	Instance02	10.1.1.3	5804 7661 8556 1810	17/04/19 11:26
5	17	21	000	487	1	Instance01	10.1.1.2	3703 9193 7960 4795	17/04/19 11:26
6	18	26	000	519	1	Instance02	10.1.1.3	2039 6424 7433 0841	17/04/19 11:26
7	19	31	000	188	1	Instance01	10.1.1.2	2641 2366 4022 9561	17/04/19 11:26
8	20	36	000	304	1	Instance02	10.1.1.3	8183 1447 8251 7421	17/04/19 11:26
9	21	41	000	877	1	Instance01	10.1.1.2	0077 1888 0446 8810	17/04/19 11:26
10	22	46	000	958	1	Instance02	10.1.1.3	9099 3896 6046 2626	17/04/19 11:26
11	23	51	000	314	1	Instance01	10.1.1.2	5782 6623 4015 7034	17/04/19 11:26
12	24	56	000	524	1	Instance02	10.1.1.3	1348 0911 3414 8006	17/04/19 11:26
13	25	61	000	462	1	Instance01	10.1.1.2	3075 0497 3140 9104	17/04/19 11:26
14	26	66	000	705	1	Instance02	10.1.1.3	6445 5256 3804 2776	17/04/19 11:26
15	27	71	000	920	1	Instance01	10.1.1.2	8661 1288 6707 5926	17/04/19 11:26
16	28	76	000	226	1	Instance02	10.1.1.3	4636 1672 3421 4879	17/04/19 11:26
17	29	81	000	696	1	Instance01	10.1.1.2	7716 5723 4439 2033	17/04/19 11:26
18	30	86	000	322	1	Instance02	10.1.1.3	2040 1146 1977 8420	17/04/19 11:26
19	31	91	000	264	1	Instance01	10.1.1.2	1675 4451 2304 4156	17/04/19 11:26
20	32	96	000	412	1	Instance02	10.1.1.3	6133 2060 7110 6597	17/04/19 11:26
21	33	101	000	259	1	Instance01	10.1.1.2	7326 2207 2052 9442	17/04/19 11:26
22	34	106	000	564	1	Instance02	10.1.1.3	3392 5426 4378 1735	17/04/19 11:26
23	35	111	000	434	1	Instance01	10.1.1.2	3550 1972 2232 3092	17/04/19 11:26
24	36	116	000	477	1	Instance02	10.1.1.3	5933 0091 5306 8815	17/04/19 11:26
25	37	121	000	117	1	Instance01	10.1.1.2	4729 1327 0222 7175	17/04/19 11:26
26	38	126	000	654	1	Instance02	10.1.1.3	2108 1552 5418 6724	17/04/19 11:26
27	39	131	000	106	1	Instance01	10.1.1.2	3488 5224 1404 9522	17/04/19 11:26
28	40	136	000	126	1	Instance02	10.1.1.3	9233 2326 4930 9558	17/04/19 11:26
29	41	141	000	85	1	Instance01	10.1.1.2	9389 8206 7067 4451	17/04/19 11:26
30	42	146	000	854	1	Instance02	10.1.1.3	3100 0276 2348 9183	17/04/19 11:26
31	43	151	000	578	1	Instance01	10.1.1.2	4289 1015 8802 2368	17/04/19 11:26
32	44	156	000	622	1	Instance02	10.1.1.3	8990 8514 3287 3813	17/04/19 11:26
33	45	161	000	525	1	Instance01	10.1.1.2	5895 5455 5700 6797	17/04/19 11:26
34	46	166	000	430	1	Instance02	10.1.1.3	3143 7595 7573 9936	17/04/19 11:26

Figura 52: Algunos de los pagos guardados en la base de datos.

Estado final del balanceador de carga del cluster:

The screenshot shows a web browser window with the title "Balancer Manager". The address bar shows "10.1.1.1/balancer-manager". The page content includes:

Load Balancer Manager for 10.1.1.1

Server Version: Apache/2.2.14 (Ubuntu)
Server Built: Nov 3 2011 03:31:27

LoadBalancer Status for balancer://si2cluster

StickySession	Timeout	FailoverAttempts	Method
JSESSIONID jsessionid 0	1		byrequests

Worker URL	Route	RouteRedir	Factor	Set	Status	Elected	To	From
http://10.1.1.2:28080	Instance01		1	0	Ok	538	292K	547K
http://10.1.1.3:28080	Instance02		1	0	Ok	514	273K	527K

Apache/2.2.14 (Ubuntu) Server at 10.1.1.1 Port 80

Figura 53: Balanceador de carga del cluster.

Cuestiones:

¿Qué algoritmo de reparto parece haber seguido el balanceador?

Puesto que las peticiones se han distribuido casi con la misma probabilidad entre ambas instancias (499 en la primera instancia y 501 en la segunda), la conclusión de que el balanceo de carga tiene una distribución uniforme en cuanto a la carga (que mantiene una diferencia de aproximadamente 20K, columnas To y From) y en cuanto a las peticiones (que como hemos comentado 409 han sido procesadas en la instancia 1 y 501 en la instancia 2). Esto se puede conseguir mediante un algoritmo de distribución *Round Robin*.

Comentarios:

Podemos concluir que el uso de un cluster junto con un balanceador de carga mejora notablemente el rendimiento de sistema a gran escala. Esto se debe a que el balanceador de carga se encarga de repartir la carga de trabajo de manera equitativa entre las distintas instancias del cluster.

Además, como hemos podido comprobar en ejercicios anteriores, el sistema es más robusto ante fallos (por el uso de dos instancias para procesar los pagos, el cliente no se da cuenta de que algo ha fallado ya que podrá seguir pagando, solo que ahora su pago será procesado por otra instancia), y también tiene la capacidad de recuperarse de los mismos.