

 <b>Escuela Politécnica Superior</b> <b>Ingeniería Informática</b> <b>Prácticas de Sistemas Informáticos 2</b>					
<b>Grupo</b>	2401	<b>Práctica</b>	3	<b>Fecha</b>	02/04/2019
<b>Alumno/a</b>	Cea Tassel, Claudia				
<b>Alumno/a</b>	Riera Gómez, Juan				

## Práctica 3: Seguridad y disponibilidad

### Ejercicio número 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.

En primer lugar generamos la clave desde la máquina virtual 1:

```
si2@si2srv01:~$ ssh-keygen -t dsa
Generating public/private dsa key pair.
Enter file in which to save the key (/home/si2/.ssh/id_dsa):
Created directory '/home/si2/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/si2/.ssh/id_dsa.
Your public key has been saved in /home/si2/.ssh/id_dsa.pub.
The key fingerprint is:
e3:b0:81:70:be:f6:77:61:e3:a5:6f:b0:14:75:bd:94 si2@si2srv01
The key's randomart image is:
+--[ DSA 1024]----+
| .. |
| .E.|
| .. . . . |
| + . . . . |
| o o S . |
| . = .* . |
| o . .+ B |
| . . . = . |
| . . . o. |
+-----+
```

Importamos la clave pública a las otras 2 máquinas virtuales y vemos que efectivamente ahora desde la máquina virtual 1 podemos acceder por ssh a la 2 y a la 3 sin tener de introducir la contraseña:

si2@10.1.3.2

```
si2@si2srv01:~$ ssh -v si2@10.1.3.2
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 10.1.3.2 [10.1.3.2] 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 '10.1.3.2' 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 435
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 LANG = C
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: Thu Apr 11 06:07:22 2019 from 10.10.65.131
Loading es
si2@si2srv02:~$ 
```

si2@10.1.3.3

```
si2@si2srv01:~$ ssh -v si2@10.1.3.3
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 10.1.3.3 [10.1.3.3] 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 '10.1.3.3' 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 435
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 LANG = C
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: Thu Apr 11 06:08:18 2019 from 10.10.65.131
Loading es
si2@si2srv03:~$ 
```

## Ejercicio número 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.

Empezamos creando los nodos correspondientes:

```

si2@si2srv01:~$ asadmin --user admin --passwordfile /opt/SI2/passwordfile create-node-ssh --sshuser si2 --nodehost 10.1.3.2 --nodedir
/opt/glassfish4 Node01
Command create-node-ssh executed successfully.
si2@si2srv01:~$ asadmin --user admin --passwordfile /opt/SI2/passwordfile create-node-ssh --sshuser si2 --nodehost 10.1.3.3 --nodedir
/opt/glassfish4 Node02
Command create-node-ssh executed successfully.
si2@si2srv01:~$ asadmin --user admin --passwordfile /opt/SI2/passwordfile list-nodes
localhost-domain1 CONFIG localhost
Node01 SSH 10.1.3.2
Node02 SSH 10.1.3.3
Command list-nodes executed successfully.
si2@si2srv01:~$ asadmin --user admin --passwordfile /opt/SI2/passwordfile ping-node-ssh Node01
Successfully made SSH connection to node Node01 (10.1.3.2)
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 (10.1.3.3)
Command ping-node-ssh executed successfully.

```

Comprobamos en Glassfish que los nodos se han creado correctamente:

Nodes						
A node represents a host on which the GlassFish Server software is installed. A node must exist for every host on which GlassFish Server instances reside.						
Nodes (3)						
<input checked="" type="checkbox"/> <input type="checkbox"/> <input type="button" value="New..."/> <input type="button" value="Delete"/> <input type="button" value="Delete and Uninstall"/>						
Select	Name	Node Host	Type	Instances	Action	
<input type="checkbox"/>	Node01	10.1.3.2	SSH		<a href="#">Ping</a>	
<input type="checkbox"/>	Node02	10.1.3.3	SSH		<a href="#">Ping</a>	
<input type="checkbox"/>	localhost-domain1	localhost	CONFIG			

Ahora procedemos a la creación del cluster y de las instancias correspondientes:

```

si2@si2srv01:~$ export AS_ADMIN_USER=admin
si2@si2srv01:~$ export AS_ADMIN_PASSWORDFILE=/opt/SI2/passwordfile
si2@si2srv01:~$ asadmin create-cluster SI2Cluster
Command create-cluster executed successfully.
si2@si2srv01:~$ asadmin list-clusters
SI2Cluster not running
Command list-clusters executed successfully.
si2@si2srv01:~$ asadmin --user admin --passwordfile /opt/SI2/passwordfile create-instance --cluster SI2Cluster --node Node01 Instance0
1
Command _create-instance-filesystem executed successfully.
Port Assignments for server instance Instance01:
OSGI_SHELL_TELNET_PORT=26666
JAVA_DEBUGGER_PORT=29009
JMS_PROVIDER_PORT=27676
HTTP_LISTENER_PORT=28080
IIOP_SSL_LISTENER_PORT=23820
ASADMIN_LISTENER_PORT=24848
IIOP_SSL_MUTUALAUTH_PORT=23920
JMX_SYSTEM_CONNECTOR_PORT=28686
HTTP_SSL_LISTENER_PORT=28181
IIOP_LISTENER_PORT=23700
The instance, Instance01, was created on host 10.1.3.2
Command create-instance executed successfully.
si2@si2srv01:~$ asadmin --user admin --passwordfile /opt/SI2/passwordfile create-instance --cluster SI2Cluster --node Node02 Instance0
2
Command _create-instance-filesystem executed successfully.
Port Assignments for server instance Instance02:
OSGI_SHELL_TELNET_PORT=26666
JAVA_DEBUGGER_PORT=29009
JMS_PROVIDER_PORT=27676
HTTP_LISTENER_PORT=28080
IIOP_SSL_LISTENER_PORT=23820
ASADMIN_LISTENER_PORT=24848
IIOP_SSL_MUTUALAUTH_PORT=23920
JMX_SYSTEM_CONNECTOR_PORT=28686
HTTP_SSL_LISTENER_PORT=28181
IIOP_LISTENER_PORT=23700
The instance, Instance02, was created on host 10.1.3.3
Command create-instance executed successfully.

```

Listamos las instancias para comprobar su correcta creación y a continuación iniciamos el cluster:

```

si2@si2srv01:~$ asadmin --user admin --passwordfile /opt/SI2/passwordfile list-instances -l
Name      Host      Port   Pid Cluster    State
Instance01 10.1.3.2 24848 -- SI2Cluster  not running
Instance02 10.1.3.3 24848 -- SI2Cluster  not running
Command list-instances executed successfully.
si2@si2srv01:~$ asadmin --user admin --passwordfile /opt/SI2/passwordfile start-cluster SI2Cluster
Command start-cluster executed successfully.

```

A continuación, modificamos la configuración de las instancias del cluster siguiendo las instrucciones del enunciado:

### Configurations

Manage configurations, and view the target server instances or clusters using the configurations.

Configurations (3)		
	Name	Instance
<input type="checkbox"/>	SI2Cluster-config	Instance01 Running ✓ Instance02 Running ✓
<input type="checkbox"/>	default-config	
<input type="checkbox"/>	server-config	server Running ✓

### Options (29)

Options (29)	
Select	Value
<input type="checkbox"/>	-Dorg.glassfish.additionalOSGiBundlesToStart=org.apache.felix.shell,org.apache.felix.gogo.runtime,org.apache.felix.gogo lokale
<input type="checkbox"/>	-Djavax.net.ssl.keyStore=\${com.sun.aas.instanceRoot}/config/keystore.jks
<input type="checkbox"/>	-server
<input type="checkbox"/>	-DANTLR_USE_DIRECT_CLASS_LOADING=true
<input type="checkbox"/>	-Dfelix.fileinstall.bundles.startTransient=true
<input type="checkbox"/>	-Djavax.net.ssl.trustStore=\${com.sun.aas.instanceRoot}/config/cacerts.jks
<input type="checkbox"/>	-Dosgi.shell.telnet.ip=127.0.0.1
<input type="checkbox"/>	-Dfelix.fileinstall.log.level=3
<input type="checkbox"/>	-XX:+UnlockDiagnosticVMOptions
<input type="checkbox"/>	-Djava.security.auth.login.config=\${com.sun.aas.instanceRoot}/config/login.conf
<input type="checkbox"/>	-Dfelix.fileinstall.disableConfigSave=false
<input type="checkbox"/>	-Djava.awt.headless=true
<input type="checkbox"/>	-Djdbc.drivers=org.apache.derby.jdbc.ClientDriver
<input type="checkbox"/>	-Djdk.corba.allowOutputStreamSubclass=true
<input type="checkbox"/>	-Dosgi.shell.telnet.port=\${OSGI_SHELL_TELNET_PORT}
<input type="checkbox"/>	-Dosgi.shell.telnet.maxconn=1
<input type="checkbox"/>	-Xms128m
<input type="checkbox"/>	-Xmx128m
<input type="checkbox"/>	-Djava.ext.dirs=\${com.sun.aas.javaRoot}/lib/ext\${path.separator}\${com.sun.aas.javaRoot}/jre/lib/ext\${path.separator}
<input type="checkbox"/>	-Djava.security.policy=\${com.sun.aas.instanceRoot}/config/server.policy
<input type="checkbox"/>	-Dgosh.args=-noshutdown -c noop=true
<input type="checkbox"/>	-Dcom.sun.enterprise.config.config_environment_factory_class=com.sun.enterprise.config.serverbeans.Appse
<input type="checkbox"/>	-XX:MaxPermSize=96m
<input type="checkbox"/>	-Djava.endorsed.dirs=\${com.sun.aas.installRoot}/modules/endorsed\${path.separator}\${com.sun.aas.installRoot}/modules/endorsed
<input type="checkbox"/>	-Dcom.sun.enterprise.security.httpsOutboundKeyAlias=s1as
<input type="checkbox"/>	-Dfelix.fileinstall.poll=5000
<input type="checkbox"/>	-Dfelix.fileinstall.bundles.new.start=true
<input type="checkbox"/>	-XX:NewRatio=2
<input type="checkbox"/>	-Dfelix.fileinstall.dir=\${com.sun.aas.installRoot}/modules/autostart/

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

si2@10.1.3.2

```

si2@si2srv02:~$ ps -afe | grep java
0 S si2      1801   1  5 80  0 - 134358 futex_ 08:10 ?    00:00:16 /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.port=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=-noshutdown -c noop=true -Dcom.sun.enterprise.security.httpsOutboundKeyAlias=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/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,,,--nodedefdir,,,/opt/glassfish4,,,--node,,,Node01,,,Instance01 -instancename Instance01 -type INSTANCE -verbose false -instanceName com.sun.enterprise.admin.cli.AdminMain
0 R si2      1959  1377  0  80  0 -   465 -    08:15 pts/0    00:00:00 grep java
si2@si2srv02:~$ 
```

si2@10.1.3.3

```

si2@si2srv03:~$ ps -afe | grep java
0 S si2      1740   1  7 80  0 - 136019 futex_ 08:10 ?    00:00:14 /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.port=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.httpsOutboundKeyAlias=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,,,--nodedefdir,,,/opt/glassfish4,,,--node,,,Node02,,,Instance02 -instancename Instance02 -type INSTANCE -verbose false -instanceName com.sun.enterprise.admin.cli.AdminMain
0 R si2      1898  1311  0  80  0 -   465 -    08:13 pts/0    00:00:00 grep java
si2@si2srv03:~$ 
```

### Ejercicio número 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).

Comenzamos modificando la tabla pago para que contenga las nuevas columnas instancia e ip:

```
-- Tabla con pagos autorizados
-- Siempre vienen precedidos por una transaccion existente
CREATE TABLE pago
(
    -- idAutorizacion se autogenera con cada insercion
    idAutorizacion serial      not null,
    idTransaccion   char(16)     not null,
    codRespuesta    char(3)      not null default '000',
    importe         float        not null,
    idComercio      char(16)     not null,
    numeroTarjeta  char(19)     not null references tarjeta,
    fecha           timestamp    not null default current_timestamp,
    instancia       varchar(50),
    ip              varchar(50),
    CONSTRAINT      Pago_UC unique(idTransaccion, idComercio),
    PRIMARY KEY (idAutorizacion)
);
```

Modificamos PagoBean.java para incorporar las dos propiedades anteriores así como sus correspondientes getters y setters:

```
public class PagoBean {

    private String idTransaccion;
    private String idComercio;
    private double importe;
    private String ruta_retorno;

    private TarjetaBean tarjeta;

    /* Calculados por el gestor de medios de pago */
    private String codRespuesta;
    private String idAutorizacion;

    /* Modificaciones P3*/
    private String instancia;
    private String ip;
```

A continuación modificamos en los servlets ComienzaPago.java y ProcesaPago.java el método creaPago para que a la hora de crear el pago se añada la información relativa a los dos nuevos parámetros. Los cambios realizados son los mismos en ambos ficheros:

```
try{
    pago.setInstancia(System.getProperty("com.sun.aas.instanceName"));
    pago.setIp(java.net.InetAddress.getLocalHost().getHostAddress());
} catch (Exception e) {
    pago.setInstancia("");
    pago.setIp("");
    return pago;
}
```

Como último paso, modificamos VisaDAO.java para que a la hora de insertar el pago se incluyan los dos parámetros anteriores en la tabla pago:

```

private static final String INSERT_PAGOS_QRY =
    "insert into pago(" +
    "idTransaccion,importe,idComercio,numeroTarjeta,instancia,ip)" +
    " values (?,?,?,?,?,?)";

if (isPrepared() == true) {
    String insert = INSERT_PAGOS_QRY;
    errorLog(insert);
    pstmt = con.prepareStatement(insert);
    pstmt.setString(1, pago.getIdTransaccion());
    pstmt.setDouble(2, pago.getImporte());
    pstmt.setString(3, pago.getIdComercio());
    pstmt.setString(4, pago.getTarjeta().getNumero());
    pstmt.setString(5, pago.getInstancia());
    pstmt.setString(6, pago.getIp());
    ret = false;
    if (!pstmt.execute()
        && pstmt.getUpdateCount() == 1) {
        ret = true;
    }
}

string getQryInsertPago(PagoBean pago) {
    String qry = "insert into pago(" +
        "idTransaccion," +
        "importe,idComercio," +
        "numeroTarjeta," +
        "instancia," +
        "ip)" +
        " values (" +
        "'" + pago.getIdTransaccion() + "'," +
        pago.getImporte() + "," +
        "'" + pago.getIdComercio() + "'," +
        "'" + pago.getTarjeta().getNumero() + "'," +
        "'" + pago.getInstancia() + "'," +
        "'" + pago.getIp() + "'"
        + ")";
    return qry;
}

```

Ahora vamos a realizar un pago en cada instancia para verificar que se anotan correctamente el nombre de la instancia y la dirección IP. Comenzamos accediendo a los Web Application Links generados:

## Web Application Links

If the server or listener is not running, the link may not work. In this evr

**Application Name:** P3

**Links:**

- [Instance01] <http://10.1.3.2:28080/P3>
- [Instance01] <https://10.1.3.2:28181/P3>
- [Instance02] <http://10.1.3.3:28080/P3>
- [Instance02] <https://10.1.3.3:28181/P3>

Instancia 1:

← → C ⚠ No es seguro | 10.1.3.2:28080/P3/comienzapago

Aplicaciones Para acceder rápido a una página, arrástrala a est

## Pago con tarjeta

Numero de visa:

Titular:

Fecha Emisión:

Fecha Caducidad:

CVV2:

---

Id Transacción: 1

Id Comercio: 1

Importe: 10.0

---

Prácticas de Sistemas Informáticos II

← → C ⓘ No es seguro | 10.1.3.2:28080/P3/procesapago

Aplicaciones Para acceder rápido a una página, arrástrala a esta barra d

## Pago con tarjeta

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

idTransaccion: 1

idComercio: 1

importe: 10.0

codRespuesta: 000

idAutorizacion: 1

[Volver al comercio](#)

---

Prácticas de Sistemas Informáticos II

Instancia 2:

← → C ⚠ No es seguro | 10.1.3.3:28080/P3/comienzapago  
Aplicaciones Para acceder rápido a una página, arrástrala a es

## Pago con tarjeta

Numero de visa:

Titular:

Fecha Emisión:

Fecha Caducidad:

CVV2:

---

Id Transacción: 2

Id Comercio: 2

Importe: 20.0

---

Prácticas de Sistemas Informáticos II

← → C ⚠ No es seguro | 10.1.3.3:28080/P3/procesapago  
Aplicaciones Para acceder rápido a una página, arrástrala a esta barra d

## Pago con tarjeta

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

idTransaccion: 2

idComercio: 2

importe: 20.0

codRespuesta: 000

idAutorizacion: 2

[Volver al comercio](#)

---

Prácticas de Sistemas Informáticos II

Comprobamos en la base de datos que, efectivamente, los pagos se han hecho de manera satisfactoria y que tanto las instancias como las ip son correctas:

idautorizacion	idtransaccion	codrespuesta	importe	idcomercio	numerotarjeta	fecha	instancia	ip
1	1	000	10	1	1111 2222 3333 4444	2019-04-23 12:17:43.942412	Instance01	10.1.3.2
2	2	000	20	2	1111 2222 3333 4444	2019-04-23 12:18:34.812797	Instance02	10.1.3.3

(2 rows)

## **Ejercicio número 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: YYYYYYYYYYYYYYYYYYYYY

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: ZZZZZZZZZZZZZZZZZZZZZZZ

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.

Creamos el archivo de configuración del módulo mod\_proxy\_balancer:

```
si2@si2srv01:~$ sudo nano /etc/apache2/mods-available/proxy_balancer.conf
si2@si2srv01:~$ cat /etc/apache2/mods-available/proxy_balancer.conf
ProxyRequests Off
<Proxy balancer://SI2Cluster>
    BalancerMember http://10.1.3.2:28080 route=Instance01
    BalancerMember http://10.1.3.3:28080 route=Instance02
</Proxy>
<Location /P3>
    Order allow,deny
    Allow from all
    ProxyPass balancer://SI2Cluster/P3 stickysession=JSESSIONID|jsessionid scolonpathdelim=On
    ProxyPassReverse balancer://SI2Cluster/P3
</Location>
<Location /balancer-manager>
    SetHandler balancer-manager
</Location>
```

Este es nuestro balanceador de carga en estado inicial:

← → ⌂ ⓘ No es seguro | 10.1.3.1/balancer-manager  
Para acceder rápido a una página, arrástrala a esta barra de marcadores. Importar

## Load Balancer Manager for 10.1.3.1

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
<a href="http://10.1.3.2:28080">http://10.1.3.2:28080</a>	Instance01		1	0	Ok	0	0	0
<a href="http://10.1.3.3:28080">http://10.1.3.3:28080</a>	Instance02		1	0	Ok	0	0	0

Apache/2.2.14 (Ubuntu) Server at 10.1.3.1 Port 80

Sin jvmRoute:

Eliminamos las cookies del navegador y realizamos pagos hasta que uno falla debido a la falta de afinidad de sesión. Esta es la cookie JSESSIONID correspondiente al pago realizado de manera incorrecta:

← → ⌂ ⓘ 10.1.3.1/P3/procesapago ... 🌐 ⌂

## Pago con tarjeta

Pago incorrecto

Prácticas de Sistemas Informáticos II

Nombre	Dominio	Ruta	Caduca el	Último acceso
JSESSIONID	10.1.3.1	/P3	Sesión	Tue, 23 Apr 2019 16:57:51 GMT

**JSESSIONID: "b25074e1f3fad4474f97998f988e"**  
CreationTime: "Tue, 23 Apr 2019 16:57:51 GMT"  
Domain: "10.1.3.1"  
Expires: "Sesión"  
HostOnly: true  
HttpOnly: true  
LastAccessed: "Tue, 23 Apr 2019 17:01:18 GMT"  
Path: "/P3"  
Secure: false  
sameSite: "Unset"

Con jvmRoute:

**System Properties**

New values successfully saved.

Configuration Name: SI2Cluster-config

Dynamic Reconfiguration:

**Additional Properties (11)**

Select	Instance Variable Name	Default Value	Instance Values
<input type="checkbox"/>	ASADMIN_LISTENER_PORT	24848	Instance Values
<input type="checkbox"/>	HTTP_LISTENER_PORT	28080	Instance Values
<input type="checkbox"/>	HTTP_SSL_LISTENER_PORT	28181	Instance Values
<input type="checkbox"/>	IIOP_LISTENER_PORT	23700	Instance Values
<input type="checkbox"/>	IIOP_SSL_LISTENER_PORT	23820	Instance Values
<input type="checkbox"/>	IIOP_SSL_MUTUALAUTH_PORT	23920	Instance Values
<input type="checkbox"/>	JAVA_DEBUGGER_PORT	29009	Instance Values
<input type="checkbox"/>	JMS_PROVIDER_PORT	27676	Instance Values
<input type="checkbox"/>	JMX_SYSTEM_CONNECTOR_PORT	28686	Instance Values
<input type="checkbox"/>	OSGI_SHELL_TELNET_PORT	26666	Instance Values
<input type="checkbox"/>	jvmRoute	\$(com.sun.aas.instanceName}	Instance Values

Eliminamos las cookies del navegador y realizamos cierto número de pagos comprobando que, en efecto, no se produce ningún error. Tomamos la cookie JSESSIONID correspondiente a uno de estos pagos:



## Pago con tarjeta

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

idTransaccion: 9  
idComercio: 9  
importe: 90.0  
codRespuesta: 000  
idAutorizacion: 4

[Volver al comercio](#)

Prácticas de Sistemas Informáticos II

The screenshot shows the Chrome DevTools Network tab. On the left, there's a sidebar with options like Inspector, Consola, Depurador, Rendimiento, Memoria, Red, Almacenamiento, Accesibilidad, and a search bar. The main area displays a table of network requests. One row is selected, showing a cookie named 'JSESSIONID' with the value 'b2b5ea879b8a40a6b41b1c7bd13c.Instance01'. To the right of the table, detailed information about this cookie is shown in a panel. It includes the creation time ('Tue, 23 Apr 2019 17:07:54 GMT'), domain ('10.1.3.1'), expiration ('Sesión'), host-only status ('true'), http-only status ('true'), last accessed time ('Tue, 23 Apr 2019 17:08:14 GMT'), path ('/P3'), secure status ('false'), and same-site policy ('Unset'). Below this, the 'Valor interpretado' section shows the cookie value as an array with two elements: '0: "b2b5ea879b8a40a6b41b1c7bd13c"' and '1: "Instance01"'. The 'length' is listed as 2, and there's a reference to the '\_proto\_' object.

Podemos observar en las imágenes anteriores que las cookies son diferentes según si añadimos la propiedad "jvmRoute" al cluster o no. Si no se ha añadido esta propiedad vemos que la cookie no contiene información alguna sobre la instancia en la que se ha generado ni en la que está almacenada. Es por esto que al iniciar un pago en una instancia y al intentar procesarlo en otra, este pago falla: las instancias no contienen información sobre las cookies que están almacenadas en otras instancias. Por otro lado, al añadir la propiedad "jvmRoute" observamos que ahora las cookies sí tienen información sobre la instancia en la que se encuentran, de modo que cuando queremos realizar un pago, el balanceador lo traspasa a la instancia donde se inició, que es donde se encuentra la cookie.

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

En este caso sí se podría porque cada instancia está en un host distinto, y por tanto cuando el balanceador distribuya en cada instancia el valor en la cookie sería distinto. Sin embargo este método no valdría en general, porque podría haber varias instancias en un mismo host.

### **Ejercicio número 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.**

Realizamos dos pago de manera simultánea desde dos ordenadores distintos. Observamos que ambos se realizan de manera correcta y que cada uno de estos pagos ha sido procesado por una instancia distinta como se indica en sus respectivas cookies.

Ordenador 1: El pago se lleva a cabo en la instancia 1.



## **Pago con tarjeta**

Numero de visa:

Titular:

Fecha Emisión:

Fecha Caducidad:

CVV2:

---

Id Transacción: 15

Id Comercion: 15

Importe: 150.0

---

## Pago con tarjeta

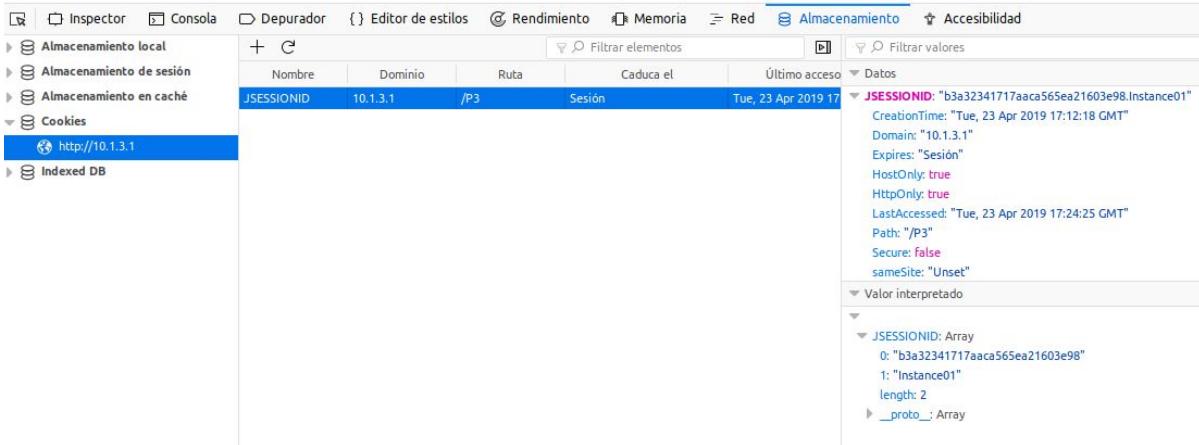
Pago realizado con éxito. A continuación se muestra el comprobante del mismo:

idTransaccion: 15  
idComercio: 15  
importe: 150.0  
codRespuesta: 000  
idAutorizacion: 6

[Volver al comercio](#)

---

Prácticas de Sistemas Informáticos II



The screenshot shows the Network tab of a browser developer tools interface. It lists several cookies, with one specific cookie highlighted: JSESSIONID. The details for this cookie are expanded, showing its creation time (Tue, 23 Apr 2019 17:12:18 GMT), domain (10.1.3.1), path (/P3), session ID (b3a32341717aaca565ea21603e98.Instance01), and expiration information. The cookie also has flags for HostOnly, HttpOnly, and Secure.

Ordenador 2: El pago se lleva a cabo en la instancia 2.



## Pago con tarjeta

Numero de visa:

Titular:

Fecha Emisión:

Fecha Caducidad:

CVV2:

---

Id Transacción: 16  
Id Comercio: 16  
Importe: 16.0

---

Prácticas de Sistemas Informáticos II

## Pago con tarjeta

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

```
idTransaccion: 16
idComercio: 16
importe: 16.0
codRespuesta: 000
idAutorizacion: 5
```

[Volver al comercio](#)

Prácticas de Sistemas Informáticos II

The screenshot shows the Network tab of a browser developer tools interface. It lists several cookies under the 'Cookies' section. One cookie is selected, highlighted with a blue background: 'JSESSIONID' with a value of 'b39f6151a70e6b22a...ab247.Instance02'. The right panel displays detailed information about this cookie, including its creation time ('Tue, 23 Apr 2019 17:23:47 GMT'), domain ('10.1.3.1'), path ('/P3'), and expiration ('Sesión'). It also shows its properties: HostOnly: true, HttpOnly: true, LastAccessed: 'Tue, 23 Apr 2019 17:24:10 GMT', Path: '/P3', Secure: false, and sameSite: 'Unset'. Below this, the 'Valor interpretado' (parsed value) is shown as an array: [0: "b39f6151a70e6b22a5f00a6ab247", 1: "Instance02", length: 2].

## Ejercicio número 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 –af | 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.**

Comenzamos identificando el pid del proceso java que parece ser el de la instancia, como veremos más adelante, acertamos al elegir el de PID 3762. En la siguiente captura vemos la terminal con cómo lo hemos localizado y finalizado:

```

si2@si2srv03:~$ ps -aef | grep java
si2      3762     1  4 10:20 ?        00:00:25 /usr/lib/jvm/java-8-oracle/bin/java -cp /opt/glassfish4/glassfish/modules/glassfish.ja
r -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.allowOutputS
treamSubclass=true -Dfelix.fileinstall.dir=/opt/glassfish4/glassfish/modules/autostart/ -Dorg.glassfish.additionalOSGiBundlesToStart=o
rg.apache.felix.shell,org.apache.felix.gogo.runtime,org.apache.felix.gogo.shell,org.apache.felix.gogo.command,org.apache.felix.fileins
tall -Dcom.sun.aas.installRoot=/opt/glassfish4/glassfish -Dfelix.fileinstall.poll=5000 -Djava.security.policy=/opt/glassfish4/glassfish/lib/endorsed
-Dfelix.fileinstall.bundles.startTransient=true -Dosgi.shell.telnet.maxconn=1 -Dfelix.fileinstall.log.level=3 -Dcom.sun.enterprise.con
fig.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/glassfish
4/Node02/Instance02 -Dosgi.shell.telnet.port=26666 -Dgosh.args[--noshutdown -c noop=true -Dcom.sun.enterprise.security.httpsOutboundKe
yAlias=sias -Dosgi.shell.telnet.ip=127.0.0.1 -DANTLR_USE_DIRECT_CLASS_LOADING=true -Djava.awt.headless=true -Djava.ext.dirs=/usr/lib/j
vm/java-8-oracle/lib/ext:/usr/lib/jvm/java-8-oracle/jre/lib/ext:/opt/glassfish4/Node02/Instance02/lib/ext -Djdbc.drivers=org.apache.de
rby.jdbc.ClientDriver -Djava.library.path=/opt/glassfish4/glassfish/lib:/usr/java/packages/lib/i386:/lib:/usr/lib com.sun.enterprise.g
lassfish.bootstrap.ASMain -upgrade false -read-stdin true -asadmin-args --host,,si2srv01,,,--port,,4848,,--secure=false,,--terse=f
alse,,--echo=false,,--interactive=false,,start-local-instance,,--verbose=false,,--watchdog=false,,--debug=false,,--nodedir,,/o
pt/glassfish4,,--node,,Node02,,Instance02 -instancename Instance02 -type INSTANCE -verbose false -instancedir /opt/glassfish4/Node0
2/Instance02 -asadmin-classpath /opt/glassfish4/glassfish/modules/admin-cli.jar -debug false -asadmin-classname com.sun.enterprise.adm
in.cli.AdminMain
si2      3958  3949  0 10:29 pts/0    00:00:00 grep java
si2@si2srv03:~$ kill -9 3762
si2@si2srv03:~$ 

```

Realizamos unas cuantas peticiones. En la siguiente captura obtenemos la salida de una de ellas:



## Pago con tarjeta

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

```

idTransaccion: 17
idComercio: 17
importe: 17.0
codRespuesta: 000
idAutorizacion: 7

```

[Volver al comercio](#)

Prácticas de Sistemas Informáticos II

Nombre	Dominio	Ruta	Caduca el	Último acceso el	Valor
JSESSIONID	10.1.3.1	/P3	Sesión	Tue, 23 Apr 2019 17:30:51...	b4017d51b9405...

The screenshot shows the Chrome DevTools Network tab. On the left, there's a sidebar with various storage types: Almacenamiento local, Almacenamiento de sesión, Almacenamiento en caché, Cookies, http://10.1.3.1 (which is selected and highlighted in blue), and Indexed DB. The main area displays a table of cookies. One cookie, JSESSIONID, is selected and expanded. Its details are shown on the right: CreationTime: "Tue, 23 Apr 2019 17:23:47 GMT", Domain: "10.1.3.1", Expires: "Sesión", HostOnly: true, HttpOnly: true, LastAccessed: "Tue, 23 Apr 2019 17:24:10 GMT", Path: "/P3", Secure: false, and sameSite: "Unset". Below this, the "Valor interpretado" section shows the value as an array: [0: "b39f6151a70e6b22a5f00a6ab247", 1: "Instance02"], and the \_\_proto\_\_ property.

La siguiente captura es una captura del *balancer-manager* en el que se observa cómo la instancia 2 ha sido marcada como errónea.

## Load Balancer Manager for 10.1.3.1

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
<a href="http://10.1.3.2:28080">http://10.1.3.2:28080</a>	Instance01		1	0	Ok	28	18K 30K
<a href="http://10.1.3.3:28080">http://10.1.3.3:28080</a>	Instance02		1	0	Err	25	13K 26K

Apache/2.2.14 (Ubuntu) Server at 10.1.3.1 Port 80

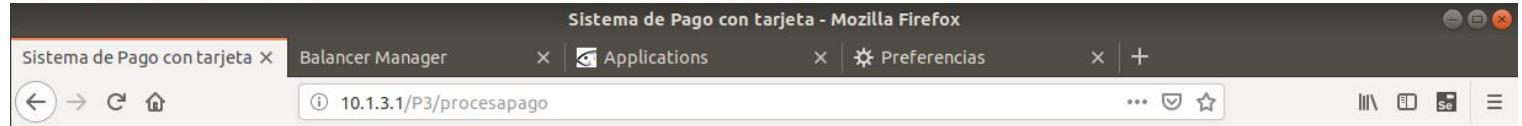
### Ejercicio número 7:

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.

Iniciamos la instancia. En la siguiente captura observamos el comando utilizado y su salida:

```
si2@si2srv01:~$ asadmin start-instance Instance02
CLI801 Instance is already synchronized
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 10.1.3.3
Command start-instance executed successfully.
si2@si2srv01:~$ 
```

Y ahora realizamos unas cuantas peticiones hasta que el balanceador detecta que la instancia ha vuelto a estar en pie. En esta captura vemos la salida de una de las peticiones que hemos hecho, la cookie de sesión abajo donde bajo el texto “Valor interpretado” podemos ver que tenemos la instancia 2:



## Pago con tarjeta

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

```
idTransaccion: 30
idComercio: 30
importe: 30.0
codRespuesta: 000
idAutorizacion: 18
```

[Volver al comercio](#)

Prácticas de Sistemas Informáticos II

Nombre	Dominio	Ruta	Caduca el	Último acceso el	Valor
JSESSIONID	10.1.3.1	/P3	Sesión	Tue, 23 Apr 2019 17:49:37...	b51449c1507e214ef6910f8b6b60;Instance02

Y efectivamente lo detecta y se refleja en el *balancer-manager*, en esta captura del mismo observamos que el estado de la instancia 2 vuelve a estar "Ok":

## Load Balancer Manager for 10.1.3.1

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
<a href="http://10.1.3.2:28080">http://10.1.3.2:28080</a>	Instance01	1	0	Ok	54	35K 62K
<a href="http://10.1.3.3:28080">http://10.1.3.3:28080</a>	Instance02	1	0	Ok	32	17K 32K

Apache/2.2.14 (Ubuntu) Server at 10.1.3.1 Port 80

### Ejercicio número 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 ComienzaPago.
- 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.

Observamos que se nos ha asociado la instancia 2 en la siguiente captura, en la que nos encontramos en mitad del proceso de pago y abajo vemos la cookie de sesión que nos mantiene asignada la instancia 2:

Sistema de Pago con tarjeta - Mozilla Firefox

Sistema de Pago con tarjeta X Balancer Manager X Applications X Preferencias X +

10.1.3.1/P3/comienzapago

## Pago con tarjeta

Numero de visa:

Titular:

Fecha Emisión:

Fecha Caducidad:

CVV2:

Id Transacción: 32  
 Id Comercio: 32  
 Importe: 32.0

Inspector Consola Depurador Editor de estilos Rendimiento Memoria Red Almacenamiento Accesibilidad

Almacenamiento local Almacenamiento de sesión Almacenamiento en caché Cookies http://10.1.3.1 Indexed DB

Nombre	Dominio	Ruta	Caduca el	Último acceso el	Valor
JSESSIONID	10.1.3.1	/P3	Sesión	Tue, 23 Apr 2019 17:52:56...	b51449c1507e214ef...b6b60.Instance02

Filtrar elementos Filtrar valores Datos JSESSIONID: "b51449c1507e214ef...b6b60.Instance02" CreationTime: "Tue, 23 Apr 2019 17:49:16 GMT" Domain: "10.1.3.1" Expires: "Sesión" HostOnly: true HttpOnly: true LastAccessed: "Tue, 23 Apr 2019 17:49:37 GMT" Path: "/P3" Secure: false sameSite: "Unset" Valor interpretado JSESSIONID: Array 0: "b51449c1507e214ef6910f8b6b60" 1: "Instance02" length: 2 \_\_proto\_\_: Array

Ahora vamos a matar el proceso de la instancia que se nos ha asociado. En esta captura de pantalla vemos la terminal con los comandos y la salida que hemos utilizado para buscar el proceso de la instancia con pid 4182, y para matarlo:

```
si2@si2srv03:~$ ps -afe | grep java
si2 4182 1 99 10:36 ? 00:00:17 /usr/lib/jvm/java-8-oracle/bin/java -cp /opt/glassfish4/glassfish/modules/glassfish.ja
r -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.allowOutputS
treamSubclass=true -Dfelix.fileinstall.dir=/opt/glassfish4/glassfish/modules/autostart/ -Dorg.glassfish.additionalOSGiBundlesToStart=o
rg.apache.felix.shell.org.apache.felix.gogo.runtime.org.apache.felix.gogo.shell.org.apache.felix.gogo.command.org.apache.felix.fileins
tall -Dcom.sun.aas.installRoot=/opt/glassfish4/glassfish -Dfelix.fileinstall.poll=5000 -Djava.security.policy=/opt/glassfish4/Node02/I
nstance02/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.con
fig.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/glassfish
4/Node02/Instance02 -Dosgi.shell.telnet.port=26666 -Dgosh.args=-noshutdown -c noop=true -Dcom.sun.enterprise.security.httpsOutboundKe
yAlias=sias -Dosgi.shell.telnet.ip=127.0.0.1 -DANTLR_USE_DIRECT_CLASS_LOADING=true -Djava.awt.headless=true -Djava.ext.dirs=/usr/lib/j
vm/java-8-oracle/lib/ext:/usr/lib/jvm/java-8-oracle/jre/lib/ext:/opt/glassfish4/Node02/Instance02/lib/ext -Djdbc.drivers=org.apache.de
rby.jdbc.ClientDriver -Djava.library.path=/opt/glassfish4/glassfish/lib:/usr/java/packages/lib/i386:/lib:/usr/lib com.sun.enterprise.g
lassfish.bootstrap.ASMain -upgrade false -read-stdin true -asadmin-args --host,,si2srv01,,,--port,,,4848,,--secure=false,,--terse=f
alse,,--echo=false,,--interactive=false,,--start-local-instance,,--verbose=false,,--watchdog=false,,--debug=false,,--nodadir,,,/o
pt/glassfish4,,,-node,,,Node02,,,Instance02 -instancename Instance02 -type INSTANCE -verbose false -instancedir /opt/glassfish4/Node0
2/Instance02 -asadmin-classpath /opt/glassfish4/glassfish/modules/admin-cli.jar -debug false -asadmin-classname com.sun.enterprise.adm
in.cli.AdminMain
si2 4334 3949 0 10:36 pts/0 00:00:00 grep java
si2@si2srv03:~$ kill -9 4182
```

- 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.

Observemos el error que aparece. Esta es la salida del proceso de pago, además, abajo se observa el contenido de la cookie de sesión que nos asigna la instancia 2:

The screenshot shows a Firefox browser window with the title "Sistema de Pago con tarjeta - Mozilla Firefox". The address bar displays "10.1.3.1/P3/procesapago". The main content area shows a heading "Pago con tarjeta" and a message "Pago incorrecto". Below this, there is a link "Prácticas de Sistemas Informáticos II". The bottom half of the screen shows the Firefox developer tools Network tab. The sidebar on the left lists "Almacenamiento local", "Almacenamiento de sesión", "Almacenamiento en caché", "Cookies", and "Indexed DB". Under "Cookies", the entry for "http://10.1.3.1" is selected, showing a table with one row for "JSESSIONID". The table columns are "Nombre", "Dominio", "Ruta", "Caduca el", "Último acceso el", and "Valor". The "Valor" column shows the value "b551e8fc61ced0...". To the right of the table, the "Almacenamiento" panel is expanded, showing detailed information for the "JSESSIONID" cookie. It includes fields like "CreationTime", "Domain", "Expires", "HostOnly", "HttpOnly", "LastAccessed", "Path", "Secure", and "sameSite". The "Valor interpretado" section shows the JSON representation of the cookie: "JSESSIONID: Array [0: "b51449c1507e214ef6910f8b6b60", 1: "Instance02", length: 2]".

Este error ocurre porque nuestra sesión, que almacenaba el id de transacción, el id de comercio y el importe se hallaban en la otra instancia. Al haberla terminado, el balanceador detecta que esa instancia no está funcionando y nos dirige la petición a la instancia 1, donde no tenemos el id de sesión. Cuando intenta hacer un pago VisaDAO no logra acceder a esa información, y devuelve pago incorrecto. Este es un error de afinidad de sesión.

## **Ejercicio número 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?

Tras haber limpiado los datos internos del clúster referentes al reparto de datos entre instancias, y haber limpiado la base de datos de pagos, hemos editado el script de pruebas JMeter para que haga 1000 peticiones. También hemos borrado la opción de jvmRoute para que las peticiones se distribuyan libremente según el criterio del balanceador con el fin de observar mejor el algoritmo, y hemos hecho las peticiones a la ruta /P3/testbd.jsp para evitar errores de afinidad de sesión.

Para llevar la contabilidad de qué peticiones van a qué instancia, hemos mirado cómo iba el reparto antes de realizar las pruebas:

## **Load Balancer Manager for 10.1.3.1**

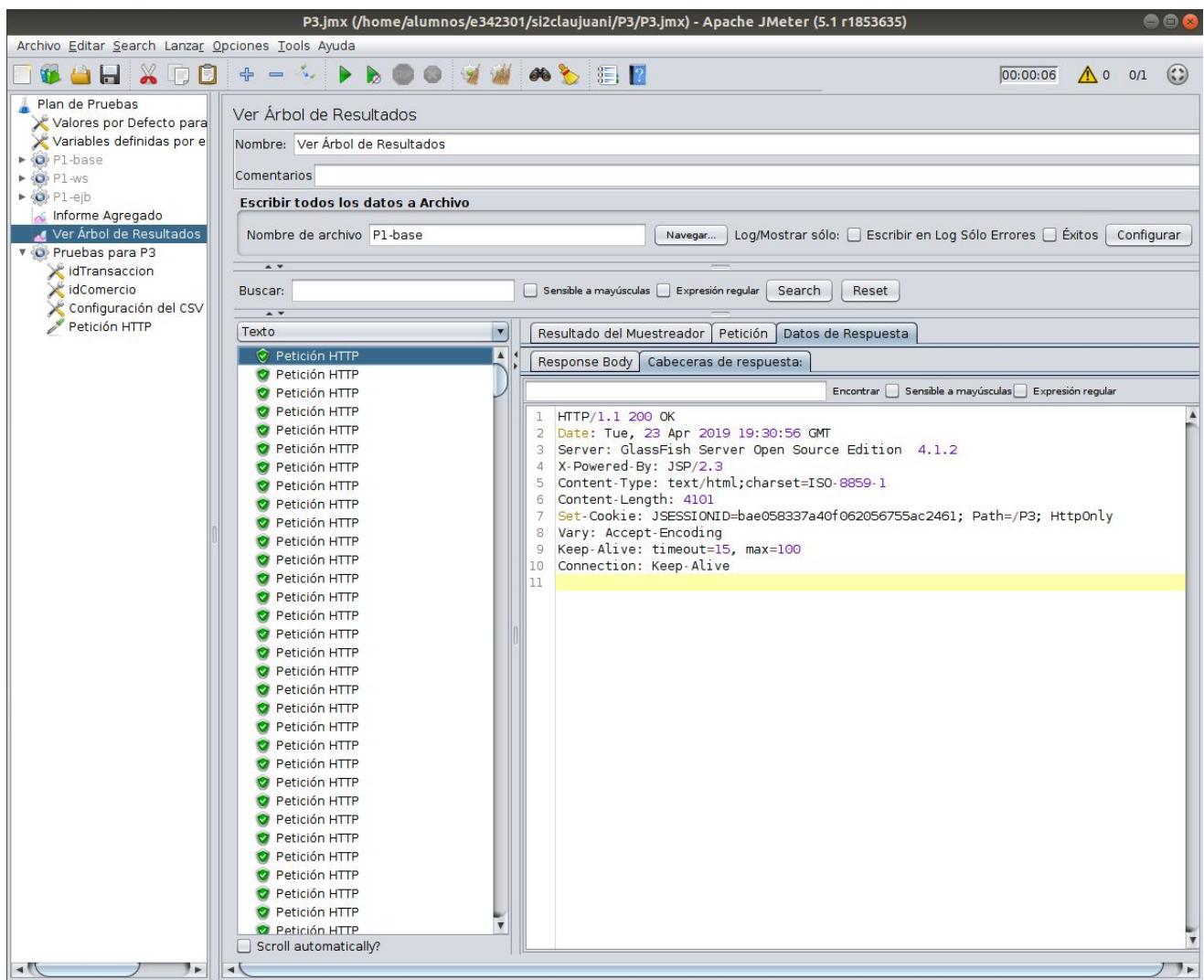
Server Version: Apache/2.2.14 (Ubuntu)  
Server Built: Nov 3 2011 03:31:27

### **LoadBalancer Status for balancer://si2cluster**

StickySession	Timeout	FailoverAttempts	Method				
JSESSIONID jsessionid	0	1	byrequests				
Worker URL	Route	RouteRedir	Factor	Set	Status	Elected To	From
<a href="http://10.1.3.2:28080">http://10.1.3.2:28080</a>	Instance01		1	0	Ok	1006	210K 1.1M
<a href="http://10.1.3.3:28080">http://10.1.3.3:28080</a>	Instance02		1	0	Ok	1004	516K 208K

*Apache/2.2.14 (Ubuntu) Server at 10.1.3.1 Port 80*

Observamos que a la instancia 1 se han hecho 1006 peticiones y a la instancia 2 se han hecho 1004 peticiones. Preparamos ahora el script de pruebas de JMeter, y realizamos 1000 peticiones. En la siguiente captura observamos el estado de JMeter cuando el script ya ha acabado:



Ahora comprobamos cómo se han repartido las peticiones en el servidor:

## Load Balancer Manager for 10.1.3.1

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
<a href="http://10.1.3.2:28080">http://10.1.3.2:28080</a>	Instance01		1	0	Ok	1505	472K 3.1M
<a href="http://10.1.3.3:28080">http://10.1.3.3:28080</a>	Instance02		1	0	Ok	1505	779K 2.2M

*Apache/2.2.14 (Ubuntu) Server at 10.1.3.1 Port 80*

Observamos que a la instancia 1 se han enviado 499 peticiones ( $1505 - 1006 = 499$ ) y a la instancia 2 se han enviado 501 ( $1505 - 1004 = 501$ ) peticiones. Esta similitud en el número de

peticiones balanceadas hacia cada una de las instancias parece indicar un algoritmo de Round-Robin o similar en el balanceo de la carga, es decir, rotación entre las instancias.