

		Escuela Politécnica Superior Ingeniería Informática Prácticas de Sistemas Informáticos 2			
Grupo	2401	Práctica	3	Fecha	03/05/2017
Alumno/a	Fuentes, de Uña, Sergio				
Alumno/a	Burrero, Perdices, Daniel				

Práctica 3: Seguridad y disponibilidad

Ejercicio 1:

Preparar 3 máquinas virtuales 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.

Una vez realizado este punto, detendremos las tres máquinas virtuales y obtendremos una copia de las mismas a algún medio externo (USB) para los consiguientes apartados de esta práctica.

También es recomendable que preserve los directorios `.ssh` de cada uno de los nodos.

Se han preparado las máquinas virtuales tal y como se indica en el enunciado. A continuación se muestran las salidas detalladas del comando `ssh` desde la máquina virtual 1 a las máquinas 2 y 3, respectivamente:

```
si2@si2srv01:~$ ssh -v si2@10.1.6.2
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 10.1.6.2 [10.1.6.2] 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 '10.1.6.2' 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: Tue Apr  4 07:39:05 2017 from 10.1.6.1
Loading es
si2@si2srv02:~$

```

```

si2@si2srv01:~$ ssh -v si2@10.1.6.3
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 10.1.6.3 [10.1.6.3] 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 '10.1.6.3' 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: Tue Apr  4 07:38:22 2017 from 10.1.0.1
Loading es
si2@si2srv03:~$

```

Se comprueba que el intercambio de claves es satisfactorio y se garantiza acceso SSH sin contraseña.

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 2 están efectivamente corriendo en cada una de las dos máquinas virtuales. Adjunte evidencias a la memoria de la práctica.

Se ha configurado el cluster según lo indicado en las instrucciones. Se muestra la salida del listado de clusters:

```

si2@si2srv01:~$ asadmin list-instances -l
Name      Host      Port  Pid   Cluster    State
Instance01 10.1.6.2  24848 1675  SI2Cluster running
Instance02 10.1.6.3  24848 1555  SI2Cluster running
Command list-instances executed successfully.

```

Y los procesos Java ejecutándose en las máquinas 2 y 3, respectivamente, incluyendo sus PID:

```

si2@si2srv02:~$ ps -aefl | grep java
0 S si2      1675      1 26  80   0 - 134765 futex_ 07:55 ?          00:00:33 /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
-Djava.security.auth.login.config=/opt/glassfish4/Node01/Instance01/config/login.conf
-Dfelix.fileinstall.disableConfigSave=false
-Djavax.net.ssl.trustStore=/opt/glassfish4/Node01/Instance01/config/cacerts.jks
-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
-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.aas.installRoot=/opt/glassfish4/glassfish -Dfelix.fileinstall.poll=5000
-Dcom.sun.enterprise.security.httpsOutboundKeyAlias=s1as
-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 -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
-Djavax.net.ssl.keyStore=/opt/glassfish4/Node01/Instance01/config/keystore.jks
-Djava.library.path=/opt/glassfish4/glassfish/lib:/usr/java/packages/lib/i386:/lib:/usr/lib
com.sun.enterprise.glassfish.bootstrap.ASMMain -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,,N
ode01,,Instance01 -instancename 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
```

```
si2@si2srv03:~$ ps -aefl | grep java
0 S si2      1555      1 13  80   0 - 137641 futex_ 07:55 ?          00:00:30 /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
-Djava.security.auth.login.config=/opt/glassfish4/Node02/Instance02/config/login.conf
-Dfelix.fileinstall.disableConfigSave=false
-Djavax.net.ssl.trustStore=/opt/glassfish4/Node02/Instance02/config/cacerts.jks
-Dfelix.fileinstall.dir=/opt/glassfish4/glassfish/modules/autostart/
-Dorg.glassfish.additionalOSGiBundlesToStart=org.apache.felix.shell,org.apache.felix.gogo.runtime,org.apache.fel
ix.gogo.shell,org.apache.felix.gogo.command,org.apache.felix.fileinstall
-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.aas.installRoot=/opt/glassfish4/glassfish -Dfelix.fileinstall.poll=5000
-Dcom.sun.enterprise.security.httpsOutboundKeyAlias=s1as
-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.AppserverConf
igEnvironmentFactory -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
-Djavax.net.ssl.keyStore=/opt/glassfish4/Node02/Instance02/config/keystore.jks
-Djava.library.path=/opt/glassfish4/glassfish/lib:/usr/java/packages/lib/i386:/lib:/usr/lib
com.sun.enterprise.glassfish.bootstrap.ASMMain -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,,,N
ode02,,,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
```

Se aprecia cómo el cluster arranca los procesos Java necesarios en los nodos de forma remota y automática.

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).

Se han identificado los Web Application Links generados para el cluster:

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://10.1.6.2:28080/P3
[Instance01]	https://10.1.6.2:28181/P3
[Instance02]	http://10.1.6.3:28080/P3
[Instance02]	https://10.1.6.3:28181/P3

A continuación, se ha realizado un único pago en cada nodo y se han listado las entradas de la tabla de pagos:

```
visa=# select * from pago;
 idautorizacion | idtransaccion | codrespuesta | importe | idcomercio | numerotarjeta |
 fecha         | instancia    | ip          |
-----+-----+-----+-----+-----+-----+-----
-----+-----+-----+-----+-----+-----+-----
```

1		33			000			33		33			4579	1165	7919	4384		
2017-04-04 08:48:01.637379		Instance01		10.1.6.2														
2		66			000			66		66			6496	3318	3339	9724		
2017-04-04 08:49:18.169249		Instance02		10.1.6.3														
(2 filas)																		

Se observa que las instancias funcionan correctamente y se almacenan su nombre e IP de manera adecuada.

Ejercicio 4:

Probar la influencia de jvmRoute en la afinidad de sesión.

- [illegible]

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

Se han realizado las pruebas de afinidad de sesión solicitadas.

Primero, se ha ejecutado un pago sin la propiedad `ivmRoute` activada en el cluster. Se muestra la cookie:

Name	Value	Domain	Path	Expires / M...	Size	HTTP
JSESSIONID	171ec481bb039bc675ea364c651f	10.1.6.1	/P3	Session	38	✓

En este caso, el valor de la cookie *JSESSIONID* es un hash aleatorio, y al intentar realizar pagos se produce un fallo, debido a que no existe afinidad de sesión entre las distintas instancias del cluster.

Una vez añadida la propiedad *jvmRoute* al cluster, eliminadas las cookies y realizado un nuevo pago:

Name	Value	Domain	Path	Expires / M...	Size	HTTP
JSESSIONID	1780b9a2911a008310720d07f7e5.Instance01	10.1.6.1	/P3	Session	49	✓

Se observa cómo el hash aleatorio de la cookie `JSESSIONID` incluye ahora el nombre de la instancia, haciendo posible mantener la afinidad de sesión. De esta manera, pueden realizarse pagos sucesivos correctamente, sin que ocurra el error previamente comentado.

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.

Se ha realizado un pago desde un ordenador distinto. Se observa que se inicia la sesión en la instancia 2:

Name	Value	Domain	Path	Expires / M...	Size	HTTP
JSESSIONID	17a6a44eac28b098fe60e487d2eb.Instance02	10.1.6.1	/P3	Session	49	✓

De esta manera, se ve cómo las peticiones y respectivas sesiones se reparten entre los nodos del cluster.

La siguiente captura de pantalla muestra el balanceador de carga con ambas instancias activas:

Load Balancer Manager for 10.1.6.1

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://10.1.6.2:28080	Instance01		1	0	Ok	23	16K	26K
http://10.1.6.3:28080	Instance02		1	0	Ok	13	8.0K	12K

Apache/2.2.14 (Ubuntu) Server at 10.1.6.1 Port 80

En él se muestra datos sobre el estado (*Status*), el número de peticiones atendidas (*Elected*) y la cantidad de información enviada (*To*) y recibida (*From*) correspondiente a cada instancia, junto con su enlace.

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.

Se ha ejecutado el comando *kill* contra el proceso Java correspondiente a la instancia número 2 del cluster.

Tras realizar un par de pagos desde ordenadores distintos, se observa que todas las peticiones se dirigen a la instancia número 1 del cluster.

Cookie de sesión al realizar un pago desde un ordenador:

Name	Value	Domain	Path	Expires / M...	Size	HTTP
JSESSIONID	17ea77970452205efd04e41b254.Instance01	10.1.6.1	/P3	Session	49	✓

Cookie de sesión al realizar un pago desde otro ordenador:

Name	Value	Domain	Path	Expires / M...	Size	HTTP
JSESSIONID	17f92622cdc2004ff755738056f0.Instance01	10.1.6.1	/P3	Session	49	✓

Se comprueba que efectivamente, ambas están siendo gestionadas por la instancia 1.

Finalmente, se verifica que el nodo 2 ha sido marcado como erróneo en la página del balanceador de carga:

Load Balancer Manager for 10.1.6.1

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://10.1.6.2:28080	Instance01		1	0	Ok	16	11K 17K
http://10.1.6.3:28080	Instance02		1	0	Err	10	6.0K 9.5K

Apache/2.2.14 (Ubuntu) Server at 10.1.6.1 Port 80

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.

Tras arrancar la instancia número 2 del cluster de nuevo, se realiza un nuevo pago y se observa que se dirige a dicha instancia, que vuelve a estar operativa y funciona correctamente:

Name	Value	Domain	Path	Expires / M...	Size	HTTP
JSESSIONID	1883a2c1d8bb7dc099ee034c3973.Instance02	10.1.6.1	/P3	Session	49	✓

Load Balancer Manager for 10.1.6.1

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://10.1.6.2:28080	Instance01		1	0	Ok	23	16K 26K
http://10.1.6.3:28080	Instance02		1	0	Ok	13	8.0K 12K

Apache/2.2.14 (Ubuntu) Server at 10.1.6.1 Port 80

Tras la recuperación del nodo 2, los pagos sucesivos se distribuyen correctamente entre ambas instancias.

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 `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.
- 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.

Se ha realizado un pago de la manera indicada. Una vez en la página *comienzapago*, la cookie de sesión muestra que el pago está siendo atendido por la instancia número 2 del cluster:

Name	Value	Domain	Path	Expires / M...	Size	HTTP
JSESSIONID	18c77db1dcc87fb6976e6c36c6ea.Instance02	10.1.6.1	/P3	Session	49	✓

Tras detener la instancia 2, completar los datos del pago y enviar la petición, la cookie de sesión cambia:

Name	Value	Domain	Path	Expires / M...	Size	HTTP
JSESSIONID	18dbf3aeabb65055433c26ca5061.Instance01	10.1.6.1	/P3	Session	49	✓

Sin embargo, el pago falla mostrando un error. Esto se debe a que, aunque la cookie de sesión se migre a la instancia 1, los datos de dicha sesión y de la petición en curso se encuentran en el nodo servidor 2, cuyo servicio está detenido. Por tanto, el flujo de datos de la petición de pago se ve detenido y produce el error mostrado.

Pago con tarjeta

Pago incorrecto

Prácticas de Sistemas Informáticos II

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.

Tras eliminar los pagos previos, se ha ejecutado el ciclo de 1000 pruebas de un solo hilo contra el cluster.

Una vez finalizada, la página de balanceo de carga muestra los siguientes datos:

Worker URL	Route	RouteRedir	Factor	Set	Status	Elected	To	From
http://10.1.6.2:28080	Instance01		1	0	Ok	500	264K	515K
http://10.1.6.3:28080	Instance02		1	0	Ok	500	264K	515K

Apache/2.2.14 (Ubuntu) Server at 10.1.6.1 Port 80

Se observa que el reparto es completamente uniforme entre las dos instancias, de manera que cada una ha gestionado la mitad de peticiones (500), y ha enviado y recibido exactamente la misma cantidad de datos.

Los datos obtenidos y la experiencia con el balanceo de carga durante el transcurso de la práctica, apuntan a la conclusión de que el balanceador de carga emplea algún algoritmo de tipo *round-robin*, distribuyendo peticiones entre instancias alternativamente. Sin embargo, los resultados no son concluyentes, ya que podría tratarse de *round-robin* por lotes (X peticiones para cada instancia alternativamente), o incluso tratarse de otro algoritmo de distribución que en este caso concreto se comporta de manera completamente uniforme.