

### 9.3. Cluster con *Apache* y *Tomcat* (balanceo de carga)

En esta práctica vamos a crear un cluster con dos servidores *Tomcat*. Utilizaremos *Apache* y el módulo **mod\_jk** para realizar un balanceo de carga entre los dos servidores (probaremos con la aplicación **Calendar**). El entorno que crearemos es el siguiente:

- *Apache* (**ServidorLinuxXX**) recibirá peticiones HTTP y realizará un balanceo de carga entre dos servidores *Tomcat*. Para ello se usará el módulo **mod\_jk**.
- Los servidores *Tomcat* en **ServidorLinuxXX** y *Tomcat* en **ServidorWindowsXX**.

**Nota:** Lo ideal sería instalar *Apache* en otra máquina diferente.

#### 1. Configuración de *mod\_jk* en *Apache*.

- 1.1. Inicia sesión en **ServidorLinuxXX** con un usuario con privilegios de administración.
- 1.2. Edita el fichero `/etc/libapache2-mod-jk/workers-properties` y realiza la siguiente configuración, Figura 9.19.
- 1.3. Reinicia *Apache*.
- 1.4. Edita el fichero `/etc/apache2/sites-available/000-default` y añade las siguientes directivas para redirigir las peticiones de la aplicación **Calendar** desde *Apache* a balanceador Figura 9.20.
- 1.5. Reinicia *Apache*.

#### 2. Configuración de *Tomcat* en **ServidorLinuxXX** (nodo1).

- 2.1. Inicia sesión en **ServidorLinuxXX** con un usuario con privilegios de administración.
- 2.2. *Tomcat* en **ServidorLinuxXX** ya está configurado. Revisemos su configuración.
- 2.3. Edita el fichero `/var/lib/tomcat7/conf/server.xml/` y observa que está habilitado el conector AJP, Figura 9.21.
- 2.4. Verifica que la aplicación **Calendar** está desplegada.

#### 3. Configuración de *Tomcat* en **ServidorWindowsXX** (nodo2).

- 3.1. Inicia sesión en **ServidorWindowsXX** con un usuario con privilegios de administración.
- 3.2. Descarga *Java SE Development Kit* de <http://www.oracle.com/technetwork/java/javase/downloads/index.html>. En el momento de realizar este manual estaba disponible la versión 7.
- 3.3. Instala la versión de JDK descargada previamente.
- 3.4. Descarga de <http://tomcat.apache.org/> la versión 32-bit/64-bit Windows Service Installer.
- 3.5. Instala la versión descargada anteriormente, marcando la instalación completa. Configura el usuario **tomcat** para usar la consola de administración web (manager) y observa que se habilita el conector AJP, Figura 9.22. Deja el resto de opciones por defecto.
- 3.6. Consulta el contenido del fichero `C:\Program Files\Apache Software Foundation\Tomcat 7.0\conf\server.xml` y observa que está habilitado el conector AJP.
- 3.7. Habilita una regla *Firewall* para que se permitan conexiones a los conectores HTTP (8080/TCP) AJP (8009/TCP) de *Tomcat*.

```

#----- worker list -----
#-----
#
# The workers that your plugins should create and work with
#
worker.list=loadbalancer
#
#----- ajp13_worker WORKER DEFINITION -----
#-----
# Defining a worker named ajp13_worker and of type ajp13
# Note that the name and the type do not have to match.
#
worker.nodo1.port=8009
worker.nodo1.host=localhost
worker.nodo1.type=ajp13
#
# Specifies the load balance factor when used with
# a load balancing worker.
# Note:
# ----> lbfactor must be > 0
# ----> Low lbfactor means less work done by the worker.
worker.nodo1.lbfactor=5

worker.nodo2.port=8009
worker.nodo2.host=192.168.1.18
worker.nodo2.type=ajp13
#
# Specifies the load balance factor when used with
# a load balancing worker.
# Note:
# ----> lbfactor must be > 0
# ----> Low lbfactor means less work done by the worker.
worker.nodo2.lbfactor=5

#
# Specify the size of the open connection cache.
#worker.ajp13_worker.cachesize
#
#----- DEFAULT LOAD BALANCER WORKER DEFINITION -----
#-----
# The loadbalancer (type lb) workers perform wighted round-robin
# load balancing with sticky sessions.
# Note:
# ----> If a worker dies, the load balancer will check its state
#         once in a while. Until then all work is redirected to peer
#         workers.
worker.loadbalancer.type=lb
worker.loadbalancer.balance_workers=nodo1,nodo2
worker.loadbalancer.sticky_session=True

```

Figura 9.19: Fichero /etc/libapache2-mod-jk/workers-properties

```

<VirtualHost *:80>
    ServerAdmin webmaster@localhost
    DocumentRoot /var/www

    ProxyPreserveHost on
    ProxyPass /sample http://localhost:8080/sample
    ProxyPassReverse /sample http://localhost:8080/sample
    ProxyPass /curso http://localhost:8080/curso
    ProxyPassReverse /curso http://localhost:8080/curso
    ProxyPass /compras http://localhost:8080/compras
    ProxyPassReverse /compras http://localhost:8080/compras

    JkMount /Calendar loadbalancer
    JkMount /Calendar/* loadbalancer

```

Figura 9.20: Fichero `/etc/apache2/sites-availables/000-default`

```

<!-- Define an AJP 1.3 Connector on port 8009 -->
<Connector port="8009" protocol="AJP/1.3" redirectPort="8443" />

```

Figura 9.21: Fichero `/var/lib/tomcat7/conf/server.xml/`

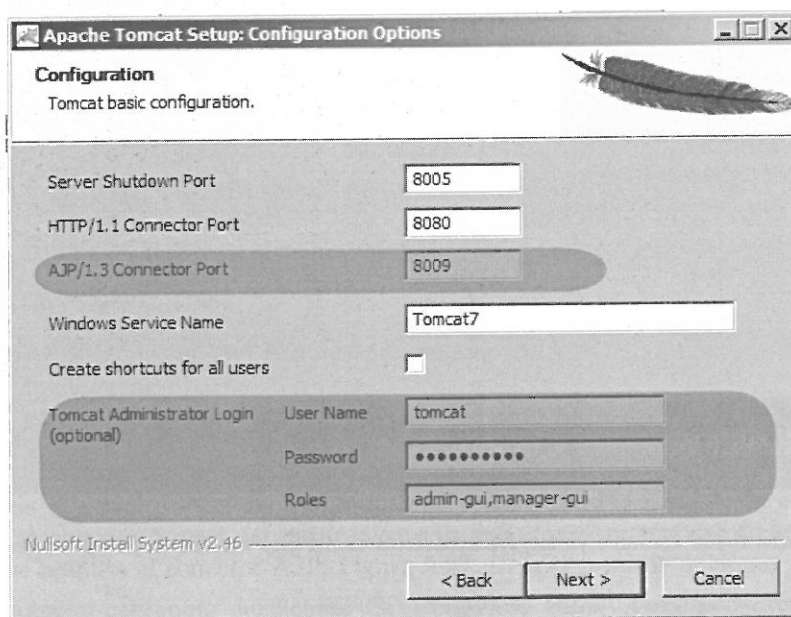


Figura 9.22: Instalación y configuración de *Tomcat* en *ServidorWindowsXX*

- a. Accede a **Inicio**, **Panel de control**, **Seguridad**, **Dejar pasar un programa a través de firewall de Windows**.
  - b. Pincha en **Agregar Programa**.
  - c. Pincha en **Examina** y selecciona **C:\Program Files\Apache Software Foundation\Tomcat7.0\bin\tomcat7.exe**.
  - d. Acepta los cambios.
- 3.8. Despliega la aplicación **Calendar**.
- 3.9. Verifica que puedes acceder a ella desde **DesarrolloW7XX**.
4. Prueba la configuración.
- 4.1. Desde **DesarrolloW7XX** abre el navegador y establece una conexión a **http://192.168.1.X7/Calendar/** o **http://servidorlinuxXX.dawXX.net/Calendar/**.
  - 4.2. Prueba a parar algunos de los servidores *Tomcat* y verifica que se puede acceder a la aplicación.
  - 4.3. Prueba a iniciar el servidor que has parado previamente y para el otro. Verifica que se puede acceder a la aplicación.
  - 4.4. Consulta los *logs* de acceso de los servidores **Tomcat** y observa que los dos están respondiendo.
5. Estadísticas de balanceo de carga.
- 5.1. Inicia sesión en **ServidorLinuxXX** con un usuario con privilegios de administración.
  - 5.2. Edita el fichero **/etc/libapache2-mod-jk/workers-properties** y realiza la siguiente configuración, Figura 9.23.

```
#
#----- worker list -----
#
#
# The workers that your plugins should create and work with
#
worker.list=loadbalancer,stat
#
#----- ajp13_worker WORKER DEFINITION -----
#
#
worker.stat.type=status
#
```

Figura 9.23: Fichero **/etc/libapache2-mod-jk/workers-properties**

- 5.3. Reinicia *Apache*.
- 5.4. Edita el fichero **/etc/apache2/sites-available/000-default** y añade las siguientes directivas para redirigir las peticiones al *worker* que nos permite ver las estadísticas Figura 9.24.
- 5.5. Reinicia *Apache*.
- 5.6. Desde **DesarrolloW7XX** abre el navegador y establece una conexión a **http://192.168.1.X7/status/**, Figura 9.25.

```

VirtualHost *:80>
    ServerAdmin webmaster@localhost
    DocumentRoot /var/www

    ProxyPreserveHost on
    ProxyPass /sall'ple http://localhost:8080/sall'ple
    ProxyPassReverse /sall'ple http://localhost:8080/sall'ple
    ProxyPass /curso http://localhost:8080/curso
    ProxyPassReverse /curso http://localhost:8080/curso
    ProxyPass /coll'ipras http://localhost:8080/coll'ipras
    ProxyPassReverse /coll'ipras http://localhost:8080/coll'ipras

    JkMount /Calendar loadbalancer
    JkMount /Calendar/* loadbalancer
    JkMount /Status stat

```

Figura 9.24: Fichero /etc/apache2/sites-availables/000-default

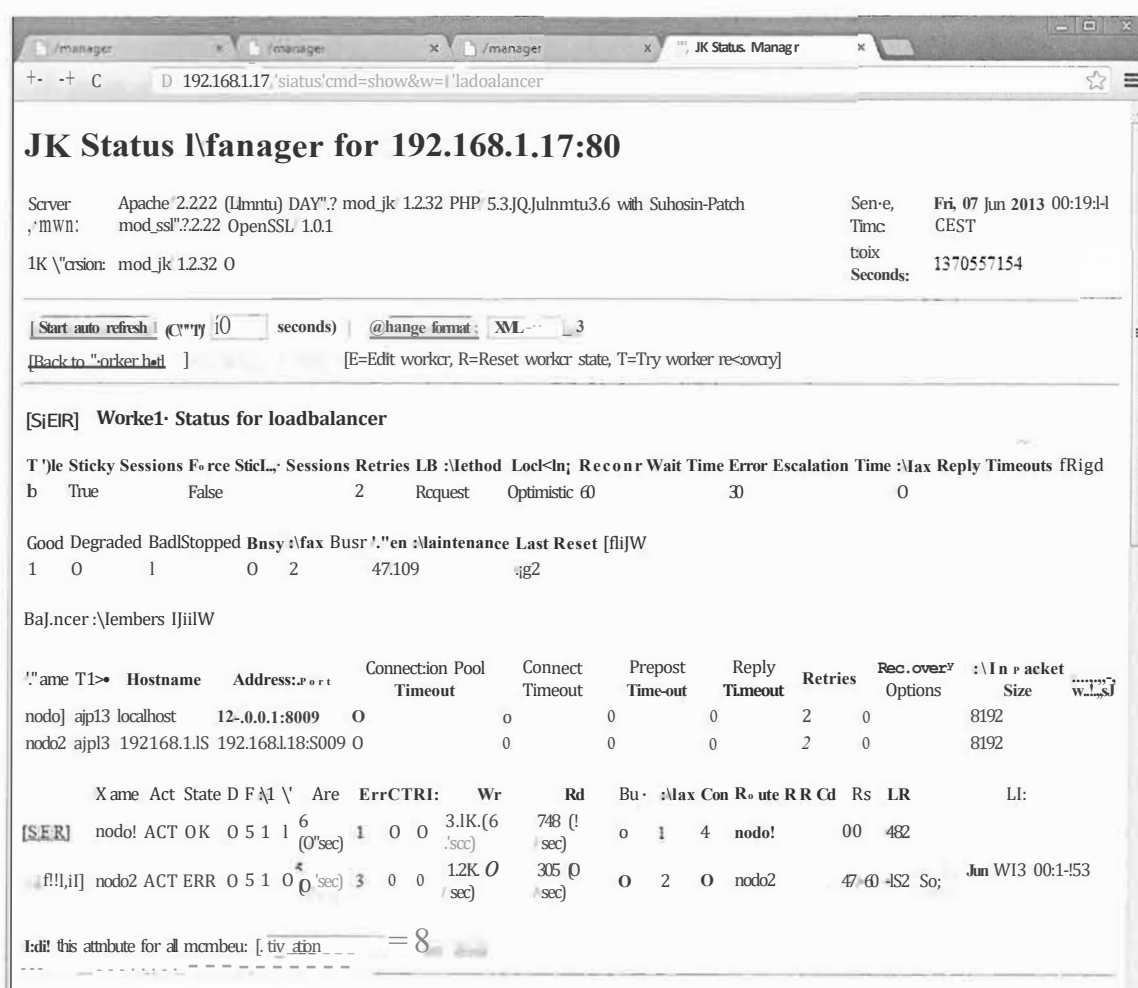


Figura 9.25: Estadísticas de balanceo de carga