

DESPLIEGUE DE APLICACIONES WEB - TAREA 3

Jonatan de la Torre Ruzafa

Diciembre 2022

- 1. Crea un archivo WAR con una aplicación para Tomcat sencilla, donde en la página principal al menos aparezca tu nombre. Despliega el WAR y comprueba que funciona la aplicación....3**
- 2. Configura Apache para se convierta en proxy de Tomcat (solo en necesario por el puerto 80). Crea un Host Virtual de Tomcat y haz que funcione en la URL <http://proxytomcat.es>.....5**
- 3. Crea un Host Virtual Seguro en Tomcat en la URL <https://tomcatseguro.es> (puerto 443) y crea alguna aplicación sobre él.....8**

1. Crea un archivo WAR con una aplicación para Tomcat sencilla, donde en la página principal al menos aparezca tu nombre. Despliega el WAR y comprueba que funciona la aplicación.

Para crear el archivo WAR primero necesitamos el árbol de directorios que creamos con:

```
jonatan@jonatan-VirtualBox:~/Escritorio$ sudo mkdir warjonatan
[sudo] contraseña para jonatan:
jonatan@jonatan-VirtualBox:~/Escritorio$ sudo mkdir warjonatan/WEB-INF
jonatan@jonatan-VirtualBox:~/Escritorio$ sudo mkdir warjonatan/WEB-INF/classes
jonatan@jonatan-VirtualBox:~/Escritorio$ sudo mkdir warjonatan/WEB-INF/lib
jonatan@jonatan-VirtualBox:~/Escritorio$ sudo gedit warjonatan/index.jsp
```

Creamos un index.jsp en la raíz de la aplicación:



The screenshot shows a code editor window titled 'index.jsp' with the file path '/home/jonatan/Escritorio/warjonatan'. The editor contains the following HTML code:

```
1 <html>
2 <head>
3 <title>APLICACIÓN WEB WAR JONATAN</title>
4 </head>
5 <body>
6 <h1>APLICACIÓN HECHA POR JONATAN DE LA TORRE RUZAFa</h1>
7 </body>
8 </html>
```

Y un xml:



The screenshot shows a code editor window titled 'web.xml' with the file path '/home/jonatan/Escritorio/warjonatan/WEB-INF'. The editor contains the following XML code:

```
1 <?xml version="1.0" encoding="ISO-8859-1"?>
2 <web-app xmlns="http://java.sun.com/xml/ns/j2ee"
3 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4 xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
5 http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd" version="2.5">
6 <display-name>Descriptor Aplicacion warjonatan</display-name>
7 <description>
8 Descriptor hecho por Jonatan de la Torre Ruzafa
9 </description>
10 </web-app>
```

Creamos el fichero WAR

```
jonatan@jonatan-VirtualBox:~/Escritorio/warjonatan$ sudo jar cvf warjonatan.war .
manifiesto agregado
agregando: WEB-INF/(entrada = 0) (salida = 0)(almacenado 0%)
agregando: WEB-INF/lib/(entrada = 0) (salida = 0)(almacenado 0%)
agregando: WEB-INF/web.xml(entrada = 417) (salida = 248)(desinflado 40%)
agregando: WEB-INF/classes/(entrada = 0) (salida = 0)(almacenado 0%)
agregando: index.jsp(entrada = 146) (salida = 106)(desinflado 27%)
```

Lo desplegamos con el manager de Tomcat:

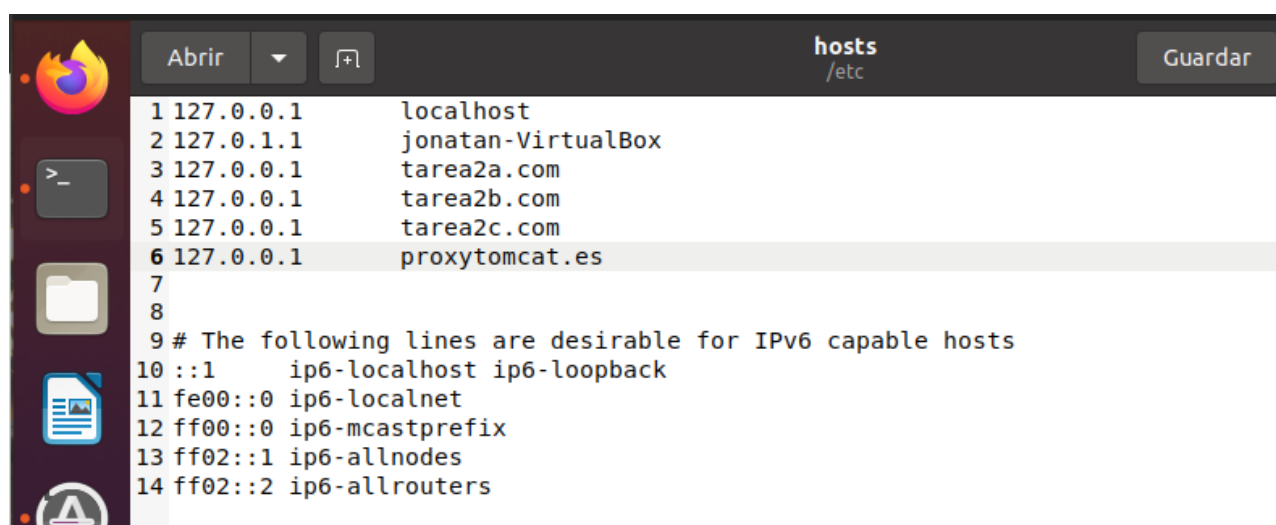
Desplegar	
Desplegar directorio o archivo WAR localizado en servidor	
Trayectoria de Contexto (opcional):	<input type="text"/>
Version (for parallel deployment):	<input type="text"/>
URL de archivo de Configuración XML:	<input type="text"/>
URL de WAR o Directorio:	<input type="text"/>
<input type="button" value="Desplegar"/>	
Archivo WAR a desplegar	
Seleccione archivo WAR a cargar	<input type="button" value="Browse..."/> warjonatan.war
<input type="button" value="Desplegar"/>	

Y ya podemos acceder a él:



2. Configura Apache para se convierta en proxy de Tomcat (solo en necesario por el puerto 80). Crea un Host Virtual de Tomcat y haz que funcione en la URL <http://proxytomcat.es>.

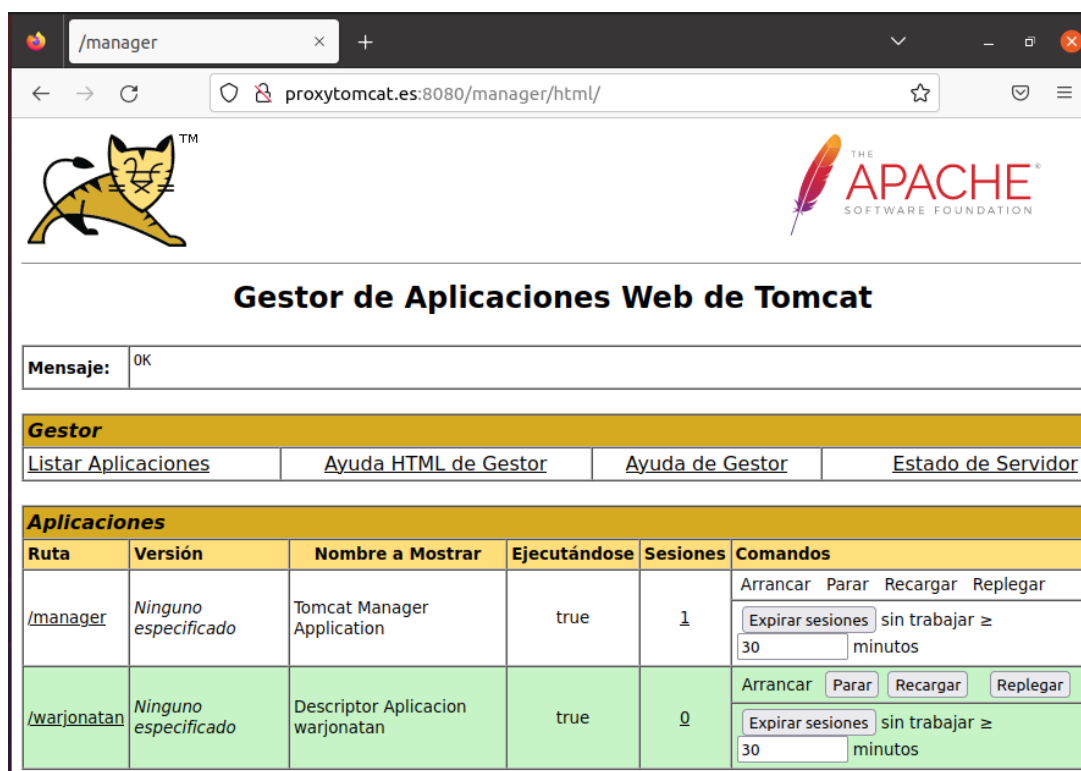
Añadimos la dirección al fichero hosts ya que no tenemos DNS:



Creamos el directorio de despliegue en webapps-proxytomcat y modificamos el fichero server.xml:

```
<Host name="proxytomcat.es" appBase="webapps-proxytomcat"
      unpackWARs="true" autoDeploy="true">
  <Valve className="org.apache.catalina.valves.AccessLogValve"
        directory="logs" prefix="proxytomcat_access_log." suffix=".txt"
        pattern="%h %l %u %t &quot;%r&quot; %s %b" />
</Host>
```

Reiniciamos tomcat y copiamos el fichero manager.xml para tenerlo en el nuevo host virtual. Copiamos nuestra aplicación WAR hecha en el apartado 1 para desplegarla.



Una vez esto habilitamos los módulos proxy en Apache con `a2enmod proxy_http` y `a2enmod proxy` y creamos un Host Virtual en Apache que nos hará de proxy y lo configuramos:

```

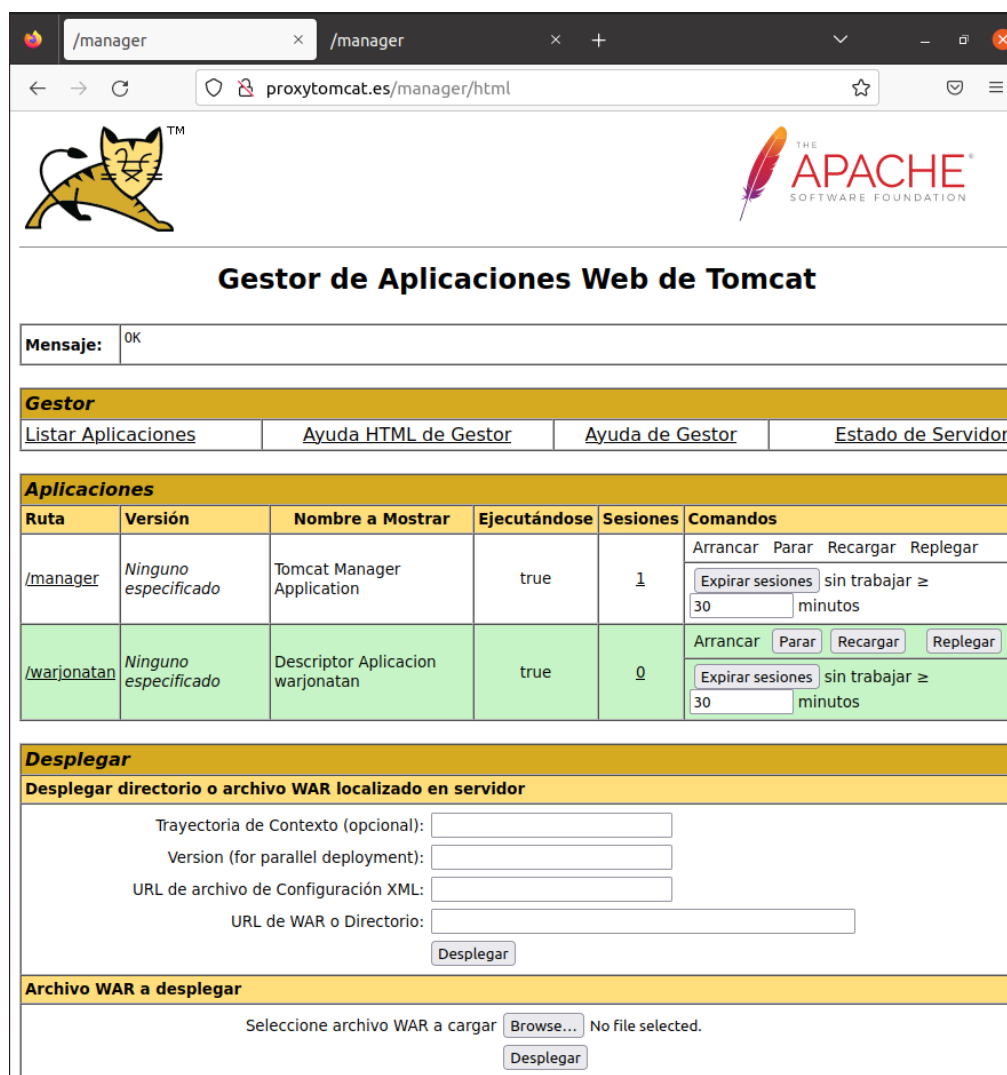
1 VirtualHost *:80
2 # The ServerName directive sets the request scheme, hostname and port that
3 # the server uses to identify itself. This is used when creating
4 # redirection URLs. In the context of virtual hosts, the ServerName
5 # specifies what hostname must appear in the request's Host: header to
6 # match this virtual host. For the default virtual host (this file) this
7 # value is not decisive as it is used as a last resort host regardless.
8 # However, you must set it for any further virtual host explicitly.
9 ServerName proxytomcat.es
10
11 ServerAdmin webmaster@localhost
12 DocumentRoot /var/www/proxytomcat.es/html
13
14 # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
15 # error, crit, alert, emerg.
16 # It is also possible to configure the loglevel for particular
17 # modules, e.g.
18 #LogLevel info ssl:warn
19
20 ErrorLog ${APACHE_LOG_DIR}/error.log
21 CustomLog ${APACHE_LOG_DIR}/access.log combined
22
23 # For most configuration files from conf-available/, which are
24 # enabled or disabled at a global level, it is possible to
25 # include a line for only one particular virtual host. For example the
26 # following line enables the CGI configuration for this host only
27 # after it has been globally disabled with "a2disconf".
28 #Include conf-available/serve-cgi-bin.conf
29
30 ProxyPass / http://proxytomcat.es:8080/
31 ProxyPassReverse / http://proxytomcat.es:8080/
32
33 <Location />
34     Order Allow,Deny
35     Allow from all
36 </Location>
37 </VirtualHost>
38
39 # vim: syntax=apache ts=4 sw=4 sts=4 sr noet

```

También necesitamos modificar el conector en server.xml de Tomcat:

```
<Connector port="8080" protocol="HTTP/1.1"
           connectionTimeout="20000"
           redirectPort="8443"
           proxyPort = "80" />
```

Reiniciamos ambos servicios y comprobamos:



The screenshot shows the Tomcat Manager web interface in a browser. The page title is "Gestor de Aplicaciones Web de Tomcat". It features the Tomcat logo and the Apache Software Foundation logo. Below the title, there is a message box showing "Mensaje: OK". The main content area is divided into two sections: "Gestor" and "Desplegar".

Gestor

[Listar Aplicaciones](#) | [Ayuda HTML de Gestor](#) | [Ayuda de Gestor](#) | [Estado de Servidor](#)

Aplicaciones

Ruta	Versión	Nombre a Mostrar	Ejecutándose	Sesiones	Comandos
/manager	Ninguno especificado	Tomcat Manager Application	true	1	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos
/warjonatan	Ninguno especificado	Descriptor Aplicacion warjonatan	true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos

Desplegar

Desplegar directorio o archivo WAR localizado en servidor

Trayectoria de Contexto (opcional):

Version (for parallel deployment):

URL de archivo de Configuración XML:

URL de WAR o Directorio:

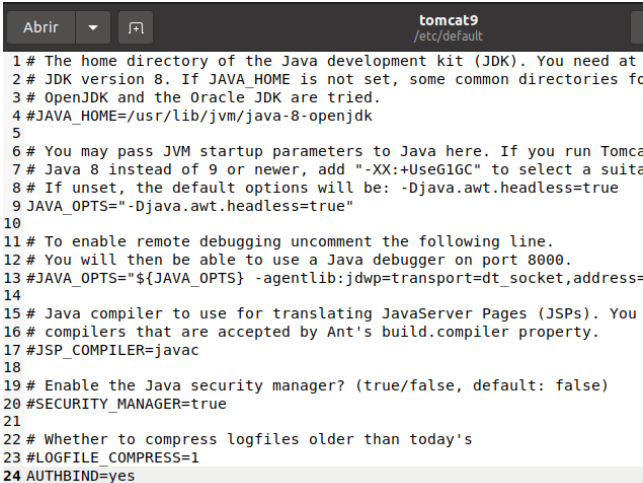
Archivo WAR a desplegar

Seleccione archivo WAR a cargar No file selected.

3. Crea un Host Virtual Seguro en Tomcat en la URL <https://tomcatseguro.es> (puerto 443) y crea alguna aplicación sobre él.

La parte de crear un host virtual en Tomcat ya la hemos visto en el apartado anterior por lo que la tendremos que volver a hacer pero aquí no queda reflejado.

Primero paramos Apache con `sudo service apache2 stop`, instalamos `authbind` y editamos el fichero `/etc/default/tomcat9` y añadimos la línea `AUTHBIND=yes`



```
tomcat9
/etc/default

1 # The home directory of the Java development kit (JDK). You need at
2 # JDK version 8. If JAVA_HOME is not set, some common directories fo
3 # OpenJDK and the Oracle JDK are tried.
4 #JAVA_HOME=/usr/lib/jvm/java-8-openjdk
5
6 # You may pass JVM startup parameters to Java here. If you run Tomca
7 # Java 8 instead of 9 or newer, add "-XX:+UseG1GC" to select a suita
8 # If unset, the default options will be: -Djava.awt.headless=true
9 JAVA_OPTS="-Djava.awt.headless=true"
10
11 # To enable remote debugging uncomment the following line.
12 # You will then be able to use a Java debugger on port 8000.
13 #JAVA_OPTS="${JAVA_OPTS} -agentlib:jdwp=transport=dt_socket,address=
14
15 # Java compiler to use for translating JavaServer Pages (JSPs). You
16 # compilers that are accepted by Ant's build.compiler property.
17 #JSP_COMPILER=javac
18
19 # Enable the Java security manager? (true/false, default: false)
20 #SECURITY_MANAGER=true
21
22 # Whether to compress logfiles older than today's
23 #LOGFILE_COMPRESS=1
24 AUTHBIND=yes
```

Creamos un archivo 443 con:

- `sudo touch /etc/authbind/byport/443`
- `sudo chmod 500 /etc/authbind/byport/443`
- `sudo chown tomcat /etc/authbind/byport/443`

Creamos un almacén de claves y una clave con sudo keytool -genkey -alias miclave -keyalg RSA -keystore /var/lib/tomcat9/mialmacen

Modificamos server.xml y añadimos un Connector:



```
79  -->
80  <Connector port="443" protocol="HTTP/1.1" SSLEnabled="true"
81             maxThreads="150" scheme="https" secure="true"
82             clientAuth="false" sslProtocol="TLS"
83             keystoreFile="/var/lib/tomcat9/mialmacen" keystorePass="123456"
84             keyAlias="miclave" keyPass="123456" />
```

Para desplegar una aplicación utilizamos el mismo procedimiento que en el anterior apartado copiar el WAR:

