Grupo Grace Hopper 19/05/2018

Certificado SSL para Tomcat 9.0

Pagina www.sslforfree.com, se introduce dominio -> Manual verification -> Descargar fichero. El fichero se pone donde dice y se da a generar. Se descargan los ficheros y se ponen en una carpeta, por ejemplo certs dentro de apache. Los ficheros son la clave privada y los dos certificados (certificate es el del dominio y ca_bundle el de la entidad certificadora y root). Los certificados tienen validez de 90 días, después es necesario renovarlo.

Generar certificado uniendo los obtenidos con:
openssl pkcs12 -export -out certificate.pfx -inkey private.key -in certificate.crt -certfile ca_bundle.crt
Poner contraseña gracehopper

Crear el almácen de claves que usará tomcat para los certificados (keystore.jsk):
keytool -importkeystore -srckeystore certificate.pfx -srcstorepass gracehopper -srcstoretype pkcs12 -destkeystore keystore.jks -deststoretype jks -deststorepass gracehopper

En keystore.jks esta el certificado a nombre de mewat1718.ddns.net producido por Let's Encription que a su vez tiene un certificado que lo ha producido por una autoridad Root (Digital Signature)

En conf/server.xml se pone el conector:

```
    <Connector port="443" maxThreads="150"
            protocol="org.apache.coyote.http11.Http11NioProtocol"
            sslImplementation="org.apache.tomcat.util.net.jsse.JSSEImplemntation"
            scheme="https" secure="true" SSLEnabled="true">
      <SSLHostConfig clientAuth="false">
            <Certificate certificateKeystoreFile="certs/keystore.jks"
                  certificateKeystorePassword="gracehopper" type="RSA"
                  certificateVerification="optionalNoCA"
                   certificateKeyAlias="1"/>
      </SSLHostConfig>
    </Connector>
```

Y se establece el puerto 80 para no SSL con redirectPort a 443.

Para que las conexiones al puerto 80 se dirigan al 443 (de http a https) poner en conf/web.xml:

```
<security-constraint>
 <web-resource-collection>
 <web-resource-name>HTTPSOnly</web-resource-name>
 <url-pattern>/*</url-pattern>
 </web-resource-collection>
 <!-- auth-constraint goes here if you requre authentication -->
 <user-data-constraint>
 <transport-guarantee>CONFIDENTIAL</transport-guarantee>
 </user-data-constraint>
</security-constraint>
```

****En internet se supone que deberia funcionar pero de momento no va

Puede ser necesario abrir puerto 443 en el router. Con openssl s_client -debug -connect mewat1718.ddns.net:443 se puede testear.

Para revisar posibles errores en la configuración de tomcat, mirar log del día correspondiente:
sudo cat /usr/local/apache-tomcat-9.0.7/logs/catalina.2018-05-19.log

Para arrancar tomcat es necesario usar privilegios de root, ya que puertos < 1024 no se pueden enlazar a servicios sin permisos de root:
sudo /usr/local/apache-tomcat-9.0.7/bin/startup.sh

Página https://www.ssllabs.com/ssltest/analyze.html?d=mewat1718.ddns.net&latest para verificar TLS en la web.

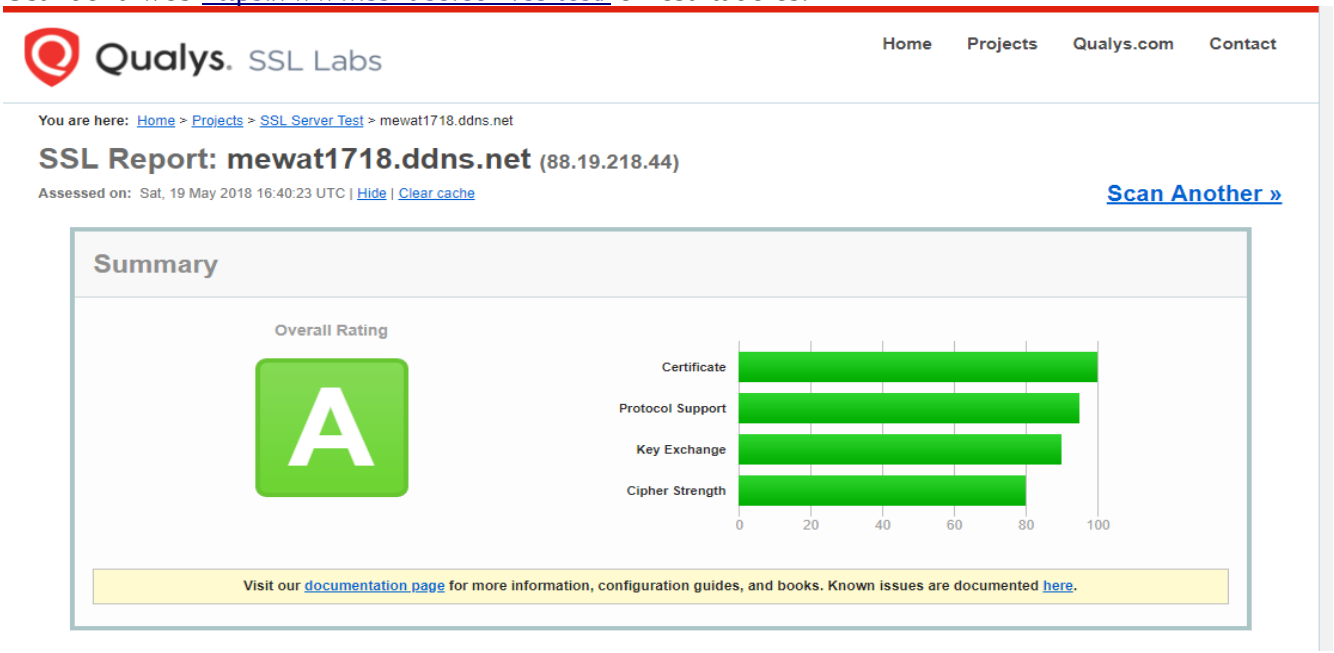
Iniciar tomcat como root no es adecuado por temas de seguridad, otra opción es mantener los puertos 8080 y 8443 de tomcat y mediante iptables hacer un forwarding:
sudo `iptables -t nat -A PREROUTING -p tcp --dport 80 -j REDIRECT --to-port 8080`
sudo `iptables -t nat -A PREROUTING -p tcp --dport 443 -j REDIRECT --to-port 8443`
sudo iptables -t nat –list (para ver las reglas actuales)
Para eliminar una regla usar -D en vez de -A.

Para que funcione en local, como desarrollador:
sudo iptables -t nat -A OUTPUT -o lo -p tcp --dport 80 -j REDIRECT --to-port 8080

De momento no se usa porque da problemas en la conexión con el servidor (psweb no puede conectar con ps)

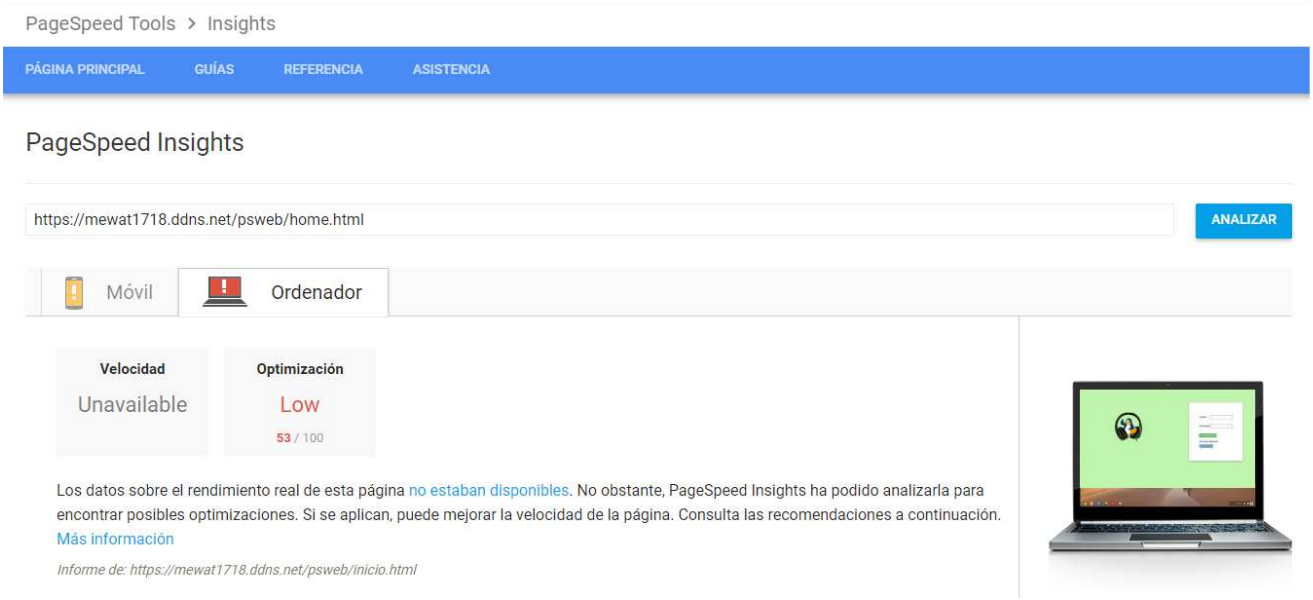
Seguridad del servidor web

Usando la web https://www.ssllabs.com/ssltest/ el resultado es:



Optimización

A continuación se va a analizar la página web de inicio.html usando la web https://developers.google.com/speed/pagespeed/insights/?hl=es cuyo resultado es:



Como se puede observar la web no está optimizada por lo que se van a llevar a cabo una serie de optimizaciones.

Primero se minimizan todos los ficheros css y js para reducir el tamaño de bytes transferidos. Con esta mejora la puntuación se incrementa hasta 63.

Luego se lleva a cabo una compresion de todos los ficheros que el servidor envía a los cliente de manera que se envien en formato gzip. Para ello es necesario modificar serverl.xml añadiendo al conector de 443 las líneas:

```
compression="force" compressionMinSize="0"
        useSendfile="false"
        compressableMimeType="text/html,text/xml,text/css"
```

Asi obliga a comprimir todo lo que envia el servidor a los clientes (con “force” es todo asi que “compressableMimeType” que selecciona los tipos de ficheros no tiene efecto)

La mejora es de 63 a 74.

Por último, se optimizan las imágenes usando un compresor online (https://tinypng.com/) de manera que se reduce el tamaño de todas las imágenes con formato .png (que son las únicas que tiene la web) sin pérdida de datos.

Finalmente la puntuación obtenida es:

Por tanto se ha optimizado adecuadamente la página.