Instalación SSL Jboss / Wildfly

Requisitos:

Archivo keystore con extensión jks que incluya la llave privada, el certificado, la raíz y el intermediario.

Este proceso es independiente del SO. Se copia el certificado jks en una ruta cualquiera con permisos de lectura. Luego se ubica la ruta del jboss o wildfly /standalone/configuration /standalone.xml (de usar otro archivo de configuración, aplicar los cambios sobre ese archivo)

```
[root@localhost configuration]# pwd
/opt/jboss-eap-6.1/standalone/configuration
[root@localhost configuration]# ls
application-roles.properties  mgmt-users.properties   standalone-ha.xml
application-users.properties  standalone-full-ha.xml  standalone-osgi.xml
logging.properties            standalone-full.xml     standalone.xml
[root@localhost configuration]#
```

Revisamos la primera etiqueta server: <server xmlns="urn:jboss:domain:1.4"> para ver la versión del archivo de configuración, en el ejemplo se tiene la versión 1.4. Según la versión se realiza uno de los siguientes casos.

```
<?xml version='1.0' encoding='UTF-8'?>

<server xmlns="urn:jboss:domain:1.4">
```

Caso 1: Versiones 1.0 a 1.9

1. Ubicamos la línea[1]: <connector name="http" protocol="HTTP/1.1" scheme="http" socket-binding="http"/>
   [1] En caso su archivo de configuración contenga varias veces la línea indicada, realizar el siguiente paso para cada una de ellas.

2. Debajo de ella, agregamos lo siguiente: <connector name="https" protocol="HTTP/1.1" scheme="https" socket-binding="https" enable-lookups="false" secure="true">
   <ssl name="jboss" password="123456" protocol="TLS" key-alias="server" certificate-key-file="/opt/ssl/keystore.jks"/>
   </connector>

   Donde:

ssl name	:	Un nombre identificador, no es relevante.
password	:	Contraseña del certificado.
protocol	:	TLS por defecto[2]
key-alias	:	Alias del certificado.
certificate-key-file	:	Ruta del certificado

   [2] En caso esté utilizando Jboss EAP 6.4, o al aplicar los cambios e intentar acceder obtenga el error SSL_CIPHER_MISMATCH, cambiar TLS por TLSv1.2

```
            <connector name="http" protocol="HTTP/1.1" scheme="http" socket-bind
ing="http"/>
            <connector name="https" protocol="HTTP/1.1" scheme="https" socket-bi
nding="https" enable-lookups="false" secure="true">
                <ssl name="jboss6-2" password="123456" protocol="TLS" key-alias=
"server" certificate-key-file="/opt/ssl/keystore.jks"/>
            </connector>
```

3. Reiniciar el jboss/wildfly y ya está el SSL activo.

Caso 2: Versiones 2.0 a 17.0

1. Ubicamos el security-realm llamado ApplicationRealm

```
        <security-realm name="ApplicationRealm">
            <server-identities>
                <ssl>
                    <keystore path="application.keystore" relative-to="jboss
.server.config.dir" keystore-password="password" alias="server" key-password="pa
ssword" generate-self-signed-certificate-host="localhost"/>
                </ssl>
            </server-identities>
            <authentication>
                <local default-user="$local" allowed-users="*" skip-group-lo
ading="true"/>
                <properties path="application-users.properties" relative-to=
"jboss.server.config.dir"/>
            </authentication>
            <authorization>
                <properties path="application-roles.properties" relative-to=
"jboss.server.config.dir"/>
            </authorization>
        </security-realm>
```

2. Debajo de este, creamos un security Realm llamado SSLRealm de la siguiente forma:

```
<security-realm name="SSLRealm">
        <server-identities>
            <ssl>
                <keystore path="/opt/ssl/keystore.jks" keystore-password="123456" />
            </ssl>
        </server-identities>
    </security-realm>
```

Donde:

path	: Ruta del certificado.
keystore-password	: Contraseña del certificado.
key-alias	: Alias del certificado (OPCIONAL)

```
            <security-realm name="SSLRealm">
            <server-identities>
                    <ssl>
                            <keystore path="/opt/keystore.jks" keystore-pass
word="123456" />
                    </ssl>
            </server-identities>
        </security-realm>
```

3. Buscamos la línea: <http-listener name="default" socket-binding="http" redirect-socket="https" enable-http2="true"/>
4. Debajo de ella, agregamos lo siguiente: <https-listener name="https" socket-binding="https" security-realm="SSLRealm" enable-http2="true"/>

Nota 1: En caso ya exista un https-listener, editarlo para que quede como el indicado.
Nota 2: En caso en el http-listener no exista la etiqueta enable-http2="true", quitarla también del https-listener.

```
                <http-listener name="default" socket-binding="http" redirect-soc
ket="https" enable-http2="true"/>
                <https-listener name="https" socket-binding="https" security-rea
lm="SSLRealm" enable-http2="true"/>
```

5. Reiniciar el jboss/wildfly y ya está el SSL activo.

Caso 3: Versiones 18.0 +

1. Ubicamos la sección tls

```
<tls>
    <key-stores>
        <key-store name="applicationKS">
            <credential-reference clear-text="password"/>
            <implementation type="JKS"/>
            <file path="application.keystore" relative-to="jboss.server.config.dir"/>
        </key-store>
    </key-stores>
    <key-managers>
        <key-manager name="applicationKM" key-store="applicationKS" generate-self-signed-certificate-host=
        "localhost">
            <credential-reference clear-text="password"/>
        </key-manager>
    </key-managers>
    <server-ssl-contexts>
        <server-ssl-context name="applicationSSC" key-manager="applicationKM"/>
    </server-ssl-contexts>
</tls>
```

2. En las dos líneas **<credential-reference clear-text="password"/>** cambiar password por la contraseña de nuestro keystore.

3. En la línea **<file path="application.keystore" relative-to="jboss.server.config.dir"/>**, cambiar application.keystore por la ruta completa de nuestro keystore. Eliminar relative-to="jboss.server.config.dir".

4. En la línea **<key-manager name="applicationKM" key-store="applicationKS" generate-self-signed-certificate-host="localhost">**, eliminar generate-self-signed-certificate-host="localhost".

   Hechos estos cambios, deberá quedar así:

```
<tls>
    <key-stores>
        <key-store name="applicationKS">
            <credential-reference clear-text="Contraseña123456"/>
            <implementation type="JKS"/>
            <file path="/opt/ssl/keystore.jks"/>
        </key-store>
    </key-stores>
    <key-managers>
        <key-manager name="applicationKM" key-store="applicationKS" >
            <credential-reference clear-text="Contraseña123456"/>
        </key-manager>
    </key-managers>
    <server-ssl-contexts>
        <server-ssl-context name="applicationSSC" key-manager="applicationKM"/>
    </server-ssl-contexts>
</tls>
```

5. Reiniciar el jboss/wildfly y ya está el SSL activo.