Since I have found this question due to another mistake I write here some possible causes of problems in similar situations.

**Wrong certificate name**

This is the case of this question. I Chrome I had the following message:

NET::ERR\_CERT\_AUTHORITY\_INVALID

You have to set **CN=localhost** to get it working.

What is your first and last name?

[Unknown]: localhost

This is specified also in GlassFish Security Guide:

For HTTPS hostname verification, it is important to ensure that the name of the certificate (CN) matches the fully-qualified hostname of your site (fully-qualified domain name). If the names do not match, clients connecting to the server will see a security alert stating that the name of the certificate does not match the name of the site.

**Wrong key algorithm**

I used the keytool -genkey command without specifying the -keyalg option and this created a certificate with SHA1withDSA.

Chrome said ERR\_CONNECTION\_CLOSED and in my server log I found

javax.net.ssl.SSLHandshakeException: no cipher suites in common

I solved this specifying **keytool -genkey -keyalg RSA**

**Change also glassfish-instance certificate**

From GlassFish Security Guide:

DAS uses the s1as alias for SSL/TLS authentication and the instances use the glassfish-instance alias

According to a comment on [another answer](https://stackoverflow.com/a/29280729/771431) "If you change the s1as certificate, you will also need to change the glassfish-instance certificate".

**Wrong keystore file**

Remember that:

* The keystore.jks file contains GlassFish Server certificate, including its **private key**.
* The cacerts.jks file contains the GlassFish Server trusted certificates, including **public keys** for other entities.

Sometimes the distraction can made you put the public key in the wrong file. This should be the correct sequence:

# Generate a key pair in keystore.jks

keytool -genkeypair -alias s1as -keystore keystore.jks -keypass changeit -storepass changeit -keyalg RSA

# Export the certificate

keytool -export -keystore keystore.jks -alias s1as -file s1as.cer -storepass changeit

# Import it into the truststore.jks

keytool -import -noprompt -trustcacerts -file s1as.cer -alias s1as -keystore cacerts.jks -storepass changeit

**How to check**

This command shows information about certificates:

keytool -v -list -alias <the\_alias> -keystore <filename>.jks

If you take a look to the original self-signed certificates provided by GlassFish you have:

* **CN**:
  + Owner: CN=localhost for s1as
  + Owner: CN=localhost-instance for glassfish-instance
* **RSA**: Signature algorithm name: SHA256withRSA
* **Entry type**:
  + Entry type: PrivateKeyEntry for keystore.jks
  + Entry type: trustedCertEntry for cacerts.jks