

Using Aspire via HTTPs



To use aspire via https you'll need to import a certificate into aspire - see here [Configuring a Certificate Store](#)

Configuring Felix Properties

The config/felix.properties file should be edited to enable HTTPS.

If we leave this property in the configuration, it will still work with HTTPS enabled. So, <http://localhost:50505/aspire/> will access the admin interface with HTTP (see notes below).

```
org.osgi.service.http.port=50505
```

To enable HTTPS, we must set the following values:

```
org.apache.felix.https.enable=true
org.osgi.service.http.port.secure=50443
```

The default secure port is 443, and if you use that port then you do not need to put the port number in the URL. However it seems like further configuration is required to avoid the browser "SSL connection error".

Optionally, we could disable unsecured HTTP traffic with:

```
org.apache.felix.http.enable=false
```

And specify the keystore file and passwords to access it:

```
org.apache.felix.https.keystore=config/myKeystore <<<< NOTE UNIX STYLE PATH SEPARATOR IS REQUIRED EVEN ON
WINDOWS, THIS PATH IS RELATIVE TO ASPIRE_HOME BY DEFAULT
org.apache.felix.https.keystore.password=myKeystorePassword
org.apache.felix.https.keystore.key.password=myKeyPassword
```

Hint: In the case that we are working with a customer's certificate the following change must be applied

```
org.osgi.service.http.port=50505
org.apache.felix.http.enable=false
org.apache.felix.https.enable=true
org.osgi.service.http.port.secure=50443
org.apache.felix.https.keystore=config/Certificates/systemWideKeystore.jks
org.apache.felix.https.keystore.password=XXX
org.apache.felix.https.truststore=config/Certificates/clientCertKeystore.jks
org.apache.felix.https.truststore.password=XXX
```

Finally, Aspire can be started and the admin page can now be accessed using HTTPS: <https://my-pc.search.local:50443/aspire>

Access to other components should be through HTTPS as well, such as the HTTP Feeder:

```
https://my-pc.search.local:50443/submitFiles?test1=hello&test2=world
```

```
<doc>
  <feederLabel>HttpFeeder</feederLabel>
  <test1 source="HttpFeeder">hello</test1>
  <test2 source="HttpFeeder">world</test2>
</doc>
```



- In some instances (possibly all) if you don't turn *http* off when you turn *https* on, requests to the *https* port will hang. It's probably best to turn *http* off when you turn *https* on
- The URL must match the server host entered into the certificate. In other words, it should match the "Issuer" name in the certificate. If accessed through another such as <https://localhost:50443/aspire> it will still work but you will get the error message: "Server's certificate does not match the URL".
- If the certificate has not been signed by a valid CA, the 'lock' icon will appear in red and will show a "Server's certificate is not trusted" message.