

React Application SSL Lab

In this lab, let us configure a react app running on a development server to serve requests using the protocol **https**!

We will need to

- Create and develop a new react project OR Use an existing react application
- Generate and Configure server key and ssl certificate
- Install the certificate to a local Certificate Authority
- Configure the **react-scripts start** to use the generated certificate and key

Use an Existing React Application

1. Open a command window, create a directory for this lab.

```
mkdir react-labs
```

```
cd react-labs
```

2. Clone the sample react application using the following git command.

```
git clone https://github.com/foxwas/was-react-apps.git
```

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19045.4170]
(c) Microsoft Corporation. All rights reserved.

C:\react-labs>git clone https://github.com/foxwas/was-react-apps.git
Cloning into 'was-react-apps'...
remote: Enumerating objects: 25, done.
remote: Counting objects: 100% (25/25), done.
remote: Compressing objects: 100% (25/25), done.
Receiving objects: 56% (14/25) 25 (delta 0), pack-reused 0
Receiving objects: 100% (25/25), 27.37 KiB | 4.56 MiB/s, done.

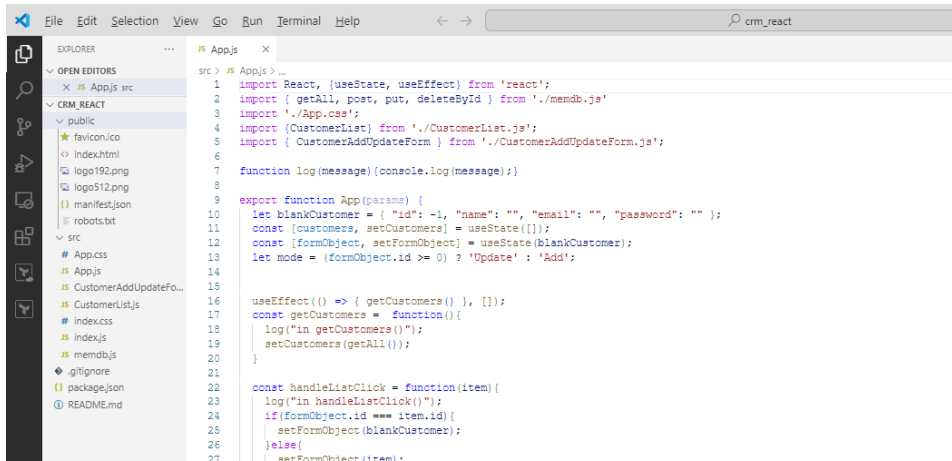
C:\react-labs>dir
Volume in drive C is Windows
Volume Serial Number is 1255-3BEE

Directory of C:\react-labs

04/04/2024 12:27 PM <DIR>      .
04/04/2024 12:27 PM <DIR>      ..
04/04/2024 12:27 PM <DIR>      was-react-apps
                0 File(s)      0 bytes
                3 Dir(s)  30,356,709,376 bytes free
```

3. In this project was-react-apps, let us use the react app: **crm_react**

Review this react app code.



4. In the command window type the following commands to change the directory to `crm_react` and then install application dependencies.

```
cd crm_react
```

```
npm install
```

```
C:\react-labs\was-react-apps>cd crm_react
C:\react-labs\was-react-apps\crm_react>npm install
```

5. Launch the react application on a development server using the following command

```
npm start
```

```
Windows PowerShell
Compiled successfully!

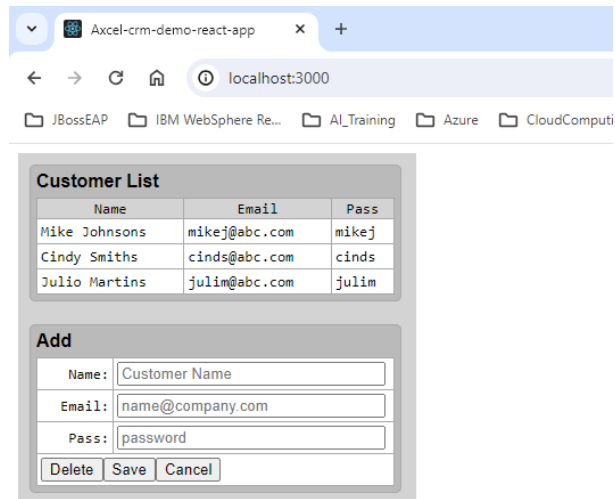
You can now view Axcel-crm-demo-react-app in the browser.

Local:      http://localhost:3000
On Your Network: http://192.168.1.76:3000

Note that the development build is not optimized.
To create a production build, use npm run build.

webpack compiled successfully
```

6. Open a browser, go to <http://localhost:3000>



7. You can view existing customers, update the customers data, and add a new customer. Test the application functional CRUD operations.

Generate and Configure server key and ssl certificate

8. Open another command window, change directory to C:\react-labs\was-react-apps\crm_react

cd C:\react-labs\was-react-apps\crm_react

Type the following command to verify the openssl version.

openssl version

```
C:\Windows\System32\cmd.exe

C:\>cd C:\react-labs\was-react-apps\crm_react

C:\react-labs\was-react-apps\crm_react>openssl version
OpenSSL 3.2.1 30 Jan 2024 (Library: OpenSSL 3.2.1 30 Jan 2024)

C:\react-labs\was-react-apps\crm_react>
```

9. Use openssl tool to Generate self-signed ssl certificate using the following command

```
openssl req -x509 -newkey rsa:2048 -nodes -sha256 -keyout server.key -out server.crt
```

Enter your data to complete the interactive shell prompts as shown in the following screenshot.

[illegible]

10. This tool will generate **server.key** and **server.crt**. Can you see the files in the current directory?

```

Directory of C:\react-labs\was-react-apps\crm_react

04/04/2024  01:02 PM  <DIR>          .
04/04/2024  01:02 PM  <DIR>          ..
04/04/2024  12:27 PM                333 .gitignore
04/04/2024  12:42 PM  <DIR>          node_modules
04/04/2024  12:40 PM                602,665 package-lock.json
04/04/2024  12:27 PM                865 package.json
04/04/2024  12:27 PM  <DIR>          public
04/04/2024  12:27 PM                3,429 README.md
04/04/2024  01:02 PM                1,348 server.crt
04/04/2024  01:02 PM                1,732 server.key
04/04/2024  12:27 PM  <DIR>          src
                   6 File(s)                610,372 bytes
                   5 Dir(s)  30,011,727,872 bytes free

C:\react-labs\was-react-apps\crm_react>

```

11. Use the following openssl command to view the contents in **server.crt**

```
openssl x509 -in server.crt -noout -text
```

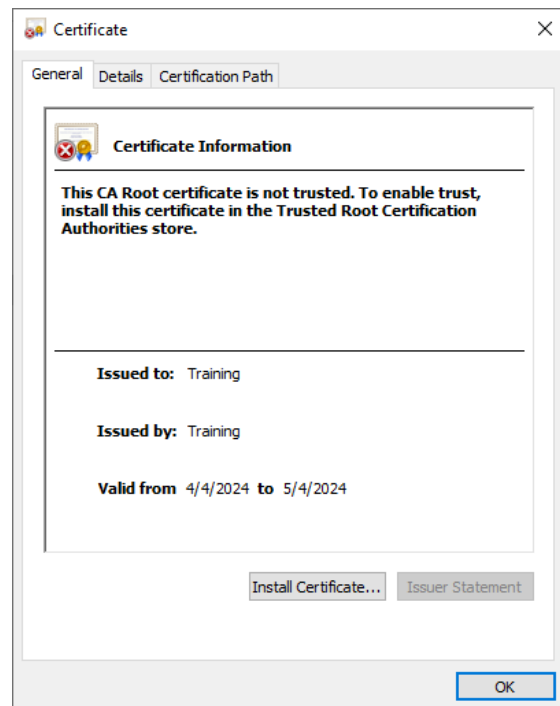
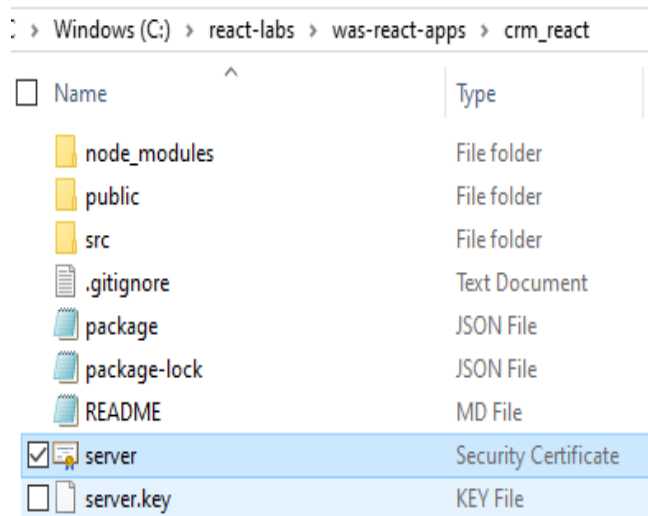
```

C:\Windows\System32\cmd.exe

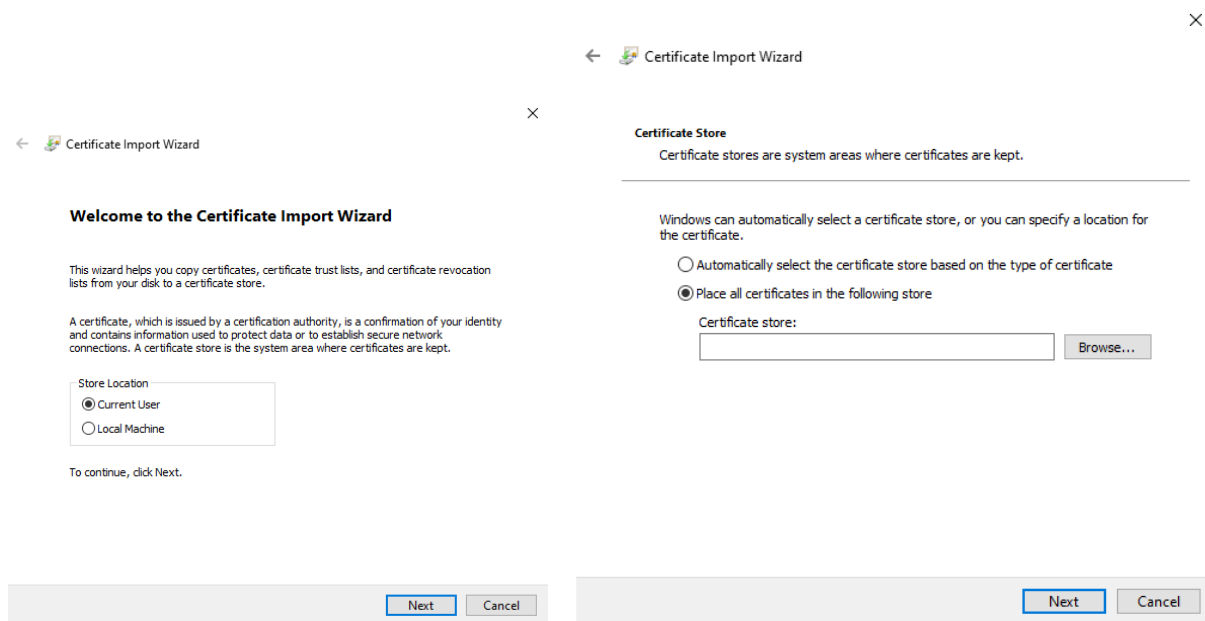
C:\react-labs\was-react-apps\crm_react>openssl x509 -in server.crt -noout -text
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            5e:80:1b:c6:e5:ce:89:da:3e:0f:5c:34:d8:9b:a7:2d:20:67:64:67
        Signature Algorithm: sha256WithRSAEncryption
        Issuer: C=US, ST=MI, L=Farmington Hills, O=WebAge Solutions, OU=Training
        Validity
            Not Before: Apr  4 17:02:50 2024 GMT
            Not After : May  4 17:02:50 2024 GMT
        Subject: C=US, ST=MI, L=Farmington Hills, O=WebAge Solutions, OU=Training
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
            Public-Key: (2048 bit)
            Modulus:
                00:ac:02:95:74:e9:fe:23:61:28:28:6d:95:41:ee:
                0a:92:3e:c2:84:fe:23:94:19:8d:86:e7:91:6f:9d:
                5b:2e:c8:b4:33:e4:b9:46:70:eb:5b:a3:e2:a2:ed:
                d1:b9:73:88:ae:f1:0d:76:0e:01:00:a0:77:4f:bc:
                7e:47:4e:10:42:d6:5c:ab:4e:59:ab:29:99:3c:c5:
                eb:24:4e:9d:52:db:4c:56:81:fe:5e:dc:91:63:f6:
                21:57:d5:93:ef:fe:b1:89:40:cc:66:af:2a:00:a6:
                c4:8b:f9:72:14:dc:cf:73:49:97:7f:71:ea:a5:6b:
                41:e4:b2:cf:9d:e4:f2:61:03:f9:ac:44:eb:49:70:
                20:2c:fe:1c:31:25:ae:6d:ee:39:3f:1e:d6:43:3f:
                b7:6c:bb:9c:16:41:dc:d6:31:f1:94:b8:72:6f:da:
                59:ed:85:9c:3d:2e:f2:c0:54:47:ff:b7:1d:58:d9:
                d3:e0:c1:a0:cf:ea:a3:22:dc:a1:7c:d3:86:27:4c:
                45:61:73:5d:b5:07:f6:b4:73:10:f8:db:09:e8:11:
                55:14:88:0c:09:2a:9a:68:05:31:2c:ea:5a:a2:0a:
                d6:e2:32:ac:2d:ce:4b:8a:31:d7:5b:85:a4:d2:13:
                77:c3:46:f1:53:7f:a7:eb:d3:9b:01:a0:88:73:b0:
                c4:c7
            Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Subject Key Identifier:
                8B:D8:4E:04:5F:29:73:FE:FC:6E:F2:A6:43:0A:CA:4B:7D:1F:51:10
            X509v3 Authority Key Identifier:
                8B:D8:4E:04:5F:29:73:FE:FC:6E:F2:A6:43:0A:CA:4B:7D:1F:51:10
            X509v3 Basic Constraints: critical
                CA:TRUE
        Signature Algorithm: sha256WithRSAEncryption
        Signature Value:
            aa:9f:d3:9a:d8:4f:f3:a8:96:24:b1:b1:15:92:9c:b6:62:f8:
            42:ac:fd:4c:b6:7d:54:ec:36:04:44:cb:9a:11:2a:90:c8:91:
            68:ed:04:d1:b0:3e:b7:26:4f:fa:ec:d6:d7:3a:11:60:5a:5d:
            d3:fe:92:60:0c:90:43:09:88:7e:26:d9:83:f7:eb:02:d7:c4:
            18:76:7f:8f:66:d7:19:52:6a:80:8e:70:78:09:ab:bb:1c:f5:
            04:bc:20:3d:fa:58:9c:d2:9b:1a:b7:f0:c0:44:bb:af:c9:a4:
  
```

Install the certificate to a local Certificate Authority

- To install the certificate, open a File Explorer, navigate to the certificate directory, and double click on the certificate.

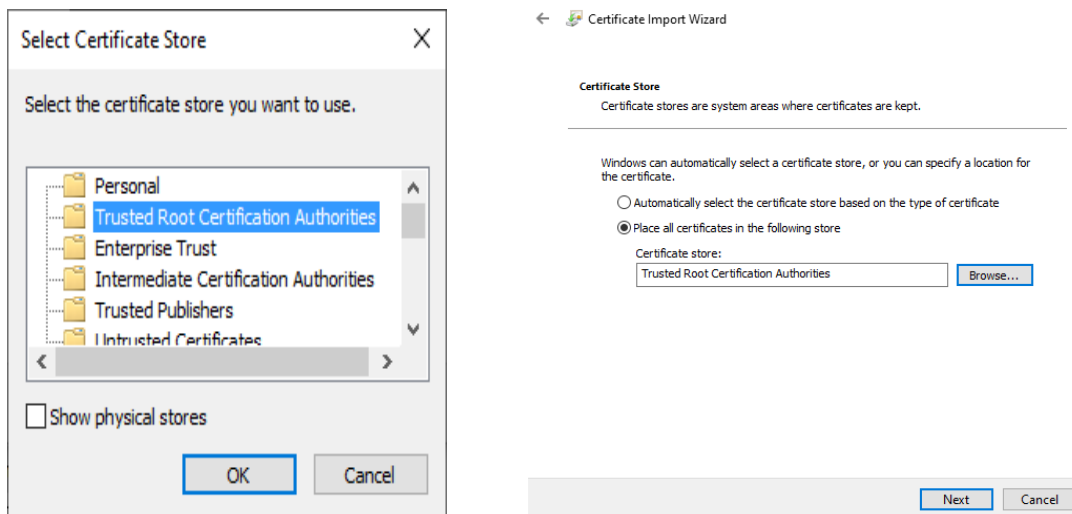


13. Click on **Install Certificate**. In the certification import wizard, select the **current user**, click on **Next**.
Select **Place all certificates in the following store**



The image shows two screenshots of the Windows Certificate Import Wizard. The left screenshot is the 'Welcome to the Certificate Import Wizard' screen. It contains instructions about copying certificates and a 'Store Location' section with two radio buttons: 'Current User' (selected) and 'Local Machine'. The right screenshot is the 'Certificate Store' screen. It explains that certificate stores are system areas where certificates are kept. It offers two options: 'Automatically select the certificate store based on the type of certificate' (unselected) and 'Place all certificates in the following store' (selected). Below this is a text box for 'Certificate store:' and a 'Browse...' button. Both screenshots have 'Next' and 'Cancel' buttons at the bottom.

Browse → **select Trusted Root Certification Authorities**



The image shows two screenshots. The left screenshot is a 'Select Certificate Store' dialog box. It asks the user to 'Select the certificate store you want to use.' and displays a list of stores: 'Personal', 'Trusted Root Certification Authorities' (highlighted), 'Enterprise Trust', 'Intermediate Certification Authorities', 'Trusted Publishers', and 'I Trusted Certificates'. There is a 'Show physical stores' checkbox and 'OK' and 'Cancel' buttons. The right screenshot is the 'Certificate Store' screen of the Certificate Import Wizard, showing the 'Place all certificates in the following store' option selected. The 'Certificate store:' text box now contains 'Trusted Root Certification Authorities', and the 'Browse...' button is highlighted. 'Next' and 'Cancel' buttons are at the bottom.

Click on **Finish**.

Configure the react-scripts start to use the generated certificate and key

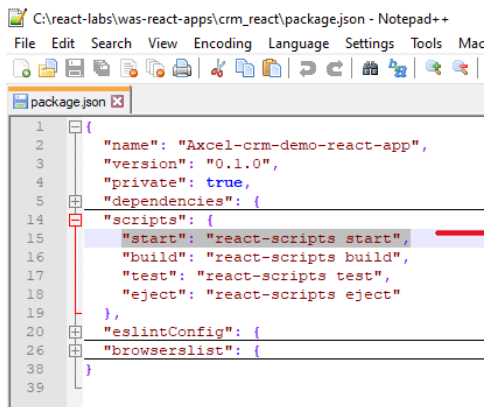
14. In your react project, modify the **package.json** scripts to start using HTTPS

Replace the existing script command: **"start": "react-scripts start"**,

With the **following script command**:

"start": "set HTTPS=true&&set SSL_CERT_FILE=server.crt&&set SSL_KEY_FILE=server.key&&react-scripts start",

package.json

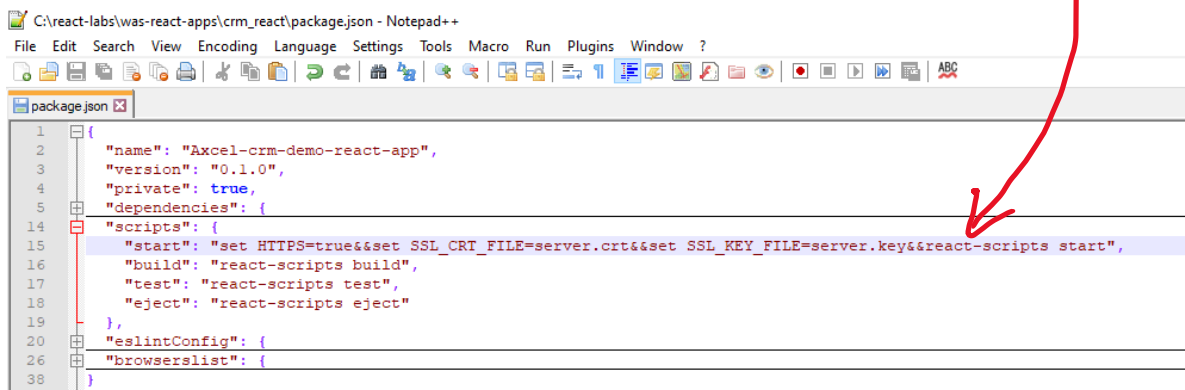


A screenshot of a Notepad++ window showing the original package.json file. The file is located at C:\react-labs\was-react-apps\crm_react\package.json. The JSON content is as follows:

```
{
  "name": "Axcel-crm-demo-react-app",
  "version": "0.1.0",
  "private": true,
  "dependencies": {
  },
  "scripts": {
    "start": "react-scripts start",
    "build": "react-scripts build",
    "test": "react-scripts test",
    "eject": "react-scripts eject"
  },
  "eslintConfig": {
  },
  "browserslist": {
  }
}
```

A red arrow originates from the "start" script command and points towards the modified version in the next screenshot.

Modified package.json using HTTPS



A screenshot of a Notepad++ window showing the modified package.json file. The file is located at C:\react-labs\was-react-apps\crm_react\package.json. The JSON content is as follows:

```
{
  "name": "Axcel-crm-demo-react-app",
  "version": "0.1.0",
  "private": true,
  "dependencies": {
  },
  "scripts": {
    "start": "set HTTPS=true&&set SSL_CERT_FILE=server.crt&&set SSL_KEY_FILE=server.key&&react-scripts start",
    "build": "react-scripts build",
    "test": "react-scripts test",
    "eject": "react-scripts eject"
  },
  "eslintConfig": {
  },
  "browserslist": {
  }
}
```

A red arrow points to the modified "start" script command.

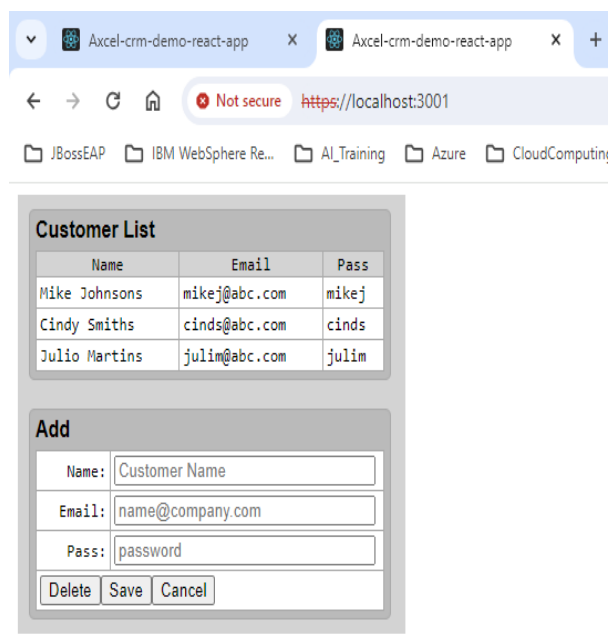
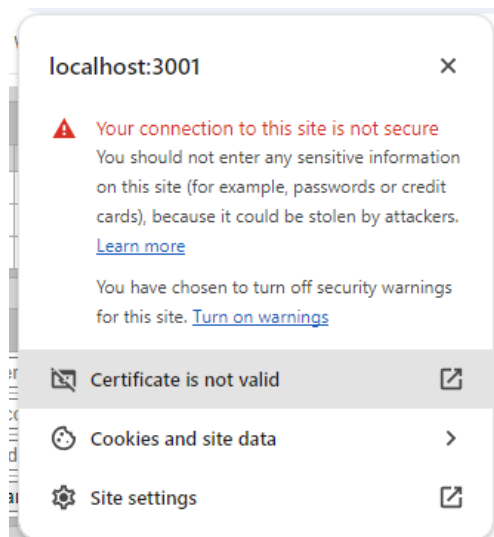
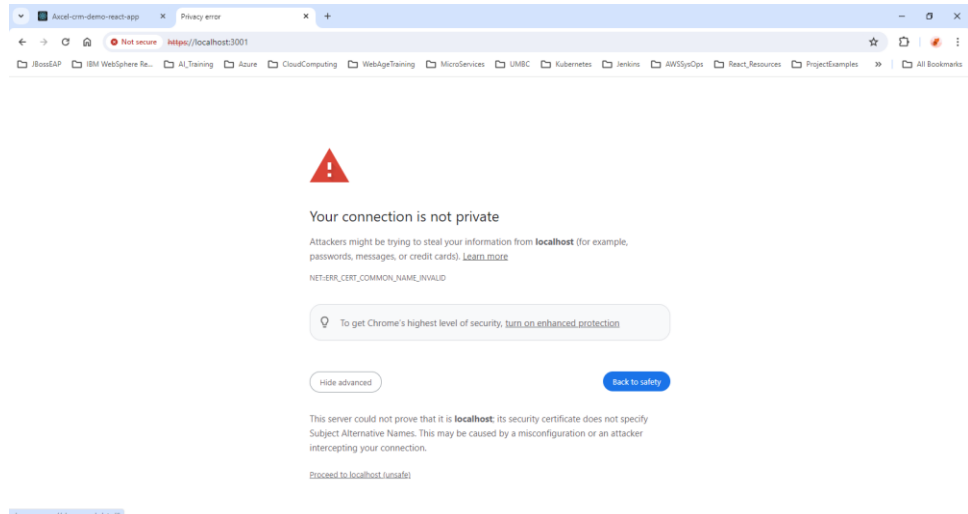
and **save** the file.

15. Open a command window, cd to C:\react-labs\was-react-apps\crm_react

Type the npm command to run the application

npm start

16. Open a browser, go to <https://localhost:3001>



Name	Email	Pass
Mike Johnsons	mikej@abc.com	mikej
Cindy Smiths	cinds@abc.com	cinds
Julio Martins	julim@abc.com	julim

Add

Name:

Email:

Pass:

What can you do to make your ssl certificate valid?