

SOFTWARE SECURITY ASSIGNMENT 2

MS20903206, MS20909796

MSC IN CYBER SECYRITY Sri Lanka Institute in Information Technology

Contents

Software Security Assignment 2	2
Building OAuth2 single sign on app using Facebook Access token.....	2
OAuth 2.0 and Federated Identity	2
Federated Identity	3
Assignment implementation: using Facebook OAuth 2.0 API to authenticate users.....	4
Step 01:	4
Step 02:	5
Application execution flow	7
Appendix	9
Screenshots	9
SC01	9
SC02	10
SC03	11
SC04	12
SC05	13
SC06	14
SC07	15
Source Codes.....	16
Index.html	16
OAuth2GroupAssignmentApplication.java	18
FacebookOAuth2Config.java.....	18
FacebookUserController.java.....	19
application.yml.....	19

Software Security Assignment 2

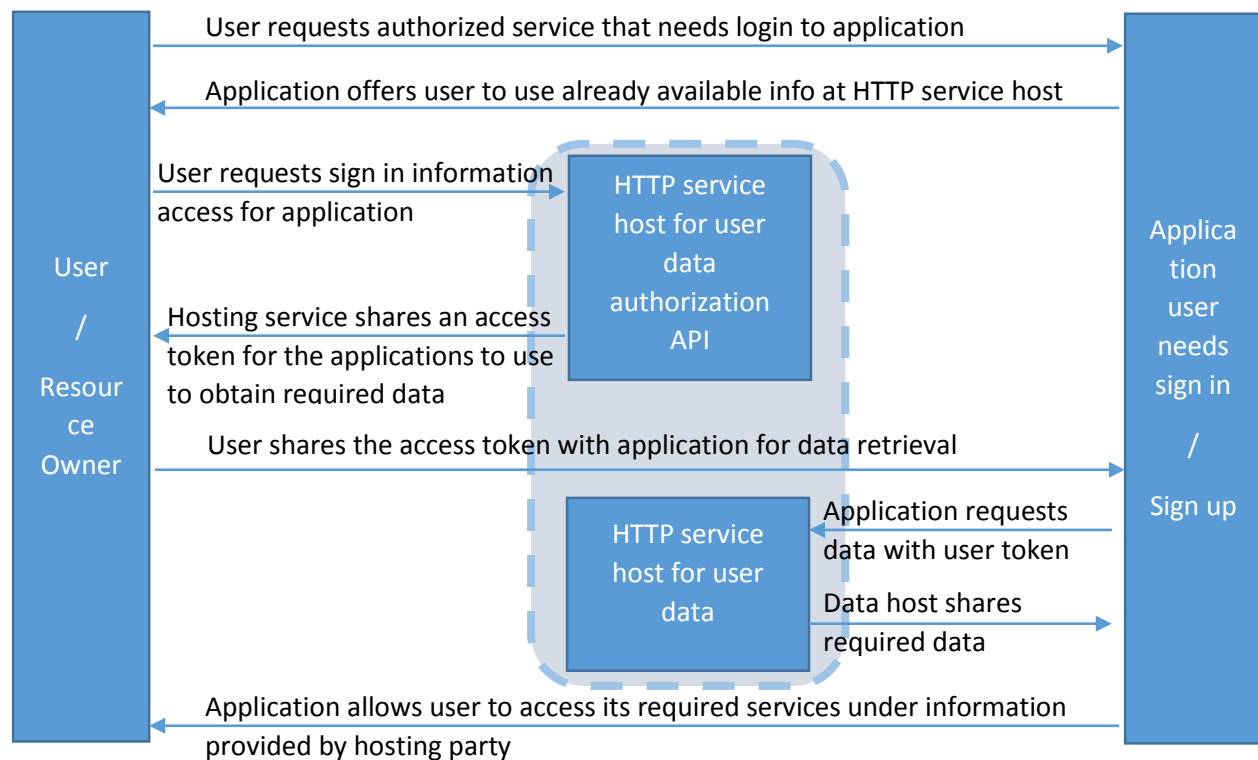
Building OAuth2 single sign on app using Facebook Access token

- MS 20903206 V.A.C.D.Vidanaarachchi
- MS 20909796 C.M. Algawatta

OAuth 2.0 and Federated Identity

RFC 6749 for OAuth 2.0 defines OAuth as “The OAuth 2.0 authorization framework enables a third-party application to obtain limited access to an HTTP service, either on behalf of a resource owner by orchestrating an approval interaction between the resource owner and the HTTP service, or by allowing the third-party application to obtain access on its own behalf.”

This denotes as OAuth 2.0 is a process/standard to grant a third party (user/ application) access to a hosted resource on an http based service owned by a subject with an process between third-party and hosted service. Following diagram shows the core process of OAuth 2.0 as described above:



1. The application requests authorization to access service resources from the user
2. If the user authorized the request, the application receives an authorization grant
3. The application requests an access token from the authorization server (API) by presenting authentication of its own identity, and the authorization grant
4. If the application identity is authenticated and the authorization grant is valid, the authorization server (API) issues an access token to the application. Authorization is complete.
5. The application requests the resource from the resource server (API) and presents the access token for authentication
6. If the access token is valid, the resource server (API) serves the resource to the application

However, all authorization service providers requires the re registration of the services users (applications or users) as end clients information protection is also important. For this purpose Auth tokens, application tokens and shred secrets as API Keys bring used.

Federated Identity

Federated Identity is the concept of using someone's digital identity in one platform across multiple domains/platforms under users' permissions without creating separate identities for each systems. As discussed above, OAuth 2.0 could be used as a means of implementing Federated Identity. It helps to use same account data used in Google account, FaceBook or Twitter to be used across multiple platforms.

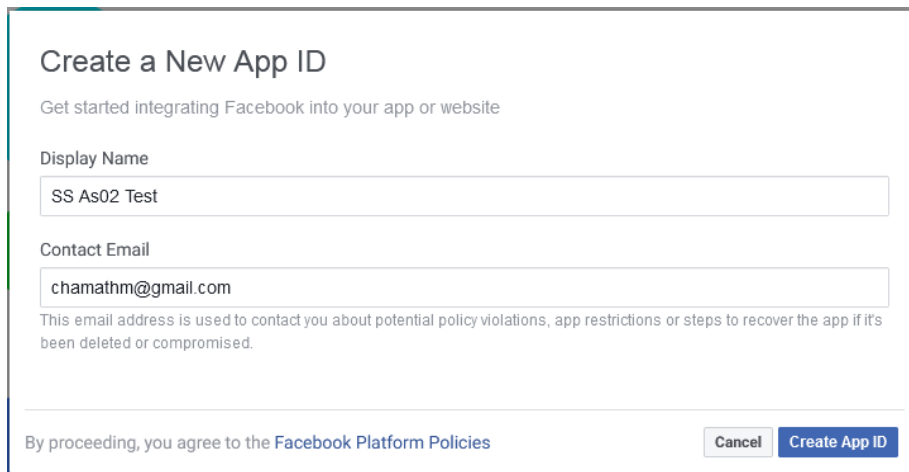
This benefits the user to manage all access over single account while elevates the risk of exposure of that account information across multiple domains.

Assignment implementation: using Facebook OAuth 2.0 API to authenticate users

As required by the assignment work, a sample program was implemented for Facebook OAuth 2.0 authorization.

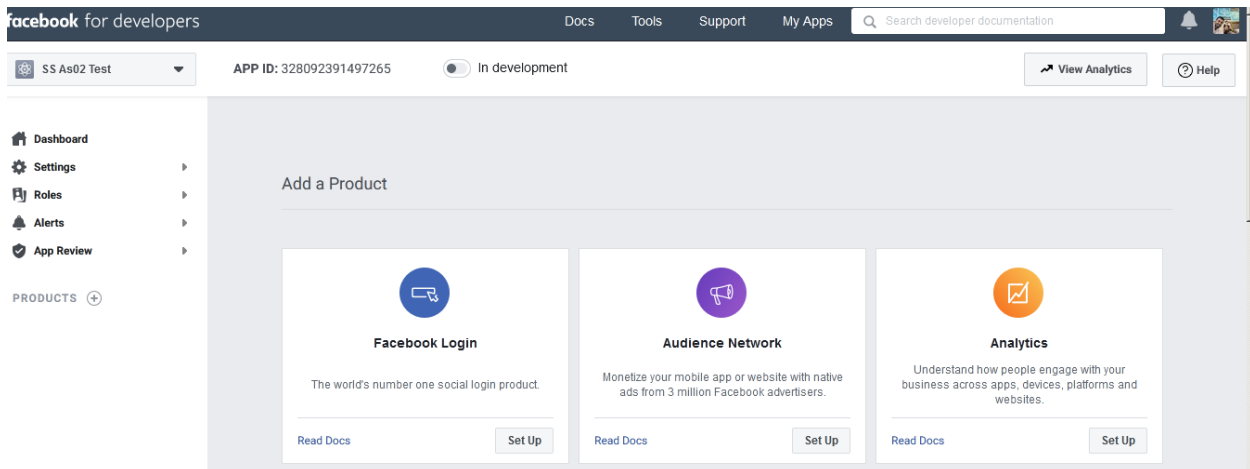
Step 01: Obtain the client ID and secret from Facebook to access OAuth service API:

Access <https://developers.facebook.com/> and create Application:



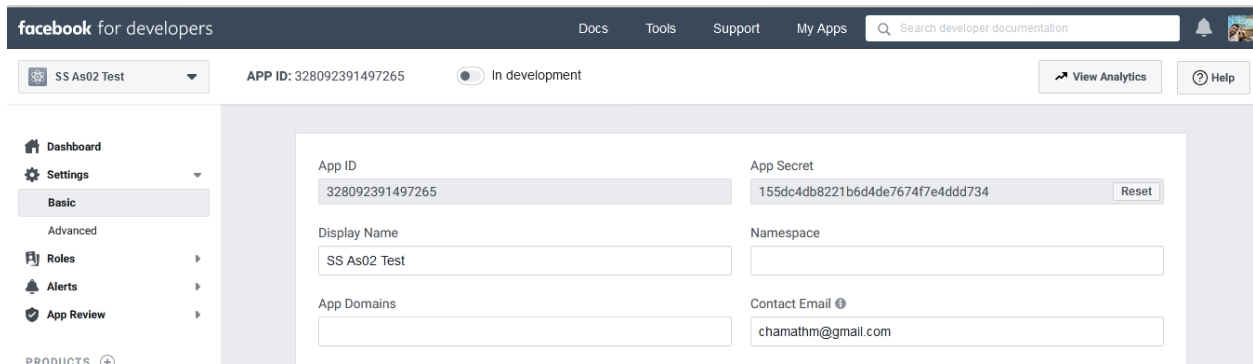
The screenshot shows the 'Create a New App ID' form. The title is 'Create a New App ID' with a subtitle 'Get started integrating Facebook into your app or website'. There are two input fields: 'Display Name' with the value 'SS As02 Test' and 'Contact Email' with the value 'chamathm@gmail.com'. Below the email field is a note: 'This email address is used to contact you about potential policy violations, app restrictions or steps to recover the app if it's been deleted or compromised.' At the bottom, there is a checkbox 'By proceeding, you agree to the Facebook Platform Policies' and two buttons: 'Cancel' and 'Create App ID'.

Then select Facebook login:



Select WWW as we are developing a web application and provide the web URL Facebook will provide Javascript code for Auth process. However, as we use `"org.springframework.boot.autoconfigure.security.oauth2.client.EnableOAuth2Sso"` we require only client ID and Secret values:

Once registration complete we would be grated with Facebook app id and secret key:



The screenshot shows the Facebook for Developers dashboard. The top navigation bar includes 'facebook for developers', 'Docs', 'Tools', 'Support', 'My Apps', and a search bar. Below this, the app 'SS As02 Test' is selected, showing its 'APP ID: 328092391497265' and a toggle for 'In development'. A sidebar on the left contains links to 'Dashboard', 'Settings' (with a sub-menu for 'Basic' and 'Advanced'), 'Roles', 'Alerts', and 'App Review'. The main content area displays the app's configuration: 'App ID' (328092391497265), 'App Secret' (155dc4db8221b6d4de7674f7e4ddd734) with a 'Reset' button, 'Display Name' (SS As02 Test), 'Namespace' (empty), 'App Domains' (empty), and 'Contact Email' (chamathm@gmail.com). There are also buttons for 'View Analytics' and 'Help'.

Step 02: Development of the application

The code developed for this application is for Proof of Concept only and represent only the OAuth 2.0 access process. Below code is developed with the use of Java spring framework as it has prebuilt in function to handle OAuth 2.0 functions.

FacebookOAuth2Config.java

```
package SS_AS02_Chamath_Chethiya;

import org.springframework.boot.autoconfigure.security.oauth2.client.EnableOAuth2Sso;
import org.springframework.context.annotation.Configuration;
import org.springframework.security.config.annotation.web.builders.HttpSecurity;
import org.springframework.security.config.annotation.web.configuration.WebSecurityConfigurerAdapter;

@EnableOAuth2Sso
@Configuration
public class FacebookOAuth2Config extends WebSecurityConfigurerAdapter {

    @Override
    protected void configure(HttpSecurity http) throws Exception {
        http
            .antMatcher("/**")
            .authorizeRequests()
            .antMatchers("/", "/login**", "/webjars/**", "/error**")
            .permitAll()
            .anyRequest()
            .authenticated();
    }
}
```

The client ID and application ID we obtained from FaceBook are configured into application.yml file as below:

```
security:
  oauth2:
    client:
      clientId: 328092391497265
      clientSecret: 155dc4db8221b6d4de7674f7e4ddd734
      accessTokenUri: https://graph.facebook.com/oauth/access_token
      userAuthorizationUri: https://www.facebook.com/dialog/oauth
      tokenName: oauth_token
      authenticationScheme: query
      clientAuthenticationScheme: form
    resource:
      userInfoUri: https://graph.facebook.com/me
```

This information is used by the OAuth 2.0 method in Spring framework to identify the access requesting application to FaceBook authorization API (<https://www.facebook.com/dialog/oauth>) using client ID and secret. Once the authorization token is received, application requests data from the Facebook API hosted in <https://graph.facebook.com/me> to obtain user account information.

On the frontend a static webpage for demo purposes is hosted only with single button to demonstrate login with FaceBook function:

Index.html

```
<div class="container unauthenticated">
  <a href="/login" class="fb btn"> <i class="fa fa-facebook fa-fw"></i>Login
  with FB account</a>
</div>

<div class="container authenticated" style="display:none">
<div class="card">
  <img src="" alt="Facebook User Image" style="width:100%">
  <p class="title">User login As</p>
  <h1><span id="user"></span></h1>
  <p><button>Logout</button></p>
</div>
```

Application execution flow

1. User clicks “Login with FB” Button

This triggers the application to initiate request for OAuth 2.0 function for FaceBook

Request header: (SC01 in Appendix)

```
Request URL:http://127.0.0.1:8080/login
Request Method:GET
Remote Address:127.0.0.1:8080
Status Code:
200
Version:HTTP/1.1
Referrer Policy:no-referrer-when-downgrade
```

As response, Application server returns the Facebook OAuth 2.0 request redirect URL with applications client ID(SC02 in Appendix):

```
HTTP/1.1 302
X-Content-Type-Options: nosniff
X-XSS-Protection: 1; mode=block
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
X-Frame-Options: DENY
Location: https://www.facebook.com/dialog/oauth?client_id=
328092391497265&redirect_uri=http://127.0.0.1:8080/login&response_type=code&state=gJOmbY
Content-Length: 0
Date: Mon, 11 May 2020 11:02:51 GMT
Keep-Alive: timeout=60
Connection: keep-alive
```

2. The Client web browser uses this information to request access token for the application from Facebook OAuth API(SC03 in Appendix) :

```
Request
URL:https://www.facebook.com/dialog/oauth?client_id=691419415009768&redirect_uri=http://127
.0.0.1:8080/login&response_type=code&state=gJOmbY
Request Method:GET
Remote Address:157.240.7.35:443
Status Code:
200
Version:HTTP/2
Referrer Policy:no-referrer-when-downgrade
```


As response fir this, Facebook verifies from the user to share access token for requested data (SC04, SC05 in Appendix) :

```
Host: www.facebook.com
User-Agent: Mozilla/5.0 (Windows NT 6.1; Win64; x64; rv:76.0) Gecko/20100101 Firefox/76.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer:
https://www.facebook.com/dialog/oauth?client_id=328092391497265&redirect_uri=http%3A%2F%
2Flocalhost%3A8080%2Flogin&response_type=code&state=K91B4Z
Content-Type: application/x-www-form-urlencoded
Content-Length: 912
Origin: https://www.facebook.com
Connection: keep-alive
Cookie: fr=1NiKOxbalFXPi4bt5.AWXN4zAOkk3Z-
VmmU7C6QnqQN4w.BeMZMw.Ei.AAA.0.0.BeUTZu.AWUo4hdY; sb=MJMxXvnlvIXV3Rm-csi_Mc6c;
datr=MJMxXkekEBbl2OXVbO_ifYzO; c_user=632449571;
xs=22%3AwpVQptNjhxKD1Q%3A2%3A1580307292%3A480%3A899; wd=1366x234;
_fbp=fb.1.1589135380402.1523472437; spin=r.1002107478_b.trunk_t.1589139405_s.1_v.2_;
act=1589197574215%2F6;
presence=EDvF3EtimeF1589197721EuserFA2632449571A2EstateFDutF1589197691117CEchF_7bCC
Upgrade-Insecure-Requests: 1
TE: Trailers
```

3. Once the user confirms Facebook shares access token for user data read to the application and forwarded to web application (SC06 – Facebook reply, SC07 – forward to application in Appendix):

```
Request
URL:http://localhost:8080/login?code=AQCzzwT9ILLjGyev7p2Am7fPkTEKqOpMQLDIOaVa2G9LMOp-
FwErjyMAdCQTyauZDvqQ1BJkAY-dN8MIZ4puDfUR-xsnePO-2PWXerCt4dhjMlye_CP-
YIMXLiggn329g1jN6T85tPojN0Ap8dzPzo3uBM6EoVXTzpRpfxTS5Gx7c3y7r1_D40vCRBltzRFTHGnm-
tbxIO0hIBNS5oF4oltDom1MV9qkbXbvCrT5YfKU6UFdAkIKO5Gut6hkDVlftskFjmgfLHVfapw46py4ZWe
UVqwZDNldFDr_NN8sHSNCgpnQ3v8TDYIhH3IYAQQ_yfB4Fs8SJ_V0Vkif1lkCZxnN&state=K91B4Z
Request Method:GET
Remote Address:[::1]:8080
Status Code:
200
Version:HTTP/1.1
```

By presenting this token to FaceBook API “graph.facebook.com/me” application is able to obtain user information from the background.

Appendix

Screenshots

SC01

The screenshot shows a web browser with three tabs: "facebook oauth 2.0 client ID and...", "SS As02 Test - Roles - Facebook f...", and "SS Assignment 02". The address bar shows "localhost:8080/#_=_". The browser is displaying a Facebook login page. The Chrome DevTools Network tab is open, showing a list of requests. The selected request is a GET request to "localhost:8080/login". The response headers are visible, showing a 302 status code and various headers including Cache-Control, Connection, Content-Length, Date, Expires, Keep-Alive, Location, Pragma, Set-Cookie, X-Content-Type-Options, X-Frame-Options, and X-XSS-Protection.

Status	Method	Domain	File	Cause	Type	Transferred	Size
302	GET	localhost:8080	login	document	html	3.16 KB	2.65 KB
302	GET	www.facebook...	oauth?client_id=328092391497265&redirect_uri=ht...	document	html	4.09 KB	2.65 KB
302	GET	localhost:8080	login?code=AQCOE9uMypRfxhGLuR69FXaXENdjap3...	document	html	3.05 KB	2.65 KB
200	GET	localhost:8080	/	document	html	3.15 KB	2.65 KB
302	GET	localhost:8080	favicon.ico	img	html	cached	2.65 KB
302	GET	localhost:8080	login	img	html	cached	2.65 KB
302	GET	www.faceb...	oauth?client_id=328092391497265&redirect_uri=ht...	img	html	cached	2.65 KB
302	GET	localhost:8080	login?code=AQCOE9uMypRfxhGLuR69FXaXENdjap3...	img	html	cached	2.65 KB
200	GET	localhost:8080	/	img	html	cached	2.65 KB

Request URL: http://localhost:8080/login
Request Method: GET
Remote Address: [::1]:8080
Status Code: 302
Version: HTTP/1.1
Referrer Policy: no-referrer-when-downgrade

Response Headers (526 B)

- Cache-Control: no-cache, no-store, max-age=0, must-revalidate
- Connection: keep-alive
- Content-Length: 0
- Date: Mon, 11 May 2020 12:52:21 GMT
- Expires: 0
- Keep-Alive: timeout=60
- Location: https://www.facebook.com/dialo...esponse_type=code&state=evXQ30
- Pragma: no-cache
- Set-Cookie: JSESSIONID=DDA7D6E32CB2FF29B52...73B06350CA3; Path=/; HttpOnly
- X-Content-Type-Options: nosniff
- X-Frame-Options: DENY
- X-XSS-Protection: 1; mode=block

Request Headers (425 B)

SC02

The screenshot shows a web browser with three tabs: "facebook oauth 2.0 client ID and...", "f SS As02 Test - Roles - Facebook", and "SS Assignment 02". The address bar shows "localhost:8080/#_=_". The DevTools interface is open, with the "Network" tab selected. It displays a list of 9 requests, with the third request (GET to localhost:8080/login?code=AQCOE9uMypRfxhGLuR69FXaXENDjap3...) selected. The "Headers" panel on the right shows the response headers for this request, including "cache-control: private, no-cache, no-store, must-revalidate", "content-length: 0", "content-security-policy: default-src * data: blob: 'self'; upgrade-insecure-requests;", "content-type: text/html; charset=utf-8", "date: Mon, 11 May 2020 12:52:21 GMT", "expires: Sat, 01 Jan 2020 00:00:00 GMT", "facebook-api-version: v7.0", "location: http://localhost:8080/login?code=AQCOE9uMypRfxhGLuR69FXaXENDjap3...", "pragma: no-cache", "strict-transport-security: max-age=15552000; preload", "x-content-type-options: nosniff", "x-fb-debug: twf62L69AzVBia6gt+Q2w12cwS5T1...JtrOhV6H/au7UIoZpUvFFUCIsPA==", and "X-Firefox-Spdy: h2". The "Console" tab at the bottom shows two messages: a warning about the non-standard "zoom" property and a red error message stating "The Components object is deprecated. It will soon be removed."

Status	Method	Domain	File	Cause	Type	Transferred	Size
302	GET	localhost:8080	login	document	html	3.16 KB	2.65 KB
302	GET	www.facebook.com	oauth?client_id=328092391497265&redirect_uri=http://localhost:8080/login?code=AQCOE9uMypRfxhGLuR69FXaXENDjap3...	document	html	4.09 KB	2.65 KB
302	GET	localhost:8080	login?code=AQCOE9uMypRfxhGLuR69FXaXENDjap3...	document	html	3.05 KB	2.65 KB
200	GET	localhost:8080	/	document	html	3.15 KB	2.65 KB
302	GET	localhost:8080	favicon.ico	img	html	cached	2.65 KB
302	GET	localhost:8080	login	img	html	cached	2.65 KB
302	GET	www.facebook.com	oauth?client_id=328092391497265&redirect_uri=http://localhost:8080/login?code=AQCOE9uMypRfxhGLuR69FXaXENDjap3...	img	html	cached	2.65 KB
302	GET	localhost:8080	login?code=AQCOE9uMypRfxhGLuR69FXaXENDjap3...	img	html	cached	2.65 KB
200	GET	localhost:8080	/	img	html	cached	2.65 KB

9 requests | 23.82 KB / 13.45 KB transferred | Finish: 2.81 s | DOMContentLoaded: 2.31 s | load: 2.32 s

Run | Filter Output | Errors | Warnings | Logs | Info | Debug | CSS | XHR | Requests | Settings

1 | Navigated to http://localhost:8080/login

⚠ This page uses the non standard property "zoom". Consider using calc() in the relevant property values, or using "transform" along with "transform-origin: 0 0". | localhost:8080

❌ The Components object is deprecated. It will soon be removed. | localhost:8080

SC03

⌵

Headers

Cookies

Params

Response

Timings

Security

Request URL: https://www.facebook.com/dialog/oauth?client_id=328092391497265&redirect_uri=http://localhost:8080/login&response_type=code&state=evXQ30

Request Method: GET

Remote Address: 157.240.13.35:443

Status Code: 302 Found ?

Version: HTTP/2

Referrer Policy: no-referrer-when-downgrade

Edit and Resend

⌵ Filter Headers

▼ Response Headers (1.441 KB) Raw Headers ☐

HTTP/2 302 Found

facebook-api-version: v7.0

cache-control: private, no-cache, no-store, must-revalidate

expires: Sat, 01 Jan 2000 00:00:00 GMT

pragma: no-cache

strict-transport-security: max-age=15552000; preload

location: http://localhost:8080/login?code=AQCOE9uMypRfxhGLuR69FXaXENDjap36a_UxNWlUNhrtrHDRTfxg6y7L6v3UAU_FQfi-9Gf6z-1m-FyndVRWEuzWPpcnHKJF-k

content-security-policy: default-src * data: blob: 'self';script-src *.facebook.com *.fbcdn.net *.facebook.net *.google-analytics.com *.virtu

x-content-type-options: nosniff

x-frame-options: DENY

x-xss-protection: 0

content-type: text/html; charset="utf-8"

x-fb-debug: twfE62L69AzVBia6gt+Q2w12cwS5T1s1f11ZZIR2EeJXUakwiGnryfd0hYLJtrOhV6H/au7UIoZpUVFFUCIsPA==

content-length: 0

date: Mon, 11 May 2020 12:52:21 GMT

X-Firefox-Spdy: h2

⌵ Request Headers (914 B) Raw Headers ☒

The screenshot shows the 'Headers' tab in a web browser's developer tools. The 'Response' section is expanded, displaying various HTTP response headers. The 'Raw Headers' toggle is turned off. The headers listed include cache-control, content-length, content-security-policy, content-type, date, expires, facebook-api-version, location, pragma, strict-transport-security, x-content-type-options, x-fb-debug, and X-Firefox-Spdy.

Headers Cookies Params Response Timings Security

Request URL: https://www.facebook.com/dialog/oauth?client_id=328092391497265&redirect_uri=http://localhost:8080/login&response_type=code&state=evXQ30

Request Method: GET

Remote Address: 157.240.13.35:443

Status Code: 302 Found ⓘ

Version: HTTP/2

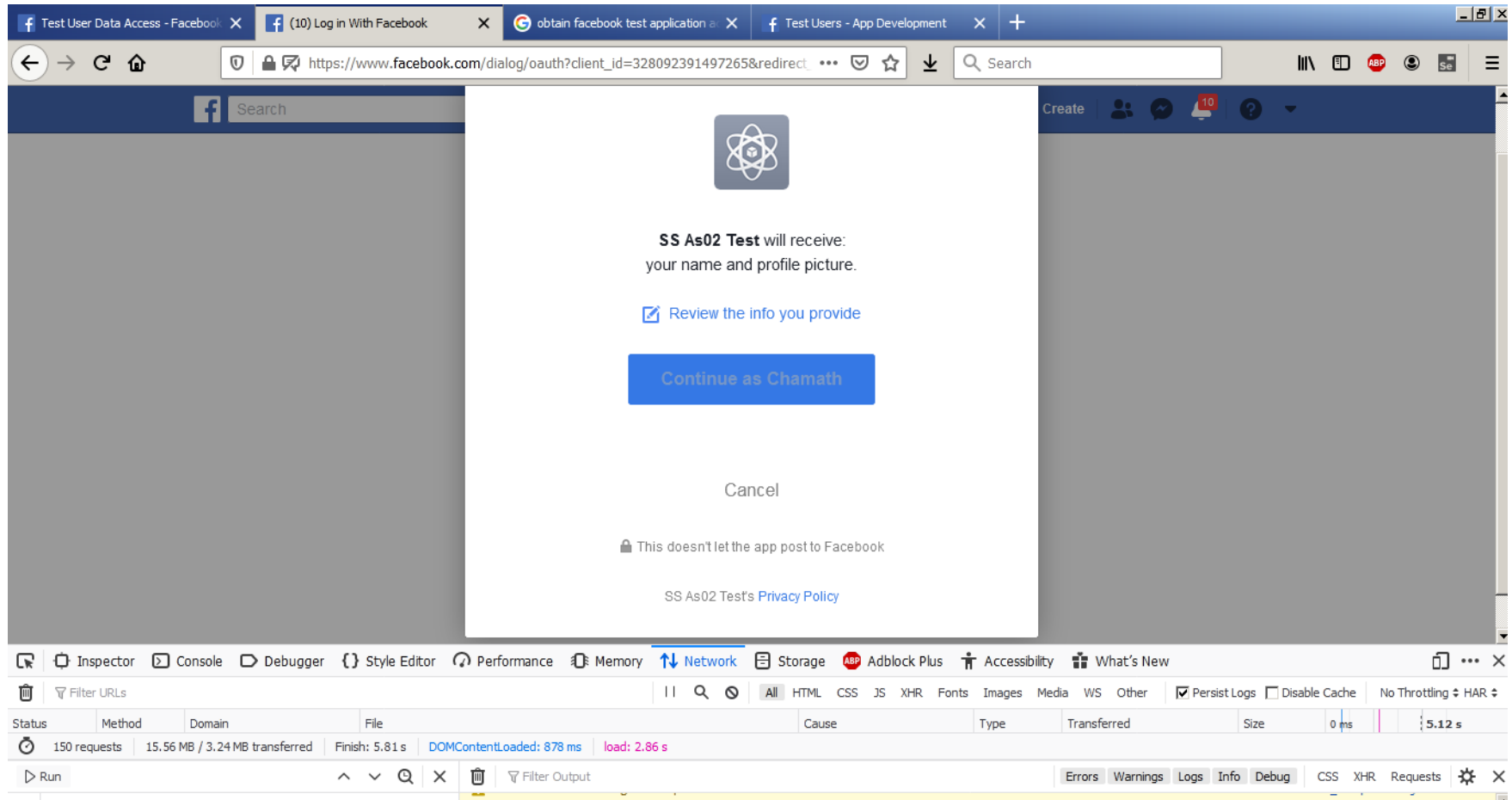
Referrer Policy: no-referrer-when-downgrade

Filter Headers

Response Headers (1.441 KB) Raw Headers ☐

- cache-control: private, no-cache, no-store, must-revalidate
- content-length: 0
- content-security-policy: default-src * data: blob: 'self'; upgrade-insecure-requests;
- content-type: text/html; charset="utf-8"
- date: Mon, 11 May 2020 12:52:21 GMT
- expires: Sat, 01 Jan 2000 00:00:00 GMT
- facebook-api-version: v7.0
- location: http://localhost:8080/login?co...AH_q2E2NvJP7c&state=evXQ30#_=_
- pragma: no-cache
- strict-transport-security: max-age=15552000; preload
- x-content-type-options: nosniff
- x-fb-debug: twfE62L69AzVBia6gt+Q2w12cwS5T1...JtrOhV6H/au7UIoZpUvFFUCIsPA==
- X-Firefox-Spdy: h2

SC05



SC06

The screenshot shows a web browser's developer tools interface, specifically the Network tab. The top navigation bar includes icons for Memory, Network, Storage, Adblock Plus, Accessibility, and What's New. Below this, a filter bar shows 'All' selected, with other filters like HTML, CSS, JS, XHR, Fonts, Images, Media, WS, and Other. A status bar at the top right indicates 'Persist Logs' is checked, 'Disable Cache' is unchecked, and 'No Throttling' is selected. The main panel displays the 'Headers' tab for a selected request. The request details are as follows:

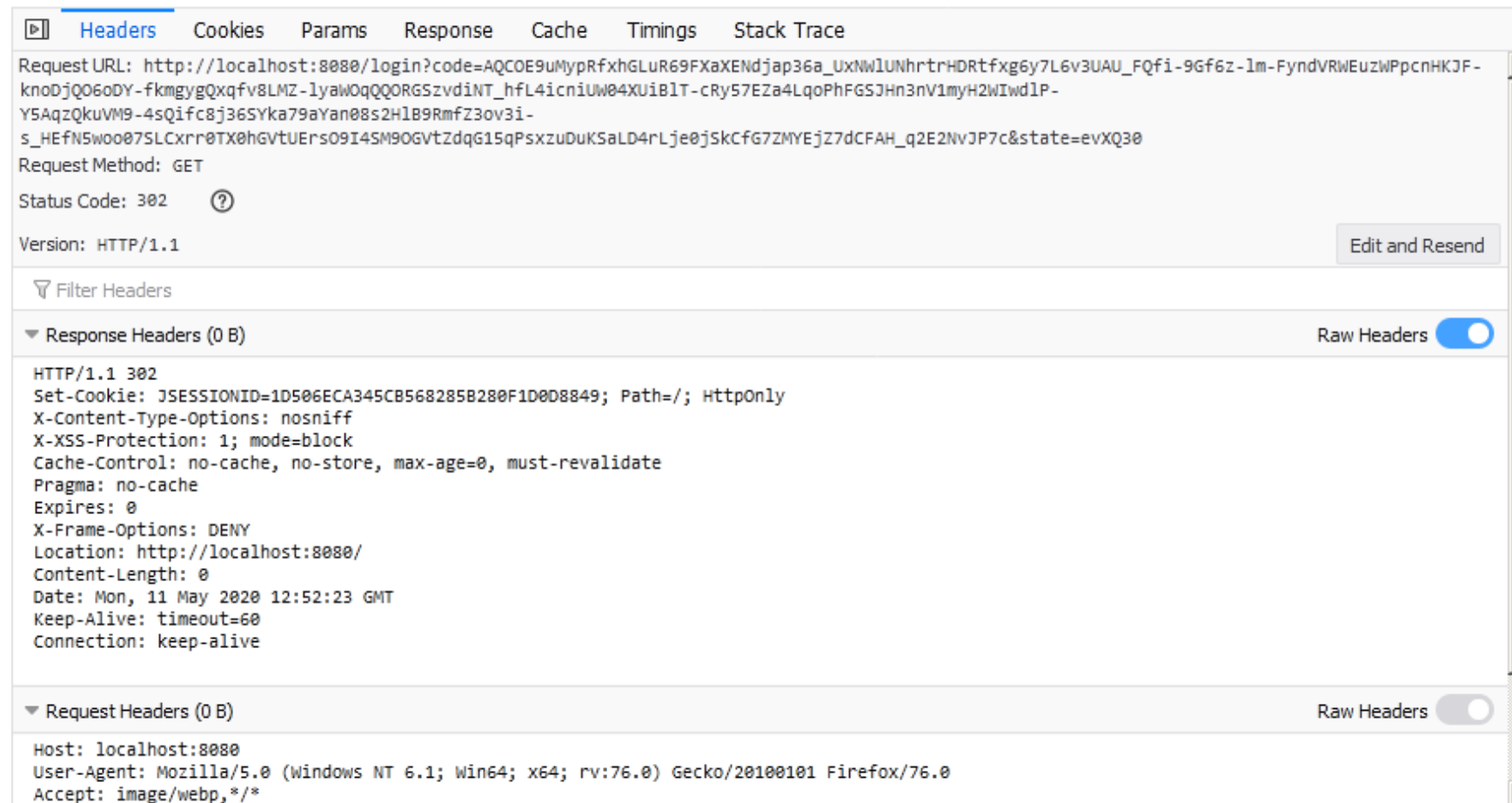
- Request URL: https://www.facebook.com/dialog/oauth?client_id=328092391497265&redirect_uri=http://localhost:8080/login&response_type=code&state=evXQ30
- Request Method: GET
- Remote Address: 157.240.13.35:443
- Status Code: 302 Found
- Version: HTTP/2
- Referrer Policy: no-referrer-when-downgrade

Below the request details, there is a 'Filter Headers' section and a 'Response Headers (1.441 KB)' section. The response headers are listed as follows:

- cache-control: private, no-cache, no-store, must-revalidate
- content-length: 0
- content-security-policy: default-src * data: blob: 'self'; upgrade-insecure-requests;
- content-type: text/html; charset=utf-8
- date: Mon, 11 May 2020 12:52:21 GMT
- expires: Sat, 01 Jan 2000 00:00:00 GMT
- facebook-api-version: v7.0
- location: http://localhost:8080/login?code=AQCOE9uMypRfxhGLuR69FXaXENDjap36a_UxNWlUNhrtrHDrftxg6y7L6v3UAU_FQfi-9Gf6z-lm-FyndVRWEuzWPpcnHKJF-knoDjQO6oDY-fkmgYGQxqfv8LMZ-lYaWOqQQORGSzvdINT_hfL4icniUW04XUiBIT-cRy57EZa4LqoPhFGSJHn3nV1myH2WIwdlP-Y5AqzQkuVM9-4sQifc8j36SYka79aYan08s2HlB9RmfZ3ov3i-s_HEfN5woo07SLCxr0TX0hGVtUErsO9I4SM9OGvtZdqG15qPsxzuDuKSaLD4rLje0jSkCfG7ZMYEjZ7dCFAH_q2E2NvJP7c&state=evXQ30#_=_
- pragma: no-cache
- strict-transport-security: max-age=31536000; includeSubDomains
- x-content-type-options: nosniff
- x-fb-debug: twfEdg
- X-Firefox-Spdy: h2

The 'location' header is highlighted in blue. A tooltip is visible over the 'location' header, displaying the full URL: http://localhost:8080/login?code=AQCOE9uMypRfxhGLuR69FXaXENDjap36a_UxNWlUNhrtrHDrftxg6y7L6v3UAU_FQfi-9Gf6z-lm-FyndVRWEuzWPpcnHKJF-knoDjQO6oDY-fkmgYGQxqfv8LMZ-lYaWOqQQORGSzvdINT_hfL4icniUW04XUiBIT-cRy57EZa4LqoPhFGSJHn3nV1myH2WIwdlP-Y5AqzQkuVM9-4sQifc8j36SYka79aYan08s2HlB9RmfZ3ov3i-s_HEfN5woo07SLCxr0TX0hGVtUErsO9I4SM9OGvtZdqG15qPsxzuDuKSaLD4rLje0jSkCfG7ZMYEjZ7dCFAH_q2E2NvJP7c&state=evXQ30#_=_

At the bottom of the developer tools, there is a 'Filter Output' section and a row of tabs for 'Errors', 'Warnings', 'Logs', 'Info', 'Debug', 'CSS', 'XHR', and 'Requests'.



Headers Cookies Params Response Cache Timings Stack Trace

Request URL: `http://localhost:8080/login?code=AQCOE9uMypRfxhGLuR69FXaXENDjap36a_UxNWlUNhrtrHDrtfxg6y7L6v3UAU_FQfi-9Gf6z-1m-FyndVRWEuzWPpcnHKJF-knoDjQ06oDY-fkmgYgQxqfv8LMZ-lyawOqQQORGSzvdINT_hfL4icniUW04XUiB1T-cRy57Eza4LqoPhFGSjHn3nV1myH2WIwd1P-Y5AqzQkuVM9-4sQifc8j36SYka79aYan08s2H1B9RmfZ3ov3i-s_HEfN5woo07SLCxr0TX0hGvtUers09I45M9OGvtZdqG15qPsxzDuKSaLD4rLje0jSkCfG7ZMYEjZ7dCFAH_q2E2NvJP7c&state=evXQ30`

Request Method: GET

Status Code: 302 ?

Version: HTTP/1.1 Edit and Resend

Filter Headers

▼ Response Headers (0 B) Raw Headers

HTTP/1.1 302
Set-Cookie: JSESSIONID=1D506ECA345CB568285B280F1D008849; Path=/; HttpOnly
X-Content-Type-Options: nosniff
X-XSS-Protection: 1; mode=block
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
X-Frame-Options: DENY
Location: `http://localhost:8080/`
Content-Length: 0
Date: Mon, 11 May 2020 12:52:23 GMT
Keep-Alive: timeout=60
Connection: keep-alive

▼ Request Headers (0 B) Raw Headers

Host: localhost:8080
User-Agent: Mozilla/5.0 (Windows NT 6.1; Win64; x64; rv:76.0) Gecko/20100101 Firefox/76.0
Accept: image/webp,*/.*

Source Codes

Index.html

```
<!doctype html>
<html lang="en">
<head>
  <meta charset="utf-8"/>
  <meta http-equiv="X-UA-Compatible" content="IE=edge"/>
  <title>SS Assignment 02</title>
  <meta name="description" content=""/>
  <meta name="viewport" content="width=device-width"/>
  <base href="/"/>
  <link rel="stylesheet" type="text/css" href="/webjars/bootstrap/css/bootstrap.min.css"/>
  <script type="text/javascript" src="/webjars/jquery/jquery.min.js"></script>
  <script type="text/javascript" src="/webjars/bootstrap/js/bootstrap.min.js"></script>
  <style>
    body {
      font-family: Arial, Helvetica, sans-serif;
    }
    * {
      box-sizing: border-box;
    }
    /* style the container */
    .container {
      position: relative;
      border-radius: 5px;
      background-color: #f2f2f2;
      padding: 20px 0 30px 0;
    }
    /* style inputs and link buttons */
    input,
    .btn {
      width: 100%;
      padding: 12px;
      border: none;
      border-radius: 4px;
      margin: 5px 0;
      opacity: 0.85;
      display: inline-block;
      font-size: 17px;
      line-height: 20px;
      text-decoration: none; /* remove underline from anchors */
    }
    input:hover,
    .btn:hover {
      opacity: 1;
    }

    /* add appropriate colors to fb, twitter and google buttons */
    .fb {
      background-color: #7a7774;
      color: white;
    }

    .card {
      box-shadow: 0 4px 8px 0 rgba(0, 0, 0, 0.2);
      max-width: 300px;
      margin: auto;
      text-align: center;
      font-family: arial;
    }

    .title {
      color: grey;
      font-size: 18px;
    }
  </style>
</head>
```

```

button {
  border: none;
  outline: 0;
  display: inline-block;
  padding: 8px;
  color: white;
  background-color: #000;
  text-align: center;
  cursor: pointer;
  width: 100%;
  font-size: 18px;
}
a {
  text-decoration: none;
  font-size: 22px;
  color: black;
}
button:hover, a:hover {
  opacity: 0.7;
}
</style>
</head>
<body>
<h1>SS Assignment 02</h1>

<div class="col">
<div class="container unauthenticated">
  <a href="/login" class="fb btn"> <i class="fa fa-facebook fa-fw"></i>Login with FB account</a>
</div>

  <div class="container authenticated" style="display:none">
<div class="card">
  <img src="" alt="Facebook User Image" style="width:100%">
  <p class="title">User login As</p>
  <h1><span id="user"></span></h1>
  <p><button>Logout</button></p>
</div>
</div>
</div>
</div>

<script type="text/javascript">
  $.get("/user", function(data) {
    $("#user").html(data.userAuthentication.details.name);
    $(".unauthenticated").hide()
    $(".authenticated").show()
  });

  var logout = function() {
    $.post("/logout", function() {
      $("#user").html('');
      $(".unauthenticated").show();
      $(".authenticated").hide();
    })
    return true;
  }
</script>
</body>

</html>

```

OAuth2GroupAssignmentApplication.java

```
package SS_AS02_Chamath_Chethiya;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

public class OAuth2GroupAssignmentApplication {
    public static void main(String[] args) {
        SpringApplication.run(OAuth2GroupAssignmentApplication.class, args);
    }
}
```

FacebookOAuth2Config.java

```
package SS_AS02_Chamath_Chethiya;

import org.springframework.boot.autoconfigure.security.oauth2.client.EnableOAuth2Sso;
import org.springframework.context.annotation.Configuration;
import org.springframework.security.config.annotation.web.builders.HttpSecurity;
import org.springframework.security.config.annotation.web.configuration.WebSecurityConfigurerAdapter;

@EnableOAuth2Sso
@Configuration

public class FacebookOAuth2Config extends WebSecurityConfigurerAdapter {
    @Override
    protected void configure(HttpSecurity http) throws Exception {
        http
            .antMatcher("/**")
            .authorizeRequests()
            .antMatchers("/", "/login**", "/webjars/**", "/error**")
            .permitAll()
            .anyRequest()
            .authenticated();
    }
}
```

FacebookUserController.java

```
package SS_AS02_Chamath_Chethiya;

import java.security.Principal;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RestController;

@RestController
@RequestMapping("/user")

public class FacebookUserController {
    @GetMapping
    public Principal getUser(Principal user) {
        return user;
    }
}
```

application.yml

```
security:
  oauth2:
    client:
      clientId: 328092391497265
      clientSecret: 155dc4db8221b6d4de7674f7e4ddd734
      accessTokenUri: https://graph.facebook.com/oauth/access_token
      userAuthorizationUri: https://www.facebook.com/dialog/oauth
      tokenName: oauth_token
      authenticationScheme: query
      clientAuthenticationScheme: form
    resource:
      userInfoUri: https://graph.facebook.com/me

spring:
  security:
    oauth2:
      client:
        registration:
          github:
            clientId: 8b0e03b365d0d0e4f6e1
            clientSecret: 27bbced66e0dc6a4700f2c44b8d0c0261410aee9
```