**25Q2 CSOC Manual Audit Test Alumni-Dev August 6**

### Introduction:

As part of the quality assurance process, a manual audit has been initiated to validate the results generated by the automated vulnerability scanning tools. Each finding is being reviewed to determine its accuracy and relevance, with the objective of classifying them as either true positives (valid security issues) or false positives (non-exploitable or misidentified issues).

This step ensures the reliability of the final report by eliminating noise from automated tools and confirming the presence and impact of vulnerabilities through manual testing techniques. The scan is done twice a month

Table of Contents

[Introduction: 2](#_Toc2910)

[2025 Q2 CSOC Manual Penetration test of Test-Portal 4](#_Toc20055)

[Executive Summary 4](#_Toc25547)

[SQL Injection 4](#_Toc28167)

[Content Security Policy allow listed Script resources 7](#_Toc32380)

[Content Security Policy allows un-trusted style execution 9](#_Toc21197)

[Content Security Policy allows click jacking 11](#_Toc19261)

[External Service interaction 13](#_Toc11892)

[Input Returned in response 14](#_Toc22829)

[Frame-able Response (Potential Click jacking) 16](#_Toc28180)

[HTML5 storage manipulation (DOM based) 16](#_Toc31125)

[Robots.txt found 17](#_Toc7203)

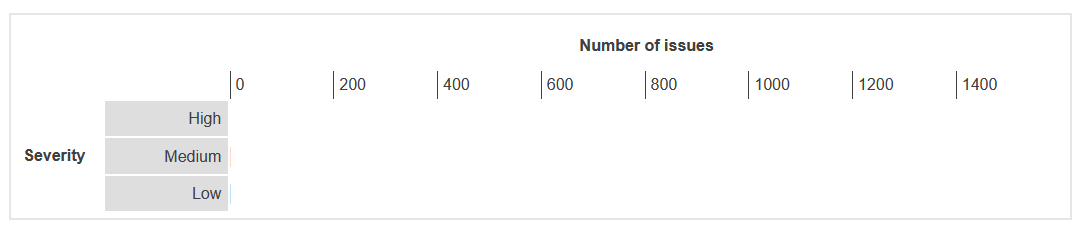
[Cache able HTTPS Response 19](#_Toc14903)

[TLS Certificate 20](#_Toc4081)

### 2025 Q2 CSOC Manual Penetration test of Test-Portal

### Executive Summary

The chart below shows the aggregated numbers of issues identified in each category. Solid colored bars represent issues with a confidence level, and the bars fade as the confidence level falls.



### SQL Injection

|  |  |
| --- | --- |
| Severity | NA |
| Finding Type | False positive |

**Description**

The URL path filename appears to be vulnerable to SQL injection attacks. The payloads 15200588' or 7421=7421-- and 80687611' or 3270=3276-- were each submitted in the URL path filename. These two requests resulted in different responses, indicating that the input is being incorporated into a SQL query in an unsafe way.   
  
Note that automated difference-based tests for SQL injection flaws can often be unreliable and are prone to false positive results. You should manually review the reported requests and responses to confirm whether a vulnerability is actually present. Links Tested

|  |
| --- |
| 1. **/api/employees/870-80859324/personal-data** |

|  |
| --- |
|  |

**Steps:**

1. The request and response from the auto scan is given below

|  |
| --- |
| Request  GET /api/employees/870-80859324/personal-data15200588'%20or%207421%3d7421--%20 HTTP/2 Host: alumni-dev.zalaris.com Cookie: ai\_user=EKPg4hpAQsAf/K5qp/yJ0e|2025-03-25T07:27:51.056Z; \_ga=GA1.1.811876401.1744031047; hubspotutk=cdefa6fdb081482ef354591f2c6802e3; \_ga\_RM4D8JWQH4=GS2.1.s1749467270$o3$g0$t1749467270$j60$l0$h0; \_\_hstc=35319230.cdefa6fdb081482ef354591f2c6802e3.1744031048050.1746019880267.1749467275069.3; ai\_session=ExlzNfRYJmuWFe+L8DI/B1|1754292533683|1754293082557 Sec-Ch-Ua-Platform: "Linux" Authorization: Bearer eyJhbGciOiJSUzI1NiIsImtpZCI6Ilg1ZVhrNHh5b2pORnVtMWtsMll0djhkbE5QNC1jNTdkTzZRR1RWQndhTmsiLCJ0eXAiOiJKV1QifQ..QqcHkWmPOsIDZaxaXIY8yEKNaqmPC7IgVUZEwIw4eWjdE2yHLBGlqdonqy6diDrLhjWC6Mq7dIlSdo91oTZm1C3OmY8EVRPovmKOF5sgOto1suEBxTX61cnPRcOnYM4O4HAKHm3qxZvlKaOEMe2ddyGqj67dSYhhRNK78TZjEcjk7NWJnUiBkAKpzLXGwZi-6TeKupeozhkZgkOPm-4VrMet6bsKEYXLmODO1OZtyaSQ\_ipvp5EfdYAaQimjhuO1Nfagux4wpVSoLnOzfA4x3xRbKJlx0rBZXiMwK2YVFbXhBukEMITQ3CSpHvwz43LZkuDKURNkch7HKvTduiuxnw Sec-Ch-Ua: "Chromium";v="133", "Not(A:Brand";v="99" Sec-Ch-Ua-Mobile: ?0 Request-Id: |29bfa73c12914d6eaf87d206b8245e0a.431d4ef8002942d4 Traceparent: 00-29bfa73c12914d6eaf87d206b8245e0a-431d4ef8002942d4-01 User-Agent: Mozilla/5.0 (X11; Linux x86\_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/133.0.0.0 Safari/537.36 Accept: application/json, text/plain, \*/\* Sec-Fetch-Site: same-origin Sec-Fetch-Mode: cors Sec-Fetch-Dest: empty Referer: https://alumni-dev.zalaris.com/personal-data Accept-Encoding: gzip, deflate, br Accept-Language: en-US,en;q=0.9 Priority: u=1, I  Response  HTTP/2 500 Internal Server Error Server: nginx Date: Mon, 04 Aug 2025 07:45:12 GMT Content-Type: application/json Vary: Origin Vary: Access-Control-Request-Method Vary: Access-Control-Request-Headers X-Xss-Protection: 0 Cache-Control: no-cache, no-store, max-age=0, must-revalidate Pragma: no-cache Expires: 0 X-Frame-Options: DENY Referrer-Policy: same-origin X-Content-Type-Options: nosniff Strict-Transport-Security: max-age=31536000; includeSubDomains Content-Security-Policy: default-src 'self' \*.zalaris.de \*.zalaris.com https://s3.scriptcdn.net/cdn/c7fa7451-6f95-4815-ac32-b8cc2537837a https://login.microsoftonline.com/ https://dzudiostorage.blob.core.windows.net/ \*.b2clogin.com https://pzudiostorage.blob.core.windows.net/ https://northeurope-2.in.applicationinsights.azure.com/v2/track https://northeurope-2.in.applicationinsights.azure.com//v2/track https://js.monitor.azure.com/scripts/b/ai.config.1.cfg.json https://cdn.form.io/ace/1.4.12/mode-html.js https://cdn.form.io/ace/1.4.12/mode-json.js https://cdn.form.io/flatpickr/flatpickr.min.js https://cdn.form.io/ace/1.4.10/ace.js https://cdn.form.io/ace/1.4.10/theme-xcode.js https://cdn.form.io/ace/1.4.10/mode-html.js https://cdn.form.io/ace/1.4.10/mode-json.js https://cdn.form.io/ace/1.4.10/worker-html.js https://cdn.form.io/ace/1.4.10/worker-json.js https://cdn.form.io/flatpickr/flatpickr.min.css https://view.officeapps.live.com/op/embed.aspx https://alumnidev.b2clogin.com/alumnidev.onmicrosoft.com/ https://helpdesk-dev.zalaris.com/ https://helpdesk-test.zalaris.com/ https://alhvbcpn2.accounts.cloud.sap dzudiostorage.blob.core.windows.net https://northeurope-0.in.applicationinsights.azure.com/v2/track https://js.monitor.azure.com/scripts/ https://ar2hdcts7.accounts.cloud.sap; script-src 'self' blob:; img-src 'self' blob: data: dzudiostorage.blob.core.windows.net https://pzudiostorage.blob.core.windows.net/ https://dzudiostorage.blob.core.windows.net/; style-src 'unsafe-inline' 'self' mypayslip-dev.zalaris.com mypayslip.zalaris.com; media-src 'self' https://dzudiostorage.blob.core.windows.net/ https://pzudiostorage.blob.core.windows.net/ https://zudioinnomotics.blob.core.windows.net/; font-src data: 'self'; frame-src 'self' blob: mypayslip.zalaris.com mypayslip-dev.zalaris.com https://dzudiostorage.blob.core.windows.net/ https://view.officeapps.live.com/; frame-ancestors 'self' blob: \*.zalaris.de \*.zalaris.com; form-action 'self'; worker-src 'self' blob:;   {"status":500,"timestamp":"2025-08-04T07:45:12.772151599Z","path":"/api/employees/870-80859324/personal-data15200588'%20or%207421%3d7421--%20","errors":[{"code":"server.error","message":"No static resource api/employees/870-80859324/personal-data15200588'%20or%207421%3d7421--%20."}]} |

The response received is normal . The application checks for the employee number

The employee is not found and server returns 500 internal server error

The alert is considered as False Positive

**WORKLOG**

|  |  |  |  |
| --- | --- | --- | --- |
| When | What | Who | Recommended Action |
| 6 Ausgust 2025 | Manually Tested the SQL injection alert received | CSOC | No action needed |

### Content Security Policy allow listed Script resources

|  |  |
| --- | --- |
| Severity | NA |
| Finding Type | False positive |

**Description**

Content Security Policy (CSP) is a security mechanism designed to mitigate cross-site scripting attacks by disabling dangerous behaviours such as untrusted JavaScript execution. Websites can specify their security policy in a response header or meta tag, enabling fine-grained control over dangerous features like scripts and stylesheets.

**Links Tested**

* [/api/employees/510-00000506](C:/Users/Mohammed Sharoz/Downloads/25Q3 CSOC Automated Audit Alumni August (1).html" \l "2.1)
* [/zalaris-site-logo.svg](C:/Users/Mohammed Sharoz/Downloads/25Q3 CSOC Automated Audit Alumni August (1).html" \l "2.2)

The Content security Policy of Alumni-dev portal is given below

|  |
| --- |
| default-src 'self' \*.zalaris.de \*.zalaris.com https://s3.scriptcdn.net/cdn/c7fa7451-6f95-4815-ac32-b8cc2537837a https://login.microsoftonline.com/ https://dzudiostorage.blob.core.windows.net/ \*.b2clogin.com https://pzudiostorage.blob.core.windows.net/ https://northeurope-2.in.applicationinsights.azure.com/v2/track https://northeurope-2.in.applicationinsights.azure.com//v2/track https://js.monitor.azure.com/scripts/b/ai.config.1.cfg.json https://cdn.form.io/ace/1.4.12/mode-html.js https://cdn.form.io/ace/1.4.12/mode-json.js https://cdn.form.io/flatpickr/flatpickr.min.js https://cdn.form.io/ace/1.4.10/ace.js https://cdn.form.io/ace/1.4.10/theme-xcode.js https://cdn.form.io/ace/1.4.10/mode-html.js https://cdn.form.io/ace/1.4.10/mode-json.js https://cdn.form.io/ace/1.4.10/worker-html.js https://cdn.form.io/ace/1.4.10/worker-json.js https://cdn.form.io/flatpickr/flatpickr.min.css https://view.officeapps.live.com/op/embed.aspx https://alumnidev.b2clogin.com/alumnidev.onmicrosoft.com/ https://helpdesk-dev.zalaris.com/ https://helpdesk-test.zalaris.com/ https://alhvbcpn2.accounts.cloud.sap dzudiostorage.blob.core.windows.net https://northeurope-0.in.applicationinsights.azure.com/v2/track https://js.monitor.azure.com/scripts/ https://ar2hdcts7.accounts.cloud.sap; script-src 'self' blob:; img-src 'self' blob: data: dzudiostorage.blob.core.windows.net https://pzudiostorage.blob.core.windows.net/ https://dzudiostorage.blob.core.windows.net/; style-src 'unsafe-inline' 'self' mypayslip-dev.zalaris.com mypayslip.zalaris.com; media-src 'self' https://dzudiostorage.blob.core.windows.net/ https://pzudiostorage.blob.core.windows.net/ https://zudioinnomotics.blob.core.windows.net/; font-src data: 'self'; frame-src 'self' blob: mypayslip.zalaris.com mypayslip-dev.zalaris.com https://dzudiostorage.blob.core.windows.net/ https://view.officeapps.live.com/; frame-ancestors 'self' blob: \*.zalaris.de \*.zalaris.com; form-action 'self'; worker-src 'self' blob:; |

**Steps:**  
Capture the content security policy of alumni-dev.zalaris.com

The script resources are not public domains but controlled domains

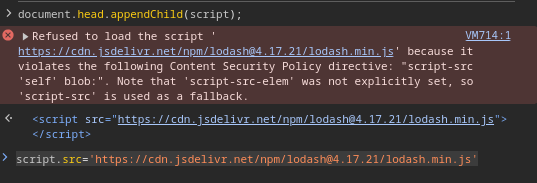
This prevents any scripts from being executed from a third party domain within the application

Test by fetching a js script from the following domain for testing : ‘https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js’

The console logs also confirms the same , screenshot for the same is attached below

|  |
| --- |
| The following command can be used to execute third party scripts  let script= document.createElement('script');  script.src='https://cdn.jsdelivr.net/npm/lodash@4.17.21/lodash.min.js'  script.onload=()=>console.log('Lodash Version:',\_.VERSION);  document.head.appendChild(script); |

the screenshot shows - that the applications content security policy detects the script fetching and prevents it



The alert can be considered as false positive

**WORKLOG**

|  |  |  |  |
| --- | --- | --- | --- |
| When | What | Who | Recommended Action |
| 6 August 2025 | Manually tested the alert ‘content security policy allow-listed script resources’ | CSOC | No action Needed the alert is considered as false positive |

### Content Security Policy allows un-trusted style execution

|  |  |
| --- | --- |
| Severity | NA |
| Finding Type | False positive |

**Description**

Content Security Policy (CSP) is a security mechanism designed to mitigate cross-site scripting attacks by disabling dangerous behaviours such as untrusted JavaScript execution. Websites can specify their security policy in a response header or meta tag, enabling fine-grained control over dangerous features like scripts and stylesheets.

The following links are tested

* [/api/employees/510-00000506](C:/Users/Mohammed Sharoz/Downloads/25Q3 CSOC Automated Audit Alumni August (1).html" \l "3.1)
* [/zalaris-site-logo.svg](C:/Users/Mohammed Sharoz/Downloads/25Q3 CSOC Automated Audit Alumni August (1).html" \l "3.2)

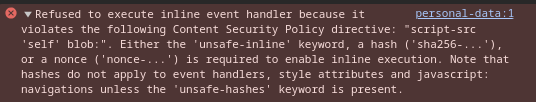
**Steps:**  
The following needs to be tested

* Inline script via innerHTML injection
* eval() execution test
* Function Constructor (also must blocked by unsafe-eval)
* Data: URL Script Injection

**Inner Script via innerHTML injection**

Console command to test for the inner script injection

|  |
| --- |
| let div=document.createElement('div');  div.innerHTML ='<img src=x onerror="alert(XSS via inner HTML)">'  document.body.appendChild(div); |

The CSP is not accepting the script execution screenshot for the same is attached below

**Function constructor (**must be blocked by unsafe-eval**)**

Console command to test for function constructor

|  |
| --- |
| try {let fn =new Function("alert('Function constructor allowed')");fn();}catch(e){console.log("Function constructor blocked by CSP:",e.message);} |

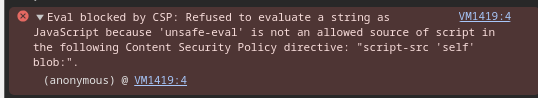
The CSP prevents the function constructor , screenshot for the same is attached below

**Eval () Execution test**

Console command to test for eval() execution

|  |
| --- |
| try {eval(“alert(‘Eval allowed - unsafe-eval is enabled’)”);}catch (e){console.error(“Eval blocked by CSP”,e.message);} |

The eval() function is not executed within the console screenshot for the same is attached below



**WORKLOG**

|  |  |  |  |
| --- | --- | --- | --- |
| When | What | Who | Recommended Action |
| 6 August 2025 | Manually Tested Content Security Policy Allows un-trusted Script execution alert | CSOC | No action needed the alert is false positive |

### Content Security Policy allows click jacking

|  |  |
| --- | --- |
| Severity | NA |
| Finding Type | False positive |

**Description**

Content Security Policy (CSP) is a security mechanism designed to mitigate cross-site scripting attacks by disabling dangerous behaviour such as un-trusted JavaScript execution. Websites can specify their security policy in a response header or meta tag, enabling fine-grained control over dangerous features like scripts and style-sheets.

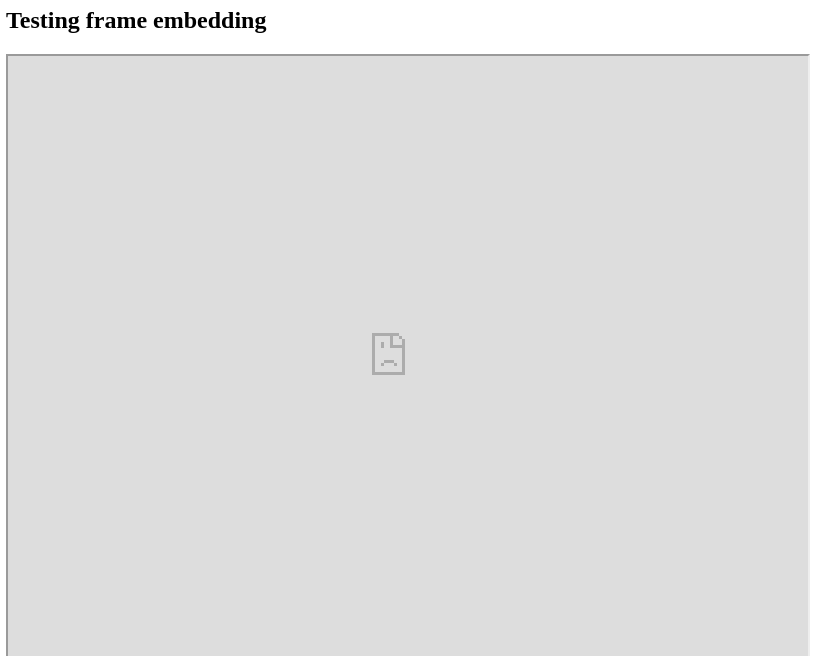
**Links Tested**

* /zalaris-site-logo.svg

**Steps:**  
Capture the content security policy of alumni-dev.zalaris.com

Analyze the CSP for frame-ancestors directive

The application prevents the iframe from being loaded. Screenshot for the same is given below



The console logs also confirms that the frame-ancestors block the site from being loaded inside the iframe

|  |
| --- |
| Refused to frame 'https://alumni-dev.zalaris.com/' because an ancestor violates the following Content Security Policy directive: "frame-ancestors 'self' blob: \*.zalaris.de \*.zalaris.com". |

**WORKLOG**

|  |  |  |  |
| --- | --- | --- | --- |
| When | What | Who | Recommended Action |
| 6 August 2025 | Manually tested the alert ‘Content security policy allows Click Jacking’ | CSOC | No action needed the alert is false positive |

### External Service interaction

|  |  |
| --- | --- |
| Severity | NA |
| Finding Type | False positive |

**Description**

It is possible to induce the application to perform server-side DNS lookups of arbitrary domain names.  
The payload **39aeo91tlr5j4gq2a21tm0rylprif8312psci07.oastify.com** was submitted in the SSL SNI value and the HTTP Host header.The application performed a DNS lookup of the specified domain.

**Steps**

The request is made by the application to collaborator URL and is not a valid test

The request and the response for the same are given below

|  |
| --- |
| **Request**  GET / HTTP/1.1 Host: 39aeo91tlr5j4gq2a21tm0rylprif8312psci07.oastify.com Pragma: no-cache Cache-Control: no-cache, no-transform Connection: close  **Response**  HTTP/1.1 200 OK Server: nginx Date: Mon, 04 Aug 2025 07:45:22 GMT Content-Type: text/html; charset=utf-8 Content-Length: 0 Last-Modified: Fri, 23 Sep 2022 12:28:13 GMT Connection: close ETag: "632da65d-0" Accept-Ranges: bytes |

The alert is considered as flase positive

**WORKLOG**

|  |  |  |  |
| --- | --- | --- | --- |
| When | What | Who | Recommended Action |
| 6 August 2025 | Manually Tested alert ‘External service interaction’ | CSOC | No action needed the alert is false positive |

### Input Returned in response

|  |  |
| --- | --- |
| Severity | NA |
| Finding Type | False positive |

**Description**

Reflection of input arises when data is copied from a request and echoed into the application's immediate response.

Input being returned in application responses is not a vulnerability in its own right. However, it is a prerequisite for many client-side vulnerabilities, including cross-site scripting, open redirection, content spoofing, and response header injection. Additionally, some server-side vulnerabilities such as SQL injection are often easier to identify and exploit when input is returned in responses. In applications where input retrieval is rare and the environment is resistant to automated testing (for example, due to a web application firewall), it might be worth subjecting instances of it to focused manual testing.

**Links Tested**

* [/api/content/news [URL path filename]](C:/Users/Mohammed Sharoz/Downloads/25Q3 CSOC Automated Audit Alumni August (1).html" \l "6.1)
* [/api/content/news [URL path folder 2]](C:/Users/Mohammed Sharoz/Downloads/25Q3 CSOC Automated Audit Alumni August (1).html" \l "6.2)
* [/api/employees/510-00000506 [URL path filename]](C:/Users/Mohammed Sharoz/Downloads/25Q3 CSOC Automated Audit Alumni August (1).html" \l "6.3)
* [/api/employees/510-00000506 [URL path folder 2]](C:/Users/Mohammed Sharoz/Downloads/25Q3 CSOC Automated Audit Alumni August (1).html" \l "6.4)
* [/api/employees/510-00000506/documents [URL path filename]](C:/Users/Mohammed Sharoz/Downloads/25Q3 CSOC Automated Audit Alumni August (1).html" \l "6.5)
* [/api/employees/510-00000506/documents [URL path folder 2]](C:/Users/Mohammed Sharoz/Downloads/25Q3 CSOC Automated Audit Alumni August (1).html" \l "6.6)
* [/api/employees/510-00000506/documents [URL path folder 3]](C:/Users/Mohammed Sharoz/Downloads/25Q3 CSOC Automated Audit Alumni August (1).html" \l "6.7)
* [/api/employees/510-00000506/personal-data [URL path filename]](C:/Users/Mohammed Sharoz/Downloads/25Q3 CSOC Automated Audit Alumni August (1).html" \l "6.8)
* [/api/employees/510-00000506/personal-data [URL path folder 2]](C:/Users/Mohammed Sharoz/Downloads/25Q3 CSOC Automated Audit Alumni August (1).html" \l "6.9)
* [/api/employees/510-00000506/personal-data [URL path folder 3]](C:/Users/Mohammed Sharoz/Downloads/25Q3 CSOC Automated Audit Alumni August (1).html" \l "6.10)
* [/api/employees/870-80859324/payroll-results [URL path filename]](C:/Users/Mohammed Sharoz/Downloads/25Q3 CSOC Automated Audit Alumni August (1).html" \l "6.11)
* [/api/employees/870-80859324/payroll-results [URL path folder 2]](C:/Users/Mohammed Sharoz/Downloads/25Q3 CSOC Automated Audit Alumni August (1).html" \l "6.12)
* [/api/employees/870-80859324/payroll-results [URL path folder 3]](C:/Users/Mohammed Sharoz/Downloads/25Q3 CSOC Automated Audit Alumni August (1).html" \l "6.13)
* [/api/employees/870-80859324/personal-data [URL path filename]](C:/Users/Mohammed Sharoz/Downloads/25Q3 CSOC Automated Audit Alumni August (1).html" \l "6.14)
* [/api/employees/870-80859324/personal-data [URL path folder 2]](C:/Users/Mohammed Sharoz/Downloads/25Q3 CSOC Automated Audit Alumni August (1).html" \l "6.15)
* [/api/employees/870-80859324/personal-data [URL path folder 3]](C:/Users/Mohammed Sharoz/Downloads/25Q3 CSOC Automated Audit Alumni August (1).html" \l "6.16)
* [/api/employees/employment/history [URL path filename]](C:/Users/Mohammed Sharoz/Downloads/25Q3 CSOC Automated Audit Alumni August (1).html" \l "6.17)
* [/api/employees/employment/history [URL path folder 2]](C:/Users/Mohammed Sharoz/Downloads/25Q3 CSOC Automated Audit Alumni August (1).html" \l "6.18)
* [/api/employees/employment/history [URL path folder 3]](C:/Users/Mohammed Sharoz/Downloads/25Q3 CSOC Automated Audit Alumni August (1).html" \l "6.19)

**Steps:**

Analyzing the requests the response are either 500 internal Server Error , 400 bad Request , 401 Unauthorized

Also the inputs are sanitized and reflected as strings and not injected conetnt type

The response is given below

|  |
| --- |
| HTTP/2 500 Internal Server Error Server: nginx Date: Mon, 04 Aug 2025 07:45:47 GMT Content-Type: application/json Vary: Origin Vary: Access-Control-Request-Method Vary: Access-Control-Request-Headers X-Xss-Protection: 0 Cache-Control: no-cache, no-store, max-age=0, must-revalidate Pragma: no-cache Expires: 0 X-Frame-Options: DENY Referrer-Policy: same-origin X-Content-Type-Options: nosniff Strict-Transport-Security: max-age=31536000; includeSubDomains Content-Security-Policy: default-src 'self' \*.zalaris.de \*.zalaris.com https://s3.scriptcdn.net/cdn/c7fa7451-6f95-4815-ac32-b8cc2537837a https://login.microsoftonline.com/ https://dzudiostorage.blob.core.windows.net/ \*.b2clogin.com https://pzudiostorage.blob.core.windows.net/ https://northeurope-2.in.applicationinsights.azure.com/v2/track https://northeurope-2.in.applicationinsights.azure.com//v2/track https://js.monitor.azure.com/scripts/b/ai.config.1.cfg.json https://cdn.form.io/ace/1.4.12/mode-html.js https://cdn.form.io/ace/1.4.12/mode-json.js https://cdn.form.io/flatpickr/flatpickr.min.js https://cdn.form.io/ace/1.4.10/ace.js https://cdn.form.io/ace/1.4.10/theme-xcode.js https://cdn.form.io/ace/1.4.10/mode-html.js https://cdn.form.io/ace/1.4.10/mode-json.js https://cdn.form.io/ace/1.4.10/worker-html.js https://cdn.form.io/ace/1.4.10/worker-json.js https://cdn.form.io/flatpickr/flatpickr.min.css https://view.officeapps.live.com/op/embed.aspx https://alumnidev.b2clogin.com/alumnidev.onmicrosoft.com/ https://helpdesk-dev.zalaris.com/ https://helpdesk-test.zalaris.com/ https://alhvbcpn2.accounts.cloud.sap dzudiostorage.blob.core.windows.net https://northeurope-0.in.applicationinsights.azure.com/v2/track https://js.monitor.azure.com/scripts/ https://ar2hdcts7.accounts.cloud.sap; script-src 'self' blob:; img-src 'self' blob: data: dzudiostorage.blob.core.windows.net https://pzudiostorage.blob.core.windows.net/ https://dzudiostorage.blob.core.windows.net/; style-src 'unsafe-inline' 'self' mypayslip-dev.zalaris.com mypayslip.zalaris.com; media-src 'self' https://dzudiostorage.blob.core.windows.net/ https://pzudiostorage.blob.core.windows.net/ https://zudioinnomotics.blob.core.windows.net/; font-src data: 'self'; frame-src 'self' blob: mypayslip.zalaris.com mypayslip-dev.zalaris.com https://dzudiostorage.blob.core.windows.net/ https://view.officeapps.live.com/; frame-ancestors 'self' blob: \*.zalaris.de \*.zalaris.com; form-action 'self'; worker-src 'self' blob:;   {"status":500,"timestamp":"2025-08-04T07:45:47.747154237Z","path":"/api/employeeshp666tsszd/510-00000506","errors":[{"code":"server.error","message":"No static resource api/employeeshp666tsszd/510-00000506."}]} |

Scripts injected at the endpoint are also sanitized

**WORKLOG**

|  |  |  |  |
| --- | --- | --- | --- |
| When | What | Who | Recommended Action |
| 6 August 2025 | Manually Tested Input Returned in response | CSOC | No action needed the alert is false positive |

### Frame-able Response (Potential Click jacking)

|  |  |
| --- | --- |
| Severity | NA |
| Finding Type | False positive |

Duplicate alert - already tested and not vulnerable during CSP - click jacking allowed

**WORKLOG**

|  |  |  |  |
| --- | --- | --- | --- |
| When | What | Who | Recommended Action |
| 6 August 2025 | Manually tested alert ‘Frame-able Response (Potential Click Jacking)’ | CSOC | No action needed . The alert is falase positive |

### HTML5 storage manipulation (DOM based)

|  |  |
| --- | --- |
| Severity | NA |
| Finding Type | False positive |

**Description**

DOM-based vulnerabilities arise when a client-side script reads data from a controllable part of the DOM (for example, the URL) and processes this data in an unsafe way.

HTML5 storage manipulation arises when a script stores controllable data in the HTML5 storage of the web browser (either localStorage or sessionStorage). An attacker may be able to use this behavior to construct a URL that, if visited by another application user, will cause the user's browser to store attacker-controllable data.

This behavior does not in itself constitute a security vulnerability. However, if the application later reads the data back from storage and processes it in an unsafe way, then an attacker may be able to leverage the storage mechanism to deliver other DOM-based attacks, such as cross-site scripting and JavaScript injection.

Burp Suite automatically identifies this issue using dynamic and static code analysis. Static analysis can lead to false positives that are not actually exploitable. If Burp Scanner has not provided any evidence resulting from dynamic analysis, you should review the relevant code and execution paths to determine whether this vulnerability is indeed present, or whether mitigations are in place that would prevent exploitation

**Links Tested**

* [/payroll-data](C:/Users/Mohammed Sharoz/Downloads/25Q3 CSOC Automated Audit Alumni August (1).html" \l "8.1)
* [/payroll-data](C:/Users/Mohammed Sharoz/Downloads/25Q3 CSOC Automated Audit Alumni August (1).html" \l "8.2)
* [/zalaris-site-logo.svg](C:/Users/Mohammed Sharoz/Downloads/25Q3 CSOC Automated Audit Alumni August (1).html" \l "8.3)
* [/zalaris-site-logo.svg](C:/Users/Mohammed Sharoz/Downloads/25Q3 CSOC Automated Audit Alumni August (1).html" \l "8.4)

**Steps:**

The alert is triggered since the application passes information to Azure insights

The value that is injected into the source is given below

|  |
| --- |
| {"item":"{\"time\":\"2025-08-04T07:41:26.784Z\",\"iKey\":\"9b454807-9d03-4ea0-ae51-041c8d9edf86\",\"name\":\"Microsoft.ApplicationInsights.9b4548079d034ea0ae51041c8d9edf86.RemoteDependency\",\ |

This is not obtained as from any user controlled inputs that can change the HTML content and has no impact on the HTML page that is rendered by the user

**WORKLOG**

|  |  |  |  |
| --- | --- | --- | --- |
| When | What | Who | Recommended Action |
| 6 August 2025 | Manually tested the alert ‘HTML5 Storage Manipulation’ | CSOC | No action needed the alert is due to data passed on to azure insights value. The alert can be considered as false positive |

### Robots.txt found

|  |  |
| --- | --- |
| Severity | NA |
| Finding Type | False positive |

**Description**

The request appears to be vulnerable to cross-site request forgery (CSRF) attacks against authenticated users.

**Links Tested**

* /robots.txt

**Steps:**  
Access the page /robots.txt

The Robots file content is given below

|  |
| --- |
| # https://www.robotstxt.org/robotstxt.html  User-agent: \*  Disallow: |

The file has allowed all the user agents and the Disallow parameter is set to ‘empty’ string.

This configuration is not secure , secure configuration is given below

|  |
| --- |
| User-agent: \*  Disallow: /admin/  Disallow: /login/  Disallow: /config/  Disallow: /internal/  Disallow: /private/  Disallow: /tmp/  Disallow: /backup/  Disallow: /cgi-bin/  Allow: / |

**WORKLOG**

|  |  |  |  |
| --- | --- | --- | --- |
| When | What | Who | Recommended Action |
| 6 August 2025 | Manually Tested the alert ‘robots.txt’ file found’ | CSOC | No issues with the current robots file configuration.No action needed |

### Cache able HTTPS Response

|  |  |
| --- | --- |
| Severity | NA |
| Finding Type | False positive |

**Description**

Unless directed otherwise, browsers may store a local cached copy of content received from web servers. Some browsers, including Internet Explorer, cache content accessed via HTTPS. If sensitive information in application responses is stored in the local cache, then this may be retrieved by other users who have access to the same computer at a future time.

**Links Tested**

* [/manifest.json](C:/Users/Mohammed Sharoz/Downloads/25Q3 CSOC Automated Audit Alumni August (1).html" \l "10.1)
* [/zalaris-site-logo.svg](C:/Users/Mohammed Sharoz/Downloads/25Q3 CSOC Automated Audit Alumni August (1).html" \l "10.2)

**Steps:**  
The pages that are reported under Cache-able HTTPs response are publicly accessible pages

The alert is considered as false positive

**WORKLOG**

|  |  |  |  |
| --- | --- | --- | --- |
| When | What | Who | Recommended Action |
| 6 August 2025 | Manually Tested cache-able HTTPS Response | CSOC | No action needed the alert is false positive |

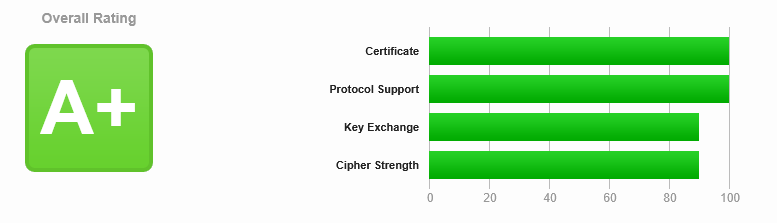
### TLS Certificate

|  |  |
| --- | --- |
| Severity | NA |
| Finding Type | Informational Alert |

**Description**

The server presented a valid, trusted TLS certificate. Additional security tests have been conducted to confirm that other security controls and configurations are in place

**Overall Score**



**Certificate Validity**

SSL/TLS certificate validation confirmed the certificate is active and within its validity period as of he test date



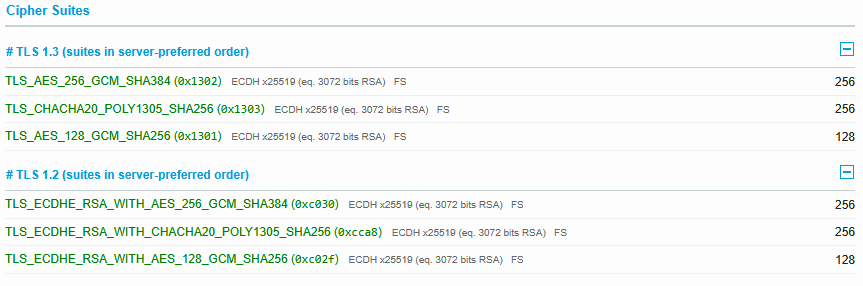
**TLS protocols supported**

The target supports TLS protocols supported are 1.2 and 1.3 which are considered as secure for current use



**Cipher Suites supported**

The following cipher suites are used and are secure



**WORKLOG**

|  |  |  |  |
| --- | --- | --- | --- |
| When | What | Who | Recommended Action |
| 6 August 2025 | Manually tested the TLS certificate | CSOC | Informational alert , no action needed |