Finding Name: Insufficient Token Entropy

Name	Team	Role	Project	Quality Assurance	Is this a re-tested Finding?
Andrew Duffy	AppAtack	Pen-Tester	OnTrack Web App	Filipe Oliveira	No

Was this Finding Successful?
Yes

Finding Description

An **Insufficient Token Entropy vulnerability** was discovered in the OnTrack web application. Session or Authentication Tokens with insufficient entropy have reduced randomness and present opportunities for threat actors to guess valid tokens through brute force techniques. The Entropy value is the indicator of how resilient a token is to this brute force guessing.

Analysis was conducted using the Burp Suite Sequencer tool, a free tool that is readily available, used for testing token randomness. The Sequencer tool sends thousands of authentication requests to the application and captures the returned authentication token. In this instance 10000 tokens were captured, which represents a significant sample size. The tool will assess the token to determine its actual randomness at a bit level and identify patterns or predictability with the final output or entropy estimate as an indication of its resilience to being brute forced.

As a result, the token randomness was assessed as being **reasonable** with an entropy of approximately **38 bits**. Entropy represents the randomness or unpredictability of a token, and higher entropy means more possible combinations an attacker must guess. The Open Worldwide Application Security Project (OWASP) recommends that session identifiers have at least 64 bits of entropy to prevent brute force attacks. According to OWASP, session IDs with 64 bits of entropy, could take a threat actor approximately 585 years to successfully guess a valid session ID. [1]

The simplistic nature of brute force attacks and ready availability of the tools, tempered with the relative time it would stake to conduct a successful attack, despite teh lower entropy, the likelihood of this vulnerability being exploited is **Moderate**. Additionally, compensating controls such as token invalidation techniques limit the impact of a successfully brute forced token, reducing the impact to **Very Minor**, resulting in an overall risk rating of **Low**.

Further analysis with a larger sample size (approximately 20,000 tokens) will likely result in an even lower assessed entropy and randomness score.

[1] OWASP. Session Management. https://cheatsheetseries.owasp.org/cheatsheets/Session_Management_Cheat_Sheet.html

Risk Rating - Low

Impact: **Very Minor**
Likelihood: **Moderate**

Impact values				
Very Minor	**Minor**	**Significant**	**Major**	**Severe**
Risk that holds little to no impact. Will not cause damage and regular activity can continue.	Risk that holds minor form of impact, but not significant enough to be of threat. Can cause some damage but not enough to impede regular activity.	Risk that holds enough impact to be somewhat of a threat. Will cause damage that can impede regular activity but will be able to run normally.	Risk that holds major impact to be of threat. Will cause damage that will impede regular activity and will not be able to run normally.	Risk that holds severe impact and is a threat. Will cause critical damage that can cease activity to be run.

Likelihood				
Rare	**Unlikely**	**Moderate**	**High**	**Certain**
Event may occur and/or if it did, it happens in specific circumstances.	Event could occur occasionally and/or could happen (at some point)	Event may occur and/or happens.	Event occurs at times and/or probably happens a lot.	Event is occurring now and/or happens frequently.

Business Impact

While the test conducted doesn't directly exploit a token vulnerability, the results are indicative of an underlying vulnerability. The presence of session tokens with low entropy provides an opportunity for threat actors to exploit and opportunity to obtain a valid session token through brute forcing techniques. A successful compromise could expose user accounts or sensitive data through authenticated requests. Although compensating session management controls are in place to limit the exposure to this vulnerability, weaknesses in token generation below the industry recommendations may undermine user trust in the applications security.

Affected Assets

OnTrack Web Applicaiton Authentication/Session Token implementation. Any API, endpoint of functionality that relies on a valid authentication token is potentially vulnerable.

Evidence

Step 1: Configure Burp Suite and Firefox

1. Configure Firefox to route all traffic through the Burp Suite proxy, ensuring that requests made to the application are processed through Burp and can be reviewed in the HTTP history.

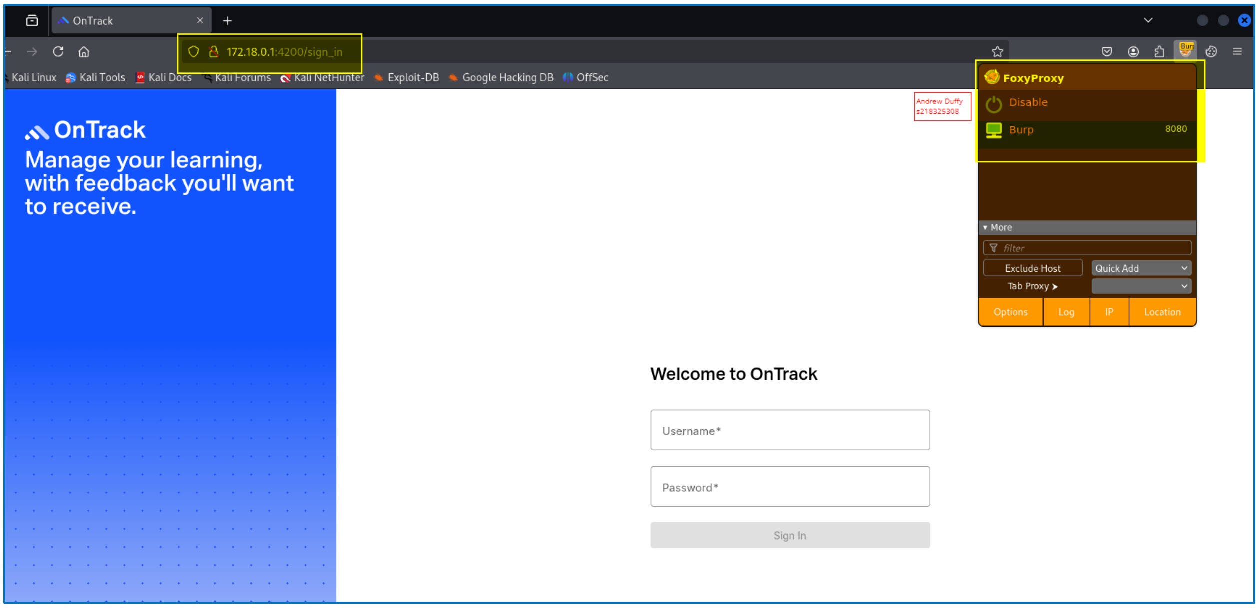
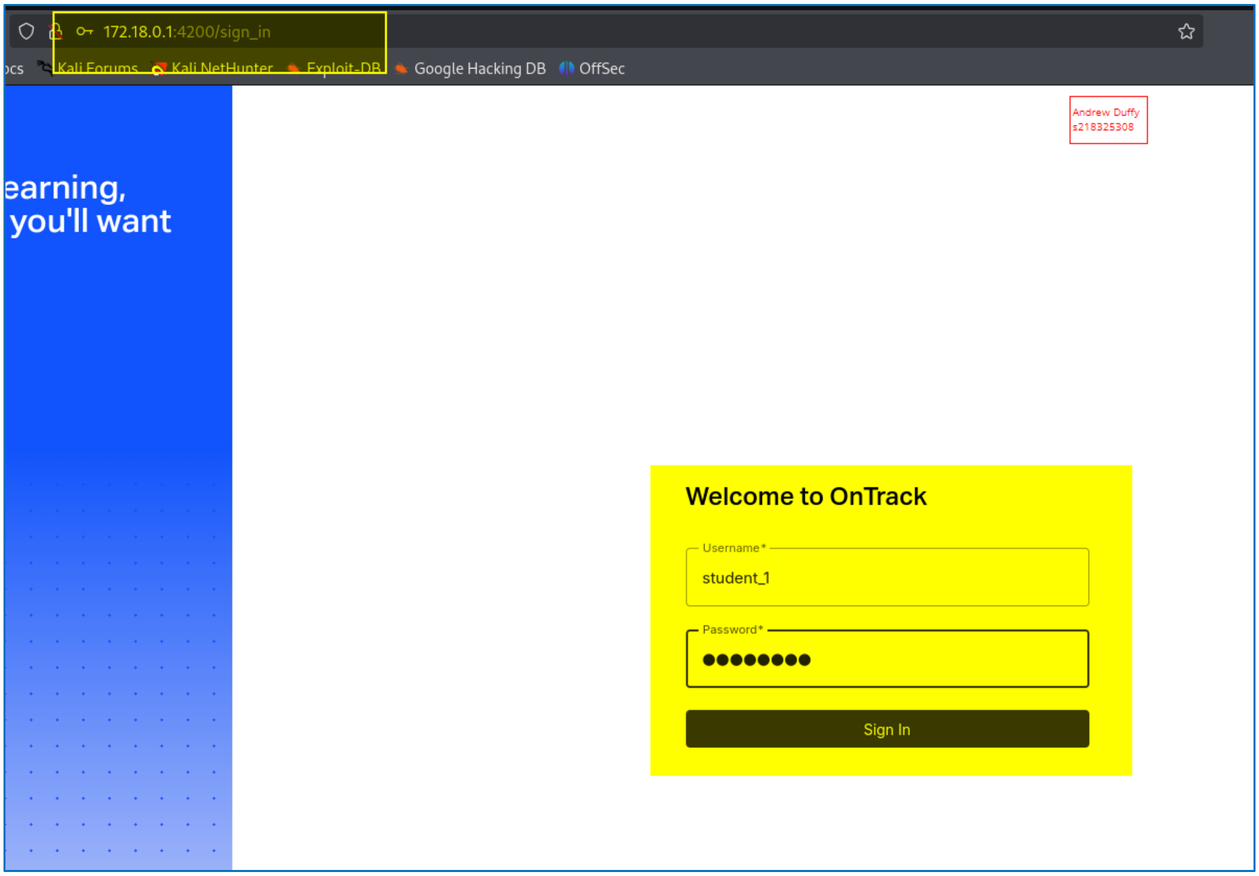


Figure – Firefox configured to route traffic through Burp Suite

Step 2: Login to the application review HTTP history in Burp Suite

1. Login to the application as student_1.



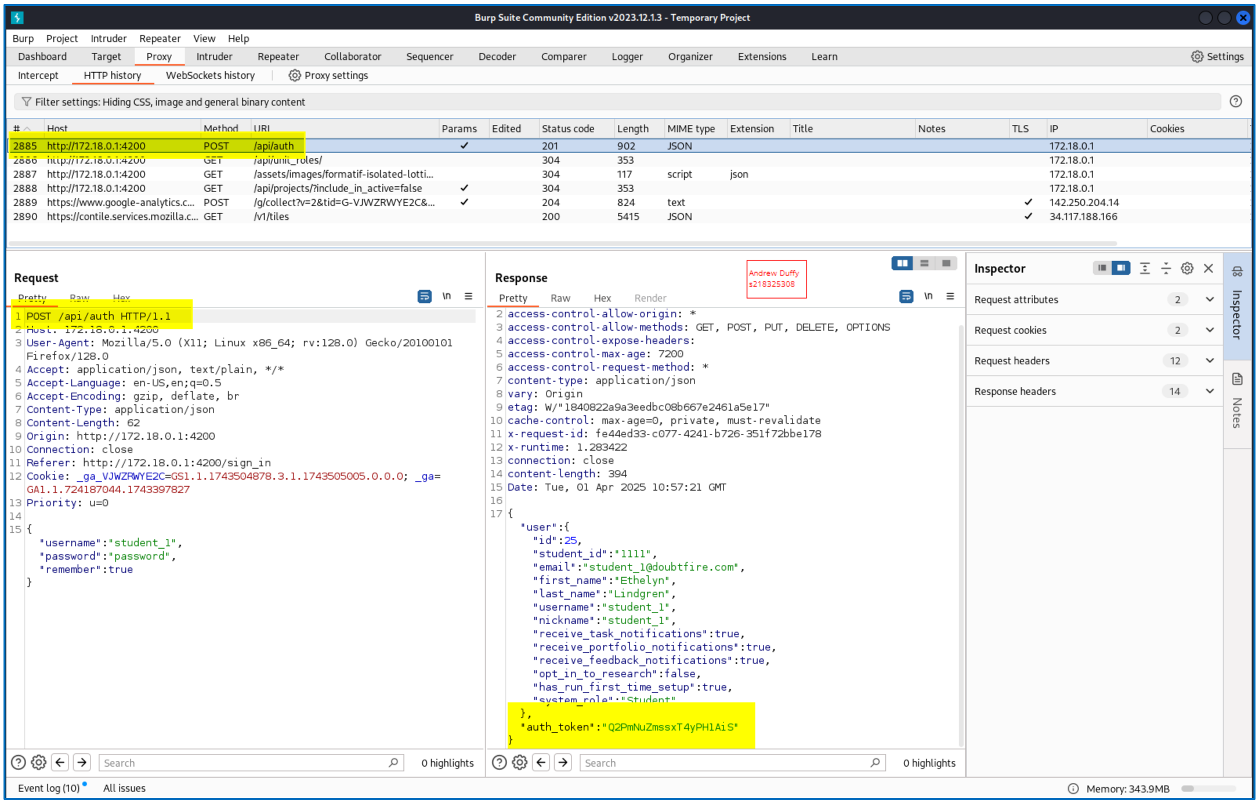
Welcome to OnTrack

Username* student_1

Password* ●●●●●●●●

Figure – Login to application

2. Go to Burp Suite and navigate to Proxy, HTTP history and identify the HTTP request that returns an authorisation token.


Burp Suite Community Edition v2023.12.1.3 - Temporary Project
Burp Project Intruder Repeater View Help
Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions Learn Settings
Intercept HTTP history WebSockets history Proxy settings
Filter settings: Hiding CSS, image and general binary content
Request
Pretty Raw Hex
Response
Pretty Raw Hex Render
Andrew Duffy
s218325308
Inspector
Request attributes 2
Request cookies 2
Request headers 12
Response headers 14
Inspector
Notes
Search
0 highlights
Search
0 highlights
Event log (10)
All issues
Memory: 343.9MB

#	Host	Method	URL	Params	Edited	Status code	Length	MIME type	Extension	Title	Notes	TLS	IP	Cookies
2885	http://172.18.0.1:4200	POST	/api/auth	✓		201	902	JSON					172.18.0.1	
2886	http://172.18.0.1:4200	GET	/api/unit_roles/			304	353						172.18.0.1	
2887	http://172.18.0.1:4200	GET	/assets/images/formatif-isolated-lotti...			304	117	script	json				172.18.0.1	
2888	http://172.18.0.1:4200	GET	/api/projects/?include_in_active=false	✓		304	353						172.18.0.1	
2889	https://www.google-analytics.c...	POST	/g/collect?v=2&tid=G-VJWZRWYE2C&...	✓		204	824	text				✓	142.250.204.14	
2890	https://contile.services.mozilla.c...	GET	/v1/tiles			200	5415	JSON				✓	34.117.188.166	

Request

```
POST /api/auth HTTP/1.1
Host: 172.18.0.1:4200
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101
Firefox/128.0
Accept: application/json, text/plain, */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Content-Type: application/json
Content-Length: 62
Origin: http://172.18.0.1:4200
Connection: close
Referer: http://172.18.0.1:4200/sign_in
Cookie: _ga_VJWZRWYE2C=GS1.1.1743504878.3.1.1743505005.0.0.0; _ga=
GA1.1.724187044.1743397827
Priority: u=0

{
  "username":"student_1",
  "password":"password",
  "remember":true
}
```

Response

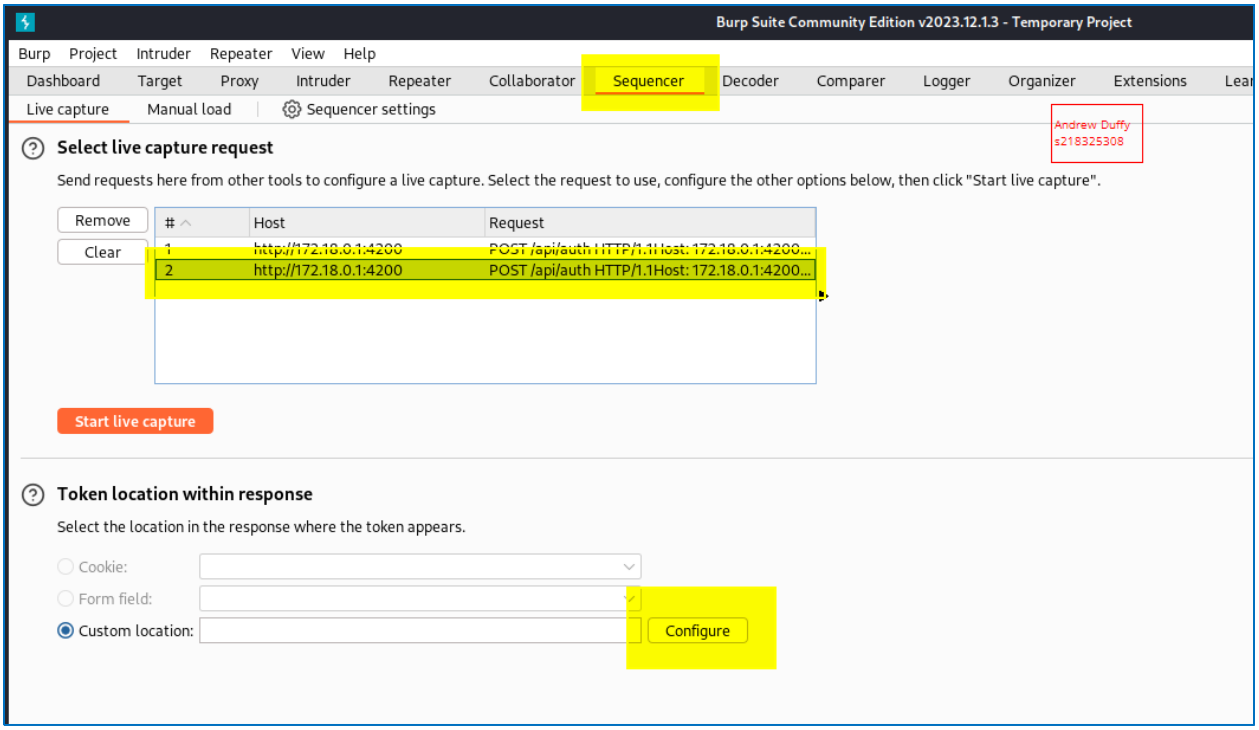
```
access-control-allow-origin: *
access-control-allow-methods: GET, POST, PUT, DELETE, OPTIONS
access-control-expose-headers:
access-control-max-age: 7200
access-control-request-method: *
content-type: application/json
vary: Origin
etag: W/"1840822a9a3eedbc08b667e2461a5e17"
cache-control: max-age=0, private, must-revalidate
x-request-id: fe44ed33-c077-4241-b726-351f72bbe178
x-runtime: 1.283422
connection: close
content-length: 394
Date: Tue, 01 Apr 2025 10:57:21 GMT

{
  "user":{
    "id":25,
    "student_id":"1111",
    "email":"student_1@doubtfire.com",
    "first_name":"Ethelyn",
    "last_name":"Lindgren",
    "username":"student_1",
    "nickname":"student_1",
    "receive_task_notifications":true,
    "receive_portfolio_notifications":true,
    "receive_feedback_notifications":true,
    "opt_in_to_research":false,
    "has_run_first_time_setup":true,
    "system_role":"Student"
  },
  "auth_token":"Q2PmNuZmssxT4yPH1AiS"
}
```

Figure – Identifying the authorisation token request / response

Step 3: Send authorisation request to Burp Suit Sequencer tool

1. Right click within the ‘Request’ box and select ‘Send to Sequencer’ in the pop up menu. Then navigate to the Sequencer tab and click on Configure to set the token location for the tool.


Burp Suite Community Edition v2023.12.1.3 - Temporary Project
Burp Project Intruder Repeater View Help
Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions Lear
Live capture Manual load Sequencer settings
Andrew Duffy
s218325308

Select live capture request

Send requests here from other tools to configure a live capture. Select the request to use, configure the other options below, then click "Start live capture".

#	Host	Request
1	http://172.18.0.1:4200	POST /api/auth HTTP/1.1Host: 172.18.0.1:4200...
2	http://172.18.0.1:4200	POST /api/auth HTTP/1.1Host: 172.18.0.1:4200...

Token location within response

Select the location in the response where the token appears.

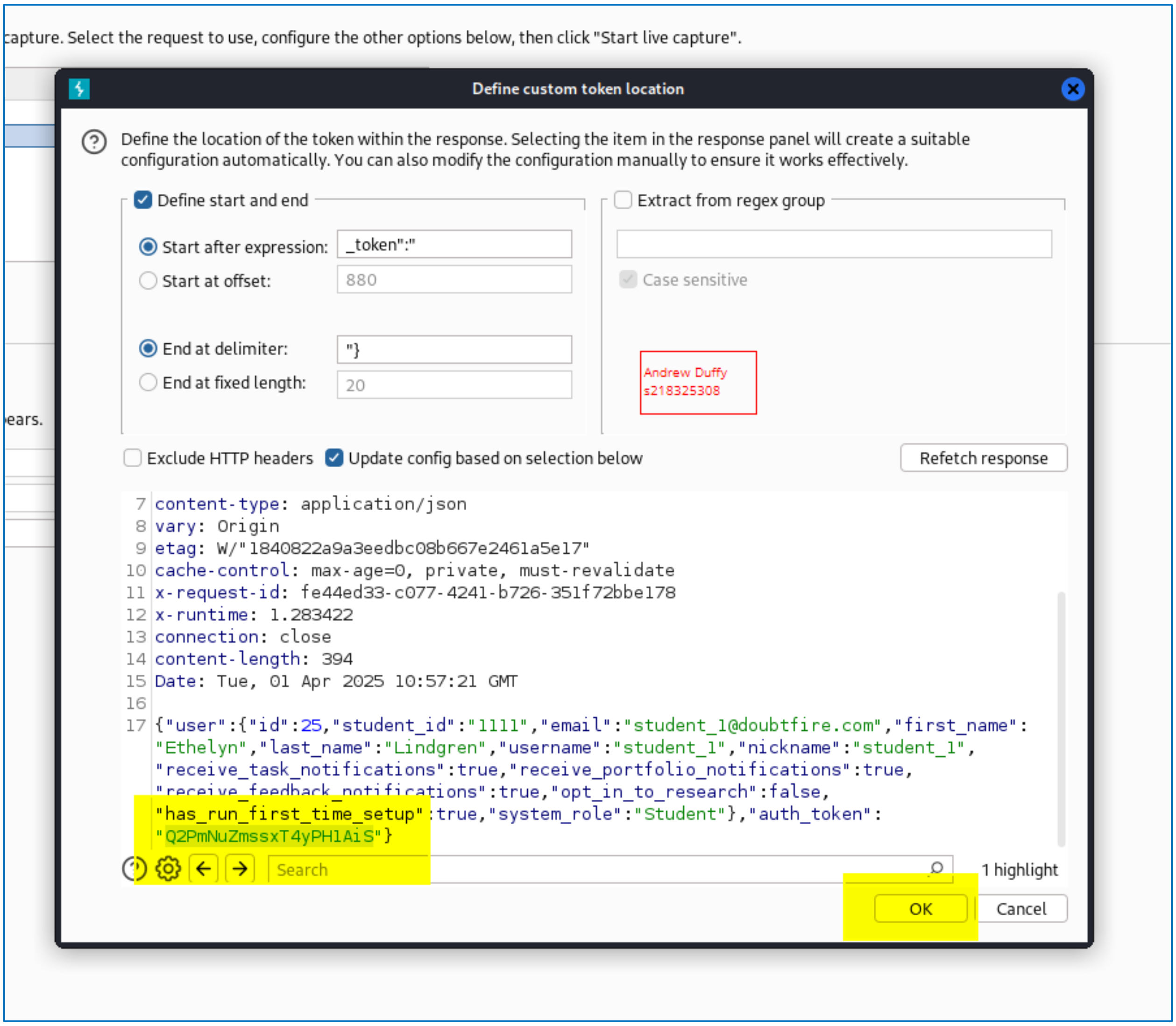
○ Cookie:

○ Form field:

● Custom location:

Figure – Navigate to Sequencer, select configure

2. In the new window pops up to configure the token location, scroll down to the token and highlight just the characters between the quotations then click ok. This will set the token for the tool and will populate the ‘Custom location’ field.



capture. Select the request to use, configure the other options below, then click "Start live capture".

Define custom token location

Define the location of the token within the response. Selecting the item in the response panel will create a suitable configuration automatically. You can also modify the configuration manually to ensure it works effectively.

☑ Define start and end

● Start after expression: _token":"

○ Start at offset: 880

● End at delimiter: "}

○ End at fixed length: 20

□ Extract from regex group

☑ Case sensitive

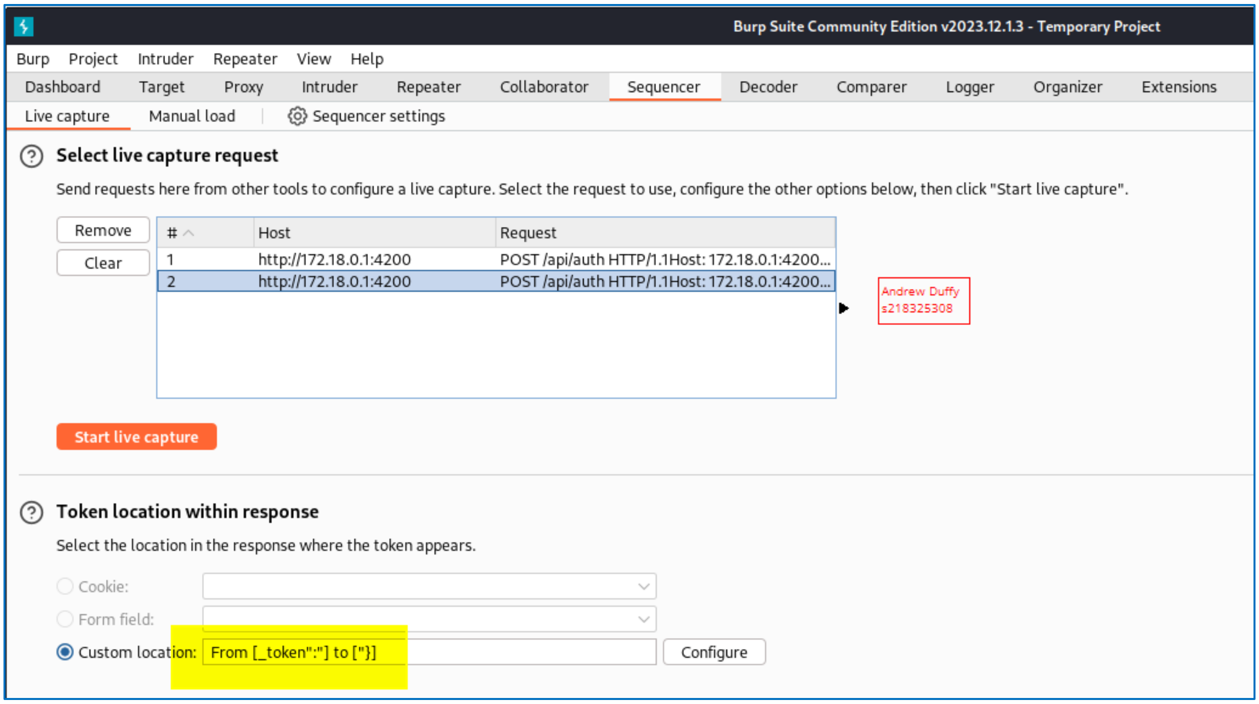
Andrew Duffy
s218325308

□ Exclude HTTP headers ☑ Update config based on selection below

```
content-type: application/json
vary: Origin
etag: W/"1840822a9a3eedbc08b667e2461a5e17"
cache-control: max-age=0, private, must-revalidate
x-request-id: fe44ed33-c077-4241-b726-351f72bbe178
x-runtime: 1.283422
connection: close
content-length: 394
Date: Tue, 01 Apr 2025 10:57:21 GMT

{"user":{"id":25,"student_id":"1111","email":"student_1@doubtfire.com","first_name":
"Ethelyn","last_name":"Lindgren","username":"student_1","nickname":"student_1",
"receive_task_notifications":true,"receive_portfolio_notifications":true,
"receive_feedback_notifications":true,"opt_in_to_research":false,
"has_run_first_time_setup":true,"system_role":"Student"},"auth_token":
"Q2PmNuZmssxT4yPH1AiS"}
```

Figure – Setting token location


Burp Suite Community Edition v2023.12.1.3 - Temporary Project
Burp Project Intruder Repeater View Help
Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions
Live capture Manual load Sequencer settings
Remove
Clear
Start live capture
Configure
Andrew Duffy
s218325308

Select live capture request

Send requests here from other tools to configure a live capture. Select the request to use, configure the other options below, then click "Start live capture".

#	Host	Request
1	http://172.18.0.1:4200	POST /api/auth HTTP/1.1Host: 172.18.0.1:4200...
2	http://172.18.0.1:4200	POST /api/auth HTTP/1.1Host: 172.18.0.1:4200...

Token location within response

Select the location in the response where the token appears.

○ Cookie:

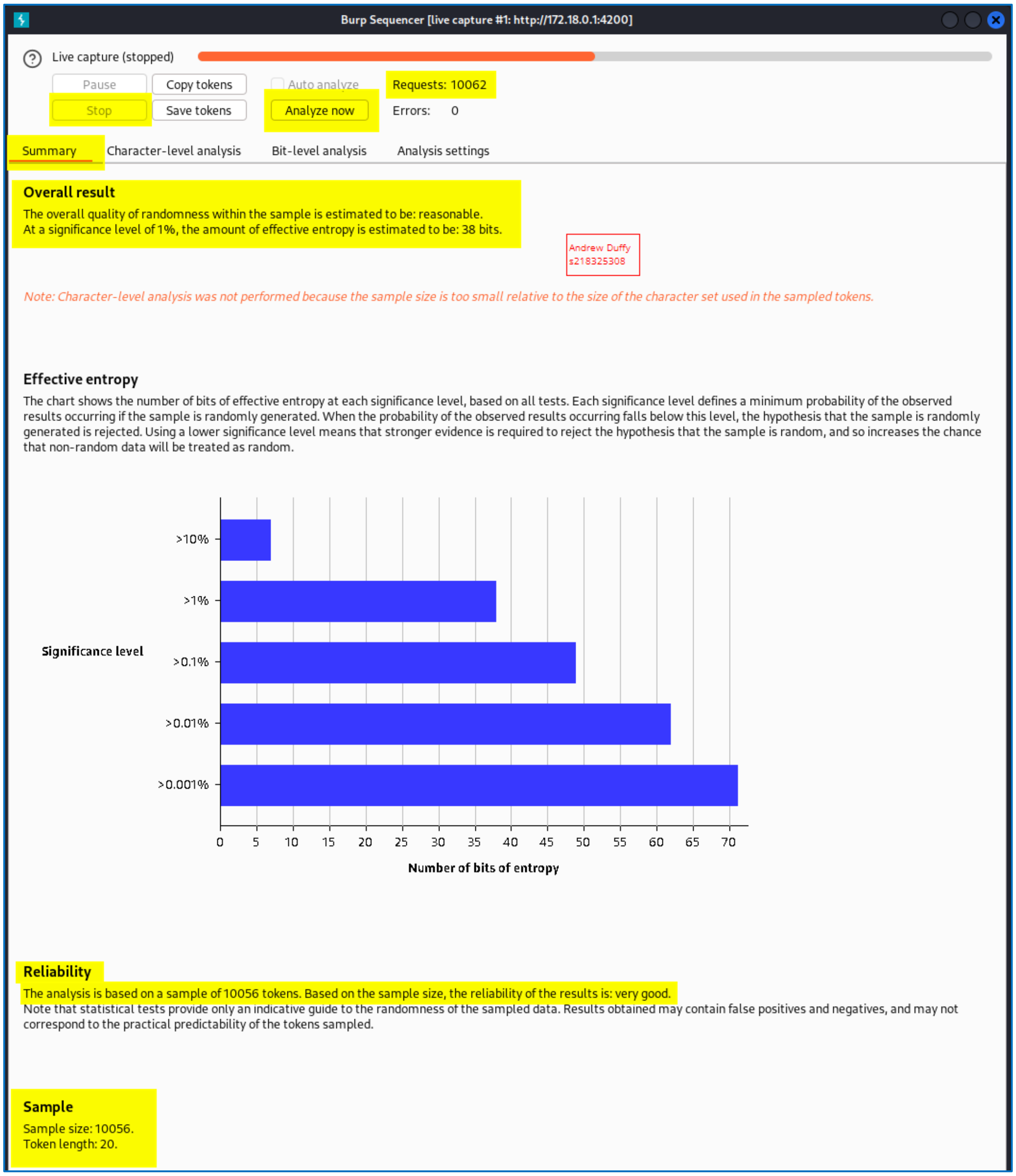
○ Form field:

● Custom location: From [_token":"] to ["}]

Figure – Custom token location populated within the tool

Step 3: Start live capture

1. Click ‘Start live capture’ this will open another window where the tool will send authorisation requests to the application and capturing the authorisation token. Let the tool run for approximately 10000 requests, this is significant enough to get an accurate indication of the token's strength. Once 10000 requests have been captured click ‘Stop’ then ‘Analyze now’. This will present a summary of the token's overall randomness and entropy.



Requests: 10062
Errors: 0

Overall result

The overall quality of randomness within the sample is estimated to be: reasonable.
At a significance level of 1%, the amount of effective entropy is estimated to be: 38 bits.

Andrew Duffy
s218325308

Note: Character-level analysis was not performed because the sample size is too small relative to the size of the character set used in the sampled tokens.

Effective entropy

The chart shows the number of bits of effective entropy at each significance level, based on all tests. Each significance level defines a minimum probability of the observed results occurring if the sample is randomly generated. When the probability of the observed results occurring falls below this level, the hypothesis that the sample is randomly generated is rejected. Using a lower significance level means that stronger evidence is required to reject the hypothesis that the sample is random, and so increases the chance that non-random data will be treated as random.

Reliability

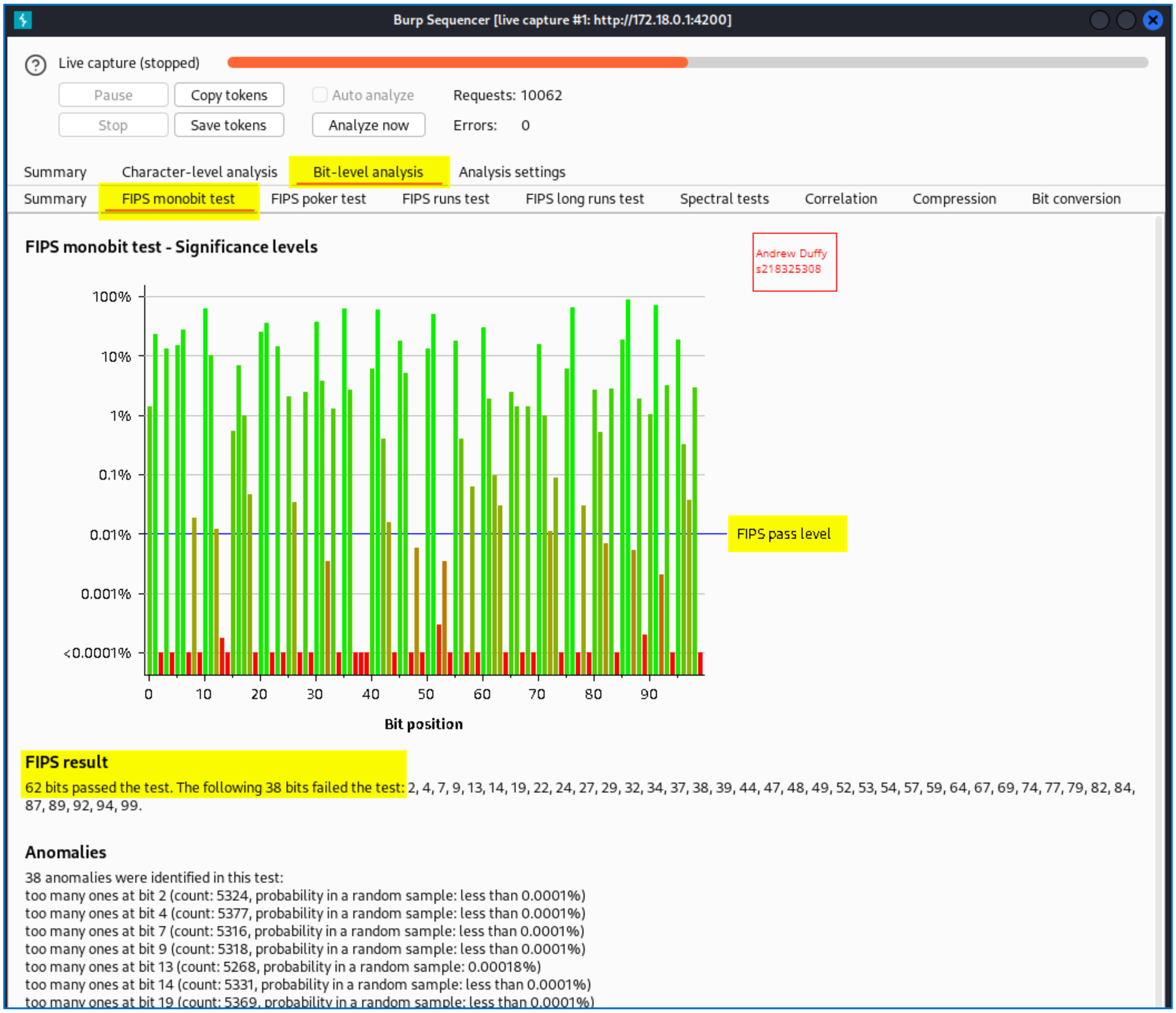
The analysis is based on a sample of 10056 tokens. Based on the sample size, the reliability of the results is: very good.
Note that statistical tests provide only an indicative guide to the randomness of the sampled data. Results obtained may contain false positives and negatives, and may not correspond to the practical predictability of the tokens sampled.

Sample

Sample size: 10056.
Token length: 20.

Figure – Summary of token analysis

2. Further analysis results are available to better understand bit predictability.



FIPS result

62 bits passed the test. The following 38 bits failed the test: 2, 4, 7, 9, 13, 14, 19, 22, 24, 27, 29, 32, 34, 37, 38, 39, 44, 47, 48, 49, 52, 53, 54, 57, 59, 64, 67, 69, 74, 77, 79, 82, 84, 87, 89, 92, 94, 99.

Anomalies

```
38 anomalies were identified in this test:
too many ones at bit 2 (count: 5324, probability in a random sample: less than 0.0001%)
too many ones at bit 4 (count: 5377, probability in a random sample: less than 0.0001%)
too many ones at bit 7 (count: 5316, probability in a random sample: less than 0.0001%)
too many ones at bit 9 (count: 5318, probability in a random sample: less than 0.0001%)
too many ones at bit 13 (count: 5268, probability in a random sample: 0.00018%)
too many ones at bit 14 (count: 5331, probability in a random sample: less than 0.0001%)
too many ones at bit 19 (count: 5369, probability in a random sample: less than 0.0001%)
```

Figure – Further analysis is available under each specific test

Remediation Advice

The application currently uses a session token generator with lower than recommended entropy, making tokens more predictable and vulnerable to brute-force attacks.

Mitigations:

- Review current session identifier implementation and uplift in line with OWASP recommendations, uplift to ensure at least 64 bits of entropy.
- Use a strong CSPRNG (Cryptographically Secure Pseudorandom Number Generator) must be used to generate session IDs.
- Retest session token using same method (e.g. Burp Suite Sequencer) to confirm improvements have been implemented effectively.

References

- OWASP - Session Management Cheat Sheet

- OWASP - Insufficient Session ID Length
- CWE - CWE-331: Insufficient Entropy

Contact Details

Name/Teams: Andrew Duffy
Email: s218325308@deakin.edu.au

Pentest Leader Feedback.

Filipe Oliveira s222478779@deaki.edu.au

- Make sure blue text is changed to black - Done
- Some bits of your text are not size 12 times new roman – Fixed, I note the sub heading ar size 13. And the table at top is 10.
- Expand on affected assets e.g any API endpoints or pages protected by session tokens – The affected item is the token, or the token generation implementation. Then by extention the whole application. -- Updated.
- You should also add more references e.g Burp, owasp auth cheat sheet - Added
- Unsure if at the bottom of step 2 there is a missing image or just accidental repeat of (Figure – Student level user accessing admin level endpoint) - Fixed. Removed.

well done!