MOP Website Security Audit



Contents

[MOP 3](#_Toc141975690)

[Vulnerability Assessment 3](#_Toc141975691)

[Tools Used: 3](#_Toc141975692)

[Results 4](#_Toc141975693)

[PII Disclosure 5](#_Toc141975694)

[Content Security Policy Header (CSP) 5](#_Toc141975695)

[Cross Domain Misconfiguration 8](#_Toc141975696)

[Vulnerable Js Libraries 8](#_Toc141975697)

[Encryption 9](#_Toc141975698)

[Tools used 9](#_Toc141975699)

[Improvements that can be made 12](#_Toc141975700)

[Error Handling 13](#_Toc141975701)

[Security Headers 15](#_Toc141975702)

[Tools used 15](#_Toc141975703)

[The Positives 16](#_Toc141975704)

[Improvements that can be made 16](#_Toc141975705)

[Method Validation 16](#_Toc141975706)

[Tools used: 16](#_Toc141975707)

[Backups 17](#_Toc141975708)

[Reference list 17](#_Toc141975709)

# MOP

The Melbourne Open Data Playground (MOP) is a significant capstone project that is worked on by students from Deakin University. During the Covid-19 pandemic, there was a growing demand for data among businesses seeking to enhance their decision-making processes. The MOP project aligns itself with two pivotal strategic documents from the Melbourne City Council. Firstly, the Economic Development Strategy emphasizes the city's ambition to become a digitally connected hub. Secondly, the 2021-2025 Council Plan highlights the objective of implementing programs that foster digital literacy skills and capabilities.

Since 2014, the City of Melbourne has been a pioneer in Open Data initiatives, yet recent research and local user engagement have identified a need to address the gap in knowledge surrounding accessing Open Data and stemming valuable insights to develop apps and solve city-related problems. As a response, the MOP project aims to provide concrete proof-of-concept examples, demonstrating how Open Data APIs can be effectively leveraged to deliver a diverse array of solutions.

## Vulnerability Assessment

This vulnerability assessment was the first test conducted for this audit as it highlights many issues that can be further researched to learn how vulnerable or how secure the website is.

## Tools Used:

ParrotOS

OWASP ZAP

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

### Results

While there are a few false positives within this website there are also things that can be changed, this section will go over what is a risk or a possible false positive, this section of the report will highlight key issues and offer solutions where possible.

### PII Disclosure

A screenshot of a computer

Description automatically generatedA screenshot of a computer

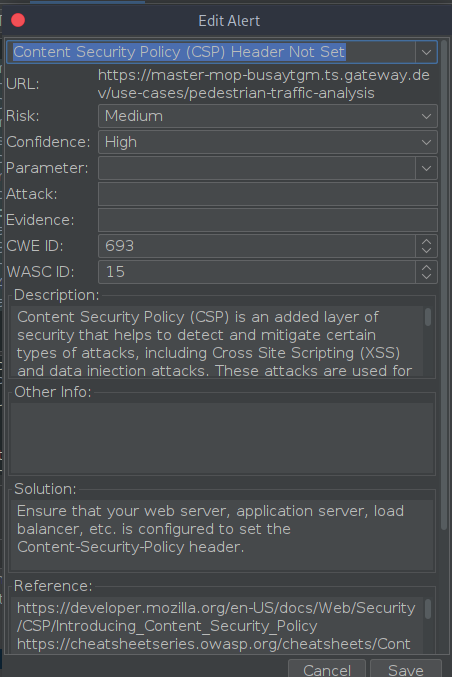
Description automatically generated

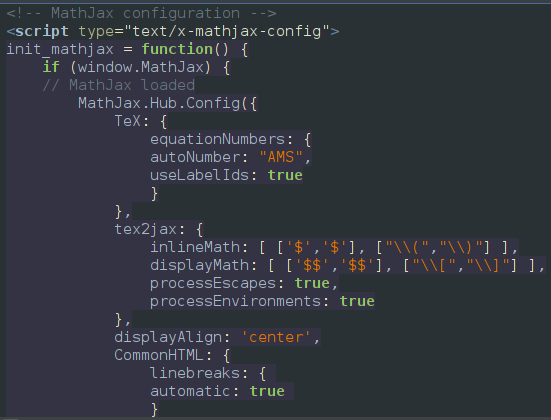
This flag is trying to say that there may be sensitive information that could help to identify people and or their personal information. Bernstein (2023) however, upon further examination of the error which believes a credit card is being exposed isn’t revealing anything but the GPS location(s) that is in the code, nothing can really be done in this situation as the website needs those GPS locations and thus is **not** a security risk.

### Content Security Policy Header (CSP)

A Content Security Policy is a HTTP response header that is in modern browsers that can allow or restrict resources and where they come from. Foundeo Inc. (n.d.)

A CSP header is a measure that prevents external resources from being loaded and executed this can help to reduce the possibility of cross site scripting attacks. Within the scan that was ran there are several pages with the same flag simply mentioning that the security policy isn’t as secure as what it should be. Below are a few images indicating what the error looks like and below there will an improved version that could be implemented.





A screenshot of a computer

Description automatically generated

In the above, we can see Mathjax, which utilises JavaScript to display equations on web pages. The reason why this is setting off a flag within the OWasp ZAP is because while MathJax is doing its job, the header hasn’t been properly configured, so that it is secure while allowing the resources for MathJax.

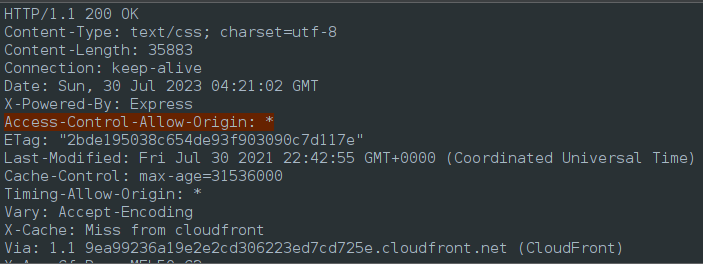
Here is an example of what the security header currently looks like:  
content-security-policy-report-only: default-src \*; report-uri /tools/csp-report

Meanwhile if we were to increase the security here is what a more security focused header would look like:

Content-Security-Policy: default-src 'self' trusted-scripts.com mathjax-cdn.com; font-src 'self' trusted-fonts.com mathjax-cdn.com;

There are a few differences between the two, including a different header format, and directives that changes it from allowing all sources (which is handy during development but not so handy in security as it allows for more attacks to take place). This updated version also only allows font resources to be loaded from 3 different sources (self, trusted-fonts.com, and MathJax), this can obviously be edited to include all the sources required and can be a possibly easy f fix for either the juniors or seniors in web development.

### Cross Domain Misconfiguration



Cross Domain Misconfiguration is a security vulnerability where a website is allowing resources from unauthorized external locations which can lead to XSS attacks or potentially leaking data. Sengupta (2022)

In the case of Cross Domain Misconfiguration with this data it is likely linked to the CSP, where editing the header to allow only trusted sources would increase security but also reduce the amount of data leakages and unauthorised scripts being ran. With the configuration being set as it is currently it will allow cross-origin requests from anywhere as it can lead to Open Cross-Site Forgery or possible Open Cross site Script Inclusion.

### Vulnerable Js Libraries

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Both above screenshots are instances where libraries are outdated, and while it isn’t a major risk to have outdated software it becomes a risk when the libraries are a few versions behind. Both; Jquery and Bootstrap have newer editions that could be implemented, there may be compatibility issues with plugins with newer versions however as of now there is no reason known as to why these older libraries are being used.

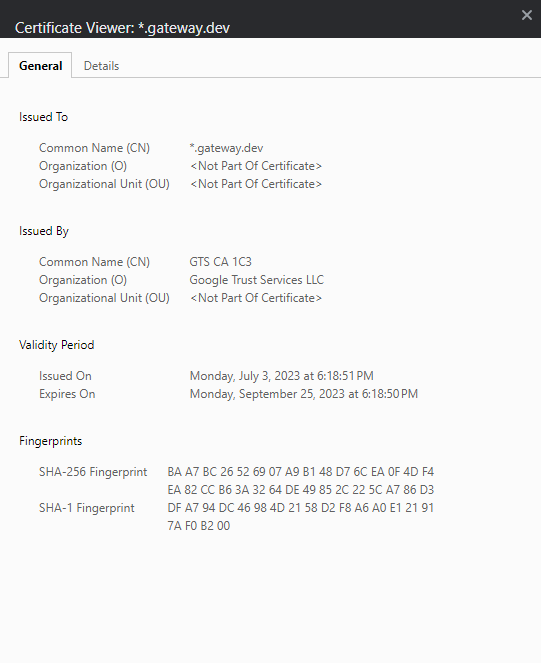
This one is an easy fix, since it simply requires using newer libraries however, finding a library that is compatible with all the work that has already been done can require some testing and thus can be time consuming. Upon further research, I was able to find out that the latest version of bootstrap is 5.3.1 and the latest version of jquery is 3.3.1.

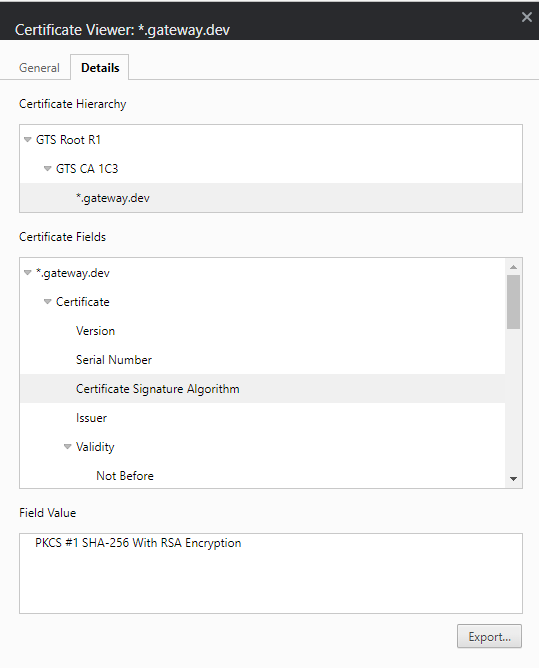
## Encryption

To ensure the most basic level of security it is important that the data on the websites we visit is encrypted. This will prevent any unauthorized attacks, and people being able to sniff packets to find out anything and everything that you view. Below is a screenshot of the security certificate that shows that the connection is indeed encrypted, and that the encryption is valid, this certificate is also valid for another 2 years so that doesn’t need to be worried about.

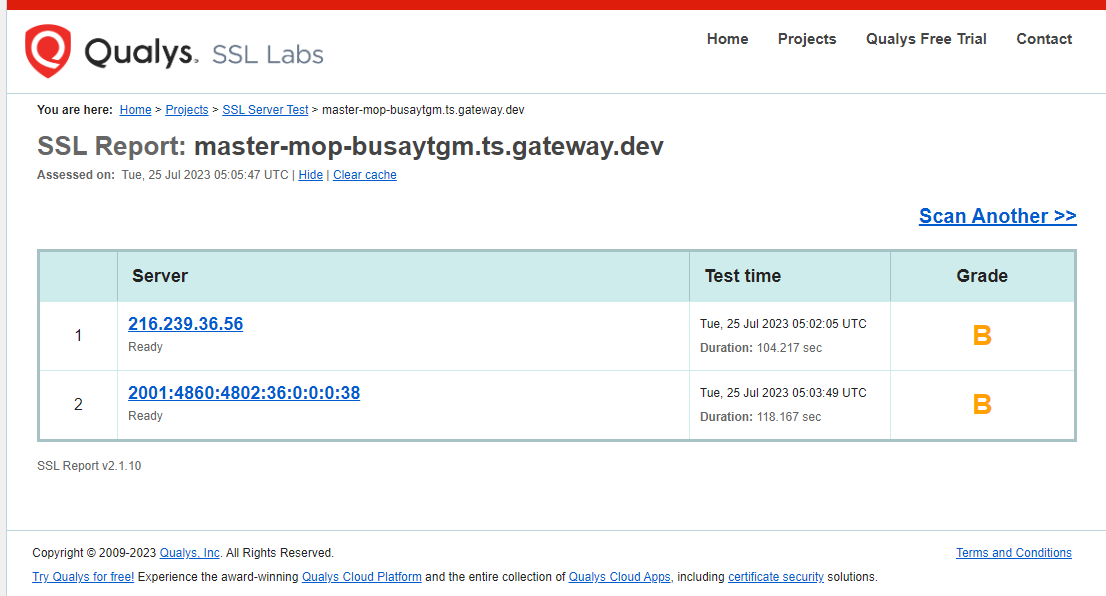
## Tools used

* ParrotOS
* Chromium
* Qualys

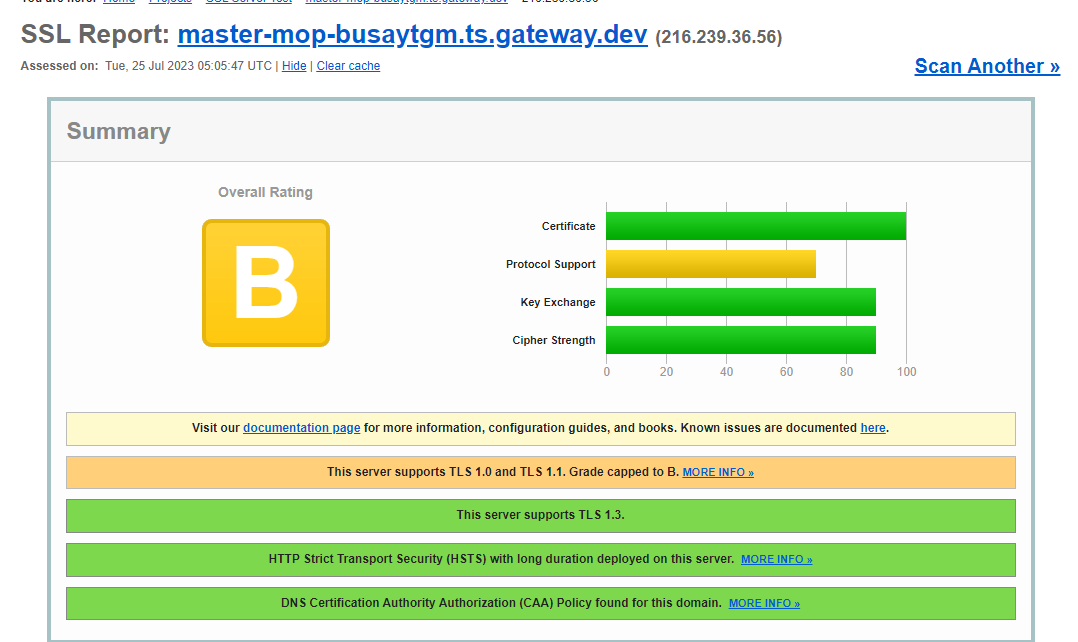




This above screenshot shows the type of encryption that is being used to encrypt the traffic on this website, for additional context RSA encryption offers a key size 2048 bits which is secure enough to prevent many attacks including brute force attacks.



In order to further test the encryption, I decided to use a third-party application dedicated to testing SSL (Qualys), to ensure that the encryption is doing what it should, and as we can see above the encryption is passing standards and below, we can see why it has a received a grade of B.



### Improvements that can be made

As we can see it is a simple change that can be made to receive an “A” when it comes to security standards and that change is removing support for TLS 1.0 and 1.1, setting a boundary of a minimum TLS 1.2 would increase security however that comes with certain risks as well, since that would lock out users who only have older devices with browsers that can’t be updated but since the world has been moving from TLS 1.0 and 1.1 Krynitsky (2023) perhaps it is time for Chameleon to do the same.

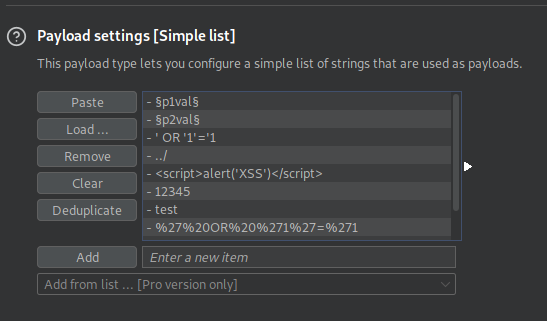
Not only is TLS 1.0 and 1.1 becoming obsolete it is becoming a regulation to have a minimum of TLS 1.2, not to mention the advantage that can be given with performance and security benefits (Stronger Algorithms, better certificate handling to name a few) (A10networks, n.d.)

## Error Handling

In this section of the report, I tested various payloads using Burp Suite, to see if the website is capable of handling unusual or potentially malicious attacks.

A screenshot of a computer program

Description automatically generated



In this example we are giving Burp Suite random payloads to put into random sections of the URL to see how the website reacts and because it is using a payload that is in a place that it shouldn’t be we are not getting access to the page and instead are greeted with an error (below)

A screenshot of a computer

Description automatically generated

To confirm that other errors were in place to handle other types of payloads, an attempt to spoof the user agent was done (see below for example payloads)

A screenshot of a computer

Description automatically generated

This returned a status of 200, rather than 405 showing that the webpage has detected errors that do not belong and continued to load the page normally. This payload and test were to ensure that spoofing attacks would not work against this website, and it passed that test. No improvements can be suggested at this time, especially since it is likely that the host of the website (GCP) is protecting the website against certain attacks.

## Security Headers

### Tools used

* ParrotOS
* Chromium
* Burp Suite

When we analyse the web page loading, we can see the following headers, and while all of them aren’t as important when it comes to security here is a summary of the purpose of these headers are:

A white screen with a black border

Description automatically generated

**Cache-Control:** Manages Caching behaviour and determines how long responses can be cached for.

**Content-Encoding:** Lets the client know about the type of encoding that was used and can help with decoding whatever is being displayed on the website.

**Content-Length**: Shows the response size, which can help with performance as it allows the client(s) to determine the complete response size in advance.

**Content-Security-Policy-Report-Only:** Reports violations of the CSP (Content Security Policy) without blocking, this can assist with security issues without leading to a loss of functionality and accessibility.

**Content type:** Self-explanatory, shows what type of content is being displayed on the webpage.

**Date:** Self-explanatory

**Permissions Policy:** This shows the permissions for the browser features which control which features can be enabled or disabled to enhance security.

**Referrer policy:** Manages the referrer information that gets sent when requests are made, this is to protect user privacy and can prevent information from being leaked.

**Server:** Shows the web server that is being used by the website.

**Strict Transport Security:** This header enforces HTTPS to be used for a specific period.

**X-Cloud-Trace-Context:** This relates to google cloud services and helps with request tracing for monitoring.

**X-Content-Type-Options:** Prevents Internet Explorer from being exploited and seeing this response as different content, this can help prevent being a victim of MIME attacks.

**X-Frame-Options:** Defends against clickjacking attacks by preventing the website from being loaded in a HTML iframe that lacks the proper configuration.

### The Positives

* **X-XSS Protection** – Additional defences against XSS attacks
* **X-Frame Options** – Prevents clickjacking
* **Strict-Transport-Security** – Prevents Man in the Middle attacks
* **X-Content-Type-Options** – Prevents MIME attacks
* **Cache-Control** – Improves performance and reduces the chance(s) of showing outdated content

### Improvements that can be made

* **Expect-CT** – Helps against transparency violations
* **Permissions-Policy** – Provides control over browser features which can increase security
* **X-Content-Security-Policy** – More security for older browsers

## Method Validation

### Tools used:

* Burp Suite
* ParrotOS
* Chromium

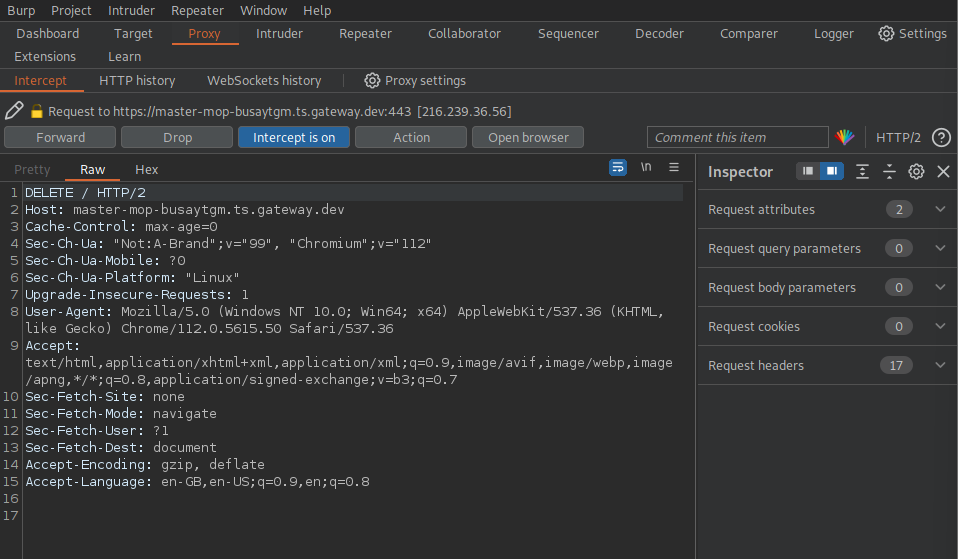
A screenshot of a computer

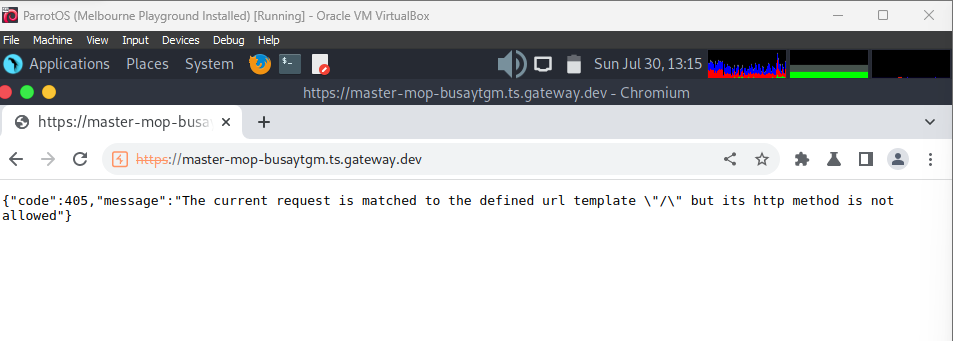
Description automatically generated

To test method validation, it was simply a matter of setting up a proxy to intercept web packets and edit them. Before forwarding them to edit them it was a matter of changing the type of request (In this case I tried multiple requests such as a GET, DELETE and POST requests, if the server was configured improperly then these requests would go through and changes could happen to the website however we are greeted with an error 405 which is pretty much saying we aren’t allowed to be doing this kind of request.

**Potential edit:**

To validate the testing method, I set up a proxy to intercept web packets and performed edits on them before forwarding them. The process included testing out different types of requests, such as GET, DELETE, and POST. If the server was not properly configured, these requests would go through, and changes could be made to the website. However, we ran into a 405-error message that said such requests were not permitted. By performing this validation process, we ensured that the website's security measures prevented unauthorized modifications and maintained system integrity.





# Backups

Upon Investigation with other leaders and the Unit chair Michelle, I was able to find out that there are no current backups or disaster recovery plans if things were to go wrong for Chameleon and because of that I will be taking the responsibility for drafting up both a disaster recovery plan and a backup plan with a both free and paid options being considered.

# Reference list

A10networks (n.d.). *Key differences Between TLS 1.2 and TLS 1.3 | Glossary*. [online] A10 Networks. Available at: https://www.a10networks.com/glossary/key-differences-between-tls-1-2-and-tls-1-3/#:~:text=TLS%201.2%20vs%20TLS%201.3.

Bernstein, C. (2023). *What is personally identifiable information (PII)? Definition from SearchSecurity*. [online] SearchSecurity. Available at: https://www.techtarget.com/searchsecurity/definition/personally-identifiable-information-PII.

Foundeo Inc. (n.d.). *Content-Security-Policy Header ⟶ CSP Reference & Examples*. [online] content-security-policy.com. Available at: https://content-security-policy.com/#:~:text=Content%2DSecurity%2DPolicy%20is%20the.

Krynitsky, J. (2023). *TLS 1.0 and TLS 1.1 soon to be disabled in Windows*. [online] TECHCOMMUNITY.MICROSOFT.COM. Available at: https://techcommunity.microsoft.com/t5/windows-it-pro-blog/tls-1-0-and-tls-1-1-soon-to-be-disabled-in-windows/ba-p/3887947 [Accessed 3 Aug. 2023].

Sengupta, S. (2022). *【CORS Error】Misconfigurations and their impacts*. [online] CrashTest Security. Available at: https://crashtest-security.com/cors-misconfiguration/#:~:text=commonly%20asked%20questions.- [Accessed 3 Aug. 2023].