

Vulnerability Assessment and Penetration Testing Report (VAPT)

Application Name : Denmark (M360 Newsroom)

Type : API

Date of Assessment : 19-12-2023

Report Prepared By: Shubham Singh

Team : Application Security Team whitehat@timesinternet.in

Executive Summary

Purpose of the Assessment

This manual security testing report provides an in-depth analysis of our application's security posture, conducted in accordance with the Open Web Application Security Project (OWASP) testing standards. The assessment focused on addressing the OWASP Top 10 vulnerabilities, aiming to identify and mitigate potential security risks within the application.

Summary of Findings

The assessment revealed a range of vulnerabilities and security issues within the application. Key findings include:

- OWASP Top 10 Vulnerabilities: High and medium-severity vulnerabilities related to the OWASP Top 10 were identified.
- General Findings: No Additional vulnerabilities were found, such as inadequate error handling etc.
- Data Exposure: Few instances of data exposure and information leakage were detected.

Issue	Host	Severity	Confidence	Status
CORS : Arbitrary Origin Trusted	https://nrapi.publishstory.co/	High	Confirmed	Open

Curl Snippet :

```
curl -i -s -k -X $'GET' \
  -H $'Host: nrapi.publishstory.co' -H $'Sec-Ch-Ua: \"Chromium\";v=\"116\", \"Not)A;Brand\";v=\"24\", \"Google Chrome\";v=\"116\"' -H $'Sec-Ch-Ua-Mobile: ?0' -H $'Sec-Ch-Ua-Platform: \"macOS\"' -H $'Upgrade-Insecure-Requests: 1' -H $'User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/116.0.0.0 Safari/537.36' -H $'Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7' -H $'Sec-Fetch-Site: none' -H $'Sec-Fetch-Mode: navigate' -H $'Sec-Fetch-User: ?1' -H $'Sec-Fetch-Dest: document' -H $'Origin: https://attacker.com' -H $'Accept-Encoding: gzip, deflate, br' -H $'Accept-Language: en-GB,en-US;q=0.9,en;q=0.8' -H $'Connection: close' \
```

```
$'https://nrapi.publishstory.co/newsroomApi/rendering/getsiteResources?clientId=2&websiteId=1'
```

Description :

The application implements an HTML5 cross-origin resource sharing (CORS) policy for this request that allows access from any domain.

The application allowed access from the requested origin **https://attacker.com**

An HTML5 cross-origin resource sharing (CORS) policy controls whether and how content running on other domains can perform two-way interaction with the domain that publishes the policy. The policy is fine-grained and can apply access controls per-request based on the URL and other features of the request.

Trusting arbitrary origins effectively disables the same-origin policy, allowing two-way interaction by third-party web sites. Unless the response consists only of unprotected public content, this policy is likely to present a security risk.

If the site specifies the header Access-Control-Allow-Credentials: true, third-party sites may be able to carry out privileged actions and retrieve sensitive information. Even if it does not, attackers may be able to bypass any IP-based access controls by proxying through users' browsers.

Steps to produce :

By altering the origin header in a request to any arbitrary domain, the application responds by providing a CORS (Cross-Origin Resource Sharing) response header, Access-Control-Allow-Origin (CAO), with the value specified in the origin header of the request.

This potentially leads to CORS issue where any domain making cross origin request is basically allowed to access its resources.

Proof of concepts :

The screenshot displays the Burp Suite interface with the 'Repeater' tab selected. The 'Request' pane on the left shows a GET request to `/newsroomApi/rendering/getsiteResources?clientId=2&websiteId=1` with various headers. The 'Response' pane on the right shows the corresponding HTTP response, which includes the `Access-Control-Allow-Origin: https://attacker.com` header. The 'Inspector' pane on the far right shows the selected text in the response, which is the `Origin: https://attacker.com` header value. The bottom status bar indicates that the request was successful (200 OK) and the response size is 4,415 bytes.

```
1 GET /newsroomApi/rendering/getsiteResources?clientId=2&websiteId=1 HTTP/1.1
2 Host: nrapi.publishstory.co
3 Sec-Ch-Ua: "Chromium";v="116", "NotA;Brand";v="24", "Google Chrome";v="116"
4 Sec-Ch-Ua-Mobile: ?0
5 Sec-Ch-Ua-Platform: "macOS"
6 Upgrade-Insecure-Requests: 1
7 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/116.0.0.0 Safari/537.36
8 Accept:
  text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
9 Sec-Fetch-Site: none
10 Sec-Fetch-Mode: navigate
11 Sec-Fetch-User: ?1
12 Sec-Fetch-Dest: document
13 Origin: https://attacker.com
14 Accept-Encoding: gzip, deflate, br
15 Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
16 Connection: close
17
18
19
20
21
22
23
```

```
1 HTTP/1.1 200 OK
2 Server: Bhoost
3 Content-Type: application/json
4 Access-Control-Allow-Credentials: true
5 Access-Control-Allow-Headers: Content-Type
6 Access-Control-Allow-Origin: https://attacker.com
7 Vary: Origin
8 Vary: Access-Control-Request-Method
9 Vary: Access-Control-Request-Headers
10 X-Frame-Options: sameorigin
11 Strict-Transport-Security: max-age=2592000; includeSubdomains
12 X-Xss-Protection: 1; mode=block
13 X-Content-Type-Options: nosniff
14 Content-Length: 3674
15 Vary: Accept-Encoding
16 Expires: Thu, 07 Dec 2023 09:49:33 GMT
17 Cache-Control: max-age=0, no-cache, no-store
18 Pragma: no-cache
19 Date: Thu, 07 Dec 2023 09:49:33 GMT
20 Connection: close
21 Server-Timing: ak_p; desc="1701942572741_824980045_38476954_12522_796_91_189_";dur=1
22
23 {
  "websiteTitle": "Demo Site 1",
  "tagline": "This is a Demo Site",
  "mainLogo": "184894223",
  "favicon": "1",
  "aboutSiteContent":
    "अप भाषा के विभिन्न Lorem Ipsum बस यादृच्छिक (random) पाठ नहीं है. यह 45 ई.पू. से 0 सदी तक लेखन साहित्य के एक टुकड़े से बना है, जो इसे 2000 वर्ष से अधिक पुरानी बनाता है. Richard McClintock, लंदन-ज्मीनिया में सिद्धी सलीम में एक लैटिन प्रोफेसर है, ने एक Lorem इन्सुम में से एक और अधिक अस्पष्ट लेखन शब्द देना और 0 सदी तक लेखन के लिए से जाने हुए अस्पष्टतमक लेख की कोश की. Lorem Ipsum सिस्ते(Sister) का "De Finibus Bonorum et Malorum" (अच्छाई और बुराई की पर म सीमा) के 1.10.32 और 1.10.33 वर्गों में आता है जो ४५ BC में लिखा गया था. यह दु सदी के "शिकवा के सिद्धांत" विषय पर लिख, जो नवगणना के दौर का एक बहुत लोकप्रिय ग्रंथ है. Lorem Ipsum की पहली पंक्ति, 1.10.32 खंड में एक पंक्ति से आती है.",
  "copyrightContent":
    "© 2020 A Times Internet Company. All rights reserved. Copyright © 2020 M360",
  "siteDescription": "This is a Demo Site description",
  "keywords": "Demo Site, Demo Siteel, Demo Site fooro new Platform",
  "fallbackImage": "12345",
  "footerLogo": "184894223",
  "metaImage": "123456",
  "adsTxtFile":
    "P Sample ads.txt<br># Direct sellersgoogle.com, pub-1234567890123456, DIRECT, f08c47fec0942fa0<br># Facebook.com, pub-9876543210987654, DIRECT, b0987c0b28e5d634<br># Resellers<br># Example.com, pub-8765432109876543, RESELLER, f08c47fec0942fa0",
  "robotsTxt":
    "P Sample robots.txt<br># User-agent: *<br>Disallow: /private/<br>Disallow: /restricted/<br>Allow: /public/",
  "siteLanguage": "Hindi",
  ...
}
```

Impact :

- Increased Attack Surface: Allowing requests from any origin significantly broadens the attack surface. Attackers can make cross-origin requests from various websites, potentially leading to security vulnerabilities.
- Cross-Site Request Forgery (CSRF): Allowing arbitrary origins may increase the risk of CSRF attacks. An attacker could trick users into making unintended requests to your API, exploiting their authenticated sessions.
- Data Exposure: Sensitive data exposed by your API might be accessible from unauthorized origins, leading to data exposure and confidentiality breaches.
- Unauthorized Access: Allowing any origin without proper authentication and authorization checks may result in unauthorized access to resources, undermining the security of your application.
- Cross-Site Scripting (XSS) Amplification: If your API returns sensitive information and is accessible from arbitrary origins, it could be used to amplify the impact of Cross-Site Scripting (XSS) attacks on other websites.

Mitigation :

Rather than using a wildcard or programmatically verifying supplied origins, use a whitelist of trusted domains.

Issue	Host	Severity	Confidence	Status
Broken Authentication	https://nrapi.publishstory.co/	High	Confirmed	Open

Curl Snippet :

```
curl -i -s -k -X $'GET' \
  -H $'Host: nrapi.publishstory.co' -H $'Cache-Control: max-age=0' -H $'Sec-Ch-Ua: \"Not_A
  Brand\";v=\"8\", \"Chromium\";v=\"120\", \"Google Chrome\";v=\"120\"' -H
  $'Sec-Ch-Ua-Mobile: ?0' -H $'Sec-Ch-Ua-Platform: \"macOS\"' -H
  $'Upgrade-Insecure-Requests: 1' -H $'User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X
  10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36' -H
  $'Accept:
  text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;
  q=0.8,application/signed-exchange;v=b3;q=0.7' -H $'Sec-Fetch-Site: none' -H
  $'Sec-Fetch-Mode: navigate' -H $'Sec-Fetch-User: ?1' -H $'Sec-Fetch-Dest: document' -H
  $'Accept-Encoding: gzip, deflate, br' -H $'Accept-Language: en-GB,en-US;q=0.9,en;q=0.8' -H
  $'Connection: close' \
```

```
$'https://nrapi.publishstory.co/newsroomApi/rendering/getsiteResources?clientId=2&websiteId
=1'
```

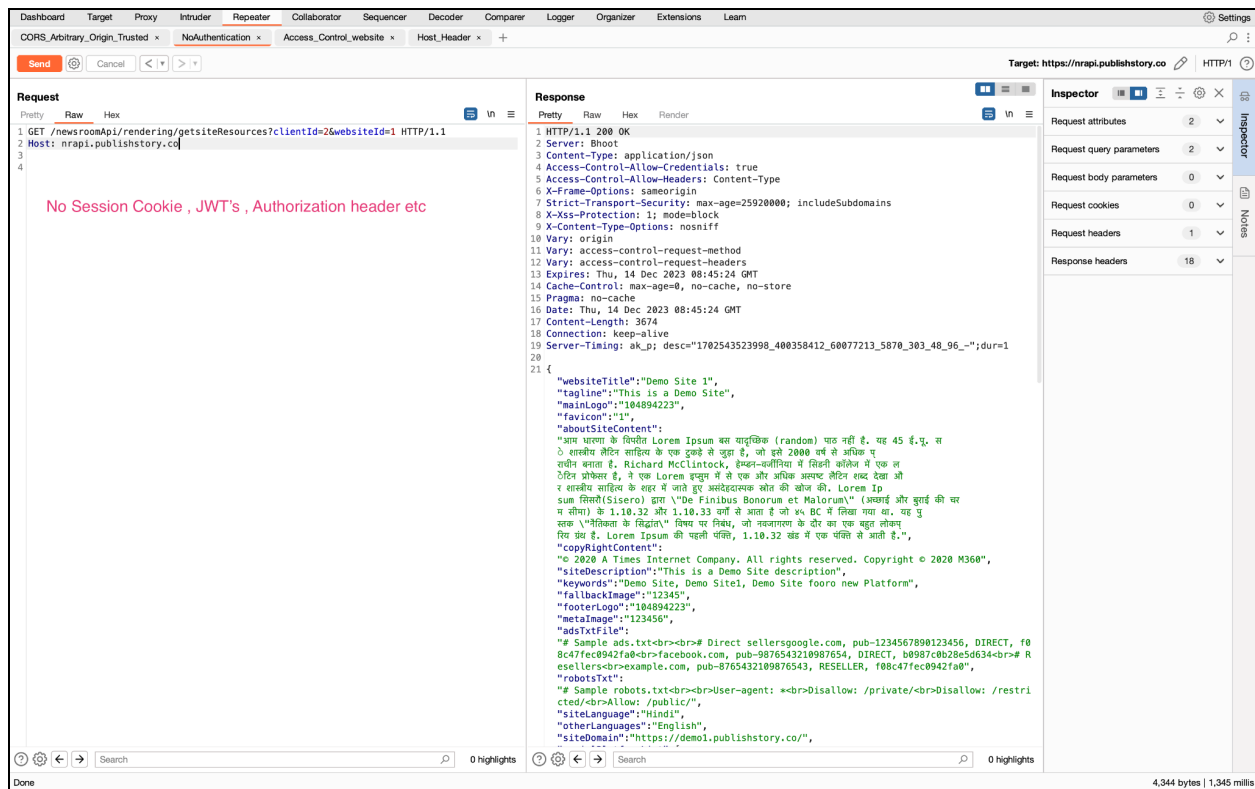
Description :

The API endpoint is serving responses without requiring proper authentication, allowing unauthenticated users to access sensitive data or perform actions within the system. This issue poses a significant security risk as it opens the door for unauthorized access, data exposure, and potential misuse of the API functionalities

Steps to produce :

Just directly hit the endpoint in the browser or via curl and its response is served back without any authentication

Proof of concepts :



Impact :

- **Unauthorized Access:**Unauthorized Access: Attackers can exploit the absence of authentication to gain unauthorized access to the API, potentially compromising sensitive information or functionalities.
- **Data Exposure:**Sensitive data transmitted or processed by the API may be exposed to unauthorized entities, leading to confidentiality breaches.
- **Misuse of Functionality :** Unauthenticated users may exploit API functionalities in unintended ways, causing disruptions or unauthorized operations within the system.

Mitigation :

- Make sure you know all the possible flows to authenticate to the API (mobile/web/deep links that implement one-click authentication/etc.). Ask your engineers what flows you missed.
- Read about your authentication mechanisms. Make sure you understand what and how they are used. OAuth is not authentication, and neither are API keys.
- Don't reinvent the wheel in authentication, token generation, or password storage. Use the standards.
- Credential recovery/forgot password endpoints should be treated as login endpoints in terms of brute force, rate limiting, and lockout protections.
- Require re-authentication for sensitive operations (e.g. changing the account owner email address/2FA phone number).
- Where possible, implement multi-factor authentication.
- Implement anti-brute force mechanisms to mitigate credential stuffing, dictionary attacks, and brute force attacks on your authentication endpoints. This mechanism should be stricter than the regular rate limiting mechanisms on your APIs.
- Implement account lockout/captcha mechanisms to prevent brute force attacks against specific users. Implement weak-password checks.
- API keys should not be used for user authentication. They should only be used for API clients authentication.

Issue	Host	Severity	Confidence	Status
Insecure Direct Object References (IDOR)	https://nrapi.publishstory.co/	High	Confirmed	Open

Curl Snippet :

```
curl -i -s -k -X $'GET' \
  -H $'Host: nrapi.publishstory.co' -H $'Cache-Control: max-age=0' -H $'Sec-Ch-Ua: \"Not_A
  Brand\";v=\"8\", \"Chromium\";v=\"120\", \"Google Chrome\";v=\"120\"' -H
  $'Sec-Ch-Ua-Mobile: ?0' -H $'Sec-Ch-Ua-Platform: \"macOS\"' -H
  $'Upgrade-Insecure-Requests: 1' -H $'User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X
  10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36' -H
  $'Accept:
  text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;
  q=0.8,application/signed-exchange;v=b3;q=0.7' -H $'Sec-Fetch-Site: none' -H
  $'Sec-Fetch-Mode: navigate' -H $'Sec-Fetch-User: ?1' -H $'Sec-Fetch-Dest: document' -H
  $'Accept-Encoding: gzip, deflate, br' -H $'Accept-Language: en-GB,en-US;q=0.9,en;q=0.8' -H
  $'Connection: close' \
```

```
$'https://nrapi.publishstory.co/newsroomApi/rendering/getsiteResources?clientId=2&websiteId
=1'
```

Description :

The API endpoint allows users to access different content by manipulating a query parameter. This behavior can be exploited by attackers to force the API to serve content that may not be intended for their access level, potentially leading to unauthorized access to sensitive information.

Steps to produce :

Changing websiteId query parameter value from original one to different numeric values allows us to enumerate different responses.

Proof of concepts :

Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions Learn

CORS_Arbitrary_Origin_Trusted x NoAuthentication x Access_Control_website x Host_Header x +

Send Cancel < >

Target: https://nrapi.publishstory.co HTTP/1

Request

Pretty Raw Hex

```
1 GET /newsroomApi/rendering/getSiteResources?clientId=2&websiteId=1 HTTP/1.1
2 Host: nrapi.publishstory.co
3 Cache-Control: max-age=0
4 Sec-Ch-Ua: "Not A Brand";v="8", "Chromium";v="120", "Google Chrome";v="120"
5 Sec-Ch-Ua-Mobile: ?0
6 Sec-Ch-Ua-Platform: "macOS"
7 Upgrade-Insecure-Requests: 1
8 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36
9 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
10 Sec-Fetch-Site: none
11 Sec-Fetch-Mode: navigate
12 Sec-Fetch-User: ?1
13 Sec-Fetch-Dest: document
14 Accept-Encoding: gzip, deflate, br
15 Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
16 Connection: close
17
18
```

0 highlights

Done

Response

Pretty Raw Hex Render

```
1 HTTP/1.1 200 OK
2 Server: Bhoht
3 Content-Type: application/json
4 Access-Control-Allow-Credentials: true
5 Access-Control-Allow-Headers: Content-Type
6 Vary: Origin
7 Vary: Access-Control-Request-Method
8 Vary: Access-Control-Request-Headers
9 X-Frame-Options: sameorigin
10 Strict-Transport-Security: max-age=2592000; includeSubdomains
11 X-Xss-Protection: 1; mode=block
12 X-Content-Type-Options: nosniff
13 Content-Length: 3674
14 Vary: Accept-Encoding
15 Expires: Thu, 14 Dec 2023 07:31:35 GMT
16 Cache-Control: max-age=0, no-cache, no-store
17 Pragma: no-cache
18 Date: Thu, 14 Dec 2023 07:31:35 GMT
19 Connection: close
20 Server-Timing: ak_p; desc="1702539094811_408358444_1363177255_60856_357_57_110_-";dur=1
21
22 {
  "websiteTitle": "Demo Site 1",
  "tagline": "This is a Demo Site",
  "mainLogo": "104894223",
  "favicon": "1",
  "aboutSiteContent": "
    *आम पृष्ठ के विषय Lorem Ipsum पर सांख्यिक (random) पाठ नहीं है. यह 45 ई.पू. स
    े शायद सेंटिन सांख्यिक के एक टुकड़े से जुड़ा है, जो इसे 2000 वर्ष से अधिक प्
    यक्ती बनाता है. Richard McClintock, हैन्स-जॉर्जिया में सिद्धी कॉलेज में एक ल
    गैरिन प्रोफेसर है, ने एक Lorem इपसुम में से एक और अधिक असंघटित सच्य देखा औ
    र शायद ही सांख्यिक के पक्ष में जारी हुए सांख्यिक सांख्यिक की खोज की. Lorem Ip
    sum विस्तर(Sisero) द्वारा "De Finibus Bonorum et Malorum" (असंख्य और खुद की पर
    म सीमा) के 1.10.32 और 1.10.33 वर्ग से आता है जो ५५ BC में लिखा गया था. यह पु
    स्तक "विश्वका के डिग्लिन" विष्णु पर लिखी, जो नवगणना के दौर का एक बहुत लोकप्र
    रित कि है. Lorem Ipsum की पहली पंक्ति, 1.10.32 खंड में एक पंक्ति से आती है.",
  "copyrightContent": "
    © 2020 A Times Internet Company. All rights reserved. Copyright © 2020 M360",
  "siteDescription": "This is a Demo Site description",
  "keywords": "Demo Site, Demo Site, Demo Site fooro new Platform",
  "fallbackImage": "12345",
  "footerLogo": "104894223",
  "metaImage": "123456",
  "adsTxtFile": "
    *# Sample ads.txt<br><br># Direct sellersgoogle.com, pub-1234567890123456, DIRECT, f0
    8c47fec0942fa0<br># facebook.com, pub-9876543210987654, DIRECT, b0987c0b28e5d634<br># R
    esellers<br># example.com, pub-8765432109876543, RESELLER, f08c47fec0942fa0",
  "robotsTxt": "
    *# Sample robots.txt<br><br># User-agent: *<br>Disallow: /private/<br>Disallow: /restr
    icted/<br>Allow: /public/",
  "siteLanguage": "Hindi",
  "otherLanguages": "English",
  "socialPlatformList": [
    {
      "name": "facebook",
      "url": "facebook.com",
      "pageId": "pageid12345",
      "admin": "admin1234",
      "appId": "appid1234"
    }
  ],
}
```

4,366 bytes | 913 milis

Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions Learn

CORS_Arbitrary_Origin_Trusted x NoAuthentication x Access_Control_website x Host_Header x +

Send Cancel < >

Target: https://nrapi.publishstory.co HTTP/1

Request

Pretty Raw Hex

```
1 GET /newsroomApi/rendering/getSiteResources?clientId=2&websiteId=2 HTTP/1.1
2 Host: nrapi.publishstory.co
3 Cache-Control: max-age=0
4 Sec-Ch-Ua: "Not A Brand";v="8", "Chromium";v="120", "Google Chrome";v="120"
5 Sec-Ch-Ua-Mobile: ?0
6 Sec-Ch-Ua-Platform: "macOS"
7 Upgrade-Insecure-Requests: 1
8 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36
9 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
10 Sec-Fetch-Site: none
11 Sec-Fetch-Mode: navigate
12 Sec-Fetch-User: ?1
13 Sec-Fetch-Dest: document
14 Accept-Encoding: gzip, deflate, br
15 Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
16 Connection: close
17
18
```

0 highlights

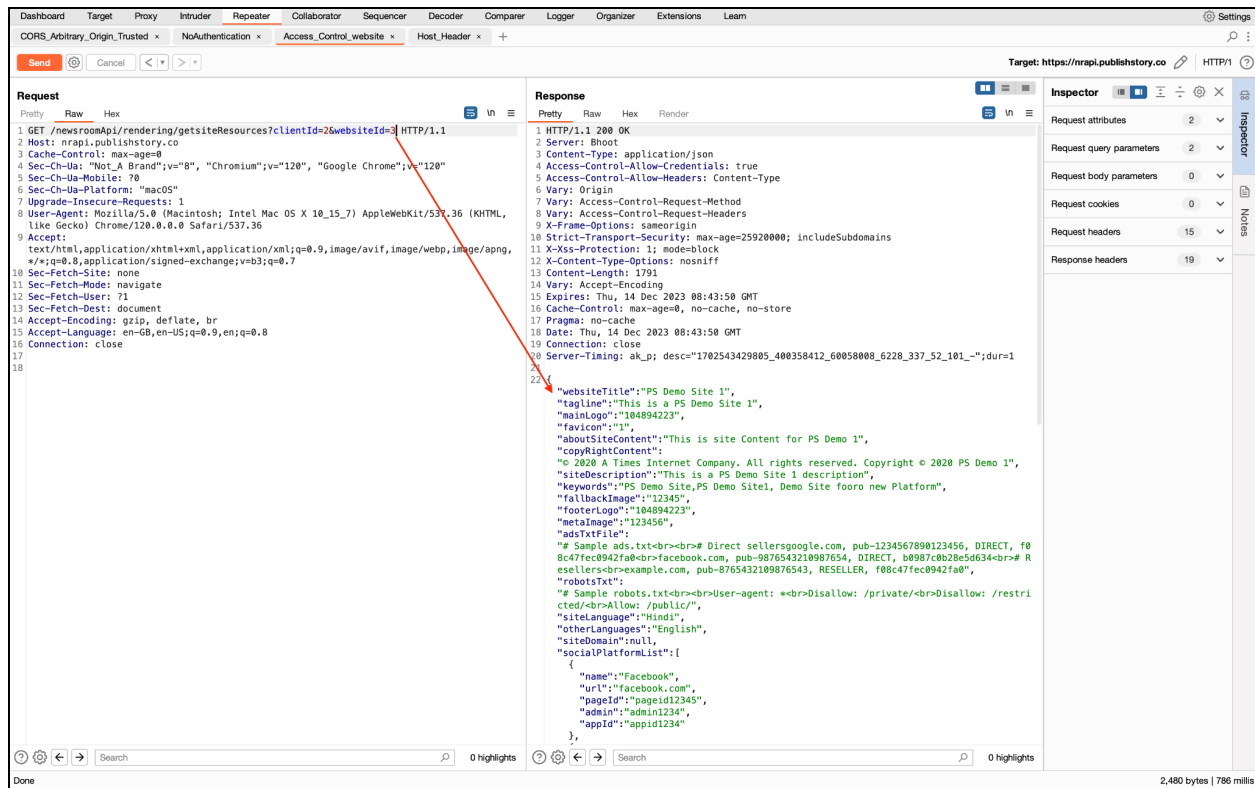
Done

Response

Pretty Raw Hex Render

```
1 HTTP/1.1 200 OK
2 Server: Bhoht
3 Content-Type: application/json
4 Access-Control-Allow-Credentials: true
5 Access-Control-Allow-Headers: Content-Type
6 Vary: Origin
7 Vary: Access-Control-Request-Method
8 Vary: Access-Control-Request-Headers
9 X-Frame-Options: sameorigin
10 Strict-Transport-Security: max-age=2592000; includeSubdomains
11 X-Xss-Protection: 1; mode=block
12 X-Content-Type-Options: nosniff
13 Content-Length: 1922
14 Vary: Accept-Encoding
15 Expires: Thu, 14 Dec 2023 08:43:27 GMT
16 Cache-Control: max-age=0, no-cache, no-store
17 Pragma: no-cache
18 Date: Thu, 14 Dec 2023 08:43:27 GMT
19 Connection: close
20 Server-Timing: ak_p; desc="1702543407029_408358412_60855087_8099_298_50_98_-";dur=1
21
22 {
  "websiteTitle": "Demo Site 2",
  "tagline": "This is a Demo Site",
  "mainLogo": "104894223",
  "favicon": "1",
  "aboutSiteContent": "This is site Content for Demo 2",
  "copyrightContent": "
    © 2020 A Times Internet Company. All rights reserved. Copyright © 2020 M360 Demo 2",
  "siteDescription": "This is a Demo Site 2 description",
  "keywords": "Demo Site, Demo Site2, Demo Site fooro new Platform",
  "fallbackImage": "12345",
  "footerLogo": "104894223",
  "metaImage": "123456",
  "adsTxtFile": "
    *# Sample ads.txt<br><br># Direct sellersgoogle.com, pub-1234567890123456, DIRECT, f0
    8c47fec0942fa0<br># facebook.com, pub-9876543210987654, DIRECT, b0987c0b28e5d634<br># R
    esellers<br># example.com, pub-8765432109876543, RESELLER, f08c47fec0942fa0",
  "robotsTxt": "
    *# Sample robots.txt<br><br># User-agent: *<br>Disallow: /private/<br>Disallow: /restr
    icted/<br>Allow: /public/",
  "siteLanguage": "Hindi",
  "otherLanguages": "EngLish",
  "siteDomain": "siteDomain",
  "socialPlatformList": [
    {
      "name": "facebook",
      "url": "facebook.com",
      "pageId": "pageid12345",
      "admin": "admin1234",
      "appId": "appid1234"
    }
  ],
}
```

2,610 bytes | 387 milis



Impact :

Unauthorized Data Access:

Attackers can manipulate the query parameter to access data or content that they are not authorized to view.

Information Disclosure:

Sensitive information may be disclosed to unauthorized users, leading to confidentiality breaches.

Data Integrity:

Manipulating the query parameter may result in unintended modifications to data, affecting data integrity.

Mitigation :

- **Implement Proper Authorization:** Ensure that access to different content is properly authorized based on user roles and permissions. Implement strong access controls to restrict access to authorized users.
- **Use Indirect References:** Instead of directly referencing objects or content, use indirect references that are harder for attackers to guess or manipulate.
- **Validate and Sanitize Input:** Implement input validation and sanitization to ensure that the values provided in query parameters are legitimate and within expected ranges.
- **Enforce Session Management:** If user sessions are involved, ensure that session management is secure and that users cannot manipulate session-related parameters to gain unauthorized access.
- **Logging and Monitoring:** Implement logging mechanisms to monitor and detect suspicious activities, such as repeated attempts to manipulate query parameters.
- **Conduct Security Testing:** Perform regular security testing, including penetration testing and code reviews, to identify and address vulnerabilities associated with direct object references.

Discussion Points:

- **Security Awareness:** Discuss the importance of secure coding practices with the development team to raise awareness about the risks associated with insecure direct object references.
- **Access Control Design:** Review and refine the access control design to ensure that only authorized users have access to specific content.
- **Security Testing:** Discuss the need for regular security testing to identify and address vulnerabilities, including those related to direct object references.
- **Documentation:** Document the proper usage of the API, including guidelines for handling query parameters and ensuring secure access.

Issue	Host	Severity	Confidence	Status
Open Redirect	https://nrapi.publishstory.co/	High	Confirmed	Open

Curl Snippet :

```
curl -i -s -k -X $'GET' \
  -H $'Host: nrapi.publishstory.co' -H $'Sec-Ch-Ua: \"Not_A Brand\";v=\"8\", \"Chromium\";v=\"120\", \"Google Chrome\";v=\"120\"' -H $'Sec-Ch-Ua-Mobile: ?0' -H $'Sec-Ch-Ua-Platform: \"macOS\"' -H $'Upgrade-Insecure-Requests: 1' -H $'User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36' -H $'Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7' -H $'Sec-Fetch-Site: same-site' -H $'Sec-Fetch-Mode: navigate' -H $'Sec-Fetch-User: ?1' -H $'Sec-Fetch-Dest: document' -H $'Referer: https://identity.publishstory.co/' -H $'Accept-Encoding: gzip, deflate, br' -H $'Accept-Language: en-GB,en-US;q=0.9,en;q=0.8' \
```

```
$'https://nrapi.publishstory.co/newsroomApi/userAuthenticate?host=https://anything.com&tempCode=W9BcBB2kMRgXlTSTVOJR7OV24jBXSE0M7XvQYaEAs--f4yB7nBibTA'
```

Description :

Open redirect vulnerabilities typically arise when a web application uses user-input to construct a URL for redirection without validating or sanitizing the input properly. Instead of redirecting users to a fixed and trusted URL, the application redirects them to the URL specified in the user-provided input.

Steps to produce :

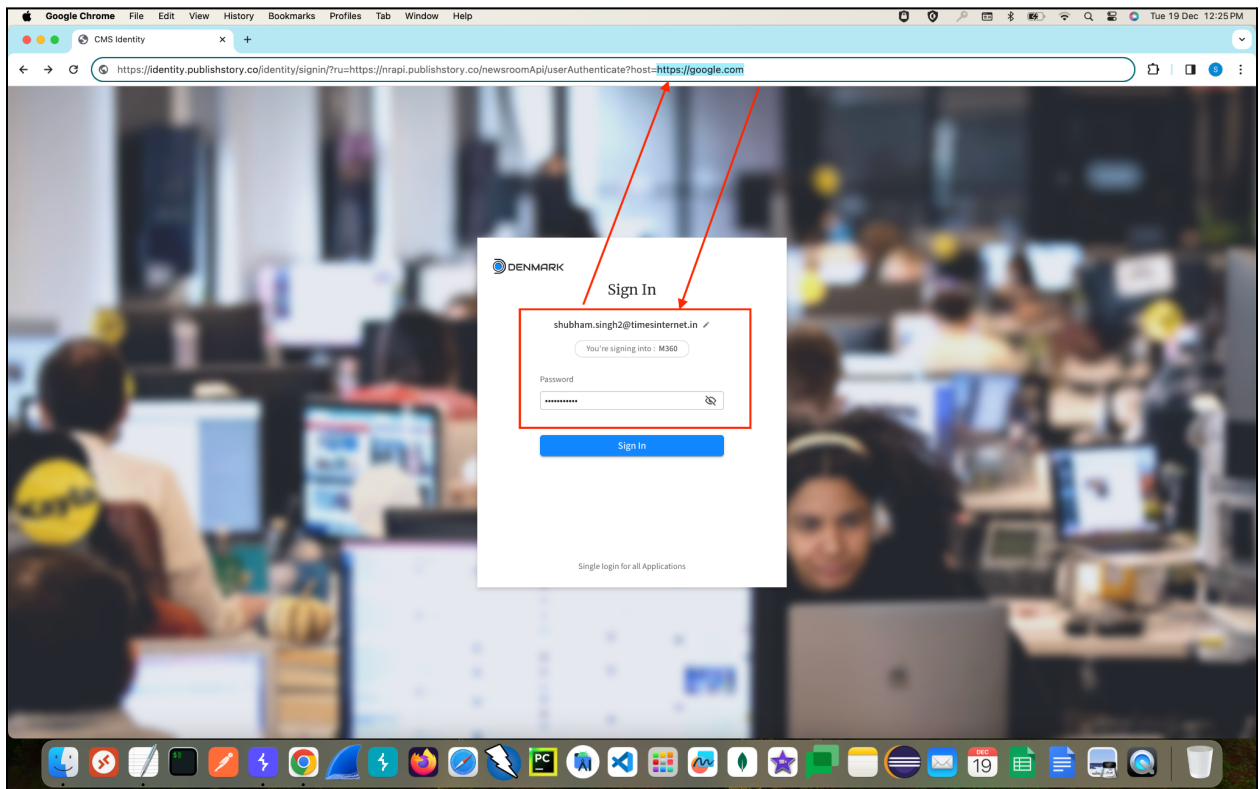
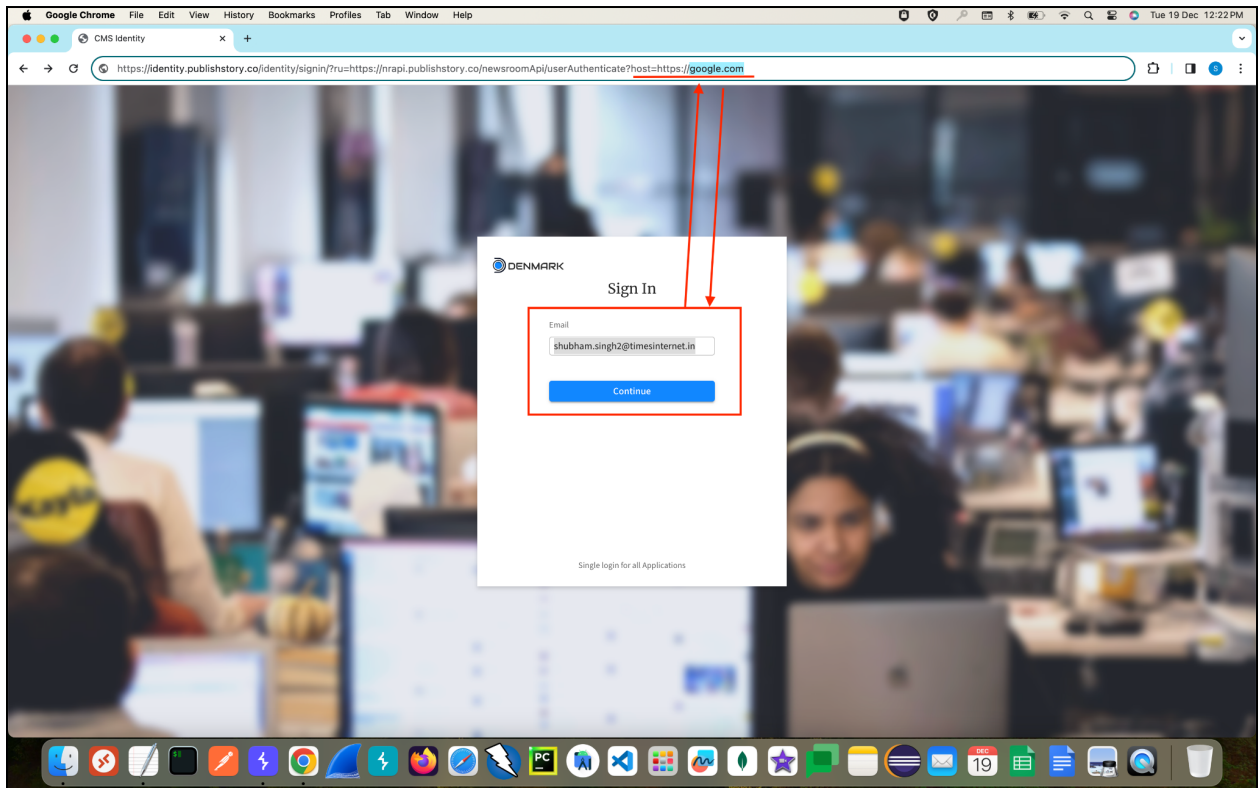
By changing the host query parameter value in the GET request , we were able to find a redirect towards the specified host.

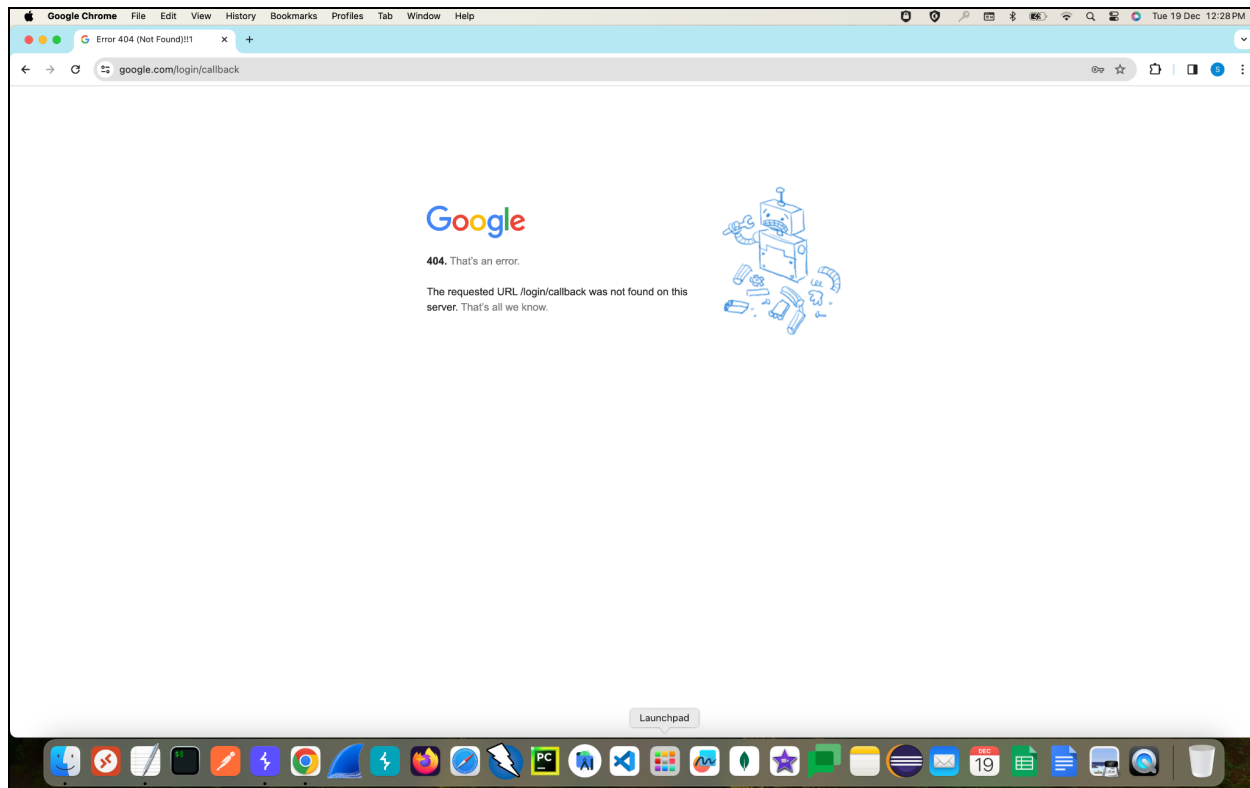
Below screenshots provides how we captured that specific request i.e. :

We requested <https://pagebuilder.publishstory.co/> in web browser and followed the normal flow by providing username and password

Captured a request where if the host parameter value changed to some arbitrary host value then it redirects us to that host

Proof of concepts :





DashboardTargetProxyIntruderRepeaterCollaboratorSequencerDecoderComparerLoggerOrganizerExtensionsLearnSensitiveDiscoverer

ID Enum × topics × RU_HOST header × 6 × +

SendCancel<>Follow redirection

Target: https://nrapi.publishstory.coHTTP/2

Request

1 GET /newsroomApi/userAuthenticate?host=https://anything.com&tempCode=W9BCB82kMRgXlTSTVQJR70V24jBXSE0M7XvQYaeAs--T4y07n8iBtA HTTP/2

2 Host: nrapi.publishstory.co

3 Sec-Ch-Ua: "Not A Brand";v="8", "Chromium";v="120", "Google Chrome";v="120"

4 Sec-Ch-Ua-Mobile: ?0

5 Sec-Ch-Ua-Platform: "macOS"

6 Upgrade-Insecure-Requests: 1

7 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.0.0 Safari/537.36

8 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7

9 Sec-Fetch-Site: same-site

10 Sec-Fetch-Mode: navigate

11 Sec-Fetch-User: ?1

12 Sec-Fetch-Dest: document

13 Referer: https://identity.publishstory.co/

14 Accept-Encoding: gzip, deflate, br

15 Accept-Language: en-GB,en-US;q=0.9,en;q=0.8

16

17

Response

1 HTTP/2 302 Found

2 Server: Bhoost

3 Content-Length: 0

4 Access-Control-Allow-Credentials: true

5 Access-Control-Allow-Headers: Content-Type

6 Vary: Origin

7 Vary: Access-Control-Request-Method

8 Vary: Access-Control-Request-Headers

9 Location: https://anything.com/login/callback

10 X-Frame-Options: sameorigin

11 Strict-Transport-Security: max-age=2592000; includeSubdomains

12 X-Xss-Protection: 1; mode=block

13 X-Content-Type-Options: nosniff

14 Expires: Tue, 19 Dec 2023 07:14:26 GMT

15 Cache-Control: max-age=0, no-cache, no-store

16 Pragma: no-cache

17 Date: Tue, 19 Dec 2023 07:14:26 GMT

18 Set-Cookie: m360auth=264e1770b18c3f1a7a90ca8d3705769ef8b843d8a65a63dbd9ce9b3b51061784; Max-Age=3600; Expires=Tue, 19 Dec 2023 08:14:26 GMT; Domain=publishstory.co; Path=/; HttpOnly

19 Server-Timing: ak_p; desc="1702970066125_835648492_75873410_19703_847_193_0_15";dur=1

20

21

Inspector

Selection 20 (0x14)

Selected text https://anything.com

Decoded from: URL encoding

https://anything.com

CancelApply changes

Request attributes 2

Request query parameters 2

Request body parameters 0

Request cookies 0

Request headers 17

Response headers 18

Search 0 highlights

Search 0 highlights

Done843 bytes | 217 millis

Impact :

The impact of an open redirect vulnerability can be significant. Attackers can exploit it to trick users into visiting malicious websites, leading to phishing attacks, spreading malware, or stealing sensitive information. It can also be used in combination with other attacks to manipulate user sessions or perform other malicious activities.

Mitigation :

To mitigate open redirect vulnerabilities, consider implementing the following best practices:

- **Input Validation:** Always validate and sanitize user inputs, especially when constructing URLs for redirection. Ensure that user-provided URLs are safe and expected.
- **Whitelisting:** Maintain a whitelist of trusted URLs to which the application is allowed to redirect users. Validate user inputs against this whitelist to ensure they are legitimate destinations.
- **Use Safe Redirection Methods:** When redirecting users, use safe and trusted redirection methods provided by the programming language or framework. Avoid using user-input directly to construct the redirect URL.
- **Security Headers:** Implement security headers such as Content Security Policy (CSP) and Strict-Transport-Security (HSTS) to enhance overall security and prevent certain types of attacks.

Environmental Limitations:

NA

Lack of Details:

NA

Recommendations :

- To address these findings and enhance the security of M360 Newsroom API , the following recommendations are provided:
- Remediate OWASP Top 10 Vulnerabilities: Prioritize and resolve OWASP Top 10 vulnerabilities promptly to reduce the application's attack surface.
- Enhance Authentication and Authorization: Strengthen authentication mechanisms and ensure proper authorization checks are in place.
- Implement Robust Session Management: Enhance session management controls to prevent session hijacking and fixation.
- Data Protection: Encrypt sensitive data both in transit and at rest to mitigate data exposure risks.
- Access Control: Review and enforce access controls to prevent unauthorized access to sensitive resources.
- Logging and Monitoring: Enhance logging practices to facilitate incident detection and response.

Conclusion :

The VAPT assessment has highlighted significant security vulnerabilities in M360 Newsroom API , which could expose it to various cyber threats. Addressing these findings is critical to ensure the confidentiality, integrity, and availability of the application.

Next Steps :

We recommend that the team should take the following steps:

- Review the detailed findings and recommendations in subsequent sections of this report.
- Prioritize and plan remediation efforts based on the severity and potential impact of the identified vulnerabilities.
- Continuously monitor the application's security posture and conduct regular security assessments to stay ahead of emerging threats.