



Penetration Test Report

PREPARED FOR
Chaintrack Company

Simon Fraser University, MPCS Security

December 2022

PREPARED BY

Alia Bohsain Navarro

Allen Shaolun Liu

Jason Chen

Kaiyu Dong

Parsa Azimi Bajestani

Sami Salamati

Table of Contents

Executive Summary	4
1. Project Scope	4
2. Project Objectives	4
3. Timeline	5
4. Summary of Findings	6
5. Summary of Recommendations	6
Methodology	7
1. ZAP	7
2. Burp Suite	7
3. Maltego	7
4. Nessus	7
5. Bash	8
6. Arachni	8
7. sqlmap	8
Detailed Findings	8
1 Information Gathering	8
Fingerprint Web Server	8
Application Entry Points	10
DNS Trace Process	10
Maltego	11
2 Configuration and Deployment Management	12
Test Network Infrastructure Configuration	12
Test Application Platform Configuration	13
Test File Extensions Handling for Sensitive Information	14
Enumerate Infrastructure and Application Admin Interfaces	15
Test HTTP Strict Transport Security	17
3 Identity Management	18
Testing for default credentials	18
4 Authorization	20
Encrypted Channel	20
Testing for Weak lockout mechanism	22
Test remember password functionality	24
Insecure Direct Object References	30
5 Session Management	33
Logout Functionality	33
Cross Site Request Forgery (CSRF)	35

Cookies Attributes	35
6 Input Validation	36
Reflected Cross Site Scripting	36
Stored Cross Site Scripting	40
SQL Injection	43
7 Error Handling	46
8 Weak Cryptography	48
Testing for Weak Transport Layer Security	48
9 Business Logic	50
Teams APIs	50
Vehicles APIs	50
Orders APIs	51
10 Client Side	53
Client Side URL Redirect	53
Clickjacking	54



Executive Summary

1. Project Scope

Generate a report with a risk analysis based on the vulnerabilities found during the recognition and penetration phase to provide recommendations to improve the security in Chaintrack.

The report will be based on the Chaintrack web application only used by the dispatchers and excluding the mobile application that runs on iOS used by the drivers.

The results will be categorized according to their risk levels.



These measurements will help identify which vulnerabilities need to be addressed faster than others.

2. Project Objectives

There are several tasks that we will accomplish in this project. First, each team member is responsible for researching and implementing the penetration test and writing the report regarding the part of their assigned tasks.

Objectives	Team member
Gathering information	Allen
Configuration & deployment management	Kaiyu
Identity management	Jason
Authentication	Jason
Authorization	Sami
Session management	Sami
Input validation	Alia
Error handling test	Allen
Weak cryptography	Kaiyu

Business logic test	Parsa
Client side test	Alia

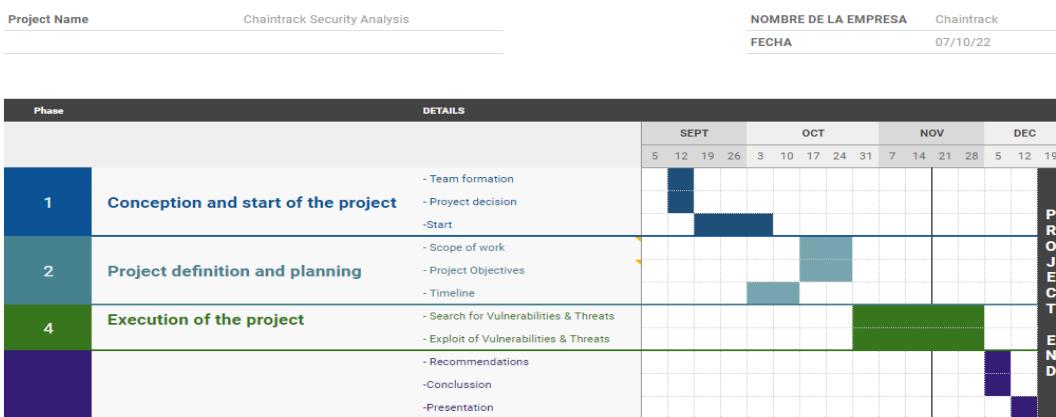
The table above indicates the main topic the penetration test covers with corresponding team members. Each member must choose several sub-topics under each main topic suitable to the chaintrack application.

More specifically, information gathering is essential because it helps penetration testers to understand the target system and identify potential vulnerabilities. Configuration and deployment management help to ensure the system is configured correctly and deployed securely. Proper configuration and deployment management can reduce the risk of vulnerabilities and attacks on the system. Identity management ensures that only authorized users have access to systems and data. Authentication and Authorization also help in preventing unauthorized access to sensitive information and reduce the risk of attacks on systems. Session management ensures the entrance to the methods and data is appropriately controlled and authorized. Input validation helps ensure the data input into the system is valid and legitimate. Error handling tests can reveal information about the application tested. Weakness of the cryptography indicates the possibility of the application being exploited by an attacker to gain unauthorized access to sensitive information. Business logic testing involves testing the logic and processes of the application. Client-side testing consists in testing the security of the client-side component.

3. Timeline

The proposed timeline to complete this project was planted as follows: each cell represents a week since the project was initiated. The execution of the project phase is related to the table of responsibilities previously provided.

Chaintrack Timeline



4. Summary of Findings

Below is a list of the vulnerabilities we found. For more details, please check the Detailed Findings section.

- (1) The remote web server is not enforcing HSTS, which allows downgrade attacks, SSL-stripping man-in-the-middle attacks, and weakens cookie-hijacking protections.
- (2) According to the server response header, the installed version of nginx is prior to 1.17.7. It is, therefore, affected by an information disclosure vulnerability.
- (3) The browser sends requests with 'Remembered Password Cookie' everytime.
- (4) No account lock or IP lock after too many failed log in attempts.
- (5) Cookie HttpOnly, Secure, SameSite attributes are not set.
- (6) Users can export other teams' orders.
- (7) Users can get a list of vehicles of other teams.
- (8) Users can get, create, and delete other teams orders.

5. Summary of Recommendations

Below is a list of the recommendations to remediate the vulnerabilities mentioned above. For more details, please check the Detailed Findings section.

- (1) Set cookie HttpOnly, Secure, SameSite attributes.
- (2) Configure the remote web server to use HSTS.
- (3) Upgrade to nginx version 1.17.7 or later.
- (4) Set the path of the remembered credential cookie in the response to the login page.
- (5) Add account lock and IP lock after too many failed log in attempts.
- (6) Control the access of the users over other teams in different APIs.



Methodology

Tests are splitted using OWASP testing guide categories.

Tools used:

1. ZAP

OWASP ZAP (short for Zed Attack Proxy) is an open-source web application security scanner. It is intended to be used by both those new to application security as well as professional penetration testers.

It is one of the most active Open Web Application Security Project (OWASP) projects and has been given Flagship status.

When used as a proxy server it allows the user to manipulate all of the traffic that passes through it, including traffic using HTTPS.

It can also run in a daemon mode which is then controlled via a REST API.

2. Burp Suite

Burp Suite is an integrated platform/graphical tool for performing security testing of web applications. Its various tools work seamlessly together to support the entire testing process, from initial mapping and analysis of an application's attack surface, through to finding and exploiting security vulnerabilities. Burp Suite is installed by default in Kali Linux.

3. Maltego

Maltego is software used for open-source intelligence and forensics, developed by Paterva from Pretoria, South Africa. Maltego focuses on providing a library of transforms for discovery of data from open sources, and visualizing that information in a graph format, suitable for link analysis and data mining.

4. Nessus

Nessus is a remote security scanning tool, which scans a computer and raises an alert if it discovers any vulnerabilities that malicious hackers could use to gain access to any computer you have connected to a network. It does this by running over 1200 checks on a

given computer, testing to see if any of these attacks could be used to break into the computer or otherwise harm it.

5. Bash

Bash is a Unix shell and command language written by Brian Fox for the GNU Project as a free software replacement for the Bourne shell.

6. Arachni

Arachni is a feature-full, modular, high-performance Ruby framework aimed towards helping penetration testers and administrators evaluate the security of modern web applications.

7. sqlmap

Sqlmap is an open source penetration testing tool that automates the process of detecting and exploiting SQL injection flaws and taking over of database servers. It comes with a powerful detection engine, many niche features for the ultimate penetration tester and a broad range of switches lasting from database fingerprinting, over data fetching from the database, to accessing the underlying file system and executing commands on the operating system via out-of-band connections.



Detailed Findings

1 Information Gathering

Fingerprint Web Server

In the information-gathering section, our group decided to obtain the web server fingerprint with banner grabbing using tools such as Burp Suite and Nmap, etc. The following image indicates the result we get from Nmap functions to identify the type and version of the server on which the target application is running. Automated testing tools can easily capture such information. Also, they need to replace the path of the "target" with their target path. This step sets the target of the web application that users want to scan.

```

8008/tcp open  http
|_ fingerprint-strings:
|   FourOhFourRequest:
|     HTTP/1.1 302 Found
|     Location: https://:8015/nice%20ports%2C/Tri%6Eity.txt%2ebak
|     Connection: close
|     X-Frame-Options: SAMEORIGIN
|     X-XSS-Protection: 1; mode=block
|     X-Content-Type-Options: nosniff
|     Content-Security-Policy: frame-ancestors 'self'
|     GenericLines, HTTPOptions, RTSPRequest, SIPOptions:
|       HTTP/1.1 302 Found
|       Location: https://:8015
|       Connection: close
|       X-Frame-Options: SAMEORIGIN
|       X-XSS-Protection: 1; mode=block
|       X-Content-Type-Options: nosniff
|       Content-Security-Policy: frame-ancestors 'self'
|     GetRequest:
|       HTTP/1.1 302 Found
|       Location: https://:8015/
|       Connection: close
|       X-Frame-Options: SAMEORIGIN
|       X-XSS-Protection: 1; mode=block
|       X-Content-Type-Options: nosniff
|       Content-Security-Policy: frame-ancestors 'self'
8010/tcp open  ssl/xmpp?
|_ fingerprint-strings:
|   GenericLines, GetRequest:
|     HTTP/1.1 200 OK
|     Content-Length: 4492
|     Connection: close
|     Cache-Control: no-cache
|     Content-Type: text/html; charset=utf-8
|     X-Frame-Options: SAMEORIGIN
|     X-XSS-Protection: 1; mode=block
|     X-Content-Type-Options: nosniff
|     Content-Security-Policy: frame-ancestors 'self'
|     <!DOCTYPE html>
|     <html lang="en">
|       <head>
|         <meta charset="UTF-8">
|         <meta http-equiv="X-UA-Compatible" content="IE=8; IE=EDGE">
|         <meta name="viewport" content="width=device-width, initial-scale=1">
|         <style type="text/css">
|           body {
|             height: 100%;
|             font-family: Helvetica, Arial, sans-serif;
|             color: #6a6a6a;
|             margin: 0;
|             display: flex;
|             align-items: center;
|             justify-content: center;
|             input[type=date], input[type=email], input[type=number], input[type=password]
|           }
|         </style>
|       </head>
|       <body>
|       </body>
|     </html>
|   
```

Banner grabbing is performed by sending an HTTP request to the web server and examining its response headers. It can also execute by using the Burp Suite application.

```

Request to https://app.chaintrack.ca:443 [52.32.229.136]
Forward Drop Intercept is on Action Open Browser Com
Pretty Raw Hex
1 GET /dashboard HTTP/2
2 Host: app.chaintrack.ca
3 Cookie: XSRF-TOKEN=
eyJpdiI6IkZ2NDUzUE5zJUMTUl4R0dpdFVXWc9PSIzInzhbHVlIjoiPK1wSHVXdjl4S3VdYkcyalUMzP89mYk9NaWvwd
Dm5eVcrXcSpEd5Q2zd5QWNTsHlIVzk0eBwQjReR09tQm14M0gjLCjtYwM=0iJ1ZGJ1YzA52wMmEzE0njUwZGQzYjc0M2
UyMDBlNTM40tBiMzJ0GE4Wm0OvxZhkMeM0M20mjYm41n0h30; chaintrack_session=
eyJpdiI6In40001SaPpjdydhxdrIjQWjlpjaEE9PSIzInzhbHVlIjoiNKJ0mZyNytaZDJUjWjlhShR0eTR0ZEP9NGdSa
FdMcInGa3hznE10du9JUtx3cG5unRaV1dhbEpiczt2mxxQ1sImhYy16ImF1N2NaMmFjNjkwnwixhd1lne1v1ntViNw
NLwE4nyr3MhkNdhij0wvwnjk2Y2Q300fUnj12ZTQ3Mtkyj0wXkjgjifQw30h30
4 Cache-Control: max-age=0
5 Authorization: Basic Qmx12wewYwLUokBwauIpTGficzUwNTAh
6 Sec-Ch-Ua: "Chromium";v="103", ".Not/A/Brand";v="99"
7 Sec-Ch-Ua-Mobile: ?0
8 Sec-Ch-Ua-Platform: "Linux"
9 Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/103.0.5060.134 Safari/537.36
1 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=0.9
2 Sec-Fetch-Site: same-origin
3 Sec-Fetch-Mode: navigate
4 Sec-Fetch-User: ?1
5 Sec-Fetch-Dest: document
6 Referer: https://app.chaintrack.ca/login
7 Accept-Encoding: gzip, deflate
8 Accept-Language: en-US,en;q=0.9
9

```

Application Entry Points

An application entry point is a point in a web application at which the execution of the program begins. It is the first code executed when a web app starts running. The entry point is crucial because it determines where the app starts executing and sets the context for the rest of the program. In this project, we decide to use Burp Suite to scan for all URLs in the application and identify all hidden parameters and all pages containing GET and POST requests.

We can go to Target and Site map tab to view the scan. Meanwhile, we need to open the application in the browser. In this case, we can see all the URLs that contain requests and corresponding parameters as well.

URL	Method	Path	Code	Size	Type	Content Type	Notes
https://app.chaintrack.ca	GET	/api/devices/locations	200	958	JSON		
https://app.chaintrack.ca	GET	/dashboard	200	19121	HTML		Dashboard - Chaintrack
https://app.chaintrack.ca	GET	/images/logo.svg	200	8534	XML		
https://app.chaintrack.ca	GET	/js/app.js?id=7c22f182df1...	✓	36054	script		
https://app.chaintrack.ca	GET	/js/markerclusterer/mark...	200	30773	script		
https://app.chaintrack.ca	GET	/js/scripts.bundle.js?id=9...	✓	61139	script		
https://app.chaintrack.ca	GET	/login	200	4533	HTML		Login - Chaintrack
https://app.chaintrack.ca	GET	/svg/cold-chain.svg	200	5863	XML		
https://app.chaintrack.ca	GET	/user/current	200	1251	JSON		
https://app.chaintrack.ca	POST	/login	✓	302	1349	HTML	Redirecting to https://app...
https://app.chaintrack.ca	POST	/login	✓	302	1369	HTML	Redirecting to https://app...
https://app.chaintrack.ca	GET	/					

In Burp Suite, we are able to get all parameters that existing in the request on every page. For instance, in the login page, we can find username, password and session token. If users going through the list on Burp Suite, they can find all parameters they need for each page that contained by the web application.

DNS Trace Process

Domain Name System trace is a process involving following DNS query path from the client computer to the DNS server. The process is helpful in trouble identifying and

troubleshooting issues, also helps in for understanding how DNS queries are handled on the network.

```
└$ dig app.chaintrack.ca +trace
; <>> DiG 9.18.7-1-Debian <>> app.chaintrack.ca +trace
;; global options: +cmd
.          86400  IN      NS      a.root-servers.net.
.          86400  IN      NS      b.root-servers.net.
.          86400  IN      NS      c.root-servers.net.
.          86400  IN      NS      d.root-servers.net.
.          86400  IN      NS      e.root-servers.net.
.          86400  IN      NS      f.root-servers.net.
.          86400  IN      NS      g.root-servers.net.
.          86400  IN      NS      h.root-servers.net.
.          86400  IN      NS      i.root-servers.net.
.          86400  IN      NS      j.root-servers.net.
.          86400  IN      NS      k.root-servers.net.
.          86400  IN      NS      l.root-servers.net.
.          86400  IN      NS      m.root-servers.net.
.          86400  IN      RRSIG   NS 0 518400 20221226170000 20221213160000
xNn0fiJyxVDkOPS/KDNaPp9vJoYkFyzqX5wThBc15P rxxo4e5Uh+Fm2z80Cq+pBrJhjtGfgM9tHK1UbAh9k3/dDpr
7EMRKHWk9aTY73bzLnJwTcERXFYdN2oL/YWoapY+z7 6tnDfnaxqsM9X3Msgvy+J2Vln2S+hT2jp1byabdHB3xdXO
ZTGfvsd/6TVmpPg/FdgpoOowPhrdhScmzXFwko74Grw2 2Aug6ETL+zVert9WeMt2gc0hcd0QnlTgQW02+yLW1DYRD4NC
;; Received 525 bytes from 10.13.37.1#53(10.13.37.1) in 8 ms

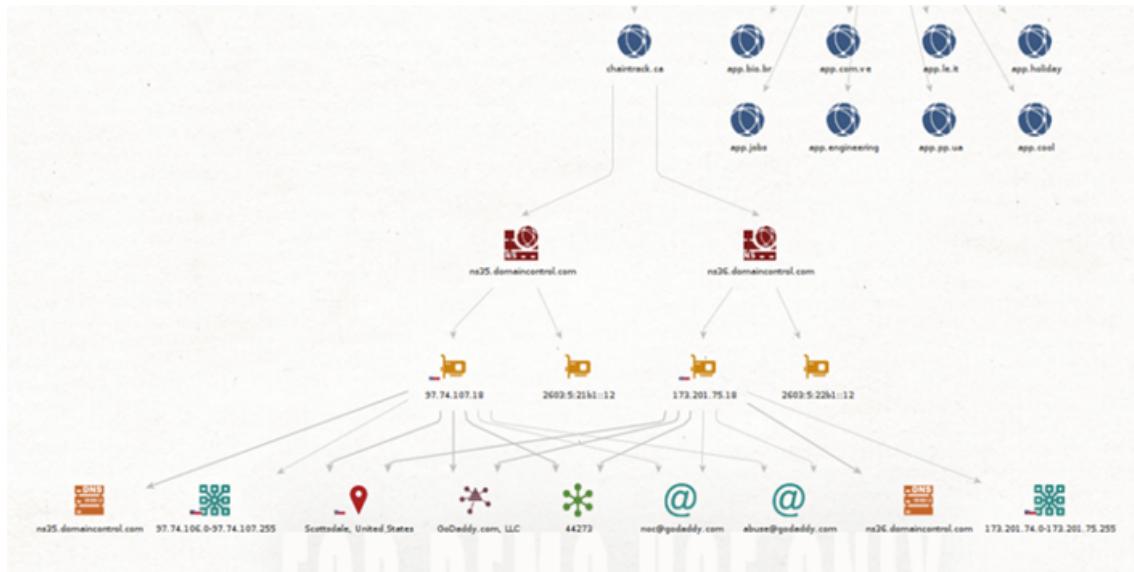
ca.          172800  IN      NS      j.ca-servers.ca.
ca.          172800  IN      NS      x.ca-servers.ca.
ca.          172800  IN      NS      any.ca-servers.ca.
ca.          172800  IN      NS      c.ca-servers.ca.
ca.          86400   IN      DS      43787 8 2 2AF70B49C542B7DACEC2D4754651598B7
96039A2C
ca.          86400   IN      RRSIG   DS 8 1 86400 20221226170000 20221213160000
ZHe8X4B7Bx51znwp40xw87X52uoWA1MbYYd/0/G3J U/DtjvAcxdABZHWKy8Tse49PU9VCvgav+pJVFBhoVF9KvKM
/N/BNmduyuXXVIMs/AlD00kYNUi1u6nyxJN3Xgvy4S /n6HUZs07+azd00U2UNbn3l+/yH1FnxtjgF25k9YnzdWxs25
3DjNyMStJYJG5X+qatFr7ZRvTngd6VA3YBRCANaj 2SlFeI/Gc9KF7/tqcT2cdXgo/A6IN5y0xvjTrzcRpAJL7pU
;; Received 666 bytes from 192.203.230.10#53(e.root-servers.net) in 120 ms

;; UDP setup with 2001:500:a7::2#53(2001:500:a7::2) for app.chaintrack.ca failed: network u
;; UDP setup with 2001:500:a7::2#53(2001:500:a7::2) for app.chaintrack.ca failed: network u
;; UDP setup with 2001:500:a7::2#53(2001:500:a7::2) for app.chaintrack.ca failed: network u
chaintrack.ca. 86400  IN      NS      ns36.domaincontrol.com.
chaintrack.ca. 86400  IN      NS      ns35.domaincontrol.com.
r66k981mhmo0vmpsgv1djat7janroai95.ca. 3600 IN  NSEC3 1 1 0 - R66PG9PTTIK200KT0J69V3IS2M57VEK9
EC3PARAM
```

The trace can help identify issues such as misconfigured nameservers or any network problem between the client and server. There are a number of tools that can help to implement such functions, such as Nmap, Dig, Nslookup, etc. in this case, we use the dig function to trace the DNS, which shows all the steps required to resolve the domain string from the root name server.

Maltego

Maltego is a powerful visualization tool for analyzing web applications, such as intelligence gathering, data mining and link analysis. It is good at demonstrating the relationship between entities such as domains, name servers, IP addresses, etc.



In this case, we can quickly gather information such as any related domains, IP addresses, locations, and email addresses from Maltego. For example, from the image above, we can tell that there are two name servers: ns35.chaintrack.ca and ns36.chaintrack.ca. Furthermore, two IP addresses are assigned to each of them.

2 Configuration and Deployment Management

Test Network Infrastructure Configuration

Test Objectives

- Review the applications' configurations set across the network and validate that they are not vulnerable.
- Validate that used frameworks and systems are secure and not susceptible to known vulnerabilities due to unmaintained software or default settings and credentials.

Steps and results

The tester should be provided with internal information of the software used, including versions and releases used and patches applied to the software.

We can partially do this by using Nessus:

Sev	Score	Name	Family	Count	Actions	
MEDIUM	6.5	HSTS Missing From HTTPS Server (RFC 6797)	Web Servers	2	● Edit	Copy
MEDIUM	5.3	nginx < 1.17.7 Information Disclosure	Web Servers	2	● Edit	Copy
INFO		HyperText Transfer Protocol (HTTP) Redirect Informa...	Web Servers	6	● Edit	Copy
INFO		Nessus SYN scanner	Port scanners	5	● Edit	Copy
INFO		HTTP Methods Allowed (per directory)	Web Servers	4	● Edit	Copy
INFO		HyperText Transfer Protocol (HTTP) Information	Web Servers	4	● Edit	Copy
INFO		Web Application Cookies Not Marked HttpOnly	Web Servers	4	● Edit	Copy
INFO		Web Application Cookies Not Marked Secure	Web Servers	4	● Edit	Copy

According to its Server response header, the installed version of nginx is prior to 1.17.7. It is, therefore, affected by an information disclosure vulnerability.

MEDIUM nginx < 1.17.7 Information Disclosure



Description

According to its Server response header, the installed version of nginx is prior to 1.17.7. It is, therefore, affected by an information disclosure vulnerability.

Solution

Upgrade to nginx version 1.17.7 or later.

See Also

<http://www.nessus.org/u?fd026623>

Output

```
URL : http://app.chaintrack.ca/
Installed version : 1.17.3
Fixed version : 1.17.7
```

NGINX before 1.17.7, with certain error_page configurations, allows HTTP request smuggling, as demonstrated by the ability of an attacker to read unauthorized web pages in environments where NGINX is being fronted by a load balancer.

Recommendation

Upgrade to nginx version 1.17.7 or later.

Test Application Platform Configuration

Test Objectives

- Ensure that defaults and known files have been removed.
- Validate that no debugging code or extensions are left in the production environments.
- Review the logging mechanisms set in place for the application.

Steps and results

No results found. CGI scanning will be disabled for this host because the host responds to requests for non-existent URLs with HTTP code 301 rather than 404.

INFO Web Server No 404 Error Code Check

Description
The remote web server is configured such that it does not return '404 Not Found' error codes when a nonexistent file is requested, perhaps returning instead a site map, search page or authentication page.

Nessus has enabled some counter measures for this. However, they might be insufficient. If a great number of security holes are produced for this port, they might not all be accurate.

Output

```
CGI scanning will be disabled for this host because the host responds
to requests for non-existent URLs with HTTP code 301
rather than 404. The requested URL was :

http://app.chaintrack.ca/RpJFBN5B6oFc.html
```

Port	Hosts
80 / tcp / www	app.chaintrack.ca

Test File Extensions Handling for Sensitive Information

Test Objectives

- Dirburst sensitive file extensions, or extensions that might contain raw data (e.g. scripts, raw data, credentials, etc.).
- Validate that no system framework bypasses exist on the rules set.

Steps and results

The following file extensions should NEVER be returned by a web server, since they are related to files which may contain sensitive information, or to files for which there is no reason to be served.

- .asa
- .inc

Black box testing using google hack:



ext:asa inurl:app.chaintrack.ca



All

Maps

Images

Videos

News

More

Tools

About 0 results (0.26 seconds)

Your search - **ext:asa inurl:app.chaintrack.ca** - did not match any documents.

Suggestions:

- Make sure all words are spelled correctly.
- Try different keywords.
- Try more general keywords.
- Try fewer keywords.



ext:inc inurl:app.chaintrack.ca



All

Maps

Images

Videos

News

More

Tools

About 0 results (0.18 seconds)

Ad • <https://www.wherethere.com/traceability> ▾

[Supply chain traceability software - Modern Food Production...](#)

Lot track/trace, inventory control & costing, manage customer orders & supplier purchases

Your search - **ext:inc inurl:app.chaintrack.ca** - did not match any documents.

Suggestions:

- Make sure all words are spelled correctly.
- Try different keywords.
- Try more general keywords.
- Try fewer keywords.

No results found.

Enumerate Infrastructure and Application Admin Interfaces

Test Objectives

- Identify hidden administrator interfaces and functionality.

Steps and results

Black box testing using OWASP ZAP:

Forced Browse

Penetration Test Report - Chaintrack Company



Req. Timestamp	Resp. Timestamp	User	Method	URL	Code	Reason	Size Resp. Header	Size Resp. Body
11/12/2022, 15:10:09	11/12/2022, 15:10:09		GET	https://app.chaintrack.ca:443/	200 OK	1,026 bytes	394 bytes	
11/12/2022, 15:10:10	11/12/2022, 15:10:10		GET	https://app.chaintrack.ca:443/images/	403 Forbidden	170 bytes	153 bytes	
11/12/2022, 15:10:10	11/12/2022, 15:10:10		GET	https://app.chaintrack.ca:443/teams/	302 Found	1,028 bytes	370 bytes	
11/12/2022, 15:10:10	11/12/2022, 15:10:10		GET	https://app.chaintrack.ca:443/login	401 Unauthorized	217 bytes	179 bytes	
11/12/2022, 15:10:24	11/12/2022, 15:10:24		GET	https://app.chaintrack.ca:443/admin/	302 Found	1,028 bytes	394 bytes	
11/12/2022, 15:10:26	11/12/2022, 15:10:26		GET	https://app.chaintrack.ca:443/admin/login	401 Unauthorized	217 bytes	179 bytes	
11/12/2022, 15:10:29	11/12/2022, 15:10:29		GET	https://app.chaintrack.ca:443/register/	200 OK	1,001 bytes	0 bytes	
11/12/2022, 15:10:33	11/12/2022, 15:10:33		GET	https://app.chaintrack.ca:443/icons/	403 Forbidden	170 bytes	153 bytes	
11/12/2022, 15:10:46	11/12/2022, 15:10:46		GET	https://app.chaintrack.ca:443/devices/	302 Found	1,026 bytes	370 bytes	
11/12/2022, 15:10:49	11/12/2022, 15:10:49		GET	https://app.chaintrack.ca:443/dashboard/	302 Found	1,030 bytes	370 bytes	
11/12/2022, 15:10:49	11/12/2022, 15:10:49		GET	https://app.chaintrack.ca:443/login/	401 Unauthorized	217 bytes	179 bytes	

There is an admin page app.chaintrack.ca/admin/login which can be accessed from the public Internet:

Recommendation

Administration interfaces are an attack surface. Only legitimate administrators should be able to communicate with them. You should isolate these interfaces using architectural controls and constrain who can connect to the system and from where. This will help to protect against attacks such as brute forcing administrator login, or using an exploit to gain access.

There are a number of ways that you can reduce the exposure of your management interfaces. For example:

- You could create a dedicated management network that only authorized administrators have physical access to.
- You could place your administration interfaces behind a VPN that only authenticated administrators and devices can use.
- You could implement an IP allow list to restrict the devices or networks that can access the administration interface.

Test HTTP Strict Transport Security

Test Objectives

- Review the HSTS header and its validity.

Steps and results

Using Burpsuite:

```

Medium
Request Response
Pretty Raw Hex Render
1 HTTP/2 302 Found
2 Server : nginx/1.17.3
3 Content-Type : text/html; charset=UTF-8
4 Location : https://app.chaintrack.ca/login
5 Cache-Control : no-cache, private
6 Date : Tue, 13 Dec 2022 05:29:33 GMT
7 Set-Cookie : XSRF-TOKEN =
eyJpdii6ImVwUEJRUsdVRXVNkWODg1TmloVke9PSIsInZhHVlIjoic05PbHYzcUlZk8zTVww
U0tMdEhETnFlQVNSMzVloss4ZE96bmlkMXR3UGcwbE04QjBPTCtvTlFMU1AyYjJQeVgiLCJtYWMi
Oii2YjQ3MjU0ZTY1NDI3MzdhMzc3YmUyYTMzYzYzNGYyN2IyYzk1Zjc4NWRLOGU4ZDBjMmNjMzKx
MGIxOWMwODY1In083D ; expires=Tue, 13-Dec-2022 07:29:33 GMT; Max-Age=7200;
path=/
8 Set-Cookie : chaintrack_session =
eyJpdii6IndtclwwMHhUcjNXRlZxRFMyOExDtWtRPT0iLCJ2YWx1ZSI6InFPYnhqYWoxtFBRRVhn
WUlsNVpvXC9TNUpUVCTOSUpQeEplYjFKVG9LZURtckliTVFnZW9sUFJkcWVuYmJkODMrIiwbWFj
IjoimDQxMWZiY2Ziy2FlyWWiN2RiyzliOGPmMjAxNjUzOWQ2NTJhNTA1OTBkYjk2ZDU1NTQ5Nzli
NzPmNGUzZDF1NCJ9 ; expires=Tue, 13-Dec-2022 07:29:33 GMT; Max-Age=7200;
path=/; httponly
9 X-Frame-Options : SAMEORIGIN
10 X-Xss-Protection : 1; mode=block
11 X-Content-Type-Options : nosniff
12
13 <!DOCTYPE html>
14 <html>
.
.
.

```

The response shows that the remote web server is not enforcing HSTS, which means the application fails to prevent users from connecting to it over unencrypted connections. An attacker able to modify a legitimate user's network traffic could bypass the application's use of SSL/TLS encryption, and use the application as a platform for attacks against its users. This attack is performed by rewriting HTTPS links as HTTP, so that if a targeted user follows a link to the site from an HTTP page, their browser never attempts to use an encrypted connection. The sslstrip tool automates this process.

To exploit this vulnerability, an attacker must be suitably positioned to intercept and modify the victim's network traffic. This scenario typically occurs when a client communicates with the server over an insecure connection such as public Wi-Fi, or a corporate or home network that is shared with a compromised computer. Common defenses such as switched networks are not sufficient to prevent this. An attacker situated in the user's ISP or the application's hosting infrastructure could also

perform this attack. Note that an advanced adversary could potentially target any connection made over the Internet's core infrastructure.

Recommendation

The application should instruct web browsers to only access the application using HTTPS. To do this, enable HTTP Strict Transport Security (HSTS) by adding a response header with the name 'Strict-Transport-Security' and the value 'max-age=expireTime', where expireTime is the time in seconds that browsers should remember that the site should only be accessed using HTTPS. Consider adding the 'includeSubDomains' flag if appropriate.

Note that because HSTS is a "trust on first use" (TOFU) protocol, a user who has never accessed the application will never have seen the HSTS header, and will therefore still be vulnerable to SSL stripping attacks. To mitigate this risk, you can optionally add the 'preload' flag to the HSTS header, and submit the domain for review by browser vendors.

3 Identity Management

Testing for default credentials

Test Objectives:

- Try default usernames such as admin, administrator, root, system, guest, operator, and superuser. Use burp intruder to send automated requests with default credentials, brute forcing the login API.

Burp Project Intruder Repeater Window Help

Dashboard Target Proxy **Intruder** Repeater Sequencer Decoder Comparer Logger Extensions Learn

12 × 13 × 14 × +

Positions **Payloads** Resource Pool Options

② Payload Sets

You can define one or more payload sets. The number of payload sets depends on the attack type defined in the Positions tab. Various payload

Payload set: 1 Payload count: 1,041
Payload type: Simple list Request count: 1,041

② Payload Options [Simple list]

This payload type lets you configure a simple list of strings that are used as payloads.

Paste	ABCD
Load ...	ACCESS
Remove	ACCORD
Clear	ADLDEMO
Deduplicate	ADMIN
Add	ADMINISTRATOR
	AIRPLANE
	ALLIN1
Add from list ... [Pro version only]	

② Payload Processing

You can define rules to perform various processing tasks on each payload before it is used.

Add	Enabled	Rule
Edit		
Remove		
Up		
Down		

② Payload Encoding

This setting can be used to URL-encode selected characters within the final payload, for safe transmission within HTTP requests.

URL-encode these characters: `\=;<>?+&*;"\|^~#`

No successful login:

Request	Response	Payload	Status	Error	Timeout	Length	Comment
uc	302	APPUSER	302			1341	
83	302	AQ	302			1345	
84	302	AQDEMO	302			1343	
85	302	AQJAVA	302			1343	
86	302	AQUSER	302			1345	
87	302	ARAdmin#	302			1343	
88	302	ARCHIVIST	302			1341	
89	302	AUDIOUSER	302			1343	
90	302	AWARD_SW	302			1341	
91	302	AWARD7SW	302			1341	
92	302	AWARD_PW	302			1345	
93	302	AWARD_SW	302			1341	
94	302	Admin	302			1345	
95	302	Admin1	302			1347	
96	302	Administrative	302			1345	
97	302	Administrator	302			1343	
98	302	Advance	302			1347	
99	302	Alraya	302			1351	
100	302	Any	302			1343	
101	302	Award	302			1339	
102	302	BACKUP	302			1343	
103	302	BRIDGE	302			1343	
104	302	BASE	302			1343	
105	302	BATCH	302			1341	
106	302	BC4J	302			1351	
107	302	BIGO	302			1343	
108	302	BIOS	302			1345	
109	302	BIOSPASS	302			1341	
110	302	BRIO_ADMIN	302			1343	
111	302	Babylon	302			1343	
112	302	BackupUSR	302			1345	
113	302	Barricade	302			1343	
114	302	Biostar	302			1343	
115	302	CAROLIAN	302			1347	
116	302						
<hr/>							
Request	Response	Pretty	Raw	Hex			
1	POST /login HTTP/1.1	POST /login HTTP/1.1					
2	Host: app.chaintrack.ca						
3	Cookie: XSRF-TOKEN=eyJhbGciOiJIUzI1NiIsInR5cGgiOiJsb2dpbiIsInRpdj6InstvR0JCSU01MnHJNVT1MSOX1STK1BMRKE9PSIaInZhhHV11jo1ODNLUsZnUnZBbz20cUNEdBhclLzcL2yZsOMJMWhc4TmZrZldSHM1NFFYYWhndEFaZzJwh3bJ0EicL2g3YWRxdnfseaiiSmihYvI6IjAwNTVvNDKXMDU3Yjeypd1fIInd0DFBwV3ptVnpOMD0uSXFOQoLzQWc9PSImljZnbhHV11jo1R0JCaXg4d0pZWY3VVJ6NVZQWhRjHy2ZzMWVbVktiaFRxNipVdithXC9P20V2N1RFXC9XcsRtVsVDT1iizWb11ivWFj1jo1YTc1zDE32DqyNTg1zDRhYtz21YjkzYjPmNmUSM						
4	Content-Length: 76						
5	Content-Type: application/x-www-form-urlencoded						
6	Authorization: Basic Cmxi2ZNojYWh1OkRwUiUpTGFiwsUwHTAh						
7	Sec-Ch-Da: "Not ?>, Brand";v="8", "Chromium";v="108"						
8	Sec-Ch-Da-Mobile: ?0						
9	Sec-Ch-Da-Platform: "Windows"						
10	Upgrade-Insecure-Requests: 1						
11	Origin: https://app.chaintrack.ca						
12	Content-Type: application/x-www-form-urlencoded						
13	User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/100.0.5356.65 Safari/537.36						
14	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=0.9						
15	Sec-Fetch-Site: same-origin						
16	Sec-Fetch-Mode: navigate						
17	Sec-Fetch-User: ?1						
18	Sec-Fetch-Dest: document						
19	Referer: https://app.chaintrack.ca/login						
20	Accept-Encoding: gzip, deflate						
21	Accept-Language: zh-CN,zh;q=0.9						
22	Connection: close						
23							
24	_token=50jxrVNVzhb4hVnBggdnVBbBwxB015YTY9GEeVGIGemail=Admin1&password=Admin1						

Result:

Low

The app passed the test, but no lockout, and no CAPTCHA challenge.

Recommendation:

Add lockout IP mechanism and CAPTCHA challenges after a certain number of failed login attempts.

4 Authorization

Encrypted Channel

Test Objectives:

- Black Box Testing
- To test if the App uses HTTPS to send login parameters, such as the user name, password, and challenge answers.

EMAIL ADDRESS

lca160@sfu.ca

PASSWORD

 Remember Me

Login

Check the login request using HTTPS and POST-method.

208 ↗ Proxy	12/12/2022, 16:53:09	GET	https://content-autofill.googleapis.com/v1/pages/ChRDaHJvbWUvMTA4LjAuNTM1...	200 OK	111 ms	112 by
209 ↗ Proxy	12/12/2022, 16:53:16	POST	https://app.chaintrack.ca/login	302 Found	285 ms	386 by
210 ↗ Proxy	12/12/2022, 16:53:17	GET	https://app.chaintrack.ca/login?token=HhpGiEpxCzbxfWP9wwe4yNzDpMcIUh4xKrB7vSmr&email=lca160%40sfu.ca&password=sfusecurity	200 OK	220 ms	10.160

```

Header: Text Body: Text
POST https://app.chaintrack.ca/login HTTP/1.1
Host: app.chaintrack.ca
Connection: keep-alive
Content-Length: 90
Cache-Control: max-age=0
Authorization: Basic Qmx1ZWNoYluOkBWaU1pTGFiczuWNTAh
sec-ch-ua: "Not A Brand";v="8", "Chromium";v="108", "Google Chrome";v="108"
sec-ch-ua-mobile: ?0
sec-ch-ua-platform: "Windows"
Upgrade-Insecure-Requests: 1
Origin: https://app.chaintrack.ca
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/108.0.0.0 Safari/537.36
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=0.9
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Referer: https://app.chaintrack.ca/login
Accept-Language: zh-CN,zh;q=0.9
Cookie: XSRF-TOKEN=
eyJpdigIkiYWGFMk53U1LZ0lybWpwcllmd1E9PSIsInZhbHVlIjo1NE41dwg0NmhcL0FvelwvaEZRWXc2U0x3Zw5JN0FyOHp2wCtnNlFteXRcl21DYTJBV1pMU3VaME5NVVF0SDdZelppdiIsI
mihYyI6IjRKY2E3ZGE1MDvhNzEwMjkzMzBhYmRhNDZjMWZj0WQ3ZGQ3YjFhMjA4NDMyNTgwZjRjOWI3ODhjZjcyMmMwNGMifQ%3D%3D; chaintrack_session=
eyJpdigI6In12Q2NaTUc0ZFTZ2toa2YzRHpLeUE9PSIsInZhbHVlIjoicVpvS1JxdmJSOWJMeitDWXQrUjNmRFJXQWxnN0lFT2FMN1QrelwvYis0VHM2czdDdmpjWnRLTFwvakdLd1ZNY29MIiwib
_token=HhpGiEpxCzbxfWP9wwe4yNzDpMcIUh4xKrB7vSmr&email=lca160%40sfu.ca&password=sfusecurity

```

Result:

We can see that the app use HTTPS send the login parameters indeed. And the referrer is also HTTPS.

Testing for Weak lockout mechanism

Test Objectives:

- The objective of testing for weak lockout mechanisms is to ensure that the system being tested has adequate security measures in place to prevent unauthorized access and protect against brute force attacks.
- A weak lockout mechanism is a security feature that is designed to block access to a system or account after a certain number of failed login attempts. This is intended to prevent malicious actors from guessing passwords or using automated tools to try multiple combinations in order to gain access to a system.
- However, if the lockout mechanism is weak, it may not be effective at preventing unauthorized access, as it may allow too many failed login attempts before triggering a lockout. This can make the system vulnerable to brute force attacks, where an attacker can try a large number of different passwords in a short period of time in order to gain access.
- To test for weak lockout mechanisms, a tester may attempt to access a system or account using multiple incorrect passwords in rapid succession, and observe whether the lockout mechanism is triggered and how long it takes to do so. The tester may also attempt to bypass the lockout mechanism by using different IP addresses or other techniques.



LOGIN AS ADMINISTATOR OR DISPATCHER

EMAIL ADDRESS

lcal61@sfu.ca



Too many login attempts. Please try again in 28 seconds.

PASSWORD

Password

Remember Me

Login

Always match Chrome's language [Switch DevTools to Chinese](#) [Don't show again](#)

Console Sources Network Performance Memory Application Security Lighthouse

Preserve log Disable cache No throttling Invert Hide data URLs **All** Fetch/XHR JS CSS Img Media Font Doc WS Wasm Manifest Other Has blocked cookies Blocked Requests 3rd-party requests

40 ms 60 ms 80 ms 100 ms 120 ms 140 ms 160 ms 180 ms 200 ms 220 ms 240 ms 260 ms

	Headers	Preview	Response	Initiator	Timing	Cookies
20	</nav>					
29	</header>					
30						
31						
32						
df1d3ec06...	<div class="container">					
:300,400,5...	<div class="col-2 mx-auto d-block mt-9">					
215e5724a...						
Dz8Z1xF...						
fecg.woff2						
3T9Z1xF...	</div>					
Ej6Z1xFQ...	<div class="h6 pt-5 small d-block text-center text-uppercase">Login as administrator or dispatcher</div>					
l:...	<div class="row no-gutters">					
	<div class="col-md-8 col-lg-7 col-xl-6 offset-md-2 offset-lg-2 offset-xl-3 u-space-2">					
44	<form action="https://app.chaintrack.ca/login" method="POST" role="form">					
45	<input type="hidden" name="_token" value="20jzrVNVEhb4hVnDggdnVBdBwx015YEY8GEkWGIG">					
46	<!-- E-Mail Address -->					
47	<div class="form-group">					
48	<label for="email" class="h6 small d-block text-uppercase">Email Address</label>					
49						
50	<input id="email" type="email" class="form-control is-invalid" name="email" value="lcal61@sfu.ca" placeholder="Email Address" autofocus>					
51						
52						
53	Too many login attempts. Please try again in 28 seconds.					
54						
55	</div>					
56						
57	<!-- Password -->					
58	<div class="form-group">					
59	<div class="d-flex justify-content-between align-items-center">					
60	<label for="password" class="h6 small d-block text-uppercase">Password</label>					

Result:

Locked out 60s after 5 failed attempts, **but no lock**.

There is no CAPTCHA challenge.

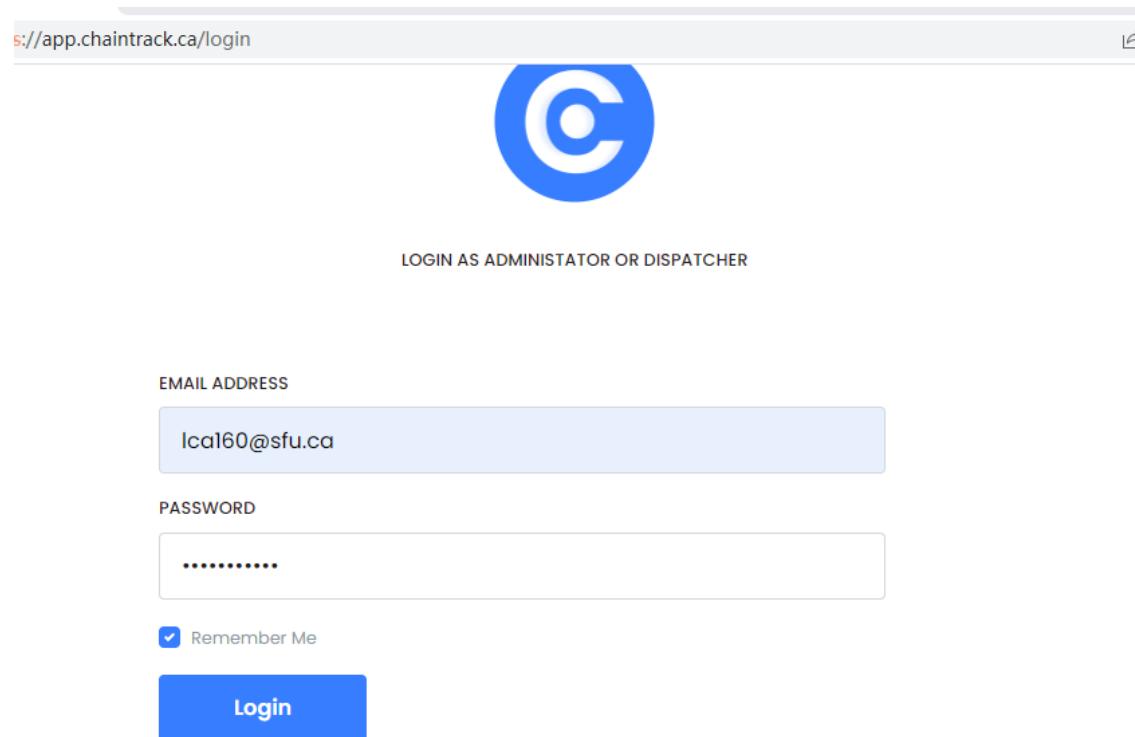


Recommendation:

Add a permanent lockout mechanism and CAPTCHA challenges after a certain number of failed login attempts.

Test remember password functionality

Look for passwords being stored in a cookie. Examine the cookies stored by the application. Verify that the credentials are not stored in clear text, but are hashed.



The screenshot shows a web browser window with the URL <http://app.chaintrack.ca/login>. At the top center is a large blue circular logo containing a white letter 'C'. Below the logo, centered text reads "LOGIN AS ADMINISTRATOR OR DISPATCHER". The main form area has two input fields: "EMAIL ADDRESS" containing "lca160@sfu.ca" and "PASSWORD" containing several dots. There is a checked checkbox labeled "Remember Me" and a blue "Login" button at the bottom.

Response

Pretty	Raw	Hex	Render	Ir
R	R	R	R	R
R	R	R	R	R
R	R	R	R	R

```

1 HTTP/2 302 Found
2 Server: nginx/1.17.3
3 Content-Type: text/html; charset=UTF-8
4 Location: https://app.chaintrack.ca/dashboard
5 Cache-Control: no-cache, private
6 Date: Tue, 13 Dec 2022 05:41:13 GMT
7 Set-Cookie: XSRF-TOKEN=
eyJpdIiE1nfRlkidhodWNRZmxtVOVMVd2ZHc9PSIsIn2hbHV1IjoY1f6cTvnaXpSV1dtVGs3cVB0V29EY2FTSOOrN1R1OWJUUkFDVUxIaWgwckhoSDgzcnsBSZ
TBsbhNiZjFXdmxrSSIsImhiYi6ImViMzAxNjA2MjEx2TdhNWFkYzQOYWY4YjNkMjNjOTEONWM0ZGU1Yzg2Nzh1YTE2MmZ1ZTU1MGRhZDE3NzKxZWYifQ43D43D;
expires=Tue, 13-Dec-2022 07:41:13 GMT; Max-Age=7200; path=/
8 Set-Cookie: chaintrack_session=
eyJpdIiE1ndXVN3Zm42UHJ2chH21T1BhY3c30Xc9PSIsIn2hbHV1Ijo1MFR2S2SLd3VcL3ZadDNFdTFwahINVE3dENBY1PVZWTNWUlqamfpYU9CSGISZ2tUMjB4T
HB1Xc9ZTYox1ZnNmWW2RIiwbWFjIjo1ZDFjOGNmNmZiOGEMjNkOTk4ODMsNTElMCNNDyvYjZkYzZhNzdjZGJiMjcxYzgiMmQxY2M30ThhOTN1Y2M4ZCJ9;
expires=Tue, 13-Dec-2022 07:41:13 GMT; Max-Age=7200; path=/; httponly
9 Set-Cookie: remember_web_59ba36adde2b1f9401580f014c7f58ea4e30589d=
eyJpdIiE1k1CVjgrcJBqNDRDefpHVUcqBGrclE9PSIsIn2hbHV1Ijo1NE1sU1fMWj1s2k2QzVnVEloUmRoTytsoJMdHhZeHExOEHtWWdMMkZnVgtKNDRRSU1ob
itOM2JQahNUUVTVTU0dER2CCEUWNJNNU4WWxremiLaERndmfjT3BjUFNQVTBXeH1cL0pYT1cyMoOzdfI2ZNbEFTNTVdu02FMXhcL1M1b214M35gk1VsavBBVG
p3a3Z3eExmejh1hRFyNHBkdjJ5MkoxRmxCOKM91iwbWFjIjo1ZTF1NjNkNmZYTcwNTMxNTMxOtc1MTi0tM1NDMSMa2NGQ5Nzcc5YjBiZTc40WNkMzExMjNjNGQ
2ZjZjOTY2ZSJ9; expires=Sun, 12-Dec-2027 05:41:13 GMT; Max-Age=157680000; path=/; httponly
10 X-Frame-Options: SAMEORIGIN
11 X-Xss-Protection: 1; mode=block
12 X-Content-Type-Options: nosniff
13
14 <!DOCTYPE html>
15 <html>
16   <head>
17     <meta charset="UTF-8" />
18     <meta http-equiv="refresh" content="0;url='https://app.chaintrack.ca/dashboard'" />
19
20   <title>
    Redirecting to https://app.chaintrack.ca/dashboard
  </title>
21 </head>
22 <body>
23   Redirecting to <a href="https://app.chaintrack.ca/dashboard">
    https://app.chaintrack.ca/dashboard
  </a>
24 </body>
25 </html>
```

Result: The password is stored in cookies and hashed.

Examine the hashing mechanism

Examine the hashing mechanism: if it is a common, well-known algorithm, check for its strength, and its homegrown hash functions, and attempt several usernames to check whether the hash function is easily guessable.

Hash reverse lookup, unhash, decrypt, search

Hash type

Md5

Hash String

eyJpdii6Ik1CVjgrobqNDRDeFpHVUtqbEgro1E9PSIsInZhbHV

[Enable mass-decrypt mode](#)

Provided hash doesn't match Md5 bitmap. Are you sure it is Md5? If not - try "Search by all hash types" option.

Hash reverse lookup, unhash, decrypt, search

Hash type

Tiger128

Hash String

HBkdjJ5MkoxRmtCQXM9IiwibwFjIjoizTFiNjNkNmMzYTowNTM:

[Enable mass-decrypt mode](#)

Hash reverse lookup, unhash, decrypt, search

Hash type

Sha256

Hash String

eyJpdii6Ik1CVjgrobqNDRDeFpHVUtqbEgro1E9PSIsInZhbHV

[Enable mass-decrypt mode](#)

Provided hash doesn't match Sha256 bitmap. Are you sure it is Sha256? If not - try "Search by all hash types" option.

Hash reverse lookup, unhash, decrypt, search

Hash type	Sha512
Hash String	HBkdjJ5MkoxRmtCQXM9IiwibWFjIjoizTFiNjNkNmMzYTowNTM:

Enable mass-decrypt mode

Result:

The hash is not breakable by simple methods, which means it is not vulnerable to hash collision attacks:

Verify that the credentials are only sent during the log-in phase and not sent together with every request to the application.

Penetration Test Report - Chaintrack Company

<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td style="width: 10%;">392</td><td style="width: 30%;">https://maps.googleapis.com</td><td style="width: 10%;">GET</td><td>/maps/vt:pb->1114;1113;1124;112;1114;1113;1124;113;1114;1113;1121;112;1114;1113;1124</td></tr> <tr><td>396</td><td>https://app.chaintrack.ca</td><td>GET</td><td>/devices</td></tr> <tr><td>401</td><td>https://app.chaintrack.ca</td><td>GET</td><td>/user/current</td></tr> <tr><td>402</td><td>https://app.chaintrack.ca</td><td>GET</td><td>/api/devices?page=1</td></tr> <tr><td>403</td><td>https://use.fontawesome.com</td><td>GET</td><td>/releases/v5.0.13/webfonts/fa-solid-900.woff2</td></tr> </table>				392	https://maps.googleapis.com	GET	/maps/vt:pb->1114;1113;1124;112;1114;1113;1124;113;1114;1113;1121;112;1114;1113;1124	396	https://app.chaintrack.ca	GET	/devices	401	https://app.chaintrack.ca	GET	/user/current	402	https://app.chaintrack.ca	GET	/api/devices?page=1	403	https://use.fontawesome.com	GET	/releases/v5.0.13/webfonts/fa-solid-900.woff2
392	https://maps.googleapis.com	GET	/maps/vt:pb->1114;1113;1124;112;1114;1113;1124;113;1114;1113;1121;112;1114;1113;1124																				
396	https://app.chaintrack.ca	GET	/devices																				
401	https://app.chaintrack.ca	GET	/user/current																				
402	https://app.chaintrack.ca	GET	/api/devices?page=1																				
403	https://use.fontawesome.com	GET	/releases/v5.0.13/webfonts/fa-solid-900.woff2																				
Request		Response																					
Pretty Raw Hex		Pretty Raw Hex Render																					
<pre> 1 GET /devices HTTP/2 2 Host: app.chaintrack.ca 3 Cookie: remember_web_59ba36addc2b2f9401580f014c7f58ea4e3098 9d= eyJpdiI6Ik9BTmVdaSs3bjZ5ZYU3JuallDNnc9PSIsInZhbHV 1Ijo1YW9wZzVys2RpKzdpVEROWHLLWGFwBGErazI30WtcLzYwZz Y1UjBcLoOzQ3NKTjVfeDBFQ1RTUK5nVHpDRkjhTdhPNXBrdFnue FZkSmZQWnUybWh3aw12cTN2MmdkTzR1dEVQSiwND8oQ1vvTVh NWk0Q3dmdWRTNm1UkxMcEY4T2hzMitLcXpPSXhUem93WWNWeFY 1bzI2SUR1dVMzcE5jMEJUZGpVVFZwOXbcL1U9IiwbWFjIjo1N2 PhOGQzM2JkMDE5YWR1NTKuYm5MjdjMjU2NjIyMDM4Mzc0MW1wZ WZhYTUSNGMyYzN1NWQ42TjmMmJjZWQwZSJ9; XSRF-TOKEN= eyJpdiI6Ilwvako2VWdtZ3hDvRUCzBldW9EMGtBPT0iLCJ2YWx 1ZS16I1ZYMDBmRng3WDBPwmSkUmuYMTBzUWJoWUxOSO4rS11eV vvZD2tWXVXQjdFVXFcL1BROxpwaU4ajNRQ3Zke1U3NyIsImlhYy y16IjBRNjdi2Td1nZdkZDAOZTNhNDBiNmQ2NzUwYZkZTAzZmE2 ZDE0ZV1NT13Mm14OTdjNzIyNTBjNWFjNzgxZmQifQ+3D+3D; chaintrack_session= eyJpdiI6ImoHdWxzWHRFRREVIRWNSeY0Z1dqEc9PSIsInZhbHV 1Ijo1ZkoyYXMsTVJcnuhDUD0eWp6djBUDvNNt2ZpQ1NmVnN4ae dMTEjER3VuakwODDR1M1I1Ymh0cDFDehNFckUyNyIsImlhYyI6I mV1MDQ2MGY02DQ2NTR1NjZkM2VknjYx2DA32TVMymJmMv1ZDU4 OTYzNGQzMzk0NWN1lMm2njNTkwZYyNWM1MzQifQ+3D+3D; 4 Authorization: Basic Qmx1ZWN0YWluOkBwaUipTGFiczuwNTAh 5 Sec-Ch-Ua: "Not? A Brand";v="8", "Chromium";v="108" 6 Sec-Ch-Ua-Mobile: ?0 7 Sec-Ch-Ua-Platform: "Windows" 8 Upgrade-Insecure-Requests: 1 9 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/108.0.5359.95 Safari/537.36 10 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=0.9 11 Sec-Fetch-Site: same-origin 12 Sec-Fetch-Mode: navigate 13 Sec-Fetch-User: ?1 14 Sec-Fetch-Dest: document 15 Referer: https://app.chaintrack.ca/dashboard 16 Accept-Encoding: gzip, deflate </pre>																							
<input type="button" value="①"/> <input type="button" value="⚙"/> <input type="button" value="↶"/> <input type="button" value="↷"/> <input type="text" value="Search..."/>		<input type="button" value="①"/> <input type="button" value="⚙"/> <input type="button" value="↶"/> <input type="button" value="↷"/> <input type="text" value="Search..."/>																					
0 matches		0 matches																					

<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td style="width: 10%;">392</td><td style="width: 30%;">https://maps.googleapis.com</td><td style="width: 10%;">GET</td><td>/maps/vt:pb->1114;1113;1124;112;1114;1113;1124;113;1114;1113;1121;112;1114;1113;1124</td></tr> <tr><td>396</td><td>https://app.chaintrack.ca</td><td>GET</td><td>/devices</td></tr> <tr><td>401</td><td>https://app.chaintrack.ca</td><td>GET</td><td>/user/current</td></tr> <tr><td>402</td><td>https://app.chaintrack.ca</td><td>GET</td><td>/api/devices?page=1</td></tr> <tr><td>403</td><td>https://use.fontawesome.com</td><td>GET</td><td>/releases/v5.0.13/webfonts/fa-solid-900.woff2</td></tr> </table>				392	https://maps.googleapis.com	GET	/maps/vt:pb->1114;1113;1124;112;1114;1113;1124;113;1114;1113;1121;112;1114;1113;1124	396	https://app.chaintrack.ca	GET	/devices	401	https://app.chaintrack.ca	GET	/user/current	402	https://app.chaintrack.ca	GET	/api/devices?page=1	403	https://use.fontawesome.com	GET	/releases/v5.0.13/webfonts/fa-solid-900.woff2
392	https://maps.googleapis.com	GET	/maps/vt:pb->1114;1113;1124;112;1114;1113;1124;113;1114;1113;1121;112;1114;1113;1124																				
396	https://app.chaintrack.ca	GET	/devices																				
401	https://app.chaintrack.ca	GET	/user/current																				
402	https://app.chaintrack.ca	GET	/api/devices?page=1																				
403	https://use.fontawesome.com	GET	/releases/v5.0.13/webfonts/fa-solid-900.woff2																				
Request		Response																					
Pretty Raw Hex		Pretty Raw Hex Render																					
<pre> 1 HTTP/2 200 OK 2 Server: nginx/1.17.3 3 Content-Type: text/html; charset=UTF-8 4 Vary: Accept-Encoding 5 Cache-Control: no-cache, private 6 Date: Tue, 13 Dec 2022 07:40:02 GMT 7 Set-Cookie: XSRF-TOKEN= eyJpdiI6IkRoAgtCQ1b5zY1AwSUdhN1PxYaGc9PSIsInZhbH V1Ijo1ZG9kWkVmZuhnsT2pWa3xyVDJJQk1UbzMcTVhOXRhZEZc L01DbVBVFpSdUpVdm5YMTJTNGrOeXBKcDNYzFwvbUUxiwiwB FjIjo1OWNhNzF1MjkzMmYzNmI5Y2UxOFjM2E3YzU4NjMOOGjj Mjg3YjFjYTU2NWY2YjBjY2NKYzcxYTl2NjhjNjcxZiJ9; expires=Tue, 13-Dec-2022 09:40:02 GMT; Max-Age=7200; path=/ 8 Set-Cookie: chaintrack_session= eyJpdiI6IkQ1ZXLSWk03WDJkSGxYVjNrEnF5NGc9PSIsInZhbH V1IjoiaFZGS1Ui1TWSVHFhWDkzTnI3a09uSExieG9IVDJackx4 ZnhFnytrVUpValbR0kweHNOSHZnWVY5Qk12NuUtvbSISimhYY I6IjZiY2F1YTksNG1LYzdKZDQ5Nzcs5NzV12DIx0WQ1ZDcwYj1j MzAzNzczYWH0ZWYxZDhmNDgONCEONmE4NTczNjKifQ+3D+3D; expires=Tue, 13-Dec-2022 09:40:02 GMT; Max-Age=7200; path=/; httponly 9 X-Frame-Options: SAMEORIGIN 10 X-Xss-Protection: 1; mode=block 11 X-Content-Type-Options: nosniff 12 13 <!DOCTYPE html> 14 <html lang="en"> 15 <head> 16 <meta charset="utf-8" /> 17 <meta http-equiv="X-UA-Compatible" content="IE=edge"> 18 <meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1, shrink-to-fit=no"> 19 20 <title> 21 IoT Devices - Chaintrack 22 </title> 23 <meta name="description" content="Get real-time monitoring for your temperature sensitive products."> 24 <meta name="author" content="Chaintrack"> </pre>																							
<input type="button" value="①"/> <input type="button" value="⚙"/> <input type="button" value="↶"/> <input type="button" value="↷"/> <input type="text" value="Search..."/>		<input type="button" value="①"/> <input type="button" value="⚙"/> <input type="button" value="↶"/> <input type="button" value="↷"/> <input type="text" value="Search..."/>																					
0 matches		0 matches																					

Request	Response
<pre> 1 GET /user/current HTTP/2 2 Host: app.chaintrack.ca 3 Cookie: 4 remember_web_59ba36addc2b2f9401580f014c7f58ea4e3098 9d= eyJpdiI6Ik9BTmVDAss3bjZ52VY2U3JUallDNnc9PSIsInZhbHV 1ijoiW9wZzVyS2RpKzdpVEROWH1LWGFbGErazI30WtcLzYwZz Y1UjBcL0ozQSNKTjV6eDBFQ1RTUK5nVHpDRkJnTdhRNXBrdFNue F2kSmZQWnUybWh3aw1zCtn2MmdkTzRIdEVQS1vwNDBoQ1vWTVhh NWkO03dmwRRTtN1UkMcEY4T2hzMitLcXpPSXhUem83WWNWeFY 1bzI2SUR1dVMzcE5jMEJU2GpVVFTw0ZBcL1U9IiwiWFjIjojN2 RhOGQz2MjKMD5YWRiNTNkYm5MjdjMjU2NjIyMDM4MzcOMWlZ W2hYTUSNGMyYzN1NWQ4TJmMmjzZWQwZSJ9; XSRF-TOKEN= eyJpdiI6IkROaGtQcUQ1bk5zY1AwSUdhN1RXaGc9PSIsInZhbHV 1ijoiZG9kWkVmZUhst2pWaaxyVDJJQk1UbzM4cTVhOXRhZEZcL0 1DbBVVFpSdUpVdm5YMTJTNGRoeXKbCdnYzfWvbUUxIiwiWFjI jojOWnhNzF1MjkzNmYzNm15Y2UxODFjM2E3YzU4NjMOOGUjMjg2 YjFjYTUCNWyYjBjY2NkYzcxYT12NhjNjcxZiJ9 5 Authorization: Basic Qmx1ZWN0YWluOkBwUiptGFiczuWNTAh 6 Sec-Ch-Ua: "Not A Brand";v="8", "Chromium";v="108" 7 X-Srf-Token: eyJpdiI6IkROaGtQcUQ1bk5zY1AwSUdhN1RXaGc9PSIsInZhbHV 1ijoiZG9kWkVmZUhst2pWaaxyVDJJQk1UbzM4cTVhOXRhZEZcL0 1DbBVVFpSdUpVdm5YMTJTNGRoeXKbCdnYzfWvbUUxIiwiWFjI jojOWnhNzF1MjkzNmYzNm15Y2UxODFjM2E3YzU4NjMOOGUjMjg2 YjFjYTUCNWyYjBjY2NkYzcxYT12NhjNjcxZiJ9 8 X-Requested-With: XMLHttpRequest 9 Sec-Ch-Ua-Mobile: ?0 10 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/108.0.5359.95 Safari/537.36 11 Sec-Ch-Ua-Platform: "Windows" 12 Sec-Fetch-Site: same-origin 13 Sec-Fetch-Mode: cors 14 Sec-Fetch-Dest: empty </pre>	<pre> 1 HTTP/2 200 OK 2 Server: nginx/1.17.3 3 Content-Type: application/json 4 Vary: Accept-Encoding 5 Cache-Control: no-cache, private 6 Date: Tue, 13 Dec 2022 07:40:02 GMT 7 Set-Cookie: XSRF-TOKEN= eyJpdiI6ImFhY2g1YzdCCVhieF10aVEExUUtDT3c9PSIsInZhbHV 1ijoiVm92NWSSRlpvYjUrUDBoOHQWXhhYVpkck5WYWiMnpNNH crZ124Vu1iRWxYWm04SjAraU2UUEBwjBS0WpKR1IsImlhYyI6I jMxYDjNWN1MmEyOTFRHTk3MzQZGM2MWEWMTR1jVmNzQ2NjIz YmNiY2Vh2TUSMjZjNTR5MmQ2MDQzZTc2NTUifQ%3D%3D; expires=Tue, 13-Dec-2022 09:40:02 GMT; Max-Age=7200; path=/ 8 Set-Cookie: chaintrack_session= eyJpdiI6ijZnXC83U2dzRjEzNm43axXfaRFQOWFBPT0iLCj2YwX 1ZSi6Im5VQTZXK1BPeFZ1jZkdHY4ZEk2cUNeNlndzhhYj VKcEVObkp62FrYVF2bmJo=eWtnbnzrOXPrYTFKbz1MliwiWFjI jojZm1YTDhY2EylNzNnNjIwNjY1ZTg4OTdkNmI2Zjg3OTQxNWRi M2F1NWRiNzYwYzQwYjNmZDd1ODkyNWF1YTE2ZiJ9; expires=Tue, 13-Dec-2022 09:40:02 GMT; Max-Age=7200; path=/; httponly 9 X-Frame-Options: SAMEORIGIN 10 X-Xss-Protection: 1; mode=block 11 X-Content-Type-Options: nosniff 12 13 { "id": 40, "name": "Long Chen", "email": "lca160@sfu.ca", "country": "CA", "countryCode": "1", "phone": "+3369820191", "currentTeamId": 9, "settings": { "distanceUnit": "metric", "temperatureUnit": "celsius" }, "connectedToWrm": false, "connectedToMetric": false, "integrations": { "wrm": { "key": null } } } </pre>
<input type="button" value="?"/> <input type="button" value="⚙"/> <input type="button" value="↶"/> <input type="button" value="↷"/> <input type="text" value="Search..."/>	<input type="button" value="?"/> <input type="button" value="⚙"/> <input type="button" value="↶"/> <input type="button" value="↷"/> <input type="text" value="Search..."/>
0 matches	0 matches

401 https://app.chaintrack.ca GET /user/current
 402 https://app.chaintrack.ca GET /api/devices?page=1
 403 https://use.fontawesome.com GET /releases/v5.0.13/webfonts/fa-solid-900.woff2

Request		Response	
Pretty	Raw	Pretty	Raw
1 GET /api/devices?page=1 HTTP/2 2 Host: app.chaintrack.ca 3 Cookie: remember_web_59ba36addc2b2f9401580f014c7f58ea4e3098 9d= eyJpdii6Ik9BThVmDaSs3bjZ5ZYU3JuallDNc9PSIsInZhbbHV 1Ijo1Y9wZzEvySRpKzdpVER0WHLWGFwbGERazI30WtcLzYzZ Y1UjBcL0ozC3NQKtV6eDBFQ1RTUK5nVhpDRkNtDhPNXbfNue FZk5nZQWnUybWh3aWlZcTN2MmdkTzR1dEVQSiwwNDBoQ1lwvTVh NWk0Q3cmdWRtTm11UkxMcEY4Tzh2MitLcxPSPXbUem93WWNWeFY 1bz1ZQWzE5jMEJUZQvVVFZwOdh1U9i1wfj1joi1N2 Rh0Qz2MjkMDE5YWRlNTNkYmY5MjdMjU2NjIyMDM4Mzc0Mw1wZ WZhYTUSNGMyYz1NwQ4ZTJmMnjJz2Qw2S9; XSRF-TOKEN= eyJpdii6IkRoAgtQcU1bk5zY1AwSuDhN1RxGc9PSIsInZhbbHV 1Ijo1ZG9kWkVmZUhsT2pWa2xyVDJJQk1UbzM4cTVhOXRhZEzCl0 1DbVBVVFpSdUpVm5YMTJTNgrOeXBKcDNY2FwvbUUxIiivibWFj1 jo1OWNhNzFlMjkzMmYzNm15Y2UxDfFjM2E3YzU4NjMOOGjjMjg2 YjfjYTU2NWY2YjBjY2NkYzcxYTI2NjhjNjcxZij9; chaintrack_session= eyJpdii6IkQ1ZXLSW03WDjkSGxWjNrenf5NGc9PSIsInZhbbHV 1IjoiaFZGSU1TWJSVHFhWDkzTnI3a09uSExiG9IVDjuckx4Zn hFnYtrVupValhBrokeWngSHzNvWY5Qk12NvutbSIsImlhYy16I jZjY2F1TTkzNGN1YzdkZDQSNzcmzV12DxQ1ZDcwYj1jMzaZ NzczYWMOzWVxZDhmNdgON2EONmE4NTczNjkitQ43D43D 4 Authorization: Basic Qmx1ZWNoYwluOkBwaUlpTGFiczuwNTAh 5 Sec-Ch-Ua: "Not ?A_Brand";v="8", "Chromium";v="108" 6 Accept: application/json, text/plain, */* 7 X-Xsrf-Token: eyJpdii6IkRoAgtQcU1bk5zY1AwSuDhN1RxGc9PSIsInZhbbHV 1Ijo1ZG9kWkVmZUhsT2pWa2xyVDJJQk1UbzM4cTVhOXRhZEzCl0 1DbVBVVFpSdUpVm5YMTJTNgrOeXBKcDNY2FwvbUUxIiivibWFj1 jo1OWNhNzFlMjkzMmYzNm15Y2UxDfFjM2E3YzU4NjMOOGjjMjg2 YjfjYTU2NWY2YjBjY2NkYzcxYTI2NjhjNjcxZij9 8 X-Requested-With: XMLHttpRequest 9 Sec-Ch-Ua-Mobile: ?0 10 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/108.0.5359.95 Safari/537.36 11 Sec-Ch-Ua-Platform: "Windows" 12 Sec-Fetch-Site: same-origin 13 Sec-Fetch-Mode: cors 14 Sec-Fetch-Dest: empty	1 HTTP/2 200 OK 2 Server: nginx/1.17.3 3 Content-Type: application/json 4 Vary: Accept-Encoding 5 Cache-Control: no-cache, private 6 Date: Tue, 13 Dec 2022 07:40:02 GMT 7 Set-Cookie: XSRF-TOKEN= eyJpdii6Ikj6d6dzdTEMzdOkysmsbDzY31TM1E9PSIsInZhbbHV 1Ijo1NHjZy0ZwYytTTNmUkRq0WUsdElxejRwcFZQavVycHZGej YveG54YnZt3hSNzcrQWjN3ZDzWNOSnZVb1Ncl2c1LCjtYWMio i1ZBmMTIyZt3hSNzcrQWjN3ZDzWNOSnZVb1Ncl2c1LCjtYWMio NTcxZW10njY3M2ZmG14NzJmMme20dg2NjVjIn03D; expires=Tue, 13-Dec-2022 09:40:02 GMT; Max-Age=7200; path=/ 8 Set-Cookie: chaintrack_session= eyJpdii6IlFKckjksFlxOVVuYTVMkdDnhRDVhTFEP9PSIsInZhbbHV 1Ijo1Jv1uZwZwSeStOVEub3RTZGwzXc9t0VpscnQ3TFZobV Z3QkFwck0yQByvXBuWUluREpqSFVPbcswdDdeUUilCjtYWMio i1jOD12OGU3NDczNDgxM2Y0NzFhNjA3YVYyNTRjMTgwNzNhNDM2 MzNmWz1YTD1njQ10GMzMm11NTViNDjmZDjJIn03D; expires=Tue, 13-Dec-2022 09:40:02 GMT; Max-Age=7200; path=/; httponly 9 X-Frame-Options: SAMEORIGIN 10 X-Xss-Protection: 1; mode=block 11 X-Content-Type-Options: nosniff 12 13 { "data": [{ "id": 114, "name": "DL50114 - Dummy", "defaultName": "DL50114 - Dummy", "type": "logger", "vehicle": "Honda Civic 1998", "status": "active", "settings": { "gps": true }, "registeredAt": "2022-10-09T15:41:48-07:00" }], "links": { "first": } } 		
	Search...		Search...
0 matches		0 matches	



Result:

The browser sends every request with the remembered credential.

Recommendation: set the path of the remembered credential cookie in the response to the login page.

Insecure Direct Object References

Insecure Direct Object References occur when an application provides direct access to objects based on user-supplied input. As a result of this vulnerability attackers can bypass authorization and access resources in the system directly, for example database records or files.

Methodology

To test for this vulnerability we first need to map out all locations in the application where user input is used to reference objects directly. Sometimes the object reference may be split between more than one parameter, and testing should be adjusted accordingly.

Structure of APIs

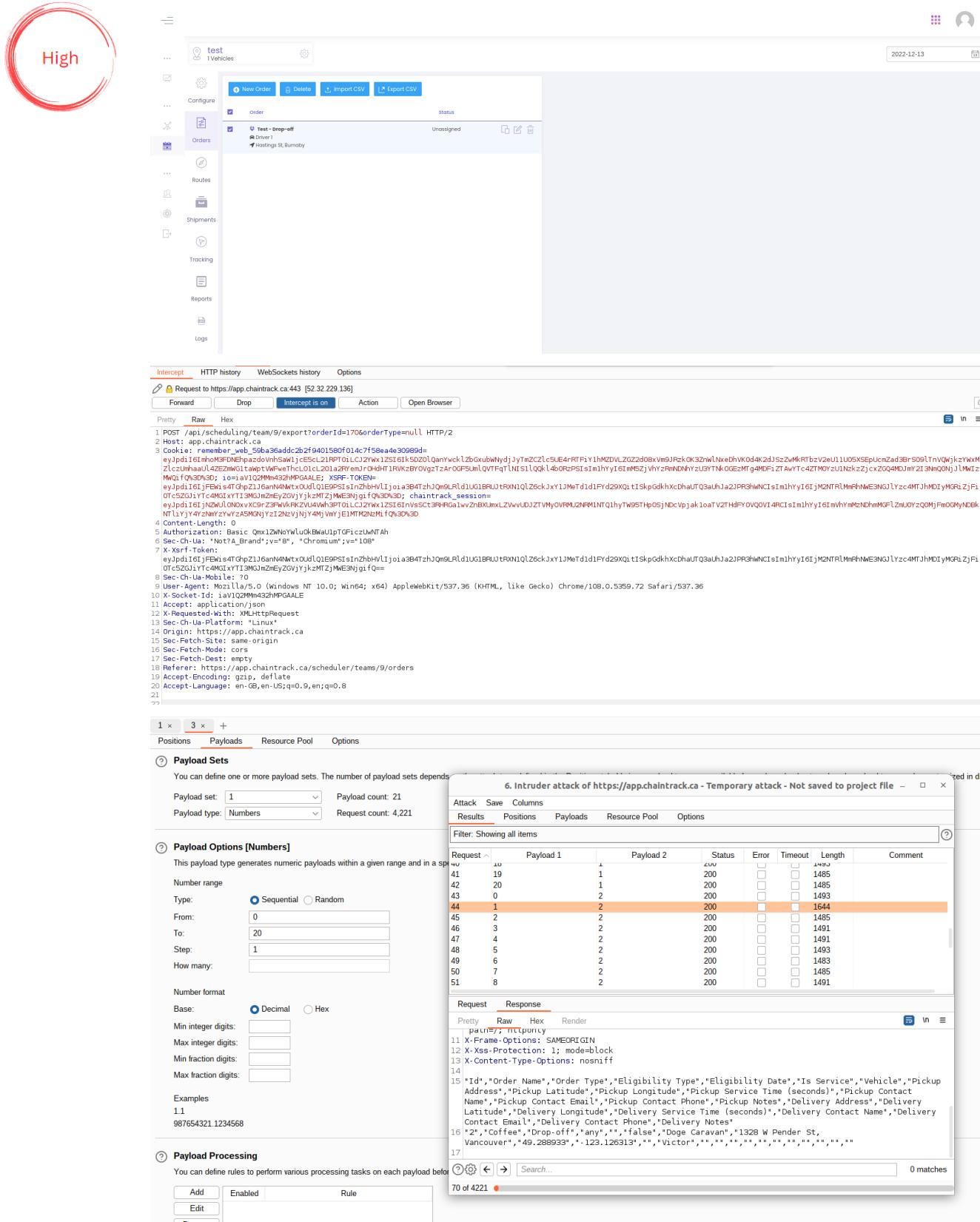
Request	Response
Pretty	Pretty
Raw	Raw
<pre> 1 GET /api/teams/9 HTTP/2 2 Host: app.chaintrack.ca 3 Cookie: io=lmnohx_b9ZQwOKORAAPm; remember_web_59ba36addc2b2f9401580f014cf7f58ea4e30989d= eyJpdiI6IkhHSkg45VNkUHF1MnRCQ01saVTU1E9PSIaInzhbHV1joiR1o3XC9wW8rVlh2elRTMLE3eUoxSTFoaEpr03jh FwtPaxNPVET1YldsdgpXkQ1vwTnZCLzBPYjk45Xl0R3ZQY09jexXBZDFpVktJUUh008rSlhnV0uanZhVmQStVd6Y2M0N0Hn ODV1vdTwbXppCt5M0jXWE5ahdvdTA3NG2xalVNOwUodvBjNG44Y1WnjZzVRQ21YMWxjMwZ02EN4SpbU50MklcL3NE629j Yz0i1CjTyWMDi015yjhNm2UzYzMyZjlyjlm4ZTc5YzA5Y2JhNwZlMTNxjUOuyjY2ZjzcN2VkmjAzTcyymjhjzASNDdjYz1z NDE2In0g3D; XSRF-TOKEN= eyJpdiI6IkhJ0aXPWMEZpcXVR0w9CQVNC1zJTTDjnPTo1LCJ2Ywx1ZSI6lFZNnpQDwcUpiQUE4Y0tiafdkZzZLdG1mXC9x WVVFUSHNOSU1DNjVoQxBOV1NTKodUYE1FbxaUnRFMGljTCtHi1wbfjijoizWY50GQ2N2Nlnz13ZTU3YzQY2JmJU0NDZm MmQyZwU30TM3ZTFmZmU1ZTMxNDgxODQ1Mz1MvkNgj1ZnQ40Cj9; chaintrack_session= eyJpdiI6Ikh9RxC90vUhQnGNTbjbjhMn12dKvrbjR3PT01LCJ2Ywx1ZSI6lFZNnpQDwcUpiQUE4Y0tiafdkZzZLdG1mXC9x bmRu1tuAlHFO5XpTRE0UKz1TvhOSExl3kZG1yj1j01NjBmYjk3DQxYTMONDdiNTM2ZTkwmGnM0dg5 OTRz0TQSNTM1YtCyZwQyj2FzmZhmCcZD1yZG1yj1j01G12MCj9 4 Authorization: Basic Qmx12wNoYlwlu0kBwUlptGFiczuWNTAh 5 Sec-Ch-Ua: "Chromium";v="103", ".Not/A/Brand";v="99" 6 Accept: application/json, text/plain, /* 7 X-Xsrftoken: eyJpdiI6IkhJ0aXPWMEZpcXVR0w9CQVNC1zJTTDjnPTo1LCJ2Ywx1ZSI6lFZNnpQDwcUpiQUE4Y0tiafdkZzZLdG1mXC9x WVVFUSHNOSU1DNjVoQxBOV1NTKodUYE1FbxaUnRFMGljTCtHi1wbfjijoizWY50GQ2N2Nlnz13ZTU3YzQY2JmJU0NDZm MmQyZwU30TM3ZTFmZmU1ZTMxNDgxODQ1Mz1MvkNgj1ZnQ40Cj9 8 X-Requested-With: XMLHttpRequest 9 Sec-Ch-Ua-Mobile: ? 10 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/103.0.5060.134 Safari/537.36 11 Sec-Ch-Ua-Platform: "Linux" 12 Sec-Fetch-Site: same-origin 13 Sec-Fetch-Mode: cors 14 Sec-Fetch-Dest: empty 15 Referer: https://app.chaintrack.ca/teams/9 16 Accept-Encoding: gzip, deflate 17 Accept-Language: en-US,en;q=0.9 18 19 </pre>	<pre> { "id": 9, "owner_id": 40, "name": "SFU", "slug": "sfu", "created_at": "2022-10-13 10:34:09", "owner": { "id": 40, "name": "Long Chen", "email": "lca160@sfu.ca", "country": "CA", "country_code": "1", "phone": "2369820191", "current_team_id": 9, "settings": { "distanceUnit": "metric", "temperatureUnit": "celsius" } }, "users": [{ "id": 40, "name": "Long Chen", "email": "lca160@sfu.ca", "country": "CA", "country_code": "1", "phone": "2369820191", "current_team_id": 9, "settings": { "distanceUnit": "metric", "temperatureUnit": "celsius" }, "pivot": { "team_id": 9, "user_id": 40, "role": "owner" } }] } </pre>

Testing some APIs by modifying parameters

Send	Cancel	< >	Target:
Request	Response		
Pretty	Pretty		
Raw	Raw		
<pre> 1 PUT /api/teams/1 HTTP/2 2 Host: app.chaintrack.ca 3 Cookie: remember_web_59ba36addc2b2f9401580f014cf7f58ea4e30989d= 9d=eyJpdiI6Ikh9RxC90vUhQnGNTbjbjhMn12dKvrbjR3PT01LCJ2Ywx1ZSI6lFZNnpQDwcUpiQUE4Y0tiafdkZzZLdG1mXC9x WVVFUSHNOSU1DNjVoQxBOV1NTKodUYE1FbxaUnRFMGljTCtHi1wbfjijoizWY50GQ2N2Nlnz13ZTU3YzQY2JmJU0NDZm MmQyZwU30TM3ZTFmZmU1ZTMxNDgxODQ1Mz1MvkNgj1ZnQ40Cj9; chaintrack_session= eyJpdiI6IkhJ0aXPWMEZpcXVR0w9CQVNC1zJTTDjnPTo1LCJ2Ywx1ZSI6lFZNnpQDwcUpiQUE4Y0tiafdkZzZLdG1mXC9x bmRu1tuAlHFO5XpTRE0UKz1TvhOSExl3kZG1yj1j01NjBmYjk3DQxYTMONDdiNTM2ZTkwmGnM0dg5 OTRz0TQSNTM1YtCyZwQyj2FzmZhmCcZD1yZG1yj1j01G12MCj9 4 Authorization: Basic Qmx12wNoYlwlu0kBwUlptGFiczuWNTAh 5 Sec-Ch-Ua: "Chromium";v="103", ".Not/A/Brand";v="99" 6 Accept: application/json, text/plain, /* 7 X-Xsrftoken: eyJpdiI6IkhJ0aXPWMEZpcXVR0w9CQVNC1zJTTDjnPTo1LCJ2Ywx1ZSI6lFZNnpQDwcUpiQUE4Y0tiafdkZzZLdG1mXC9x WVVFUSHNOSU1DNjVoQxBOV1NTKodUYE1FbxaUnRFMGljTCtHi1wbfjijoizWY50GQ2N2Nlnz13ZTU3YzQY2JmJU0NDZm MmQyZwU30TM3ZTFmZmU1ZTMxNDgxODQ1Mz1MvkNgj1ZnQ40Cj9 8 X-Requested-With: XMLHttpRequest 9 Sec-Ch-Ua-Mobile: ? 10 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/103.0.5060.134 Safari/537.36 11 Sec-Ch-Ua-Platform: "Linux" 12 Sec-Fetch-Site: same-origin 13 Sec-Fetch-Mode: cors 14 Sec-Fetch-Dest: empty 15 Referer: https://app.chaintrack.ca/teams/1 16 Accept-Encoding: gzip, deflate 17 Accept-Language: en-US,en;q=0.9 18 19 </pre>	<pre> 1 HTTP/2 403 Forbidden 2 Server: nginx/1.17.9 3 Content-Type: application/json 4 Vary: Accept-Encoding 5 Cache-Control: no-cache, private 6 Date: Wed, 14 Dec 2022 17:27:07 GMT 7 Strict-Transport-Security: max-age=31536000 8 eyJpdiI6Ikh9RxC90vUhQnGNTbjbjhMn12dKvrbjR3PT01LCJ2Ywx1ZSI6lFZNnpQDwcUpiQUE4Y0tiafdkZzZLdG1mXC9x WVVFUSHNOSU1DNjVoQxBOV1NTKodUYE1FbxaUnRFMGljTCtHi1wbfjijoizWY50GQ2N2Nlnz13ZTU3YzQY2JmJU0NDZm MmQyZwU30TM3ZTFmZmU1ZTMxNDgxODQ1Mz1MvkNgj1ZnQ40Cj9; chaintrack_session= eyJpdiI6IkhJ0aXPWMEZpcXVR0w9CQVNC1zJTTDjnPTo1LCJ2Ywx1ZSI6lFZNnpQDwcUpiQUE4Y0tiafdkZzZLdG1mXC9x bmRu1tuAlHFO5XpTRE0UKz1TvhOSExl3kZG1yj1j01NjBmYjk3DQxYTMONDdiNTM2ZTkwmGnM0dg5 OTRz0TQSNTM1YtCyZwQyj2FzmZhmCcZD1yZG1yj1j01G12MCj9 4 Authorization: Basic Qmx12wNoYlwlu0kBwUlptGFiczuWNTAh 5 Sec-Ch-Ua: "Chromium";v="103", ".Not/A/Brand";v="99" 6 Accept: application/json, text/plain, /* 7 X-Xsrftoken: eyJpdiI6IkhJ0aXPWMEZpcXVR0w9CQVNC1zJTTDjnPTo1LCJ2Ywx1ZSI6lFZNnpQDwcUpiQUE4Y0tiafdkZzZLdG1mXC9x WVVFUSHNOSU1DNjVoQxBOV1NTKodUYE1FbxaUnRFMGljTCtHi1wbfjijoizWY50GQ2N2Nlnz13ZTU3YzQY2JmJU0NDZm MmQyZwU30TM3ZTFmZmU1ZTMxNDgxODQ1Mz1MvkNgj1ZnQ40Cj9 8 X-Requested-With: XMLHttpRequest 9 Sec-Ch-Ua-Mobile: ? 10 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/103.0.5060.134 Safari/537.36 11 Sec-Ch-Ua-Platform: "Linux" 12 Sec-Fetch-Site: same-origin 13 Sec-Fetch-Mode: cors 14 Sec-Fetch-Dest: empty 15 Referer: https://app.chaintrack.ca/teams/1 16 Accept-Encoding: gzip, deflate 17 Accept-Language: en-US,en;q=0.9 18 19 </pre>		

Penetration Test Report - Chaintrack Company

Exporting orders API works for other teams



Tools

Burp

Result

We can export orders of other teams.

Test

<https://app.chaintrack.ca/scheduler/teams/9/orders>

Findings

Orders page is vulnerable to Insecure Direct Object References. By changing the export url, we were able to see other teams orders which include information about their cars, locations, depots, drivers, etc. This is a critical vulnerability.

Recommendations

The API should check if the user is authorized for exporting orders before returning the data.

5 Session Management

Logout Functionality

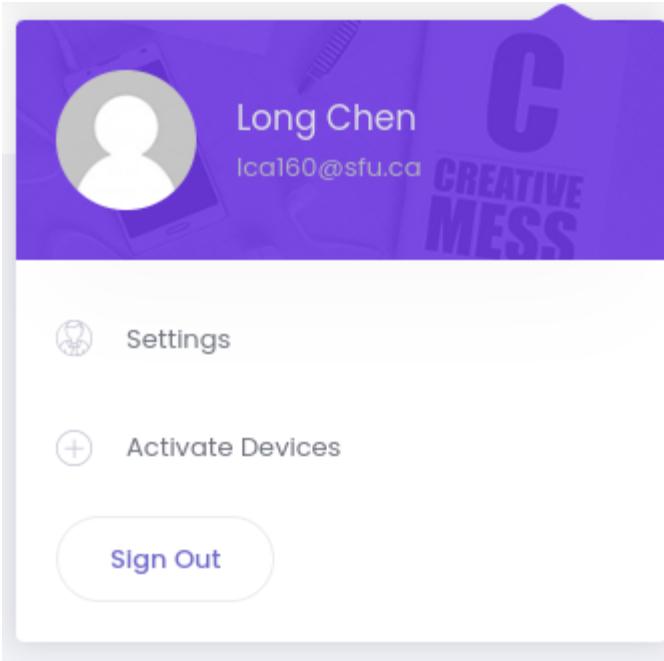
Methodology

There are some properties which indicate a good log out user interface:

1. A logout button is present on all pages of the web application.
2. The log out button should be identified quickly by a user who wants to log out from the web application.
3. After loading a page the logout button should be visible without scrolling.

Ideally the log out button is placed in an area of the page that is fixed in the viewport of the browser and not affected by scrolling of the content.

Chaintrack Logout



Tools

Browser

Test

All pages after login to the system.

Findings

Logout UI and functionality is good.

Recommendations

-

Cross Site Request Forgery (CSRF)

CSRF is an attack which forces an end user to execute unwanted actions on a web application in which he/she is currently authenticated.

Methodology

For a black box test we must know URLs in the restricted (authenticated) area. As we have valid credentials, we can find the URLs to be tested just by browsing around the application.

Tools

OWASP ZAP

Test

Scan the app using OWASP ZAP to find CSRF vulnerability.

Findings

The app uses CSRF token and is not vulnerable to CSRF attack.

Recommendations

-

Cookies Attributes



Low

Cookies are often a key attack vector for malicious users (typically targeting other users) and the application should always take due diligence to protect cookies.

Methodology

We should trap all responses where a cookie is set by the application (using the Set-cookie directive) and inspect the cookie for the following attributes:

1. Secure: This attribute tells the browser to only send the cookie if the request is being sent over a secure channel such as HTTPS. This will help protect the cookie from being passed over unencrypted requests. If the application can be accessed over both HTTP and HTTPS, then there is the potential that the cookie can be sent in clear text.
2. HttpOnly: This attribute is used to help prevent attacks such as cross-site scripting, since it does not allow the cookie to be accessed via a client side script such as JavaScript. Note that not all browsers support this functionality.
3. Domain Attribute - Verify that the domain has not been set too loosely. It should only be set for the server that needs to receive the cookie. For example if the application resides on server app.mysite.com, then it should be set to "; domain=app.mysite.com" and NOT "; domain=.mysite.com" as this would allow other potentially vulnerable servers to receive the cookie.

Tools

OWASP ZAP

Findings

- ›  Cookie No HttpOnly Flag (12)
- ›  Cookie Without Secure Flag (27)
- ›  Cookie without SameSite Attribute (27)

Cookie HttpOnly, Secure, SameSite flags are not set.

Recommendations

Mentioned attributes should be set.

6 Input Validation

Reflected Cross Site Scripting

Reflected Cross-site Scripting (XSS) occurs when an attacker injects browser executable code within a single HTTP response. The injected attack is not stored within the application itself; it is non-persistent and only impacts users.

The attack string is included as part of the crafted URI or HTTP parameters, processed by the application, and returned to the victim. Reflected XSS are the most frequent type of XSS attacks.

Methodology

For testing against Reflected XSS we are going to test pages since that requires some user input.

We are going to test in 2 ways a Black box, that consists of sending our own crafted URI or HTTP parameters and checking manually different attacks, and a grey box in which we will use the help of a tool.

Black Box:

Classic attack

```
<script>alert(Pentest)</ script>
“ onfocus=”alert(document.cookie)
```

Different syntax or encoding:

```
“ onfocus=”alert(document.cookie)
“><script>alert(document.cookie)</ script>
“><script>alert(Pentest)</ script>
“%3cscript%3ealert(document.cookie)%3c/script%3e
“%3cscript%3ealert(Pentest)%3c/script%3e
```

Bypassing non-recursive filtering

```
<src<script>ipt>alert(document.cookie)</ script>
```

Tools

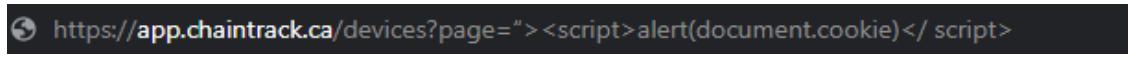
ZAP, Arachni

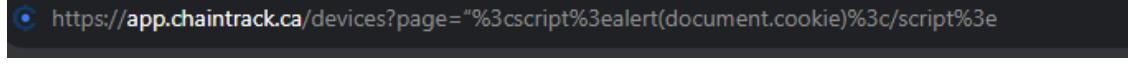
Test

For the first step we have identify the following URLs

<https://app.chaintrack.ca/devices?page=1>

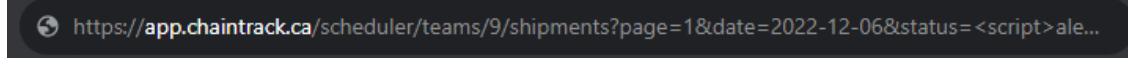
 https://app.chaintrack.ca/devices?page=<script>alert(Pentest)</ script>

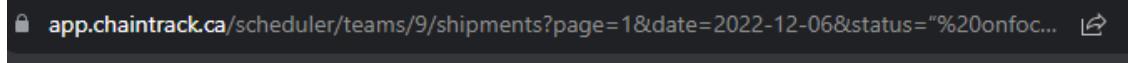
 https://app.chaintrack.ca/devices?page="><script>alert(document.cookie)</ script>

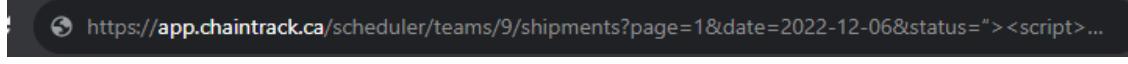
 https://app.chaintrack.ca/devices?page="%3cscript%3ealert(document.cookie)%3c/script%3e

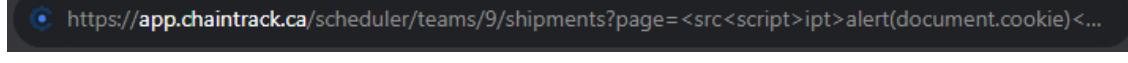
 https://app.chaintrack.ca/devices?page=<src<script>ipt>alert(document.cookie)</ script>

<https://app.chaintrack.ca/scheduler/teams/9/shipments?page=1&date=2022-12-06&status=all>

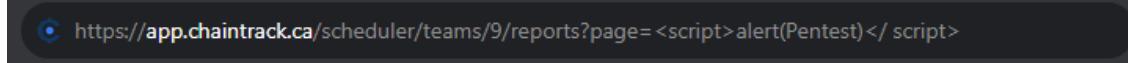
 https://app.chaintrack.ca/scheduler/teams/9/shipments?page=1&date=2022-12-06&status=<script>ale...

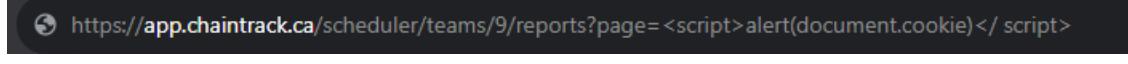
 https://app.chaintrack.ca/scheduler/teams/9/shipments?page=1&date=2022-12-06&status="%20onfoc... ↗

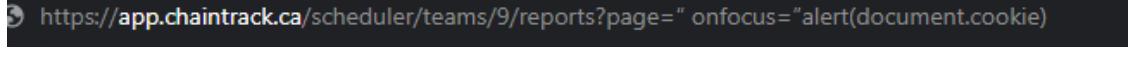
 https://app.chaintrack.ca/scheduler/teams/9/shipments?page=1&date=2022-12-06&status="><script>...

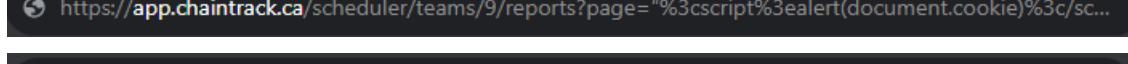
 https://app.chaintrack.ca/scheduler/teams/9/shipments?page=<src<script>ipt>alert(document.cookie)<...">

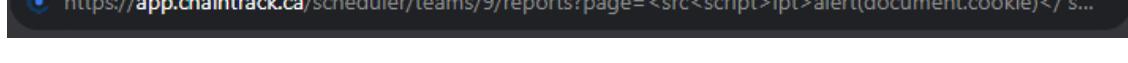
<https://app.chaintrack.ca/scheduler/teams/9/reports?page=1&date=2022-12-09>

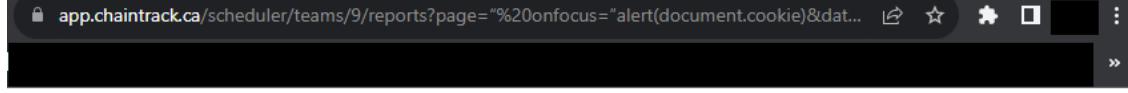
 https://app.chaintrack.ca/scheduler/teams/9/reports?page=<script>alert(Pentest)</ script>

 https://app.chaintrack.ca/scheduler/teams/9/reports?page=<script>alert(document.cookie)</ script>

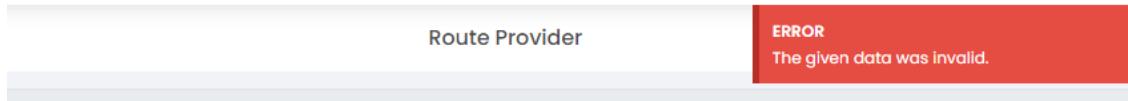
 https://app.chaintrack.ca/scheduler/teams/9/reports?page=" onfocus="alert(document.cookie)"

 https://app.chaintrack.ca/scheduler/teams/9/reports?page="%3cscript%3ealert(document.cookie)%3c/sc...

 https://app.chaintrack.ca/scheduler/teams/9/reports?page=<src<script>ipt>alert(document.cookie)< s...



RACK



Black Box testing:

Classic attack

Result: Not vulnerable

Different syntax or encoding:

Result: Not vulnerable

Bypassing non-recursive filtering

Result: Not vulnerable

Including external script

Result: Not vulnerable

Gray Box testing

ZAP

URL to attack:

Use traditional spider:

Use ajax spider: with

Progress: Failed to attack the URL: received a 401 response code, expected 2xx.

URL to attack:

Use traditional spider:

Use ajax spider: with

Progress: Failed to attack the URL: received a 401 response code, expected ...

URL to attack:

Use traditional spider:

Use ajax spider: with

Progress: Failed to attack the URL: received a 401 response code, expected 2...

TEST with Arachni

<https://app.chaintrack.ca/>

[Edit description](#)

✓ The scan completed in 00:00:06 .

Issues [0]

All [0] * Fixed [0] ✓ Verified [0] ⚡ Pending verification [0] ✗ False positives [0] ⚠ Awaiting review [0]

No issues discovered.

<https://app.chaintrack.ca/login>

<https://app.chaintrack.ca/devices?page=1>

Findings

No vulnerabilities found.

Recommendations: NA

Stored Cross Site Scripting

Stored XSS occurs when a web application gathers input from a user which might be malicious, and then stores that input in a data store for later use. A successful exploitation occurs when a user visits a page with a stored XSS.

Methodology

We need to identify all points where user input is stored into the back-end and then displayed by the application. Using the same methodology of a black box and grey box.

Black box:

Basic injection

```
aaa@aa.com"><script>alert(document.cookie)</script>
aaa@aa.com%22%3E%3Cscript%3Ealert(Pentest)%3C%2Fscript%3E
```

File Upload

If HTML or TXT files are allowed, XSS payload can be injected in the file uploaded. The pen-tester should also verify if the file upload allows setting arbitrary MIME types.

Tools

ZAP

Test

Black box:

Basic injection

URLS tested:

<https://app.chaintrack.ca/login>

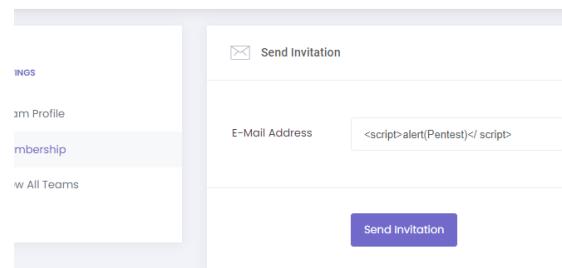


<https://app.chaintrack.ca/scheduler/teams/9/orders>

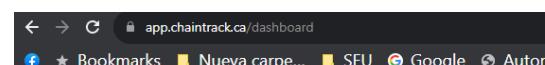
<https://app.chaintrack.ca/teams/9#/membership>



RACK



<https://app.chaintrack.ca/dashboard>



The search bar is greyed out so it is not possible to test.

Upload file.

The allowed files to be uploaded are .csv files, so we have tested with different files .txt .py and .js

.txt file:

The screenshot shows the Chaintrack application interface. At the top, there is a navigation bar with icons for location, vehicles, and settings. Below the navigation bar, there are buttons for 'New Order', 'Delete', 'Import CSV', and 'Export CSV'. A modal window titled 'New Order' is open, showing tabs for 'Pickup', 'Drop-off', 'Pickup & Drop-off', and 'Service'. The 'Import CSV' button is highlighted. On the right side of the screen, there are two red error boxes. The top one says 'ERROR: There was an error on row 1. Not enough rows!' and the bottom one says 'ERROR: The given data was invalid.'.

The screenshot shows the 'Route Provider' interface. It has a 'Status' section with a blue 'SUCCESS' banner that says 'Csv file is being imported'. Below this, there are 'Import CSV' and 'Export CSV' buttons.

This screenshot shows the Network tab of a browser's developer tools. A request to 'https://app.chaintrack.ca/api/scheduling/team/9/import' is listed. The request method is 'POST', status code is 200, remote address is '52.32.229.136:443', and referrer policy is 'strict-origin-when-cross-origin'. The payload section shows the raw CSV data being sent.

.js file:

This screenshot shows the Network tab of a browser's developer tools. A request to 'https://app.chaintrack.ca/api/scheduling/team/9/import' is listed. The request method is 'POST', status code is 200, remote address is '52.32.229.136:443', and referrer policy is 'strict-origin-when-cross-origin'. The payload section shows a JSON object with fields like 'orders?date=2022-12-14' and 'transport=polling&t...'. The response headers section includes 'cache-control: no-cache, private', 'content-encoding: gzip', 'content-type: application/json', and a date header.

.py file:

Name	Headers	Payload	Preview	Response	Initiator	Timing	Cookies
General							
?	Request URL: https://app.chaintrack.ca/api/scheduling/team/9/import						
EIO=3&transport=polling&t...	Request Method: POST						
?	Status Code: 200						
import	Remote Address: 52.32.229.136:443						
orders?date=2022-12-14	Referrer Policy: strict-origin-when-cross-origin						
Response Headers							
cache-control: no-cache, private							
content-encoding: gzip							
content-type: application/json							
date: Wed, 14 Dec 2022 08:23:48 GMT							
server: nginx/1.17.3							

Grey box:

For the grey box testing we will be using the ZAP ZAP

URL to attack:

Use traditional spider:

Use ajax spider: with

Progress: Failed to attack the URL: received a 401 response code, expected 2xx.

URL to attack:

Use traditional spider:

Use ajax spider: with

Progress: Failed to attack the URL: received a 401 response code, expected ...

URL to attack:

Use traditional spider:

Use ajax spider: with

Progress: Failed to attack the URL: received a 401 response code, expected 2...

Findings

He has found that for the file upload the URL being used is <https://app.chaintrack.ca/api/scheduling/team/9/import> and it accepts multiple file types as long as it has more than one row in it. We also can see that we got a code 200 successful for any type of file, which by standard means that everything worked as planned, the problem here is that the uploaded files do not look to be up loading in the corresponding page of orders, not even legit .csv files containing an order; What made us guess that there is a kind of bug in the import link and then we can not determinate if this can be used for future exploits.

Recommendations

Check on the <https://app.chaintrack.ca/api/scheduling/team/9/import> to validate that the behaviour found is a bug or an intentional feature. If it is a bug we would recommend checking on the parameters of the accepted file types.

SQL Injection

An SQL injection attack consists of insertion or “injection” of either a partial or complete SQL query via the data input or transmitted from the client (browser) to the web application.

A successful SQL injection attack can read sensitive data from the database, modify database data, execute administration operations on the db, recover content or write files into the file system, or even issue commands to the operating system.

Methodology

We need to identify when the application interacts with a DB Server in order to access some data. We also Would try to fingerprint the database. Using the same methodology of a black box and grey box.

Black box:

Standard SQL Injection

```
$username = '1' or '1' = '1
$password = '1' or '1' = '1
' or 1=1;--
$username = '1' or '1' = '1')/*
?username=1'%20or%20 '1'%20=%20'1&password=1'%20or%20'1'%20=%20'1
?username=1'%20or%20 '1'%20=%20'1'))%20LIMIT%201/*&password=foo
```

SQL command that imposes a condition that the number of the returned results must be one.

```
$username = '1' or '1' = '1') LIMIT 1/*
```

Simple SELECT statement

in URL:

```
=10 AND 1=2
```

Fingerprinting the Database

- 1) The first way to find out what back end database is used is by observing the error returned by the application.
- 2) If there is no error message or a custom error message, try to inject it into the string field using a concatenation technique.

Tools

Sqlmap, burp

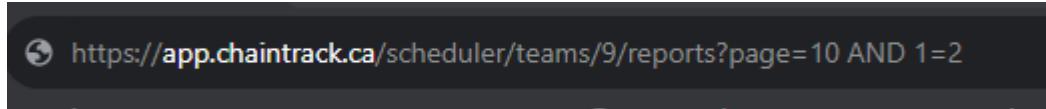
Test

<https://app.chaintrack.ca/login>

The screenshot shows a login form for 'ADMINISTRATOR OR DISPATCHER'. It features a large blue circular logo with a white 'C' in the center. The form has fields for 'EMAIL ADDRESS' containing 'T or T = 1', 'PASSWORD' containing '*****', and a checked 'Remember Me' checkbox. A blue 'Login' button is at the bottom. The background is white with light gray horizontal lines separating the sections.

The screenshot shows a login form for 'ADMINISTRATOR OR DISPATCHER'. It features a large blue circular logo with a white 'C' in the center. The form has fields for 'EMAIL ADDRESS' containing 'T or T = ?)) LIMIT 1/*', 'PASSWORD' containing '*****', and an unchecked 'Remember Me' checkbox. A blue 'Login' button is at the bottom. The background is white with light gray horizontal lines separating the sections.

<https://app.chaintrack.ca/scheduler/teams/9/reports?page=1>



Fingerprinting the Database

For this task we used burp but we were not able to generate any DB error that gave us a hint on the DB used..

So we did a guess and moved to sqlmap but there is no sitemap.



```
sqlmap -v --method=GET --url=https://sqlmap.org --threads=1 --random-agent  
[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the responsibility of the user to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for any damages caused by sqlmap in any way.  
[*] starting @ 23:07:35 /2022-12-13/-TOKEN=  
do you want to check for the existence of site's sitemap(.xml) [y/N] y  
[23:07:44] [WARNING] 'sitemap.xml' not found  
[23:07:44] [INFO] chaintrack_session  
[23:07:44] [INFO] starting crawler for target URL 'https://app.chaintrack.ca/login'  
[23:07:44] [INFO] searching for links with depth 1  
[23:07:44] [INFO] found 1 link(s) in total  
[23:07:44] [INFO] crawled 1 link(s) in total  
[23:07:44] [INFO] finished crawling
```

Findings

After testing the page against SQL injection we can say that this app is not vulnerable to it

Recommendations

NA

7 Error Handling

Nessus is a widely used tool that helps identify network and web application vulnerabilities and security issues. It uses various techniques, such as port scans, and checks for missing patches and misconfigurations. We use Nessus to scan for any vulnerabilities and security weaknesses in the web application.

webapp / Plugin #10386

[◀ Back to Vulnerabilities](#)

Vulnerabilities 5

INFO Web Server No 404 Error Code Check

Description
The remote web server is configured such that it does not return '404 Not Found' error code returning instead a site map, search page or authentication page.

Nessus has enabled some counter measures for this. However, they might be insufficient. If this port, they might not all be accurate.

Output

```
CGI scanning will be disabled for this host because the host responds to requests for non-existent URLs with HTTP code 302 rather than 404. The requested URL was :

http://ns35.domaincontrol.com:8008/xyir2spZizpZ.html

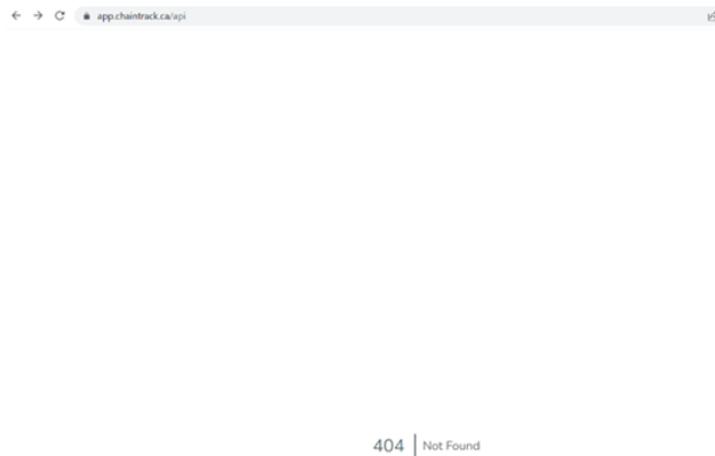
To see debug logs, please visit individual host
```

Port ▲	Hosts
8008 / tcp / www	97.74.107.18

One is the 404 error code check, part of the error handling test. The result shows that the application handles the error code appropriately. And we can also use tools such as Telnet and Burp Suite to scan for any URLs that may have potential error-handling issues. In this case, we can also visit from any browser for the URLs we found in information gathering and looking for files or web pages that do not exist.

The screenshot shows a web browser window with the following details:

- Address bar: app.chaintrack.ca/images/
- Page title: 403 Forbidden
- Page content: nginx/1.17.3



The images above show the example of our exploration, that code 403 means users do not have permission to visit and code 404 means the page does not exist. Improper error handling can allow attackers to understand the internal use of APIs and gather versions and types of applications.

8 Weak Cryptography

Testing for Weak Transport Layer Security

Test Objectives

- Validate the service configuration.
- Review the digital certificate's cryptographic strength and validity.
- Ensure that the TLS security is not bypassable and is properly implemented across the application.

Steps and Results

Identifying SSL services with Nessus:

Output

TLSv1.2 is enabled and the server supports at least one cipher.

Port	Hosts
443 / tcp / www	app.chaintrack.ca

Identifying weak cipher with <https://www.ssllabs.com/projects/index.html>

Low

Cipher Suites	
# TLS 1.2 (suites in server-preferred order)	[x]
TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 (0xc02c) ECDH x25519 (eq. 3072 bits RSA) FS	256
TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256 (0xccaa) ECDH x25519 (eq. 3072 bits RSA) FS	256
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 (0xc024) ECDH x25519 (eq. 3072 bits RSA) FS WEAK	256
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA (0xc00a) ECDH x25519 (eq. 3072 bits RSA) FS WEAK	256

Identifying SSL Certificate validation with sslscan:

```

SSL Certificate:
Signature Algorithm: sha256WithRSAEncryption
ECC Curve Name:      secp384r1
ECC Key Strength:    192
Subject:   app.chaintrack.ca
AltNames:  DNS:app.chaintrack.ca
Issuer:    R3

Not valid before: Dec 11 14:42:13 2022 GMT
Not valid after:  Mar 11 14:42:12 2023 GMT

```

Recommendations

The server should disable the TLS 1.2 cipher suites
 TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 and
 TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA, which are considered weak.

9 Business Logic

Testing business logic includes thinking in unconventional methods and checking what happens if the user does not act in the normal, intended process. The hard part of this section is that tools are incapable of detecting logical vulnerabilities. As a result, for this part, we didn't use any tool other than Burp, and it mostly included intercepting requests and changing the destination and body of the request. As a result, the structure of this part is a little bit different, and the most detrimental vulnerability is also found in this part.

This testing section includes tests for Data validation, Unexpected file type upload, Defense against application misuse, and Ability to forge requests. Some of the tests in this section overlap with the previous sections. For example, uploading files is tested in the previous section and the results have been shared there.

Teams APIs

While testing APIs related to teams, it could be seen that deleting or editing teams is returning a 403 Forbidden status. By forging the requests, we tried different bodies and team numbers but our requests were successfully blocked, which shows correct behavior for these APIs.

Vehicles APIs

By using get API for vehicles, the list of the other teams' vehicles can be retrieved.



```

Send | Cancel | < | > | □ |
Request
Pretty Raw Hex
1 GET /api/scheduling/team/1/vehicles HTTP/2
2 Host: app.chaintrack.ca
3 Cookie: remember_web=59ba36addc2b2f9401580f0147f5eae4e30989d...
eyJpdIiE1KhhsKg4SVNkUHETMnRQ01s1vVU1lEP9S1s1nZbHh1IjoiRl03XG9wBrV1h2elRTMUE3eUoxSTPoaEprQ3jh
RwtPabzqEV1LmWzqCz1yvTzClzBPyj45xh0R3Z02099jx8B2zDFpVkt1Uuhw0GgrSihrv0unZhVmQ5tVd6y2M0n0h
0V01wh-vb-0C5M01AxE5o-aHvdtA3NG2xavnW0uJodvB)NG44yWn)ZzVRQ21YMwrxMw202EN45Opribu50MkLc3Ned29j
YzctC1tyah015y1jMhmcu2r2Mrj1jYm42TC5Y2AS12JhWzLHTMXY20UjY2Zj)2h2V4M)AzTycymhMjZ45h0d)yz1z
M0E21h0h3D; XSRF-TOKEN=...
eyJpdIiE1rhysUljUj1UxLPwHcLR2aCtFQk1JTpBPT01LCJ2Yw12SI61ndyRFBncnltekrnhUckYxNjhj3k0VmdQTHzm
an1xM2wUjwvSD0Rzhe6XpVzD0vVdaShc3YmzIUEVfUj1iwiwfj1jo00QzYz3Wfph0WyX2M2M1NDRl2Mv1NgIwYTg2
NwM01TLM2M42N2FkZm15YwM0N2e2MdU4YThmQ1kMjA52wUyY19; chaintrack_session=...
eyJpdIiE1rhhsKg4SVNkUHETMnRQ01s1vVU1lEP9S1s1nZbHh1IjoiRl03XG9wBrV1h2elRTMUE3eUoxSTPoaEprQ3jh
RwtPabzqEV1LmWzqCz1yvTzClzBPyj45xh0R3Z02099jx8B2zDFpVkt1Uuhw0GgrSihrv0unZhVmQ5tVd6y2M0n0h
0V01wh-vb-0C5M01AxE5o-aHvdtA3NG2xavnW0uJodvB)NG44yWn)ZzVRQ21YMwrxMw202EN45Opribu50MkLc3Ned29j
YzctC1tyah015y1jMhmcu2r2Mrj1jYm42TC5Y2AS12JhWzLHTMXY20UjY2Zj)2h2V4M)AzTycymhMjZ45h0d)yz1z
4 Authorization: Basic Qmx1ZwNoYwLuKkBwauUpTGFiczuNNTAh
5 Sec-Ch-Ua: "Chromium";v="103", ".Not/A/Brand";v="99"
6 Accept: application/json
7 X-XsrF-Token: ...
eyJpdIiE1rhysUljUj1UxLPwHcLR2aCtFQk1JTpBPT01LCJ2Yw12SI61ndyRFBncnltekrnhUckYxNjhj3k0VmdQTHzm
an1xM2wUjwvSD0Rzhe6XpVzD0vVdaShc3YmzIUEVfUj1iwiwfj1jo00QzYz3Wfph0WyX2M2M1NDRl2Mv1NgIwYTg2
NwM01TLM2M42N2FkZm15YwM0N2e2MdU4YThmQ1kMjA52wUyY19
8 X-Powered-By: PHP/8.1.12
9 Sec-Ch-Ua-Mobile: 10
10 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
    Chrome/103.0.5060.134 Safari/537.36
11 Sec-Ch-Ua-Platform: "Linux"
12 Sec-Fetch-Site: same-origin
13 Sec-Fetch-Mode: cors
14 Sec-Fetch-Dest: empty
15 Referer: https://app.chaintrack.ca/scheduler/teams/9/routes
16 Accept-Encoding: gzip, deflate
17 Accept-Language: en-US;q=0.9
18
19

```

Response			
Pretty	Raw	Hex	Render
11 X-Content-Type-Options: nosniff			
12			
13 {			
"code":0,			
"data":{			
"vehicles":{			
{ <td></td> <td></td> <td></td>			
"id":14,			
"team_id":1,			
"iot_device_id":null,			
"gps_device_id":null,			
"external_id":Qondo-1",			
"color":#339900,			
"make":"Nissan",			
"model":"Mde1",			
"year":2010,			
"plate_number":"354",			
"vin_number":"VIN",			
"full_name":"Nissan Mde 2010",			
"gps_device":null,			
"device":null			
},			
{ <td></td> <td></td> <td></td>			
"id":8,			
"team_id":1,			
"iot_device_id":null,			
"gps_device_id":null,			
"external_id":Toyota Hiace",			
"color":#009900,			
"make":"Toyota",			
"model":"Hiace",			
"year":2015,			
"plate_number":"PLATE",			
"vin_number":"VIN",			
"full_name":"Toyota Hiace 2015",			
"gps_device":null,			
"device":null			
},			
},			

Orders APIs

Critical

The bigger problem happens in this part. Trying to get the orders for another team, results in a 404 Not Found response. It shows that users' of the teams cannot reach the orders of other teams. But by trying to create an order for another team, you will get a 201 Created response:

Request	Response
Pretty	Pretty
Raw	Raw
<pre> 1 POST /api/scheduling/team/1/orders HTTP/2 2 Host: app.chaintrack.ca 3 Cookie: remember_web_59ba3addc2b2f9401580f014c7f58ea4e309898d= eyJpdiI6InQ2MWF3Y18jUjZEQU9xR0hKUjV1NkE9PSIsInZhbhVlIjoi UHbhZLx1VUjzSHhRNfp60EJHS0ZLQwglrVNIdw10mjvJfLZDvZFKQwLI WmlzbEzyWwE3dwLBODErSmdXZXsVOFTMnArbHNyOEk3029rWrRnSTzs R0jXUjNmBvd0VgYyXBUU1RaVEra1VFMnlCtwLoQxtc21Kz1cRDvo FvZvMuJoxZE1ZS2tEcmPsmnRuhCp_tmOU15dnFDUDMock4yS2fQTGVt RxhVPSIsIm1hyi16mjzWE2YTjRjNj_dzD1LnFkMjEwODuxYQyTfL MjkzNzlkyThLzmFnjJkOEON2NjNjNKNMMy2M2N2e4ZQjifq%3Dn3D ;io=KGDnSeAKDHwLnt2AP; XSRF-TOKEN= eyJpdiI6ljVXZGdTeHjFaGowXGe9eduhCS3F2V1FrPToIcLj2Ywx1ZS16 ImhBa9CU96NyrenREY3M4eHk1M3RuazBZMDzoNFwvZzlBQuTTUV2 eFhnd1NU0GV5ZvpWmd1dk50Ww5CQkZHiiwibWFj1joiZT1ZDYY3Ytkw YwExYdhnNt1yNjUwz1zYmNmNnZTNkNGQ0MTj1ZTQxMfrkZWULYTk2 ZjZkZDFHNT1OYzkz1J9 4 Content-Length: 362 5 Authorization: Basic Qmx1ZwNoYwluokBwau1pTGF1czUwNtAh 6 Sec-Ch-Ua: "Chromium";v="103", ".Not/A BRAND";v="99" 7 X-Xsrf-Token: eyJpdiI6IkVHV224UJxdwMmZxTURydnMw0GQ2K3c9PSIsInZhbhVlIjoi YlpBKLYUmFvT0nDeko4QlVgtk5SiNLwvcWc5N01NV2cyYXpQMwJrQ3d1 blZQMLpzaFwvdFV1bE1HOTM4ZUvhVnoSi1wibWFj1joiZTmxODEzNjI4 YzMxY2Yx0tQw0tC4ZD12YzFhMymzZU2MmRhZDphzlh0HdmDg2M2Rl ZjZkZDFHNT1OYzkz1J9 11 12 { "code":0, "data":null </pre>	<pre> 1 HTTP/2 201 Created 2 Server: nginx/1.17.3 3 Content-Type: application/json 4 Cache-Control: no-cache, private 5 Date: Wed, 14 Dec 2022 05:19:04 GMT 6 Set-Cookie: XSRF-TOKEN= eyJpdiI6IkxkYzLjMw1hekdiVss4awN0wWpvMnc9PSIsInZhbhVlIjoi V09GbstPMmhDewtLVTjYQwvVmJ1c1E5bTYydlhkOHJx0cyeW4ySORM UmLME1mZVjDTGjyxK94wmhMnJ2dzJM1iwbwfj1joiNjdhYTQ1NTYO MzK0ZTAZTikyMDA1ZGFLnmNLYTUYYT1wZdkyyWwODYSMTLjZQjxNzMo YWU4MDAwNzQ5jz1ZCj9; expires=Wed, 14-Dec-2022 07:19:04 GMT; Max-Age=7200; path=/ 7 Set-Cookie: chaintrack_session= eyJpdiI6Ik1NdXhlt0102vFyt0llTkvxUm1xFRe9PSIsInZhbhVlIjoi VUFydwZ1avQSPiHRL3JUwEVPRQxpUSFArNONK29mcEtMVR6YkZZR2xa b2oxRddt0w6yUs19udSUxL0ODJkemciLCjtyWm1.0.1.2NtlhNjBmZnhV Yz14Mj1kMTAAy1vhWrfjYTk22WIwMzc0ZG15ZjBmZwYx0TcwZmY1YmFh ZmES2I0MjUwNjJlin%3D; expires=Wed, 14-Dec-2022 07:19:04 GMT; Max-Age=7200; path=/; httponly 8 X-Frame-Options: SAMEORIGIN 9 X-Xss-Protection: 1; mode=block 10 X-Content-Type-Options: nosniff 11 12 { "code":0, "data":null </pre>

As we couldn't see the orders of other teams at this step, for checking that the order is really created we used 2 approaches: the first one was using the export API vulnerability explained earlier. By using that, it can be seen that the order is successfully created. The other problem is that the order id is incremental among different teams. For example, if the latest order id is 183, by creating an order for another team and then creating an order for our team, the new order id was 185, showing that order creation for the other team was successful. It even proved to us more concrete after finding the vulnerability of delete order API.

The first problem with delete order API is that by calling it even with a non-existing order id, it will return a 200 OK status with a complete list of existing orders. It is a wrong status code as the act of deleting didn't happen here. But the biggest problem is that by calling this API for another team, it returns all the orders of that team. Using this problem, it could be seen clearly that we were able to create an order for another team. The vulnerability doesn't end here, and as it could be expected, deleting orders of other teams is also possible.

By using the mentioned vulnerabilities, a user in team x, can get a complete list of orders of team y, create fake orders for them, and even delete whatever order for team y. This counts as a critical problem for this service.

Penetration Test Report - Chaintrack Company

Recommendations

This problem can be easily fixed by using proper access control for users. It seems that for getting orders everything works fine, and by applying the same policy on creating and deleting orders the problems can be fixed.

10 Client Side

Client Side URL Redirect

Also known as Open Redirection. It is an input validation flaw that exists when an application accepts an user controlled input which specifies a link that leads to an external URL that could be malicious.

A phishing attack example could be the following:

<http://www.target.site?#redirect=www.fake-target.site>

The victim that visits target.site will be automatically redirected to fake-target.site where an attacker could place a fake page to steal the victim's credentials.

Methodology

Following with previous methodologies we found that Black box testing for Client Side URL Redirect is not usually performed since access to the source code is always available as it needs to be sent to the client to be executed.

We will be using Burp for the grey box testing:

1. Spider target site
2. Filter sitemap by status code such as 3xx [Redirection]
3. Analysis results , modify and scan

Tools

Burp

Test

1. Spider target site
2. Filter sitemap by status code such as 3xx [Redirection]

#	Request	Response	Count	Time	Size	Format
222	https://app.chaintrack.ca	GET /api/teams/9/invitations	200	1409	JSON	
223	https://app.chaintrack.ca	GET /api/teams/roles	200	942	JSON	
1	https://app.chaintrack.ca	GET /	302	1343	HTML	
14	https://app.chaintrack.ca	POST /login	302	1365	HTML	

3. Analysis results , modify and scan

Findings

We have tried to modify the redirects without a successful result so we can determine that it is not vulnerable to Client Side URL Redirect/Open Redirection.

Recommendations

NA

Clickjacking

Clickjacking is a malicious technique that consists of deceiving a web user into interacting (in most cases by clicking) with something different to what the user believes they are interacting with. The term “Clickjacking” was coined by Jeremiah Grossman and Robert Hansen in 2008.

Methodology

We will use the help of burp and The Burp Clickbandit to test.

From burp we go to the Burp menu and select Burp Clickbandit. Then use the following steps:

1. Click the Copy Clickbandit to clipboard button. This will copy the Clickbandit script to your clipboard.
 2. In your browser, visit the web page to test.
 3. Open the web developer console.
 4. Paste the Clickbandit script into the web developer console, and run it.

Tools

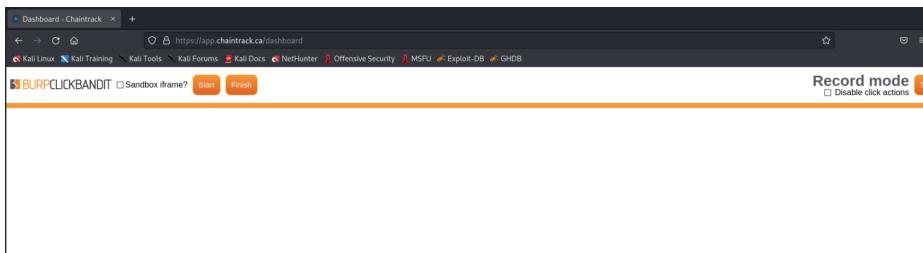
BurpSuite

Test

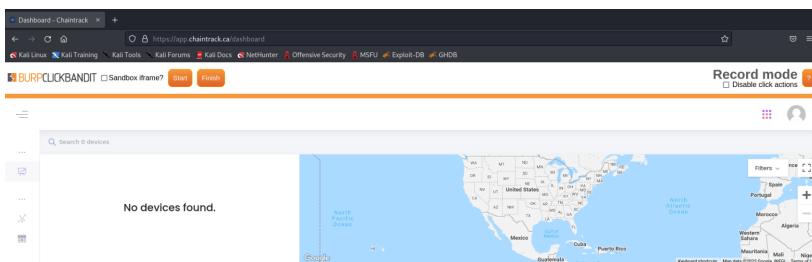
From burp we go to the Burp menu and select Burp Clickbandit. Then use the following steps:

1. Click the Copy Clickbandit to clipboard button. This will copy the Clickbandit script to your clipboard.
2. In your browser, visit the web page to test.
3. Open the web developer console.
4. Paste the Clickbandit script into the web developer console, and run it.

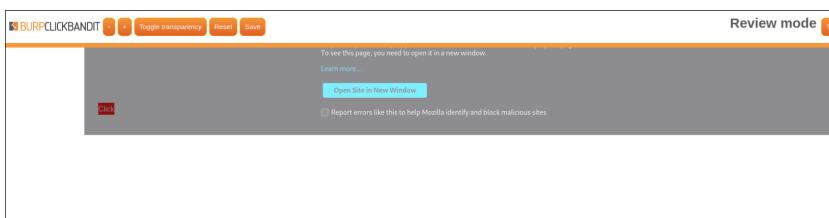
After this The Burp Clickbandit banner will appear at the top of the browser window and the original page will be reloaded within a frame, ready for the attack to be performed.



While recording we need to navigate and perform the steps that we want the user to do. Once it is over we click on finish.



Then we should be able to see the page in order to configure where we want to locate our button.



In this case the real page does not load and when we check on the error we got the following message.



Website will not allow Firefox to display the page if another site has embedded it

If you see this error, it is probably because a website is trying to display another website without the consent of its owner. This is usually the result of a security misconfiguration.

Websites can use [x-frame-options](#) or a [content security policy](#) to control whether other websites may embed them in their own pages. They are important security tools designed to prevent [clickjacking](#), which is an attack that allows malicious sites to trick users into clicking their links.

To visit a site that has shown this message, you can open the link in a New Tab or New Window in Firefox. Note that in some cases, the embedding page will not work correctly without access to the blocked page. In this case, you will need to contact the owner of the broken site for troubleshooting.

Share this article: <https://mzl.la/3biUfW>

Findings

We can see that this page is not vulnerable to Clickjacking.

Recommendations

NA.