

COMPLEX APPLICATIONS SECURITY ASSESSMENT

For: Cede Labs

By: Hacken

Date: October 22nd, 2023

Table of Contents

Introduction	1
Executive Summary	2
Security Assessment Overview	3
Scope	4
Team Composition	5
Methodology	6
Objectives	7
Limitations and Assumptions	8
Disclaimer	9
Definitions & Abbreviations	10
Summary of Findings	11
Remediation Check Protocol	12
Key Findings	13
Web Applications Specific Vulnerabilities	14
Private Keys For Exchanges Could Be Restored From Locked Application	15
Using Components With Known Vulnerabilities	16
Arbitrary Origin Trusted	17
Appendix A. OWASP Testing Checklist	18

This document is proprietary and confidential. No part of this document may be disclosed in any manner to a third party without the prior written consent of Hacken.

This document contains confidential information about IT systems and the network infrastructure of the customer, as well as information about potential vulnerabilities and methods of their exploitation.

This confidential information is for internal use by the customer only and shall not be disclosed to third parties.

Document

Name:	COMPLEX SECURITY ASSESSMENT FOR Cede Labs
Type:	Detailed Penetration Test Report
Revision:	Version 3
Date:	October 22 nd , 2023

Contractor Contacts

Role	Name	Email
Project Lead	Andrew Matiukhin	a.matiukhin@hacken.io

This document is proprietary and confidential. No part of this document may be disclosed in any manner to a third party without the prior written consent of Hacken.

Introduction

We thank Cede Labs for allowing us to conduct a Web Application Security Assessment. This document outlines our methodology, limitations, and results of the security assessment.

Executive Summary

Hacken OÜ (Consultant) was contracted by Cede Labs (Customer) to perform a third-party, independent security assessment of their web/mobile applications.

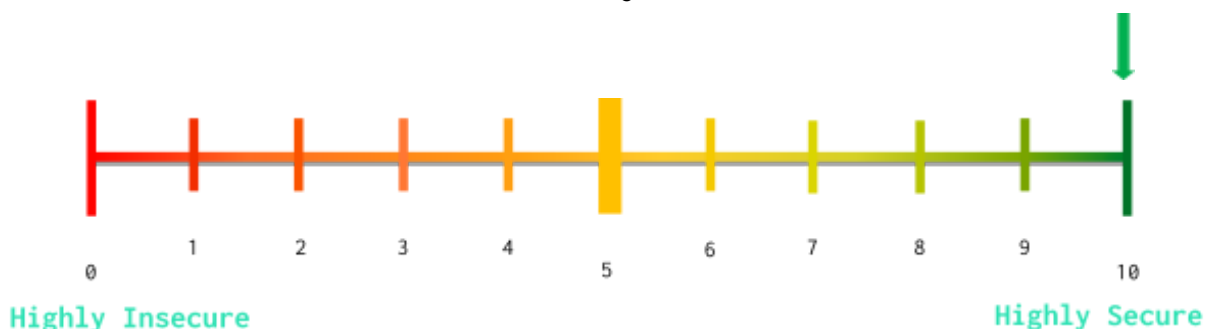
This report presents the findings of the security assessment of Web application & API security testing that was conducted between September 19nd, 2023 - October 22nd, 2023.

The purpose of the engagement was to utilize active exploitation techniques to evaluate the security of the web application against best practices and to validate its security mechanisms.

Next vulnerabilities and mistakes were identified during the assessment:

	Web	Overall (after remediation check)	Unable to check
Critical	0	0	-
High	1	0	-
Medium	1	0	-
Low	0	0	-
Informational	1	0	-

Overall Security Benchmark



Based on our understanding of the environment, as well as the nature of the vulnerabilities discovered, their exploitability, and the potential impact we have

This document is proprietary and confidential. No part of this document may be disclosed in any manner to a third party without the prior written consent of Hacken.

assessed the level of risk for your organization to be **LOW**. No direct path of external attacker to full system compromise was discovered.

The overall rating of Customer Applications, after the security assessment by the Consultant's Security Team, stands out to be **10 out of 10**. The security assessment was carried out following the in-house test cases, manual methods, exploitation, and automated tools.

Security Assessment Overview

Scope

The following table provides a synopsis of target systems that were within the scope of this Security Assessment.

#	Name	Type
1	https://cede.store/	WEB
2	https://api.cedelabs.io/	API

Security Assessment start and end dates were coordinated by email according to the following table:

Testing start date:	September 19 th , 2023
Testing end date:	September 29 th , 2023
Reporting:	October 22 nd , 2023

Team Composition

The project team consisted of 3 security experts with the following roles, certifications, and responsibilities:

Role	Responsibility
Project Manager	Customer communication Project delivery and quality control
Penetration Tester #1 (Lead Penetration tester, OSCP, Node.js, React, PHP, Websockets)	Project planning and executing Penetration Testing Identify security and business risks for the application Preparing artefacts and deliverables Results Presentation
Penetration Tester #2 (Penetration tester, Java, PHP, Node.js, Databases)	Penetration Testing Identify security and business risks for infrastructure

This document is proprietary and confidential. No part of this document may be disclosed in any manner to a third party without the prior written consent of Hacken.

Methodology

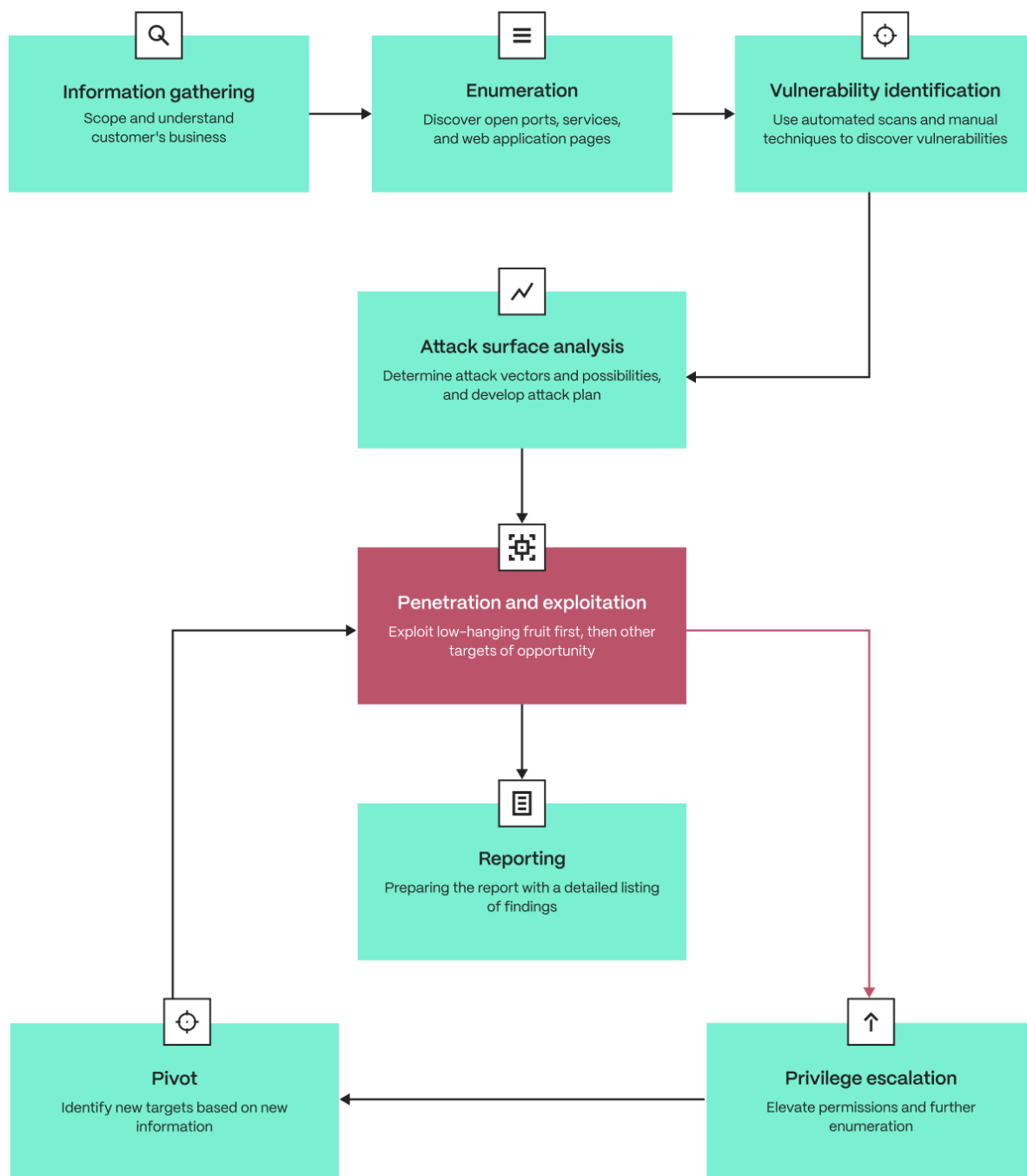
Our methodology for Security Assessment is based on our own experience, best practices in the area of information security, international methodologies, and guides such as PTES and OWASP.

Security Assessment has been conducted following workflow:

- Pre-engagement Interactions
- White box security assessment
 - Intelligence gathering activities against a target
 - Service detection and identification
 - Vulnerabilities detection, verification, and analysis
 - Business logic flows
 - The exploitation of vulnerabilities
 - Lateral movement and privilege escalation
- Mapping application code against industry best practices OWASP ASVS
- Preparing the final report with a detailed listing of findings, along with the related risks and recommendations.

The diagram below illustrates the standard security assessment methodology followed by Hacken Team. A cyclical approach to security assessment is leveraged so new information is incorporated into subsequent on the environment.

This document is proprietary and confidential. No part of this document may be disclosed in any manner to a third party without the prior written consent of Hacken.



Objectives

Web application security assessment was conducted in a “white box” mode (with an approved account) and had the following objectives:

- Identify technical and functional vulnerabilities
- Estimate their severity level (ease of use, impact on information systems)

This document is proprietary and confidential. No part of this document may be disclosed in any manner to a third party without the prior written consent of Hacken.

- Modeling the “most likely” attack vectors against the Customer’s Information System
- Proof of concept and exploitation of vulnerabilities
- Draw up a prioritized list of recommendations to address identified weaknesses

Limitations and Assumptions

This project is limited by the scope of this document

During this project, the Consultant will follow the following limitations:

- The operational impact to the networks will be maintained to the minimum and coordinated with the client
- No denial of service attacks will be used
- No active backdoor or Trojans will be installed
- No client data will be copied, modified, or destroyed

The following security tests shall be considered Out of Scope for this assessment:

- Internal networks assessment
- Denial of Service testing
- Physical Social Engineering testing

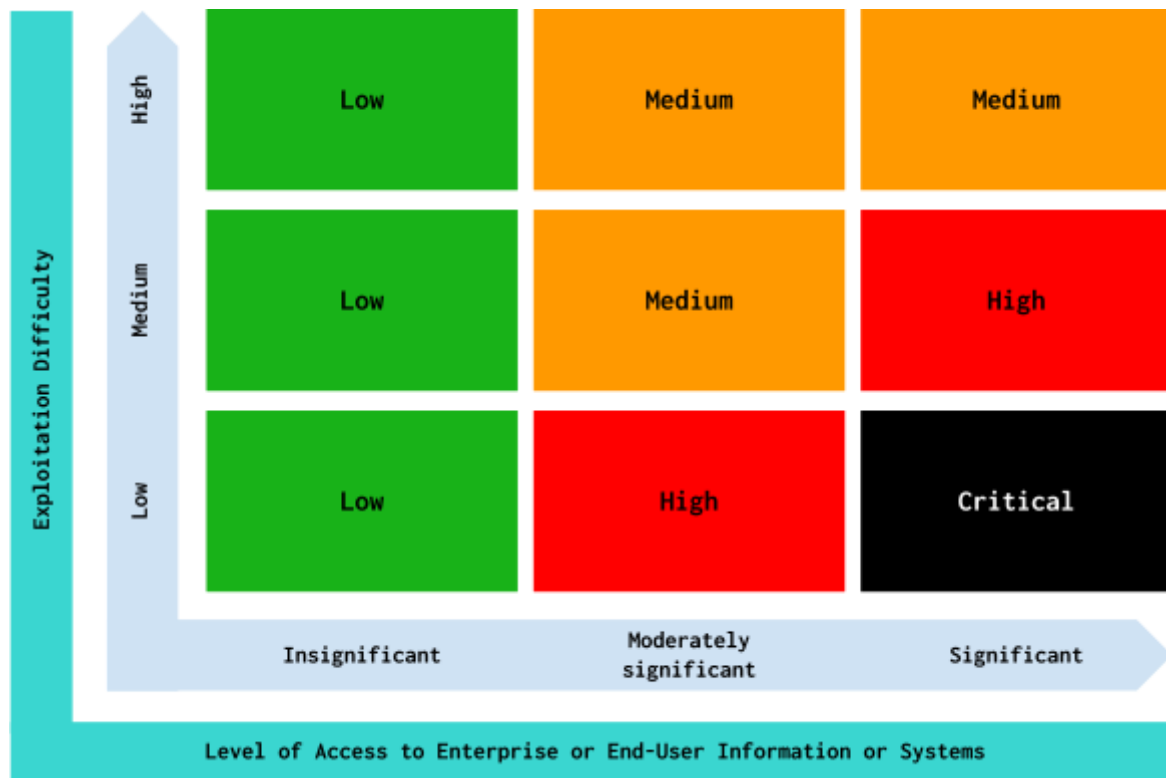
Disclaimer

This security assessment was conducted for the Customer prod environment and is valid on the date of the report submission hereto. The description of findings, recommendations, and risks was valid on the date of submission of the report hereto. Any projection to the future of the report’s information is subject to risk due to changes in the Infrastructure architecture, and it may no longer reflect its logic and controls.

Definitions & Abbreviations

The severity level (criticality level) of each vulnerability is determined based on the exploitation difficulty and the access level to an enterprise or end-user information system an attacker can gain in case of successful exploitation. The lower the exploitation difficulty level and the higher the access level which an attacker can get, the higher the vulnerability severity level will be. The matrix below illustrates the general methodology followed for identifying the severity level of each finding:

This document is proprietary and confidential. No part of this document may be disclosed in any manner to a third party without the prior written consent of Hacken.





To be the CVSS Compliant Hacken Security Assessment Team utilizes the Common Vulnerability Scoring System (CVSS) values for each level of vulnerability, such a solution will help prioritize the fixing vulnerabilities approach and make the results of the Security Assessment more objective.



The table below fully describes each level of vulnerabilities and ties to CVSS:

Severity	Color Map	Description
Informational		This level refers to vulnerabilities that do not pose an immediate security risk or require exploitation. Instead, they provide valuable information or insights about the system's configuration, weaknesses, or potential areas for improvement. While they may not directly lead to a security breach, addressing these informational

This document is proprietary and confidential. No part of this document may be disclosed in any manner to a third party without the prior written consent of Hacken.

Severity	Color Map	Description
		vulnerabilities can contribute to overall security enhancements and proactive risk mitigation efforts.
Low	 CVSS (0.1 - 3.9)	This level encompasses vulnerabilities with a low exploitation difficulty and low level of access. These vulnerabilities pose a relatively lower risk to the system's security as they are easier to exploit and grant minimal access privileges to potential attackers. While they still require attention and remediation, their impact is limited due to the restricted level of access gained.
Medium	 CVSS (4.0 - 6.9)	Vulnerabilities falling under this level have a moderate exploitation difficulty but the access level which can be gained by the attacker is greater compared to low-level vulnerabilities. They represent a medium level of risk to the system's security. Although they may be more challenging to exploit compared to low-level vulnerabilities, they still do not pose an immediate and severe threat. Appropriate measures should be taken to address these vulnerabilities promptly to prevent potential exploitation.

This document is proprietary and confidential. No part of this document may be disclosed in any manner to a third party without the prior written consent of Hacken.

Severity	Color Map	Description
High	 CVSS (7.0 - 8.9)	<p>Vulnerabilities categorized as high-level vulnerabilities have a low exploitation difficulty and grant a higher access level for the attacker in case of successful exploitation. These vulnerabilities pose a significant risk to the system's security and require immediate attention. While they may be more challenging to exploit, their potential impact is substantial. Timely remediation and mitigation measures should be implemented to address these vulnerabilities effectively.</p>
Critical	 CVSS (9.0 - 10.0)	<p>This level encompasses vulnerabilities with a low exploitation difficulty but the highest access level granted to the attacker in case of successful exploitation. These vulnerabilities are considered critical and pose the most severe threat to the system's security. Immediate action should be taken to remediate and address these vulnerabilities to prevent potential unauthorized access and significant security breaches.</p>

This document is proprietary and confidential. No part of this document may be disclosed in any manner to a third party without the prior written consent of Hacken.

Summary of Findings

Value	Number of issues (after remediation check)
Informational	0
Low	0
Medium	0
High	0
Critical	0
Unable to check	0

This document is proprietary and confidential. No part of this document may be disclosed in any manner to a third party without the prior written consent of Hacken.

Remediation Check Protocol

Issue Status Definition:

Status	Description
New	The issue has just been encountered by consultant. The remediation after the initial discovery was not yet made.
Acknowledged	The issue has not been fixed according to the given suggestion, or the implemented solution does not do enough to solve the issue.
Fixed	The issue has been fixed according to the given suggestion by the auditors.

Issue Status:

#	Finding	Status
1	Private Keys For Exchanges Could Be Restored From Locked Application	Fixed
2	Using Components With Known Vulnerabilities	Fixed
3	Arbitrary Origin Trusted	Fixed

This document is proprietary and confidential. No part of this document may be disclosed in any manner to a third party without the prior written consent of Hacken.

Key Findings

Web Applications Specific Vulnerabilities

■■■■■ **Critical** - CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H

■■■■■ **High** - CVSS:3.1/AV:N/AC:H/PR:N/UI:N/S:U/C:H/I:H/A:H

■■■■■ **Medium** - CVSS:3.1/AV:N/AC:H/PR:L/UI:N/S:U/C:L/I:L/A:H

■■■■■ **Low** - CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N

■■■■■ **Informational** - CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N

■■■■■ Private Keys For Exchanges Could Be Restored From Locked Application

#1	Description	CVSS:3.1/AV:P/AC:H/PR:L/UI:N/S:C/C:H/I:H/A:H
<p>An attacker who gains access to a user's computer can recover keys (including private ones) to crypto exchanges from a locked application. This is possible because the application stores the keys in clear text in the computer's RAM, even when it is in a locked state. Prerequisite for attack:</p> <ol style="list-style-type: none">1. User worked with application2. User locked application (But not closed browser)3. After that attacker has access to computer with user privilege)		
Vulnerable urls		
Evidences	<div><ol style="list-style-type: none">1. Unlock application and add at least one key pair for exchanges, for example for binance2. Lock application3. Go to dev-tools in Memory tab4. Create and save snapshot of background.js5. Then make next command in CLI<pre>cat cede_2_snapshot_after_lock.heapsnapshot grep -A 2 serverTime</pre><ol style="list-style-type: none">6. You should see your keys pair in output of command</div>	
<pre>n.razumovskiy@NB0845 Downloads % cat cede_2_snapshot_after_lock.heapsnapshot grep -A 2 serverTime "serverTime", "0x5b1f874d0b0c5ee17a495cbb70ab8bf64107a3bd", "fetch_deposits_withdrawals", -- {"serverTime":1695408541441}", "mOTzH1hzAM03S92v7785mWDRN2V19N1Qc25VxHcucJbylv7Kr8i4zRTHB1A4N6bu", "59YGZduu5j5HHYtPZNyAHv7uOK1jr41lADZple3pCGSF3gUaqigBPQX4pkYspPB1",</pre>		
Recommendations	<p>You need to store key pairs in encrypted form when application is locked. For example encrypted with user password</p>	

This document is proprietary and confidential. No part of this document may be disclosed in any manner to a third party without the prior written consent of Hacken.

Using Components With Known Vulnerabilities

#2	Description	CVSS:3.1/AV:N/AC:H/PR:H/UI:R/S:U/C:L/I:L/A:N				
The web application uses components with known vulnerabilities. That can make a vulnerable web application itself.						
Vulnerable files		monorepo-c1684ca5b842f006ff734118bec012-LINK-263-transfers-history-component-v2/yarn.lock				
Evidences						
		Library	Vulnerability	Installed Version	Fixed Version	Title
		@adobe/css-tools	CVE-2023-26364	4.0.1	4.3.1	@adobe/css-tools Regular Expression Denial of Service (ReDOS) while Parsing CSS https://avd.aquasec.com/nvd/cve-2023-26364
		aws-cdk-lib	CVE-2023-35165	2.50.0	2.80.0	AWS CDK EKS overly permissive trust policies https://avd.aquasec.com/nvd/cve-2023-35165
		braces	CVE-2018-1109	1.8.5	2.3.1	nodejs-braces: Regular Expression Denial of Service (ReDoS) in lib/parsers.js https://avd.aquasec.com/nvd/cve-2018-1109
		cookiejar	CVE-2022-25901	2.1.3	2.1.4	Regular Expression

This document is proprietary and confidential. No part of this document may be disclosed in any manner to a third party without the prior written consent of Hacken.

				Denial of Service (ReDoS) https://avd.aquasec.com/nvd/cve-2022-25901
d3-color	GHSA-36jr-mh4h-2g58	2.0.0	3.1.0	d3-color vulnerable to ReDoS https://github.com/advisories/GHSA-36jr-mh4h-2g58
decode-uri-component	CVE-2022-38900	0.2.0	0.2.1	improper input validation resulting in DoS https://avd.aquasec.com/nvd/cve-2022-38900
glob-parent	CVE-2020-28469	2.0.0/3.1.0	5.1.2	Regular expression denial of service https://avd.aquasec.com/nvd/cve-2020-28469
got	CVE-2022-33987	7.1.0	12.1.0, 11.8.5	missing verification of requested URLs allows redirects to UNIX sockets https://avd.aquasec.com/nvd/cve-2022-33987
http-cache-semantics	CVE-2022-25881	4.1.0	4.1.1	Regular Expression Denial of Service (ReDoS) vulnerability https://avd.aquasec.com/

This document is proprietary and confidential. No part of this document may be disclosed in any manner to a third party without the prior written consent of Hacken.

				nvd/cve-2022-25881
http-proxy	GHS-6x33-pw7p-hmpq	1.11.1	1.18.1	Denial of Service in http-proxy https://github.com/advisories/GHS-6x33-pw7p-hmpq
json5	CVE-2022-46175	0.4.0,1.0.1,2.2.1	2.2.2, 1.0.2	Prototype Pollution in JSON5 via Parse Method https://avd.aquasec.com/nvd/cve-2022-46175
loader-utils	CVE-2022-37601,CVE-2022-37599,CVE-2022-37603	1.4.0/2.0.2/3.2.0	1.4.2/2.0.4/3.2.1	prototype pollution https://avd.aquasec.com/nvd/cve-2022-37601 , regular expression denial of service https://avd.aquasec.com/nvd/cve-2022-37599 , Regular expression denial of service https://avd.aquasec.com/nvd/cve-2022-37603
lodash	CVE-2019-10744,CVE-2018-16487,CVE-2020-8203,CVE-2021-23337,CVE-2019-1010266,CVE-2020-28500,CVE-2018-3721	3.10.1	4.17.21	Multiple prototype pollutants, template injections and ReDoS. https://avd.aquasec.com/nvd/cve-2019-10744 , https://avd.aquasec.com/nvd/cve-2021-23337

This document is proprietary and confidential. No part of this document may be disclosed in any manner to a third party without the prior written consent of Hacken.

					-23337, https://avd.aquasec.com/nvd/cve-2019-1010266
	minimatch	CVE-2016-10540,CVE-2022-3517	2.0.10/3.0.4	3.0.5	https://avd.aquasec.com/nvd/cve-2016-10540 , ReDoS via the braceExpand function https://avd.aquasec.com/nvd/cve-2022-3517
	nth-check	CVE-2021-3803	1.0.2	2.0.1	inefficient regular expression complexity https://avd.aquasec.com/nvd/cve-2021-3803
	semver	CVE-2022-25883	5.7.1/6.3.0/7.0.0/7.3.7/7.3.8	7.5.2, 6.3.1, 5.7.2	Regular expression denial of service https://avd.aquasec.com/nvd/cve-2022-25883
	socket.io-parser	CVE-2022-2421	3.3.2	4.0.5, 4.2.1, 3.3.3, 3.4.2	Insufficient validation when decoding a Socket.IO packet https://avd.aquasec.com/nvd/cve-2022-2421
	tough-cookie	CVE-2023-26136	2.5.0/4.1.2	4.1.3	prototype pollution in cookie memstore https://avd.aquasec.com/nvd/cve-2023-26136
	trim	CVE-2020-775	0.0.1	0.0.3	nodejs-trim:

This document is proprietary and confidential. No part of this document may be disclosed in any manner to a third party without the prior written consent of Hacken.

		3			Regular Expression Denial of Service (ReDoS) in trim function https://avd.aquasec.com/nvd/cve-2020-7753
	trim-newlines	CVE-2021-33623	1.0.0	3.0.1, 4.0.1	ReDoS in .end() method https://avd.aquasec.com/nvd/cve-2021-33623
	vite	CVE-2023-34092	3.1.8	3.2.7, 4.0.5, 4.1.5, 4.2.3, 4.3.9	Vite Server Options (server.fs.deny) can be bypassed using double forward-slash (//) https://avd.aquasec.com/nvd/cve-2023-34092
	vm2	CVE-2023-29017, CVE-2023-29199, CVE-2023-30547, CVE-2023-32314, CVE-2023-37466, CVE-2023-37903, CVE-2023-32313	3.9.11	3.9.18	Sandbox Escape https://avd.aquasec.com/nvd/cve-2023-32314
	webpack	CVE-2023-28154	5.74.0, 5.75.0	5.76.0	avoid cross-realm objects https://avd.aquasec.com/nvd/cve-2023-28154
	word-wrap	CVE-2023-26115	1.2.3	1.2.4	ReDoS https://avd.aquasec.com/nvd/cve-2023-26115

This document is proprietary and confidential. No part of this document may be disclosed in any manner to a third party without the prior written consent of Hacken.

	yaml	CVE-2023-2251	2.2.1	2.2.2	Uncaught Exception in GitHub repository eemeli/yaml prior to 2.0.0-5 https://avd.aquasec.com/nvd/cve-2023-2251
Recommendations		Update packages to the latest versions			

■ Arbitrary Origin Trusted

#3	Description	CVSS:3.1/AV:P/AC:H/PR:L/UI:N/S:C/C:H/I:H/A:H
API trusted arbitrary origin, it reflects any domain in the origin header. For now it's safe but in future if you add new functions with backend interaction it could lead to security risks		
Vulnerable urls	https://api.cedelabs.io/*	
Evidences	<p>Make any request to API with arbitrary Origin header, for example</p> <pre>GET /v2/price/available_vs_currencies HTTP/2 Host: api.cedelabs.io Sec-Ch-Ua: "Chromium";v="117", "Not;A=Brand";v="8" Accept: application/json, text/plain, */* X-Token: cc4f48b2-c932-4dac-91ce-8acb0c28b18c Sec-Ch-Ua-Mobile: ?0 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/117.0.5938.63 Safari/537.36 Sec-Ch-Ua-Platform: "macOS" Sec-Fetch-Site: none Sec-Fetch-Mode: cors Sec-Fetch-Dest: empty Accept-Encoding: gzip, deflate, br Accept-Language: ru-RU,ru;q=0.9,en-US;q=0.8,en;q=0.7 Origin: https://evil.com</pre>	

This document is proprietary and confidential. No part of this document may be disclosed in any manner to a third party without the prior written consent of Hacken.

You can see reflected domain in Access-Control-Allow-Origin response header

Request	Response
<pre> 1 GET /v2/price/available_vs_currencies HTTP/2 2 Host: api.cedelabs.io 3 Sec-Ch-Ua: "Chromium";v="117", "Not;A=Brand";v="8" 4 Accept: application/json, text/plain, */* 5 X-Token: ccaf48b2-c932-4dac-91ce-8acb0c28b18c 6 Sec-Ch-Ua-Mobile: ?0 7 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/117.0.5938.63 Safari/537.36 8 Sec-Ch-Ua-Platform: "macOS" 9 Sec-Fetch-Site: none 10 Sec-Fetch-Mode: cors 11 Sec-Fetch-Dest: empty 12 Accept-Encoding: gzip, deflate, br 13 Accept-Language: ru-RU, ru;q=0.9, en-US;q=0.8, en;q=0.7 14 Origin: https://evil.com 15 16 </pre>	<pre> 1 HTTP/2 200 OK 2 Date: Sun, 24 Sep 2023 18:35:29 GMT 3 Content-Type: application/json; charset=utf-8 4 Content-Length: 458 5 Content-Security-Policy: default-src 'self';base-uri 'self';font-src 'self' https: data:;form-action 'self';frame-ancestors 'self';img-src 'self' data:;object-src 'none';script-src 'self';script-src-attr 'none';style-src 'self' https: 'unsafe-inline';upgrade-insecure-requests 6 Cross-Origin-Embedder-Policy: require-corp 7 Cross-Origin-Opener-Policy: same-origin 8 Cross-Origin-Resource-Policy: same-origin 9 Origin-Agent-Cluster: ?1 10 Referrer-Policy: no-referrer 11 Strict-Transport-Security: max-age=15552000; includeSubDomains 12 X-Content-Type-Options: nosniff 13 X-Dns-Prefetch-Control: off 14 X-Download-Options: noopen 15 X-Frame-Options: SAMEORIGIN 16 X-Permitted-Cross-Domain-Policies: none 17 X-Xss-Protection: 0 18 Vary: Origin 19 Access-Control-Allow-Origin: https://evil.com 20 21 { "data":{ "updatedAt":"Sun, 24 Sep 2023 18:34:34 GMT", "currencies":{ "fiat":{ "USD", "EUR" } } } } </pre>

Recommendations

Whitelist reflection domains to sensitive API functions

This document is proprietary and confidential. No part of this document may be disclosed in any manner to a third party without the prior written consent of Hacken.

Appendix A. OWASP Testing Checklist

Category	Test Name	Result	Details
Information Gathering			
OTG-INFO-001	Conduct Search Engine Discovery and Reconnaissance for Information Leakage	Done	Manual testing
OTG-INFO-002	Fingerprint Web Server	Done	Done with whatweb and nmap
OTG-INFO-003	Review Webserver Metafiles for Information Leakage	Done	Manual testing
OTG-INFO-004	Enumerate Applications on Webserver	Done	Done with whatweb and nmap
OTG-INFO-005	Review Webpage Comments and Metadata for Information Leakage	Done	Done with dirbuster
OTG-INFO-006	Identify application entry points	Done	Done with Burp Suite
OTG-INFO-007	Map execution paths through application	Done	Done with Burp Suite
OTG-INFO-008	Fingerprint Web Application Framework	Done	Done with whatweb and nmap
OTG-INFO-009	Fingerprint Web Application	Done	Done with whatweb and nmap
OTG-INFO-010	Map Application Architecture (WAF, Application server, identify application architecture)	Tested	Private Keys For Exchanges Could Be Restored From Locked Application
Configuration and Deploy Management Testing			
OTG-CONFIG-001	Test Network/Infrastructure Configuration	Tested	No vulnerability detected
OTG-CONFIG-002	Test Application Platform Configuration	Tested	Using Components With Known Vulnerabilities
OTG-CONFIG-003	Test File Extensions Handling for Sensitive Information	Tested	No vulnerability detected
OTG-CONFIG-004	Backup and Unreferenced Files for Sensitive Information	Tested	No vulnerability detected
OTG-CONFIG-005	Enumerate Infrastructure and Application Admin Interfaces	Tested	No vulnerability detected
OTG-CONFIG-006	Test HTTP Methods	Tested	No vulnerability detected
OTG-CONFIG-007	Test HTTP Strict Transport Security	Tested	No vulnerability detected
OTG-CONFIG-008	Test RIA cross domain policy	Tested	No vulnerability detected
Identity Management Testing			
OTG-IDENT-001	Test Role Definitions	Tested	No vulnerability detected
OTG-IDENT-002	Test User Registration Process	Tested	No vulnerability detected
OTG-IDENT-003	Test Account Provisioning Process	Tested	No vulnerability detected
OTG-IDENT-004	Testing for Account Enumeration and Guessable User Account	Tested	No vulnerability detected
OTG-IDENT-005	Testing for Weak or unenforced username policy	Tested	No vulnerability detected

This document is proprietary and confidential. No part of this document may be disclosed in any manner to a third party without the prior written consent of Hacken.

OTG-IDENT-006	Test Permissions of Guest/Training Accounts	Tested	No vulnerability detected
OTG-IDENT-007	Test Account Suspension/Resumption Process	Tested	No vulnerability detected
Authentication Testing			
OTG-AUTHN-001	Testing for Credentials Transported over an Encrypted Channel	Tested	No vulnerability detected
OTG-AUTHN-002	Testing for default credentials	Tested	No vulnerability detected
OTG-AUTHN-003	Testing for Weak lock out mechanism	Tested	No vulnerability detected
OTG-AUTHN-004	Testing for bypassing authentication schema	Tested	No vulnerability detected
OTG-AUTHN-005	Test remember password functionality	Tested	No vulnerability detected
OTG-AUTHN-006	Testing for Browser cache weakness	Tested	No vulnerability detected
OTG-AUTHN-007	Testing for Weak password policy	Tested	No vulnerability detected
OTG-AUTHN-008	Testing for Weak security question/answer	Tested	No vulnerability detected
OTG-AUTHN-009	Testing for weak password change or reset functionalities	Tested	No vulnerability detected
OTG-AUTHN-010	Testing for Weaker authentication in alternative channel	Tested	No vulnerability detected
Authorization Testing			
OTG-AUTHZ-001	Testing Directory traversal/file include	Tested	No vulnerability detected
OTG-AUTHZ-002	Testing for bypassing authorization schema	Tested	No vulnerability detected
OTG-AUTHZ-003	Testing for Privilege Escalation	Tested	No vulnerability detected
OTG-AUTHZ-004	Testing for Insecure Direct Object References	Tested	No vulnerability detected
Session Management Testing			
OTG-SESS-001	Testing for Bypassing Session Management Schema	Tested	No vulnerability detected
OTG-SESS-002	Testing for Cookies attributes	Tested	No vulnerability detected
OTG-SESS-003	Testing for Session Fixation	Tested	No vulnerability detected
OTG-SESS-004	Testing for Exposed Session Variables	Tested	No vulnerability detected
OTG-SESS-005	Testing for Cross Site Request Forgery	Tested	No vulnerability detected
OTG-SESS-006	Testing for logout functionality	Tested	No vulnerability detected
OTG-SESS-007	Test Session Timeout	Tested	No vulnerability detected
OTG-SESS-008	Testing for Session puzzling	Tested	No vulnerability detected
Data Validation Testing			
OTG-INPVAL-001	Testing for Reflected Cross Site Scripting	Tested	No vulnerability detected
OTG-INPVAL-002	Testing for Stored Cross Site Scripting	Tested	No vulnerability detected
OTG-INPVAL-003	Testing for HTTP Verb Tampering	Tested	No vulnerability detected
OTG-INPVAL-004	Testing for HTTP Parameter pollution	Tested	No vulnerability detected
OTG-INPVAL-005	Testing for SQL Injection	Tested	No vulnerability detected
OTG-INPVAL-006	Testing for LDAP Injection	Tested	No vulnerability detected
OTG-INPVAL-007	Testing for ORM Injection	Tested	No vulnerability detected
OTG-INPVAL-008	Testing for XML Injection	Tested	No vulnerability detected
OTG-INPVAL-009	Testing for SSI Injection	Tested	No vulnerability detected

This document is proprietary and confidential. No part of this document may be disclosed in any manner to a third party without the prior written consent of Hacken.

OTG-INPVAL-010	Testing for XPath Injection	Tested	No vulnerability detected
OTG-INPVAL-011	IMAP/SMTP Injection	Tested	No vulnerability detected
OTG-INPVAL-012	Testing for Code Injection	Tested	No vulnerability detected
OTG-INPVAL-013	Testing for Command Injection	Tested	No vulnerability detected
OOTG-INPVAL-014	Testing for Buffer overflow	Tested	No vulnerability detected
OTG-INPVAL-015	Testing for incubated vulnerabilities	Tested	No vulnerability detected
OTG-INPVAL-016	Testing for HTTP Splitting/Smuggling	Tested	No vulnerability detected
Error Handling			
OTG-ERR-001	Analysis of Error Codes	Tested	No vulnerability detected
OTG-ERR-002	Analysis of Stack Traces	Tested	No vulnerability detected
Cryptography			
OTG-CRYPST-001	Testing for Weak SSL/TLS Ciphers, Insufficient Transport Layer Protection	Tested	No vulnerability detected
OTG-CRYPST-002	Testing for Padding Oracle	Tested	No vulnerability detected.
OTG-CRYPST-003	Testing for Sensitive information sent via unencrypted channels	Tested	No vulnerability detected
Client Side Testing			
OTG-CLIENT-001	Testing for DOM based Cross Site Scripting	Tested	No vulnerability detected.
OTG-CLIENT-002	Testing for JavaScript Execution	Tested	No vulnerability detected
OTG-CLIENT-003	Testing for HTML Injection	Tested	No vulnerability detected
OTG-CLIENT-004	Testing for Client Side URL Redirect	Tested	No vulnerability detected
OTG-CLIENT-005	Testing for CSS Injection	Tested	No vulnerability detected
OTG-CLIENT-006	Testing for Client Side Resource Manipulation	Tested	No vulnerability detected
OTG-CLIENT-007	Test Cross Origin Resource Sharing	Tested	Arbitrary Origin Sharing
OTG-CLIENT-008	Testing for Cross Site Flashing	Tested	No vulnerability detected
OTG-CLIENT-011	Test Web Messaging	Tested	No vulnerability detected
OTG-CLIENT-012	Test Local Storage	Tested	No sensitive data stored in Local or Session storage detected
Business Logic Testing			
OTG-BUSLOGIC-001	Test Business Logic Data Validation	Tested	No vulnerability detected
OTG-BUSLOGIC-002	Test Ability to Forge Request	Tested	No vulnerability detected
OTG-BUSLOGIC-003	Test Integrity Checks	Tested	No vulnerability detected
OTG-BUSLOGIC-004	Test for Process Timing	Tested	No vulnerability detected
OTG-BUSLOGIC-005	Test Numbers of Times a Function Can be Used Limits	Tested	No vulnerability detected
OTG-BUSLOGIC-006	Test for the Circumvention of Work Flows	Tested	No vulnerability detected
OTG-BUSLOGIC-007	Test Upload of Unexpected File Types	Tested	No vulnerability detected
OTG-BUSLOGIC-008	Test Upload of Malicious Files	Tested	No vulnerability detected

This document is proprietary and confidential. No part of this document may be disclosed in any manner to a third party without the prior written consent of Hacken.