// HALBORN

Umee

WebApp Pentest

Prepared by: **Halborn**

Date of Engagement: **March 3rd, 2022 - March 17th, 2022**

Visit: **Halborn.com**

DOCUMENT REVISION HISTORY

VERSION	MODIFICATION	DATE	AUTHOR
0.1	Document Creation	03/03/2022	Alessandro Cara
0.2	Document Amended	03/09/2022	Alessandro Cara
0.3	Document Draft Review	03/10/2022	Gabi Urrutia
0.4	Document Amended	03/15/2022	Alessandro Cara
0.5	Second Draft Review	03/17/2022	Gabi Urrutia
1.0	Remediation Plan	04/29/2022	Alessandro Cara
1.1	Final Document Review	04/29/2022	Gabi Urrutia

CONTACTS

CONTACT	COMPANY	EMAIL
Rob Behnke	Halborn	Rob.Behnke@halborn.com
Steven Walbroehl	Halborn	Steven.Walbroehl@halborn.com
Gabi Urrutia	Halborn	Gabi.Urrutia@halborn.com

EXECUTIVE OVERVIEW

1.1 INTRODUCTION

Umee engaged Halborn to conduct a security audit on three of their applications, beginning on March 3rd, 2022 and ending on March 17th, 2022 . The security assessment was scoped to the web application described in the scope section of this report.

The platform allowed its users to:

- Borrow and lend Crypto
- Bridge Tokens between Umee and Ethereum
- Review open positions and collateral provided
- Redirect to the Keplr application to stake tokens and participate in the governance of the protocol

1.2 AUDIT SUMMARY

The team at Halborn was provided two weeks for the engagement and assigned a full-time security engineer to audit the security of the platform. The security engineer is a blockchain and smart-contract security expert with advanced penetration testing, smart-contract hacking, and deep knowledge of multiple blockchain protocols.

The purpose of this audit was to:

- Ensure that the applications work as intended
- Identify potential security issues with the applications

The attack surface of the application was limited, due to the architecture of the application requiring interaction between a web3 wallet (Metamask and Keplr) and the infrastructure provider Infura. Finally, the backend logic was handled by the smart contracts rather than being processed by a middleware service such as an API.

After the first draft of the report was provided to the Umee team, functionality to communicate and lend and borrow assets on their native blockchain was introduced. This consisted of small code changes, with some GET requests to a middleware API, which subsequently communicated with the native blockchain. All the requests to the API did not involve any sensitive user data, and transactions were handled by the Keplr wallet.

Although the application contained a limited attack surface, Halborn was able to identify few security risks which diminished the overall security posture of the application.

First, the platform used some third-party dependencies which were outdated and contained publicly disclosed vulnerabilities. Should an attacker be able to understand how these libraries are used, they would be able to execute malicious code and potentially access sensitive information.

Furthermore, the application lacked certain defense-in-depth controls, such as security related headers. While this does not provide a risk on its own, attackers might be able to exploit additional vulnerabilities which these controls could have prevented.

1.3 TEST APPROACH & METHODOLOGY

RISK METHODOLOGY:

Vulnerabilities or issues observed by Halborn are ranked based on the risk assessment methodology by measuring the **LIKELIHOOD** of a security incident and the **IMPACT** should an incident occur. This framework works for communicating the characteristics and impacts of technology vulnerabilities. The quantitative model ensures repeatable and accurate measurement while enabling users to see the underlying vulnerability characteristics that were used to generate the Risk scores. For every vulnerability, a risk level will be calculated on a scale of 5 to 1 with 5 being the highest likelihood or impact.

RISK SCALE - LIKELIHOOD

5 - Almost certain an incident will occur.
4 - High probability of an incident occurring.
3 - Potential of a security incident in the long term.
2 - Low probability of an incident occurring.
1 - Very unlikely issue will cause an incident.

RISK SCALE - IMPACT

5 - May cause devastating and unrecoverable impact or loss.
4 - May cause a significant level of impact or loss.
3 - May cause a partial impact or loss to many.
2 - May cause temporary impact or loss.
1 - May cause minimal or un-noticeable impact.

The risk level is then calculated using a sum of these two values, creating a value of 10 to 1 with 10 being the highest level of security risk.

10 - CRITICAL
9 - 8 - HIGH
7 - 6 - MEDIUM
5 - 4 - LOW
3 - 1 - VERY LOW AND INFORMATIONAL

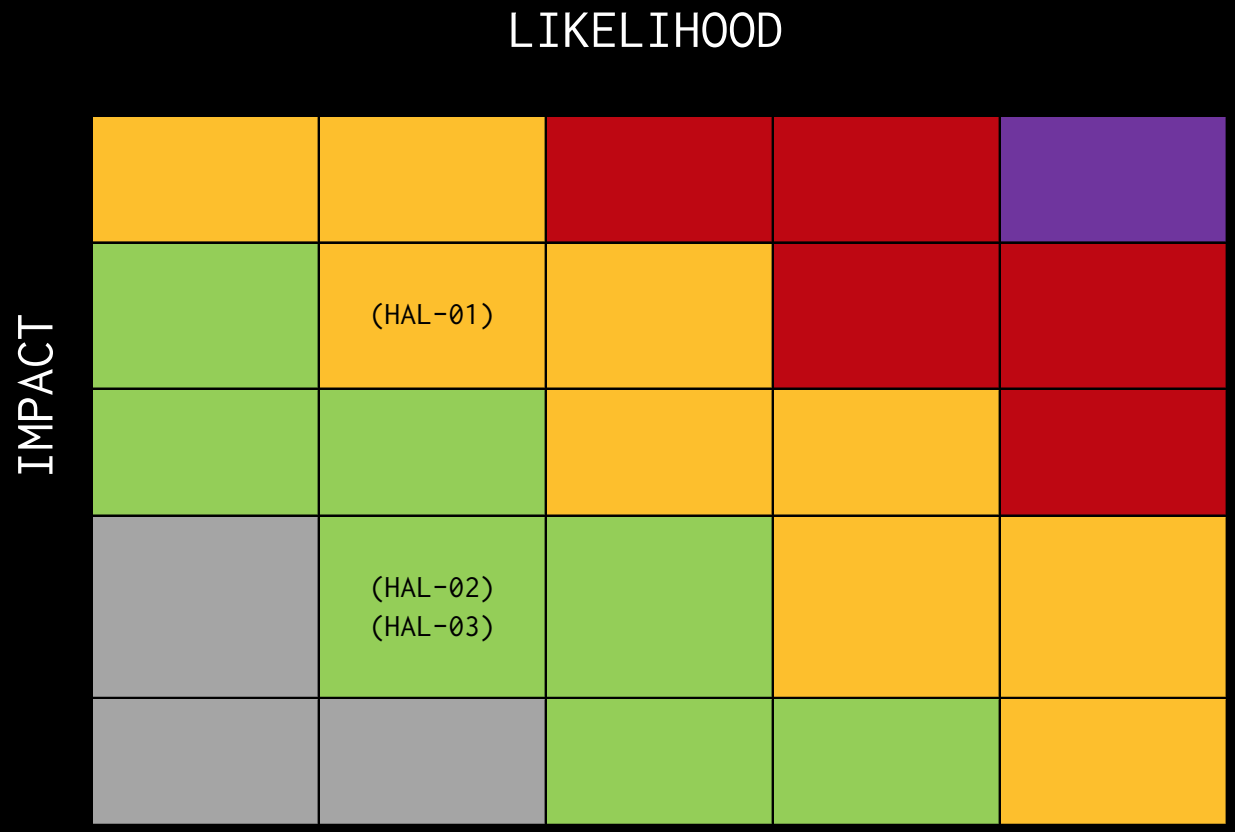
1.4 SCOPE

The assessment was scoped to the application available under a private GitHub repository dev branch. The application was run locally and connected with the Goerli testnet.

After the first draft of the report was completed, Umee amended the solution to connect to their native chain too. The changes were applied to the convexity branch of their private GitHub repository.

2. ASSESSMENT SUMMARY & FINDINGS OVERVIEW

CRITICAL	HIGH	MEDIUM	LOW	INFORMATIONAL
0	0	1	2	0



EXECUTIVE OVERVIEW

SECURITY ANALYSIS	RISK LEVEL	REMEDIATION DATE
HAL-01 - USE OF PACKAGES WITH KNOWN VULNERABILITIES	Medium	RISK ACCEPTED
HAL-02 - DEPENDENCIES NOT PINNED TO THE EXACT VERSION	Low	SOLVED - 04/29/2022
HAL-03 - MISSING HTTP SECURITY HEADERS	Low	RISK ACCEPTED

FINDINGS & TECH DETAILS

3.1 (HAL-01) USE OF PACKAGES WITH KNOWN VULNERABILITIES - MEDIUM

Description:

The platform components use third-party dependencies to delegate the handling of different types of operations, e.g., generation of documents in a specific format, HTTP communications, data parsing of a specific format, etc. However, the dependency has an expected downside where the actual application's security posture now rests on it.

Vulnerabilities List:

Title	Package	Severity
Prototype Pollution	immer	Critical
Authorization Bypass	url-parse	Critical
Uncontrolled Resource Consumption	ansi-html	High
Incorrect Comparison	axios	High
Exposure of Sensitive Information	follow-redirects	High
Exposure of Sensitive Information	node-fetch	High
Exposure of Sensitive Information	simple-get	High
Denial of Service (DoS)	glob-parent	High

Risk Level:

Likelihood - 2
Impact - 4

Recommendation:

It is strongly recommended to perform an automated analysis of dependencies from project birth to ensure they do not contain security issues. Developers should be aware of this and apply the required mitigation measures to protect the affected components.

Update of packages can be done with `npm update`. On the other hand, `npm audit` can be used to monitor outdated and vulnerable third-party dependencies.

Remediation Plan:

RISK ACCEPTED: Umee updated all packages that could be updated without introducing breaking changes. However, certain packages make use of nested vulnerable packages, which would require Umee to downgrade them to a safe version. This might introduce further vulnerabilities and cause errors in the code. The development team will continue to monitor updates to the affected packages to ensure they are fixed as soon as possible.

3.2 (HAL-02) DEPENDENCIES NOT PINNED TO EXACT VERSIONS - LOW

Description:

The application contains some external dependencies, some of which are not locked to an exact version, but are set to a compatible version (^x.x.x). This can potentially allow dependency attacks, as seen with the flow of events package with Copay Bitcoin Wallet.

Details:

Listing 1

```
{
    "@babel/core": "^7.12.3",
    "@cosmjs/launchpad": "^0.24.1",
    "@cosmjs/stargate": "^0.26.3",
    "@emotion/react": "^11.7.1",
    "@emotion/styled": "^11.6.0",
    "@headlessui/react": "^1.4.2",
    "@keplr-wallet/common": "^0.9.7",
    "@keplr-wallet/cosmos": "^0.9.7",
    "@keplr-wallet/crypto": "^0.9.7",
    "@keplr-wallet/hooks": "^0.9.7",
    "@keplr-wallet/provider": "^0.9.6",
    "@keplr-wallet/stores": "^0.9.7",
    "@keplr-wallet/types": "^0.9.6",
    "@keplr-wallet/unit": "^0.9.6",
    "@keplr-wallet/wc-client": "^0.9.7",
    "@keplr-wallet/wc-qrcode-modal": "^0.9.0",
    "@types/lodash": "^4.14.172",
    "@types/parcel-env": "^0.0.1",
    "@walletconnect/browser-utils": "^1.6.6",
    "@walletconnect/client": "^1.6.6",
    "@walletconnect/types": "^1.6.6",
    "@walletconnect/web3-provider": "^1.4.1",
    "bcryptjs": "^2.4.3",
    "bech32": "^2.0.0",
    "clsx": "^1.1.1",
```

```
27      "ethers": "^5.3.1",
28      "google-protobuf": "^3.19.1",
29      "grommet": "^2.17.3",
30      "grommet-icons": "^4.6.0",
31      "lodash": "^4.17.21",
32      "mobx-react-lite": "^3.2.2",
33      "parcel-bundler": "^1.12.5",
34      "polished": "^4.1.3",
35      "react": "^17.0.2",
36      "react-bootstrap": "^1.6.4",
37      "react-dom": "^17.0.2",
38      "react-error-boundary": "^3.1.3",
39      "react-query": "^3.33.5",
40      "react-router-dom": "^5.2.0",
41      "react-scripts": "4.0.3",
42      "react-toastify": "^7.0.4",
43      "styled-components": "^5.3.1",
44      "web3": "^1.4.0",
45      "web3modal": "^1.9.3"
46    }
```

Risk Level:

Likelihood - 2
Impact - 2

Recommendation:

Pinning dependencies to an exact version (=x.x.x) can reduce the chance of inadvertently introducing a malicious version of a dependency in the future.

Remediation Plan:

SOLVED: Umee pinned the packages to the exact version.

3.3 (HAL-03) MISSING HTTP SECURITY HEADERS - LOW

Description:

The assessment revealed that in-scope application did not enforce various security headers. These headers are used by client browsers to ensure that various security controls are properly implemented during normal application operation.

- X-Frame-Options, which indicates whether to allow a browser to render a page in a <frame>, <iframe>, <embed> or <object>. Attackers might exploit a Clickjacking vulnerability to embed sensitive pages into a frame and have the user submit data without their knowledge.
- Content-Security-Policy, which allows website administrators to control what resources the user agent can load for a given page.
- Strict-Transport-Security (HSTS) - which enforces secure transmission by allowing a website to tell browsers that it should only be accessed via HTTPS, rather than HTTP.
- Referrer-Policy - specifies what information, or parts of it, should be sent in the Referer header with the request that caused the redirection.

It should be noted that in some situations Strict-Transport-Security and X-Content-Type-Options (and in some cases X-Frame-Options) do not necessarily have to be enabled by the application **directly**. These headers can be injected by compatible load balancers or web application accelerators.

Risk Level:

Likelihood - 2
Impact - 2

Recommendation:

Umee should review the above security headers and ensure that, where applicable, these headers are included on all exposed endpoints and services. This allows the development team to ensure that the defence-in-depth approach is achieved throughout the application.
The following sections provide suggestions as to how to implement these controls. For further information, please refer to the links provided for further reading provided.

HSTS

The following header should be added by the web server to enforce HSTS:

Listing 2

```
1 Strict-Transport-Security "max-age=31536000; includeSubDomains
```

Frame Options

The following header should be enforced to prevent the page from being included in a frame of another website:

Listing 3

```
1  X-Frame-Options: sameorigin
```

or the following should be used to always prevent the browser from loading the page in a frame, even from within the same site.

Listing 4

```
1 X-Frame-Options: DENY
```

CSP

The following header presents a locked down content security policy for web applications:

Listing 5

```
1 Content-Security-Policy "default-src 'none'; script-src 'self';
↳ connect-src 'self'; img-src 'self'; style-src 'self'; frame-
↳ ancestors 'self';" always
```

REFERRER POLICY

Please refer to the guide on Mozilla's website available at this link.

References:

X-Content-Type-Options

X-Frame-Options

Content-Security-Policy

HSTS

Remediation Plan:

RISK ACCEPTED: Umee will look to implement headers on the hosting side at a later date.

THANK YOU FOR CHOOSING

// HALBORN