|  |
| --- |
| Nomesh |
| Vulnerability Report |
| undefined |

modifications history

| **Version** | **Date** | **Author** | **Description** |
| --- | --- | --- | --- |
| 1.0 | undefined | Nomesh | Initial Version |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

1. General Information 4

1.1 Scope 4

1.2 Organisation 4

2. Executive Summary 5

3. Technical Details 6

3.1 title 6

4. Vulnerabilities summary 8

# General Information

## Scope

undefined has mandated us to perform security tests on the following scope:

## Organisation

The testing activities were performed between undefined and undefined.

# Executive Summary

# Vulnerabilities summary

Following vulnerabilities have been discovered:

|  |  |  |  |
| --- | --- | --- | --- |
| **Risk** | **ID** | **Vulnerability** | **Affected Scope** |
| Medium | VULN-001 | XSS |  |
| Medium | VULN-003 | CSRF |  |

# Technical Details

## XSS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| CVSS Severity | Medium | | CVSSv3 Score | | 4.6 |
| **CVSSv3 criterias** | Attack Vector : | **Network** | Scope : | **Unchanged** | |
| Attack Complexity : | **High** | Confidentiality : | **Low** | |
| Required Privileges : | **Low** | Integrity : | **Low** | |
| User Interaction : | **Required** | Availability : | **Low** | |
| **Affected scope** |  | | | | |
| **Description** | Summary:  Stored XSS can be submitted on reports, and anyone who will check the report the XSS will trigger.  Description:  Stored XSS, also known as persistent XSS, is the more damaging than non-persistent XSS. It occurs when a malicious script is injected directly into a vulnerable web application. | | | | |
| **Observation** | Steps To Reproduce:  Go to https://app.mopub.com/reports/custom/  Click New network report.  On the name, enter payload: "><img src=x onerror=alert(document.domain)>  Click Run and save then XSS will trigger.  Demonstration of the vulnerability:  PoC: xssed.webm (F412243)  Tested on Firefox and chrome. | | | | |
| **Test details** | | | | | |
| **Remediation** | The attacker can steal data from whoever checks the report. | | | | |
| **References** | https://hackerone.com/reports/485748 | | | | |

## CSRF

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| CVSS Severity | Medium | | CVSSv3 Score | | 4.5 |
| **CVSSv3 criterias** | Attack Vector : | **Physical** | Scope : | **Unchanged** | |
| Attack Complexity : | **High** | Confidentiality : | **None** | |
| Required Privileges : | **Low** | Integrity : | **Low** | |
| User Interaction : | **Required** | Availability : | **High** | |
| **Affected scope** |  | | | | |
| **Description** | Cross-site request forgery (also known as CSRF) is a web security vulnerability that allows an attacker to induce users to perform actions that they do not intend to perform. It allows an attacker to partly circumvent the same origin policy, which is designed to prevent different websites from interfering with each other. | | | | |
| **Observation** | SITE uses the authenticity\_token token during login to prevent CSRF. However, the authenticity\_token token is not properly verified, so an attacker can log in via CSRF without the authenticity\_token token. In other words, Hacker One thinks that it implemented CSRF token through authenticity\_token token, but it is not. | | | | |
| **Test details** | | | | | |
| **Remediation** | The victim may add sensitive payment information to the attacker's new account | | | | |
| **References** | https://hackerone.com/reports/834366 | | | | |