EQS Integrity Line Cross Site Scripting / Information Disclosure

Authored by Giovanni Pellerano | Site ush.it

Posted Jul 6, 2022

EQS Integrity Line versions through 2022-07-01 suffer from cross site scripting and sensitive information disclosure vulnerabilities.

tags | exploit, vulnerability, xss, info disclosure
advisories | CVE-2022-34007
SHA-256 | 21a47b9fbd11e6b29f25b46678eefbe67cd2ecd6ec063d64543e0d2d12795718 Download | Favorite | View

Related Files

Share This

Like 0 Tweet LinkedIn Reddit Digg StumbleUpon

Change Mirror Download

```
EQS Integrity Line: Multiple Vulnerabilities

 Name              Multiple Vulnerabilities in EQS Integrity Line
 Systems Affected  EQS Integrity Line through 2022-07-01
 Severity          High
 Impact (CVSSv2)   High 8.8/10, score: (AV:N/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:H)
 Vendor            EQS Group AG (https://www.eqs.com/)
 Advisory          http://www.ush.it/team/ush/advisory-eqs-integrity-line/eqs_integrity_line.txt
 Authors           Giovanni "evilaliv3" Pellerano (evilaliv3 AT ush DOT it)
 Date              20220706

I. BACKGROUND

EQS Integrity Line is a proprietary whistleblowing software which enables
employees to report misconduct such as corruption, abuses of power and
discrimination internally before complaints become public and, in serious
cases, result in financial losses as well as reputational damage.

II. DESCRIPTION

Multiple Vulnerabilities exist in EQS Integrity Line software.

The present advisory highlights two distinct vulnerabilities, namely (A)
XSS Vulnerability (stored) [CVE-2022-34007] and (B) Use of GET Request
Method With Sensitive Query Strings [CWE-598].

III. ANALYSIS

A) XSS Vulnerability (stored) [CVE-2022-34007]

EQS Integrity Line through 2022-07-01 allows a stored XSS via a crafted
whistleblower entry.

In order to exploit this vulnerability no account is required on the
whistleblowing software.

The vulnerability resides in the whistleblowing questionnaire
implementation that enables anonymous, non authenticated, users to inject
malicious XSS vectors due to missing or improper input sanitization.
Also content security policies (CSP) that could prevent or limit the attack
are absent.

The vulnerability is present on the whistleblowing form, and can be
triggered using the following example input:

--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--

<img src= onerror=alert(document.cookie)>

--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--8<--

Due to the vulnerability, an attacker posing as a whistleblower could
upload an XSS vector in the submission form loading malicious code to be
reflected and executed in the context of the browser session of the
Recipient of the submission, that is typically an Anticorruption Officer
or an Internal Auditor.

Being able to execute code in the context of the target, and due to the
absence of additional mitigations (e.g. the HttpOnly flag for cookies)
the attacker could possibly obtain a copy of the target session cookie
useful to impersonate and operate in place of the target user and
execute automated operations on behalf of the target user by accessing
all the reports present on the system or possibly impact the integrity
of the system by deleting reports or interfering with ongoing
communications with a real whistleblower.

In short: a standard XSS attack scenario.

The test for the presence of this vulnerability has been performed on the
first input only, to not risk to cause any damage to the application.
It is advised to execute a proper complete audit of the application with
respect to this kind of vulnerability.
```


File Archive: November 2022 <

Su	Mo	Tu	We	Th	Fr	Sa
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30			

Top Authors In Last 30 Days

- **Red Hat** 188 files
- **Ubuntu** 57 files
- **Gentoo** 44 files
- **Debian** 28 files
- **Apple** 25 files
- **Google Security Research** 14 files
- **malvuln** 10 files
- **nu11secur1ty** 6 files
- **mjurczyk** 4 files
- **George Tsimpidas** 3 files

File Tags

- ActiveX (932)
- Advisory (79,557)
- Arbitrary (15,643)
- BBS (2,859)
- Bypass (1,615)
- CGI (1,015)
- Code Execution (6,913)
- Conference (672)
- Cracker (840)
- CSRF (3,288)
- DoS (22,541)
- Encryption (2,349)
- Exploit (50,293)
- File Inclusion (4,162)
- File Upload (946)
- Firewall (821)
- Info Disclosure (2,656)

File Archives

- November 2022
- October 2022
- September 2022
- August 2022
- July 2022
- June 2022
- May 2022
- April 2022
- March 2022
- February 2022
- January 2022
- December 2021
- Older

Systems

- AIX (426)
- Apple (1,926)

```
The vulnerability was first identified performing an independent security
audit to evaluate and ensure the security of the EU Sanctions Whistleblower
Tool of the European Commission enabling whistleblowers to report possible
violation of EU sanctions hosted at:

https://eusanctions.integrityline.com/

B) Use of GET Request Method With Sensitive Query Strings [CWE-598]

EQS Integrity Line through 2022-07-01 leaves sensitive traces in the browser
history of whistleblowers using the application and possibly in the logs
of other network appliances involved in the communication.

When a whistleblower makes a submission, the system assigns a unique
identifier to the submission and enables to choose a pin that is intended
to be used by users in combination with the unique identifier to access
the system in order to communicate with the recipients of their own report.

The implementation of the session makes use of GET variables that include
the unique identifier in the navigated URL to access the report.
Such an implementation is prone to sensible information leakage making it
possible for an auditor accessing the browser history of the
whistleblower's device to clearly identify the evidence of a performed
submission.

It is advised to perform full review of the application to get sure that
the application reduces the sensible traces left in the browser history of
the user.

IV. WORKAROUND

The vendor has fixed the XSS and implemented a CSP in date 2022-07-01

V. CVE INFORMATION

XSS Vulnerability (stored) [CVE-2022-34007]
Use of GET Request Method With Sensitive Query Strings [CWE-598]

VI. DISCLOSURE TIMELINE

20220617 USH: Bugs discovered
20220617 USH: Contacted Mitre for CVE Assignment
20220621 USH: First vendor contact (Lorenzo Trevisiol, Laura Santeusanio)
20220622 USH: Advisory provided to the vendor (Goran Kozomara)
20220701 Vendor response: XSS confirmed and CSP implemented (Marco Ermini)
         The vendor does not acknowledge the second reported vulnerability
         in the specific context of use but has planned future improvement
         the application of the application replacing the GET request with
         a POST request.
20220701 USH: The team confirms prompt and effective remediation of the
         XSS vulnerability but points out suboptimal CSP implementation.
         The implementation seems to involve a central proxy or device and
         to always include a list of 10 vendor clients and other third
         parties CDN probably used for other reasons different from the
         audited integrity line app (e.g. bootstrap CDN). The team advises
         to implement a policy per-site and app to avoid listing sensible
         resources and limit any possible exposure.
20220701 Advisory release scheduled for 20220706
20220706 Advisory released

VII. REFERENCES

[1] EQS Integrity Line: Multiple Vulnerabilities
    http://www.ush.it/team/ush/advisory-eqs-integrity-line/eqs_integrity_line.txt

VIII. CREDIT

Giovanni Pellerano, is credited with the discovery of this vulnerability.

Giovanni Pellerano
web site: http://www.ush.it/
mail: evilaliv3@ush.it

IX. LEGAL NOTICES

Copyright (c) 2022 Giovanni Pellerano

Permission is granted for the redistribution of this alert
electronically. It may not be edited in any way without mine express
written consent. If you wish to reprint the whole or any
part of this alert in any other medium other than electronically,
please email me for permission.

Disclaimer: The information in the advisory is believed to be accurate
at the time of publishing based on currently available information. Use
of the information constitutes acceptance for use in an AS IS condition.
There are no warranties with regard to this information. Neither the
author nor the publisher accepts any liability for any direct, indirect,
or consequential loss or damage arising from use of, or reliance on,
this information.
```


Intrusion Detection (866)
Java (2,888)
JavaScript (817)
Kernel (6,255)
Local (14,173)
Magazine (586)
Overflow (12,390)
Perl (1,417)
PHP (5,087)
Proof of Concept (2,290)
Protocol (3,426)
Python (1,449)
Remote (30,009)
Root (3,496)
Ruby (594)
Scanner (1,631)
Security Tool (7,768)
Shell (3,098)
Shellcode (1,204)
Sniffer (885)
Spoof (2,165)
SQL Injection (16,089)
TCP (2,377)
Trojan (685)
UDP (875)
Virus (661)
Vulnerability (31,104)
Web (9,329)
Whitepaper (3,728)
x86 (946)
XSS (17,478)
Other

BSD (370)
CentOS (55)
Cisco (1,917)
Debian (6,620)
Fedora (1,690)
FreeBSD (1,242)
Gentoo (4,272)
HPUX (878)
iOS (330)
iPhone (108)
IRIX (220)
Juniper (67)
Linux (44,118)
Mac OS X (684)
Mandriva (3,105)
NetBSD (255)
OpenBSD (479)
RedHat (12,339)
Slackware (941)
Solaris (1,607)
SUSE (1,444)
Ubuntu (8,147)
UNIX (9,150)
UnixWare (185)
Windows (6,504)
Other

Site Links

News by Month
News Tags
Files by Month
File Tags
File Directory

About Us

History & Purpose
Contact Information
Terms of Service
Privacy Statement
Copyright Information

Hosting By

Rokasec

Follow us on Twitter

Subscribe to an RSS Feed