


SECURETHY: ATTACK SCENARIOS & AUDIT GUIDES

SEARCH 

Educational Purpose - Case study, attacks' scenarios and audit guidelines on vulnerabilities AI Powered

Home

Live CTF[New]

Bug Bounty

CVE-2018-12652: Reflected Cross Site Scripting(XSS) in Adrenalin 5.4 HRMS Software | LeaveEmployeeSearch [issue 4 of 5]

February 17, 2019

As cyber security professional, I come across many various vulnerabilities from critical

SHARE

to low and sometimes informational(How to categorize- CVSS v3). Sometime back I was doing as usual my security assessment activity for a Bank(Confidential) for their HRMS web application which was 3rd party software whose vendor was "Adrenalin".

CVE ID: CVE-2018-12652

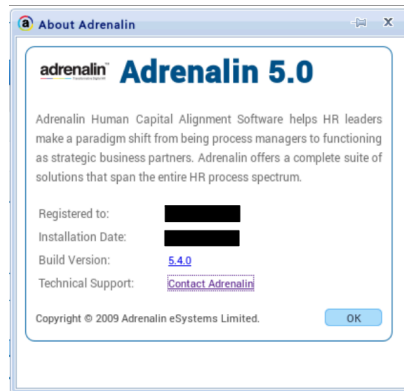
Vulnerability Name:

Reflected Cross Site Scripting(XSS)

Product: Adrenalin HRMS

Affected Version: 5.4.0

Credits: **Rishu Ranjan**



CVE-2018-12652 Details

The Common
Vulnerabilities and
Exposures (CVE) project
has assigned the ID CVE-
2018-12652 to this issue
provided by MITRE
Corporation (MITRE)(As
Vendor is not CVE
Numbering Authorities
(CNAs))

CVSS Score

CVSS Base Score: 6.1

Vector:

AV:N/AC:L/PR:N/UI:R/S:

C/C:L/I:L/A:N

Impact Subscore:2.7

Exploitability Subscore: 2.8

Current Description

A Reflected Cross Site Scripting(XSS) Vulnerability was discovered in Adrenalin 5.4 HRMS which is publically available. The user supplied input containing special characters such as <, >, /, etc. is echoed back in javascript code in HTML response without any output encoding performed. This allows an attacker to input malicious JavaScript which can steal cookie, redirect them to other malicious website, etc.

CVE-ID	CVE-2018-12652
Description	Request Method(s): [+] GET Vulnerable Product: [+] Adrenalin HRMS

	Software 5.4.0
URL	<code>/LeaveEmployeeSearch.aspx? popToken=emp&prntFrmName=frmLeaveReq uest&prntDDLContrlName=hdnNominee23578 %22%3balert(1)%2f%2f213&HRShow=0&Contrl Type=txt</code>
Parameter	<code>prntDDLContrlName prntFrmName</code>

Impact

In Reflected Cross Site Scripting, the malicious payload has to be send as a part of URL and user should be tricked to visit that URL. However, it has the same impact as that of a persistent XSS. XSS can be used to hijack victim's session and thereby gaining complete access to his/her user account. Additionally, it can be used to redirect

victim to a malicious website which may contain browser exploits or a phishing page.

Remediation, Solution

At a basic level XSS works by tricking your application into inserting a `<script>` tag into your rendered page, or by inserting an `On*` event into an element.

Developers should use the following prevention steps to avoid introducing XSS into their application.

1. Never put untrusted data into your HTML input, unless you follow the rest of the steps below.

Untrusted data is any data that may be controlled by an attacker, HTML form

inputs, query strings, HTTP headers, even data sourced from a database as an attacker may be able to breach your database even if they cannot breach your application.

2. Before putting untrusted data inside an HTML element ensure it's HTML encoded. HTML encoding takes characters such as < and changes them into a safe form like <

3. Before putting untrusted data into an HTML attribute ensure it's HTML attribute encoded. HTML attribute encoding is a superset of HTML

encoding and
encodes additional
characters such as "
and '.

4. Before putting
untrusted data into
JavaScript place the
data in an HTML
element whose
contents you retrieve
at runtime. If this
isn't possible then
ensure the data is
JavaScript encoded.
JavaScript encoding
takes dangerous
characters for
JavaScript and
replaces them with
their hex, for
example < would be
encoded as \u003C.

5. Before putting
untrusted data into a
URL query string
ensure it's URL
encoded.

Reference

<https://cwe.mitre.org/data/definitions/79.html>

[https://www.owasp.org/index.php/Testing_for_Reflected_Cross_site_scripting_\(OTG-INPVAL-001\)](https://www.owasp.org/index.php/Testing_for_Reflected_Cross_site_scripting_(OTG-INPVAL-001))

<https://docs.microsoft.com/en-us/aspnet/core/security/cross-site-scripting?view=aspnetcore-2.1>

<https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2018-12652>

Update Timelines

22th June,2018- Reflected XSS Vulnerability found in the software.

22th June,2018- Few query asked about the CVE process with same

information reported to
Mitre without Vendor
name and version
information.

22th June,2018- Mitre
replied with the CVE-ID
and asked to inform the
vendor about the
vulnerability and CVE-ID is
generated.

28th June, 2018- The
vulnerability report
containing all the details
was submitted to the
Vendor.

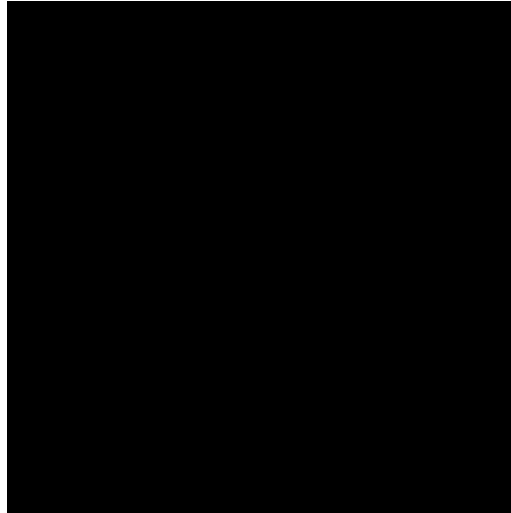
28th June,2018- The
Vendor Replied and the full
report of the vulnerability
sent to the Vendor.

2nd July,2018- The XSS
Instance was patched on
cloud version of the
software.

SHARE

Security blog of the month

CVE-2018-12651: Reflected Cross Site Scripting(XSS) in Adrenalin 5.4 HRMS Software | ShiftEmployeeSearch [issue 3 of 5]



December 02, 2018

As cyber security professional, I come across ...

SHARE

READ MORE



About Me



Web Application Vulnerability Assessment using Burp Community

Part 3 covers File Upload Bypass via Linux Filename limit

-

Part 2 covers 1 - Account Takeover via Forgot Password—A Practical Attack Scenario of Host Header Injection 2 - IP Spoofing (Bypass Whitelisting)

-

Part 1 covers 1 - HTTP response header injection 2 - Server-side request forgery (SSRF) - Out-of-band resource load (HTTP) 3 - HTTP PUT method is enabled

-

[Privacy Policy](#)
[Sitemap](#)

© 2025 - 2026 **Securethy.com**.
All Rights Reserved.