Malicious OPEN Redirection *CVE-2025–61166*

 Nikhil Thakur Follow 2 min read · Feb 11, 2026

SigningHub by Ascertia v.8.6.8 — Malicious OPEN Redirection

> Discovered by: Nikhil Thakur | Yazan Abu-Nadi

```
################################################################################
# Title: SigningHub by Ascertia v.8.6.8 - Malicious OPEN Redirection
# Author: Mr. Nikhil Thakur | Yazan Abu-Nadi
# Vendor Homepage: https://www.signinghub.com/
# Version: v.8.6.8
# Tested on: Latest version of Chrome, Firefox on Windows and Linux.
# CVE: CVE-2025-61166
################################################################################
```

— — — — — — — — — — — — — — — OPEN Redirection — — — — — — — — — — — — — —

> Open redirection arises when a script writes controllable data into the target of a redirection in an unsafe way. An attacker may be able to use the

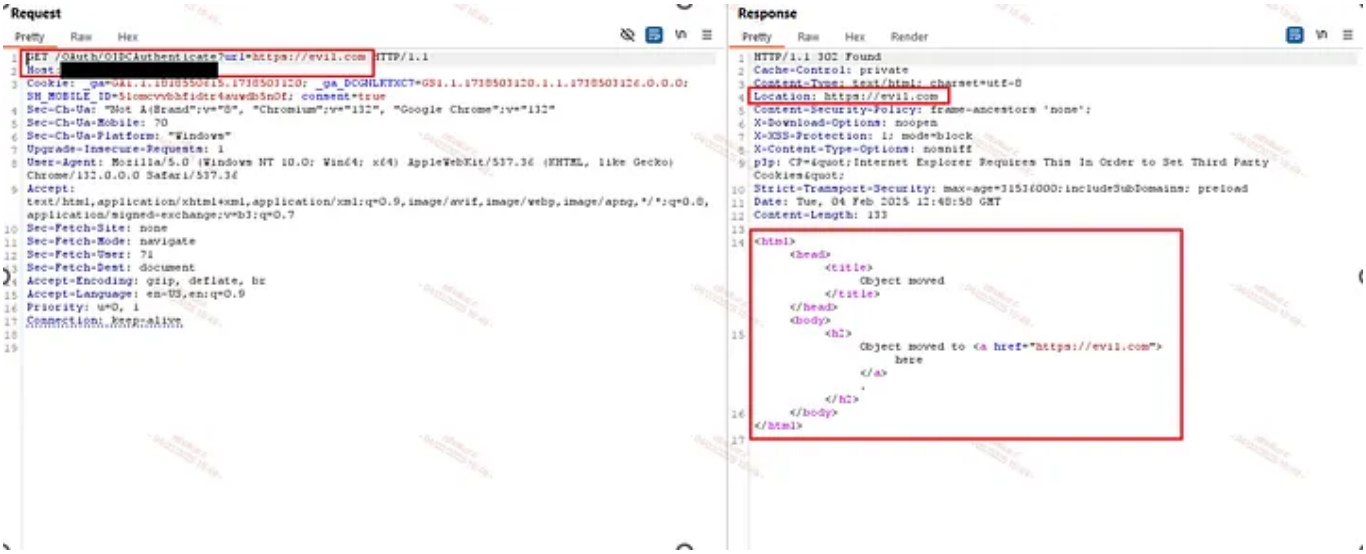
> vulnerability to construct a URL that, if visited by another application user, will cause a redirection to an arbitrary external domain. This behavior can be leveraged to facilitate phishing attacks against users of the application. The ability to use an authentic application URL, targeting the correct domain and with a valid SSL certificate (if SSL is used), lends credibility to the phishing attack because many users, even if they verify these features, will not notice the subsequent redirection to a different domain. If an attacker can control the start of the string that is passed to the redirection API, then it may be possible to escalate this vulnerability into a JavaScript injection attack, by using a URL with the javascript: pseudo-protocol to execute arbitrary script code when the URL is processed by the browser.

— —

Reproduction Steps:

> *Step 1: Initial Discovery and Reconnaissance:*
> *As an unauthenticated user, navigate to the target application's base URL (e.g., https://*.signinghub.com). Using browser developer tools (F12 → Network tab) or an intercepting proxy like Burp Suite, spider/crawl the application to identify all endpoints. Look specifically for parameters named url, redirect, return, next, or destination. You should identify the endpoint: /OAuth/OIDCAuthenticate?url=*
>
> *Step 2: Craft the Malicious Request:*
> *Construct a GET request to the vulnerable endpoint with an external domain in the url parameter:*
> *https://*.signinghub.com/OAuth/OIDCAuthenticate?url=https://evil.com*
> *(Replace url= value with the actual subdomain with a domain you control or a trusted external site like https://evil.com for testing)*

> *Step 3: Execute and Observe:*
> *Submit the crafted URL directly in a browser address bar or via a crafted link. Observe if the application immediately redirects the browser to https://evil.com without any user warning, interstitial page, or validation prompt.*
>
> *Step 4: Traffic will be redirected to https://evil.com ;)*



Successfully Redirected to https://evil.com

****An unauthenticated attacker can able to find this vulnerable “url=” parameter endpoint by just spidering the SigningHub application portal url to identify associated endpoints.****

— — — Many Thanks — — —

Written by Nikhil Thakur

3 followers · 1 following

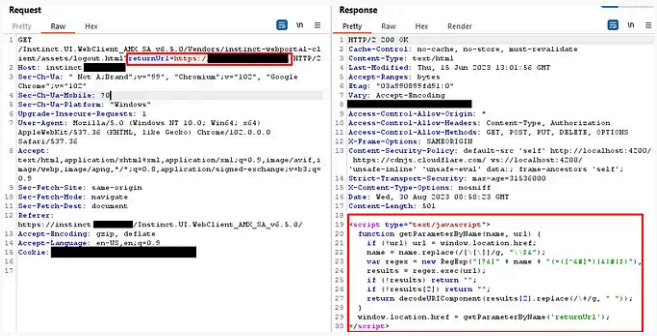
Follow

No responses yet

Write a response

What are your thoughts?

More from Nikhil Thakur



Nikhil Thakur

Nikhil Thakur

DOM Based Malicious Redirection *CVE-2024–28287*

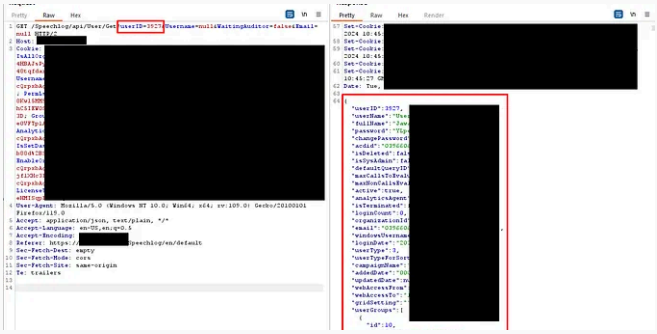
Instinct Web UI v.6.5.0 — DOM-Based Malicious Redirection

Mar 22, 2024 150

Stored XSS in Chat Box Component *CVE-2025–56320*

CobbleStone Software — Enterprise Contract Management Software v.22.2.1

Oct 14, 2025

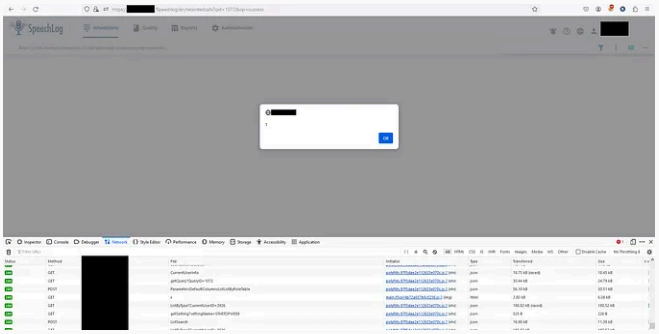


Nikhil Thakur

Insecure Direct Object References *CVE-2024–33818*

SpeechLog v.8.1 — Privilege Escalation

May 10, 2024 50



Nikhil Thakur

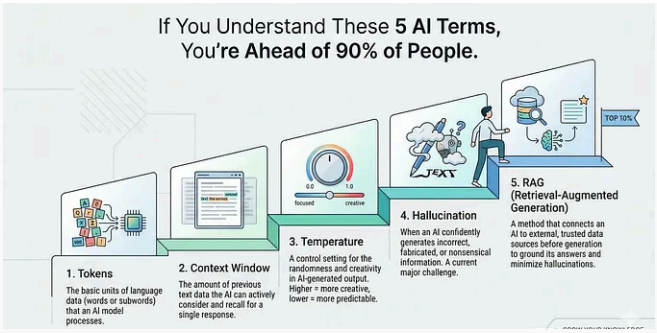
Stored Cross Site Scripting *CVE-2024–33819*

SpeechLog v.8.1 — Stored XSS

May 10, 2024 50

See all from Nikhil Thakur

Recommended from Medium

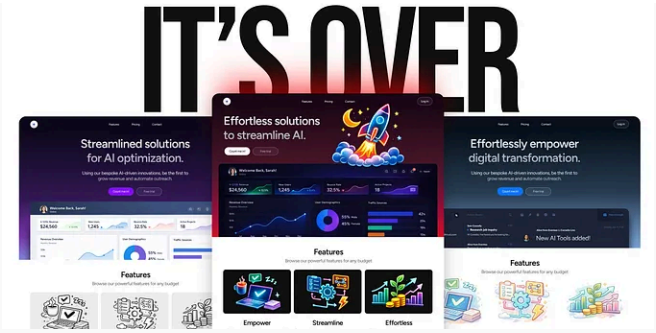


In Towards AI by Shreyas Naphad

If You Understand These 5 AI Terms, You're Ahead of 90% of...

Master the core ideas behind AI without getting lost

Mar 29 7.8K 157

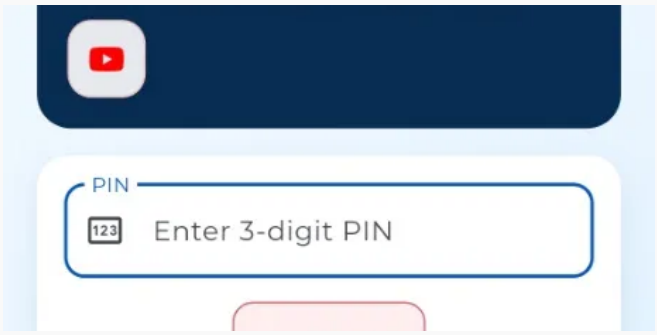


Michal Malewicz

Vibe Coding is OVER.

Here's What Comes Next.

Mar 24 4.8K 173



Bejiamen

Mobile CTF: Cracking an Android App PIN with ADB Brute-Forcing

Introduction

Mar 30

In OSINT Team by Vivek PS

How a $62,500 Self-XSS Became a Full Facebook and Instagram...

Note: This article is a review and narrative analysis of a bug bounty write-up published...

Mar 27 70

Krishna Kumar

💀 From SSRF to AWS Pwnage: A Hacker’s Guide to Cloud-Native…

💡 Ever found a bug that lets you make a server visit a URL? It might seem small, but…

Feb 27 6

Aya Ayman(GERR4Y)

One Mobile Number = Full Wishlist Takeover (No Authentication…

بسم الله والصلاة والسلام على رسول الله الحمد لله الذي علم بالقلم علم الإنسان ما لم يعلم والصلاة…

Apr 1 506

See more recommendations