

Reflected Cross Site Scripting (XSS) vulnerability was found in `"/ptms/normal-search.php"` in PHPGurukul Park Ticketing Management System v1.0 allows remote attackers to execute arbitrary code via `"searchdata"` POST request parameter.

Affected Project: PHPGurukul Park Ticketing Management System v1.0

Official Website: <https://phpgurukul.com/park-ticketing-management-system-using-php-and-mysql/Version: 1.0>

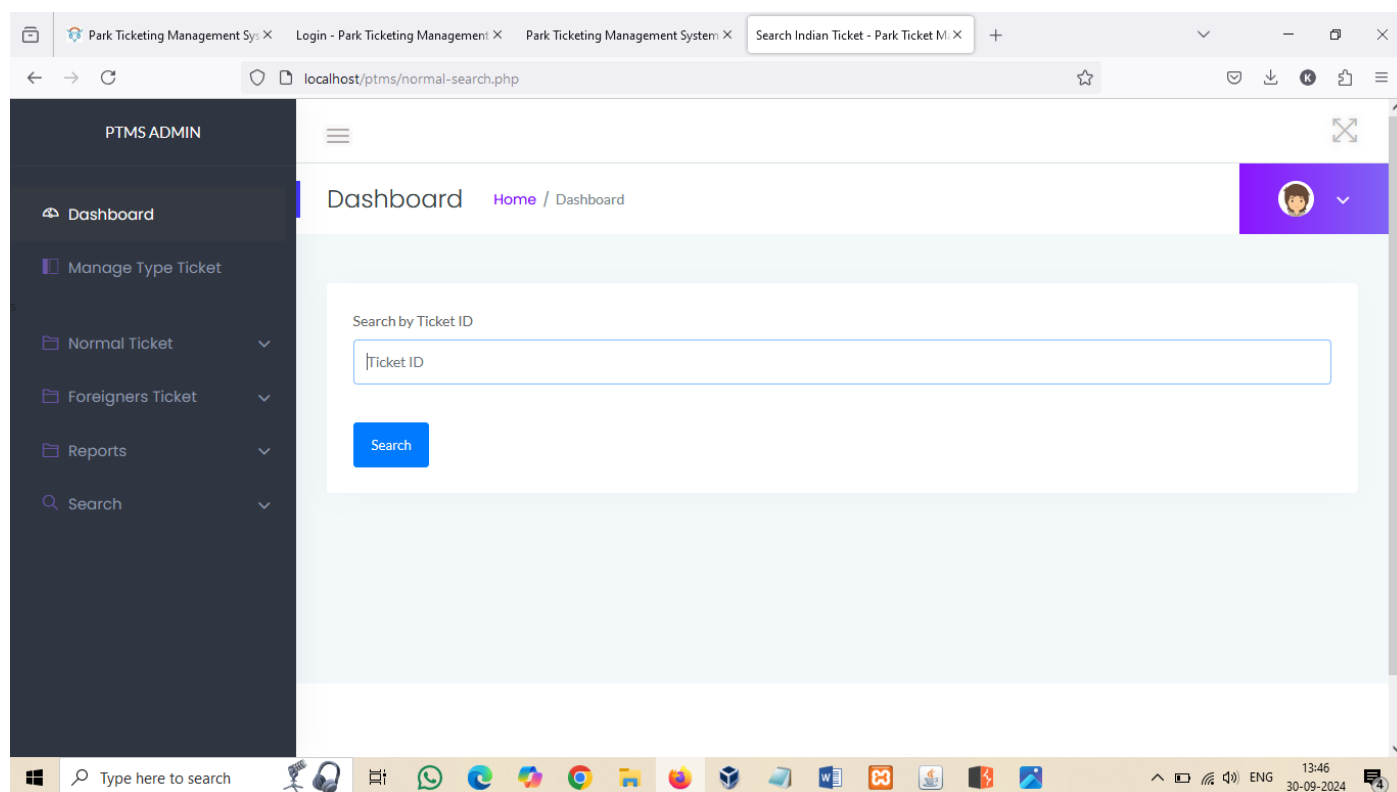
Affected Components:

Affected File: </ptms/normal-search.php>

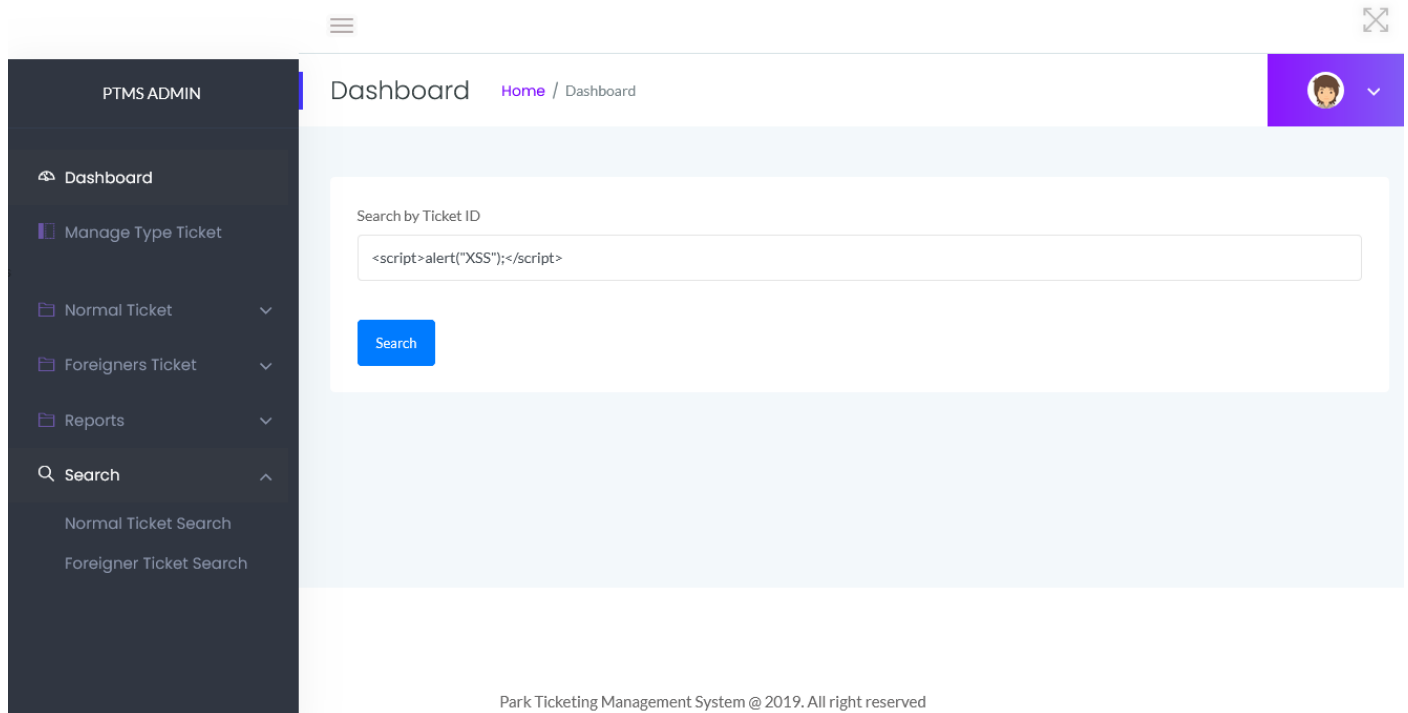
Affected Parameter: `"search"` URL parameter

Step:

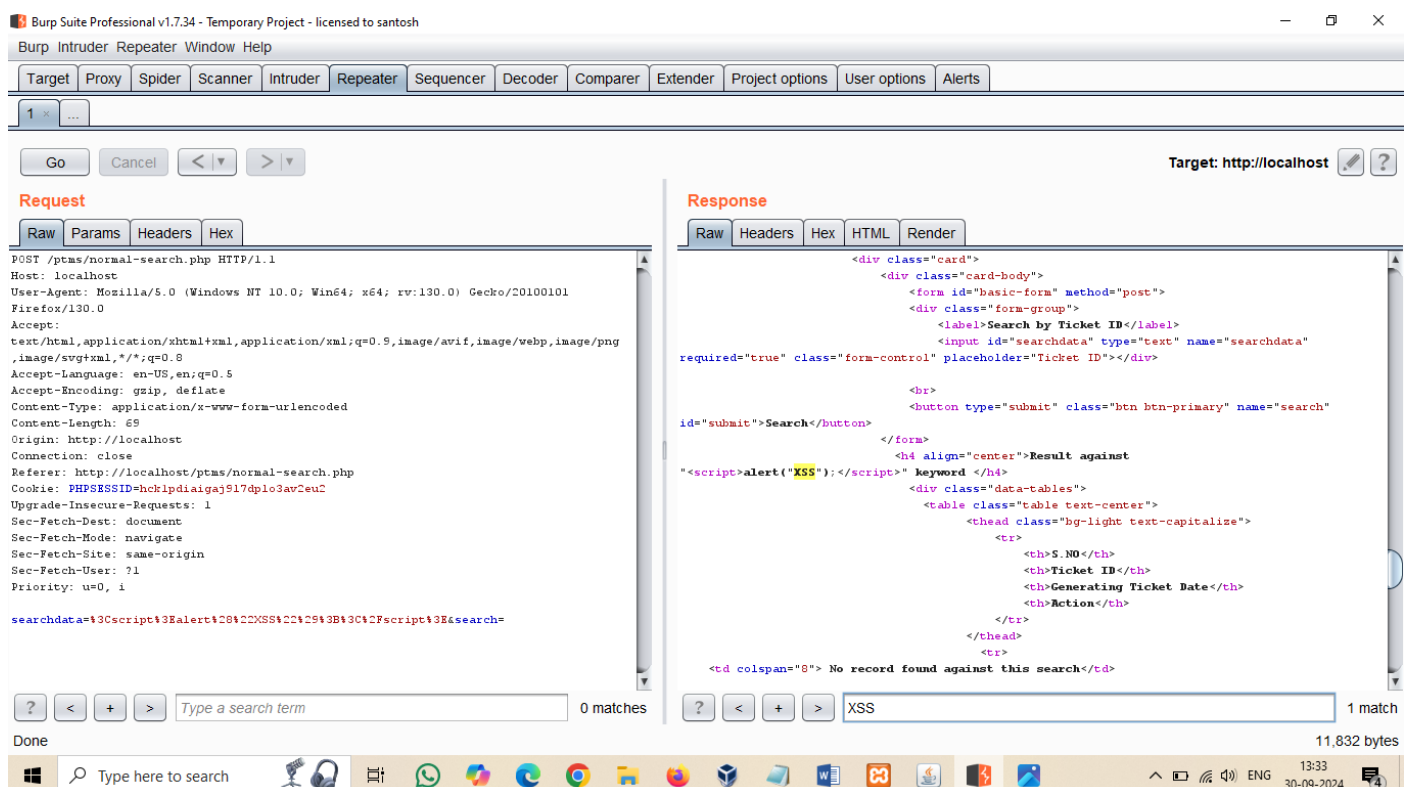
1. Access the Search Page URL: <http://localhost/ptms/normal-search.php>

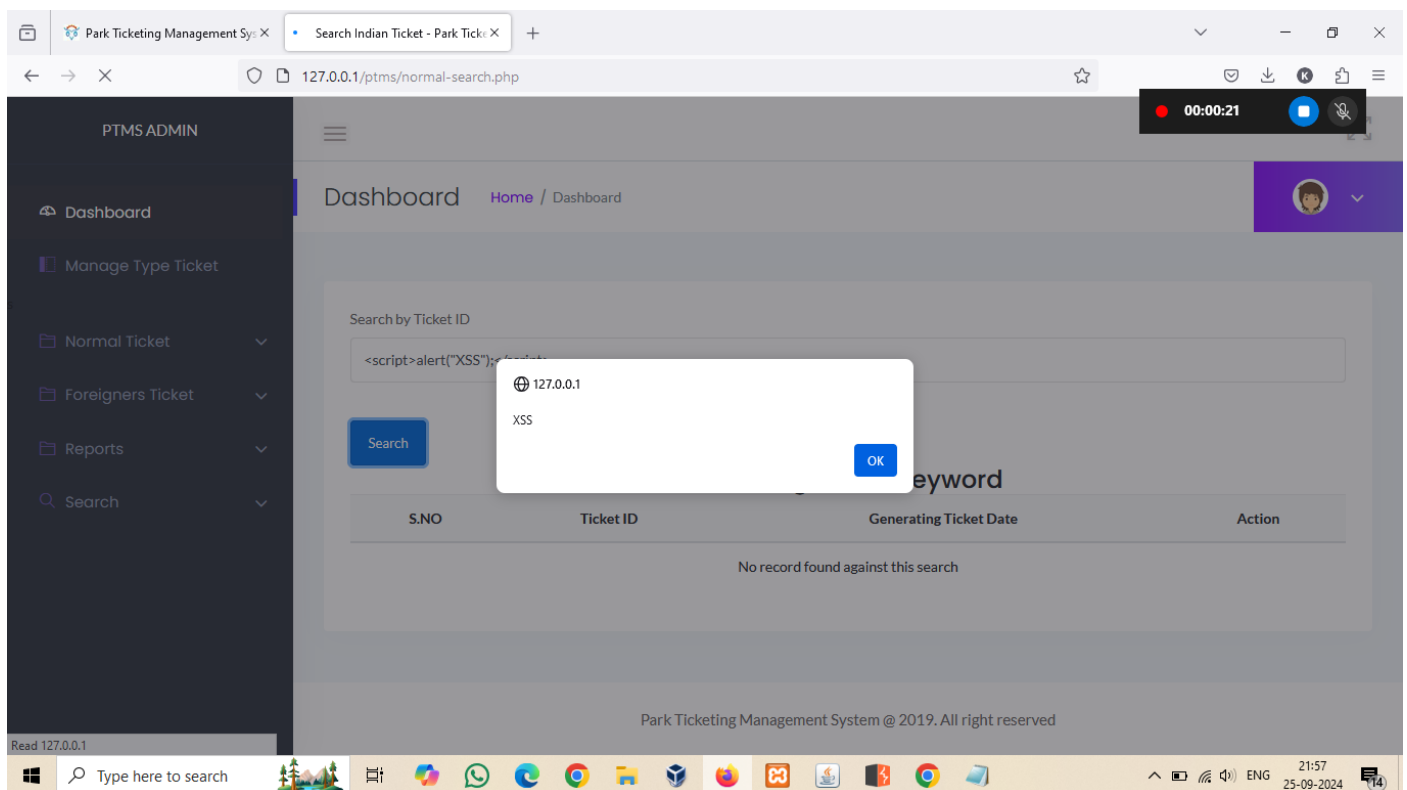


2. Enter the test payload with XSS `<script>alert("XSS");</script>` in the search box and click "Search" button



3. The XSS script is reflected back in the browser. The XSS script will get executed.





Solution/Good Reads:

Output Encoding -> When you need to safely display data exactly as a user types it in, output encoding is recommended.

[What is cross-site scripting \(XSS\) and how to prevent it? | Web Security Academy \(portswigger.net\)](https://portswigger.net/web-security/cross-site-scripting/what-is-cross-site-scripting)

[https://cheatsheetseries.owasp.org/cheatsheets/Cross Site Scripting Prevention Cheat Sheet.html](https://cheatsheetseries.owasp.org/cheatsheets/Cross_Site_Scripting_Prevention_Cheat_Sheet.html)