Bug Bounty Report: Shopify Admin Panel UI Breakdown via Search Query Injection

Report Summary

- Vulnerability Type: Business Logic Error / Injection
- Target Asset: `admin.shopify.com`
- Impact: UI Breakdown, Potential Denial of Service
- Reporter: hajajweu
- Date: 09-09-2025

1. Overview

- Entering a malformed search query (1=1-- or ' OR 1=1--) in the Shopify admin panel search bar triggers a UI rendering failure. Instead of valid search results, the admin panel shows incorrect, generic dashboard elements. This suggests improper handling of user input and a potential injection flaw in the backend query parser.

2. Steps to Reproduce

Method 1: Direct Browser Input (Easiest)

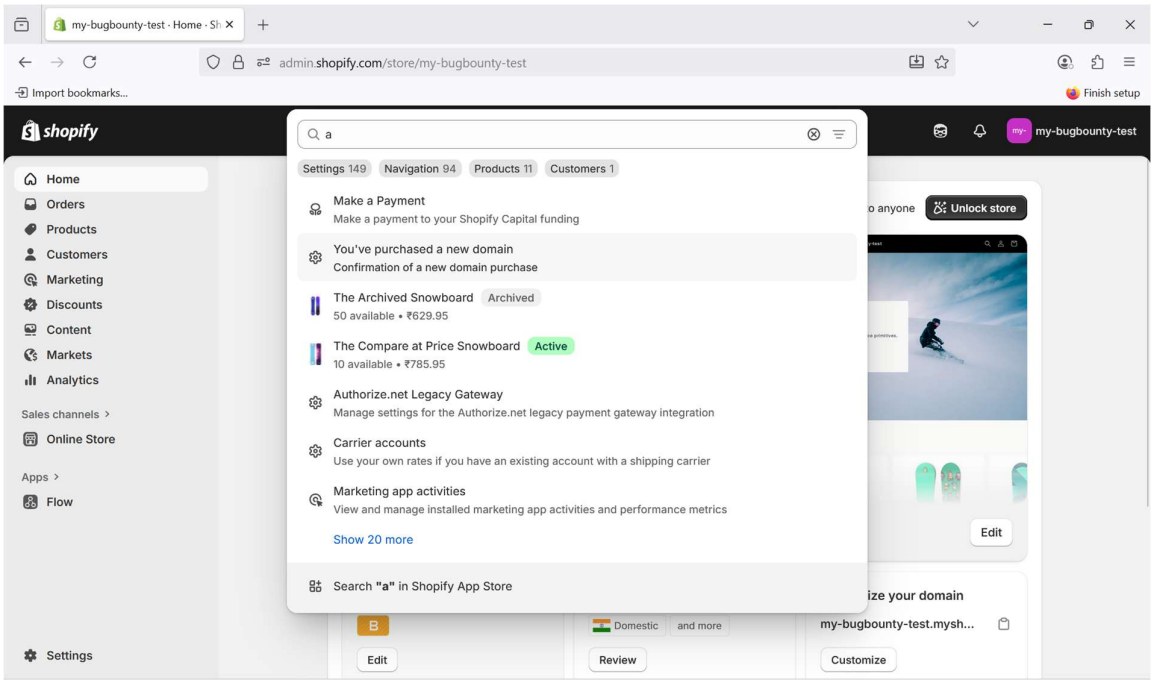
1. Log in to the Shopify admin panel of a development store (e.g., `https://my-bugbounty-test.myshopify.com/admin`).
2. Locate the global search bar at the top of the page.
3. Type the following payload exactly: `1=1--`
4. Press Enter to execute the search.
5. Observed Result: The admin interface fails to display valid search results and instead renders unexpected generic modules. Instead of showing relevant search results, it displays an unexpected rendering state with generic dashboard elements and module titles (e.g., "Navigation 3", "Products 3", "Draft orders 2", "Shopify Support", etc.). The actual product listings are also shown out of context. The expected search functionality is completely disrupted.

Method 2: HTTP Request Manipulation (Confirms Backend Issue)

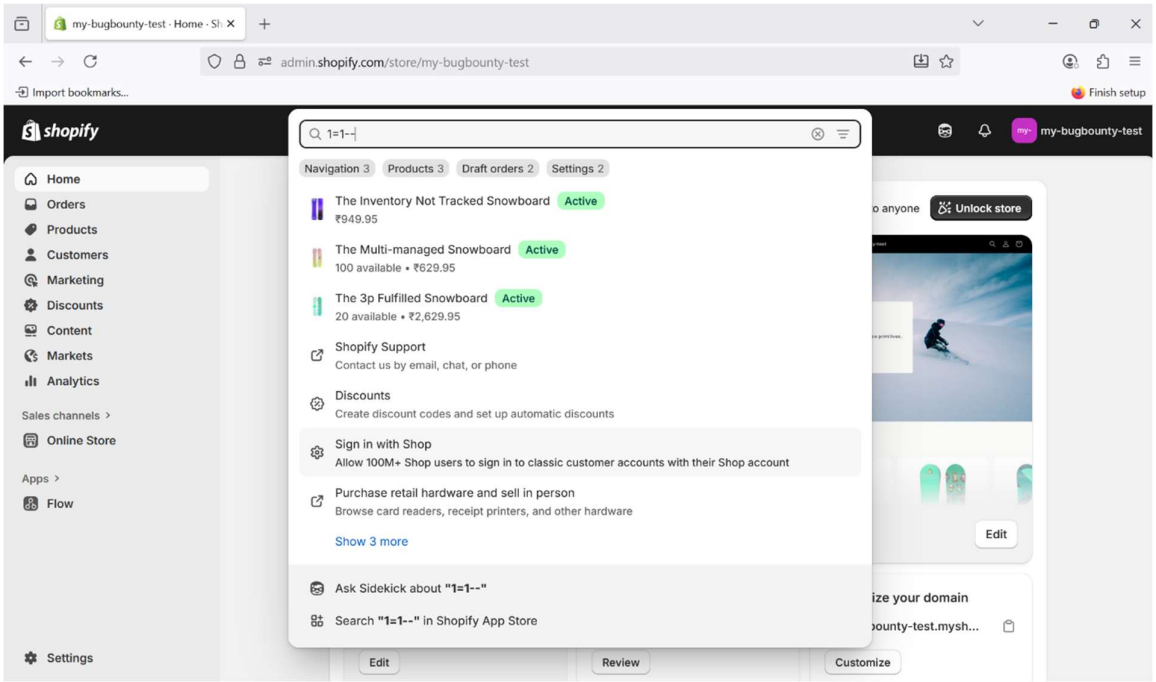
1. Configure a proxy tool (Burp Suite) to intercept traffic from the browser.
2. In the admin search bar, type a benign term like `a` and press Enter.
3. The HTTP request will be intercepted in the proxy. It is a POST request to `https://admin.shopify.com/api/shopify/[store-name]?operation=Search`.
4. The request consists of a JSON body with a GraphQL query. Find the parameter "query":"a".
5. Modify the parameter to `'"query":"' OR 1=1--"'`.
6. Forward the modified request to the server.
7. Observed Result: The same UI breakdown occurs, confirming the issue is server-side and triggered by the payload in the HTTP request.

3. Proof of Concept Screenshots

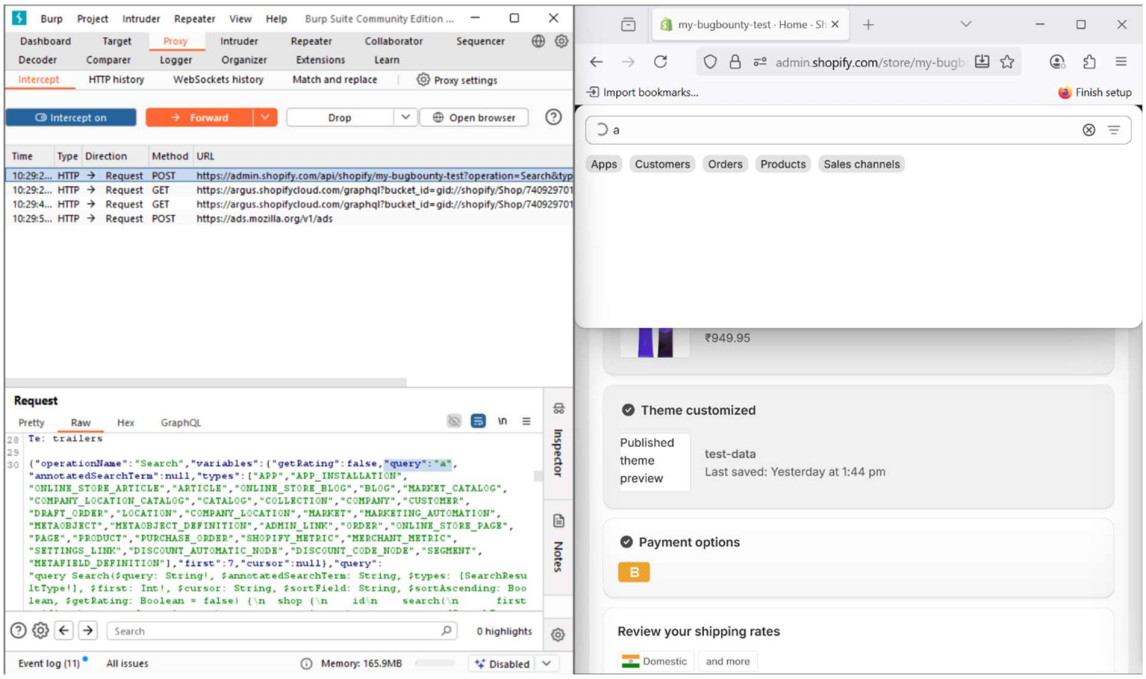
Screenshot 1: *Normal admin UI before payload execution*

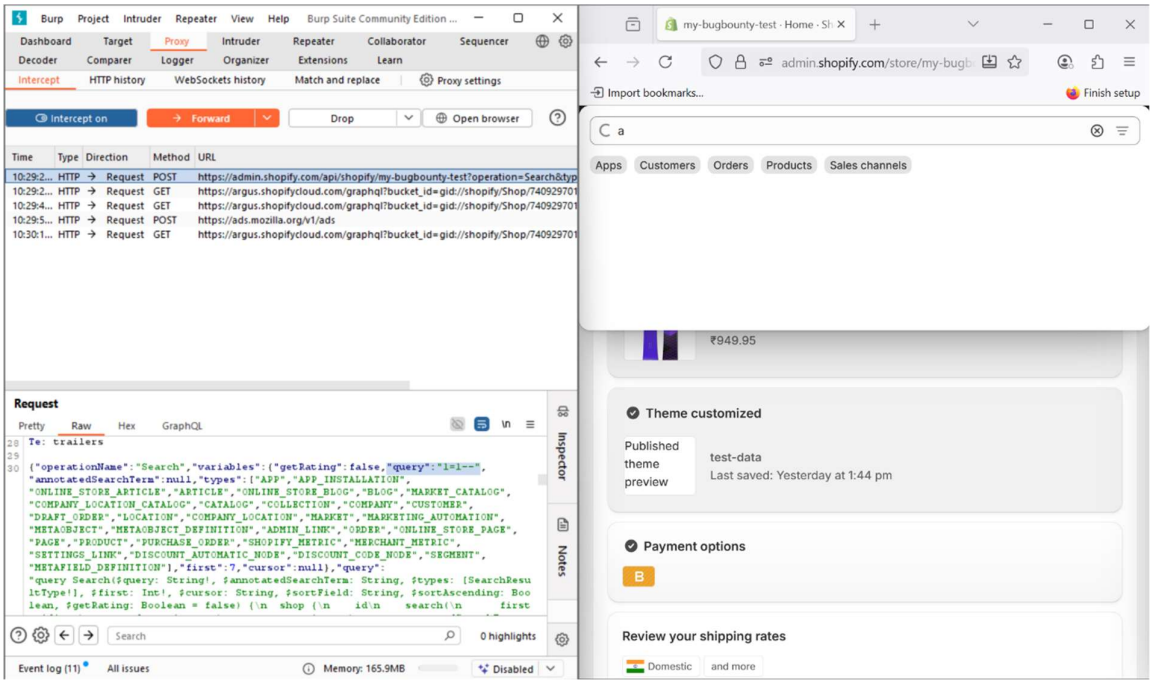


Screenshot 2: *Broken UI after entering payload 1=1--*

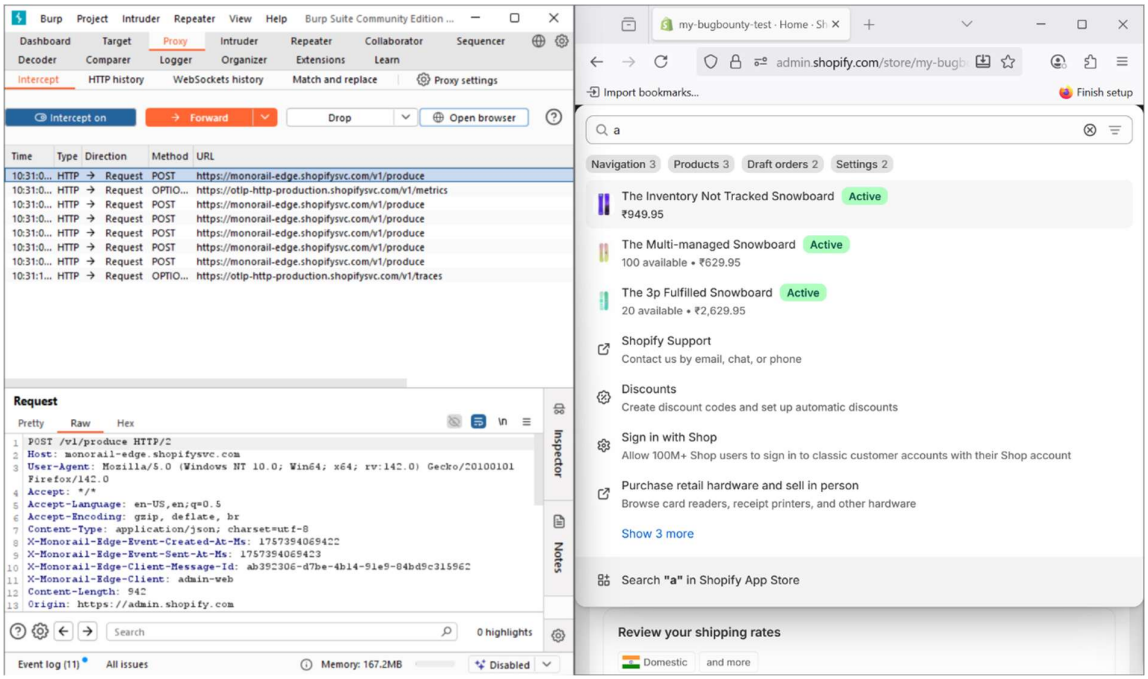


Screenshot 3: *Burp Suite request modification with injection payload*





Screenshot 4: *Broken UI after sending modified request*



4. Impact Analysis

- This vulnerability disrupts the normal functionality of the Shopify admin panel's search feature, resulting in a Denial of Service for administrators using search. Since the root cause is likely an injection flaw in the query parser, this could be escalated to more severe issues if exploited with advanced payloads. The issue is directly exploitable from a browser without requiring special tools.

5. Environment

- **Testing Store:** `my-bugbounty-test.myshopify.com` (A development store created via Shopify Partners for the purpose of bug bounty testing)
- **Browser:** Firefox with Burp Suite proxy
- **Testing Account**: hajajweu456@gmail.com

6. Recommendation

- Sanitize and validate all search inputs on both frontend and backend.
- Implement parameterized queries for GraphQL search requests.
- Conduct a security review of query parsing logic to prevent unsafe string concatenation.

Submitted in accordance with Shopify's Bug Bounty Program terms and conditions.