Authentication bypass via information disclosure

Executive Summary

This report outlines the discovery of an authentication bypass vulnerability within the admin interface of the target web application. The vulnerability stems from the disclosure of a custom HTTP header that governs access control. By identifying and manipulating this header, an attacker can gain unauthorized administrative access and perform privileged actions. The severity of this vulnerability is **high**, as it undermines the security of the admin panel.

Introduction

The purpose of this assessment was to evaluate the effectiveness of access controls protecting administrative functionality within the application. A focus was placed on identifying insecure authentication mechanisms and sensitive information disclosures that may aid an attacker in bypassing security controls.

Methodology

1. Used **Burp Suite** to intercept and replay HTTP requests.
2. Examined responses for signs of access control logic.
3. Sent a **TRACE** request to identify reflected HTTP headers.
4. Discovered a custom header used to validate local requests.
5. Used Burp Suite's **Match and Replace** feature to manipulate the custom header.
6. Verified the ability to access the admin interface and delete a user without proper authentication.

Vulnerability Findings

- **Type**: Authentication Bypass via Information Disclosure
- **Location**: Admin interface access control logic
- **Severity**: High

Description

The application restricts access to the admin interface based on user role or originating IP. However, the **TRACE** method discloses an internal custom HTTP header `X-Custom-IP-Authorization` that the server uses to determine if a request originated from localhost. By modifying this header value to `127.0.0.1`, an attacker can impersonate a local request and gain unauthorized access to protected admin functionality.

Proof of Concept (PoC):

1. Sent a 'GET /admin' request Access denied with message: "Only accessible to localhost or admin users."
2. Sent a `TRACE /admin` request Response revealed header: `X-Custom-IP-Authorization: [client-IP]`.

3. Configured **Burp Suite Match and Replace** to inject:
   ‘Makefile’
   ‘CopyEdit’
   ‘`X-Custom-IP-Authorization: 127.0.0.1`’
4. Reloaded the home page with modified headers.
5. Admin interface became accessible.
6. Deleted selected user successfully.

Impact Assessment:

This vulnerability allowed **unauthenticated attackers** to gain full administrative control over the application by spoofing internal traffic. This can lead to:
- Deletion or modification of user accounts.
- Unauthorized access to sensitive data.
- Disruption of critical system functionality.

Recommendations:

1. **Disable TRACE method** on all web servers to prevent header disclosure.
2. Implement **strict server-side validation** that does not rely on spoofable headers for authentication or trust evaluation.
3. Monitor and log access to admin endpoints, particularly suspicious header values.
4. Use secure, token-based authentication mechanisms instead of trust-by-IP methods.

Conclusion: The authentication bypass vulnerability highlighted the risks of relying on insecure headers and exposing server behaviour through HTTP methods like TRACE. Immediate remediation steps should be taken to harden access control logic and prevent unauthorized privilege escalation.