Unprotected admin functionality

**1. Summary**

**Bug Title:**  Unprotected admin functionality logged in robots.txt file whicn any unuthenticated have full access to it

**Severity**: Critical.

**Description**: Access control enforces policy such that users cannot act outside of their intended permissions. Failures typically lead to unauthorized information disclosure, modification, or destruction of all data or performing a business function outside the user's limits..

**Date Discovered:** 1/8/2024.

**Status:**  Solved.

**2. Bug Details**

**Vulnerability Type:** Broken Access Control.

**Affected URL/Endpoint:**  /administrator-panel.

**Description:** A critical vulnerability has been identified in the web application where an attacker can access unprotected admin panel with full control . This vulnerability highlights a severe case of broken access control, allowing unauthorized/unauthenticated users to gain elevated privileges and potentially compromise the entire system.

**Steps to Reproduce:**

1. Go to lab URL: <https://portswigger.net/web-security/access-control/lab-unprotected-admin-functionality> and access the lab

2. Go to robots.txt file via this URL: <https://0ab60001038f870a823eecf20094004d.web-security-academy.net/robots.txt>



3. you will find an admin panal into robots.txt file /administrator-panel

4. access this admin functionality via this URL: <https://0ab60001038f870a823eecf20094004d.web-security-academy.net//administrator-panel>

5. delete user carlos to solve the lab

**Proof of Concept (PoC):** lab solved



**Impact:** The application improperly expose unprotected admin functionality. By accessing it , an attacker can gain unauthorized access to administrative functionalities, including the ability to delete user accounts..

**3. Recommendations**

* Role-Based Access Control (RBAC): Ensure that access to administrative functions is based on clearly defined roles and permissions. Only authenticated and authorized users should have administrative privileges
* erver-Side Authorization Checks: Always perform authorization checks on the server side to verify user permissions before granting access to sensitive operations.
* protect admin functinality via stricted authentication scheme

**4. Conclusion**

**Summary:** A critical vulnerability has been identified in the web application where an attacker can escalate privileges to an administrative level by access un protected admin functionality. This allows unauthorized users to access admin functionalities and delete user accounts, highlighting a severe case of broken access control.

**5. Appendices**

**Tools Used:** Burp Suite.

**References:**

* <https://owasp.org/Top10/A01_2021-Broken_Access_Control/>
* <https://portswigger.net/web-security/access-control>