Web Application Penetration Testing Report

Engagement Summary

This document represents a penetration testing report for a vulnerable web application provided as part of the picoCTF training platform. The engagement focused exclusively on assessing the security of the application's authentication mechanism.

The purpose of this assessment was to identify security weaknesses, validate exploitability, and provide actionable remediation guidance. This report is written in a professional pentesting format rather than a challenge walkthrough.

1. Engagement Details

Target: picoCTF – ‘Login’ Web Application

Assessment Type: Web Application Penetration Test (Authentication)

Methodology: Black-box testing

Category: Authentication and Access Control

Risk Level: High

2. Scope

In Scope

Login functionality

Authentication logic

Access control enforcement

Out of Scope

Denial-of-service testing

Infrastructure-level attacks

Automated scanning beyond manual verification

3. Testing Methodology

The assessment followed a controlled manual testing approach aligned with common industry practices:

Application reconnaissance

Request and response analysis

Authentication testing

Logic flaw validation

Impact confirmation

4. Executive Risk Summary

A critical authentication flaw was identified that allows unauthenticated users to bypass login controls and gain access to restricted application functionality. Successful exploitation requires no valid credentials and minimal technical effort.

If exploited in a production environment, this vulnerability could lead to complete compromise of protected resources.

5. Vulnerability Details

5.1 Authentication Bypass

Severity: High

OWASP Mapping: A01: Broken Access Control / A07: Identification and Authentication Failures

Description

The application fails to properly enforce server-side authentication controls. Authentication decisions rely on insecure logic that can be manipulated by an attacker, allowing unauthorized access without valid credentials.

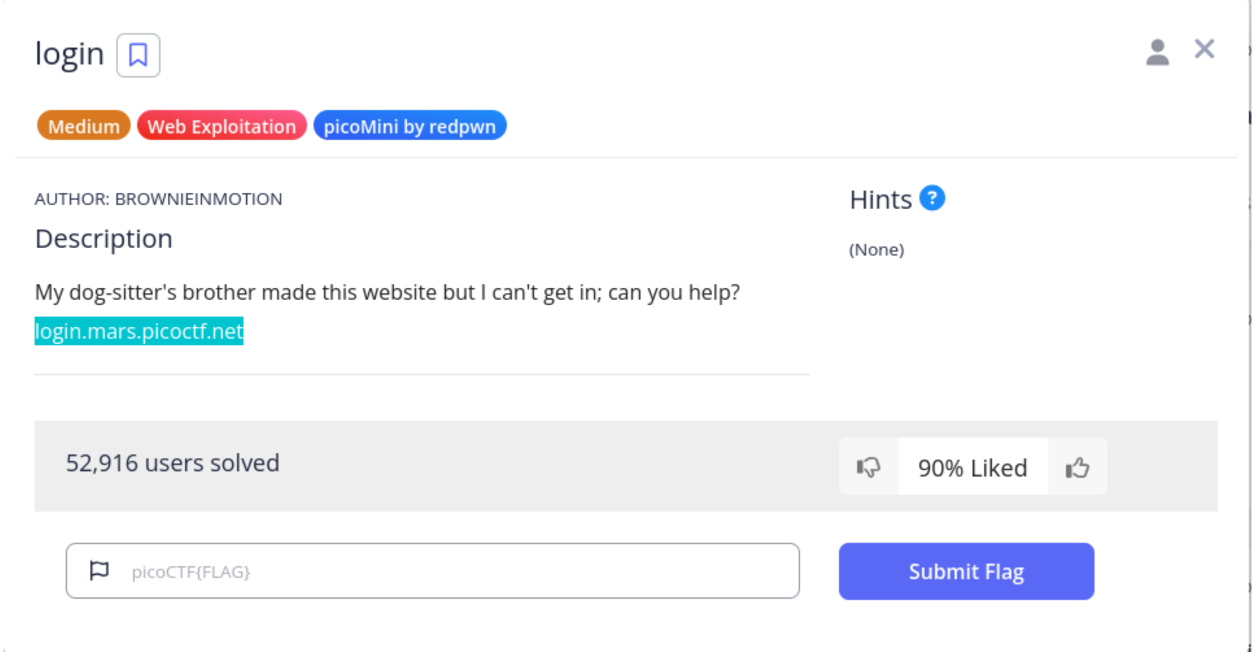
Evidence

Successful login without valid username or password

Access to restricted content following bypass

6. Proof of Exploitation

During testing, authentication controls were bypassed by manipulating request logic. Upon successful bypass, the application disclosed sensitive protected information, including the picoCTF flag.



AUTHOR: BROWNIEINMOTION

Description

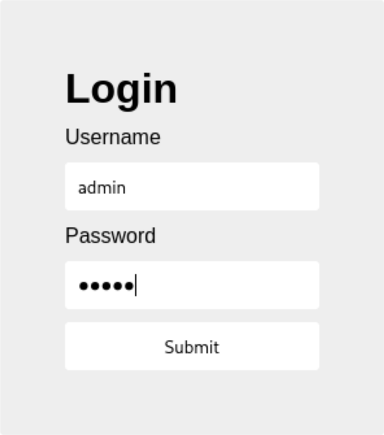
My dog-sitter's brother made this website but I can't get in; can you help?

login.mars.picoctf.net

(None)

52,916 users solved

90% Liked



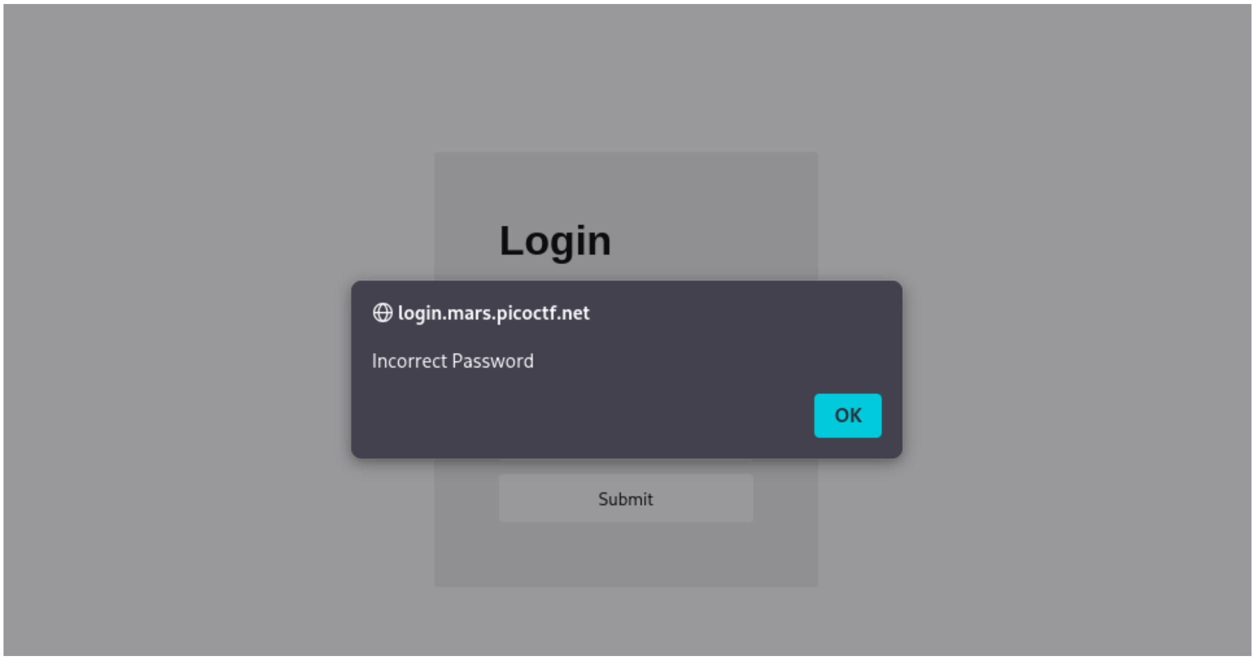
Login

Username

admin

Password

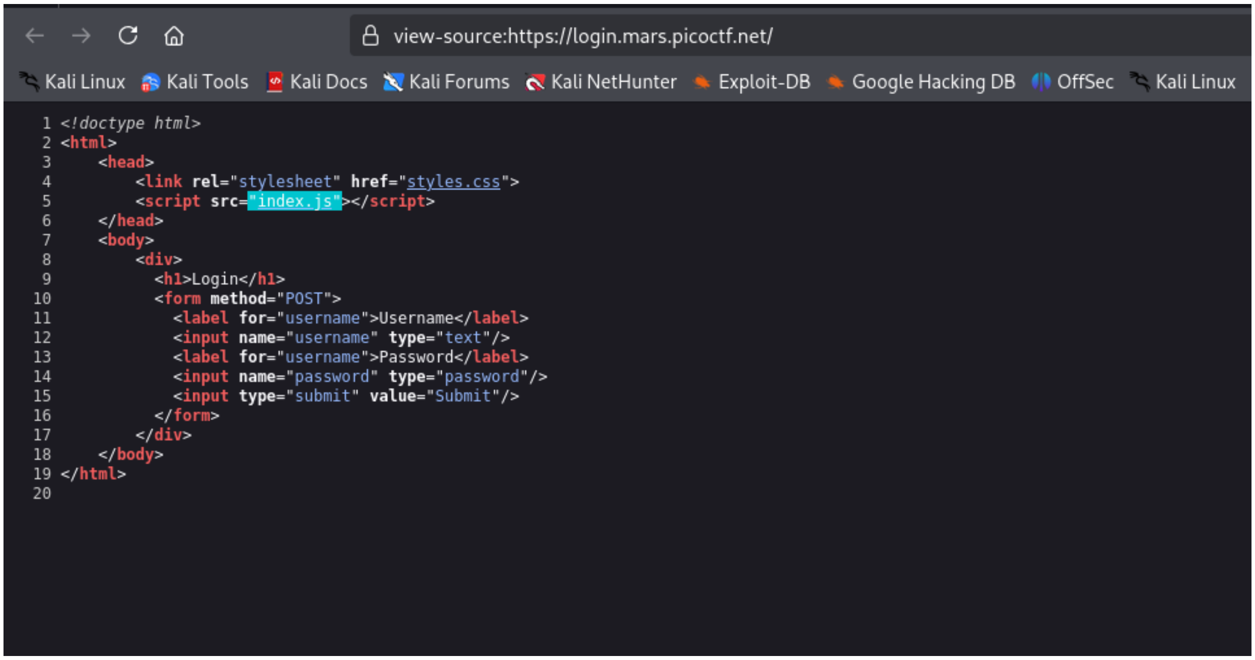
•••••



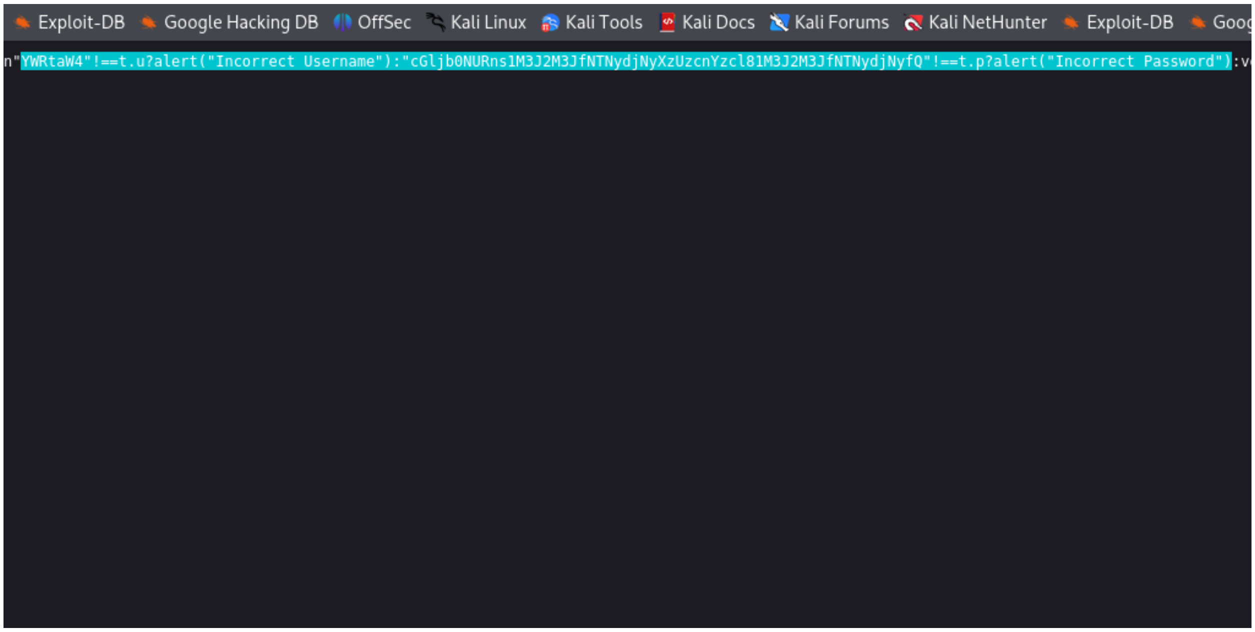
Login

login.mars.picoctf.net

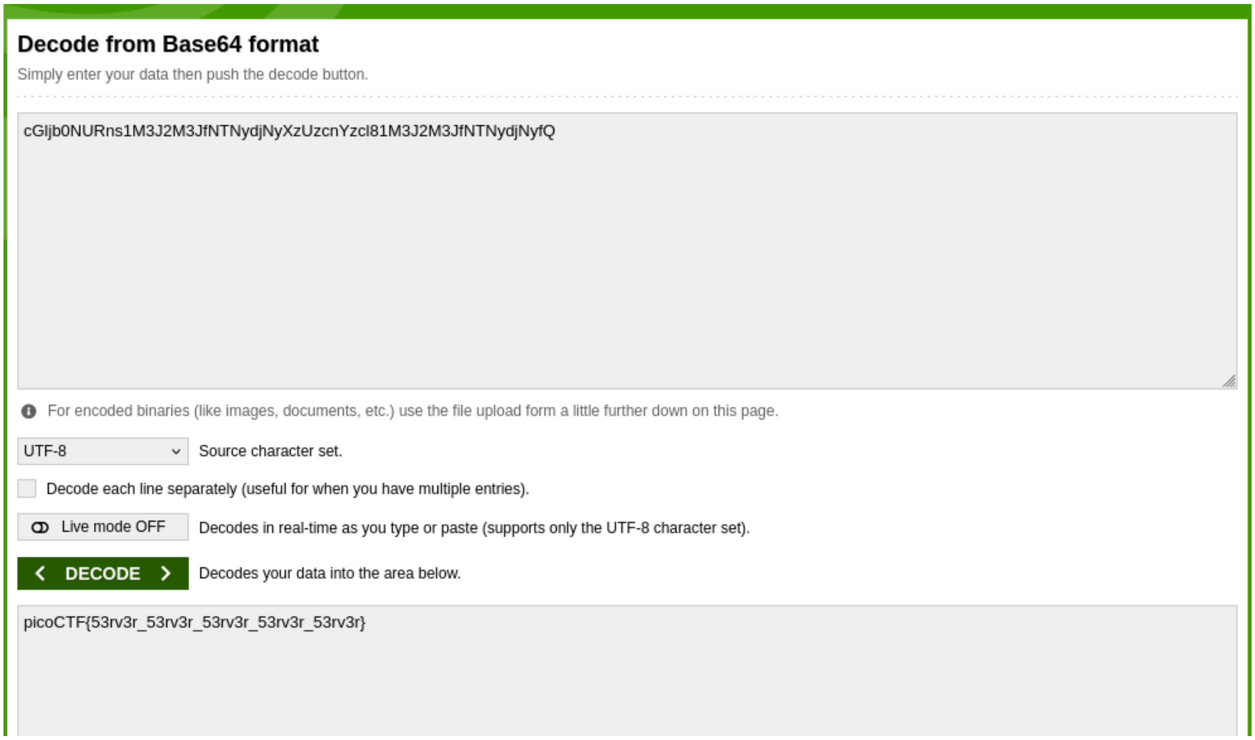
Incorrect Password



```
<!doctype html>
<html>
    <head>
        <link rel="stylesheet" href="styles.css">
        <script src="index.js"></script>
    </head>
    <body>
        <div>
          <h1>Login</h1>
          <form method="POST">
            <label for="username">Username</label>
            <input name="username" type="text"/>
            <label for="username">Password</label>
            <input name="password" type="password"/>
            <input type="submit" value="Submit"/>
          </form>
        </div>
    </body>
</html>
```



```
n"YWRtaW4"!==t.u?alert("Incorrect Username"):"cGljb0NURns1M3J2M3JfNTNydjNyXzUzcnYzcl81M3J2M3JfNTNydjNyfQ"!==t.p?alert("Incorrect Password"):v
```



Decode from Base64 format

Simply enter your data then push the decode button.

```
cGljb0NURns1M3J2M3JfNTNydjNyXzUzcnYzcl81M3J2M3JfNTNydjNyfQ
```

For encoded binaries (like images, documents, etc.) use the file upload form a little further down on this page.

UTF-8 Source character set.

□ Decode each line separately (useful for when you have multiple entries).

Live mode OFF Decodes in real-time as you type or paste (supports only the UTF-8 character set).

< DECODE > Decodes your data into the area below.

```
picoCTF{53rv3r_53rv3r_53rv3r_53rv3r_53rv3r}
```

This confirms the vulnerability is exploitable and not theoretical.

7. Impact Assessment

If this vulnerability were present in a real-world application, it could result in:

Unauthorized user access

Data exposure

Account takeover

Privilege escalation

Loss of confidentiality and integrity

8. Remediation Recommendations

To remediate the identified vulnerability:

Enforce strict server-side authentication validation

Remove all client-side trust for access control decisions

Implement secure session management

Validate authentication state on every protected request

Perform security code reviews and regular penetration testing

9. Conclusion

The penetration test identified a high-risk authentication vulnerability that allows complete access control bypass. Immediate remediation is recommended. This assessment highlights the importance of secure authentication enforcement in web applications.

Report Classification: Training / Lab Environment

Assessment Outcome: Vulnerability Confirmed and Exploited