Systems and Network Programming - IE2012

Assignment

Ashan G Punchihewa

IT22327680

Table of Contents

1 Abstract

In the realm of cybersecurity there has been an investigation that has uncovered a vulnerability known as CVE-2017-7410. This vulnerability affects WebsiteBaker versions up, to 2.10.0 and is classified as an SQL injection flaw. Essentially this means that attackers can gain access and execute SQL commands through this vulnerability. In this report we provide an analysis of the vulnerability, its consequences and suggest practical measures to mitigate the risks.

The SQL injection vulnerability allows malicious individuals to inject SQL code into the "display_name" parameters within the account/signup.php script. The impact of this can range from access to data to manipulation of database records and potential compromise of administrator credentials.

To address these risks it is recommended to update WebsiteBaker installations to version 2.10.1 or a recent release which includes critical fixes for this specific vulnerability. Additionally developers should thoroughly review their codebase. Prioritize input validation and sanitization techniques in order to minimize the risk of SQL injection vulnerabilities.

This report provides context about the WebsiteBaker framework by detailing the discovery process of this vulnerability explaining how it can be exploited and even offers a proof of concept demonstration. It concludes by emphasizing the importance of software updates and adopting coding practices as essential safeguards against potential threats.

For exploration and practical guidance, on mitigating this identified vulnerability within WebsiteBaker we have included reference links that serve as resources.

2 CVE-2017-7410

2.1 Introduction

This report explores a security vulnerability known as CVE-2017-7410 found in WebsiteBaker versions up, to 2.10.0. This vulnerability, categorized as an SQL injection presents a risk as it allows remote attackers to execute SQL commands. The purpose of this report is to provide an analysis of the vulnerability its impact and recommendations for mitigation.

2.2 Exploitation Details

The mentioned vulnerability enables an attacker without authentication to inject SQL code into the "and "display_name" parameters within the account/signup.php script. Because of input validation attackers can execute any SQL commands they wish potentially resulting in access to data and manipulation of records.

2.3 Potential Impact

Exploiting this vulnerability can have consequences for the system including;

- Unauthorized access to sensitive data.
- Manipulation of database records.
- Possible compromise of administrator credentials (administrator password MD5 hash).

2.4 Recommended Mitigation Strategies

To effectively address this vulnerability the following measures are recommended:

2.4.1 Update Software

Users are strongly advised to update their WebsiteBaker installation to version 2.10.1 or any newer release available since these updates include fixes for the identified SQL injection vulnerability.

2.4.2 Code Review

Developers should conduct a review of their code with a focus, on input validation and sanitization techniques. Using statements or parameterized queries can greatly minimize the chances of SQL injection vulnerabilities.

2.5 Background

2.5.1 Background

WebsiteBaker is a CMS (content management system) that's source and aims to make it easier to create flexible and secure websites. However there has been a security issue identified that compromises the security of this CMS.

2.5.2 Discovery of Vulnerability

On 03/24/2017 a SQL injection vulnerability was. It was made public on 04/03/2017. The specific problem affects a feature, within the file account/signup.php. When certain parameters like "username" and "display_name" are manipulated as part of the request it opens up the possibility for SQL injection attacks, which falls under CWE 89 classification.

2.5.3 Exploitation Details

The get_one function uses POST parameters "username" and "display_name" in SQL queries making it vulnerable to SQL commands due to the lack of prepared statements or proper escaping in the code.

2.5.4 Attack Method

This attack can be carried out remotely without requiring any authentication. It falls under the category T1505 in ATT&CKs classification of exploitation techniques.

2.5.5 Zero Day Exploit

For at 10 days this vulnerability was treated as a non public zero day exploit. During this time individuals estimated that exploiting this vulnerability could fetch $1k $2k in markets. To identify targets, with this vulnerability using Google Hacking techniques one could search for "inurl;account/signup.php".

2.6 Proof of concept

2.6.1 SQL Injection No1 (Parameter: username)

Payload: sql' OR SLEEP(5)--

POST /account/signup.php HTTP/1.1

Host: localhost

Content-Type: application/x-www-form-urlencoded

Content-Length: 184

action=send&redirect=http%3A%2F%2Flocalhost&submitted_when=1490134734&email-address=&name=&full_name=username=sql' OR SLEEP(5)--&display_name=test&email=test&captcha=&submit=Sign-up

The response will have a 5 second delay.

2.6.2 SQL Injection No.2 (Parameter: display_name)

Payload: sql' OR SLEEP(5)--

POST /account/signup.php HTTP/1.1

Host: localhost

Content-Type: application/x-www-form-urlencoded

Content-Length: 184

action=send&redirect=http%3A%2F%2Flocalhost&submitted_when=1490134833&email-address=&name=&full_name=&username=test&display_name=sql' OR SLEEP(5)--&email=test&captcha=&submit=Sign-up

The response will have a 5 second delay.

2.7 Solution

2.7.1 Proposed Solution

We highly recommend upgrading, to WebsiteBaker version 2.10.1 or a recent release. This update specifically tackles the SQL injection vulnerability that has been identified ensuring that the security loophole is closed.

2.8 Conclusion

The presence of a SQL injection vulnerability in WebsiteBaker versions up to 2.10.0 emphasizes the importance of updating software and following secure coding practices diligently. It is crucial to apply security patches and establish input validation mechanisms in order to protect web applications, from potential threats.

2.9 References

2.9.1 External references

- Vendor Advisory - http://forum.websitebaker.org/index.php/topic,30187.0.html.
- Vendor Advisory on Project Website - http://project.websitebaker.org/issues/39.
- SecurityFocus Advisory - http://www.securityfocus.com/bid/97495.
- SecurityTracker advidory - http://www.securitytracker.com/id/1038173.