Product: Tony

Report: Pentest Report Example

PEN-DOC20240115144451

Shub, https//:www.shub_pentest.com

14-01-2024

Project Overview

Description

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam ac felis quis eros porttitor aliquam. Ut at risus vel libero imperdiet gravida. Orci varius natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Orci varius natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Sed gravida nisl id iaculis laoreet. Proin orci libero, ultrices efficitur ultricies vitae, vehicula vitae risus. Vivamus semper fringilla velit a molestie. Phasellus eget luctus ipsum. Quisque malesuada vulputate elementum. Donec ut tempor nibh, vitae vestibulum nisl. Vivamus feugiat sollicitudin posuere. Maecenas in enim feugiat, mollis nulla eget, ullamcorper eros. Aliquam non nisi in velit pellentesque dictum sit amet ut sapien. Aliquam vitae nunc rhoncus, condimentum diam vitae, consequat velit. Phasellus laoreet eu eros nec tincidunt. Morbi nibh leo, ullamcorper ac volutpat nec, commodo vitae purus. Mauris vulputate odio ac consectetur iaculis. Nunc at odio varius felis finibus accumsan sed at velit. Sed non diam dapibus, rutrum justo non, consequat diam. Maecenas pharetra odio eget tortor venenatis, nec mattis lorem fringilla. Nunc nec volutpat erat. Nulla a lacus hendrerit, vulputate dolor vel, varius lacus. Proin vel tristique est. Phasellus sit amet malesuada ipsum.

Executive Summary

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam ac felis quis eros porttitor aliquam. Ut at risus vel libero imperdiet gravida. Orci varius natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Orci varius natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Sed gravida nisl id iaculis laoreet. Proin orci libero, ultrices efficitur ultricies vitae, vehicula vitae risus. Vivamus semper fringilla velit a molestie. Phasellus eget luctus ipsum. Quisque malesuada vulputate elementum. Donec ut tempor nibh, vitae vestibulum nisl. Vivamus feugiat sollicitudin posuere. Maecenas in enim feugiat, mollis nulla eget, ullamcorper eros. Aliquam non nisi in velit pellentesque dictum sit amet ut sapien. Aliquam vitae nunc rhoncus, condimentum diam vitae, consequat velit. Phasellus laoreet eu eros nec tincidunt.

Morbi nibh leo, ullamcorper ac volutpat nec, commodo vitae purus. Mauris vulputate odio ac consectetur iaculis. Nunc at odio varius felis finibus accumsan sed at velit. Sed non diam dapibus, rutrum justo non, consequat diam. Maecenas pharetra odio eget tortor venenatis, nec mattis lorem fringilla. Nunc nec volutpat erat. Nulla a lacus hendrerit, vulputate dolor vel, varius lacus. Proin vel tristique est. Phasellus sit amet malesuada ipsum. Donec vel lobortis mauris, vel egestas eros. Cras tempus turpis ut dolor fringilla, non luctus quam pellentesque. Nam mauris leo, eleifend vitae rutrum at, tristique sed turpis. Etiam eget purus sit amet dolor congue commodo ac accumsan eros. Phasellus tristique rutrum efficitur. Morbi ex nibh, condimentum sit amet massa id, pharetra maximus velit.

Summary of Findings Identified

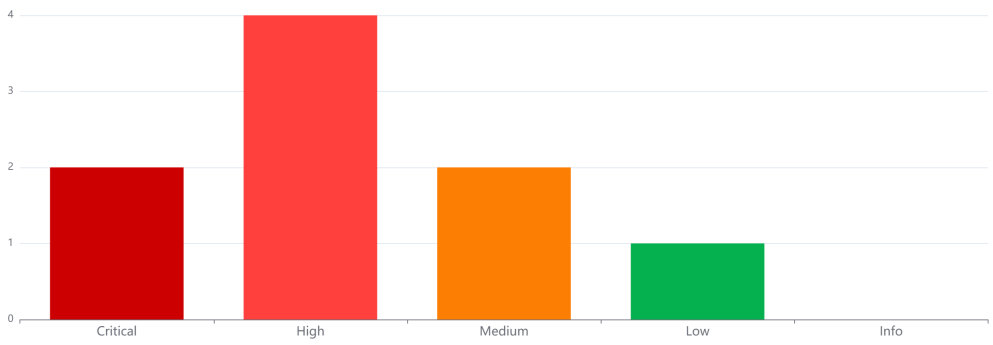


Figure 1: Executive Summary

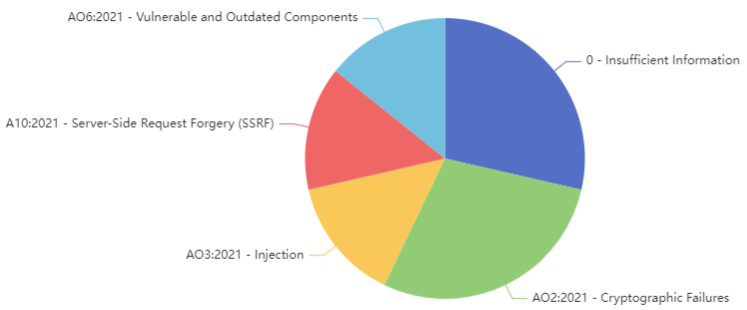


Figure 2: Breakdown by OWASP Categories

 # 1 Critical test media

 # 2 Critical MD5 Usage

 # 3 High Test Template

 # 4 High SQL Injecion

 # 5 High Crypto Failure

 # 6 High SSRF

 # 7 Medium Missing 'HttpOnly' Cookie Attribute (HTTP)

 # 8 Medium Backup File Scanner (HTTP) - Reliable Detection Reporting

 # 9 Low Bloop Change

Scope

In Scope

Test

Out of Scope

Test

Methodology

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam ac felis quis eros porttitor aliquam. Ut at risus vel libero imperdiet gravida. Orci varius natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Orci varius natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Sed gravida nisl id iaculis laoreet. Proin orci libero, ultrices efficitur ultricies vitae, vehicula vitae risus. Vivamus semper fringilla velit a molestie. Phasellus eget luctus ipsum. Quisque malesuada vulputate elementum. Donec ut tempor nibh, vitae vestibulum nisl. Vivamus feugiat sollicitudin posuere. Maecenas in enim feugiat, mollis nulla eget, ullamcorper eros. Aliquam non nisi in velit pellentesque dictum sit amet ut sapien. Aliquam vitae nunc rhoncus, condimentum diam vitae, consequat velit. Phasellus laoreet eu eros nec tincidunt. Morbi nibh leo, ullamcorper ac volutpat nec, commodo vitae purus. Mauris vulputate odio ac consectetur iaculis. Nunc at odio varius felis finibus accumsan sed at velit. Sed non diam dapibus, rutrum justo non, consequat diam. Maecenas pharetra odio eget tortor venenatis, nec mattis lorem fringilla. Nunc nec volutpat erat. Nulla a lacus hendrerit, vulputate dolor vel, varius lacus. Proin vel tristique est. Phasellus sit amet malesuada ipsum. Donec vel lobortis mauris, vel egestas eros. Cras tempus turpis ut dolor fringilla, non luctus quam pellentesque. Nam mauris leo, eleifend vitae rutrum at, tristique sed turpis. Etiam eget purus sit amet dolor congue commodo ac accumsan eros. Phasellus tristique rutrum efficitur. Morbi ex nibh, condimentum sit amet massa id, pharetra maximus velit.

Recommendations

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam ac felis quis eros porttitor aliquam. Ut at risus vel libero imperdiet gravida. Orci varius natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Orci varius natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Sed gravida nisl id iaculis laoreet. Proin orci libero, ultrices efficitur ultricies vitae, vehicula vitae risus. Vivamus semper fringilla velit a molestie. Phasellus eget luctus ipsum. Quisque malesuada vulputate elementum. Donec ut tempor nibh, vitae vestibulum nisl. Vivamus feugiat sollicitudin posuere. Maecenas in enim feugiat, mollis nulla eget, ullamcorper eros. Aliquam non nisi in velit pellentesque dictum sit amet ut sapien. Aliquam vitae nunc rhoncus, condimentum diam vitae, consequat velit. Phasellus laoreet eu eros nec tincidunt. Morbi nibh leo, ullamcorper ac volutpat nec, commodo vitae purus. Mauris vulputate odio ac consectetur iaculis. Nunc at odio varius felis finibus accumsan sed at velit. Sed non diam dapibus, rutrum justo non, consequat diam. Maecenas pharetra odio eget tortor venenatis, nec mattis lorem fringilla. Nunc nec volutpat erat. Nulla a lacus hendrerit, vulputate dolor vel, varius lacus. Proin vel tristique est. Phasellus sit amet malesuada ipsum. Donec vel lobortis mauris, vel egestas eros. Cras tempus turpis ut dolor fringilla, non luctus quam pellentesque. Nam mauris leo, eleifend vitae rutrum at, tristique sed turpis. Etiam eget purus sit amet dolor congue commodo ac accumsan eros. Phasellus tristique rutrum efficitur. Morbi ex nibh, condimentum sit amet massa id, pharetra maximus velit.

Findings and Risk Analysis

test media

Severity: Critical
CVSS Score: 10.0
CVSS Vector: CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:N/VI:N/VA:N/SC:N/SI:N/SA:N

OWASP

0 - Insufficient Information

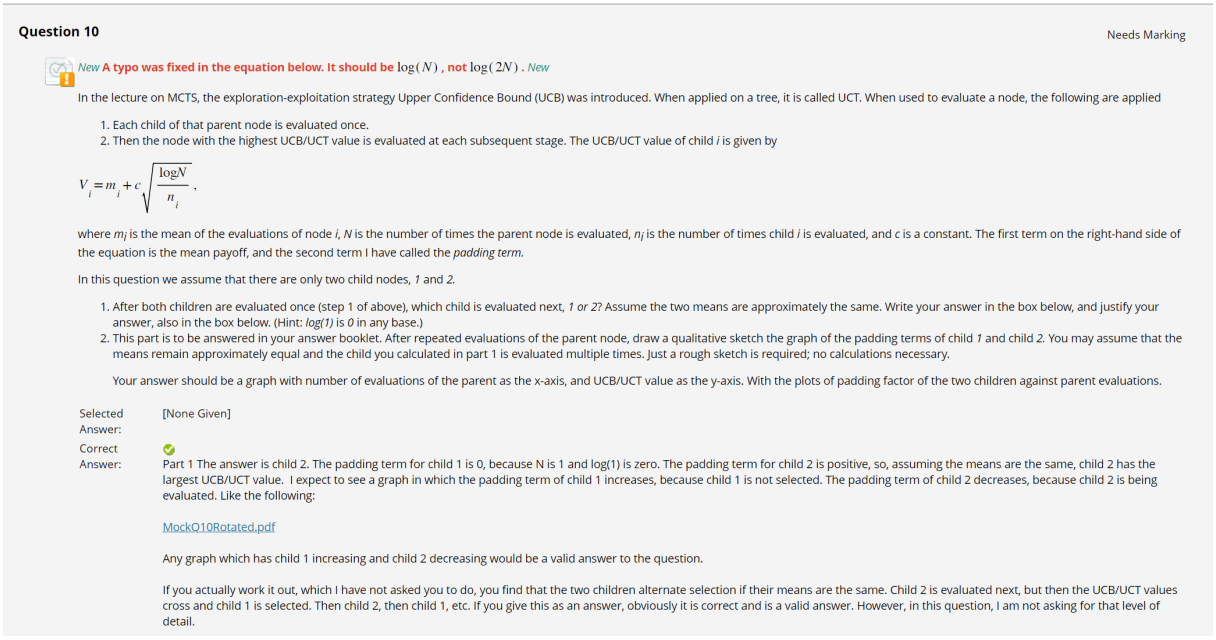
Description

TBC

Location

TBC

Impact



Question 10

Needs Marking

New **A typo was fixed in the equation below. It should be** $\log(N)$ **, not** $\log(2N)$ **.** *New*

In the lecture on MCTS, the exploration-exploitation strategy Upper Confidence Bound (UCB) was introduced. When applied on a tree, it is called UCT. When used to evaluate a node, the following are applied

1. Each child of that parent node is evaluated once.
2. Then the node with the highest UCB/UCT value is evaluated at each subsequent stage. The UCB/UCT value of child *i* is given by

$$V_i = m_i + c\sqrt{\frac{\log N}{n_i}},$$

where m_i is the mean of the evaluations of node *i*, *N* is the number of times the parent node is evaluated, n_i is the number of times child *i* is evaluated, and *c* is a constant. The first term on the right-hand side of the equation is the mean payoff, and the second term I have called the *padding term.*

In this question we assume that there are only two child nodes, *1* and *2*.

1. After both children are evaluated once (step 1 of above), which child is evaluated next, *1 or 2*? Assume the two means are approximately the same. Write your answer in the box below, and justify your answer, also in the box below. (Hint: *log(1)* is *0* in any base.)
2. This part is to be answered in your answer booklet. After repeated evaluations of the parent node, draw a qualitative sketch the graph of the padding terms of child *1* and child *2*. You may assume that the means remain approximately equal and the child you calculated in part 1 is evaluated multiple times. Just a rough sketch is required; no calculations necessary.

   Your answer should be a graph with number of evaluations of the parent as the x-axis, and UCB/UCT value as the y-axis. With the plots of padding factor of the two children against parent evaluations.

Selected Answer: [None Given]

Correct Answer: Part 1 The answer is child 2. The padding term for child 1 is 0, because N is 1 and log(1) is zero. The padding term for child 2 is positive, so, assuming the means are the same, child 2 has the largest UCB/UCT value. I expect to see a graph in which the padding term of child 1 increases, because child 1 is not selected. The padding term of child 2 decreases, because child 2 is being evaluated. Like the following:

MockQ10Rotated.pdf

Any graph which has child 1 increasing and child 2 decreasing would be a valid answer to the question.

If you actually work it out, which I have not asked you to do, you find that the two children alternate selection if their means are the same. Child 2 is evaluated next, but then the UCB/UCT values cross and child 1 is selected. Then child 2, then child 1, etc. If you give this as an answer, obviously it is correct and is a valid answer. However, in this question, I am not asking for that level of detail.

Figure 3: mock_exm_uest.png

Proof of Concept

TBC

Recommendation

TBC

References

TBC

MD5 Usage

Severity: Critical
CVSS Score: 9.3
CVSS Vector: CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:H/VI:N/VA:N/SC:H/SI:L/SA:H

OWASP

2 - Cryptographic Failures

Description

The application in question was observed using the MD5 cryptographic algorithm for hashing passwords. This algorithm is known to be deprecated and weak, making it an unsuitable choice for modern security requirements.

Location

https://example.com/user/settings

Impact

The utilization of a deprecated and weak cryptographic algorithm like MD5 makes the application highly susceptible to cyber attacks. A potential attacker can leverage well-known vulnerabilities within the MD5 algorithm to compromise user passwords, leading to unauthorized access to sensitive information.

Proof of Concept

TBC

Recommendation

- It is recommended to immediately upgrade the password hashing system to a more secure cryptographic algorithm.
- Alternatives like SHA-256 or bcrypt should be considered for password hashing.
- A thorough security review of the entire application needs should be carried out to identify and correct any other outdated security practices.

References

- https://owasp.org/www-project-top-ten/2021/A02_2021-Cryptographic_Failures
- https://en.wikipedia.org/wiki/MD5
- https://csrc.nist.gov/Projects/Hash-Functions/NIST-Policy-on-Hash-Functions

Additional notes

Wow!!!

Test Template

Severity: High
CVSS Score: 7.9
CVSS Vector: CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:N/VI:N/VA:N/SC:H/SI:L/SA:H

OWASP

0 - Insufficient Information

Description

TBC

Location

TBC

Impact

TBC

Recommendation

TBC

References

TBC

SQL Injecion

Severity: High
CVSS Score: 7.8
CVSS Vector: CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:N/VI:N/VA:N/SC:L/SI:H/SA:L

OWASP

3 - Injection

Description

An Injection flaw was identified in the target system. This flaw allows an attacker to send untrusted data to an interpreter that is incorporated into a command or query. The attacker's hostile data can trick the interpreter into executing unintended commands or accessing unauthorized data.

Location

/login.php

Impact

If successfully exploited, an attacker can take advantage of code injection to execute arbitrary code, modify the database, inject malicious content into outputs, compromise user information or even take over the server which can potentially lead to a complete system compromise.

Proof of Concept

TBC

Recommendation

- Implement a whitelist for server-side input validation and filtering.
- Use parameterized queries or prepared statements to prevent SQL injections.
- Ensure that user privileges are limited to the minimum necessary for their role to reduce the impact of a successful attack.
- Regularly update and patch all systems, software, and plugins.
- Conduct regular security reviews of your application and server.

References

- https://owasp.org/www-project-top-ten/2017/A1_2017-Injection.html
- https://cwe.mitre.org/data/definitions/77.html

- https://owasp.org/www-project-web-security-testing-guide/latest/4-Web_Application_Security_Testing/07-Input_Validation_Testing/05-Testing_for_SQL_Injection

Crypto Failure

Severity: High
CVSS Score: 7.7
CVSS Vector: CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:N/VI:N/VA:N/SC:N/SI:N/SA:H

OWASP

2 - Cryptographic Failures

Description

Cryptographic failures refer to the vulnerabilities and weaknesses in the implementation of cryptographic algorithms and protocols. These failures can lead to security breaches by attackers who can exploit the vulnerabilities to gain unauthorized access, tamper with sensitive data, or perform other malicious activities.

Location

https://example.com/user/settings

Impact

The impact of cryptographic failures can be significant and widespread. It can result in the compromise of confidential information, such as passwords, credit card details, and other sensitive data. Attackers can use this information for identity theft, financial fraud, or other forms of malicious activities. Furthermore, cryptographic failures can also lead to the loss of integrity and authenticity of data, as well as the potential for unauthorized modifications or tampering.

Proof of Concept

TBC

Recommendation

- Ensure that cryptographic algorithms and protocols are implemented correctly and securely.
- Regularly update and patch cryptographic libraries and components to address any known vulnerabilities.
- Follow secure coding practices and guidelines while developing and implementing cryptographic functions.
- Perform thorough security testing, including cryptographic testing, to identify and fix any vulnerabilities or weaknesses.
- Stay updated with the latest cryptographic standards and best practices.

- Regularly monitor and analyze cryptographic logs and alerts to detect any potential attacks or breaches.
- Implement a strong key management system to protect cryptographic keys from unauthorized access.
- Employ proper encryption mechanisms, such as using strong algorithms and key sizes.
- Ensure that proper key exchange and authentication mechanisms are in place to prevent man-in-the-middle attacks.
- Consider using reputable and audited cryptographic libraries and components.

References

- https://owasp.org/www-project-top-ten/2021/A02_2021-Cryptographic_Failures

SSRF

Severity: High
CVSS Score: 7.7
CVSS Vector: CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:N/VI:L/VA:N/SC:H/SI:N/SA:N

OWASP

10 - Server-Side Request Forgery (SSRF)

Description

During the penetration testing engagement, a Server-Side Request Forgery (SSRF) vulnerability, labelled as OWASP A10:2021, was identified. This vulnerability allows an external attacker to manipulate the system into executing requests on behalf of the server. This could be used to create requests to internal services, resulting in potential unauthorized access to sensitive data or internal management interfaces.

Location

https://example.com/user/profile

Impact

The potential impact of this vulnerability is high, given that successful exploitation can offer an attacker internal network access. With the help of this vulnerability, an attacker can bypass firewalls, probe internal servers, and access restricted data. This can lead to unauthorized access, data leakage, denial of service or even command execution.

Proof of Concept

TBC

Recommendation

- Implement a whitelist of IP addresses or ranges that it will communicate with.
- Use a server-side proxy to validate, filter, and restrict any network request made by the application.
- Always use strong access controls and least privilege policies for internal services.
- Regularly patch and update applications and servers to prevent vulnerabilities that might be used as a part of SSRF attacks.

References

- https://owasp.org/www-community/attacks/Server_Side_Request_Forgery
- https://cheatsheetseries.owasp.org/cheatsheets/Server_Side_Request_Forgery_Prevention_Cheat_Sheet.html
- https://cwe.mitre.org/data/definitions/918.html

Missing 'HttpOnly' Cookie Attribute (HTTP)

Severity: Medium
CVSS Score: 5.0
CVSS Vector:

Description

The remote HTTP web server / application is missing to set the 'HttpOnly' cookie attribute for one or more sent HTTP cookie.

Location

10.10.124.125

Proof of Concept

The cookie(s):

Set-Cookie: PHPSESSID=***replaced***; path=/ Set-Cookie: PHPSESSID=***replaced***; path=/ Set-Cookie: security=low

is/are missing the "HttpOnly" cookie attribute.

Recommendation

- Set the 'HttpOnly' cookie attribute for any session cookie
  - Evaluate / do an own assessment of the security impact on the web server / application and create an override for this result if there is none (this can't be checked automatically by this VT)

References

- https://www.rfc-editor.org/rfc/rfc6265#section-5.2.6
- https://owasp.org/www-community/HttpOnly
- https://wiki.owasp.org/index.php/Testing_for_cookies_attributes_(OTG-SESS-002)

Backup File Scanner (HTTP) - Reliable Detection Reporting

Severity: Medium
CVSS Score: 5.0
CVSS Vector:

Description

The script reports backup files left on the web server.

Location

10.10.124.125

Impact

Based on the information provided in these files an attacker might be able to gather sensitive information stored in these files.

Proof of Concept

The following backup files were identified (<URL>:<Matching pattern>):

http://10.10.124.125/config/config.inc.php.bak:^<?(php|=)

Recommendation

Delete the backup files.

References

- http://www.openwall.com/lists/oss-security/2017/10/31/1

Bloop Change

Severity: Low
CVSS Score: 1.0
CVSS Vector: CVSS:4.0/AV:A/AC:H/AT:P/PR:L/UI:P/VC:L/VI:L/VA:L/SC:L/SI:L/SA:H

CWE

108 - Struts: Unvalidated Action Form

OWASP

6 - Vulnerable and Outdated Components

Description

Yada ada yada get heked yada yada yada

Location

Test

Impact

Test

Proof of Concept

Test

Recommendation

Test

References

Test

NMap Scan Data

Host: 10.10.243.102

- **Port 22** (closed): ssh unknown
  - No scripts found.
- **Port 80** (open): http Apache httpd
  - **http-title**: Site doesn't have a title (text/html).
  - **http-server-header**: Apache
- **Port 443** (open): http Apache httpd
  - **http-server-header**: Apache
  - **http-title**: Site doesn't have a title (text/html).
  - **ssl-cert**: Subject: commonName=www.example.com Not valid before: 2015-09-16T10:45:03 Not valid after: 2025-09-13T10:45:03

Additional Notes

Wow!!!

WHy you using this still? MD5 Version Edited