Netwin Infosolutions

Black-box Penetration Testing Report

COPYRIGHT:

All intellectual property rights in this work belong to Netwin Infosolutions Pvt. Ltd. The information contained in this penetration testing report must not be reproduced or distributed to others in any form or by any means, electronic or mechanical, for any purpose, without the prior permission of Netwin Infosolutions Pvt. Ltd., or used except as expressly authorized in writing by Netwin Infosolutions Pvt. Ltd. This document is provided on the understanding that its use will be confined to the officers of your Company, and that no part of its contents will be disclosed to third parties or our direct competitors without prior written consent of Netwin Infosolutions Pvt. Ltd Copyright © 2016 Netwin Infosolutions Pvt. Ltd. (All Rights Reserved.)

CONFIDENTIALITY:

The materials and information contained herein constitute confidential information of Netwin Infosolutions Pvt. Ltd and Customer shall not disclose or transfer any of these materials or information to any third party. The recipient should not disclose this information to any third party without the prior written permission of Netwin Infosolutions Pvt. Ltd.

TRADEMARKS:

All company, brand, and product names are referenced for identification purposes only and may be trademarks or registered trademarks that are the sole property of their respective owners.

DOCUMENT INFORMATION

Attribute	Value
Date	03-01-2017
Document Title	Intermittent Penetration Testing Report
Document Classification	Confidential
Document Circulation	Private (not for public circulation)
Author	Netwin Infosolutions Pvt. Ltd.

Table of Contents

1. EXECUTIVE SUMMARY:

Company has assigned the task of carrying out Black Box Penetration Testing of Plus Educational Developments website http://192.168.43.97/plus_site/ to Netwin Infosolutions Pvt. Ltd.

This is Back Box Penetration Testing report. This test was performed during 28/12/.2017 to 03/01/2018. The detailed report about each task and our findings are described below.

The purpose of the test is to determine security vulnerabilities in the server configurations and the web application running on the server specified as part of the scope. The tests are carried out assuming the identity of an attacker or a user with malicious intent. At the same time due care is taken not to harm the server.

2. APPROACH:

- Perform broad scans to identify potential areas of exposure and services that may act as entry points.
- Perform targeted scans and manual investigation to validate vulnerabilities.
- Identify and validate vulnerabilities.
- Rank vulnerabilities based on threat level, loss potential and likelihood of exploitation.
- Perform supplemental research and development activities to support analysis.
- Identify issues of immediate consequence and recommend solutions.
- Develop long-term recommendations to enhance security.
- Transfer knowledge.

3. SCOPE:

The scope of this testing is limited to the front end URL http://192.168.43.97/plus_site/ at staging server. This is “Black Box Penetration Testing Report”; it is an integral part of complete penetration testing report.

4. INTRODUCTION TO VULNERABILITIES BASED ON OWASP:

Vulnerability	Security Risk	Recommendation	More Information
SQL injection	Malicious users can inject SQL commands into an SQL statement, via web page input. Injected SQL commands can alter SQL statement and compromise the security.	-Use SQL parameters or stored procedures. - Escape all user supplied inputs.	https://www.owasp.org/index.php/SQL_Injection
HTML Injection	Malicious users can inject their own content into the webpage. (purpose is to deface web page)	Plain text input should not be included directly to HTML.	https://www.owasp.org/index.php/HTML_Injection
Cross Site Scripting (XSS)	Attackers can execute scripts in a victim's browser to hijack user sessions, deface web sites, insert hostile content, redirect users, and hijack the user's browser using malware.	Sanitize user input.	https://www.owasp.org/index.php/Top_10_2013-A3-Cross-Site_Scripting_(XSS)
Blind SQL Injection	It is possible to execute SQL commands on the SQL Server through the application.	User input must not directly be embedded in SQL statements. Instead, parameterized statements must be used. User input must be carefully escaped or filtered.	https://www.owasp.org/index.php/Blind_SQL_Injection
Application Misconfiguration	Attackers can get unauthorized access to some system data or functionality and occasionally, such flaws result in a complete system compromise.	A process for keeping abreast of and deploying all new software updates and patches in a timely manner to each deployed environment. This need to include all code libraries as well, which are	https://www.owasp.org/index.php/Top_10_2013-A5-Security_Misconfiguration

		frequently overlooked.	
HTTP Response Splitting	It is possible to steal or manipulate customer session and cookies, which might be used to impersonate a legitimate user, allowing the hacker to view or alter user records, and to perform transactions as that user It is possible to deface the site content through web cache poisoning	The generic solution is to URL-encode strings before inclusion into HTTP headers such as Location or Set-Cookie.	https://www.owasp.org/index.php/HTTP_Response_Splitting
Directory Listing	It is possible to retrieve information about the site's file system structure, which may help the attacker to map the web site	If the forbidden resource is not required, remove it from the site. If possible, use Vulnerability a "404 - Not Found" response status code instead of "403 - Forbidden". This change will obfuscate the presence of the directory in the site, and will prevent the site structure from being exposed.	https://www.owasp.org/index.php/OWASP_Periodic_Table_of_Vulnerabilities_-_Directory_Indexing
LDAP Injection	This could result in the execution of arbitrary commands such as granting permissions to unauthorized users, and content modifications.	Sanitize user input.	https://www.owasp.org/index.php/LDAP_injection
Cross-Site Request Forgery (CSRF)	Attackers can trick victims into performing any state changing operation the victim is authorized to perform.	-Include the unique token in a hidden field - Use Captcha - Use OTP	https://www.owasp.org/index.php/Top_10_2013-A8-Cross-Site_Request_Forgery_(CSRF)
Improper Input Handling	An attacker could read confidential data if he/she is able to control resource	Never use primitives in custom code.	https://www.owasp.org/index.php/OWASP_Periodic_Tab

	references.		le_of_Vulnerabilities_-_Improper_Input_Handling
Insufficient Authentication	Incorrect verification of identity and permissions can result in an unauthorized attacker accessing sensitive data or functionality.	Always apply least privilege principle to all transactions and data access. Define access control matrix for all features and implement policy before implementing the feature.	https://www.owasp.org/index.php/OWASP_Periodic_Table_of_Vulnerabilities_-_Insufficient_Authentication/Authorization
Using Components with Known Vulnerabilities	The full range of weaknesses is possible, including injection, broken access control, XSS, etc. The impact could range from minimal to complete host takeover and data compromise.	-Identify all components and the versions you are using, including all dependencies. -Monitor the security of these components in public databases, project mailing lists, and security mailing lists, and keep them up to date. -Consider adding security wrappers around components to disable unused functionality	https://www.owasp.org/index.php/Top_10_2013-A9-Using_Components_with_Known_Vulnerabilities
Server Misconfiguration	Attackers can get unauthorized access to some system data or functionality and occasionally, such flaws result in a complete system compromise.	A process for keeping abreast of and deploying all new software updates and patches in a timely manner to each deployed environment. This need to include all code libraries as well, which are frequently overlooked.	https://www.owasp.org/index.php/Top_10_2010-A6-Security_Misconfiguration
Invalidated Redirects and	Such redirects may attempt to install malware or trick	Avoid using redirects and forwards. Don't involve user	https://www.owasp.org/index.php/Top

Forwards	victims into disclosing passwords or other sensitive information. Unsafe forwards may allow access control bypass.	parameters in calculating the destination. If destination parameters can't be avoided, ensure that the supplied value is valid, and authorized for the user.	_10_2013-A10-Unvalidated_Redirects_and_Forwards
Insecure Direct Object References	It can compromise all the data that can be referenced by the parameter. Unless object references are unpredictable, it's easy for an attacker to access all available data of that type.	-Use per user or session indirect object references. -Check access.	https://www.owasp.org/index.php/Top_10_2013-A4-Insecure_Direct_Object_References
Broken Authentication and Session Management	Such flaws may allow some or even all accounts to be attacked. Once successful, the attacker can do anything the victim could do. Privileged accounts are frequently targeted.	Proper authentication mechanism should be implemented along with good password policy	https://www.owasp.org/index.php/Top_10_2013-A2-Broken_Authentication_and_Session_Management

5. TYPES OF TESTS PERFORMED ON THIS APPLICATION:

Missing functional level access	HTML Injection	Header injection	Cross Site Request Forgery
Cross site Scripting	Clickjacking	HTTP Methods	Cookies Management
Directory Traversal	Cross Origin Resource Sharing	Error Code analysis	Request and Response tampering

6. TABULAR SUMMARY:

The following table summarizes the application vulnerability assessment:

Category	**Description**
No. of vulnerabilities identified	**6**

Severity of vulnerabilities:

High	**0**
Medium	**2**
Low	**4**

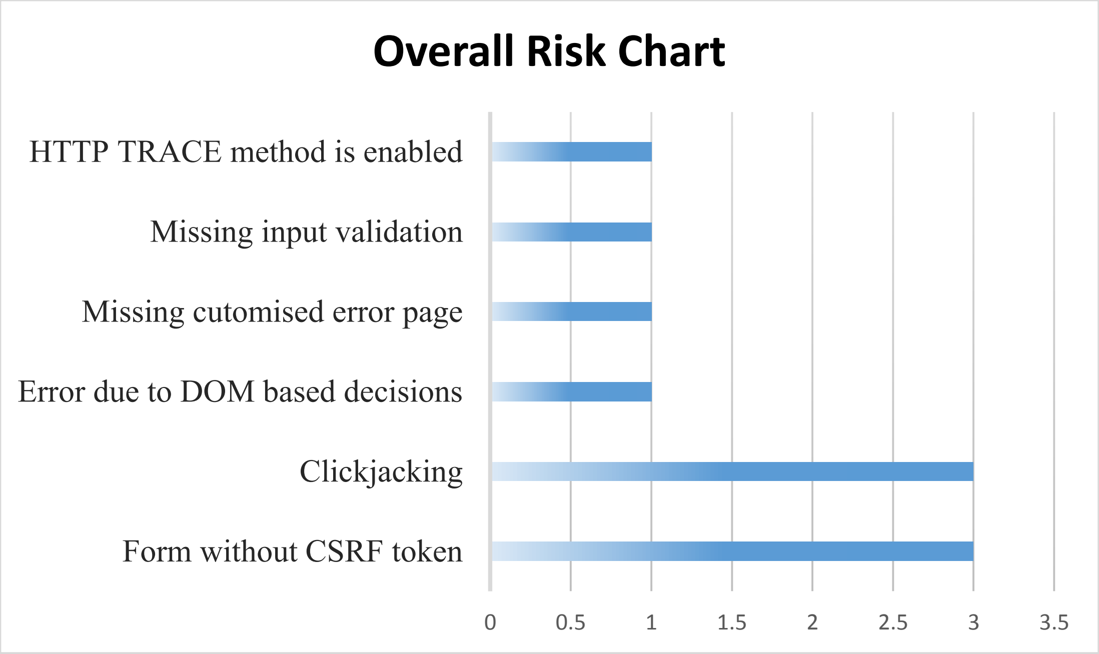
7. OVERALL RISK CHART:

On X-axis: Risk Factor of vulnerabilities found in application

On Y-axis: Names of vulnerabilities found in application

On X-axis:

1=LOW SEVERITY
3=MEDIUM SEVERITY
5=HIGH SEVERITY



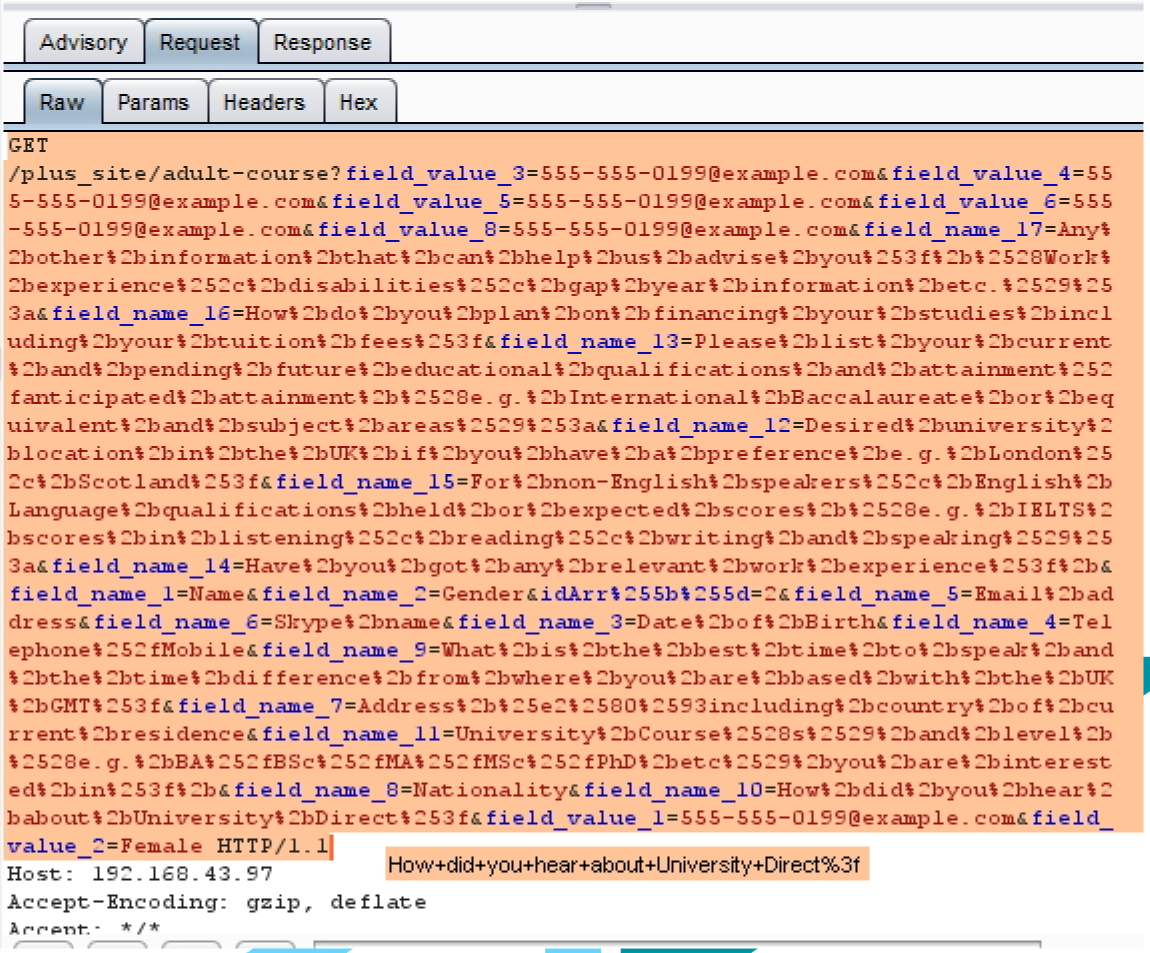
8. TECHNICAL REPORT:

Vulnerability: 1

Identified Vulnerability	Form without CSRF token
Vulnerability Description	CSRF attack is occurs when a malicious Web site, email, blog, instant message, or program causes a user's Web browser to perform an unwanted action on a trusted site for which the user is currently authenticated.
Risk Description	This attack could result in a transfer of funds, changing a password, or purchasing an item in the user's context. In effect, CSRF attacks are used by an attacker to make a target system perform a function (funds Transfer, form submission etc.) via the target's browser without knowledge of the target user, at least until the unauthorized function has been committed.
Severity	MEDIUM
Remediation	- Implement anti-CSRF tokens into all requests that perform actions which change the application state or which add/modify/delete content. An anti-CSRF token should be a long randomly generated value unique to each user so that attackers cannot easily brute-force it. It is important that anti-CSRF tokens are validated when user requests are handled by the application. The application should both verify that the token exists in the request, and also check that it matches the user's current token. If either of these checks fails, the application should reject the request.

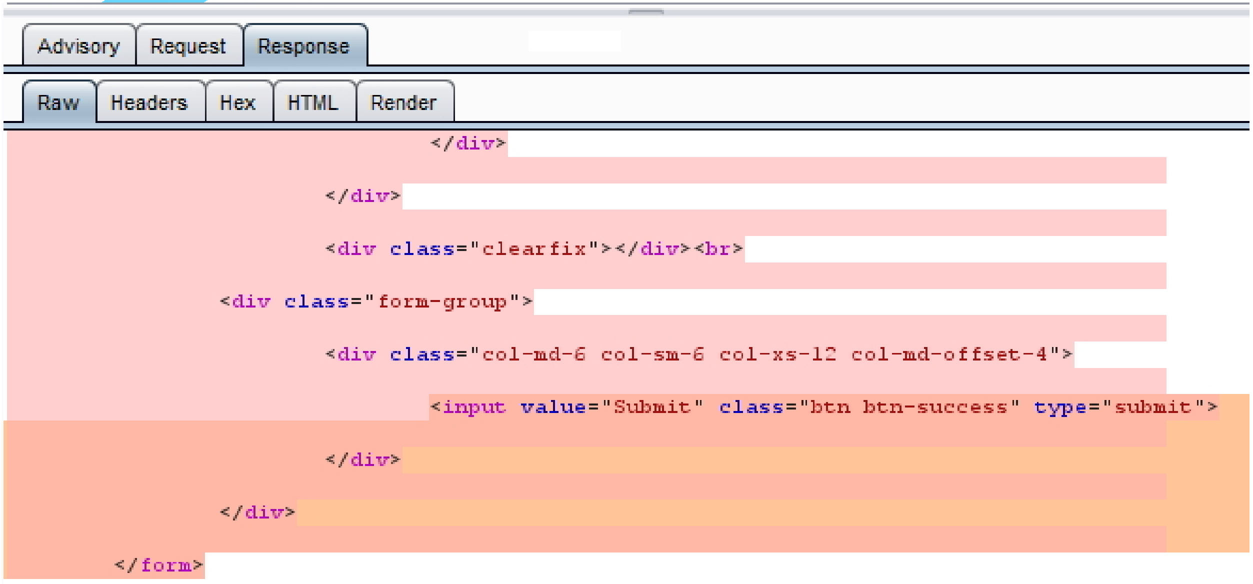
Screenshot:

Request:



```
GET
/plus_site/adult-course?field_value_3=555-555-0199@example.com&field_value_4=55
5-555-0199@example.com&field_value_5=555-555-0199@example.com&field_value_6=555
-555-0199@example.com&field_value_8=555-555-0199@example.com&field_name_17=Any%
2bother%2binformation%2bthat%2bcan%2bhelp%2bus%2badvise%2byou%253f%2b%2528Work%
2bexperience%252c%2bdisabilities%252c%2bgap%2byear%2binformation%2betc.%2529%25
3a&field_name_16=How%2bdo%2byou%2bplan%2bon%2bfinancing%2byour%2bstudies%2bincl
uding%2byour%2btuition%2bfees%253f&field_name_13=Please%2blist%2byour%2bcurrent
%2band%2bpending%2bfuture%2beducational%2bqualifications%2band%2battainment%252
fanticipated%2battainment%2b%2528e.g.%2bInternational%2bBaccalaureate%2bor%2beq
uivalent%2band%2bsubject%2bareas%2529%253a&field_name_12=Desired%2buniversity%2
blocation%2bin%2bthe%2bUK%2bif%2byou%2bhave%2ba%2bpreference%2be.g.%2bLondon%25
2c%2bScotland%253f&field_name_15=For%2bnon-English%2bspeakers%252c%2bEnglish%2b
Language%2bqualifications%2bheld%2bor%2bexpected%2bscores%2b%2528e.g.%2bIELTS%2
bscores%2bin%2blistening%252c%2breading%252c%2bwriting%2band%2bspeaking%2529%25
3a&field_name_14=Have%2byou%2bgot%2bany%2brelevant%2bwork%2bexperience%253f%2b&
field_name_1=Name&field_name_2=Gender&idArr%255b%255d=2&field_name_5=Email%2bad
dress&field_name_6=Skype%2bname&field_name_3=Date%2bof%2bBirth&field_name_4=Tel
ephone%252fMobile&field_name_9=What%2bis%2bthe%2bbest%2btime%2bto%2bspeak%2band
%2bthe%2btime%2bdifference%2bfrom%2bwhere%2byou%2bare%2bbased%2bwith%2bthe%2bUK
%2bGMT%253f&field_name_7=Address%2b%25e2%2580%2593including%2bcountry%2bof%2bcu
rrent%2bresidence&field_name_11=University%2bCourse%2528s%2529%2band%2blevel%2b
%2528e.g.%2bBA%252fBSc%252fMA%252fMSc%252fPhD%2betc%2529%2byou%2bare%2binterest
ed%2bin%253f%2b&field_name_8=Nationality&field_name_10=How%2bdid%2byou%2bhear%2
babout%2bUniversity%2bDirect%253f&field_value_1=555-555-0199@example.com&field_
value_2=Female HTTP/1.1
Host: 192.168.43.97
Accept-Encoding: gzip, deflate
Accept: */*
```

Response:

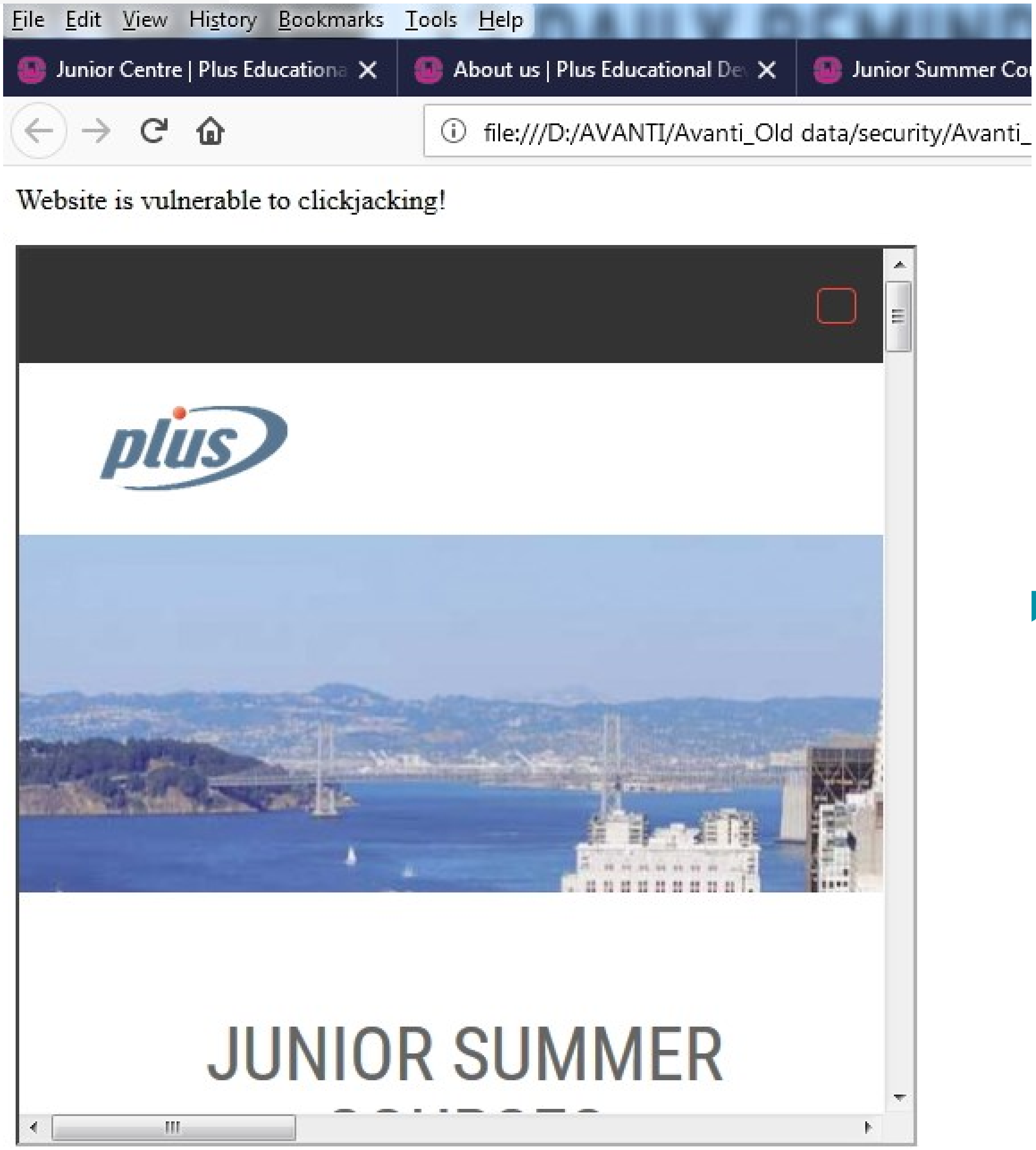


```
                                </div>

                        </div>

                        <div class="clearfix"></div><br>

                <div class="form-group">

                        <div class="col-md-6 col-sm-6 col-xs-12 col-md-offset-4">

                                <input value="Submit" class="btn btn-success" type="submit">

                        </div>

                </div>

        </form>
```

Vulnerability: 2

Identified Vulnerability	Clickjacking
Vulnerability Description	Clickjacking, also known as a "UI redress attack", is when an attacker uses multiple transparent or opaque layers to trick a user into clicking on a button or link on another page when they were intending to click on the top level page. Thus, the attacker is "hijacking" clicks meant for their page and routing them to another page, most likely owned by another application, domain, or both. By inducing victim users to perform actions such as mouse clicks and keystrokes, the attacker can cause them to unwittingly carry out actions within the application that is being targeted. For example: form submission, add/delete user or data table entries (if user logged in as admin).
Affected URL	All webpages are vulnerable to clickjacking
Severity	Medium
Remediation	-The application should return a response header with the name X-Frame-Options and the value DENY to prevent framing altogether, or the value SAMEORIGIN to allow framing only by pages on the same origin as the response itself.

Screenshot:


File Edit View History Bookmarks Tools Help
Junior Centre | Plus Educationa
About us | Plus Educational De
Junior Summer Co
JUNIOR SUMMER

file:///D:/AVANTI/Avanti_Old data/security/Avanti_

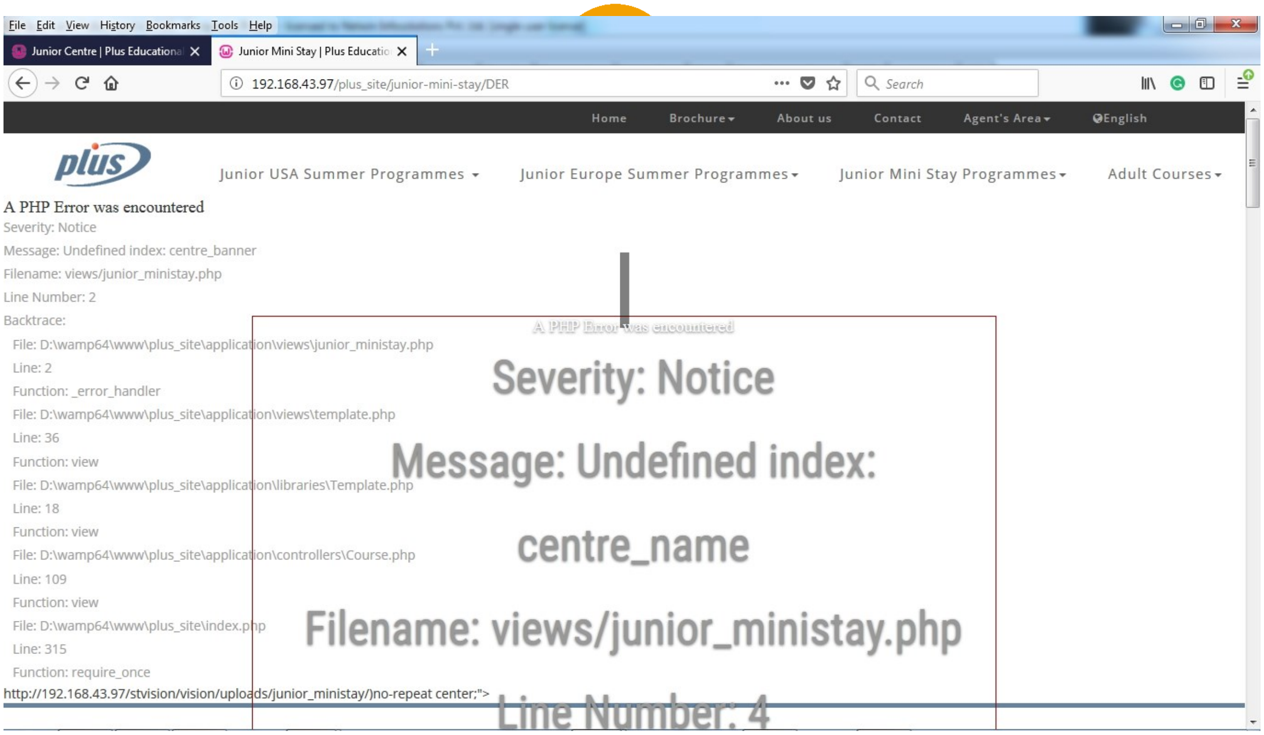
Website is vulnerable to clickjacking!

Vulnerability: 3

Identified Vulnerability	Error due to DOM based decisions
Vulnerability	Webpage is taking decisions on the basis of user input. If user gives an invalid input then page is showing errors.

Description	An important aspect of secure application development is to prevent information leakage. Error messages give an attacker great insight into the inner workings of an application. The purpose of reviewing the Error Handling code is to assure the application fails safely under all possible error conditions, expected and unexpected. No sensitive information is presented to the user when an error occurs.
Affected URL	http://192.168.43.97/plus_site/junior-mini-stay/DER
Severity	LOW
Remediation	-Review the source code for security misconfigurations. -Implement custom error page for unusual behaviours of web

Screenshot:



File Edit View History Bookmarks Tools Help
Junior Centre | Plus Educational
Junior Mini Stay | Plus Educatio
192.168.43.97/plus_site/junior-mini-stay/DER
Search
Home
Brochure
About us
Contact
Agent's Area
English
plus
Junior USA Summer Programmes
Junior Europe Summer Programmes
Junior Mini Stay Programmes
Adult Courses

```
A PHP Error was encountered
Severity: Notice
Message: Undefined index: centre_banner
Filename: views/junior_ministay.php
Line Number: 2
Backtrace:
  File: D:\wamp64\www\plus_site\application\views\junior_ministay.php
  Line: 2
  Function: _error_handler
  File: D:\wamp64\www\plus_site\application\views\template.php
  Line: 36
  Function: view
  File: D:\wamp64\www\plus_site\application\libraries\Template.php
  Line: 18
  Function: view
  File: D:\wamp64\www\plus_site\application\controllers\Course.php
  Line: 109
  Function: view
  File: D:\wamp64\www\plus_site\index.php
  Line: 315
  Function: require_once
http://192.168.43.97/stvision/vision/uploads/junior_ministay/)no-repeat center;">
```

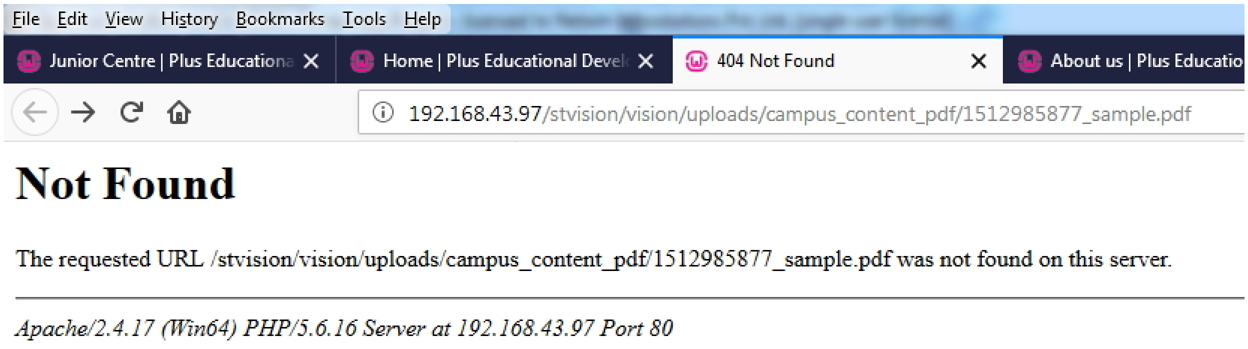
```
A PHP Error was encountered
Severity: Notice
Message: Undefined index:
centre_name
Filename: views/junior_ministay.php
Line Number: 4
```

Vulnerability: 4

Identified Vulnerability	Missing customized error page
Vulnerability Description	An important aspect of secure application development is to prevent information leakage. Error messages give an attacker great insight into the inner workings of an application. The purpose of reviewing the Error Handling code is to assure the application fails safely under all possible error conditions, expected and

	unexpected. No sensitive information is presented to the user when an error occurs.
Severity	LOW
Remediation	-Review the source code for security misconfigurations. -Implement custom error page for unusual behaviours of web

Screenshot:


File Edit View History Bookmarks Tools Help
Junior Centre | Plus Educationa
Home | Plus Educational Devel
404 Not Found
About us | Plus Educatio
192.168.43.97/stvision/vision/uploads/campus_content_pdf/1512985877_sample.pdf

Not Found

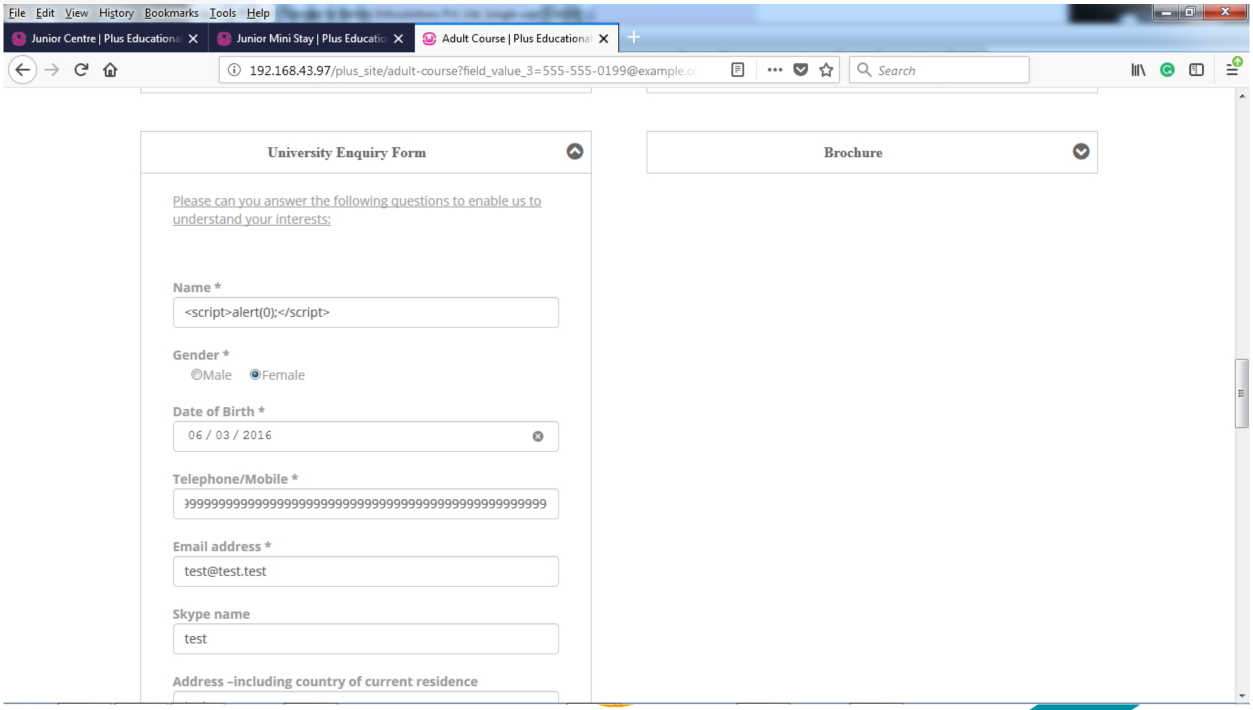
The requested URL /stvision/vision/uploads/campus_content_pdf/1512985877_sample.pdf was not found on this server.

Apache/2.4.17 (Win64) PHP/5.6.16 Server at 192.168.43.97 Port 80

Vulnerability: 5

Identified Vulnerability	Missing input validations
Vulnerability Description	Form present on the webpage is not having proper input validation. Attacker may perform buffer overflow attack and other scripting attacks.
Severity	LOW
Remediation	Validate each input taken from user. Give max character limit to avoid attacks on the field.

Screenshot:



File Edit View History Bookmarks Tools Help
Junior Centre | Plus Educational
Junior Mini Stay | Plus Educatio
Adult Course | Plus Educational
192.168.43.97/plus_site/adult-course?field_value_3=555-555-0199@example.c
Search

University Enquiry Form

Brochure

Please can you answer the following questions to enable us to understand your interests:

Name *
<script>alert(0);</script>

Gender *
○Male ●Female

Date of Birth *
06 / 03 / 2016

Telephone/Mobile *
99999999999999999999999999999999999999999999999999

Email address *
test@test.test

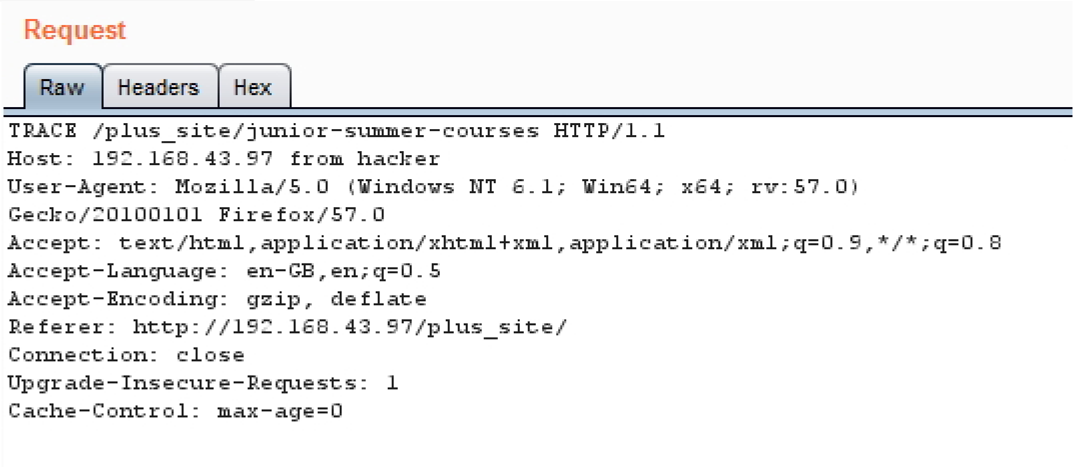
Skype name
test

Address –including country of current residence

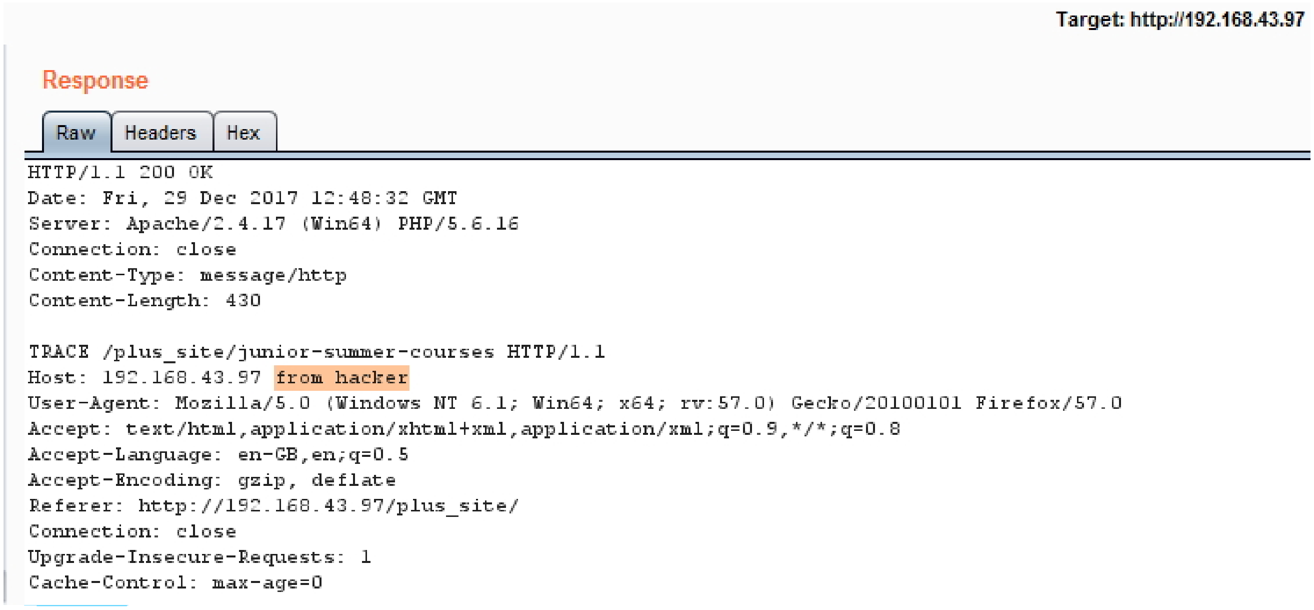
Vulnerability: 6

Identified Vulnerability	HTTP TRACE method is enabled
Vulnerability Description	TRACE allows the client to see what is being received at the other end of the request chain and use that data for testing or diagnostic information.
Risk Description	TRACE enabled may lead to Cross-Site Tracing (XST). XST could be used as a method to steal user's cookies via Cross-site Scripting (XSS) even if the cookie has the "HttpOnly" flag set and/or exposes the user's Authorization header.
Severity	LOW
Remediation	-Keep TraceEnable directive “off” in server configuration.

Screenshot:



```
TRACE /plus_site/junior-summer-courses HTTP/1.1
Host: 192.168.43.97 from hacker
User-Agent: Mozilla/5.0 (Windows NT 6.1; Win64; x64; rv:57.0)
Gecko/20100101 Firefox/57.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-GB,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.43.97/plus_site/
Connection: close
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
```



```
HTTP/1.1 200 OK
Date: Fri, 29 Dec 2017 12:48:32 GMT
Server: Apache/2.4.17 (Win64) PHP/5.6.16
Connection: close
Content-Type: message/http
Content-Length: 430

TRACE /plus_site/junior-summer-courses HTTP/1.1
Host: 192.168.43.97 from hacker
User-Agent: Mozilla/5.0 (Windows NT 6.1; Win64; x64; rv:57.0) Gecko/20100101 Firefox/57.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-GB,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.43.97/plus_site/
Connection: close
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
```

9. CONCLUSION:

The focused effort to address the problems mention in this report can result in dramatic security improvements. Most of the identified problems do not require high-tech solutions. It requires just knowledge and commitment to good practice.

Good approach must be evaluated and improved continuously for secure system. Establishing the organizational structure that will support these ongoing improvements is essential in order to maintain control of corporate information systems.

We conclude that the overall security needs to improve. We hope that the issues mentioned in this report will be addressed.