



BUDDY TUTOR BUYONIA SOFT
WEB APPLICATION PENETRATION TEST
V1.0

SUMMARY OF WORK PERFORMED	1
METHODOLOGY.....	2
TOOLS USED.....	3
EXECUTIVE SUMMARY.....	4-5
INFORMATION GATHERING.....	6
SUMMARY OF FINDINGS.....	7-8
WEB APPLICATION PENETRATION TEST.....	9-35



SUMMARY OF WORK PERFORMED

A web application penetration test was performed on the buddy tutor education site. This application consists of a web portal that provides access to account information and the ability to manage teachers, students and parents. The following web server provided by Buyonia Soft for testing:

- <https://staging.buddytutor.co.uk/>



METHODOLOGY

An Application Penetration Test is designed to identify vulnerabilities in an application which could negatively impact the organization if exploited by an attacker. Manual and automated assessment will be conducted using a testing methodology based on expert knowledge, in combination with information provided by Buyonia Soft. Application testing is conducted in accordance with OWASP Top 10, application security best practices, and internal checklists developed to ensure thorough coverage. CrusherslabQA Application Penetration Testing consists of the following phases:

- **Pre-Assessment** – CrusherslabQA will request access to the systems in advance of the test and guide the customer through the testing process on a pre-assessment Call.
- **Enumeration** – Using resources such as DNS, Google, and Bing, CrusherslabQA will look for information that is available that may be helpful to an attacker.
- **Unauthenticated Testing** – CrusherslabQA will evaluate the application from the perspective of an attacker who does not have authenticated access to the application.
- **Authentication Testing** – A large number of vulnerabilities result from weaknesses in the authentication process. This phase examines the various authentication mechanisms in place.
- **Authenticated Testing** – This phase simulates an attacker who has access, or maliciously obtains access, to credentials to log in to the application.
- **Reporting** – A report outlining all identified issues will be prepared which focuses on presenting identified vulnerabilities in a manner which makes them effective to remediate.

TOOLS USED

In addition to a variety of open source tools, exploits, and utilities used on an as-needed basis, the following tools may be used when conducting an Application Penetration Test:

- Google or other search engines
- Mozilla Firefox
- Nessus Network Vulnerability Scanner
- Nikto
- Nmap
- Burp Suite
- SQLMap
- Zap
- dirbuster/gobuster
- Wireshark



EXECUTIVE SUMMARY

During the application penetration test, nineteen vulnerabilities were discovered, the high risk vulnerabilities are four, medium risk are seven and low risk are eight. Their most high risk is unauthorized access. And some MiM attacks are done for this application. There are cryptographic weaknesses that allow an attacker to bypass the password. And leaked personal information. And use some vulnerable js libraries.

Scan some vulnerable ports and detect some problems. Some vulnerabilities are Absence of Anti-CSRF Token, Content Security Policy (CSP) Header Not Set, Cross-Domain Misconfiguration, HTTP to HTTPS Insecure Transition in Form Post, Missing Anti-clickjacking Header. Here some problems are in code, some misconfiguration and server.

There are some low risks like Cross-Domain JavaScript Source File Inclusion. The page includes one or more script files from a third-party domain. Strict-Transport-Security Header Not Set, Server Leaks Version Information via "Server" HTTP Response Header Field, Timestamp Disclosure - Unix, X-Content-Type-Options Header Missing, Big Redirect Detected (Potential Sensitive Information Leak), Cookie No HttpOnly Flag, Cookie Without Secure Flag. There are also some misconfigurations and lagging in code.



The following chart shows the distribution of findings across all severity levels:

Severity	Count of Findings
High	4
Medium	7
Low	8

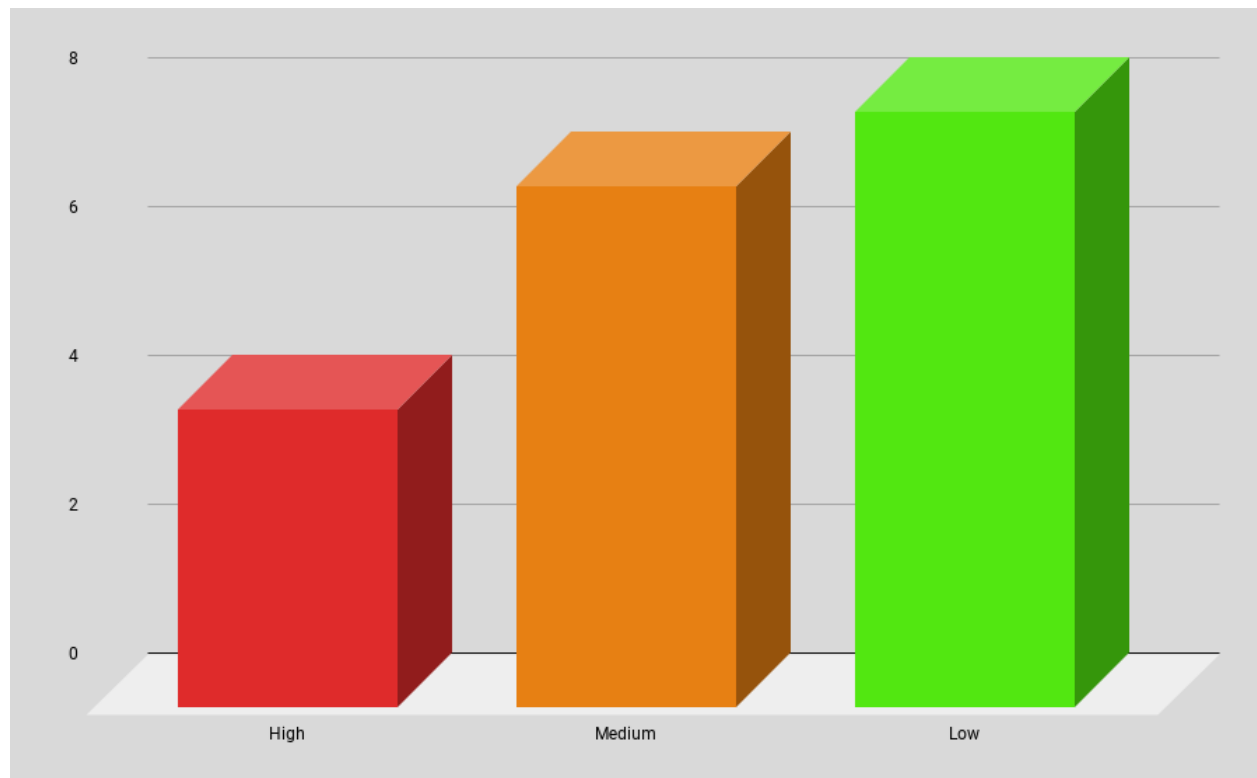


Chart1: Severity and findings

INFORMATION GATHERING

Here we found some ports:

```
└─$ nmap -sV staging.buddytutor.co.uk
Starting Nmap 7.93 ( https://nmap.org ) at 2023-03-20 18:01 +06
Nmap scan report for staging.buddytutor.co.uk (88.208.212.212)
Host is up (0.24s latency).
Not shown: 988 filtered tcp ports (no-response)
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsftpd 3.0.3
22/tcp    open  ssh          OpenSSH 8.2p1 Ubuntu 4ubuntu0.5 (Ubuntu Linux; protocol 2.0)
80/tcp    open  http         Apache httpd 2.4.41 ((Ubuntu))
443/tcp    open  ssl/http     Apache httpd 2.4.41
1433/tcp   open  ms-sql-s     Microsoft SQL Server 2019 15.00.4298
3389/tcp   closed ms-wbt-server
7000/tcp   closed afs3-fileserver
8000/tcp   closed http-alt
8100/tcp   closed xprint-server
8443/tcp   closed https-alt
8500/tcp   closed fftp
8600/tcp   closed asterix
Service Info: Host: demo.buyoniasoftksa.com; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 37.96 seconds
```

Figure1: Nmap scanning ports

SUMMARY OF FINDINGS

Finding information in this report is presented in order of severity. Severity ratings within this report are presented without the knowledge of the business risk that the vulnerabilities present to Buddy tutor or its customers.

The following vulnerabilities were discovered during the Application Penetration Test:

#	Finding	Severity
1	Unauthorized access	High
2	Successful Brute force attack	High
3	PII Disclosure	High
4	Vulnerable JS Library	High
5	Security restrictions bypass in vsftpd	Medium
6	MitM attack in OpenSSH client	Medium
7	Absence of Anti-CSRF Token	Medium
8	Content Security Policy (CSP) Header Not Set	Medium
9	Cross-Domain Misconfiguration	Medium
10	HTTP to HTTPS Insecure Transition in Form Post	Medium
11	Missing Anti-clickjacking Header	Medium
12	Cross-Domain JavaScript Source File Inclusion	Low
13	Server Leaks Version Information via "Server" HTTP Response Header Field.	Low

14	Strict-Transport-Security Header Not Set	Low
15	Timestamp Disclosure - Unix	Low
16	X-Content-Type-Options Header Missing	Low
17	Big Redirect Detected (Potential Sensitive Information Leak)	Low
18	Cookie No HttpOnly Flag	Low
19	Cookie Without Secure Flag	Low



WEB APPLICATION PENETRATION TEST

1. Unauthorized access

Severity : High

Description:

A user is presented with a list based on his user_id, from this list he can select a user which presents him with the following link in the browser:

<https://staging.buddytutor.co.uk/teacher-dashboard/10>

Now if any user changes the 10 into a 9, he is able to view and edit the data of another user.

Evidence:

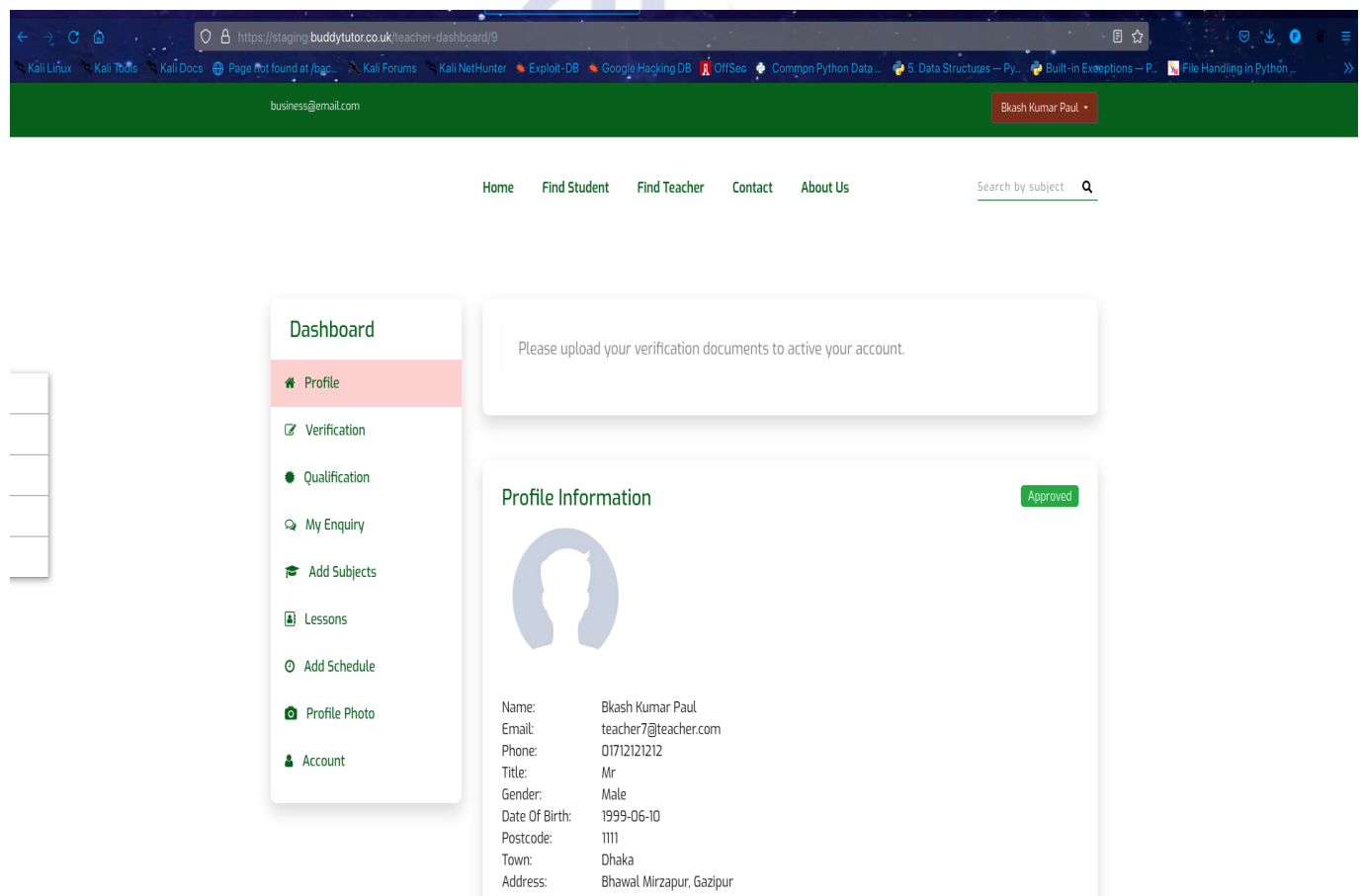


Figure2: Unauthorized access

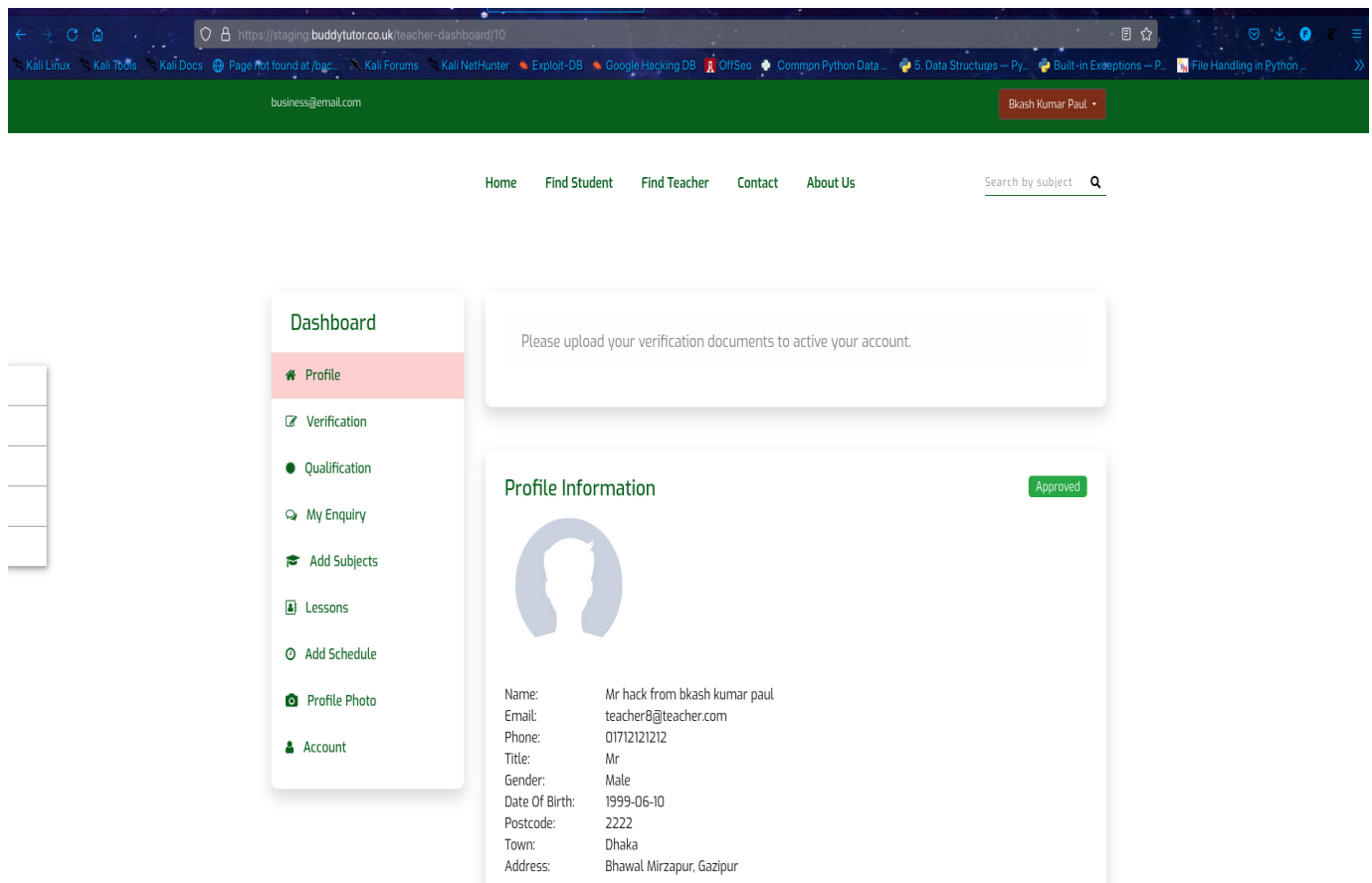


Figure3: Unauthorized access

Impact:

Companies make specific materials private for many reasons, and the unauthorized disclosure of information can happen if they fail to effectively safeguard their content. Employees who can easily access these materials can deliver the information to competitors or hackers or could use it for personal financial gain. Additionally, this unauthorized disclosure can harm a company's reputation and could serve as a black mark on its otherwise sterling operations.

There are several ways businesses can prevent the unauthorized release of confidential materials. Companies can monitor network traffic to identify potential releases of secure information, and they can also use vulnerability assessment tools to constantly update their security systems, thus ensuring that they stay ahead of data breach threats.

Request ^	Position	Payload	Status	Error	Timeout	Length	Comment
0			302	<input type="checkbox"/>	<input type="checkbox"/>	1590	
1	1	12345678	405	<input type="checkbox"/>	<input type="checkbox"/>	649550	
2	1	22kk3333	405	<input type="checkbox"/>	<input type="checkbox"/>	649542	
3	1	ksakkkal	405	<input type="checkbox"/>	<input type="checkbox"/>	649548	
4	1	987654232	405	<input type="checkbox"/>	<input type="checkbox"/>	649555	
5	1	99990000	405	<input type="checkbox"/>	<input type="checkbox"/>	649551	
6	1	7676469978	405	<input type="checkbox"/>	<input type="checkbox"/>	649564	
7	2	12345678	302	<input type="checkbox"/>	<input type="checkbox"/>	1530	
8	2	22kk3333	419	<input type="checkbox"/>	<input type="checkbox"/>	88380	
9	2	ksakkkal	419	<input type="checkbox"/>	<input type="checkbox"/>	88381	
10	2	987654232	419	<input type="checkbox"/>	<input type="checkbox"/>	88374	
11	2	99990000	419	<input type="checkbox"/>	<input type="checkbox"/>	88369	
12	2	7676469978	419	<input type="checkbox"/>	<input type="checkbox"/>	88378	

Figure5: Brute force attack

Impact:

The breach can have far-reaching effects on both users and businesses. They include:

- **Identity theft** – stealing someone's identity to access their accounts, such as bank accounts or credit cards. This enables the attacker to purchase goods using these details. In addition, information such as social security numbers can be sold for use in other cyber attacks.
- **Loss of data** – due to loss of confidentiality if data is stolen which could destroy company reputation. Additionally, there may be reputational damage caused by a leak of sensitive customer information that leads to public distrust and dissatisfaction with the business.
- **Downtime** – this refers to system outages where websites or computer networks cannot be accessed due to a cyber attack. This is costly to the business in terms of lost revenue, customer satisfaction as well as loss of image.

Solution:

Brute force attacks are entirely preventable. You can keep brute force attacks at bay and drastically improve your data security by having a strong password policy, limiting login attempts, enabling two-factor authentication, using CAPTCHAs, and blocking malicious IP addresses.

However, you can further enhance your network security by working with experts. Receiving ongoing IT support from an MSP means that you have people who can help you integrate safe practices, such as using 2FA, and monitor changes in the environment, so you're ready if anything new comes around the corner. Enlisting the help of a managed service provider can take the guesswork out of staying secure in the evolving cyber threat landscape.

3. PII Disclosure

Severity: High

Description:

The response contains Personally Identifiable Information, such as CC number, SSN and similar sensitive data.

Information collect:

Credit Card Type detected: Visa
Bank Identification Number: 440186
Brand: VISA
Category: BUSINESS
Issuer: F BANK

Evidence:



```

</li>
<li class="icon_list">
<a target="_blank" class="icon" style="background-color: "
href="https://api.whatsapp.com/send?phone=4401862206220&fbclid=IwAR20E4Yz4czNZZsx17XUC1C_AtNhvouHu7Ahy7vPF5_c_FwkUXLWzKCMmw">
<i class="fa fa-whatsapp"></i>
<span>Whatsapp</span>
</a>
</li>

```

Figure6: PII Disclosure

Impact:

The potential harm that could result to the subject individuals and/or the organization if

PII were inappropriately accessed, used, or disclosed.

Solution:

Check the response for the potential presence of personally identifiable information (PII), ensure nothing sensitive is leaked by the application.

4. Vulnerable JS Library

Severity : High

Description:

The identified library jquery, version 3.3.1 is vulnerable.

CVE-2020-11023

CVE-2020-11022

CVE-2019-11358

Evidence:



```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>

<link rel="stylesheet" type="text/css"
      href="https://cdnjs.cloudflare.com/ajax/libs/toastr.js/latest/toastr.min.css">

<script src="https://cdnjs.cloudflare.com/ajax/libs/toastr.js/latest/js/toastr.min.js"></script>
```

Figure7: Vulnerable JS Library

Impact:

JavaScript library's security vulnerabilities can be exploited to perform cross-site scripting, cross-site request forgery, and buffer overflow.

Solution:

Please upgrade to the latest version of jquery.

5. Security restrictions bypass in vsftpd

Severity: Medium

Description:

The vulnerability exists due to a logic error in TLS implementation when handling different protocols but using compatible certificates, such as multi-domain or wildcard certificates. A remote attacker with ability to perform TCP/IP layer MitM attack can redirect traffic from one subdomain to another, resulting in a valid TLS session. This breaks the authentication of TLS and cross-protocol attacks may be possible where the behavior of one protocol service may compromise the other at the application layer.

This attack technique was dubbed ALPACA (application layer protocol content confusion attack).

Evidence:

```

$ sudo nmap -sV --script vuln staging.buddytutor.co.uk
[sudo] password for faisal:
Starting Nmap 7.93 ( https://nmap.org ) at 2023-03-20 17:15 +06
Pre-scan script results:
| broadcast-avahi-dos:
|   Discovered hosts:
|     224.0.0.251
|   After NULL UDP avahi packet DoS (CVE-2011-1002).
|_  Hosts are all up (not vulnerable).
Nmap scan report for staging.buddytutor.co.uk (88.208.212.212)
Host is up (0.25s latency).
Not shown: 988 filtered tcp ports (no-response)
PORT      STATE SERVICE VERSION
21/tcp    open  ftp      vsftpd 3.0.3

```

Figure8: Security restrictions bypass in vsftpd

Impact:

The vulnerability allows a remote attacker to bypass implemented security restrictions.

Solution:

Install updates from the vendor's website.

6.1 Remote code execution in OpenSSH client

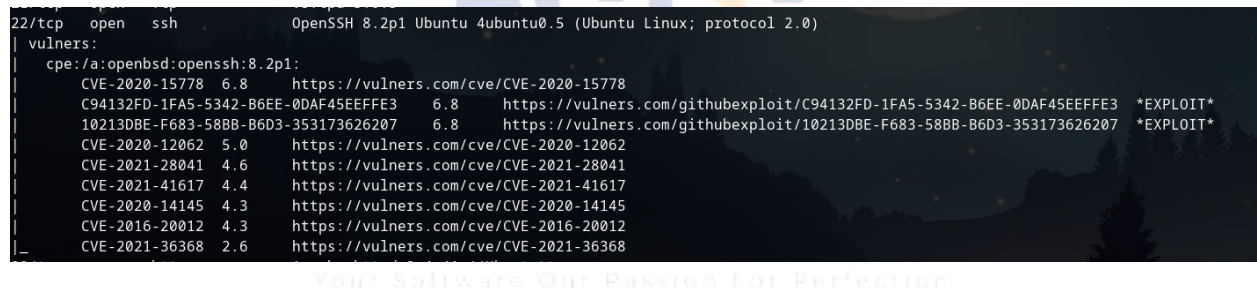
Severity: Medium

Description:

The vulnerability exists due to a boundary error in ssh-agent. A remote attacker can trick the victim to connect to a server, where the attacker has root privileges, pass specially crafted data to the ssh client, trigger a double free error and execute arbitrary code on the target system.

Successful exploitation of this vulnerability may result in complete compromise of the vulnerable system.

Evidence:



```

22/tcp open ssh OpenSSH 8.2p1 Ubuntu 4ubuntu0.5 (Ubuntu Linux; protocol 2.0)
vulners:
cpe:/a:openbsd:openssh:8.2p1:
CVE-2020-15778 6.8 https://vulners.com/cve/CVE-2020-15778
C94132FD-1FA5-5342-B6EE-0DAF45EEFFE3 6.8 https://vulners.com/githubexploit/C94132FD-1FA5-5342-B6EE-0DAF45EEFFE3 *EXPLOIT*
10213DBE-F683-58BB-B6D3-353173626207 6.8 https://vulners.com/githubexploit/10213DBE-F683-58BB-B6D3-353173626207 *EXPLOIT*
CVE-2020-12062 5.0 https://vulners.com/cve/CVE-2020-12062
CVE-2021-28041 4.6 https://vulners.com/cve/CVE-2021-28041
CVE-2021-41617 4.4 https://vulners.com/cve/CVE-2021-41617
CVE-2020-14145 4.3 https://vulners.com/cve/CVE-2020-14145
CVE-2016-20012 4.3 https://vulners.com/cve/CVE-2016-20012
CVE-2021-36368 2.6 https://vulners.com/cve/CVE-2021-36368

```

Figure9: Remote code execution in OpenSSH client

Impact:

The vulnerability allows a remote attacker to execute arbitrary code on the target system.

Solution:

Install updates from vendor's website.

6.2 MitM attack in OpenSSH client

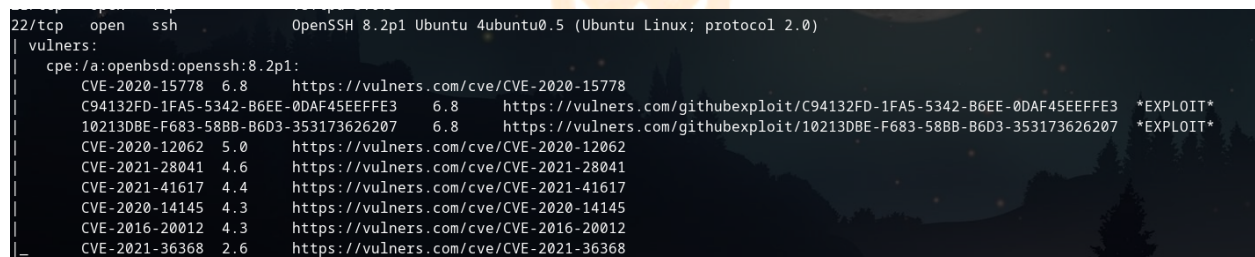
Severity: Medium

Description:

The vulnerability allows a remote attacker to perform MitM attack.

The vulnerability exists in openssh client during algorithm negotiation due to observable discrepancy. A remote attacker can perform a Man-in-the-Middle (MitM) attack.

Evidence:



```

22/tcp open  ssh      OpenSSH 8.2p1 Ubuntu 4ubuntu0.5 (Ubuntu Linux; protocol 2.0)
vulners:
cpe:/a:openbsd:openssh:8.2p1:
CVE-2020-15778 6.8 https://vulners.com/cve/CVE-2020-15778
C94132FD-1FA5-5342-B6EE-0DAF45EEFFE3 6.8 https://vulners.com/githubexploit/C94132FD-1FA5-5342-B6EE-0DAF45EEFFE3 *EXPLOIT*
10213DBE-F683-58BB-B6D3-353173626207 6.8 https://vulners.com/githubexploit/10213DBE-F683-58BB-B6D3-353173626207 *EXPLOIT*
CVE-2020-12062 5.0 https://vulners.com/cve/CVE-2020-12062
CVE-2021-28041 4.6 https://vulners.com/cve/CVE-2021-28041
CVE-2021-41617 4.4 https://vulners.com/cve/CVE-2021-41617
CVE-2020-14145 4.3 https://vulners.com/cve/CVE-2020-14145
CVE-2016-20012 4.3 https://vulners.com/cve/CVE-2016-20012
CVE-2021-36368 2.6 https://vulners.com/cve/CVE-2021-36368

```

Figure10: MitM attack in OpenSSH client

Impact:

A man-in-the-middle attack may permit the attacker to completely subvert encryption and gain access to the encrypted contents, including passwords. A successful attacker is able to inject commands into terminal session, to modify data in transit, or to steal data.

Solution:

Cybersecurity Help is currently unaware of any official solution to address this vulnerability.

Partial mitigation against this issue was included in OpenSSH 8.4

7. Absence of Anti-CSRF Token

Severity: Medium

Description:

No Anti-CSRF tokens were found in a HTML submission form.

A cross-site request forgery is an attack that involves forcing a victim to send an HTTP request to a target destination without their knowledge or intent in order to perform an action as the victim. The underlying cause is application functionality using predictable URL/form actions in a repeatable way. The nature of the attack is that CSRF exploits the trust that a website has for a user. By contrast, cross-site scripting (XSS) exploits the trust that a user has for a web site. Like XSS, CSRF attacks are not necessarily cross-site, but they can be. Cross-site request forgery is also known as CSRF, XSRF, one-click attack, session riding, confused deputy, and sea surf.

Your Software Our Passion For Perfection

CSRF attacks are effective in a number of situations, including:

- * The victim has an active session on the target site.
- * The victim is authenticated via HTTP auth on the target site.
- * The victim is on the same local network as the target site.

CSRF has primarily been used to perform an action against a target site using the victim's privileges, but recent techniques have been discovered to disclose information by gaining access to the response. The risk of information disclosure is dramatically

increased when the target site is vulnerable to XSS, because XSS can be used as a platform for CSRF, allowing the attack to operate within the bounds of the same-origin policy.

Evidence:

```
<div class="col-lg-2 position-static ms-auto">
  <div class="header-search">
    <form action="https://staging.buddytutor.co.uk/find-teachers">
      <input type="text" placeholder="Search by subject" name="subject_search" value="">
      <button><i class="fas fa-search"></i></button>
    </form>
  </div>
```

Figure11: Absence of Anti-CSRF Token

Impact:

The absence of Anti-CSRF tokens may lead to a Cross-Site Request Forgery attack that can result in executing a specific application action as another logged in user, e.g. steal their account by changing their email and password or silently adding a new admin user account when executed from the administrator account.

Solution:

Phase: Architecture and Design

Use a vetted library or framework that does not allow this weakness to occur or provides constructs that make this weakness easier to avoid.

For example, use anti-CSRF packages such as the OWASP CSRFGuard.

Phase: Implementation

Ensure that your application is free of cross-site scripting issues, because most CSRF defenses can be bypassed using attacker-controlled script.

Phase: Architecture and Design

Generate a unique nonce for each form, place the nonce into the form, and verify the nonce upon receipt of the form. Be sure that the nonce is not predictable (CWE-330).

Note that this can be bypassed using XSS.

Identify especially dangerous operations. When the user performs a dangerous operation, send a separate confirmation request to ensure that the user intended to perform that operation.

Note that this can be bypassed using XSS.

Use the ESAPI Session Management control.

This control includes a component for CSRF.

Do not use the GET method for any request that triggers a state change.

Phase: Implementation

Check the HTTP Referer header to see if the request originated from an expected page. This could break legitimate functionality, because users or proxies may have disabled sending the Referrer for privacy reasons.

8. Content Security Policy (CSP) Header Not Set

Severity: Medium

Description:

Content Security Policy (CSP) is an added layer of security that helps to detect and mitigate certain types of attacks, including Cross Site Scripting (XSS) and data injection attacks. These attacks are used for everything from data theft to site defacement or distribution of malware. CSP provides a set of standard HTTP headers that allow website owners to declare approved sources of content that browsers should be allowed to load on that page — covered types are JavaScript, CSS, HTML frames, fonts, images and embeddable objects such as Java applets, ActiveX, audio and video files.

Evidence:

```
HTTP/1.1 200 OK
Date: Thu, 23 Mar 2023 09:52:57 GMT
Server: Apache/2.4.41 (Ubuntu)
Cache-Control: no-cache, private
Set-Cookie: XSRF-TOKEN=
eyJpdii6Inh5MmFzMGNMZW2yaHgyVE9rWVl3aWc9PSIsInZhbnVlIjoibFhBRzRlZU9EdkJKdZNRMD0x3WmNjaWRINzFwZlJ5RTRQR2Q5ZTZib0ZmcDRTOFk0R2F1RzV
6RZlad2Q2VWVlSHhTYXhvODZrVDlRWhwbHJSbDdxcU9LUgPzVXE5WGFYRnB5aFFjSDVmb1hJK2lreFNlY2VYVC9nN2xiVWlySmYiLCJtYWMiOiIyZmY1ODQ3ZWVjN2
ZiYThiYjZlNjF1ZDZmNTVmZjllNTE1MjIyZWVkdHh1MjIxYjY0OTcwOWI1NWZhMzc2ZTk2IiwidGFuIjoiiIn0%3D; expires=Thu, 23-Mar-2023 11:52:57 GMT
; Max-Age=7200; path=/; samesite=lax
Set-Cookie: staging_buddy_t_session=
eyJpdii6Ikw3dGNEU3FidjFZVlRVWkhsV3F0cVE9PSIsInZhbnVlIjoik3NkTjVFcTVRRVo0VmZoQk1RR29jMVlSM1E3dE9rQTJHbE94ZG9Rb25iN2p0RVgwb0RqawJ
nVHU5OC9yaHE1N3hMNU1QNTM2Y1hCKzR3THZUWXZleEFUMTljR3pNU1NwTHZJZXdkb0RwVjJLY0l0dzlsSfdNUy9SWTF5aDRtMk8iLCJtYWMiOiIwODhmOWFjZTc2Ym
FmNjc3OGUxODg4ZjIjNTRjYmE1ZjY4NDk4Mjk3ZDdmM2Y4ZTJlMDIzNDM5ZmFiZjBmZTQzIiwidGFuIjoiiIn0%3D; expires=Thu, 23-Mar-2023 11:52:57 GMT
; Max-Age=7200; path=/; httponly; samesite=lax
Vary: Accept-Encoding
Content-Type: text/html; charset=UTF-8
Content-Length: 102451
```

Figure12: Content Security Policy (CSP) Header Not Set

```
<!--===== Title =====>
<title>Buddy-Tutor</title>

<meta name="description" content="">
<meta name="viewport" content="width=device-width, initial-scale=1">

<!--===== Favicon Icon =====>
<link rel="shortcut icon" href="https://staging.buddytutor.co.uk/assets/images/favicon.png" type="image/png">
```

Figure13: Content Security Policy (CSP) Header Not Set

Impact:

There is no direct impact of not implementing CSP on your website. However, if your website is vulnerable to a Cross-site Scripting attack CSP can prevent successful exploitation of that vulnerability. By not implementing CSP you'll be missing out on this extra layer of security.

Solution:

Ensure that your web server, application server, load balancer, etc. is configured to set the Content-Security-Policy header.

9. Cross-Domain Misconfiguration

Severity: Medium

Description:

Web browser data loading may be possible, due to a Cross Origin Resource Sharing (CORS) misconfiguration on the web server

The CORS misconfiguration on the web server permits cross-domain read requests from arbitrary third party domains, using unauthenticated APIs on this domain. Web browser implementations do not permit arbitrary third parties to read the response from authenticated APIs, however. This reduces the risk somewhat. This misconfiguration could be used by an attacker to access data that is available in an unauthenticated manner, but which uses some other form of security, such as IP address white-listing.

Evidence:

```
HTTP/1.1 200 OK
Date: Thu, 23 Mar 2023 09:54:23 GMT
Content-Type: application/javascript; charset=utf-8
Connection: keep-alive
Access-Control-Allow-Origin: *
Cache-Control: public, max-age=30672000
ETag: W/"5eb03ffe-15a1"
Last-Modified: Mon, 04 May 2020 16:17:02 GMT
cf-cdnjs-via: cfworker/kv
Cross-Origin-Resource-Policy: cross-origin
Timing-Allow-Origin: *
X-Content-Type-Options: nosniff
CF-Cache-Status: HIT
Age: 879658
Expires: Tue, 12 Mar 2024 09:54:23 GMT
Report-To: {"endpoints":[{"url":"https://a.nel.cloudflare.com/v3/7s5ADXAjgP9ftfJCg12BtbEEG8xY9p612PcGHLm8u1IZdeADyMuk7JF6NKz%2B1CprpP8rjWw7IR4gWpW89E5Gf0YGWufKH1Hutp2BuIrtTHHe%2Bxv2ayRRU70aq0%2Bv74jTL01ysE"}], "group": "cf-nel", "max_age": 604800}
NEL: {"success_fraction": 0.01, "report_to": "cf-nel", "max_age": 604800}
Strict-Transport-Security: max-age=15780000
Server: cloudflare
CF-RAY: 7ac5caf17b6e84cb-BOM
alt-svc: h3=":443"; ma=86400, h3-29=":443"; ma=86400
Content-Length: 5537
```

Figure14: Cross-Domain Misconfiguration

Impact:

Manipulation or theft of the victim's cookies. Creation and execution of invalid requests. Execution of malicious code within the vulnerable web server.

Solution:

Ensure that sensitive data is not available in an unauthenticated manner (using IP address white-listing, for instance).

Configure the "Access-Control-Allow-Origin" HTTP header to a more restrictive set of domains, or remove all CORS headers entirely, to allow the web browser to enforce the Same Origin Policy (SOP) in a more restrictive manner.

10. HTTP to HTTPS Insecure Transition in Form Post

Severity: Medium

Description:

This check looks for insecure HTTP pages that host HTTPS forms. The issue is that an insecure HTTP page can easily be hijacked through MITM and the secure HTTPS form can be replaced or spoofed.

The response to the following request over HTTP included an HTTPS form tag action attribute value:

http://staging.buddytutor.co.ukThe context was:

```
<form action="https://staging.buddytutor.co.uk/find-teachers">
```

```
    <input    type="text"    placeholder="Search    by    subject"
name="subject_search" value="">
```

```
    <button><i class="fas fa-search"></i></button>
```

```
</form>
```

Evidence:

```
<div class="header-search">
  <form action="https://staging.buddytutor.co.uk/find-teachers">
    <input type="text" placeholder="Search by subject" name="subject_search" value="">
    <button<i class="fas fa-search"></i></button>
  </form>
```

Figure15: HTTP to HTTPS Insecure Transition in Form Post

```
HTTP/1.1 200 OK
Date: Mon, 20 Mar 2023 12:17:48 GMT
Server: Apache/2.4.41 (Ubuntu)
Cache-Control: no-cache, private
Set-Cookie: XSRF-TOKEN=
eyJpdiI6IjNzRk9IUS93UFQ3NWwKZExQSm1PZmc9PSIsInZhbnV1IjoisIpiUnpZZ29STWdWNEI0bWk0TDJoV2tKWGFyYW9SUHJNekt0eEhua0JnTTJxeEhWSThwZWFlbEZkR3JLdFN3MHRxN0VaeFk3d1VnT0ZRU9mb
m4wN2E1NVVwV1lQZi8xczF3SXFFTFp1Y0c3ejVnZRRRDFZSWRka1BBZG82YWciLCJtYWMI0iI3Y2M1NWFnMWY4ODg1NjF1ZTAyYjRlMmQyY2U3MDg0ZjQ0ZjE3MzUxMzRmZjY1MjB1OWJmNWISYzY2YjY0NDUxIiwidG
FnIjoIn0%3D; expires=Mon, 20-Mar-2023 14:17:48 GMT; Max-Age=7200; path=/; samesite=lax
Set-Cookie: staging_buddy_t_session=
eyJpdiI6IjNzRk9IUS93UFQ3NWwKZExQSm1PZmc9PSIsInZhbnV1IjoisIpiUnpZZ29STWdWNEI0bWk0TDJoV2tKWGFyYW9SUHJNekt0eEhua0JnTTJxeEhWSThwZWFlbEZkR3JLdFN3MHRxN0VaeFk3d1VnT0ZRU9mb
1JRSndCRHV6UFVWNEtoQVkyN2JRS0wyNkMvbl1FU0JKb0N4TWZlU3hSN0xvSmsiLCJtYWMI0iIyMTgxNDZjZTE3ZGUxYzNlODEMDk0YWZlZDh1N2QyOGI1NzkyZTE2NDUyMGFiMTF1MWE0NDE3ZTM2Y2YxYWJjIiwidG
FnIjoIn0%3D; expires=Mon, 20-Mar-2023 14:17:48 GMT; Max-Age=7200; path=/; httponly; samesite=lax
Vary: Accept-Encoding
Content-Type: text/html; charset=UTF-8
Content-Length: 98430
```

Figure15: HTTP to HTTPS Insecure Transition in Form Post

Impact:

This check looks for insecure HTTP pages that host HTTPS forms. The issue is that an insecure HTTP page can easily be hijacked through MITM and the secure HTTPS form can be replaced or spoofed.

Solution:

Use HTTPS for landing pages that host secure forms.

Solution:

Modern Web browsers support the Content-Security-Policy and X-Frame-Options HTTP headers. Ensure one of them is set on all web pages returned by your site/app. If you expect the page to be framed only by pages on your server (e.g. it's part of a FRAMESET) then you'll want to use SAMEORIGIN, otherwise if you never expect the page to be framed, you should use DENY. Alternatively consider implementing Content Security Policy's "frame-ancestors" directive.

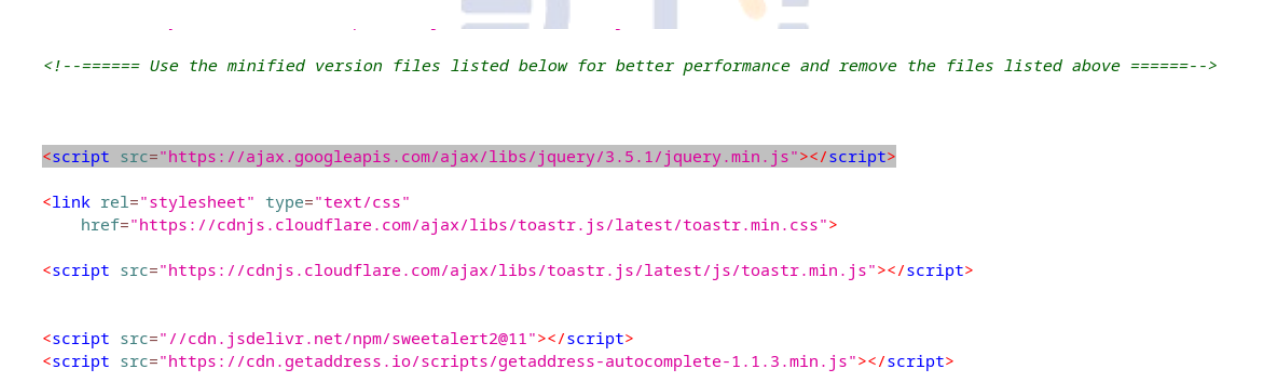
12. Cross-Domain JavaScript Source File Inclusion

Severity: Low

Description:

The page includes one or more script files from a third-party domain.

Evidence:



```
<!--===== Use the minified version files listed below for better performance and remove the files listed above =====-->

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>

<link rel="stylesheet" type="text/css"
      href="https://cdnjs.cloudflare.com/ajax/libs/toastr.js/latest/toastr.min.css">

<script src="https://cdnjs.cloudflare.com/ajax/libs/toastr.js/latest/js/toastr.min.js"></script>

<script src="//cdn.jsdelivr.net/npm/sweetalert2@11"></script>
<script src="https://cdn.getaddress.io/scripts/getaddress-autocomplete-1.1.3.min.js"></script>
```

Figure17: Cross-Domain JavaScript Source File Inclusion

Impact:

It can lead to the leakage of user data. Sensitive user data can be user's authentication data (tokens, session IDs, cookies, etc) or personal information (email, home address, phone numbers, social security numbers, etc)

Solution:

Ensure JavaScript source files are loaded from only trusted sources, and the sources can't be controlled by end users of the application.

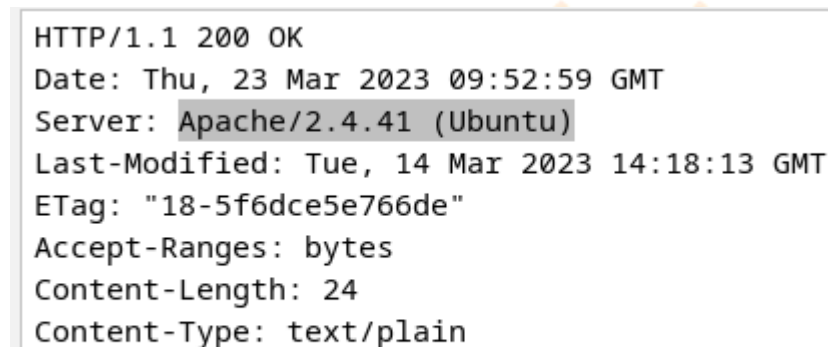
13. Server Leaks Version Information via "Server" HTTP Response Header Field.

Severity: Low

Description:

The web/application server is leaking version information via the "Server" HTTP response header. Access to such information may facilitate attackers identifying other vulnerabilities your web/application server is subject to.

Evidence:



```
HTTP/1.1 200 OK
Date: Thu, 23 Mar 2023 09:52:59 GMT
Server: Apache/2.4.41 (Ubuntu)
Last-Modified: Tue, 14 Mar 2023 14:18:13 GMT
ETag: "18-5f6dce5e766de"
Accept-Ranges: bytes
Content-Length: 24
Content-Type: text/plain
```

Figure18: Server Leaks Version Information via "Server" HTTP Response Header Field.

Impact:

Why "Server Leaks Version Information via "Server" HTTP Response Header Field" can be dangerous. If your application leaks web server version details via "Server" HTTP response header field the attacker may use it to find and exploit security vulnerabilities present specifically in the reported web server information

Solution:

Ensure that your web server, application server, load balancer, etc. is configured to suppress the "Server" header or provide generic Description.

14. Strict-Transport-Security Header Not Set

Severity : Low

Description:

HTTP Strict Transport Security (HSTS) is a web security policy mechanism whereby a web server declares that complying user agents (such as a web browser) are to interact with it using only secure HTTPS connections (i.e. HTTP layered over TLS/SSL). HSTS is an IETF standard track protocol and is specified in RFC 6797.

Evidence:

```
HTTP/1.1 200 OK
Accept-Ranges: bytes
Access-Control-Allow-Origin: *
Content-Security-Policy-Report-Only: require-trusted-types-for 'script'; report-uri https://csp.withgoogle.com/csp/hosted-libraries-pushers
Cross-Origin-Resource-Policy: cross-origin
Cross-Origin-Opener-Policy: same-origin; report-to="hosted-libraries-pushers"
Report-To: {"group": "hosted-libraries-pushers", "max_age": 2592000, "endpoints": [{"url": "https://csp.withgoogle.com/csp/report-to/hosted-libraries-pushers"}]}
Timing-Allow-Origin: *
Content-Length: 89476
X-Content-Type-Options: nosniff
Server: sffe
X-XSS-Protection: 0
Date: Mon, 20 Mar 2023 16:53:04 GMT
Expires: Tue, 19 Mar 2024 16:53:04 GMT
Cache-Control: public, max-age=31536000, stale-while-revalidate=2592000
Last-Modified: Fri, 08 May 2020 07:05:03 GMT
Content-Type: text/javascript; charset=UTF-8
Vary: Accept-Encoding
Age: 234079
Alt-Svc: h3=":443"; ma=2592000, h3-29=":443"; ma=2592000
```



Figure19: Strict-Transport-Security Header Not Set

```
HTTP/1.1 200 OK
...
/*! jQuery v3.5.1 | (c) JS Foundation and other contributors | jquery.org/license */
!function(e,t){var strict="";object==typeof module&&"object"==typeof module.exports?module.exports=e.document?t(e,!0):function(e){if(!e.document)throw new Error("jQuery requires a window with a document");return e}:function(e){return e.nodeName&&e.nodeType===1}.CLASS=function(e){var t=[e];return t[0]=new RegExp("(^|"+M+"")+"e"+"(")+M+"($)"))&&(e,function(e){return t.test("string"==typeof e?e:a.apply(n,r))==o.promise()})throw new TypeError("Theenable self-resolution");t=e&&"object"==typeof e||"function"==typeof e}&&e.then,m(t)?t.call(e,l(u,o,R,s),l(u,o,M,s)):u++,t.call(e,l(u,o,R,s),l(u,o,M,s)).extend(this,t),this.timestamp=e&&e.timestamp||Date.now(),this[5.expando]=!0,S.Event.prototype=(constructor:S.Event,isDefaultPrevented:Ee,isPropagationStopped:Ee,isImmediatePropagationStopped:Ee,length:0;if(a)return this;for(a=!0;t<n;t++)l tweens[t].run(1);return e?(s.notifyWith(o,[1,1,0]),s.resolveWith(o,[1,e]),s.rejectWith(o,[1,e]),this)),c=l.props;for(function(e,t){var n,e|res|widget)}$/-test(Tt.protocol),global:!0,processData:!0,async:!0,contenttype:"application/x-www-form-urlencoded; charset=UTF-8",accepts:{"*":!t,text:"text/plain",html:"text/html",xml:"application
```

Figure20: Strict-Transport-Security Header Not Set

Impact:

Why “Strict-Transport-Security Header Not Set” can be dangerous. The missing Strict-Transport-Security header results in communication over HTTP being allowed to the specified domain. That makes the website vulnerable to man-in-the-middle attacks, presenting a fake login page being one of the options.

Solution:

Ensure that your web server, application server, load balancer, etc. is configured to enforce Strict-Transport-Security.

15. Timestamp Disclosure - Unix

Severity: Low

Description:

A timestamp was disclosed by the application/web server - Unix

Other information:

1675524420, which evaluates to: 2023-02-04 21:27:00

Evidence:

```
<link rel='stylesheet' type='text/css' property='stylesheet' href='//staging.buddytutor.co.uk/_debugbar/assets/stylesheets?v=1675524420&theme=auto' data-turbolinks-eval='false' data-turbo-eval='false'>
<script> Sfdump = window.Sfdump || (function (doc) { var refStyle = doc.createElement('style'), rxEsc = /([.*+?${}()|\[\]\/\\])/g, idRx = /\bsf-dump-\d+-ref[012]\w+\b/, keyHint = 0 <= navigator.pla
border-bottom-left-radius: 3px; color: #000; min-width: 15px; width: 100%; } .phpdebugbar pre.sf-dump .sf-dump-search-wrapper > .sf-dump-search-input-next, .phpdebugbar pre.sf-dump .sf-dump-search-i
</head>
```

Figure21: Timestamp Disclosure - Unix

Impact:

A timestamp disclosed by the application server or web server can be used to retrieve other sensitive information e.g. when used as a salt or a token during authentication or encryption.

Typically a timestamp is disclosed as a Unix epoch time, see https://en.wikipedia.org/wiki/Unix_time, e.g. 1594200097 represents Wednesday July 8 2020 09:21:37 GMT.

If the server timestamp is used e.g. as a salt to hash specific sensitive information (authentication code, password, anti-CSRF token) the attacker can retrieve it from the server and synchronize the local attacking code to minimize the number of brute force attempts required to reproduce the result of the application hashing algorithm.

Solution:

Manually confirm that the timestamp data is not sensitive, and that the data cannot be aggregated to disclose exploitable patterns.

16. X-Content-Type-Options Header Missing

Severity: Low

Description:

The Anti-MIME-Sniffing header X-Content-Type-Options was not set to 'nosniff'. This allows older versions of Internet Explorer and Chrome to perform MIME-sniffing on the response body, potentially causing the response body to be interpreted and displayed as a content type other than the declared content type. Current (early 2014) and legacy versions of Firefox will use the declared content type (if one is set), rather than performing MIME-sniffing.

This issue still applies to error type pages (401, 403, 500, etc.) as those pages are often still affected by injection issues, in which case there is still concern for browsers sniffing pages away from their actual content type.

At "High" threshold this scan rule will not alert on client or server error responses.

Evidence:

```
HTTP/1.1 200 OK
Content-Length: 30516
Content-Type: application/javascript
Content-MD5: jxFMkT0kK9zqJKx1GPh8Ig==
Last-Modified: Mon, 06 Jun 2022 09:57:03 GMT
ETag: 0x8DA47A2E8A3F831
X-Cache: TCP_MISS
x-ms-request-id: 54544f45-601e-004b-436d-5dc5d0000000
x-ms-version: 2009-09-19
x-ms-lease-status: unlocked
x-ms-blob-type: BlockBlob
X-Azure-Ref-OriginShield: 00CEcZAAAAABjhZAKiY1fTLvBUok56wgrU0lOMjIxMDGwNzE4MDI3ADRhZjQyZWZlLTJjYTQtNDhlMy1hMzgwLTl1ZTg5ZmViYzNlNg==
X-Azure-Ref: 00CEcZAAAAADIX1psv0JZSJ1BSTPIib8IU0lOMzBFREdFMDIxNQAOYWY0MmVmZS0yY2E0LTQ4ZTMtYTM4MCO5YmU4OWZlYmMzZTY=
Date: Thu, 23 Mar 2023 09:54:24 GMT
```

Figure22: X-Content-Type-Options Header Missing

Impact:

The HTTP 'X-Content-Type-Options' response header prevents the browser from MIME-sniffing a response away from the declared content-type. The server did not return a correct 'X-Content-Type-Options' header, which means that this website could be at risk of a Cross-Site Scripting (XSS) attack.

Solution:

Ensure that the application/web server sets the Content-Type header appropriately, and that it sets the X-Content-Type-Options header to 'nosniff' for all web pages.

If possible, ensure that the end user uses a standards-compliant and modern web browser that does not perform MIME-sniffing at all, or that can be directed by the web application/web server to not perform MIME-sniffing.

17. Big Redirect Detected (Potential Sensitive Information Leak)

Severity: Low

Description:

The server has responded with a redirect that seems to provide a large response. This may indicate that although the server sent a redirect it also responded with body content (which may include sensitive Description, PII, etc.).

Other information

Location header URI length: 40 [https://staging.buddytutor.co.uk/contact].

Predicted response size: 340.

Response Body Length: 406.

Evidence:

```
HTTP/1.1 302 Found
Date: Thu, 23 Mar 2023 09:53:04 GMT
Server: Apache/2.4.41 (Ubuntu)
Cache-Control: no-cache, private
Location: https://staging.buddytutor.co.uk/contact
Set-Cookie: XSRF-TOKEN=eyJpdjI6Im14b2syeEhaWVU1cHVHVSj1Nmd0anc9PSIsInZhbHVlIjo1UVB0Qm01RzdWmkfaeDFQT1B3bE045zBUTi9HL2VmqZdnQ112dn1GQ1FUbEMyeGxwVHN6OWpizNjdTJoenB0eGk0VHZObDVkK9rcC9JV3HbXRhcnM1WStZb040SkF1cmN0TlEzdXptOTIqaktanNnRjdW0yQ0IzZDp0MFB1YUhlLCJtYVM1O1IwODkxZmEyZWZlYzY2YmRlMGNYj1kNGU3Nj10ZDc4NmV1OTNjM2QzYjdhM2k2KnJhYWI1NDY2MGViNzkyOGVjIiwidGFnIjo1In0%3D; expires=Thu, 23-Mar-2023 11:53:04 GMT; Max-Age=7200; path=/; samesite=lax
Set-Cookie: staging_buddy_t_session=eyJpdjI6Im14b2syeEhaWVU1cHVHVSj1Nmd0anc9PSIsInZhbHVlIjo1Q1NsK21MNDh5UjFlVWVpNSG12c1lG2ZVmdE51WRWVVGXTmNBuY9SUENK7z8Jv1E1cE1DM1E5N2RNSFDM0F02SDJz5nRRR11Rd3dqQjK15CtCZFP651ZuTXR1c0LzQytkMlcaE96Tk9avGpxY3A2blh0c0ZzXUza0BJhFRlYnAILCJtYVM1O1I4NTU4ZTYwYj1jNjE1ZjY4MTM2ZDEzOHQwMzZlZGZyYjM3YTNlYTI1ZmJhNmVhN2U1ZD13NGRlZjkwNzI5YTQyIiwidGFnIjo1In0%3D; expires=Thu, 23-Mar-2023 11:53:04 GMT; Max-Age=7200; path=/; httponly; samesite=lax
Content-Type: text/html; charset=UTF-8
Content-Length: 406
```

Figure23: Big Redirect Detected (Potential Sensitive Information Leak)

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="refresh" content="0;url='https://staging.buddytutor.co.uk/contact' " />
    <title>Redirecting to https://staging.buddytutor.co.uk/contact</title>
  </head>
  <body>
    Redirecting to <a href="https://staging.buddytutor.co.uk/contact">https://staging.buddytutor.co.uk/contact</a>.
  </body>
</html>
```

Figure23: Big Redirect Detected (Potential Sensitive Information Leak)

Impact:

Why Big Redirect Detected (Potential Sensitive Information Leak) can be dangerous. This means that the server responds with a redirect which seems to provide a response

larger than it is expected to be. This may indicate that although the server sent a redirect in a response it also attached a body content to it.

Solution:

Ensure that no sensitive information is leaked via redirect responses. Redirect responses should have almost no content.

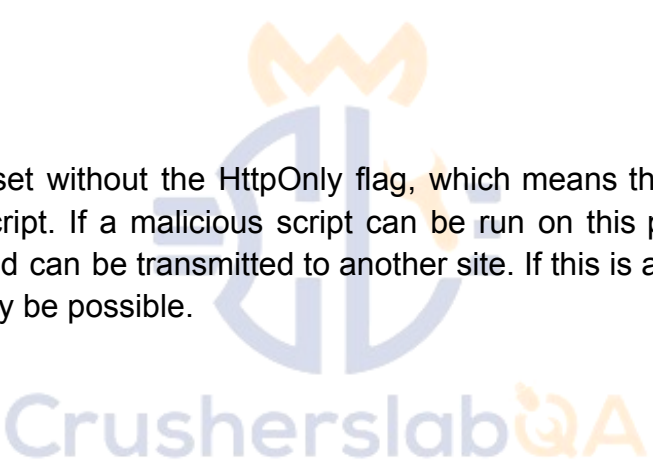
18. Cookie No HttpOnly Flag

Severity: Low

Description:

A cookie has been set without the HttpOnly flag, which means that the cookie can be accessed by JavaScript. If a malicious script can be run on this page then the cookie will be accessible and can be transmitted to another site. If this is a session cookie then session hijacking may be possible.

Evidence:



```
HTTP/1.1 200 OK
Date: Thu, 23 Mar 2023 09:52:57 GMT
Server: Apache/2.4.41 (Ubuntu)
Cache-Control: no-cache, private
Set-Cookie: XSRF-TOKEN=eyJpdiI6InhsMmFzMGNMZ2tyaGgyeV9tWV13awc9PSIsInZhbHVlIjoibFhBRzRIZU9EdkJmdzNRM0x3NmMjaWRlbnZwZlJ5RTQRQ2Q5ZTZlZmZmcDRlOFk0R2F1RzV6Rzlad2Q2VWV1SHhTYXhvODZrVDlnRWhwBHJScDdxcU9LUgpZVXESWGFYRnB5aFFjSDVmb1hJK2lreFNlY2VYVC9nN2x1VnlySmY1LCJtyVMiOiIyZmY1ODQ3ZWVjN2ZlYThiYjZlNjF1ZDZmNTVmZjllNTE1MjIyZWVhMDh1MjIyYjY0OTcwOWI1NWZlMzc2ZTk2IiwidGFuIjoiaWln0%3D"; expires=Thu, 23-Mar-2023 11:52:57 GMT; Max-Age=7200; path=/; samesite=lax
Set-Cookie: staging_buddy_t_session=eyJpdiI6Ikw3dGNEU3FidjF2VlRvWkhsV3F0cVE9PSIsInZhbHVlIjoik3NKtjVFcTVRRVo0VmZ0Qk1RR29jMV1SM1E3dE9xQTJHbE94ZG9Bb25lN2p0RVgwb0RqaWJnVHUSOC9yaHE1N3hMNUIQNTM2Y1hCKzR3THZUWXZleEFUMTljR3pNU1NwTHZJZXdkb0RwVjJlY010dz1sSFdNUy9SMTF5aDRlMk81LCJtyVMiOiIwODhmOWFjZTc2YmFmNjc3OGUxODg4ZjZlNTRjYmE1ZjY4NDk4Mjk3ZDdmM2Y4ZTJlMDIzNDM5ZWFiZjBmZTQ2IiwidGFuIjoiaWln0%3D"; expires=Thu, 23-Mar-2023 11:52:57 GMT; Max-Age=7200; path=/; httponly; samesite=lax
Vary: Accept-Encoding
Content-Type: text/html; charset=UTF-8
Content-Length: 102451
```

Figure24: Cookie No HttpOnly Flag

Impact:

This measure makes certain client-side attacks, such as cross-site scripting, slightly harder to exploit by preventing them from trivially capturing the cookie's value via an injected script.

Solution:

Whenever a cookie contains sensitive information or is a session token, then it should always be passed using an encrypted channel. Ensure that the secure flag is set for cookies containing such sensitive information.

