Security Assessment Findings Report

FortifyTech

Date: May 8th, 2024

Confidentiality Statement

This document is the exclusive property of FortifyTech and CyberShield Security (CSS). It contains proprietary and confidential information that must not be disclosed, duplicated, or utilized, in whole or in part, in any form, without the consent of both FortifyTech and CSS.

FortifyTech may share this document with auditors under non-disclosure agreements to demonstrate compliance with penetration testing requirements.

Disclaimer

A penetration test is considered a snapshot in time. The findings and recommendations reflect the information gathered during the assessment and not any changes or modifications made outside of that period.

Time-limited engagements do not allow for a full evaluation of all security controls. Practitioner prioritized the assessment to identify the weakest security controls an attacker would exploit. TCMS recommends conducting similar assessments on an annual basis by internal or third-party assessors to ensure the continued success of the controls.

Contact Information

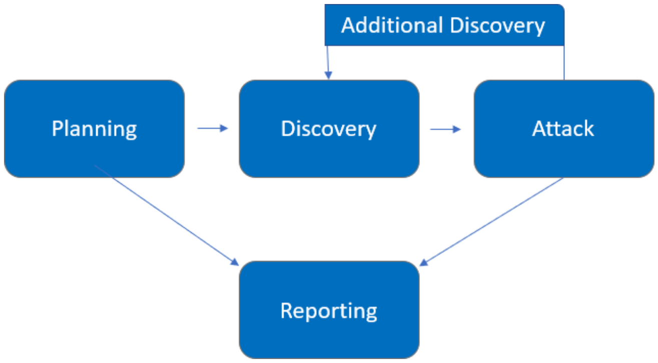
Name	Title	Contact Information
Information Technology -ITS		
Salsabila Amalia Harjanto	Ethical Hacking Practitioner	NRP 5027221023 Email: harjantosalsabila@gmail.com

Assessment Overview

From May 5th, 2019 to May 7th, 2024, FortifyTech engaged CyberShield Security (CSS) to evaluate the security posture of its infrastructure compared to industry best practices outlined in Module 4-6. This evaluation included an external penetration test. All testing conducted was based on customized testing frameworks derived from Module 4-6 guidelines.

Phases of penetration testing activities include the following:

- Planning – Customer goals are gathered and rules of engagement obtained.
- Discovery – Perform scanning and enumeration to identify potential vulnerabilities, weak areas, and exploits.
- Attack – Confirm potential vulnerabilities through exploitation and perform additional discovery upon new access.
- Reporting – Document all found vulnerabilities and exploits, failed attempts, and company strengths and weaknesses.



Finding SeverityRatings

The following table defines levels of severity and corresponding CVSS score range that are used throughout the document to assess vulnerability and risk impact.

Severity	CVSS V3 Score Range	Definition
Critical	9.0-10.0	Exploitation is straightforward and usually results in system-level compromise. It is advised to form a plan of action and patch immediately.
High	7.0-8.9	Exploitation is more difficult but could cause elevated privileges and potentially a loss of data or downtime. It is advised to form a plan of action and patch as soon as possible.
Medium	4.0-6.9	Vulnerabilities exist but are not exploitable or require extra steps such as social engineering. It is advised to form a plan of action and patch after high-priority issues have been resolved.
Low	0.1-3.9	Vulnerabilities are non-exploitable but would reduce an organization’s attack surface. It is advised to form a plan of action and patch during the next maintenance window.
Information al	N/A	No vulnerability exists. Additional information is provided regarding items noticed during testing, strong controls, and additional documentation.

Risk Factors

Risk is measured by two factors: Likelihood and Impact:

Likelihood

Likelihood measures the potential of a vulnerability being exploited. Ratings are given based on the difficulty of the attack, the available tools, attacker skill level, and client environment.

Impact

Impact measures the potential vulnerability’s effect on operations, including confidentiality, integrity, and availability of client systems and/or data, reputational harm, and financial loss.

Scope

Assesment	Details
External Penetration Test	10.15.42.36 10.15.42.7

Scope Exclusions

Per client request, the practitioner did not perform any illegal activities during testing.

Client Allowances

Client did not provide any allowances to assist the testing.

Executive Summary

The practitioner evaluated DC's external security posture through an external network penetration test conducted from May 5th, 2024 to May 7th, 2024. By employing a series of reconnaissance methods, the practitioner discovered vulnerabilities ranging from low to medium severity that grant access to the target IP. It is strongly advised that DC promptly addresses these vulnerabilities, as they can be easily identified through basic reconnaissance and exploited with minimal effort.

Step	Action	Recommendation
1	Successfully obtained backup.sql, containing an SQL database export with admin username and password, via FTP at 10.15.42.36	Disable standard FTP and switch to FTPS or SFTP. Upgrade SSH software to utilize robust ciphers and the latest TLS version (avoid SSL). If feasible, deactivate anonymous FTP access.
2	Nuclei scan to get WordPress Username Enumeration	To prevent attackers from enumerating WordPress usernames, install and activate the "Unified Login Error Messages" WordPress plugin.
3	Nuclei scan to get Vulnerable to Terrapin (CVE-2023-48795)	Disable the vulnerable ChaCha20-Poly1305 cipher in the OpenSSH client and server configurations.
4	Zap scan to get more information about 10.15.42.7 vulnerability	-

Vulnerability Summary & Report Card

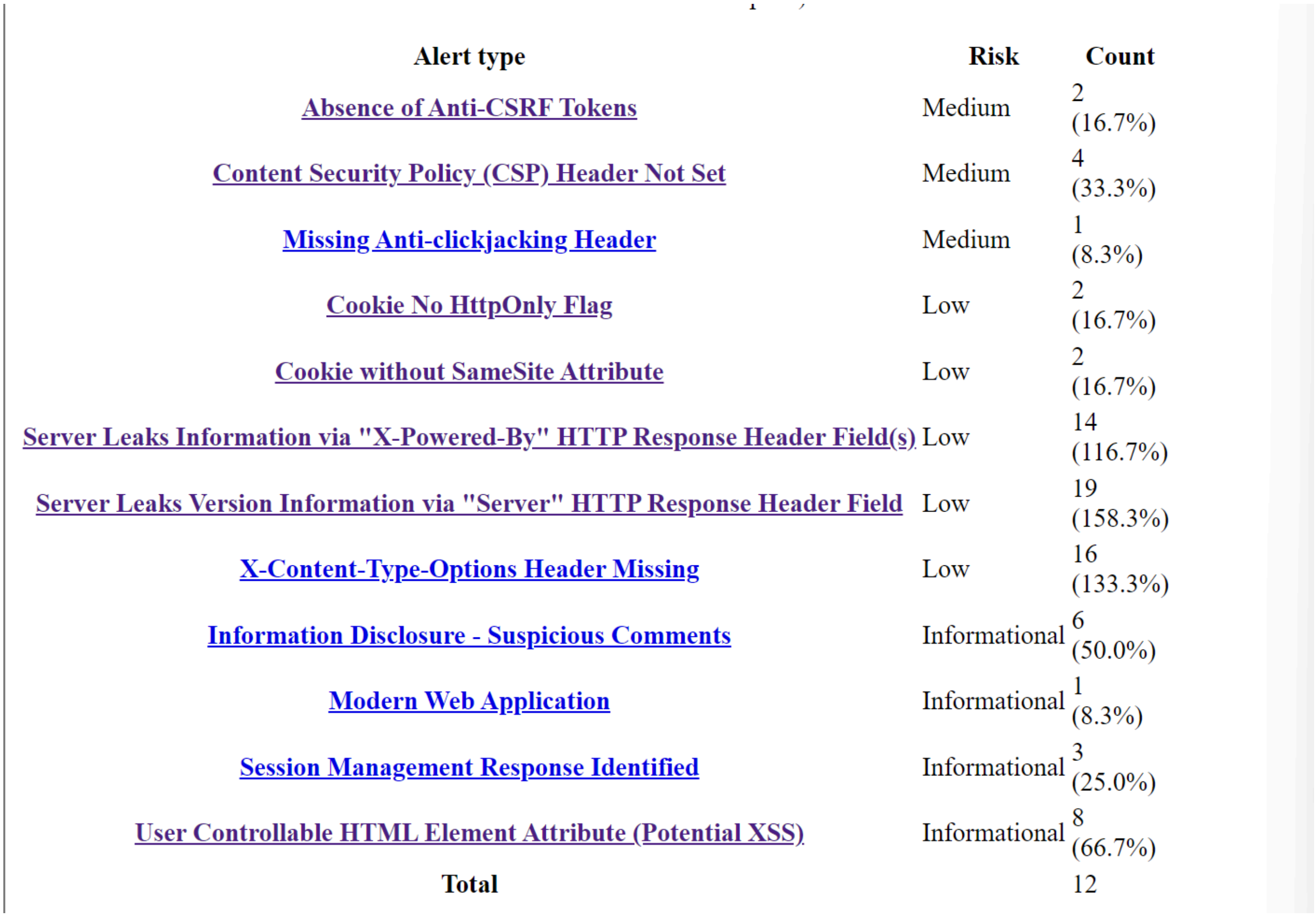
The following tables illustrate the vulnerabilities found by impact and recommended remediations:

External Penetration Test Findings

13	5	6	0	1
Critical	High	Medium	Low	Informational

Finding	Severity	Recommendation
External Penetration Test		
Anonymous FTP is enabled	Medium	Disable anonymous FTP access on the FTP server to prevent unauthorized users from accessing the system without authentication
Vulnerable to Terrapin	Medium	ensure WordPress is updated to version 4.7.1 or later to patch the vulnerability.
WordPress Username Enumeration	Low	update affected SSH implementations, including OpenSSH, to version 9.6 or later, or apply patches provided by vendors as soon as they become available

10.15.42.7 - Zap scanning result



Alert type	Risk	Count
Absence of Anti-CSRF Tokens	Medium	2 (16.7%)
Content Security Policy (CSP) Header Not Set	Medium	4 (33.3%)
Missing Anti-clickjacking Header	Medium	1 (8.3%)
Cookie No HttpOnly Flag	Low	2 (16.7%)
Cookie without SameSite Attribute	Low	2 (16.7%)
Server Leaks Information via "X-Powered-By" HTTP Response Header Field(s)	Low	14 (116.7%)
Server Leaks Version Information via "Server" HTTP Response Header Field	Low	19 (158.3%)
X-Content-Type-Options Header Missing	Low	16 (133.3%)
Information Disclosure - Suspicious Comments	Informational	6 (50.0%)
Modern Web Application	Informational	1 (8.3%)
Session Management Response Identified	Informational	3 (25.0%)
User Controllable HTML Element Attribute (Potential XSS)	Informational	8 (66.7%)
Total		12

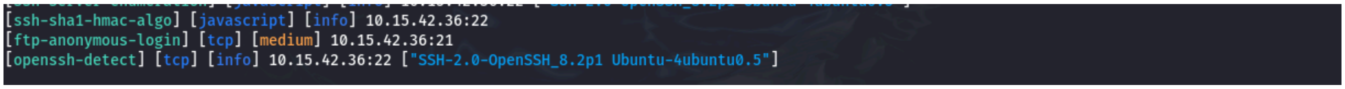
Technical Findings

External Penetration Test Findings

Anonymous FTP is enabled – 10.15.42.36 (Medium)

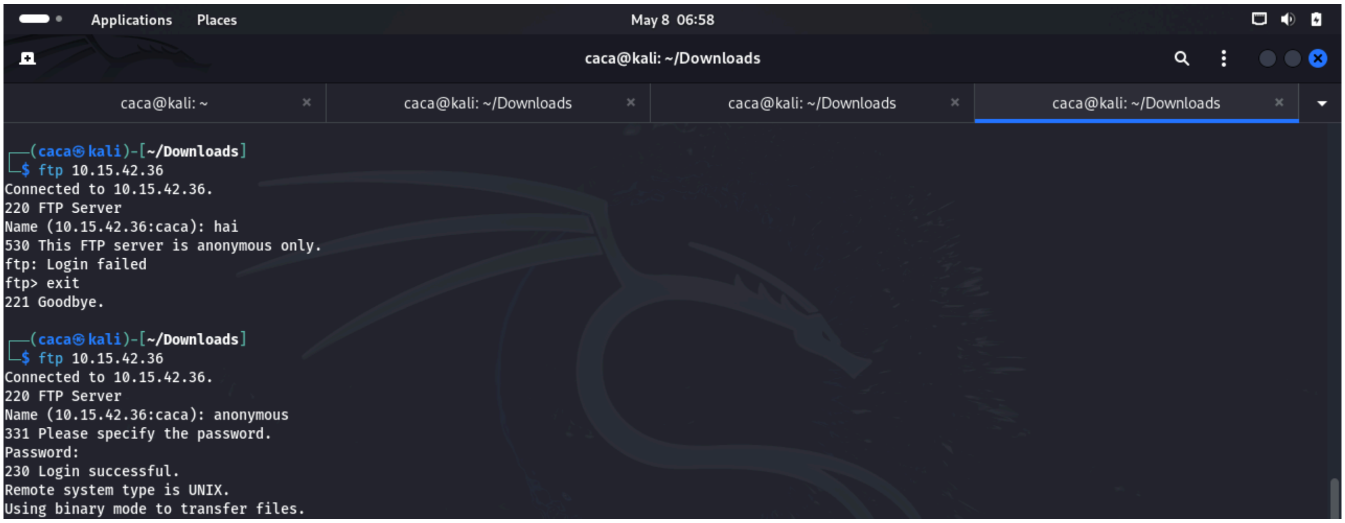
Description:	During the reconnaissance with nmap, the practitioner discovered that FTP on 10.15.42.36 allows anonymous users to log in without a password. This configuration grants access to anyone to access the FTP protocol IP. Anonymous FTP is Enabled.
Risk:	Medium
System:	10.15.42.36
Tools:	FTP
References:	CVE-1999-0497 - Anonymous FTP is enabled.

Evidence



```
[ssh-sha1-hmac-algo] [javascript] [info] 10.15.42.36:22
[ftp-anonymous-login] [tcp] [medium] 10.15.42.36:21
[openssh-detect] [tcp] [info] 10.15.42.36:22 ["SSH-2.0-OpenSSH_8.2p1 Ubuntu-4ubuntu0.5"]
```

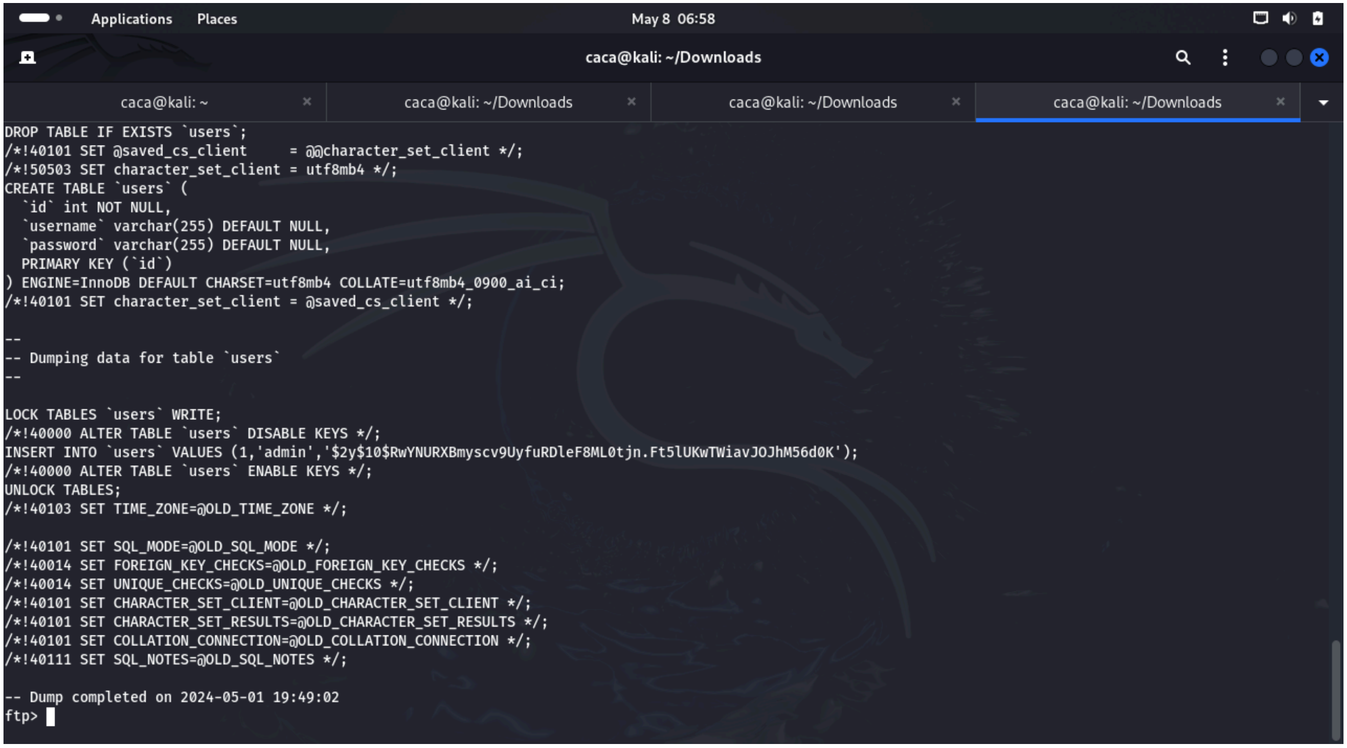
Figure 1: Anonymous FTP is enabled



```
┌──(caca㉿kali)-[~/Downloads]
└─$ ftp 10.15.42.36
Connected to 10.15.42.36.
220 FTP Server
Name (10.15.42.36:caca): hai
530 This FTP server is anonymous only.
ftp: Login failed
ftp> exit
221 Goodbye.

┌──(caca㉿kali)-[~/Downloads]
└─$ ftp 10.15.42.36
Connected to 10.15.42.36.
220 FTP Server
Name (10.15.42.36:caca): anonymous
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
```

Figure 2: Logging in as anonymous



```
DROP TABLE IF EXISTS `users`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!50503 SET character_set_client = utf8mb4 */;
CREATE TABLE `users` (
  `id` int NOT NULL,
  `username` varchar(255) DEFAULT NULL,
  `password` varchar(255) DEFAULT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
/*!40101 SET character_set_client = @saved_cs_client */;


--
-- Dumping data for table `users`
--

LOCK TABLES `users` WRITE;
/*!40000 ALTER TABLE `users` DISABLE KEYS */;
INSERT INTO `users` VALUES (1,'admin','$2y$10$RwYNURXBmyscv9UyfuRDleF8ML0tjn.Ft5lUKwTWiavJOJhM56d0K');
/*!40000 ALTER TABLE `users` ENABLE KEYS */;
UNLOCK TABLES;
/*!40103 SET TIME_ZONE=@OLD_TIME_ZONE */;

/*!40101 SET SQL_MODE=@OLD_SQL_MODE */;
/*!40014 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS */;
/*!40014 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS */;
/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
/*!40111 SET SQL_NOTES=@OLD_SQL_NOTES */;

-- Dump completed on 2024-05-01 19:49:02
ftp>
```

Figure 3: accessing database .sql to get users and hash password information

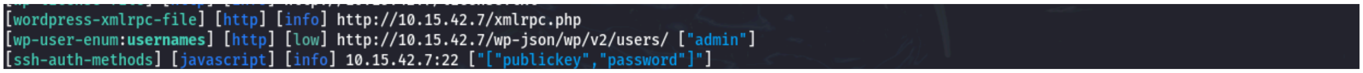
Remediation

Firstly, it's crucial to immediately disable anonymous FTP access on the FTP server to prevent unauthorized users from accessing the system without authentication. Additionally, ensuring that the FTP server software is regularly patched and updated with the latest security patches is essential to address any known vulnerabilities or weaknesses. Furthermore, reviewing and configuring the FTP server securely according to industry best practices and security guidelines can help mitigate risks. This may include restricting access to authorized users only, implementing strong authentication mechanisms, and encrypting data transmissions. Regular security audits and assessments should also be conducted to identify and address any potential security risks or vulnerabilities in the system, including FTP-related issues. Implementing monitoring and logging mechanisms to track FTP activity on the system is crucial for detecting and responding to any suspicious or unauthorized access attempts. Lastly, providing training and awareness programs for users and administrators to educate them about the risks associated with anonymous FTP and the importance of following secure practices can further enhance security measures.

WordPress Username Enumeration – 10.15.42.7 (Low)

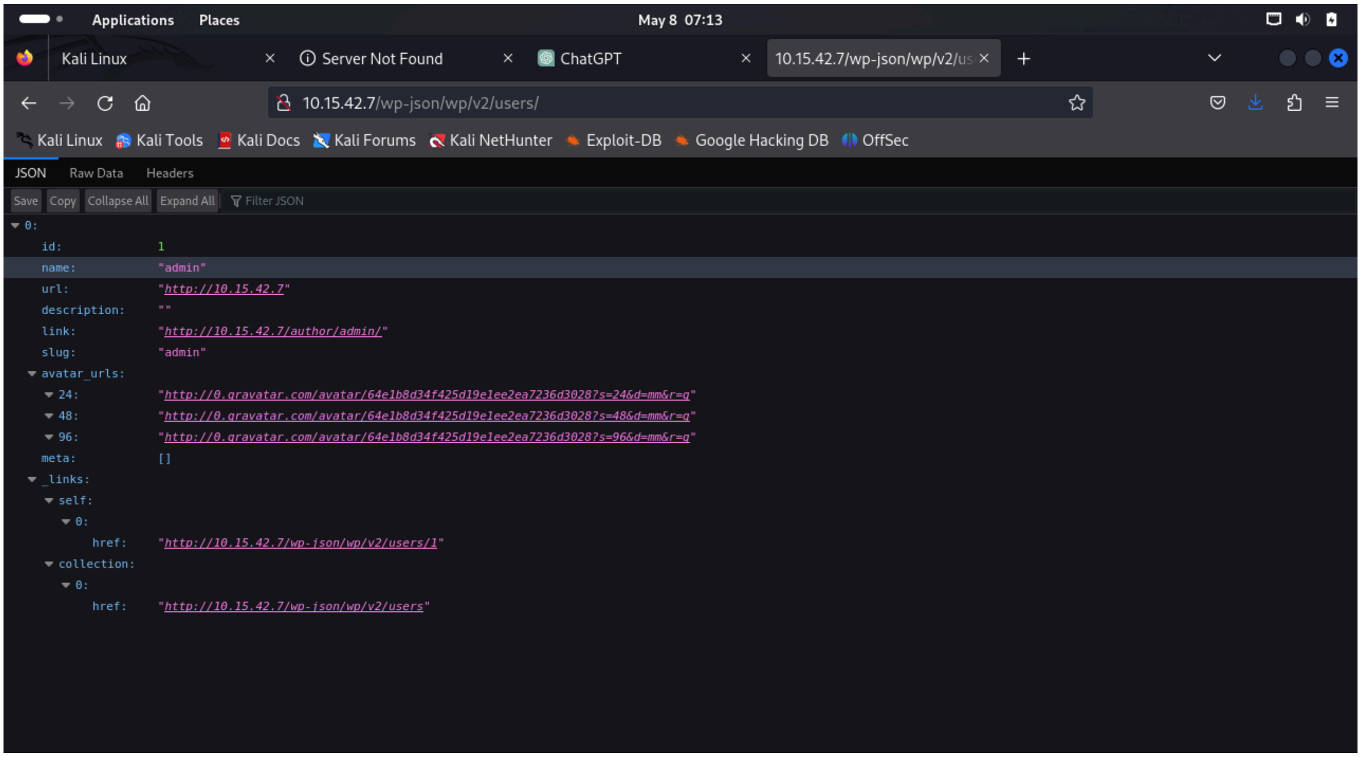
Description:	wp-includes/rest-api/endpoints/class-wp-rest-users-controller.php in the REST API implementation in WordPress 4.7 before 4.7.1 does not properly restrict listings of post authors, which allows remote attackers to obtain sensitive information via a wp-json/wp/v2/users request.
Impact:	Medium
System:	10.15.42.7
Tools Used:	Nuclei
References:	CVE-2017-5487 - WordPress Username Enumeration

Evidence



```
[wordpress-xmlrpc-file] [http] [info] http://10.15.42.7/xmlrpc.php
[wp-user-enum:usernames] [http] [low] http://10.15.42.7/wp-json/wp/v2/users/ ["admin"]
[ssh-auth-methods] [javascript] [info] 10.15.42.7:22 ["["publickey","password"]"]
```

Figure 4: Nuclei report wp-user-enum 10.15.42.7



```
▼ 0:
    id:             1
    name:           "admin"
    url:            "http://10.15.42.7"
    description:    ""
    link:           "http://10.15.42.7/author/admin/"
    slug:           "admin"
  ▼ avatar_urls:
    ▼ 24:           "http://0.gravatar.com/avatar/64e1b8d34f425d19e1ee2ea7236d3028?s=24&d=mm&r=g"
    ▼ 48:           "http://0.gravatar.com/avatar/64e1b8d34f425d19e1ee2ea7236d3028?s=48&d=mm&r=g"
    ▼ 96:           "http://0.gravatar.com/avatar/64e1b8d34f425d19e1ee2ea7236d3028?s=96&d=mm&r=g"
    meta:           []
  ▼ _links:
    ▼ self:
      ▼ 0:
          href:     "http://10.15.42.7/wp-json/wp/v2/users/1"
    ▼ collection:
      ▼ 0:
          href:     "http://10.15.42.7/wp-json/wp/v2/users"
```

Figure 5: WebPage 10,.15.42.7/wp-json/wp/v2/users

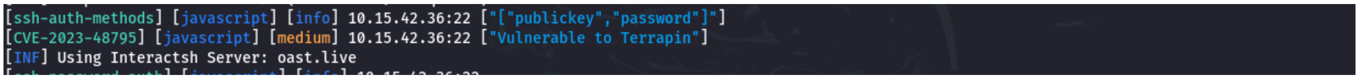
Remediation

Firstly, ensure WordPress is updated to version 4.7.1 or later to patch the vulnerability. Consider installing security plugins to bolster website defenses against such attacks. If the REST API endpoints are non-essential, restrict or disable access to them. Implement rate limiting to thwart automated enumeration attempts. Educate users on the risks of weak usernames and monitor access logs for suspicious activity.

Vulnerable to Terrapin – 10.15.42.36 (Medium)

Description:	The SSH transport protocol with certain OpenSSH extensions, found in OpenSSH before 9.6 and other products, allows remote attackers to bypass integrity checks such that some packets are omitted (from the extension negotiation message), and a client and server may consequently end up with a connection for which some security features have been downgraded or disabled, aka a Terrapin attack. This occurs because the SSH Binary Packet Protocol (BPP), implemented by these extensions, mishandles the handshake phase and mishandles use of sequence numbers. For example, there is an effective attack against SSH's use of ChaCha20-Poly1305 (and CBC with Encrypt-then-MAC). The bypass occurs in chacha20-poly1305@openssh.com and (if CBC is used) the -etm@openssh.com MAC algorithms. This also affects Maverick Synergy Java SSH API before 3.1.0-SNAPSHOT, Dropbear through 2022.83, Ssh before 5.1.1 in Erlang/OTP, PuTTY before 0.80, AsyncSSH before 2.14.2, golang.org/x/crypto before 0.17.0, libssh before 0.10.6, libssh2 through 1.11.0, Thorn Tech SFTP Gateway before 3.4.6, Tera Term before 5.1, Paramiko before 3.4.0, jsch before 0.2.15, SFTPGo before 2.5.6, Netgate pfSense Plus through 23.09.1, Netgate pfSense CE through 2.7.2, HPN-SSH through 18.2.0, ProFTPD before 1.3.8b (and before 1.3.9rc2), ORYX CycloneSSH before 2.3.4, NetSarang XShell 7 before Build 0144, CrushFTP before 10.6.0, ConnectBot SSH library before 2.2.22, Apache MINA sshd through 2.11.0, sshj through 0.37.0, TinySSH through 20230101, trilead-ssh2 6401, LANCOM LCOS and LANconfig, FileZilla before 3.66.4, Nova before 11.8, PKIX-SSH before 14.4, SecureCRT before 9.4.3, Transmit5 before 5.10.4, Win32-OpenSSH before 9.5.0.0p1-Beta, WinSCP before 6.2.2, Bitvise SSH Server before 9.32, Bitvise SSH Client before 9.33, KiTTY through 0.76.1.13, the net-ssh gem 7.2.0 for Ruby, the mscdex ssh2 module before 1.15.0 for Node.js, the thrussh library before 0.35.1 for Rust, and the Russh crate before 0.40.2 for Rust.
Impact:	Medium
System:	10.15.42.36
Tools Used:	Nuclei
References:	CVE-2023-48795 - Vulnerable to Terrapin

Evidence



```
[ssh-auth-methods] [javascript] [info] 10.15.42.36:22 ["["publickey","password"]"]
[CVE-2023-48795] [javascript] [medium] 10.15.42.36:22 ["Vulnerable to Terrapin"]
[INF] Using Interactsh Server: oast.live
[ssh-password-auth] [javascript] [info] 10.15.42.36:22
```

Figure 6: Nuclei report 10.15.42.36 Vulnerability to Terrapin

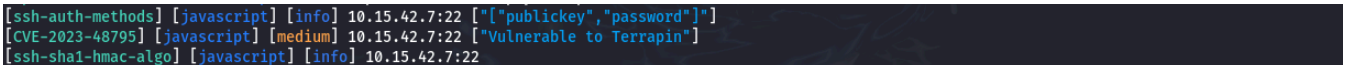
Remediation

Firstly, it's crucial to update affected SSH implementations, including OpenSSH, to version 9.6 or later, or apply patches provided by vendors as soon as they become available. Additionally, organizations should consider implementing additional security measures, such as network segmentation and access controls, to limit the impact of potential exploitation. Regular monitoring and auditing of SSH traffic can also help detect and respond to any suspicious activity associated with the vulnerability. Furthermore, organizations should stay informed about updates and advisories from vendors and security organizations to ensure they are aware of any new developments or emerging threats related to this vulnerability.

Vulnerable to Terrapin – 10.15.42.7 (Medium)

Description:	The SSH transport protocol with certain OpenSSH extensions, found in OpenSSH before 9.6 and other products, allows remote attackers to bypass integrity checks such that some packets are omitted (from the extension negotiation message), and a client and server may consequently end up with a connection for which some security features have been downgraded or disabled, aka a Terrapin attack. This occurs because the SSH Binary Packet Protocol (BPP), implemented by these extensions, mishandles the handshake phase and mishandles use of sequence numbers. For example, there is an effective attack against SSH's use of ChaCha20-Poly1305 (and CBC with Encrypt-then-MAC). The bypass occurs in chacha20-poly1305@openssh.com and (if CBC is used) the -etm@openssh.com MAC algorithms. This also affects Maverick Synergy Java SSH API before 3.1.0-SNAPSHOT, Dropbear through 2022.83, Ssh before 5.1.1 in Erlang/OTP, PuTTY before 0.80, AsyncSSH before 2.14.2, golang.org/x/crypto before 0.17.0, libssh before 0.10.6, libssh2 through 1.11.0, Thorn Tech SFTP Gateway before 3.4.6, Tera Term before 5.1, Paramiko before 3.4.0, jsch before 0.2.15, SFTPGo before 2.5.6, Netgate pfSense Plus through 23.09.1, Netgate pfSense CE through 2.7.2, HPN-SSH through 18.2.0, ProFTPD before 1.3.8b (and before 1.3.9rc2), ORYX CycloneSSH before 2.3.4, NetSarang XShell 7 before Build 0144, CrushFTP before 10.6.0, ConnectBot SSH library before 2.2.22, Apache MINA sshd through 2.11.0, sshj through 0.37.0, TinySSH through 20230101, trilead-ssh2 6401, LANCOM LCOS and LANconfig, FileZilla before 3.66.4, Nova before 11.8, PKIX-SSH before 14.4, SecureCRT before 9.4.3, Transmit5 before 5.10.4, Win32-OpenSSH before 9.5.0.0p1-Beta, WinSCP before 6.2.2, Bitvise SSH Server before 9.32, Bitvise SSH Client before 9.33, KiTTY through 0.76.1.13, the net-ssh gem 7.2.0 for Ruby, the mscdex ssh2 module before 1.15.0 for Node.js, the thrussh library before 0.35.1 for Rust, and the Russh crate before 0.40.2 for Rust.
Risk:	Medium
System:	10.15.42.7
Tools Used:	Nuclei
References:	CVE-2023-48795 - Vulnerable to Terrapin

Evidence



```
[ssh-auth-methods] [javascript] [info] 10.15.42.7:22 ["["publickey","password"]"]
[CVE-2023-48795] [javascript] [medium] 10.15.42.7:22 ["Vulnerable to Terrapin"]
[ssh-sha1-hmac-algo] [javascript] [info] 10.15.42.7:22
```

Figure 7: Nuclei report 10.15.42.7 Vulnerability to Terrapin

Remediation

Firstly, it's crucial to update affected SSH implementations, including OpenSSH, to version 9.6 or later, or apply patches provided by vendors as soon as they become available. Additionally, organizations should consider implementing additional security measures, such as network segmentation and access controls, to limit the impact of potential exploitation. Regular monitoring and auditing of SSH traffic can also help detect and respond to any suspicious activity associated with the vulnerability. Furthermore, organizations should stay informed about updates and advisories from vendors and security organizations to ensure they are aware of any new developments or emerging threats related to this vulnerability.

10.25.42.36:8888 - Zap scanning result detail

Absence of Anti-CSRF Tokens (Medium)

Description:	The web application does not, or can not, sufficiently verify whether a well-formed, valid, consistent request was intentionally provided by the user who submitted the request.
Impact:	Medium
System:	10.15.42.36:8888
References:	CWE-352: Cross-Site Request Forgery (CSRF)

Content Security Policy (CSP) Header Not Set (Medium)

Description:	The product does not use or incorrectly uses a protection mechanism that provides sufficient defense against directed attacks against the product.
Impact:	Medium
System:	10.15.42.36:8888
References:	CWE-693: Protection Mechanism Failure

Missing Anti-clickjacking Header (Medium)

Description:	The web application does not restrict or incorrectly restricts frame objects or UI layers that belong to another application or domain, which can lead to user confusion about which interface the user is interacting with.
Impact:	Medium
System:	10.15.42.36:8888
References:	CWE-1021: Improper Restriction of Rendered UI Layers or Frames

Server Leaks Version Information via "Server" HTTP Response Header Field (Low)

Description:	The product exposes sensitive information to an actor that is not explicitly authorized to have access to that information.
Impact:	Low

System:	10.15.42.36:8888
References:	CWE-200: Exposure of Sensitive Information to an Unauthorized Actor

X-Content-Type-Options Header Missing (Low)

Description:	The product does not use or incorrectly uses a protection mechanism that provides sufficient defense against directed attacks against the product.
Impact:	Low
System:	10.15.42.36:8888
References:	CWE-693: Protection Mechanism Failure

10.15.42.7 - Zap scanning result detail
Absence of Anti-CSRF Tokens – 10.15.42.7 (Medium)

Source	raised by a passive scanner (Absence of Anti-CSRF Tokens)
CWE ID	352
WASC ID	9
Reference	

http://projects.webappsec.org/Cross-Site-Request-Forgery
https://cwe.mitre.org/data/definitions/352.html

Content Security Policy (CSP) Header Not Set – 10.15.42.7 (Medium)

Source	raised by a passive scanner (Content Security Policy (CSP) Header Not Set)
CWE ID	693
WASC ID	15
Reference	

https://developer.mozilla.org/en-US/docs/Web/Security/CSP/Introducing_Content_Security_Policy
https://cheatsheetseries.owasp.org/cheatsheets/Content_Security_Policy_Cheat_Sheet.html
http://www.w3.org/TR/CSP/

http://w3c.github.io/webappsec/specs/content-security-policy/csp-specification.dev.html
http://www.html5rocks.com/en/tutorials/security/content-security-policy/
http://caniuse.com/#feat=contentsecuritypolicy
http://content-security-policy.com/

Missing Anti-clickjacking Header – 10.15.42.7 (Medium)

Source	raised by a passive scanner (Anti-clickjacking Header)
CWE ID	1021
WASC ID	15

Reference
https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options

Cookie No HttpOnly Flag – 10.15.42.7 (Low)

Source	raised by a passive scanner (Cookie No HttpOnly Flag)
CWE ID	1004
WASC ID	13

Reference
https://owasp.org/www-community/HttpOnly

Cookie without SameSite Attribute – 10.15.42.7 (Low)

Source	raised by a passive scanner (Cookie without SameSite Attribute)
CWE ID	1275
WASC ID	13

Reference
https://tools.ietf.org/html/draft-ietf-httpbis-cookie-same-site

Server Leaks Information via "X-Powered-By" HTTP Response Header Field(s) – 10.15.42.7 (Low)

Source	raised by a passive scanner (Server Leaks Information via "X-Powered-By" HTTP Response Header Field(s))
CWE ID	200
WASC ID	13

Reference
http://blogs.msdn.com/b/varunm/archive/2013/04/23/remove-unwanted-http-response-headers.aspx
http://www.troyhunt.com/2012/02/shhh-dont-let-your-response-headers.html

Server Leaks Version Information via "Server" HTTP Response Header Field – 10.15.42.7 (Low)

Source	raised by a passive scanner (HTTP Server Response Header)
CWE ID	200
WASC ID	13

Reference
http://httpd.apache.org/docs/current/mod/core.html#servertokens

http://msdn.microsoft.com/en-us/library/ff648552.aspx#ht_urlscan_007
http://blogs.msdn.com/b/varunm/archive/2013/04/23/remove-unwanted-http-response-headers.aspx
http://www.troyhunt.com/2012/02/shhh-dont-let-your-response-headers.html

X-Content-Type-Options Header Missing – 10.15.42.7 (Low)
Source raised by a passive scanner (X-Content-Type-Options Header Missing)
CWE ID 693
WASC ID 15
Reference
http://msdn.microsoft.com/en-us/library/ie/gg622941%28v=vs.85%29.aspx
https://owasp.org/www-community/Security_Headers

Information Disclosure - Suspicious Comments – 10.15.42.7 (Informational)
Source raised by a passive scanner (Information Disclosure - Suspicious Comments)
CWE ID 200
WASC ID 13

Modern Web Application – 10.15.42.7 (Informational)
Source raised by a passive scanner (Modern Web Application)
Session Management Response Identified
Source raised by a passive scanner (Session Management Response Identified)
Reference
https://www.zaproxy.org/docs/desktop/addons/authentication-helper/session-mgmt-id

User Controllable HTML Element Attribute (Potential XSS) – 10.15.42.7 (Informational)
Source raised by a passive scanner (User Controllable HTML Element Attribute (Potential XSS))
CWE ID 20
WASC ID 20
Reference
http://websecuritytool.codeplex.com/wikipage?title=Checks#user-controlled-html-attribute