

**Dynamic Vulnerability Assessment**

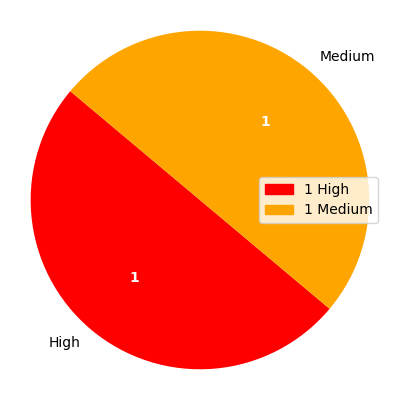
**sdgjhbvkdss**

**Requested by:** fhdhjdghj

## Version Information

|  |  |  |
| --- | --- | --- |
| Date | Application Version | Reviewer |
| 22-May-2025 | Initial Draft | dgjdgjdgj |
| 22-May-2025 | Peer Review |  |
| 22-May-2025 | Approved |  |

**Vulnerability Severity Distribution**



## Summary Table

|  |  |  |  |
| --- | --- | --- | --- |
| Sl. No. | Security Observation | Risk Rating | Page No. |
| **High Severity** | | | |
| 1 | Misconfigured Content-Security-Policy Header | High | 4 |
| **Medium Severity** | | | |
| 2 | Insecure Transport: Weak SSL Ciphers | Medium | 5 |

## URLs and Scope

URLs: dgjdgjdgjd

Scope: jdgjhdjgdggj

## Vulnerability Details

### 1. Insecure Transport: Weak SSL Ciphers

**Severity: Medium**

CVSS Score: 6.3

CVSS Vector: CVSS:3.1/AV:A/AC:L/PR:L/UI:R/S:U/C:L/I:L/A:H

#### Description

Improper The Transport Layer Security (TLS) and Secure Sockets Layer (SSL) protocols provide a  
mechanism to help protect authenticity, confidentiality and integrity of the data transmitted between a  
client and web server. The strength of this protection mechanism is determined by the authentication,  
encryption and hashing algorithms. These are collectively known as a cipher suite chosen for the  
transmission of sensitive information over the TLS/SSL channel. Most web servers support a range of such  
cipher suites of varying strengths. Using a weak cipher or an encryption key of insufficient length, for  
example, could enable an attacker to defeat the protection mechanism and steal or modify sensitive  
information. If misconfigured, a web server could be manipulated into choosing weak cipher suites. A  
weak encryption scheme can be subjected to brute force attacks that have a reasonable chance of  
succeeding using current methods and resources. An attacker could possibly execute a man in the middle  
attack which would allow them to intercept, monitor and tamper with sensitive data. Each weak cipher  
was enumerated by establishing an SSL connection with the target host and specifying the cipher to test  
in the Client Hello message of the SSL handshake.

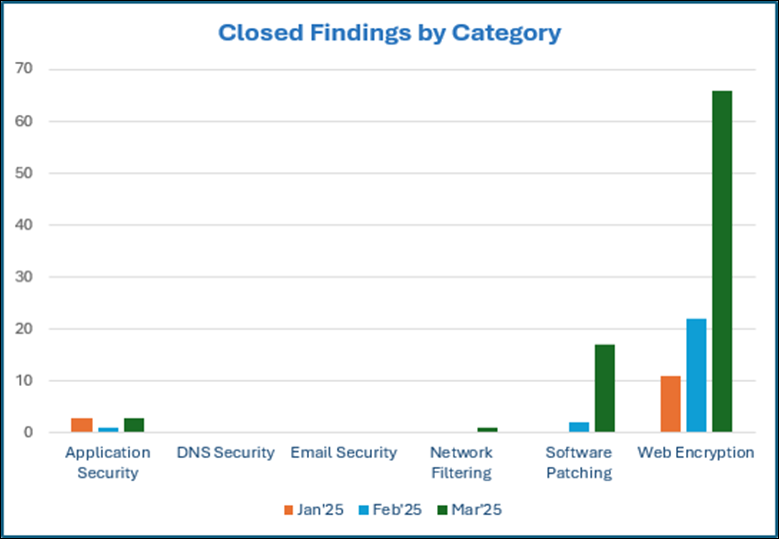
#### Evidence

Step 1: gndfgndgn



[Invalid image: uploaded\_evidence/124144b01c724b669f9063c1b04930fe.avif]

Step 2: dfhdhdhdh



#### Recommendation

It is recommended not to use RC4, CBC,SHA, SHA1, MD5 etc ciphers

#### Reference

https://www.acunetix.com/blog/articles/tls-ssl-cipher-hardening/  
http://zero.webappsecurity.com

### 2. Misconfigured Content-Security-Policy Header

**Severity: High**

CVSS Score: 7.5

CVSS Vector: CVSS:3.1/AV:A/AC:H/PR:L/UI:R/S:C/C:H/I:L/A:H

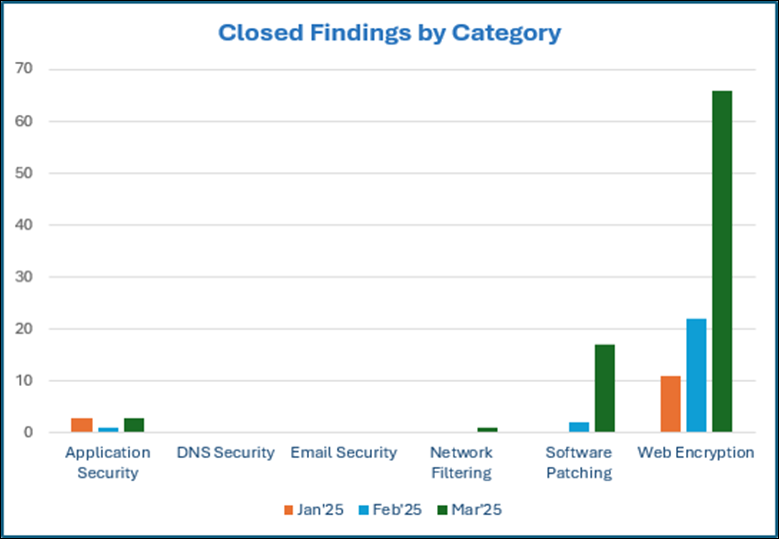
#### Description

It is observed the CSP header value is not implemented properly in the response. Configuring Content  
Security Policy involves adding the proper CSP HTTP header to a web page and giving it values to control  
resources the user agent is allowed to load for that page. For example, a page that uploads and displays  
images could allow images from anywhere but restrict a form action to a specific endpoint. A properly  
designed Content Security Policy helps protect a page against a cross site scripting attack.

#### Evidence

Step 1: thsififpoihflsihglskdbgnlks gb

[Invalid image: uploaded\_evidence/48b2ebae25214b4cae72ce76cb30920c.avif]



Step 2: dsgslkjhfgweiukgbskigblsknbg



#### Recommendation

Configuring Content Security Policy involves adding the proper CSP HTTP header to a web page and giving  
it values to control resources the user agent is allowed to load for that page. For example, a page that  
uploads and displays images could allow images from anywhere but restrict a form action to a specific  
endpoint. A properly designed Content Security Policy helps protect a page against a cross site scripting  
attack.  
It is suggested to implement CSP as per the application requirements. For example, content security policy  
can be designed as mentioned below: “Content-Security-Policy: default-src 'self’ https://Appdomain;  
child-src 'none'; object-src 'none

#### Reference

https://owasp.org/www-community/controls/Content\_Security\_Policy  
https://developer.mozilla.org/en-US/docs/Web/HTTP/CSP