Dynamic Vulnerability Assessment

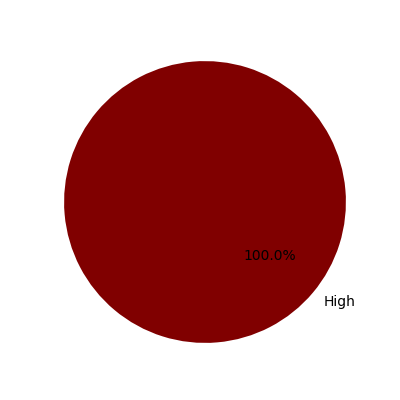
**dsrhdh**

**Requested by:** dfhdh

# Version Information

|  |  |  |
| --- | --- | --- |
| Initial Draft | Peer Review | Approved By |
| 2025-05-02 |  |  |

**Vulnerability Severity Distribution**



# Summary of Findings

|  |  |  |  |
| --- | --- | --- | --- |
| Security Observation | Severity | CVSS Score | CVSS Vector |
| Insecure Transport: Weak SSL Ciphers | High | 7.1 | CVSS:3.1/AV:A/AC:H/PR:L/UI:N/S:U/C:H/I:H/A:H |

# Assessment Scope

URLs: dfn

Scope: dxdfndn

# Insecure Transport: Weak SSL Ciphers

Severity: High

CVSS Score: 7.1

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

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/

## Evidences

Step 1: xcfndfndndgn

Step 2: dfndgndgndgndgnd