Sri Lanka Institute of Information Technology

Bug Bounty Report 04

cnsz03nsbc01pp.aswatson.com

IE2062 – Web Security

Submitted by:

IT22227836 – ADHIKARI A.M.V.B.M

Date of submission

2024.10.31

Table of Contents

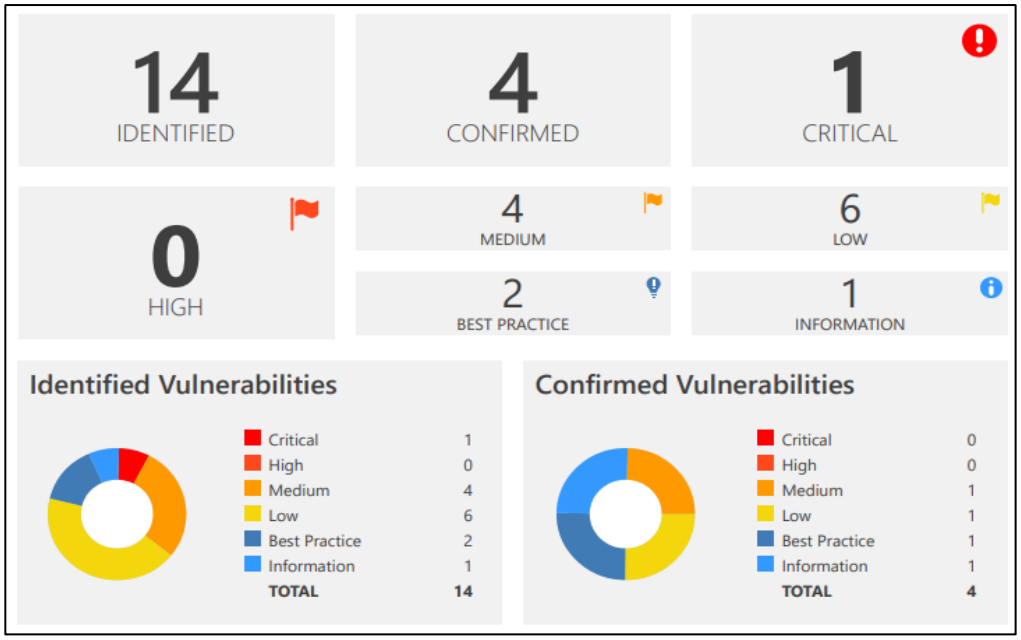
Report 04 – cnsz03nsbc01pp.aswatson.com

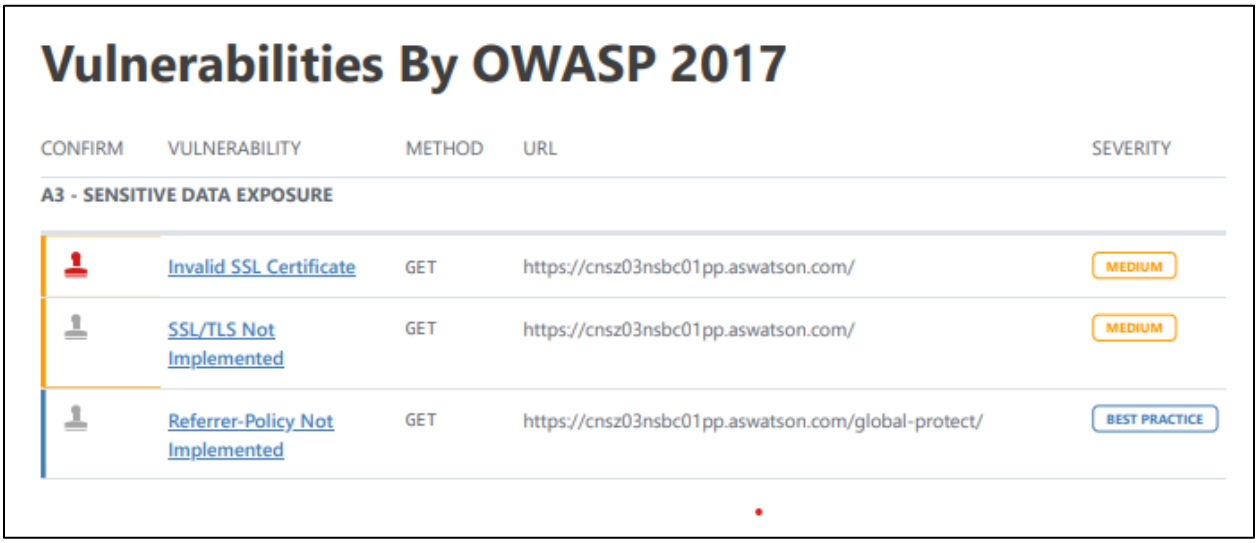
Main domain	https://www.aswatson.com/
Sub domain	cnsz03nsbc01pp.aswatson.com
IP address	120.31.161.33
platform	HackerOne

The A.S. Watson Group is the world's largest health and beauty retail group, with over 15,700 stores in 25 markets worldwide serving over 28 million customers per week, and over 3 billion customers and members.

A.S. Watson Group looks forward to working with the security community to discover vulnerabilities to keep our businesses and customers safe. As we operate in many different countries, we will be rolling out our bug bounty program in phases. Our focus within this rollout, is our retail websites (and soon, mobile apps on both Android and IOS)

Vulnerability detected



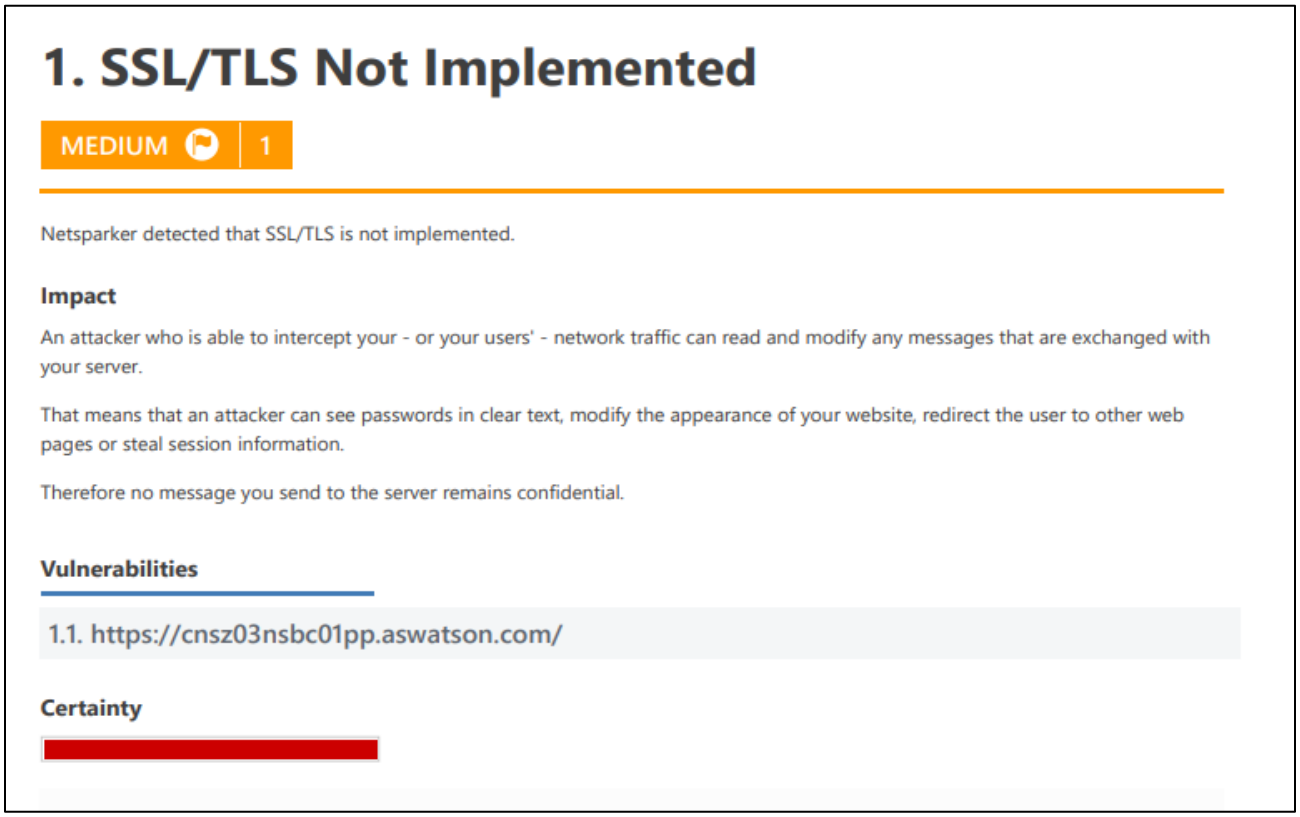


Vulnerabilities By OWASP 2017

CONFIRM	VULNERABILITY	METHOD	URL	SEVERITY
A3 - SENSITIVE DATA EXPOSURE				
	Invalid SSL Certificate	GET	https://cnsz03nsbc01pp.aswatson.com/	MEDIUM
	SSL/TLS Not Implemented	GET	https://cnsz03nsbc01pp.aswatson.com/	MEDIUM
	Referrer-Policy Not Implemented	GET	https://cnsz03nsbc01pp.aswatson.com/global-protect/	BEST PRACTICE

Vulnerability

1. Title – SSL/TLS Not Implemented



- Risk – Medium

Description

When SSL/TLS is not deployed, the communication between websites happens in mere HTTP, which is less secure because it doesn't encrypt the data. Therefore, any attacker can intercept and read the data being transmitted by the server and users. This could mean passwords, credit card information, personal identification details, or other sensitive data.

The SSL Certificate is used for encrypting the data sent from a user's browser to the server, in which unauthorized access is not allowed to get such data. Without SSL/TLS, there is no guarantee that the data is in transit securely but is open to various attacks like MitM or eavesdropping.

Affected components

The vulnerability concerns either SSL/TLS Not Implemented. The impacted components are as follows:

- **Web Server:** This is the primary component, which, in this context, serves the website over unencrypted HTTP. Servers like Apache, Nginx, or Microsoft IIS will have this vulnerability if they are not appropriately set up with SSL/TLS.
- **Client-Server Communication:** It affects any communication that happens between the client, which could be the user's browser or an application, and the server. Data goes out in plain text without SSL/TLS in these communications and can easily be intercepted.
- **API Endpoints:** If the website or service uses APIs for transferring data, those API requests are also set up for attack upon transmission without encryption. This puts any data transmitted via those endpoints at risk.
- **Login Systems:** User login pages/forms transmitting credentials in plaintext by HTTP are very prone to leakage of sensitive login information to potential attackers.
- **User Devices:** Affected, too, are the clients whose devices connect to those very websites or services since their personal or sensitive information is exposed by not having SSL/TLS in place.
- **Subdomains:** Moreover, if the subdomains of a website are not under SSL/TLS, those parts will also be at risk because, through the same set of vulnerabilities, an attacker is able to exploit other parts of the website.

The above components are integral in setting up secure communication; in their absence, SSL/TLS presents security threats to both organizations and users.

Impact Assessment

- **Web Server**: The major component that gets affected includes the web server, which serves the web site over HTTP without encrypting the communication. For example, servers like Apache, Nginx, or Microsoft IIS, when not properly configured with SSL/TLS, would remain vulnerable.
- **Client-Server Communication:** The communication between the client-to-server application, a browser, is impacted. The data sent over wire is in plain text and can be intercepted if not using SSL/TLS.
- **API Endpoints:** If the website or service uses APIs for data transfer, those API requests are also at risk when sent unencrypted. In that case, any data sent through such endpoints would be presented in a fragile scenario.

- **Login Systems:** User login pages or forms transmitting credentials over HTTP without encryption are highly exploitable; thus, they put sensitive login information at risk of being exposed to malicious attackers.
- **User Devices:** The clients using the website or service are also vulnerable because their sensitive data, such as passwords or credit card numbers, is compromised in the absence of SSL/TLS.
- **Subdomains:** If subdomains of the website are deployed without the use of SSL/TLS, they can also be vulnerable to attacks because an attacker may carry out the attack on multiple subdomains on the website.

These are very basic building blocks for secure communication, and without SSL/TLS, there is vulnerability in both the organizational and user perspective to potential security compromise.

Steps to reproduce

- **Access Website**:

Open the web browser and access the path of the affected website, which can be any, like https://cnsz03nsbc01pp.aswatson.com/.
See if the website opens with HTTP instead of HTTPS, indicating that SSL/TLS encryption is not in use.

- **Browser Security to Check:**

On the address bar, check the left-hand side of the URL for a warning, "Not Secure". This will show the website is without SSL/TLS; this communication is not secure.

- **Check the Certificate:**

Click on the security icon, which most of the time is a padlock beside the URL in the browser address bar. Inspect the security certificate. A missing or invalid security certificate implies an inappropriate implementation of SSL/TLS.

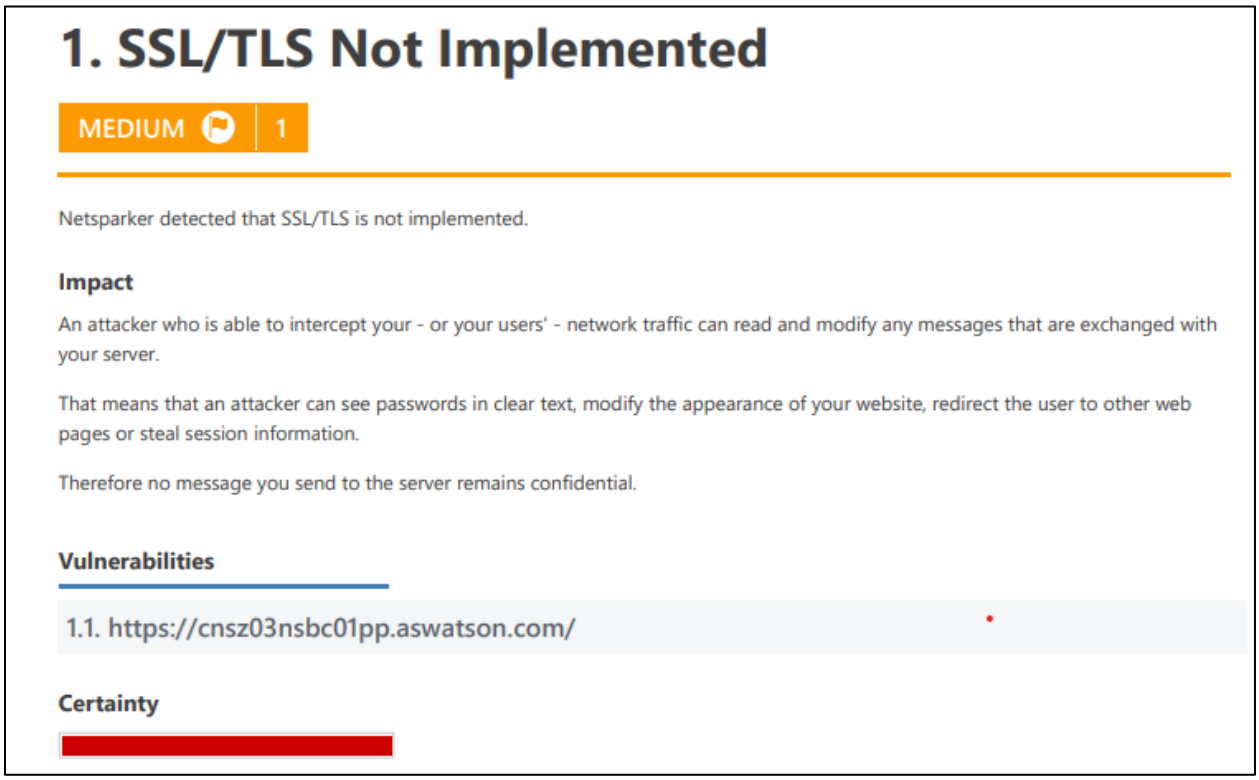
- **Browser Developer Tools:**

At last, one can check the browser developer tool to check the connection protocol of the website. For example, if the website does not use HTTPS, the browser will point toward an insecure connection.

Perform the following steps to easily ensure that SSL/TLS is either missing or ill-configured on a website.

Proof of concept (if applicable)

Vulnerability scanning using Netsparker



1. SSL/TLS Not Implemented

MEDIUM | 1

Netsparker detected that SSL/TLS is not implemented.

Impact

An attacker who is able to intercept your - or your users' - network traffic can read and modify any messages that are exchanged with your server.

That means that an attacker can see passwords in clear text, modify the appearance of your website, redirect the user to other web pages or steal session information.

Therefore no message you send to the server remains confidential.

Vulnerabilities

1.1. https://cnsz03nsbc01pp.aswatson.com/

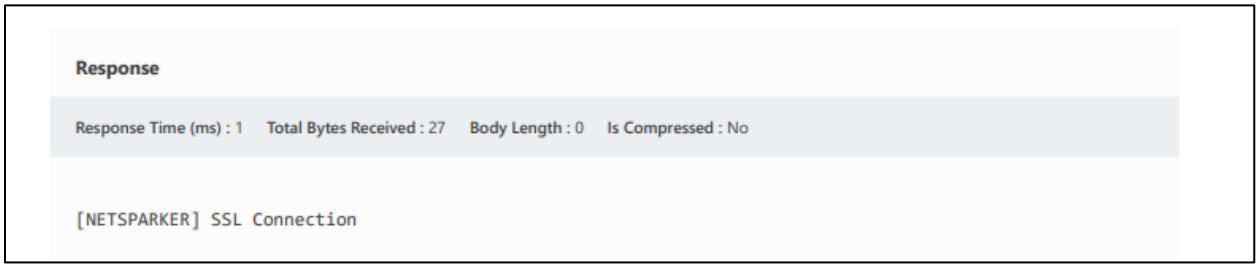
Certainty

- **Request**

Request

```
[NETSPARKER] SSL Connection
```

- **Response**



Response

Response Time (ms) : 1 Total Bytes Received : 27 Body Length : 0 Is Compressed : No

```
[NETSPARKER] SSL Connection
```

Vulnerability finding using namp

Used command

- nmap cnsz03nsbc01pp.aswatson.com

```
┌──(malmi㉿kali)-[~/Desktop]
└─$ nmap cnsz03nsbc01pp.aswatson.com
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-10-09 01:25 EDT
Nmap scan report for cnsz03nsbc01pp.aswatson.com (120.31.161.33)
Host is up (0.49s latency).
rDNS record for 120.31.161.33: ns2.eflydns.net
Not shown: 997 filtered tcp ports (no-response)
PORT     STATE SERVICE
25/tcp   open  smtp
443/tcp  open  https
4443/tcp open  pharos

Nmap done: 1 IP address (1 host up) scanned in 113.12 seconds
```

Founded open port

- 25/tcp open smtp
- 443/tcp open https
- 4443/tcp open pharos

Proposed mitigation or fix

Remedy

We suggest that you implement SSL/TLS properly, for example by using the Certbot toolprovided by the Let's Encrypt certificate authority. It can automatically configure most modern web servers, e.g. Apache and Nginx to use SSL/TLS. Both the tool and the certificates are free and are usually installed within minutes.

- **SSL/TLS:** The web server should be HTTPS-enabled, meaning there should be SSL/TLS to encrypt the communication between the client and the server. Do this by using tools such as Certbot, from Let's Encrypt, an efficient tool which will handle HTTPS for free using its free SSL certificates.

- **Get a valid SSL Certificate:** Purchase a certificate from a renowned CA or use any free service like Let's Encrypt. The SSL certificate should be valid and properly installed on all the domains, including subdomains.

- **Redirect HTTP to HTTPS:** Configure your server to automatically redirect HTTP to HTTPS to make sure that all traffic is delivered over an encrypted channel. This is because no kind of communication should be left occurring over HTTP.

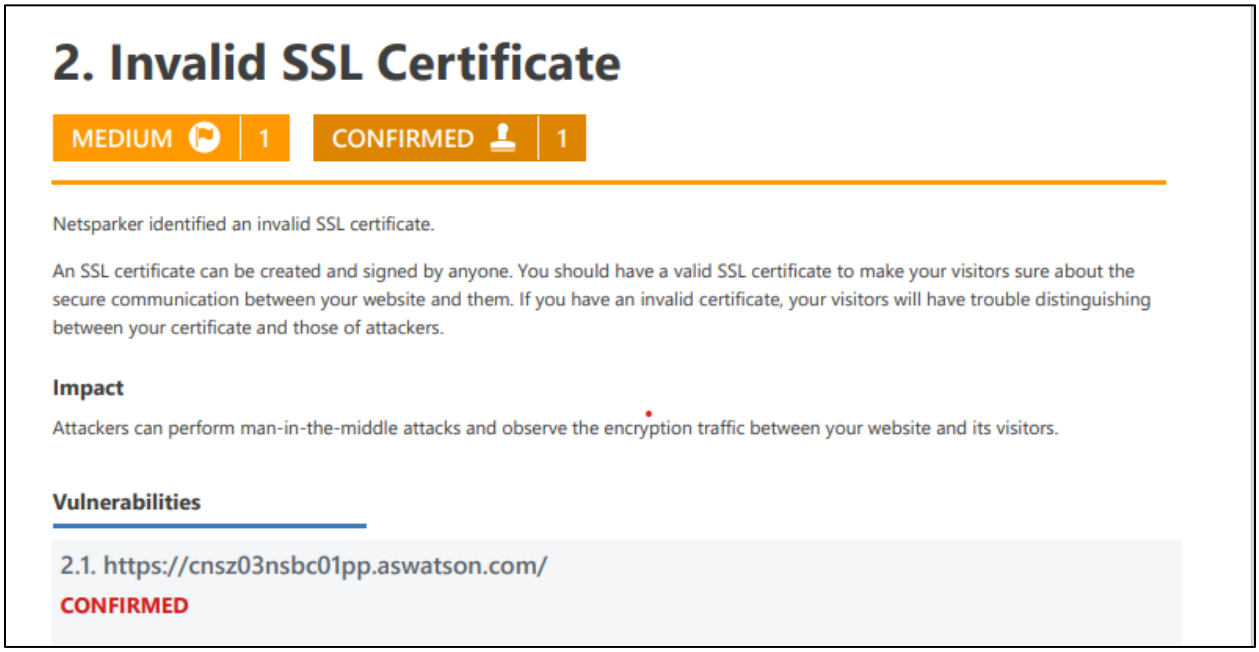
- **Proper Configuration of Cipher Suite:** This will ensure that only strong cipher suites are supported, while weak ones-which would include RC4 or SSLv3-are disabled.

- **Hardening of SSL Configuration:** Implement and test your configuration with SSL testing tools such as SSL Labs, and tune it according to best practices in terms of SSL/TLS security.

- **SSL/TLS Protocols**: Make sure the SSL/TLS implementation on your server is updated so as to allow only the latest security standards and protocols. For example, deprecating all the old protocols like SSLv2/SSLv3 and enabling TLS 1.2 or higher.

With these patches, communications between the web server and its clients will be encrypted, including, for example, man-in-the-middle attacks, which would guarantee the security of data exchange.

2. Tittle – Invalid SSL Certificate



Description

An SSL certificate plays an important role in setting up a secure, encrypted link between a web server and a browser. This ensures that data in transit remains private and cannot be intercepted or tampered with. A faulty SSL certificate occurs due to several issues, such as:

- **CN Mismatch:** A mismatch in the CN means there is a difference between the domain name being accessed and its identification in the SSL certificate. The simple example of this can be accessing the site by using an IP address instead of the domain name.

- **Untrusted CA:** A CA is a third-party company or organization that issues SSLs. If an SSL is self-signed-meaning the owner has created it themselves rather than through one of these trusted companies-or if it was issued by an unrecognized CA, modern browsers will not trust it and show warnings or errors to users trying to access the site.

- **Expired Certificates:** These are usually certificates that have expiration dates. A certificate that has expired cannot be trusted; hence, browser warnings due to its inability to secure user data.

Affected components

Following is some of the critical components which could be affected because of an Invalid SSL Certificate:

- **Web Server Software:**

  Servers such as Apache or Nginx, which should be rightly configured in a way to serve HTTPS traffic. Any kind of misconfiguration could lead to SSL certificate issues.

- **SSL Certificate:**

  It is the most critical component for secure communications. Problems such as expiration, self-signing, or incorrect Common Name CN make it invalid.

- **Certificate Authority:**

  The trust in the certificate is also affected by the CA from which the SSL certificate is issued. Certificates issued through untrusted CAs or compromised CAs would result in failures of validation.

- **Browser:**

  The browser enforces tight validation of the SSL certificate. If that fails, warnings may pop up, undermining user trust

- **Application Layer:**

  Any kind of application whose functioning is dependent upon secure communication-for example, APIs-is influenced. An invalid SSL might disrupt data transmission and show vulnerabilities.

- **User Trust:**

  In other words, this makes the end user be affected because such an invalid SSL may affect trust and make one shy away from interacting with the application.

Impact Assessment

The Invalid SSL Certificate vulnerability has numerous major impacts on both the web application and its users:

- **Man-in-the-Middle Attacks**:

  An invalid SSL certificate lets attacks intercept the communications of users from and to the server. This probably grants them access to reading, modifying, or injecting malicious content into the communication in some manner that compromises sensitive data such as usernames and passwords.

- **Data Integrity Risks:**

  A valid SSL certificate is needed; otherwise, the data transferred between the server and client can be tampered with undetectably. This could lead to malicious payloads being added or phishing sites being redirected to another site .

- **User Trust and Reputation:**

  Users expect secure communications with sites, especially when keying in sensitive information into the sites. An invalid certificate can trigger browser warnings and may result in a lack of trust. Consequences in the long run include lower user engagement, transaction abandonment, and reputational harm.

- **Compliance Issues**:

  Many of the industry standards and regulations, such as PCI DSS and HIPAA, require valid SSL certificates in handling sensitive information. Non-compliance can lead to financial penalties, legal consequences, or loss of certification.

- **Financial Consequences:**

  The combined impacts of reduced customer trust, actual data breaches, and any assessed compliance-related fines or penalties may result in significant financial losses. This involves financial losses stemming from the loss of customers, legal fees, costs related to remedial actions.

Steps to reproduce

- **Log in to the given URL:**
  Open browser and enter the URL:https://cnsz03nsbc01pp.aswatson.com/.

- **Observe the details of the SSL Certificate**
  On the address bar, click on the padlock icon which shows the detailed information about the SSL. The CN should be the domain name. If it is different-namely, "CN=120.31.161.33"-then something is wrong with it.

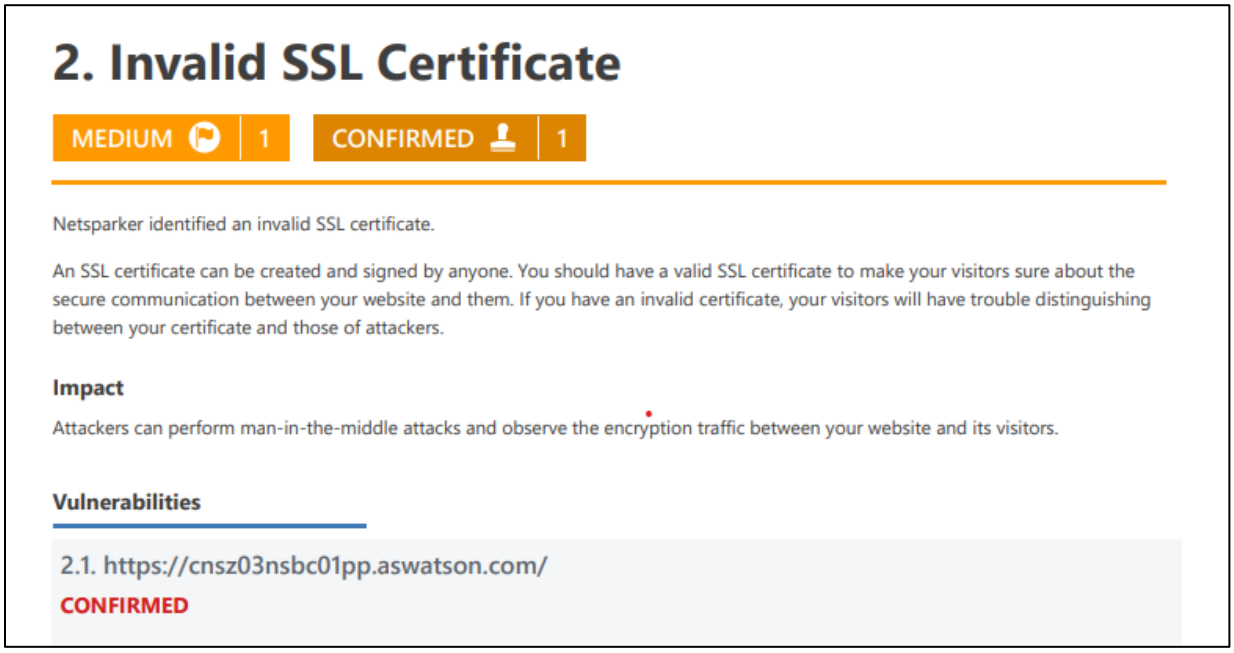
- **Verify the Certificate Chain:**
  The next step will be to test the certificate chain with online tools like SSL Labs or any other SSL testing service. This helps identify whether it is self-signed or issued from an untrusted authority.

- **Browser Warnings:**
  Reload the webpage and see whether your browser emits warnings about the SSL certificate. Most modern browsers warn the user about problems in the SSL certificate.

- **Use SSL Testing Tools:**
  Use SSL testing tools to scan the configuration of the certificate. Look for any validity issues in the SSL certificate that indicate whether it is trusted or not.

Proof of concept (if applicable)

Vulnerability scanning using Netsparker



2. Invalid SSL Certificate

MEDIUM 1 CONFIRMED 1

Netsparker identified an invalid SSL certificate.

An SSL certificate can be created and signed by anyone. You should have a valid SSL certificate to make your visitors sure about the secure communication between your website and them. If you have an invalid certificate, your visitors will have trouble distinguishing between your certificate and those of attackers.

Impact

Attackers can perform man-in-the-middle attacks and observe the encryption traffic between your website and its visitors.

Vulnerabilities

2.1. https://cnsz03nsbc01pp.aswatson.com/

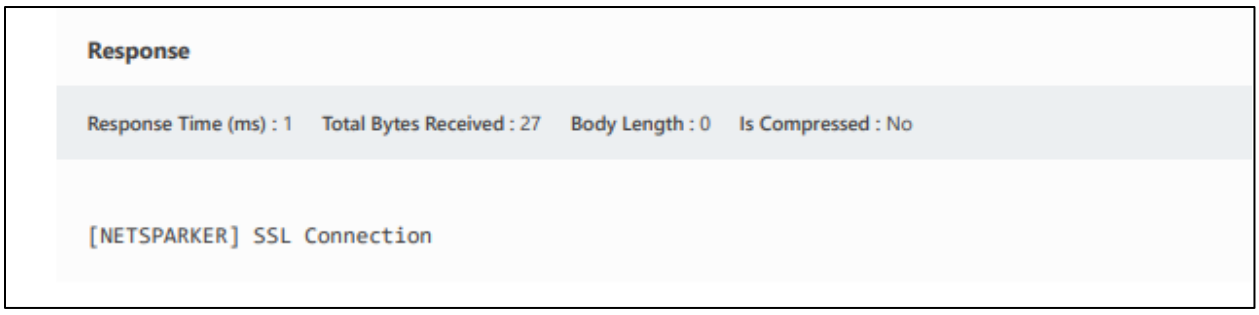
CONFIRMED

- **Request**

Request

```
[NETSPARKER] SSL Connection
```

- **Response**



Response

Response Time (ms) : 1 Total Bytes Received : 27 Body Length : 0 Is Compressed : No

```
[NETSPARKER] SSL Connection
```

Vulnerability finding using namp

Used command

- nmap cnsz03nsbc01pp.aswatson.com

```
┌──(malmi㉿kali)-[~/Desktop]
└─$ nmap cnsz03nsbc01pp.aswatson.com
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-10-09 01:25 EDT
Nmap scan report for cnsz03nsbc01pp.aswatson.com (120.31.161.33)
Host is up (0.49s latency).
rDNS record for 120.31.161.33: ns2.eflydns.net
Not shown: 997 filtered tcp ports (no-response)
PORT     STATE SERVICE
25/tcp   open  smtp
443/tcp  open  https
4443/tcp open  pharos

Nmap done: 1 IP address (1 host up) scanned in 113.12 seconds
```

Founded open port

- 25/tcp open smtp
- 443/tcp open https
- 4443/tcp open pharos

Vulnerability finding using openssl

- **Used command**
    - openssl s_client -connect cnsz03nsbc01pp.aswatson.com:443

```
┌──(malmi㉿kali)-[~]
└─$ openssl s_client -connect cnsz03nsbc01pp.aswatson.com:443

CONNECTED(00000003)
depth=0 CN = 120.31.161.33
verify error:num=18:self-signed certificate
verify return:1
depth=0 CN = 120.31.161.33
verify error:num=26:unsuitable certificate purpose
verify return:1
depth=0 CN = 120.31.161.33
verify return:1
---
Certificate chain
 0 s:CN = 120.31.161.33
   i:CN = 120.31.161.33
   a:PKEY: rsaEncryption, 2048 (bit); sigalg: RSA-SHA256
   v:NotBefore: Aug  9 09:26:31 2022 GMT; NotAfter: Nov 10 09:26:31 2024 GMT
---
Server certificate
-----BEGIN CERTIFICATE-----
MIIC0DCCAbigAwIBAgIJANJrBfCFMgR2MA0GCSqGSIb3DQEBCwUAMBgxFjAUBgNV
BAMTDTEyMC4zMS4xNjEuMzMwHhcNMjIwODA5MDkyNjMxWhcNMjQxMTEwMDkyNjMx
WjAYMRYwFAYDVQQDEw0xMjAuMzEuMTYxLjMzMIIBIjANBgkqhkiG9w0BAQEFAAOC
AQ8AMIIBCgKCAQEAwuF28sA2sXDDh2sZcNKiOD3URqcqZmXQPxi2XaovanPmmtRs
1zKndiTbCDNY/qts7AE5k8uLxKTk7RqD5P2599lnSCc1P7/eSiRUzmx7pGfUXjvU
/a5jIw82hPnPFWrJ+VRooYdXOXpFV0sW2JHCEDAHOypuxaotgtHpoz7mIlDceDgp
c21mGZJ40WUUy5gWsTX+F77qczO7Ik0AHTZrxMqmHSaCD9N8iePy+VNkveFxdONH
Y3ENR4fgK84YvC768SWfNY1Vtq8417dvtsk+DUnsCd11u0exKRYEq+9Nr4czKWsn
9SV+hTf+0dr1k2vwNDAHFlZVJhjDNTgQuhP8OQIDAQABox0wGzAMBgNVHRMEBTAD
AQH/MAsGA1UdDwQEAwICBDANBgkqhkiG9w0BAQsFAAOCAQEAQ+IZ1Ct2pQzqGngt
+BORNgWQILw1ZKLuknGhnhJNGHIs7LoR6oCXc3Cyg4SW3NiuGVLk04p5dQym/JFO
bTMGaGoEJnqNJeYVTNl7CJqFDMp2IxQR4owzFjUaNbGTTCgFfS0jCVHnM0nEStpf
NZ/7CIzbinLlUa14KF8mGirwkjeC7QiI0yZ3Y4ItQ3rLbGC1U2TQwsBAwR0ILgH/
SmJetppbjZ0CvEJSBZHbnuxOSDVUt0lSHwPBOMoUtt6LbQkEhNImraRcmulSjWIq
```

- **Results**

  The certificate shows a CN of 120.31.161.33, an IP address, instead of the site's domain name (like cnsz03nsbc01pp.aswatson.com). This mismatch can cause trust issues for users and browsers.

Proposed mitigation or fix

Remedy

Fix the problem with your SSL certificate to provide secure communication between your website and its visitors.

The following will be helpful in mitigating the Invalid SSL Certificate vulnerability

- **Acquiring a Valid SSL Certificate:**
  Instead of using a self-generated certificate, get one from a trusted Certificate Authority. This ensures that the browser and users will confirm that it is valid. Some services, such as Let's Encrypt, provide free SSL certificates that most browsers trust

- **Correct Configuration of the Certificate:**
  The common name or the alternative name of the subject for the certificate should match your website's domain name at the time of installation. This would avoid any warning mismatches and build confidence in the users

- **Enable Certificate Chain:**
  Make sure the entire certificate chain is installed on the server. The chain should include all the intermediate certificates required to verify the trust relationship. It may result in failure during the process of validation if the chain is incomplete.

- **Renew and Update the Certificates Periodically:**
  Keep your SSL certificate fresh by renewing it before expiration. Expired or outdated certificates show warnings of insecurity and low trust

- **Regular Security Audits:**
  Periodically analyze your implementation of SSL/TLS with the help of tools such as SSL Labs or Netsparker. These tools can discover weaknesses or misconfigurations in your SSL/TLS setup

- **Enforce HSTS (HTTP Strict Transport Security**):
  Set up HSTS on your server to enforce all secure connections. This will tell the browser to only connect over HTTPS, reducing man-in-the-middle attacks

- **Inform Users About Security:**
  Tell the users why security is important, and how to ensure the certificate being shown is valid. It makes users believe in the security of your site.