TECHNOLOGY TRACK: AI FOR CYBERSECURITY WITH IBM QRADAR

PROJECT TITLE: Design an AI-driven incident response platform that assists cybersecurity teams in automating incident triage and response tasks.

TEAM NUMBER: 6.1

TEAM MEMBERS:

- Raghavendra Reddy Orra
- Greeshma Reddy Basireddy
- Shaik Muhammed Faizaan Ali
- Farzeen Naiz

---

1. **Vulnerability name:** Cross-site scripting (DOM based XSS)

   CWE: 79

   OWASP category: A03: Injections

   Description: It occurs when a malicious script is injected directly into a vulnerable web application. Reflected XSS involves the reflection of a malicious script of a web application, onto a user's browser.

   DOM-XSS: DOM stands for Document Object Model and is a programming interface for HTML and XML documents. It represents that programs can change the document structure, style and content. A webpage is a document and this document can be either displayed in the browser window or as the HTML Source.

   Business Impact: The application stores dangerous data in a database, message forum, visitor log, or other trusted data store. Later, the dangerous data is subsequently read back into the application and included in dynamic content. From an attacker's perspective, the optimal place to inject malicious content is in an area displayed to many users or particularly interesting users. Interesting users typically have elevated privileges in the application or interact with sensitive data that is valuable to the attacker. If one of these users executes malicious content, the attacker may be able to perform privileged operations on behalf of the user or gain access to sensitive data belonging to the user.

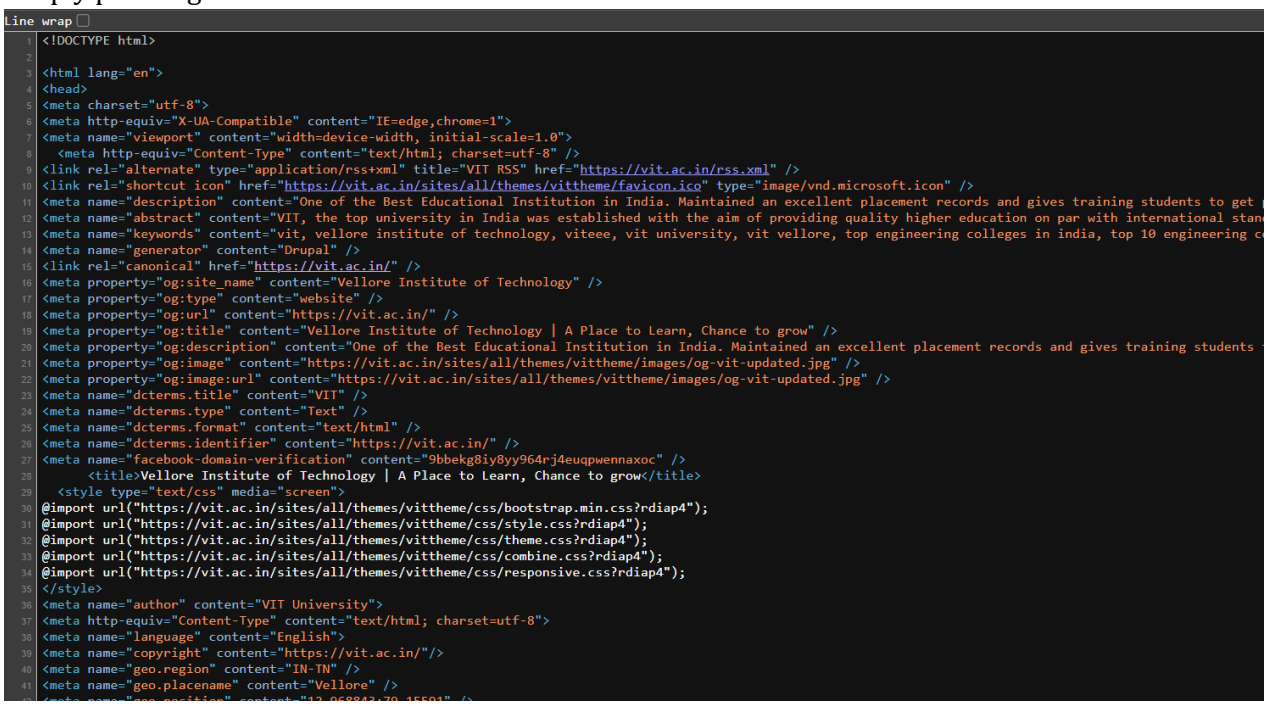
   Vulnerability path: https://vit.ac.in/

   Vulnerability parameter: view-source:https://vit.ac.in/

   Steps to Reproduce:

   - Access the URL

- We will try to perform the DOM XSS attack which is displaying the source code of the webpage.
- Now we can access the source code of the webpage in two ways by clicking on the inspect option or by simply pressing Ctrl + U.



```
Line wrap
<!DOCTYPE html>

<html lang="en">
<head>
<meta charset="utf-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<link rel="alternate" type="application/rss+xml" title="VIT RSS" href="https://vit.ac.in/rss.xml" />
<link rel="shortcut icon" href="https://vit.ac.in/sites/all/themes/vittheme/favicon.ico" type="image/vnd.microsoft.icon" />
<meta name="description" content="One of the Best Educational Institution in India. Maintained an excellent placement records and gives training students to get
<meta name="abstract" content="VIT, the top university in India was established with the aim of providing quality higher education on par with international stan
<meta name="keywords" content="vit, vellore institute of technology, viteee, vit university, vit vellore, top engineering colleges in india, top 10 engineering c
<meta name="generator" content="Drupal" />
<link rel="canonical" href="https://vit.ac.in/" />
<meta property="og:site_name" content="Vellore Institute of Technology" />
<meta property="og:type" content="website" />
<meta property="og:url" content="https://vit.ac.in/" />
<meta property="og:title" content="Vellore Institute of Technology | A Place to Learn, Chance to grow" />
<meta property="og:description" content="One of the Best Educational Institution in India. Maintained an excellent placement records and gives training students
<meta property="og:image" content="https://vit.ac.in/sites/all/themes/vittheme/images/og-vit-updated.jpg" />
<meta property="og:image:url" content="https://vit.ac.in/sites/all/themes/vittheme/images/og-vit-updated.jpg" />
<meta name="dcterms.title" content="VIT" />
<meta name="dcterms.type" content="Text" />
<meta name="dcterms.format" content="text/html" />
<meta name="dcterms.identifier" content="https://vit.ac.in/" />
<meta name="facebook-domain-verification" content="9bbekg8iy8yy964rj4euqpwennaxoc" />
      <title>Vellore Institute of Technology | A Place to Learn, Chance to grow</title>
  <style type="text/css" media="screen">
@import url("https://vit.ac.in/sites/all/themes/vittheme/css/bootstrap.min.css?rdiap4");
@import url("https://vit.ac.in/sites/all/themes/vittheme/css/style.css?rdiap4");
@import url("https://vit.ac.in/sites/all/themes/vittheme/css/theme.css?rdiap4");
@import url("https://vit.ac.in/sites/all/themes/vittheme/css/combine.css?rdiap4");
@import url("https://vit.ac.in/sites/all/themes/vittheme/css/responsive.css?rdiap4");
</style>
<meta name="author" content="VIT University">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<meta name="language" content="English">
<meta name="copyright" content="https://vit.ac.in/"/>
<meta name="geo.region" content="IN-TN" />
<meta name="geo.placename" content="Vellore" />
```

- Many programs can help to alter this source code to find vulnerable spots.

- So, Getting access to the source code of the webpage can lead to potential vulnerabilities as attackers can abuse the loop holes, exploits, vulnerabilities in the code and act accordingly.

Recommendation:

- When interacting with databases, use parameterized statements or prepared statements to avoid SQL injection, which can be a vector for XSS.
- Sanitize and validate all user inputs and ensure that any data displayed on the web page is properly encoded. Use output encoding libraries or functions to prevent script injection.
- Avoid using DOM manipulation methods that can introduce vulnerabilities. Be cautious with inner HTML, document. Write, and other methods that can execute scripts.
- Consider using well-established JavaScript frameworks and libraries that include built-in security features to prevent DOM XSS.

2. **Vulnerability name:** JQuery 1.2 < 3.5.0 Multiple XSS

CWE: 79

OWASP category: A03: Injections

Description: It occurs when a malicious script is injected directly into a vulnerable web application. Reflected XSS involves the reflection of a malicious script of a web application, onto a user's browser.

Stored XSS (Persistent XSS): In this type of attack, the malicious script is permanently stored on a server or in a database, often in user-generated content like comments or forum posts. When other users view the affected content, the script executes in their browsers, potentially compromising their data or sessions.

Reflected XSS: Reflected XSS occurs when the malicious script is embedded in a URL and immediately executed when a victim clicks on a manipulated link. The script is not stored on a server but is reflected off a web application, making it a one-time attack.

DOM-based XSS: DOM stands for Document Object Model and is a programming interface for HTML and XML documents. It represents that programs can change the document structure, style and content. A webpage is a document and this document can be either displayed in the browser window or as the HTML Source.

Business Impact: The application stores dangerous data in a database, message forum, visitor log, or other trusted data store. Later, the dangerous data is subsequently read back into the application and included in dynamic content. From an attacker's perspective, the optimal place to inject malicious content is in an area displayed to many users or particularly interesting users. Interesting users typically have elevated privileges in the application or interact with sensitive data that is valuable to the attacker. If one of these users executes malicious content, the attacker may be able to perform privileged operations on behalf of the user or gain access to sensitive data belonging to the user.

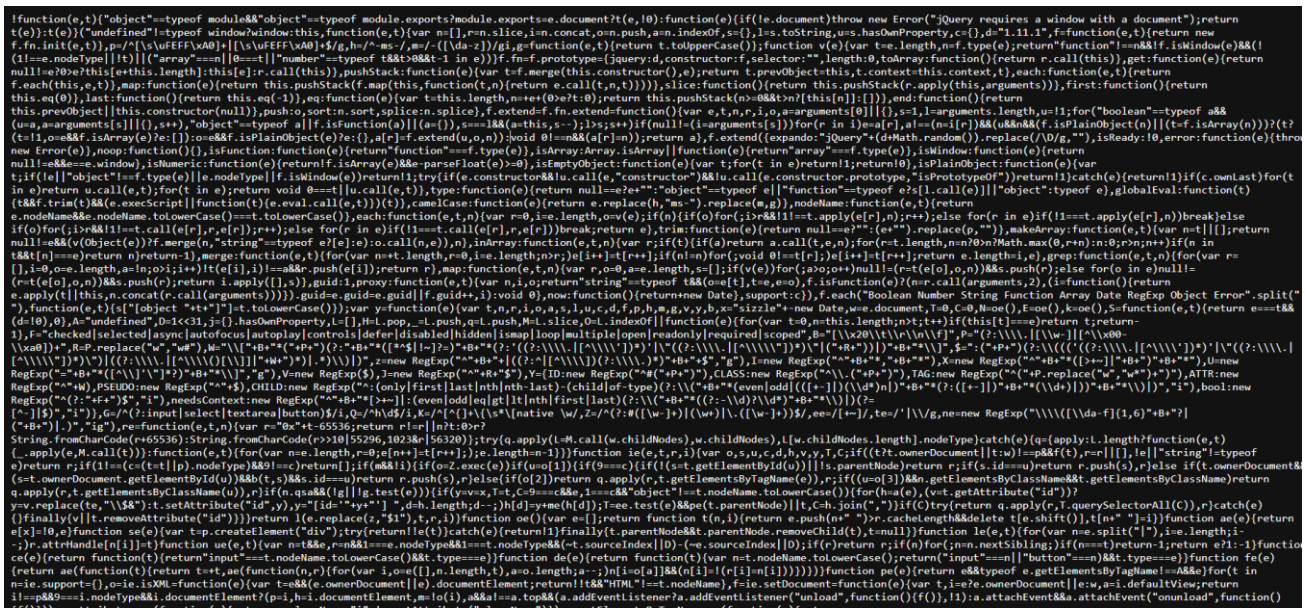
Vulnerability path: https://vit.ac.in/

Vulnerability parameter: https://vit.ac.in/sites/all/themes/vittheme/js/jquery-2.1.1.min.js?rdiap4

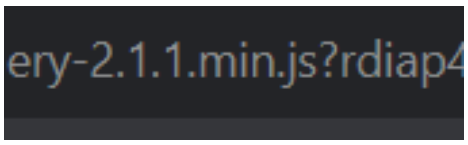
Steps to Reproduce:

- Access the URL

- Try to access the URL https://vit.ac.in/sites/all/themes/vittheme/js/jquery- . This will lead us to the webpage with jquery script in it .



- We can also see the jquery version that has been used in the website



- We can access the source code of the jquery script that has been used in making the website, which can lead to multiple potential threats

Recommendation:

- The most important recommendation is to update to the latest version of jQuery. Newer versions typically include security patches, bug fixes, and performance improvements.
- Stay vigilant about jQuery updates and security announcements. Subscribe to relevant security mailing lists or forums to receive notifications about new releases.
- A Content Security Policy (CSP) can help mitigate various types of client-side vulnerabilities, including Cross-Site Scripting (XSS). Consider implementing a CSP in your web application to restrict the sources from which scripts can be loaded.
- If updating jQuery is not immediately possible, review your code for potential security issues. Ensure that you are escaping or sanitizing user inputs and that you are using safe coding practices to prevent XSS and other security vulnerabilities.

---

3. **Vulnerability name:** Information Exposure (Web Server HTTP Header Internal IP Disclosure)

CWE: 200

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

Business Impact: To effectively reduce the business consequences associated with CWE-200, organizations must prioritize the implementation of robust data access controls, secure data handling practices, and privacy protection measures. Regular security assessments and testing are instrumental in detecting and rectifying

vulnerabilities related to information exposure. Furthermore, strict adherence to data protection regulations and industry standards is indispensable. These actions are of paramount importance in the protection of sensitive data, the preservation of user trust, and the upholding of the organization's esteemed reputation.

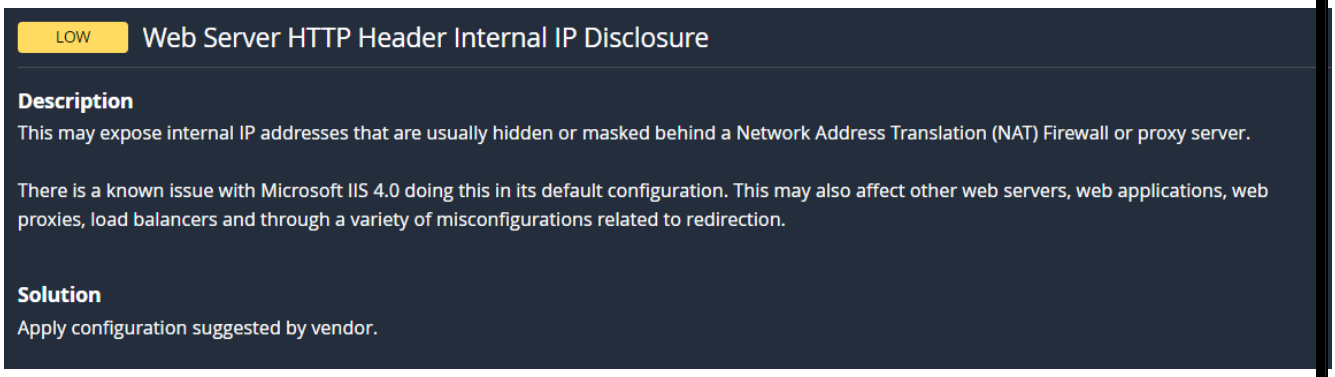
Vulnerability path: https://vit.ac.in/

Vulnerability parameter: https://vit.ac.in/

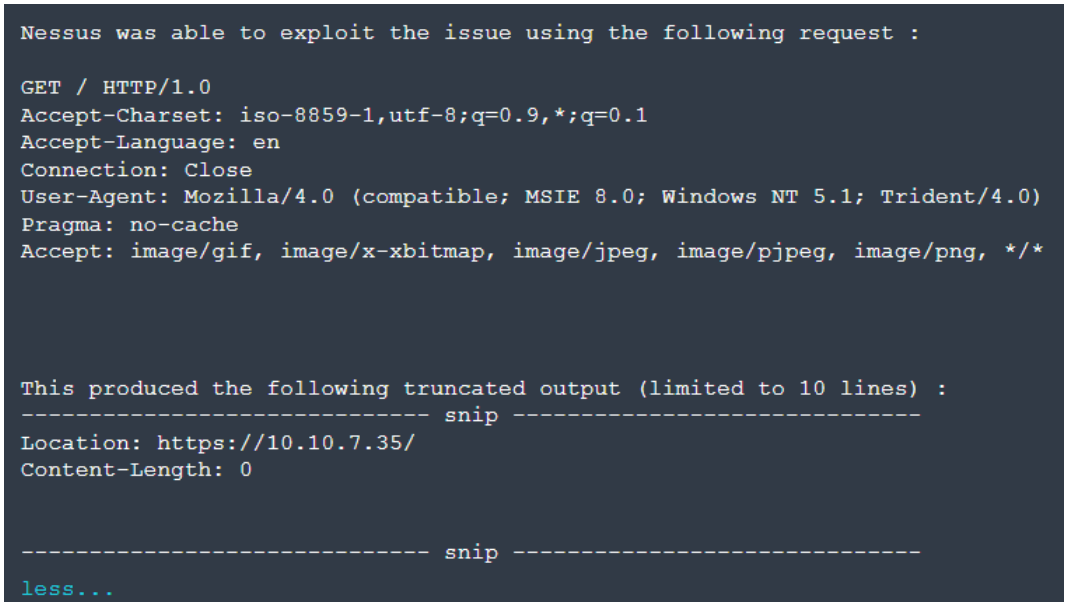
Steps to Reproduce:

- Access the URL

- We will be using a tool to perform a scan on this website. (Nessus Scanner)



- Through this tool we can see the vulnerability in the scan along with detailed information regarding that vulnerability.



- Now this image tells us the internal IP address of the target website, the attackers can easily exploit the website using the ip address.

Recommendation:

- Carefully review your web server's configuration files (e.g., Apache's httpd.conf, Nginx's nginx.conf) to identify anywhere internal IP addresses are inadvertently included in HTTP response headers.
- Remove or replace any internal IP addresses in the server configuration files with appropriate placeholders or public IP addresses.
- If you use proxy servers or load balancers, ensure that they are configured to hide internal IP addresses and only expose public IP addresses instances in HTTP headers.
- Review and configure HTTP response headers (e.g., Server, X-Powered-By) to limit the exposure of server and technology details, reducing the potential for disclosing internal IP addresses.

---

4. **Vulnerability name:** Information Exposure (SSL (Multiple Issues))

CWE: 200

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

Business Impact: To effectively reduce the business consequences arising from SSL issues, organizations must give utmost priority to establishing a secure SSL/TLS configuration, conducting frequent security assessments and testing. Adhering to data protection regulations and industry standards is imperative. These actions play a pivotal role in safeguarding sensitive data, upholding user trust, and safeguarding the organization's esteemed reputation.

Vulnerability path: https://vit.ac.in/

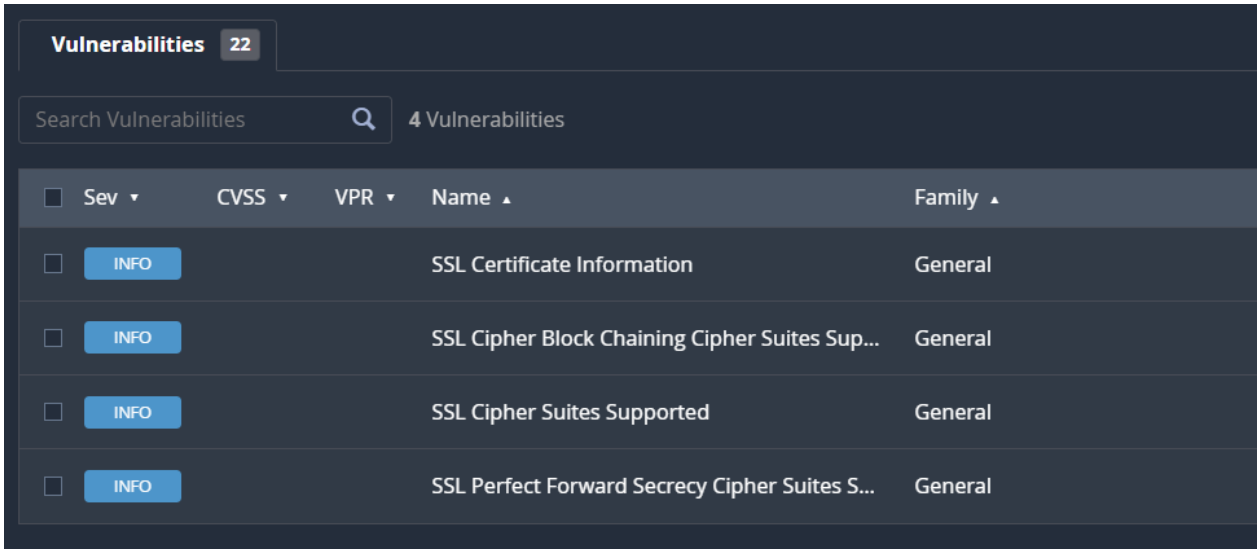
Vulnerability parameter: view-source: https://vit.ac.in/

Steps to Reproduce:

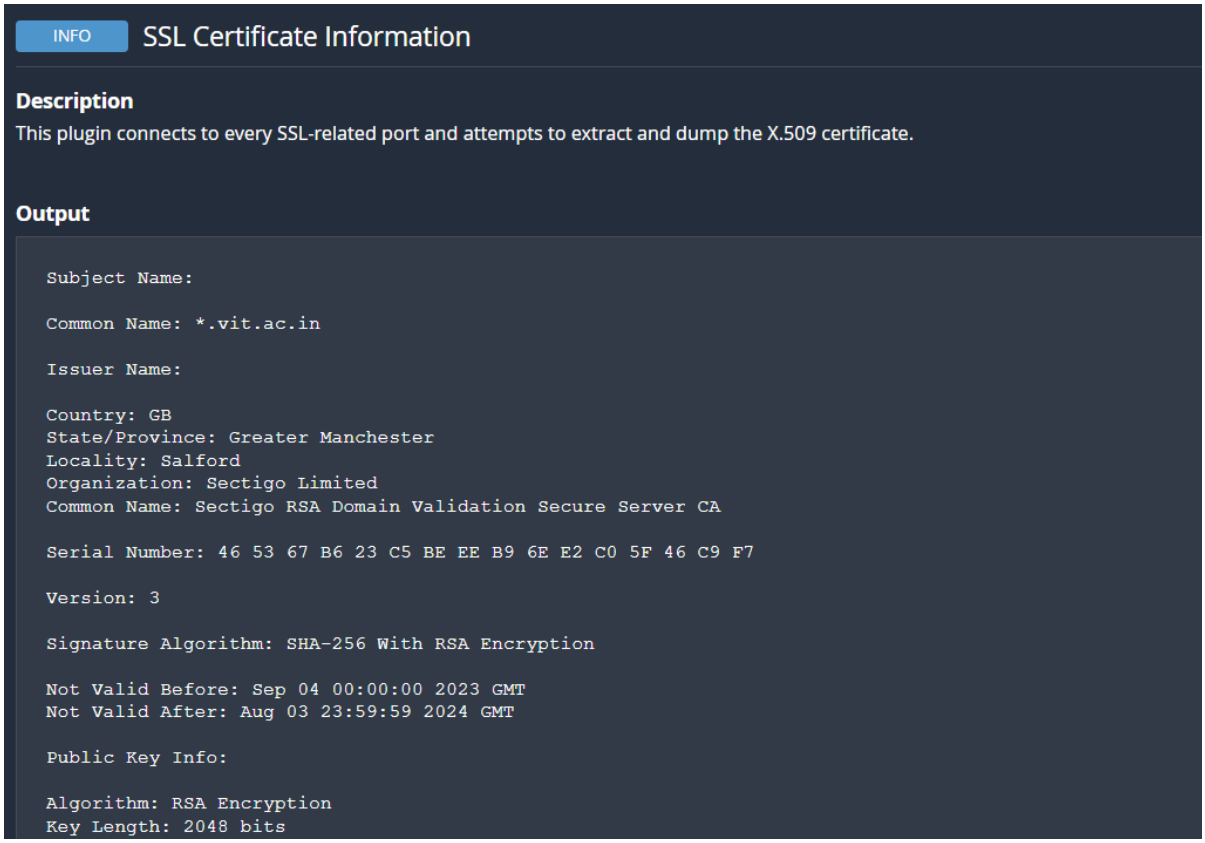
Steps to Reproduce:

- Access the URL

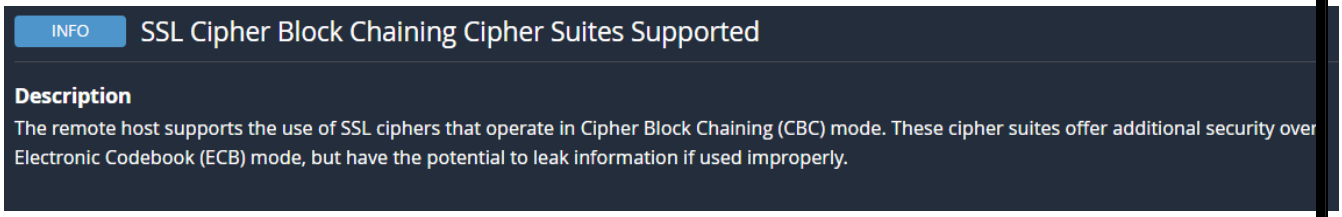
- We will be using a tool to perform a scan on this website. (Nessus Scanner)



- Through this tool we can see the vulnerability in the scan along with detailed information regarding that vulnerability.



- Now this image tells us the type of encryption used in the target website with the public key details as well.



INFO **SSL Cipher Suites Supported**

Description

This plugin detects which SSL ciphers are supported by the remote service for encrypting communications.

INFO **SSL Perfect Forward Secrecy Cipher Suites Supported**

Description

The remote host supports the use of SSL ciphers that offer Perfect Forward Secrecy (PFS) encryption. These cipher suites ensure that recorded SSL traffic cannot be broken at a future date if the server's private key is compromised.

- Through this we can understand what different types of encryptions are used in the website.

Here is the list of SSL CBC ciphers supported by the remote server :

High Strength Ciphers (>= 112-bit key)

Name	Code	KEX	Auth	Encryption	MAC
ECDHE-RSA-CAMELLIA-CBC-128	0xC0, 0x76	ECDH	RSA	Camellia-CBC(128)	SHA256
ECDHE-RSA-CAMELLIA-CBC-256	0xC0, 0x77	ECDH	RSA	Camellia-CBC(256)	SHA384
DHE-RSA-AES128-SHA	0x00, 0x33	DH	RSA	AES-CBC(128)	SHA1
DHE-RSA-AES256-SHA	0x00, 0x39	DH	RSA	AES-CBC(256)	SHA1
DHE-RSA-CAMELLIA128-SHA	0x00, 0x45	DH	RSA	Camellia-CBC(128)	SHA1
DHE-RSA-CAMELLIA256-SHA	0x00, 0x88	DH	RSA	Camellia-CBC(256)	SHA1
DHE-RSA-SEED-SHA	0x00, 0x9A	DH	RSA	SEED-CBC(128)	SHA1
ECDHE-RSA-AES128-SHA	0xC0, 0x13	ECDH	RSA	AES-CBC(128)	SHA1
ECDHE-RSA-AES256-SHA	0xC0, 0x14	ECDH	RSA	AES-CBC(256)	SHA1
AES128-SHA	0x00, 0x2F	RSA	RSA	AES-CBC(128)	SHA1
AES256-SHA	0x00, 0x35	RSA	RSA	AES-CBC(256)	SHA1
CAMELLIA128-SHA	0x00, 0x41	RSA	RSA	Camellia-CBC(128)	SHA1
CAMELLIA256-SHA	0x00, 0x84	RSA	RSA	Camellia-CBC(256)	SHA1
SEED-SHA	0x00, 0x96	RSA	RSA	SEED-CBC(128)	SHA1
DHE-RSA-AES128-SHA256	0x00, 0x67	DH	RSA	AES-CBC(128)	SHA256
DHE-RSA-AES256-SHA256	0x00, 0x6B	DH	RSA	AES-CBC(256)	SHA256
DHE-RSA-CAMELLIA128-SHA256	0x00, 0xBE	DH	RSA	Camellia-CBC(128)	SHA256
DHE-RSA-CAMELLIA256-SHA256	0x00, 0xC4	DH	RSA	Camellia-CBC(256)	SHA256
ECDHE-RSA-AES128-SHA256	0xC0, 0x27	ECDH	RSA	AES-CBC(128)	SHA256
ECDHE-RSA-AES256-SHA384	0xC0, 0x28	ECDH	RSA	AES-CBC(256)	SHA384
RSA-AES128-SHA256	0x00, 0x3C	RSA	RSA	AES-CBC(128)	SHA256
RSA-AES256-SHA256	0x00, 0x3D	RSA	RSA	AES-CBC(256)	SHA256
RSA-CAMELLIA128-SHA256	0x00, 0xBA	RSA	RSA	Camellia-CBC(128)	SHA256
RSA-CAMELLIA256-SHA256	0x00, 0xC0	RSA	RSA	Camellia-CBC(256)	SHA256

Recommendation:

- Configure your web server to use the latest and strongest encryption protocols, such as TLS 1.2 or higher. Disable outdated and insecure protocols like SSLv3.
- Implement Perfect Forward Secrecy to ensure that even if the private key is compromised, previously recorded encrypted traffic remains secure.
- Safeguard your private keys by storing them securely and ensuring they are not accessible to unauthorized parties.

5. **Vulnerability name:** Information Exposure (HTTP (Multiple Issues))

CWE: 200

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

Business Impact: To effectively reduce the business repercussions of HTTP issues, organizations must place a strong emphasis on ongoing monitoring and enhancement of their web services. Consistent performance testing and security assessments, coupled with proactive maintenance, are indispensable to ensure a seamless and secure online presence. Furthermore, offering timely and helpful customer support and transparent communication when HTTP issues arise is instrumental in mitigating adverse effects on user trust and satisfaction.

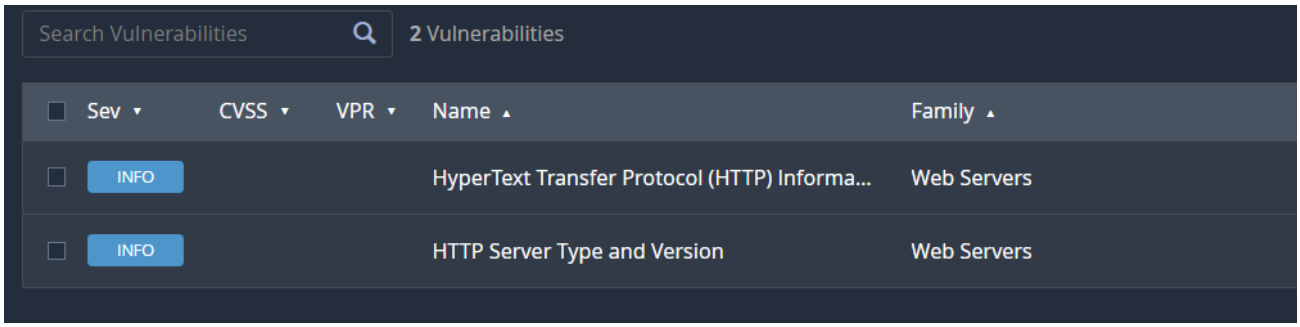
Vulnerability path: https://vit.ac.in/

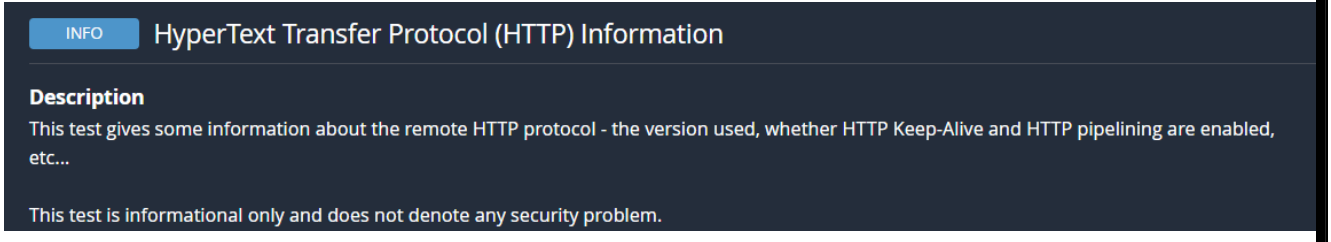
Vulnerability parameter: https://vit.ac.in/

Steps to Reproduce:

- Access the URL

- We will be using a tool to perform a scan on this website. (Nessus Scanner)





```
Response Code : HTTP/1.1 307 Moved Temporarily

Protocol version : HTTP/1.1
SSL : no
Keep-Alive : no
Options allowed : (Not implemented)
Headers :

  Location: https://vit.ac.in/
  Content-Length: 0

Response Body :
```

- From the above image we can see the details of protocol version of the website and its status.

INFO **HTTP Server Type and Version**

Description

This plugin attempts to determine the type and the version of the remote web server.

```
The remote web server type is :

Apache
```

- We can see the type of remote webserver here.
- All this information can lead to multiple potential threats such as data breaches by hackers.

Recommendation:

- Minimize the use of unnecessary redirects as they can increase page load times. Use 301 (permanent) redirects for consistent content.
- Regularly check for broken links (404 errors) and fix them to ensure users find the content they are looking for.
- Properly configure CORS to control which domains are allowed to make requests to your server.
- Migrate your website from HTTP to HTTPS to ensure secure data transmission. This is essential for user trust and search engine ranking.

---

6. **Vulnerability name:** OS Identification Failed

CWE: 200

Description: OS identification failed refers to the inability of a system or software to accurately recognize the operating system running on a device or network. This issue can result from various factors such as misconfigurations, network issues, or compatibility problems. The failure to identify the OS correctly can lead to security vulnerabilities, hindered software updates, and potential operational disruptions. Resolving this issue promptly is crucial to ensure accurate security measures and the smooth functioning of the system or network.

Business Impact: It may lead to operational inefficiencies, potential security vulnerabilities, non-compliance with industry regulations, and a possible loss of customer trust. Such failures could result in downtime, compromised data security, regulatory penalties, and damage to the company's reputation, potentially

leading to financial losses and decreased market competitiveness. Resolving this issue promptly is crucial to mitigate these risks and maintain the smooth functioning of the organization.

Vulnerability path: https://vit.ac.in/

Vulnerability parameter: https://vit.ac.in/

Steps to Reproduce:

- Access the URL

- We will be using a tool to perform a scan on this website. (Nessus Scanner)

INFO **OS Identification Failed**

Description

Using a combination of remote probes (TCP/IP, SMB, HTTP, NTP, SNMP, etc), it was possible to gather one or more fingerprints from the remote system. Unfortunately, though, Nessus does not currently know how to use them to identify the overall system.

```
If you think these signatures would help us improve OS fingerprinting,
please send them to :

  os-signatures@nessus.org

Be sure to include a brief description of the device itself, such as
the actual operating system or product / model names.


SSLcert:!:i/CN:Sectigo RSA Domain Validation Secure Server CAi/O:Sectigo
Limiteds/CN:*.vit.ac.in
02ed800c9e7f764615d7e9ffd74b643a6cace446

SinFP:!:
   P1:B10113:F0x12:W65535:O0204ffff:M496:
   P2:B10113:F0x12:W65535:O0204ffff010303030101040201010 80affffffff44454144:M1300:
   P3:B00000:F0x00:W0:O0:M0
   P4:190701_7_p=80R
```

- From the above image, we can see the details of SSL Certificate Information and SinFP Information.
- If it is released without proper authorization, it can lead to serious consequences such as increased vulnerability to cyberattacks, potential data breaches, privacy violations, and legal repercussions.

- All this information can lead to multiple potential threats such as data breaches by hackers.

Recommendation:

- Minimize the use of unnecessary redirects as they can increase page load times. Use 301 (permanent) redirects for consistent content.
- Regularly check for broken links (404 errors) and fix them to ensure users find the content they are looking for.
- Properly configure CORS to control which domains are allowed to make requests to your server.
- Migrate your website from HTTP to HTTPS to ensure secure data transmission. This is essential for user trust and search engine ranking.