MPCDF ERP - Security Test Report

invicti

12/8/2023 11:59:09 AM (UTC+05:30)

Detailed Scan Report

https://mpcdf.tserver.co.in/mis/Login.aspx

Scan Time	: 12/8/2023 11:57:52 AM (UTC+05:30)
Scan Duration	: 00:00:00:52
Total Requests	: 131
Average Speed	: 2.5r/s

Risk Level:
MEDIUM

16 IDENTIFIED	5 CONFIRMED	0 CRITICAL
0 HIGH	3 MEDIUM	6 LOW
	4 BEST PRACTICE	3 INFORMATION

Identified Vulnerabilities

Critical	0
High	0
Medium	3
Low	6
Best Practice	4
Information	3
TOTAL	**16**

Confirmed Vulnerabilities

Critical	0
High	0
Medium	2
Low	1
Best Practice	2
Information	0
TOTAL	**5**

Vulnerability Summary

CONFIRM	VULNERABILITY	METHOD	URL	PARAMETER	PARAMETER TYPES
	HTTP Strict Transport Security (HSTS) Policy Not Enabled	GET	https://mpcdf.tserver.co.in/mis/Login.aspx	No Parameters	No Parameter Types
	Session Cookie Not Marked as Secure	POST	https://mpcdf.tserver.co.in/mis/Login.aspx	No Parameters	No Parameter Types
	Weak Ciphers Enabled	GET	https://mpcdf.tserver.co.in/mis/Login.aspx	No Parameters	No Parameter Types
	[Possible] Cross-site Request Forgery in Login Form	GET	https://mpcdf.tserver.co.in/mis/Login.aspx	No Parameters	No Parameter Types
	Missing X-Frame-Options Header	GET	https://mpcdf.tserver.co.in/mis/Login.aspx	No Parameters	No Parameter Types
	Version Disclosure (ASP.NET)	GET	https://mpcdf.tserver.co.in/mis/Login.aspx	No Parameters	No Parameter Types
	Version Disclosure (IIS)	GET	https://mpcdf.tserver.co.in/mis/Login.aspx	No Parameters	No Parameter Types
	ViewState is not Encrypted	GET	https://mpcdf.tserver.co.in/mis/Login.aspx	No Parameters	No Parameter Types
	Insecure Transportation Security Protocol Supported (TLS 1.0)	GET	https://mpcdf.tserver.co.in/mis/Login.aspx	No Parameters	No Parameter Types
	Missing X-XSS-Protection Header	GET	https://mpcdf.tserver.co.in/mis/Login.aspx	No Parameters	No Parameter Types
	Referrer-Policy Not Implemented	GET	https://mpcdf.tserver.co.in/mis/Login.aspx	No Parameters	No Parameter Types

CONFIRM	VULNERABILITY	METHOD	URL	PARAMETER	PARAMETER TYPES
	Content Security Policy (CSP) Not Implemented	GET	https://mpcdf.tserver.co.in/mis/Login.aspx	No Parameters	No Parameter Types
	Insecure Transportation Security Protocol Supported (TLS 1.1)	GET	https://mpcdf.tserver.co.in/mis/Login.aspx	No Parameters	No Parameter Types
	[Possible] Login Page Identified	GET	https://mpcdf.tserver.co.in/mis/Login.aspx	No Parameters	No Parameter Types
	ASP.NET Identified	GET	https://mpcdf.tserver.co.in/mis/Login.aspx	No Parameters	No Parameter Types
	IIS Identified	GET	https://mpcdf.tserver.co.in/mis/Login.aspx	No Parameters	No Parameter Types

1. HTTP Strict Transport Security (HSTS) Policy Not Enabled

MEDIUM | 1

Invicti Standard identified that HTTP Strict Transport Security (HSTS) policy is not enabled.

The target website is being served from not only HTTPS but also HTTP and it lacks of HSTS policy implementation.

HTTP Strict Transport Security (HSTS) is a web security policy mechanism whereby a web server declares that complying user agents (such as a web browser) are to interact with it using only secure (HTTPS) connections. The HSTS Policy is communicated by the server to the user agent via a HTTP response header field named "Strict-Transport-Security". HSTS Policy specifies a period of time during which the user agent shall access the server in only secure fashion.

When a web application issues HSTS Policy to user agents, conformant user agents behave as follows:

- Automatically turn any insecure (HTTP) links referencing the web application into secure (HTTPS) links. (For instance, http://example.com/some/page/ will be modified to https://example.com/some/page/ before accessing the server.)
- If the security of the connection cannot be ensured (e.g. the server's TLS certificate is self-signed), user agents show an error message and do not allow the user to access the web application.

Vulnerabilities

1.1. https://mpcdf.tserver.co.in/mis/Login.aspx

Certainty

Request | Response

Request

```
GET /mis/Login.aspx HTTP/1.1
Host: mpcdf.tserver.co.in
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Accept-Encoding: gzip, deflate
Accept-Language: en-us,en;q=0.5
Cache-Control: no-cache
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/108.0.5359.71 Safari/537.36
```

Response

Response Time (ms) : 22.8161
Total Bytes Received : 27133
Body Length : 26868
Is Compressed : Yes

```
HTTP/1.1 200 OK
Server: Microsoft-IIS/10.0
X-Powered-By: ASP.NET
Vary: Accept-Encoding
X-AspNet-Version: 4.0.30319
Content-Length: 6708
Content-Type: text/html; charset=utf-8
Content-Encoding:
Date: Fri, 08 Dec 2023 06:28:20 GMT
Cache-Control: private



<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head><meta charset="utf-8" /><meta http-equiv="X-UA-Compatible" content="IE=edge" /><title>
MPCDF
</title><link rel="shortcut icon" href="image/favicon.ico" type="image/ico" /><meta
content="width=device-width, initial-scale=1, maximum-scale=1, user-scalable=no" name="viewport" />
<link href="css/bootstrap.css" type="text/css" rel="stylesheet" /><link href="css/AdminLTE.css"
rel="stylesheet" type="text/css" /><link href="css/blue.css" rel="stylesheet" type="text/css" />
<link href="css/font-awesome/css/font-awesome.css" type="text/css rel="stylesheet" />
<!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and media queries -->
<!-- WARNING: Respond.js doesn't work if you view the page via file:// -->
<!--[if lt IE 9]>
<script src="https://oss.maxcdn.com/html5shiv/3.7.3/html5shiv.min.js"></script>
<script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
<![endif]-->
<style type="text/css">
body {
background: #0074e2 url(image/login_left_bg.jpg);
}

.content-wrapper {
margin-left: 0;
color: #fff;
}

label {
color: #fff;
}

.row {
```

```
display: flex;
flex-wrap: wrap;
margin-right: -15px;
margin-left: -15px;
}

.align-self-end {
align-self: flex-end !important;
}

.form-group label {
line-height: 1.4rem;
vertical-align: top;
margin-bottom: .5rem;
}


.input-group-text {
font-weight: 400;

…
```

Remedy

Configure your webserver to redirect HTTP requests to HTTPS.

i.e. for Apache, you should have modification in the httpd.conf. For more configurations, please refer to External References section.

```
# load module
LoadModule headers_module modules/mod_headers.so

# redirect all HTTP to HTTPS (optional)
<VirtualHost *:80>
       ServerAlias *
       RewriteEngine On
       RewriteRule ^(.*)$ https://%{HTTP_HOST}$1 [redirect=301]
</VirtualHost>

# HTTPS-Host-Configuration
<VirtualHost *:443>
      # Use HTTP Strict Transport Security to force client to use secure connections only
      Header always set Strict-Transport-Security "max-age=31536000; includeSubDomains"

      # Further Configuration goes here
      [...]
</VirtualHost>
```

External References

- Wikipedia - HTTP Strict Transport Security
- Configure HSTS (HTTP Strict Transport Security) for Apache/Nginx

- HTTP Strict Transport Security (HSTS) HTTP Header
- Mozilla SSL Configuration Generator

CLASSIFICATION

OWASP 2013	A6
OWASP 2017	A3
CWE	523
CAPEC	217
WASC	4
ASVS 4.0	14.4.5
NIST SP 800-53	SC-8
DISA STIG	V-6136
ISO27001	A.14.1.2
ISO27001 2022	A.8.24
OWASP Top Ten 2021	A02

CVSS 3.0 SCORE

Base	7.7 (High)
Temporal	7.7 (High)
Environmental	7.7 (High)

CVSS Vector String

CVSS Vector String
CVSS:3.0/AV:N/AC:H/PR:N/UI:N/S:U/C:H/I:H/A:L

CVSS 3.1 SCORE	
Base	7.7 (High)
Temporal	7.7 (High)
Environmental	7.7 (High)

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

2. Session Cookie Not Marked as Secure

MEDIUM 1 | CONFIRMED 1

Invicti Standard identified a session cookie not marked as secure, and transmitted over HTTPS.

This means the cookie could potentially be stolen by an attacker who can successfully intercept the traffic, following a successful man-in-the-middle attack.

It is important to note that Invicti Standard inferred from the its name that the cookie in question is session related.

Impact

This cookie will be transmitted over a HTTP connection, therefore an attacker might intercept it and hijack a victim's session. If the attacker can carry out a man-in-the-middle attack, he/she can force the victim to make an HTTP request to your website in order to steal the cookie.

Vulnerabilities

2.1. https://mpcdf.tserver.co.in/mis/Login.aspx

CONFIRMED

Method	Parameter	Parameter Type	Value
POST	txtUserName	Post	E0001
POST	__EVENTVALIDATION	Post	/wEdAAW9mCzqwsX9O1ZLoMJDYfIvY3plgk0YBAefRz3MyBlTcDl1UbblslFK/AJeK3bZrfv3RTGY94IgNr0Jq9ytscQRop4oRunf...
POST	__EVENTTARGET	Post	
POST	txtUserPassword	Post	372895bd498fdf6ec872a5fbbaec61919011099fa91fd04fc4f5b81e62f9204cd241683fb836c1d492a99d7f3cc7e855e128...
POST	__VIEWSTATEGENERATOR	Post	BEEA8346
POST	__VIEWSTATE	Post	/wEPDwUKMTU5NjMxOTIyNQ8WAh4KUmFuZG9tVGV4dAUsYXhtMmc2WUJhNjNlWDZiS3NyMDlJclhnWFVlK0wxWmo3dFZYaHR4eDg0...
POST	btnLogin	Post	Sign In
POST	__EVENTARGUMENT	Post	

Identified Cookie(s)

- ASP.NET_SessionId

Cookie Source

- HTTP Header

Request | Response

Request

```
POST https://mpcdf.tserver.co.in/mis/Login.aspx HTTP/1.1
Content-Type: application/x-www-form-urlencoded
Origin: https://mpcdf.tserver.co.in
Referer: https://mpcdf.tserver.co.in/mis/Login.aspx
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/108.0.5359.71 Safari/537.36
Cookie: ASP.NET_SessionId=yq1ujknlrpfsvnv4waioeofm
```

Response

Response Time (ms) : 0
Total Bytes Received : 431
Body Length : 0
Is Compressed : No

```
cache-control: private
content-length: 173
content-type: text/html; charset=utf-8
date: Fri, 08 Dec 2023 06:27:54 GMT
location: /mis/Dashboard/UnionWiseProgressReport.aspx?IsMainPage=1
server: Microsoft-IIS/10.0
set-cookie: ASP.NET_SessionId=yq1ujknlrpfsvnv4waioeofm; path=/; HttpOnly; SameSite=Lax
ASP.NET_SessionId=khpwgbrf1xwwnovibzwsntkg; path=/; HttpOnly; SameSite=Lax

x-aspnet-version: 4.0.30319
x-powered-by: ASP.NET
```

Actions to Take

1. See the remedy for solution.
2. Mark all cookies used within the application as secure. *(If the cookie is not related to authentication or does not carry any personal information, you do not have to mark it as secure.)*

Remedy

Mark all cookies used within the application as secure.

Required Skills for Successful Exploitation

To exploit this issue, the attacker needs to be able to intercept traffic. This generally requires local access to the web server or to the victim's network. Attackers need to understand layer 2 and have gained access to a system between the victim and the web server.

External References

- .NET Cookie.Secure Property
- How to Create Totally Secure Cookies
- Invicti Standard - Security Cookies - Secure Flag

CLASSIFICATION

PCI DSS v3.2	6.5.10
OWASP 2013	A6
OWASP 2017	A3
CWE	614
CAPEC	102
WASC	15
ASVS 4.0	3.4.1
NIST SP 800-53	AC-16
DISA STIG	V-16786
ISO27001	A.14.1.2
ISO27001 2022	A.8.27
OWASP Top Ten 2021	A02

CVSS 3.0 SCORE

Base	5.3 (Medium)
Temporal	5.3 (Medium)
Environmental	5.3 (Medium)

CVSS Vector String
CVSS:3.0/AV:A/AC:H/PR:N/UI:N/S:U/C:H/I:N/A:N

CVSS 3.1 SCORE	
Base	5.3 (Medium)
Temporal	5.3 (Medium)
Environmental	5.3 (Medium)

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

3. Weak Ciphers Enabled

MEDIUM 1 | CONFIRMED 1

Invicti Standard detected that weak ciphers are enabled during secure communication (SSL).

You should allow only strong ciphers on your web server to protect secure communication with your visitors.

Impact

Attackers might decrypt SSL traffic between your server and your visitors.

Vulnerabilities

3.1. https://mpcdf.tserver.co.in/mis/Login.aspx

CONFIRMED

List of Supported Weak Ciphers

- TLS_RSA_WITH_3DES_EDE_CBC_SHA (0x000A)
- TLS_RSA_WITH_AES_256_CBC_SHA (0x0035)
- TLS_RSA_WITH_AES_128_CBC_SHA (0x002F)
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (0xC014)
- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (0xC013)
- TLS_RSA_WITH_AES_256_CBC_SHA256 (0x003D)
- TLS_RSA_WITH_AES_128_CBC_SHA256 (0x003C)
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 (0xC028)
- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 (0xC027)
- TLS_RSA_WITH_AES_256_GCM_SHA384 (0x009D)
- TLS_RSA_WITH_AES_128_GCM_SHA256 (0x009C)

Request | Response

Request

```
[SSL Connection]
```

Response

Response Time (ms) : 1
Total Bytes Received : 16
Body Length : 0
Is Compressed : No

```
[SSL Connection]
```

Actions to Take

1. For Apache, you should modify the SSLCipherSuite directive in the `httpd.conf`.

```
SSLCipherSuite HIGH:MEDIUM:!MD5:!RC4
```

2. Lighttpd:

```
ssl.honor-cipher-order = "enable"
ssl.cipher-list = "EECDH+AESGCM:EDH+AESGCM"
```

3. For Microsoft IIS, you should make some changes to the system registry. **Incorrectly editing the registry may severely damage your system. Before making changes to the registry, you should back up any valued data on your computer.**

   a.Click Start, click Run, type `regedt32`or type `regedit`, and then click OK.
   b.In Registry Editor, locate the following registry key: `HKLM\SYSTEM\CurrentControlSet\Control\SecurityProviders`
   c.Set "Enabled" DWORD to "0x0" for the following registry keys:

```
SCHANNEL\Ciphers\DES 56/56
SCHANNEL\Ciphers\RC4 64/128
SCHANNEL\Ciphers\RC4 40/128
SCHANNEL\Ciphers\RC2 56/128
SCHANNEL\Ciphers\RC2 40/128
SCHANNEL\Ciphers\NULL
SCHANNEL\Hashes\MD5
```

Remedy

Configure your web server to disallow using weak ciphers.

External References

- OWASP - Insecure Configuration Management
- OWASP Top 10-2017 A3-Sensitive Data Exposure
- Zombie Poodle - Golden Doodle (CBC)
- Mozilla SSL Configuration Generator
- Strong Ciphers for Apache, Nginx and Lighttpd

CLASSIFICATION

PCI DSS v3.2	6.5.4
OWASP 2013	A6
OWASP 2017	A3
CWE	327
CAPEC	217
WASC	4
ASVS 4.0	6.2.5
NIST SP 800-53	SC-8
DISA STIG	V-6136
ISO27001	A.14.1.3
ISO27001 2022	A.8.24
OWASP Top Ten 2021	A02

CVSS 3.0 SCORE

Base	6.8 (Medium)
Temporal	6.8 (Medium)
Environmental	6.8 (Medium)

CVSS Vector String
CVSS:3.0/AV:A/AC:H/PR:N/UI:N/S:U/C:H/I:H/A:N

CVSS 3.1 SCORE	
Base	6.8 (Medium)
Temporal	6.8 (Medium)
Environmental	6.8 (Medium)

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

4. [Possible] Cross-site Request Forgery in Login Form

LOW | 1

Invicti Standard identified a possible Cross-Site Request Forgery in Login Form.

In a login CSRF attack, the attacker forges a login request to an honest site using the attacker's user name and password at that site. If the forgery succeeds, the honest server responds with a Set-Cookie header that instructs the browser to mutate its state by storing a session cookie, logging the user into the honest site as the attacker. This session cookie is used to bind subsequent requests to the user's session and hence to the attacker's authentication credentials. The attacker can later log into the site with his legitimate credentials and view private information like activity history that has been saved in the account.

Impact

In this particular case CSRF affects the login form in which the impact of this vulnerability is decreased significantly. Unlike normal CSRF vulnerabilities this will only allow an attacker to exploit some complex XSS vulnerabilities otherwise it can't be exploited.

For example;

If there is a page that's different for every user (such as "edit my profile") and vulnerable to XSS (Cross-site Scripting) then normally it cannot be exploited. However if the login form is vulnerable, an attacker can prepare a special profile, force victim to login as that user which will trigger the XSS exploit. Again attacker is still quite limited with this XSS as there is no active session. However the attacker can leverage this XSS in many ways such as showing the same login form again but this time capturing and sending the entered username/password to the attacker.

In this kind of attack, attacker will send a link containing html as simple as the following in which attacker's user name and password is attached.

```
<form method="POST" action="http://honest.site/login">
  <input type="text" name="user" value="h4ck3r" />
  <input type="password" name="pass" value="passw0rd" />
</form>
<script>
    document.forms[0].submit();
</script>
```

When the victim clicks the link then form will be submitted automatically to the honest site and exploitation is successful, victim will be logged in as the attacker and consequences will depend on the website behavior.

- **Search History**
  Many sites allow their users to opt-in to saving their search history and provide an interface for a user to review his or her personal search history. Search queries contain sensitive details about the user's interests and activities and could be used by the attacker to embarrass the user, to steal the user's identity, or to spy on the user. Since the victim logs in as the attacker, the victim's search queries are then stored in the attacker's search history, and the attacker can retrieve the queries by logging into his or her own account.

- **Shopping**
  Merchant sites might save the credit card details in user's profile. In login CSRF attack, when user funds a purchase and enrolls the credit card, the credit card details might be added to the attacker's account.

Vulnerabilities

4.1. https://mpcdf.tserver.co.in/mis/Login.aspx

Form Name(s)

- form1

Certainty

Request | Response

Request

```
GET /mis/Login.aspx HTTP/1.1
Host: mpcdf.tserver.co.in
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Accept-Encoding: gzip, deflate
Accept-Language: en-us,en;q=0.5
Cache-Control: no-cache
Cookie: ASP.NET_SessionId=khpwgbrf1xwwnovibzwsntkg
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/108.0.5359.71 Safari/537.36
```

Response

Response Time (ms) : 203.1202
Total Bytes Received : 27133
Body Length : 26868
Is Compressed : Yes

```
HTTP/1.1 200 OK
Server: Microsoft-IIS/10.0
X-Powered-By: ASP.NET
Vary: Accept-Encoding
X-AspNet-Version: 4.0.30319
Content-Length: 6706
Content-Type: text/html; charset=utf-8
Content-Encoding:
Date: Fri, 08 Dec 2023 06:28:13 GMT
Cache-C
…
.rightbg {
display: none;
}

.col-md-6 {
width: 100%;
}
}
</style>
</head>
<body>
<form name="form1" method="post" action="./Login.aspx" onsubmit="javascript:return WebForm_OnSubmit();"
id="form1">
<div>
<input type="hidden" name="__EVENTTARGET" id="__EVENTTARGET" value="" />
<input type="hidden
…
```

Remedy

- Send additional information in each HTTP request that can be used to determine whether the request came from an authorized source. This "validation token" should be hard to guess for attacker who does not already have access to the user's account. If a request is missing a validation token or the token does not match the expected value, the server should reject the request.
- If you are posting form in ajax request, custom HTTP headers can be used to prevent CSRF because the browser prevents sites from sending custom HTTP headers to another site but allows sites to send custom HTTP headers to themselves using XMLHttpRequest.
    - For native XMLHttpRequest (XHR) object in JavaScript;

```
xhr = new XMLHttpRequest();
xhr.setRequestHeader('custom-header', 'valueNULL');
```

For JQuery, if you want to add a custom header (or set of headers) to
a. **individual request**

```
$.ajax({
    url: 'foo/bar',
    headers: { 'x-my-custom-header': 'some value' }
});
```

b. **every request**

```
$.ajaxSetup({
    headers: { 'x-my-custom-header': 'some value' }
});
OR
$.ajaxSetup({
    beforeSend: function(xhr) {
        xhr.setRequestHeader('x-my-custom-header', 'some value');
    }
});
```

External References

- OWASP Cross-Site Request Forgery (CSRF)
- Robust Defenses for Cross-Site Request Forgery
- Identifying Robust Defenses for Login CSRF

Remedy References

- OWASP Cross-Site Request Forgery (CSRF) Prevention Cheat Sheet

CLASSIFICATION

PCI DSS v3.2	6.5.9
OWASP 2013	A8
OWASP 2017	A5
CWE	352
CAPEC	62
WASC	9
HIPAA	164.306(a)
ASVS 4.0	4.2.2
NIST SP 800-53	SC-23
DISA STIG	V-21500
ISO27001	A.14.2.5
ISO27001 2022	A.8.26
ISO27001 2022	A.8.27
OWASP Top Ten 2021	A01

5. Insecure Transportation Security Protocol Supported (TLS 1.0)

LOW | 1 | CONFIRMED | 1

Invicti Standard detected that insecure transportation security protocol (TLS 1.0) is supported by your web server.

TLS 1.0 has several flaws. An attacker can cause connection failures and they can trigger the use of TLS 1.0 to exploit vulnerabilities like BEAST (Browser Exploit Against SSL/TLS).

Websites using TLS 1.0 are considered non-compliant by PCI since 30 June 2018.

Impact

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

Vulnerabilities

5.1. https://mpcdf.tserver.co.in/mis/Login.aspx

CONFIRMED

Request | Response

Request

```
[SSL Connection]
```

Response

Response Time (ms) : 1
Total Bytes Received : 16
Body Length : 0
Is Compressed : No

```
[SSL Connection]
```

Actions to Take

We recommended to disable TLS 1.0 and replace it with TLS 1.2 or higher. See Remedy section for more details.

Remedy

Configure your web server to disallow using weak ciphers. You need to restart the web server to enable changes.

- For Apache, adjust the SSLProtocol directive provided by the mod_ssl module. This directive can be set either at the server level or in a virtual host configuration.

```
SSLProtocol +TLSv1.2
```

- For Nginx, locate any use of the directive ssl_protocols in the `nginx.conf`file and remove `TLSv1`.

```
ssl_protocols TLSv1.2;
```

- For Microsoft IIS, you should make some changes on the system registry. **Incorrectly editing the registry may severely damage your system. Before making changes to the registry, you should back up any valued data on your computer.**
    1. Click on Start and then Run, type `regedt32`or `regedit`, and then click OK.
    2. In Registry Editor, locate the following registry key or create if it does not exist:

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\TLS 1.0\
```

    3. Locate a key named `Server`or create if it doesn't exist.
    4. Under the `Server`key, locate a DWORD value named `Enabled`or create if it doesn't exist and set its value to "0".
- For lighttpd, put the following lines in your configuration file:

```
ssl.use-sslv2 = "disable"
ssl.use-sslv3 = "disable"
ssl.openssl.ssl-conf-cmd = ("Protocol" => "-TLSv1.1, -TLSv1, -SSLv3") # v1.4.48 or up
ssl.ec-curve = "secp384r1"
```

External References

- How to Disable TLS v1.0 on Windows Server 2019 and Windows Server 2016
- How to Disable TLS v1.0 on Windows Server 2012 and Windows Server 2008
- OWASP - Insecure Configuration Management
- OWASP Top 10 - 2017 A3 - Sensitive Data Exposure
- How to disable PCT 1.0, SSL 2.0, SSL 3.0, or TLS 1.0 in Internet Information Services
- IIS Crypto is a free tool that gives administrators the ability to enable or disable protocols, ciphers, hashes and key exchange algorithms on Windows Server 2003, 2008 and 2012
- Date Change for Migrating from SSL and Early TLS
- Browser Exploit Against SSL/TLS Attack (BEAST)
- Are You Ready for 30 June 2018? Saying Goodbye to SSL/early TLS

CLASSIFICATION

PCI DSS v3.2	6.5.4
OWASP 2013	A6
OWASP 2017	A3
CWE	326
CAPEC	217
WASC	4
HIPAA	164.306
ASVS 4.0	9.1.2
NIST SP 800-53	SC-8
DISA STIG	V-6136
ISO27001	A.14.1.3
ISO27001 2022	A.8.24
OWASP Top Ten 2021	A02

6. Missing X-Frame-Options Header

LOW | 1

Invicti Standard detected a missing `X-Frame-Options`header which means that this website could be at risk of a clickjacking attack.

The `X-Frame-Options`HTTP header field indicates a policy that specifies whether the browser should render the transmitted resource within a `frame`or an `iframe`. Servers can declare this policy in the header of their HTTP responses to prevent clickjacking attacks, which ensures that their content is not embedded into other pages or frames.

Impact

Clickjacking is when an attacker uses multiple transparent or opaque layers to trick a user into clicking on a button or link on a framed page when they were intending to click on the top level page. Thus, the attacker is "hijacking" clicks meant for their page and routing them to other another page, most likely owned by another application, domain, or both.

Using a similar technique, keystrokes can also be hijacked. With a carefully crafted combination of stylesheets, iframes, and text boxes, a user can be led to believe they are typing in the password to their email or bank account, but are instead typing into an invisible frame controlled by the attacker.

Vulnerabilities

6.1. https://mpcdf.tserver.co.in/mis/Login.aspx

Certainty

Request | Response

Request

```
GET /mis/Login.aspx HTTP/1.1
Host: mpcdf.tserver.co.in
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Accept-Encoding: gzip, deflate
Accept-Language: en-us,en;q=0.5
Cache-Control: no-cache
Cookie: ASP.NET_SessionId=khpwgbrf1xwwnovibzwsntkg
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/108.0.5359.71 Safari/537.36
```

Response

Response Time (ms) : 203.1202
Total Bytes Received : 27133
Body Length : 26868
Is Compressed : Yes

```
HTTP/1.1 200 OK
Server: Microsoft-IIS/10.0
X-Powered-By: ASP.NET
Vary: Accept-Encoding
X-AspNet-Version: 4.0.30319
Content-Length: 6706
Content-Type: text/html; charset=utf-8
Content-Encoding:
Date: Fri, 08 Dec 2023 06:28:13 GMT
Cache-Control: private



<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head><meta charset="utf-8" /><meta http-equiv="X-UA-Compatible" content="IE=edge" /><title>
MPCDF
</title><link rel="shortcut icon" href="image/favicon.ico" type="image/ico" /><meta
content="width=device-width, initial-scale=1, maximum-scale=1, user-scalable=no" name="viewport" />
<link href="css/bootstrap.css" type="text/css" rel="stylesheet" /><link href="css/AdminLTE.css"
rel="stylesheet" type="text/css" /><link href="css/blue.css" rel="stylesheet" type="text/css" />
<link href="css/font-awesome/css/font-awesome.css" type="text/css rel="stylesheet" />
<!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and media queries -->
<!-- WARNING: Respond.js doesn't work if you view the page via file:// -->
<!--[if lt IE 9]>
<script src="https://oss.maxcdn.com/html5shiv/3.7.3/html5shiv.min.js"></script>
<script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
<![endif]-->
<style type="text/css">
body {
background: #0074e2 url(image/login_left_bg.jpg);
}

.content-wrapper {
margin-left: 0;
color: #fff;
}

label {
color: #fff;
}

.row {
```

```
display: flex;
flex-wrap: wrap;
margin-right: -15px;
margin-left: -15px;
}

.align-self-end {
align-self: flex-end !important;
}

.form-group label {
line-height: 1.4rem;
vertical-align: top;
margin-bottom: .5rem;
}


.input-group-text {
font-weight: 400;

…
```

Remedy

- Sending the proper X-Frame-Options in HTTP response headers that instruct the browser to not allow framing from other domains.
    - `X-Frame-Options: DENY`It completely denies to be loaded in frame/iframe.
    - `X-Frame-Options: SAMEORIGIN`It allows only if the site which wants to load has a same origin.
    - `X-Frame-Options: ALLOW-FROM` *`URL`*It grants a specific URL to load itself in a iframe. However please pay attention to that, not all browsers support this.
- Employing defensive code in the UI to ensure that the current frame is the most top level window.

External References

- Clickjacking
- Can I Use X-Frame-Options
- X-Frame-Options HTTP Header

Remedy References

- Clickjacking Defense Cheat Sheet

CLASSIFICATION

OWASP 2013	A5
OWASP 2017	A6
CWE	693
CAPEC	103
ASVS 4.0	14.4.7
NIST SP 800-53	CM-6
DISA STIG	V-16786
OWASP API Top Ten 2019	API7
ISO27001	A.14.2.5
ISO27001 2022	A.8.27
OWASP Top Ten 2021	A05

7. Version Disclosure (ASP.NET)

LOW | 1

Invicti Standard identified a version disclosure (ASP.NET) in the target web server's HTTP response.

This information can help an attacker gain a greater understanding of the systems in use and potentially develop further attacks targeted at the specific version of ASP.NET.

Impact

An attacker might use the disclosed information to harvest specific security vulnerabilities for the version identified.

Vulnerabilities

7.1. https://mpcdf.tserver.co.in/mis/Login.aspx

Extracted Version

- 4.0.30319

Certainty

Request | Response

Request

```
GET /mis/Login.aspx HTTP/1.1
Host: mpcdf.tserver.co.in
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Accept-Encoding: gzip, deflate
Accept-Language: en-us,en;q=0.5
Cache-Control: no-cache
Cookie: ASP.NET_SessionId=khpwgbrf1xwwnovibzwsntkg
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/108.0.5359.71 Safari/537.36
```

Response

Response Time (ms) : 203.1202
Total Bytes Received : 27133
Body Length : 26868
Is Compressed : Yes

```
HTTP/1.1 200 OK
Server: Microsoft-IIS/10.0
X-Powered-By: ASP.NET
Vary: Accept-Encoding
X-AspNet-Version: 4.0.30319
Content-Length: 6706
Content-Type: text/html; charset=utf-8
Content-Encoding:
Date: Fri, 08 Dec 2023 06:28:13 GMT
Cache-CHTTP/1.1 200 OK
Server: Microsoft-IIS/10.0
X-Powered-By: ASP.NET
Vary: Accept-Encoding
X-AspNet-Version: 4.0.30319

Content-Length: 6706
Content-Type: text/html; charset=utf-8
Content-Encoding:
Date: Fri, 08 Dec 2023 06:28:13 GMT
Cache-Control: private



<!DOCTYPE html>

<html xmlns="http://www.w3.org/
…
```

Remedy

Apply the following changes to your `web.config`file to prevent information leakage by using custom error pages and removing `X-AspNet-Version`from HTTP responses.

```
<System.Web>
     <httpRuntime enableVersionHeader="false" />
     <customErrors mode="On" defaultRedirect="~/error/GeneralError.aspx">
          <error statusCode="403" redirect="~/error/Forbidden.aspx" />
          <error statusCode="404" redirect="~/error/PageNotFound.aspx" />
          <error statusCode="500" redirect="~/error/InternalError.aspx" />
     </customErrors>
</System.Web>
```

Remedy References

- Error Handling in ASP.NET Pages and Applications
- Remove Unwanted HTTP Response Headers

CLASSIFICATION

OWASP 2013	A5
OWASP 2017	A6
CWE	205
CAPEC	170
WASC	13
HIPAA	164.306(a) , 164.308(a)
ASVS 4.0	14.3.3
NIST SP 800-53	AC-22
DISA STIG	V-16814
OWASP API Top Ten 2019	API7
ISO27001	A.18.1.3
ISO27001 2022	A.8.27
OWASP Top Ten 2021	A05

8. Version Disclosure (IIS)

LOW | 1

Invicti Standard identified a version disclosure (IIS) in the target web server's HTTP response.

This information can help an attacker gain a greater understanding of the systems in use and potentially develop further attacks targeted at the specific version of IIS.

Impact

An attacker might use the disclosed information to harvest specific security vulnerabilities for the version identified.

Vulnerabilities

8.1. https://mpcdf.tserver.co.in/mis/Login.aspx

Extracted Version

- 10.0

Certainty

Request | Response

Request

```
GET /mis/Login.aspx HTTP/1.1
Host: mpcdf.tserver.co.in
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Accept-Encoding: gzip, deflate
Accept-Language: en-us,en;q=0.5
Cache-Control: no-cache
Cookie: ASP.NET_SessionId=khpwgbrf1xwwnovibzwsntkg
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/108.0.5359.71 Safari/537.36
```

Response

Response Time (ms) : 203.1202
Total Bytes Received : 27133
Body Length : 26868
Is Compressed : Yes

```
HTTP/1.1 200 OK
Server: Microsoft-IIS/10.0
X-Powered-By: ASP.NET
Vary: Accept-Encoding
X-AspNet-Version: 4.0.30319
Content-Length: 6706
Content-Type: text/html; charset=utf-8
Content-Encoding:
Date: Fri, 08 Dec 2023 06:28:13 GMT
Cache-CHTTP/1.1 200 OK
Server: Microsoft-IIS/10.0

X-Powered-By: ASP.NET
Vary: Accept-Encoding
X-AspNet-Version: 4.0.30319
Content-Length: 6706
Content-Type: text/html; charset=utf-8
Content-Encoding:
Date: Fri, 08 Dec 2023 06:28:13 GMT
Cache
…
```

Remedy

Configure your web server to prevent information leakage from the `SERVER`header of its HTTP response.

CLASSIFICATION

OWASP 2013	A5
OWASP 2017	A6
CWE	205
CAPEC	170
WASC	13
HIPAA	164.306(a), 164.308(a)
ASVS 4.0	14.3.3
NIST SP 800-53	AC-22
DISA STIG	V-16814
OWASP API Top Ten 2019	API7
OWASP Proactive Controls	N/A
ISO27001	A.18.1.3
ISO27001 2022	A.8.27
OWASP Top Ten 2021	A05

9. ViewState is not Encrypted

LOW | 1

Invicti Standard detected that ViewState Encryption is disabled.

Impact

An attacker can study the application's state management logic for possible vulnerabilities; if your application stores application-critical information in the ViewState, it will also be revealed.

Vulnerabilities

9.1. https://mpcdf.tserver.co.in/mis/Login.aspx

ViewState Version

- .NET Framework 2.x

Certainty

Request | Response

Request

```
GET /mis/Login.aspx HTTP/1.1
Host: mpcdf.tserver.co.in
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Accept-Encoding: gzip, deflate
Accept-Language: en-us,en;q=0.5
Cache-Control: no-cache
Cookie: ASP.NET_SessionId=khpwgbrf1xwwnovibzwsntkg
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/108.0.5359.71 Safari/537.36
```

Response

Response Time (ms) : 203.1202
Total Bytes Received : 27133
Body Length : 26868
Is Compressed : Yes

```
HTTP/1.1 200 OK
Server: Microsoft-IIS/10.0
X-Powered-By: ASP.NET
Vary: Accept-Encoding
X-AspNet-Version: 4.0.30319
Content-Length: 6706
Content-Type: text/html; charset=utf-8
Content-Encoding:
Date: Fri, 08 Dec 2023 06:28:13 GMT
Cache-C
…
return WebForm_OnSubmit();" id="form1">
<div>
<input type="hidden" name="__EVENTTARGET" id="__EVENTTARGET" value="" />
<input type="hidden" name="__EVENTARGUMENT" id="__EVENTARGUMENT" value="" />
<input type="hidden" name="__VIEWSTATE" id="__VIEWSTATE"
value="/wEPDwUKMTU5NjMxOTIyNQ8WAh4KUmFuZG9tVGV4dAUsYjNtc3ZHMWdETE1sOHNOaUtHWkpqYmNIUmFTVit1NkFya0ZKOUNY
TGFIQT1kZEawVkdCEEAvdF8jDU/0xI0b/+gH+Is1+fXl3lwMdFpg" />
</div>

<script type="text/javascript">
//<![CDATA[
var theForm = document.forms['form1'];
if (!theForm) {
theForm = document.form1;
}
function __doPostBack(eventTarget, eventArgument) {
…
```

Remedy

ASP.NET provides encryption for ViewState parameters.

For page based protection, place the following directive at the top of affected page.

```
<%@Page ViewStateEncryptionMode="Always" %>
```

You can also set this option for the whole application by using `web.config`files. Apply the following configuration for your application's `web.config`file.

```
<System.Web>
    <pages viewStateEncryptionMode="Always">
</System.Web>
```

Remedy References

- ASP.NET View State Security

CLASSIFICATION

OWASP 2017	A6
CWE	16
WASC	15
HIPAA	164.306(a) , 164.308(a)
ASVS 4.0	6.2.7
NIST SP 800-53	CM-6
DISA STIG	V-16787
ISO27001	A.14.2.5
ISO27001 2022	A.8.9
OWASP Top Ten 2021	A05

10. Content Security Policy (CSP) Not Implemented

BEST PRACTICE | 1 | CONFIRMED | 1

CSP is an added layer of security that helps to mitigate mainly Cross-site Scripting attacks.

CSP can be enabled instructing the browser with a Content-Security-Policy directive in a response header;

```
Content-Security-Policy: script-src 'self';
```

or in a meta tag;

```
<meta http-equiv="Content-Security-Policy" content="script-src 'self';">
```

In the above example, you can restrict script loading only to the same domain. It will also restrict inline script executions both in the element attributes and the event handlers. There are various directives which you can use by declaring CSP:

- **script-src:**Restricts the script loading resources to the ones you declared. By default, it disables inline script executions unless you permit to the evaluation functions and inline scripts by the unsafe-eval and unsafe-inline keywords.
- **base-uri:**The base element is used to resolve a relative URL to an absolute one. By using this CSP directive, you can define all possible URLs which could be assigned to the base-href attribute of the document.
- **frame-ancestors**: It is very similar to X-Frame-Options HTTP header. It defines the URLs by which the page can be loaded in an iframe.
- **frame-src / child-src**: frame-src is the deprecated version of child-src. Both define the sources that can be loaded by iframe on the page. (Please note that frame-src was brought back in CSP 3)
- **object-src**: Defines the resources that can be loaded by embedding such as Flash files, Java Applets.
- **img-src**: As its name implies, it defines the resources where the images can be loaded from.
- **connect-src**: Defines the whitelisted targets for XMLHttpRequest and WebSocket objects.
- **default-src**: It is a fallback for the directives that mostly end with -src suffix. When the directives below are not defined, the value set to default-src will be used instead:
    - child-src
    - connect-src
    - font-src
    - img-src
    - manifest-src
    - media-src
    - object-src
    - script-src
    - style-src

When setting the CSP directives, you can also use some CSP keywords:

- **none**: Denies loading resources from anywhere.
- **self** : Points to the document's URL (domain + port).
- **unsafe-inline**: Permits running inline scripts.
- **unsafe-eval**: Permits execution of evaluation functions such as eval().

In addition to CSP keywords, you can also use wildcard or only a scheme when defining whitelist URLs for the points. Wildcard can be used for subdomain and port portions of the URLs:

```
Content-Security-Policy: script-src https://*.example.com;
Content-Security-Policy: script-src https://example.com:*;
Content-Security-Policy: script-src https:;
```

It is also possible to set a CSP in Report-Only mode instead of forcing it immediately in the migration period. Thus you can see the violations of the CSP policy in the current state of your web site while migrating to CSP:

```
Content-Security-Policy-Report-Only: script-src 'self'; report-uri: https://example.com;
```

Impact

There is no direct impact of not implementing CSP on your website. However, if your website is vulnerable to a Cross-site Scripting attack CSP can prevent successful exploitation of that vulnerability. By not implementing CSP you'll be missing out on this extra layer of security.

Vulnerabilities

10.1. https://mpcdf.tserver.co.in/mis/Login.aspx

CONFIRMED

Request | Response

Request

```
GET /mis/Login.aspx HTTP/1.1
Host: mpcdf.tserver.co.in
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Accept-Encoding: gzip, deflate
Accept-Language: en-us,en;q=0.5
Cache-Control: no-cache
Cookie: ASP.NET_SessionId=khpwgbrf1xwwnovibzwsntkg
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/108.0.5359.71 Safari/537.36
```

Response

Response Time (ms) : 203.1202
Total Bytes Received : 27133
Body Length : 26868
Is Compressed : Yes

```
HTTP/1.1 200 OK
Server: Microsoft-IIS/10.0
X-Powered-By: ASP.NET
Vary: Accept-Encoding
X-AspNet-Version: 4.0.30319
Content-Length: 6706
Content-Type: text/html; charset=utf-8
Content-Encoding:
Date: Fri, 08 Dec 2023 06:28:13 GMT
Cache-Control: private



<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head><meta charset="utf-8" /><meta http-equiv="X-UA-Compatible" content="IE=edge" /><title>
MPCDF
</title><link rel="shortcut icon" href="image/favicon.ico" type="image/ico" /><meta
content="width=device-width, initial-scale=1, maximum-scale=1, user-scalable=no" name="viewport" />
<link href="css/bootstrap.css" type="text/css" rel="stylesheet" /><link href="css/AdminLTE.css"
rel="stylesheet" type="text/css" /><link href="css/blue.css" rel="stylesheet" type="text/css" />
<link href="css/font-awesome/css/font-awesome.css" type="text/css rel="stylesheet" />
<!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and media queries -->
<!-- WARNING: Respond.js doesn't work if you view the page via file:// -->
<!--[if lt IE 9]>
<script src="https://oss.maxcdn.com/html5shiv/3.7.3/html5shiv.min.js"></script>
<script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
<![endif]-->
<style type="text/css">
body {
background: #0074e2 url(image/login_left_bg.jpg);
}

.content-wrapper {
margin-left: 0;
color: #fff;
}

label {
color: #fff;
}

.row {
```

```
display: flex;
flex-wrap: wrap;
margin-right: -15px;
margin-left: -15px;
}

.align-self-end {
align-self: flex-end !important;
}

.form-group label {
line-height: 1.4rem;
vertical-align: top;
margin-bottom: .5rem;
}


.input-group-text {
font-weight: 400;

…
```

Actions to Take

- Enable CSP on your website by sending the `Content-Security-Policy`in HTTP response headers that instruct the browser to apply the policies you specified.
- Apply the whitelist and policies as strict as possible.
- Rescan your application to see if Invicti Standard identifies any weaknesses in your policies.

Remedy

Enable CSP on your website by sending the `Content-Security-Policy`in HTTP response headers that instruct the browser to apply the policies you specified.

External References

- An Introduction to Content Security Policy
- Content Security Policy (CSP) HTTP Header
- Content Security Policy (CSP)

CLASSIFICATION

CWE	16
WASC	15
ASVS 4.0	14.4.3
NIST SP 800-53	CM-6
DISA STIG	V-16786
ISO27001	A.14.2.5
ISO27001 2022	A.8.27

11. Insecure Transportation Security Protocol Supported (TLS 1.1)

BEST PRACTICE | 1 | CONFIRMED | 1

Invicti Standard detected that a deprecated, insecure transportation security protocol (TLS 1.1) is supported by your web server.

TLS 1.1 will be considered as deprecated by major web browsers (i.e. Chrome, Firefox, Safari, Edge, Internet Explorer) starting in 2020.

Impact

Your website will be inaccessible due to web browser deprecation.

Vulnerabilities

11.1. https://mpcdf.tserver.co.in/mis/Login.aspx

CONFIRMED

Request | Response

Request

```
[SSL Connection]
```

Response

Response Time (ms) : 1
Total Bytes Received : 16
Body Length : 0
Is Compressed : No

```
[SSL Connection]
```

Actions to Take

We recommended to disable TLS 1.1 and replace it with TLS 1.2 or higher. See Remedy section for more details.

Remedy

Configure your web server to disallow using weak ciphers. You need to restart the web server to enable changes.

- For Apache, adjust the SSLProtocol directive provided by the mod_ssl module. This directive can be set either at the server level or in a virtual host configuration.

```
SSLProtocol +TLSv1.2
```

- For Nginx, locate any use of the directive ssl_protocols in the `nginx.conf`file and remove `TLSv1.1`.

```
ssl_protocols TLSv1.2;
```

- For Microsoft IIS, you should make some changes on the system registry. **Incorrectly editing the registry may severely damage your system. Before making changes to the registry, you should back up any valued data on your computer.**
    1. Click on Start and then Run, type `regedt32`or `regedit`, and then click OK.
    2. In Registry Editor, locate the following registry key or create if it does not exist:

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\TLS 1.1\
```

    3. Locate a key named `Server`or create if it doesn't exist.
    4. Under the `Server`key, locate a DWORD value named `Enabled`or create if it doesn't exist and set its value to "0".
- For lighttpd, put the following lines in your configuration file:

```
ssl.use-sslv2 = "disable"
ssl.use-sslv3 = "disable"
ssl.openssl.ssl-conf-cmd = ("Protocol" => "-TLSv1.1, -TLSv1, -SSLv3") # v1.4.48 or up
ssl.ec-curve = "secp384r1"
```

External References

- Deprecating TLSv1.0 and TLSv1.1 draft-ietf-tls-oldversions-deprecate-00
- Google Security Blog: Modernizing Transport Security
- OWASP - Insecure Configuration Management
- OWASP Top 10 - 2017 A3 - Sensitive Data Exposure
- IIS Crypto is a free tool that gives administrators the ability to enable or disable protocols, ciphers, hashes and key exchange algorithms on Windows Server 2003, 2008 and 2012
- Date Change for Migrating from SSL and Early TLS

CLASSIFICATION

PCI DSS v3.2	6.5.4
OWASP 2013	A6
OWASP 2017	A3
CWE	326
CAPEC	217
WASC	4
HIPAA	164.306
ASVS 4.0	9.1.2
NIST SP 800-53	SC-8
DISA STIG	V-6136
ISO27001	A.14.1.3
ISO27001 2022	A.8.24
OWASP Top Ten 2021	A02

12. Missing X-XSS-Protection Header

BEST PRACTICE | 1

Invicti Standard detected a missing `X-XSS-Protection`header which means that this website could be at risk of a Cross-site Scripting (XSS) attacks.

Impact

This issue is reported as additional information only. There is no direct impact arising from this issue.

Vulnerabilities

12.1. https://mpcdf.tserver.co.in/mis/Login.aspx

Certainty

Request | Response

Request

```
GET /mis/Login.aspx HTTP/1.1
Host: mpcdf.tserver.co.in
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Accept-Encoding: gzip, deflate
Accept-Language: en-us,en;q=0.5
Cache-Control: no-cache
Cookie: ASP.NET_SessionId=khpwgbrf1xwwnovibzwsntkg
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/108.0.5359.71 Safari/537.36
```

Response

Response Time (ms) : 203.1202
Total Bytes Received : 27133
Body Length : 26868
Is Compressed : Yes

```
HTTP/1.1 200 OK
Server: Microsoft-IIS/10.0
X-Powered-By: ASP.NET
Vary: Accept-Encoding
X-AspNet-Version: 4.0.30319
Content-Length: 6706
Content-Type: text/html; charset=utf-8
Content-Encoding:
Date: Fri, 08 Dec 2023 06:28:13 GMT
Cache-Control: private



<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head><meta charset="utf-8" /><meta http-equiv="X-UA-Compatible" content="IE=edge" /><title>
MPCDF
</title><link rel="shortcut icon" href="image/favicon.ico" type="image/ico" /><meta
content="width=device-width, initial-scale=1, maximum-scale=1, user-scalable=no" name="viewport" />
<link href="css/bootstrap.css" type="text/css" rel="stylesheet" /><link href="css/AdminLTE.css"
rel="stylesheet" type="text/css" /><link href="css/blue.css" rel="stylesheet" type="text/css" />
<link href="css/font-awesome/css/font-awesome.css" type="text/css rel="stylesheet" />
<!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and media queries -->
<!-- WARNING: Respond.js doesn't work if you view the page via file:// -->
<!--[if lt IE 9]>
<script src="https://oss.maxcdn.com/html5shiv/3.7.3/html5shiv.min.js"></script>
<script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
<![endif]-->
<style type="text/css">
body {
background: #0074e2 url(image/login_left_bg.jpg);
}

.content-wrapper {
margin-left: 0;
color: #fff;
}

label {
color: #fff;
}

.row {
```

```
display: flex;
flex-wrap: wrap;
margin-right: -15px;
margin-left: -15px;
}

.align-self-end {
align-self: flex-end !important;
}

.form-group label {
line-height: 1.4rem;
vertical-align: top;
margin-bottom: .5rem;
}


.input-group-text {
font-weight: 400;

…
```

Remedy

Add the X-XSS-Protection header with a value of "1; mode= block".

- ```
  X-XSS-Protection: 1; mode=block
  ```

Please also be advised that in some specific cases enabling XSS filter can be abused by attackers. However, in most cases, it provides basic protection for users against XSS attacks.

External References

- Internet Explorer 8 Security Features - MSDN
- X-XSS-Protection HTTP Header
- X-XSS-Protection
- XSS Auditors - Abuses, Updates and Protection
  ```

CLASSIFICATION

CWE	16
WASC	15
HIPAA	164.308(a)
NIST SP 800-53	CM-6
DISA STIG	V-16787
ISO27001	A.14.2.5
ISO27001 2022	A.8.27

13. Referrer-Policy Not Implemented

BEST PRACTICE | 1

Invicti Standard detected that no Referrer-Policy header implemented.

Referrer-Policy is a security header designed to prevent cross-domain Referer leakage.

Impact

Referer header is a request header that indicates the site which the traffic originated from. If there is no adequate prevention in place, the URL itself, and even sensitive information contained in the URL will be leaked to the cross-site.

The lack of Referrer-Policy header might affect privacy of the users and site's itself

Vulnerabilities

13.1. https://mpcdf.tserver.co.in/mis/Login.aspx

Certainty

Request | Response

Request

```
GET /mis/Login.aspx HTTP/1.1
Host: mpcdf.tserver.co.in
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Accept-Encoding: gzip, deflate
Accept-Language: en-us,en;q=0.5
Cache-Control: no-cache
Cookie: ASP.NET_SessionId=khpwgbrf1xwwnovibzwsntkg
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/108.0.5359.71 Safari/537.36
```

Response

Response Time (ms) : 203.1202
Total Bytes Received : 27133
Body Length : 26868
Is Compressed : Yes

```
HTTP/1.1 200 OK
Server: Microsoft-IIS/10.0
X-Powered-By: ASP.NET
Vary: Accept-Encoding
X-AspNet-Version: 4.0.30319
Content-Length: 6706
Content-Type: text/html; charset=utf-8
Content-Encoding:
Date: Fri, 08 Dec 2023 06:28:13 GMT
Cache-Control: private



<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head><meta charset="utf-8" /><meta http-equiv="X-UA-Compatible" content="IE=edge" /><title>
MPCDF
</title><link rel="shortcut icon" href="image/favicon.ico" type="image/ico" /><meta
content="width=device-width, initial-scale=1, maximum-scale=1, user-scalable=no" name="viewport" />
<link href="css/bootstrap.css" type="text/css" rel="stylesheet" /><link href="css/AdminLTE.css"
rel="stylesheet" type="text/css" /><link href="css/blue.css" rel="stylesheet" type="text/css" />
<link href="css/font-awesome/css/font-awesome.css" type="text/css rel="stylesheet" />
<!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and media queries -->
<!-- WARNING: Respond.js doesn't work if you view the page via file:// -->
<!--[if lt IE 9]>
<script src="https://oss.maxcdn.com/html5shiv/3.7.3/html5shiv.min.js"></script>
<script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
<![endif]-->
<style type="text/css">
body {
background: #0074e2 url(image/login_left_bg.jpg);
}

.content-wrapper {
margin-left: 0;
color: #fff;
}

label {
color: #fff;
}

.row {
```

```
display: flex;
flex-wrap: wrap;
margin-right: -15px;
margin-left: -15px;
}

.align-self-end {
align-self: flex-end !important;
}

.form-group label {
line-height: 1.4rem;
vertical-align: top;
margin-bottom: .5rem;
}


.input-group-text {
font-weight: 400;

…
```

Actions to Take

In a response header:

```
Referrer-Policy: no-referrer | same-origin | origin | strict-origin | no-origin-when-downgrading
```

In a META tag

```
<meta name="Referrer-Policy" value="no-referrer | same-origin"/>
```

In an element attribute

```
<a href="http://crosssite.example.com" rel="noreferrer"></a>
```

or

```
<a href="http://crosssite.example.com" referrerpolicy="no-referrer | same-origin | origin | strict-
origin | no-origin-when-downgrading"></a>
```

Remedy

Please implement a Referrer-Policy by using the Referrer-Policy response header or by declaring it in the meta tags. It’s also possible to control referrer information over an HTML-element by using the rel attribute.

External References

- Referrer Policy
- Referrer Policy - MDN
- Referrer Policy HTTP Header
- A New Security Header: Referrer Policy
- Can I Use Referrer-Policy

CLASSIFICATION

OWASP 2013	A6
OWASP 2017	A3
CWE	200
ASVS 4.0	14.4.6
NIST SP 800-53	AC-22
DISA STIG	V-16814
ISO27001	A.14.2.5
ISO27001 2022	A.8.27

14. [Possible] Login Page Identified

INFORMATION | 1

Invicti Standard identified a login page on the target website.

Impact

This issue is reported as additional information only. There is no direct impact arising from this issue.

Vulnerabilities

14.1. https://mpcdf.tserver.co.in/mis/Login.aspx

form.action

- ./Login.aspx

window.location.pathname

- /mis/Login.aspx

input.id

- txtUserName

Certainty

Request | Response

Request

```
GET /mis/Login.aspx HTTP/1.1
Host: mpcdf.tserver.co.in
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Accept-Encoding: gzip, deflate
Accept-Language: en-us,en;q=0.5
Cache-Control: no-cache
Cookie: ASP.NET_SessionId=khpwgbrf1xwwnovibzwsntkg
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/108.0.5359.71 Safari/537.36
```

Response

Response Time (ms) : 203.1202
Total Bytes Received : 27133
Body Length : 26868
Is Compressed : Yes

```
HTTP/1.1 200 OK
Server: Microsoft-IIS/10.0
X-Powered-By: ASP.NET
Vary: Accept-Encoding
X-AspNet-Version: 4.0.30319
Content-Length: 6706
Content-Type: text/html; charset=utf-8
Content-Encoding:
Date: Fri, 08 Dec 2023 06:28:13 GMT
Cache-C
…
600px) {
.rightbg {
display: none;
}

.col-md-6 {
width: 100%;
}
}
</style>
</head>
<body>
<form name="form1" method="post" action="./Login.aspx" onsubmit="javascript:return WebForm_OnSubmit();"
id="form1"><form name="form1" method="post" action="./Login.aspx" onsubmit="javascript:return
WebForm_OnSubmit();" id="form1"><form name="form1" method="post" action="./Login.aspx"
onsubmit="javascript:return WebForm_OnSubmit();" id="form1">
<div>
<input type="hidden" name="__EVENTTARGET" id="__EVENTTARGET" value="" />
<input type="hidden" name="__EVENTARGUMENT" id="__EVENTARGUMENT" value="" />
<input type="hidden" name="__VIEWSTATE"
…
```

CLASSIFICATION

OWASP Proactive Controls	C6

15. ASP.NET Identified

INFORMATION | 1

Invicti Standard identified that the target website is using ASP.NET as its web application framework.

This issue is reported as extra information only.

Impact

This issue is reported as additional information only. There is no direct impact arising from this issue.

Vulnerabilities

15.1. https://mpcdf.tserver.co.in/mis/Login.aspx

Certainty

Request | Response

Request

```
GET /mis/Login.aspx HTTP/1.1
Host: mpcdf.tserver.co.in
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Accept-Encoding: gzip, deflate
Accept-Language: en-us,en;q=0.5
Cache-Control: no-cache
Cookie: ASP.NET_SessionId=khpwgbrf1xwwnovibzwsntkg
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/108.0.5359.71 Safari/537.36
```

Response

Response Time (ms) : 203.1202
Total Bytes Received : 27133
Body Length : 26868
Is Compressed : Yes

```
HTTP/1.1 200 OK
Server: Microsoft-IIS/10.0
X-Powered-By: ASP.NET
Vary: Accept-Encoding
X-AspNet-Version: 4.0.30319
Content-Length: 6706
Content-Type: text/html; charset=utf-8
Content-Encoding:
Date: Fri, 08 Dec 2023 06:28:13 GMT
Cache-CHTTP/1.1 200 OK
Server: Microsoft-IIS/10.0
X-Powered-By: ASP.NET
Vary: Accept-Encoding
X-AspNet-Version: 4.0.30319
Content-Length: 6706
Content-Type: text/html; charset=utf-8
Content-Encoding:
Date: Fri, 08 Dec 2023 06:28:13 GMT
Cache-Control: private



…
```

CLASSIFICATION

OWASP 2017	A6
CWE	205
WASC	13
ASVS 4.0	14.3.3
NIST SP 800-53	AC-22
DISA STIG	V-16814
OWASP API Top Ten 2019	API7
OWASP Proactive Controls	C7
ISO27001	A.14.2.5
OWASP Top Ten 2021	A05

CVSS 3.0 SCORE

Base	5.3 (Medium)
Temporal	5.1 (Medium)
Environmental	5.1 (Medium)

CVSS Vector String

CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N/E:H/RL:O/RC:C

CVSS 3.1 SCORE	
Base	5.3 (Medium)
Temporal	5.1 (Medium)
Environmental	5.1 (Medium)

CVSS Vector String
CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N/E:H/RL:O/RC:C

16. IIS Identified

INFORMATION | 1

Invicti Standard identified a web server (IIS) in the target web server's HTTP response.

Impact

This issue is reported as additional information only. There is no direct impact arising from this issue.

Vulnerabilities

16.1. https://mpcdf.tserver.co.in/mis/Login.aspx

Certainty

Request | Response

Request

```
GET /mis/Login.aspx HTTP/1.1
Host: mpcdf.tserver.co.in
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Accept-Encoding: gzip, deflate
Accept-Language: en-us,en;q=0.5
Cache-Control: no-cache
Cookie: ASP.NET_SessionId=khpwgbrf1xwwnovibzwsntkg
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/108.0.5359.71 Safari/537.36
```

Response

Response Time (ms) : 203.1202
Total Bytes Received : 27133
Body Length : 26868
Is Compressed : Yes

```
HTTP/1.1 200 OK
Server: Microsoft-IIS/10.0
X-Powered-By: ASP.NET
Vary: Accept-Encoding
X-AspNet-Version: 4.0.30319
Content-Length: 6706
Content-Type: text/html; charset=utf-8
Content-Encoding:
Date: Fri, 08 Dec 2023 06:28:13 GMT
Cache-CHTTP/1.1 200 OK
Server: Microsoft-IIS/10.0

X-Powered-By: ASP.NET
Vary: Accept-Encoding
X-AspNet-Version: 4.0.30319
Content-Length: 6706
Content-Type: text/html; charset=utf-8
Content-Encoding:
Date: Fri, 08 Dec 2023 06:28:13 GMT
Cache
…
```

External References

- IIS Official Website

CLASSIFICATION

OWASP 2017	A6
CWE	205
WASC	13
ASVS 4.0	14.3.3
NIST SP 800-53	AC-22
DISA STIG	V-16814
OWASP API Top Ten 2019	API7
OWASP Proactive Controls	C7
ISO27001	A.14.2.5
OWASP Top Ten 2021	A05

CVSS 3.0 SCORE

Base	5.3 (Medium)
Temporal	5.1 (Medium)
Environmental	5.1 (Medium)

CVSS Vector String

CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N/E:H/RL:O/RC:C

CVSS 3.1 SCORE	
Base	5.3 (Medium)
Temporal	5.1 (Medium)
Environmental	5.1 (Medium)

CVSS Vector String
CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N/E:H/RL:O/RC:C

Show Scan Detail

Enabled Security Checks : **Apache Struts S2-045 RCE,**
Apache Struts S2-046 RCE,
Arbitrary Files (IAST),
BREACH Attack,
Code Evaluation,
Code Evaluation (IAST),
Code Evaluation (Out of Band),
Command Injection,
Command Injection (Blind),
Command Injection (IAST),
Configuration Analyzer (IAST),
Content Security Policy,
Content-Type Sniffing,
Cookie,
Cross Frame Options Security,
Cross-Origin Resource Sharing (CORS),
Cross-Site Request Forgery,
Cross-site Scripting,
Cross-site Scripting (Blind),
Cross-site Scripting (DOM based),
Custom Script Checks (Active),
Custom Script Checks (Passive),
Custom Script Checks (Per Directory),
Custom Script Checks (Singular),
Drupal Remote Code Execution,

Expression Language Injection,
File Upload,
GraphQL Library Detection,
Header Analyzer,
Heartbleed,
HSTS,
HTML Content,
HTTP Header Injection,
HTTP Header Injection (IAST),
HTTP Methods,
HTTP Status,
HTTP.sys (CVE-2015-1635),
IFrame Security,
Insecure JSONP Endpoint,
Insecure Reflected Content,
JavaScript Libraries,
JSON Web Token,
LDAP Injection (IAST),
Local File Inclusion,
Local File Inclusion (IAST),
Log4j Code Evaluation (Out of Band),
Login Page Identifier,
Mail Header Injection (IAST),
Malware Analyzer,
Mixed Content,
MongoDB Injection (Boolean),
MongoDB Injection (IAST),
MongoDB Injection (Operator),
Open Redirection,
Oracle WebLogic Remote Code Execution,
Referrer Policy,
Reflected File Download,
Remote File Inclusion,
Reverse Proxy Detection,
RoR Code Execution,
Security Assertion Markup Language (SAML),
Sensitive Data,
Server-Side Request Forgery (DNS),
Server-Side Request Forgery (IP Combinations),
Server-Side Request Forgery (Pattern Based),
Server-Side Template Injection,
Server-Side Template Injection (IAST),
Signatures,
Software Composition Analysis (SCA),
Spring4Shell Remote Code Execution,
SQL Injection (Blind),
SQL Injection (Boolean),
SQL Injection (Error Based),
SQL Injection (IAST),
SQL Injection (Out of Band),
SSL,
Static Resources (All Paths),
Static Resources (Only Root Path),

		Unicode Transformation (Best-Fit Mapping), **WAF Identifier,** **Web App Fingerprint,** **Web Cache Deception,** **WebDAV,** **Windows Short Filename,** **Wordpress Plugin Detection,** **Wordpress Theme Detection,** **XML External Entity,** **XML External Entity (Out of Band),** **XML External Entity Injection (IAST),** **XPath Injection (IAST)**
URL Rewrite Mode	:	**None**
Detected URL Rewrite Rule(s)	:	**None**
Included URL Patterns	:	**gtm\.js** **WebResource\.axd** **ScriptResource\.axd**
Authentication	:	**Form Authentication**
Authentication Profile	:	**None**
Scheduled	:	**No**
Additional Website(s)	:	**None**

This report created with 23.8.0.41720-release_is-23.8.0-a9dc73c
https://www.invicti.com

SFA Technologies