netsparker

12/23/2020 11:16:33 AM (UTC-05:00)

Detailed Scan Report

https://cdr-stage.cancer.gov/

Scan Time	12/22/2020 4:01:17 PM (UTC-05:00)
Scan Duration	00:15:42:44
Total Requests	: 28,954
Average Speed	: 0.5 r/s

Risk Level:
LOW

14 IDENTIFIED	**2** CONFIRMED	**0** CRITICAL
0 HIGH	0 MEDIUM	5 LOW
	5 BEST PRACTICE	4 INFORMATION

Identified Vulnerabilities

Critical	0
High	0
Medium	0
Low	5
Best Practice	5
Information	4
TOTAL	**14**

Confirmed Vulnerabilities

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

Vulnerability Summary

CONFIRM	VULNERABILITY	METHOD	URL	PARAMETER
	[Possible] Cross-site Request Forgery	GET	https://cdr-stage.cancer.gov/cgi-bin/cdr/TerminologyReports.py	
	Missing X-Frame-Options Header	GET	https://cdr-stage.cancer.gov/cgi-bin/	
	Programming Error Message	GET	https://cdr-stage.cancer.gov/trace.axd	
	Stack Trace Disclosure (ASP.NET)	GET	https://cdr-stage.cancer.gov/trace.axd	
	Version Disclosure (ASP.NET)	GET	https://cdr-stage.cancer.gov/trace.axd	
	Content Security Policy (CSP) Not Implemented	GET	https://cdr-stage.cancer.gov/cgi-bin/	
	Expect-CT Not Enabled	GET	https://cdr-stage.cancer.gov/	
	Missing X-XSS-Protection Header	GET	https://cdr-stage.cancer.gov/cgi-bin/	
	Referrer-Policy Not Implemented	GET	https://cdr-stage.cancer.gov/cgi-bin/	
	Subresource Integrity (SRI) Not Implemented	GET	https://cdr-stage.cancer.gov/cgi-bin/cdr/GuestUsers.py	
	ASP.NET Identified	GET	https://cdr-stage.cancer.gov/	
	Version Disclosure (IIS)	GET	https://cdr-stage.cancer.gov/	
	Authorization Required	GET	https://cdr-stage.cancer.gov/cgi-bin/secure/admin.py	
	Forbidden Resource	GET	https://cdr-stage.cancer.gov/cgi-bin/	

1. [Possible] Cross-site Request Forgery

LOW 1

Netsparker identified a possible Cross-Site Request Forgery.

CSRF is a very common vulnerability. It's an attack which forces a user to execute unwanted actions on a web application in which the user is currently authenticated.

Impact

Depending on the application, an attacker can mount any of the actions that can be done by the user such as adding a user, modifying content, deleting data. All the functionality that's available to the victim can be used by the attacker. Only exception to this rule is a page that requires extra information that only the legitimate user can know (such as user's password).

Vulnerabilities

1.1. https://cdr-stage.cancer.gov/cgi-bin/cdr/TerminologyReports.py

Form Action(s)

- TerminologyReports.py

Certainty

Request

```
GET /cgi-bin/cdr/TerminologyReports.py HTTP/1.1
Host: cdr-stage.cancer.gov
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; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/79.0.39
45.0 Safari/537.36
X-Scanner: NCI-Netsparker
```

Response

Response Time (ms) : 1327.8743 Total Bytes Received : 2241 Body Length : 2012 Is Compressed : No

```
HTTP/1.1 200 OK
Server: Microsoft-IIS/10.0
X-Powered-By: ASP.NET
Connection: close
Content-Length: 930
Content-Type: text/html;charset=utf-8
Content-Encoding:
Date: Tue, 22 Dec 2020 21:02:47 GMT
Vary:
…
.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script><script src="https://ajax.googleapis.com/ajax/li
bs/jqueryui/1.12.1/jquery-ui.min.js"></script>
</head>
<body id="cdr-page" class="admin-menu"><form action="TerminologyReports.py" method="post" id="primar
y-form">
<input name="Session" value="guest" type="hidden"><header><h1>CDR Administration<span id="header-but
tons"><input value="Reports Menu" name="Request" type="submit"><in
…
```

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)

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)
ISO27001	A.14.2.5

2. Missing X-Frame-Options Header

LOW 1

Netsparker 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

2.1. https://cdr-stage.cancer.gov/cgi-bin/

Certainty

Request

```
GET /cgi-bin/ HTTP/1.1
Host: cdr-stage.cancer.gov
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; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/79.0.39
45.0 Safari/537.36
X-Scanner: NCI-Netsparker
```

Response

Response Time (ms) : 0 Total Bytes Received : 1394 Body Length : 1233 Is Compressed : No

```
HTTP/1.1 403 Forbidden
Server: Microsoft-IIS/10.0
X-Powered-By: ASP.NET
Content-Length: 1233
Content-Type: text/html
Date: Tue, 22 Dec 2020 21:01:37 GMT

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-str
ict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1"/>
<title>403 - Forbidden: Access is denied.</title>
<style type="text/css">
<!--
body{margin:0;font-size:.7em;font-family:Verdana, Arial, Helvetica, sans-serif;background:#EEEEEE;}
fieldset{padding:0 15px 10px 15px;}
h1{font-size:2.4em;margin:0;color:#FFF;}
h2{font-size:1.7em;margin:0;color:#CC0000;}
h3{font-size:1.2em;margin:10px 0 0 0;color:#000000;}
#header{width:96%;margin:0 0 0 0;padding:6px 2% 6px 2%;font-family:"trebuchet MS", Verdana, sans-ser
if;color:#FFF;
background-color:#555555;}
#content{margin:0 0 0 2%;position:relative;}
.content-container{background:#FFF;width:96%;margin-top:8px;padding:10px;position:relative;}
-->
</style>
</head>
<body>
<div id="header"><h1>Server Error</h1></div>
<div id="content">
 <div class="content-container"><fieldset>
  <h2>403 - Forbidden: Access is denied.</h2>
  <h3>You do not have permission to view this directory or page using the credentials that you suppl
ied.</h3>
 </fieldset></div>
</div>
</body>
</html>
```

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
ISO27001	A.14.2.5

3. Programming Error Message

LOW 1

Netsparker identified a Programming Error Message.

Impact

The error message may disclose sensitive information and this information can be used by an attacker to mount new attacks or to enlarge the attack surface. Source code, stack trace, etc. data may be disclosed. Most of these issues will be identified and reported separately by Netsparker.

Vulnerabilities

3.1. https://cdr-stage.cancer.gov/trace.axd

Method	Parameter	Value
GET	URI-BASED	trace.axd

Identified Error Message

- Exception of type 'System.Web.HttpException' was thrown.

Certainty

Request

```
GET /trace.axd HTTP/1.1
Host: cdr-stage.cancer.gov
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; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/79.0.39
45.0 Safari/537.36
X-Scanner: NCI-Netsparker
```

Response

Response Time (ms) : 136.8441 Total Bytes Received : 3629 Body Length : 3400 Is Compressed : No

```
HTTP/1.1 403 Forbidden
Server: Microsoft-IIS/10.0
X-Powered-By: ASP.NET
X-AspNet-Version: 4.0.30319
Content-Length: 3400
Content-Type: text/html; charset=utf-8
Date: Tue, 22 Dec 2020 21:07:57 GMT

…
=silver>

            <b>Version Information:</b> Microsoft .NET Framework Version:4.0.30319; ASP.NET Ver
sion:4.7.3701.0

            </font>

    </body>
</html>
<!--
[HttpException]: Exception of type &#39;System.Web.HttpException&#39; was thrown.
   at System.Web.Handlers.TraceHandler.System.Web.IHttpHandler.ProcessRequest(HttpContext context)
   at System.Web.HttpApplication.CallHandlerExecutionStep.System.Web.HttpApplication.IExecutionSte
…
```

Remedy

Do not provide error messages on production environments. Save error messages with a reference number to a backend storage such as a log, text file or database, then show this number and a static user-friendly error message to the user.

CLASSIFICATION

PCI DSS v3.2	6.5.5
OWASP 2013	A5
OWASP 2017	A6
CWE	210
CAPEC	118
WASC	13
HIPAA	164.306(A), 164.308(A)

ISO27001	A.18.1.3

4. Stack Trace Disclosure (ASP.NET)

LOW 1

Netsparker identified a stack trace disclosure (ASP.NET) in the target web server's HTTP response.

Impact

An attacker can obtain information such as:

- ASP.NET version.
- Physical file path of temporary ASP.NET files.
- Information about the generated exception and possibly source code, SQL queries, etc.

This information might help an attacker gain more information and potentially focus on the development of further attacks for the target system.

Vulnerabilities

4.1. https://cdr-stage.cancer.gov/trace.axd

Method	Parameter	Value
GET	URI-BASED	trace.axd

Certainty

Request

```
GET /trace.axd HTTP/1.1
Host: cdr-stage.cancer.gov
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; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/79.0.39
45.0 Safari/537.36
X-Scanner: NCI-Netsparker
```

Response

Response Time (ms) : 136.8441 Total Bytes Received : 3629 Body Length : 3400 Is Compressed : No

```
HTTP/1.1 403 Forbidden
Server: Microsoft-IIS/10.0
X-Powered-By: ASP.NET
X-AspNet-Version: 4.0.30319
Content-Length: 3400
Content-Type: text/html; charset=utf-8
Date: Tue, 22 Dec 2020 21:07:57 GMT

…
 .NET Framework Version:4.0.30319; ASP.NET Version:4.7.3701.0

            </font>

    </body>
</html>
<!--
[HttpException]: Exception of type &#39;System.Web.HttpException&#39; was thrown.
   at System.Web.Handlers.TraceHandler.System.Web.IHttpHandler.ProcessRequest(HttpContext context)

   at System.Web.HttpApplication.CallHandlerExecutionStep.System.Web.HttpApplication.IExecutionStep.
Execute()

   at System.Web.HttpApplication.ExecuteStepImpl(IExecutionStep step)

   at System.Web.HttpApplication.ExecuteStep(IExecutionStep step, Boolean& completedSynchronously)

--><!--
This error page might contain sensitive information because ASP.NET is configured to show verbose er
ror messages using &lt;customErrors mode="Off"/&gt;. Consider using &lt;customErrors mode
…
```

Remedy

Apply following changes on your `web.config` file to prevent information leakage by applying custom error pages.

```
<System.Web>
     <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

CLASSIFICATION

PCI DSS v3.2	6.5.5
OWASP 2013	A5
OWASP 2017	A6
CWE	248
CAPEC	214
WASC	14
HIPAA	164.306(A), 164.308(A)
ISO27001	A.9.2.3

5. Version Disclosure (ASP.NET)

LOW 1

Netsparker identified a version disclosure (ASP.NET) in 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

5.1. https://cdr-stage.cancer.gov/trace.axd

Method	Parameter	Value
GET	URI-BASED	trace.axd

Extracted Version

- 4.7.3701.0

Certainty

Request

```
GET /trace.axd HTTP/1.1
Host: cdr-stage.cancer.gov
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; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/79.0.39
45.0 Safari/537.36
X-Scanner: NCI-Netsparker
```

Response

Response Time (ms) : 136.8441 Total Bytes Received : 3629 Body Length : 3400 Is Compressed : No

```
HTTP/1.1 403 Forbidden
Server: Microsoft-IIS/10.0
X-Powered-By: ASP.NET
X-AspNet-Version: 4.0.30319
Content-Length: 3400
Content-Type: text/html; charset=utf-8
Date: Tue, 22 Dec 2020 21:07:57 GMT

…
e></code>

                </td>
              </tr>
            </table>

            <br>

            <hr width=100% size=1 color=silver>

            <b>Version Information:</b> Microsoft .NET Framework Version:4.0.30319; ASP.NET Ver
sion:4.7.3701.0

            </font>

    </body>
</html>
<!--
[HttpException]: Exception of type &#39;System.Web.HttpException&#39; was thrown.
   at System.Web.Handlers.TraceHandler.System.Web.IHttpHandle
…
```

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	45
HIPAA	164.306(A), 164.308(A)
ISO27001	A.18.1.3

6. Content Security Policy (CSP) Not Implemented

BEST PRACTICE | 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:** Base element is used to resolve relative URL to absolute one. By using this CSP directive, you can define all possible URLs which could be assigned to 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 in 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 ends 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 this extra layer of security.

Vulnerabilities

6.1. https://cdr-stage.cancer.gov/cgi-bin/

Certainty

Request

```
GET /cgi-bin/ HTTP/1.1
Host: cdr-stage.cancer.gov
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; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/79.0.39
45.0 Safari/537.36
X-Scanner: NCI-Netsparker
```

Response

Response Time (ms) : 0 Total Bytes Received : 1394 Body Length : 1233 Is Compressed : No

```
HTTP/1.1 403 Forbidden
Server: Microsoft-IIS/10.0
X-Powered-By: ASP.NET
Content-Length: 1233
Content-Type: text/html
Date: Tue, 22 Dec 2020 21:01:37 GMT

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-str
ict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1"/>
<title>403 - Forbidden: Access is denied.</title>
<style type="text/css">
<!--
body{margin:0;font-size:.7em;font-family:Verdana, Arial, Helvetica, sans-serif;background:#EEEEEE;}
fieldset{padding:0 15px 10px 15px;}
h1{font-size:2.4em;margin:0;color:#FFF;}
h2{font-size:1.7em;margin:0;color:#CC0000;}
h3{font-size:1.2em;margin:10px 0 0 0;color:#000000;}
#header{width:96%;margin:0 0 0 0;padding:6px 2% 6px 2%;font-family:"trebuchet MS", Verdana, sans-ser
if;color:#FFF;
background-color:#555555;}
#content{margin:0 0 0 2%;position:relative;}
.content-container{background:#FFF;width:96%;margin-top:8px;padding:10px;position:relative;}
-->
</style>
</head>
<body>
<div id="header"><h1>Server Error</h1></div>
<div id="content">
 <div class="content-container"><fieldset>
  <h2>403 - Forbidden: Access is denied.</h2>
  <h3>You do not have permission to view this directory or page using the credentials that you suppl
ied.</h3>
 </fieldset></div>
</div>
</body>
</html>
```

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 Netsparker 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
ISO27001	A.14.2.5

7. Expect-CT Not Enabled

BEST PRACTICE | 1

Netsparker identified that Expect-CT is not enabled.

Certificate Transparency is a technology that makes impossible (or at least very difficult) for a CA to issue an SSL certificate for a domain without the certificate being visible to the owner of that domain.

Google announced that, starting with April 2018, if it runs into a certificate that is not seen in Certificate Transparency (CT) Log, it will consider that certificate invalid and reject the connection. Thus sites should serve certificate that takes place in CT Logs. While handshaking, sites should serve a valid Signed Certificate Timestamp (SCT) along with the certificate itself.

Expect-CT can also be used for detecting the compatibility of the certificates that are issued before the April 2018 deadline. For instance, a certificate that was signed before April 2018, for 10 years it will be still posing a risk and can be ignored by the certificate transparency policy of the browser. By setting Expect-CT header, you can prevent misissused certificates to be used.

Vulnerabilities

7.1. https://cdr-stage.cancer.gov/

Certainty

Request

```
GET / HTTP/1.1
Host: cdr-stage.cancer.gov
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; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/79.0.39
45.0 Safari/537.36
X-Scanner: NCI-Netsparker
```

Response

Response Time (ms) : 181.434 Total Bytes Received : 413 Body Length : 175 Is Compressed : No

```
HTTP/1.1 302 Redirect
Server: Microsoft-IIS/10.0
X-Powered-By: ASP.NET
Content-Length: 175
Content-Type: text/html; charset=UTF-8
Location: https://cdr-stage.cancer.gov/cgi-bin/secure/admin.py
Date: Tue, 22 Dec 2020 21:01:27 GMT

<head><title>Document Moved</title></head>
<body><h1>Object Moved</h1>This document may be found <a HREF="https://cdr-stage.cancer.gov/cgi-bin/
secure/admin.py">here</a></body>
```

Remedy

Configure your web server to respond with Expect-CT header.

```
Expect-CT: enforce, max-age=7776000, report-uri="https://ABSOLUTE_REPORT_URL"
```

Note: We strongly suggest you to use Expect-CT header in **report-only mode** first. If everything goes well and your certificate is ready, go with the Expect-CT enforce mode. To use **report-only mode** first, omit **enforce** flag and see the browser's behavior with your deployed certificate.

```
Expect-CT: max-age=7776000, report-uri="https://ABSOLUTE_REPORT_URL"
```

External References

- Expect-CT Extension for HTTP
- Expect-CT HTTP Header
- Expect-CT Header

CLASSIFICATION

CWE	16
WASC	15
ISO27001	A.14.1.2

8. Missing X-XSS-Protection Header

BEST PRACTICE | 1

Netsparker 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

8.1. https://cdr-stage.cancer.gov/cgi-bin/

Certainty

Request

```
GET /cgi-bin/ HTTP/1.1
Host: cdr-stage.cancer.gov
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; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/79.0.39
45.0 Safari/537.36
X-Scanner: NCI-Netsparker
```

Response

Response Time (ms) : 0 Total Bytes Received : 1394 Body Length : 1233 Is Compressed : No

```
HTTP/1.1 403 Forbidden
Server: Microsoft-IIS/10.0
X-Powered-By: ASP.NET
Content-Length: 1233
Content-Type: text/html
Date: Tue, 22 Dec 2020 21:01:37 GMT

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-str
ict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1"/>
<title>403 - Forbidden: Access is denied.</title>
<style type="text/css">
<!--
body{margin:0;font-size:.7em;font-family:Verdana, Arial, Helvetica, sans-serif;background:#EEEEEE;}
fieldset{padding:0 15px 10px 15px;}
h1{font-size:2.4em;margin:0;color:#FFF;}
h2{font-size:1.7em;margin:0;color:#CC0000;}
h3{font-size:1.2em;margin:10px 0 0 0;color:#000000;}
#header{width:96%;margin:0 0 0 0;padding:6px 2% 6px 2%;font-family:"trebuchet MS", Verdana, sans-ser
if;color:#FFF;
background-color:#555555;}
#content{margin:0 0 0 2%;position:relative;}
.content-container{background:#FFF;width:96%;margin-top:8px;padding:10px;position:relative;}
-->
</style>
</head>
<body>
<div id="header"><h1>Server Error</h1></div>
<div id="content">
 <div class="content-container"><fieldset>
  <h2>403 - Forbidden: Access is denied.</h2>
  <h3>You do not have permission to view this directory or page using the credentials that you suppl
ied.</h3>
 </fieldset></div>
</div>
</body>
</html>
```

Remedy

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

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

External References

- Internet Explorer 8 Security Features - MSDN
- X-XSS-Protection HTTP Header
- Internet Explorer 8 XSS Filter

CLASSIFICATION

CWE	16
WASC	15
HIPAA	164.308(A)
ISO27001	A.14.2.5

9. Referrer-Policy Not Implemented

BEST PRACTICE | 1

Netsparker 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

9.1. https://cdr-stage.cancer.gov/cgi-bin/

Certainty

Request

```
GET /cgi-bin/ HTTP/1.1
Host: cdr-stage.cancer.gov
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; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/79.0.39
45.0 Safari/537.36
X-Scanner: NCI-Netsparker
```

Response

Response Time (ms) : 0 Total Bytes Received : 1394 Body Length : 1233 Is Compressed : No

```
HTTP/1.1 403 Forbidden
Server: Microsoft-IIS/10.0
X-Powered-By: ASP.NET
Content-Length: 1233
Content-Type: text/html
Date: Tue, 22 Dec 2020 21:01:37 GMT

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-str
ict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1"/>
<title>403 - Forbidden: Access is denied.</title>
<style type="text/css">
<!--
body{margin:0;font-size:.7em;font-family:Verdana, Arial, Helvetica, sans-serif;background:#EEEEEE;}
fieldset{padding:0 15px 10px 15px;}
h1{font-size:2.4em;margin:0;color:#FFF;}
h2{font-size:1.7em;margin:0;color:#CC0000;}
h3{font-size:1.2em;margin:10px 0 0 0;color:#000000;}
#header{width:96%;margin:0 0 0 0;padding:6px 2% 6px 2%;font-family:"trebuchet MS", Verdana, sans-ser
if;color:#FFF;
background-color:#555555;}
#content{margin:0 0 0 2%;position:relative;}
.content-container{background:#FFF;width:96%;margin-top:8px;padding:10px;position:relative;}
-->
</style>
</head>
<body>
<div id="header"><h1>Server Error</h1></div>
<div id="content">
 <div class="content-container"><fieldset>
  <h2>403 - Forbidden: Access is denied.</h2>
  <h3>You do not have permission to view this directory or page using the credentials that you suppl
ied.</h3>
 </fieldset></div>
</div>
</body>
</html>
```

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
ISO27001	A.14.2.5

10. Subresource Integrity (SRI) Not Implemented

BEST PRACTICE | 1

Subresource Integrity (SRI) provides a mechanism to check integrity of the resource hosted by third parties like Content Delivery Networks (CDNs) and verifies that the fetched resource has been delivered without unexpected manipulation.

SRI does this using hash comparison mechanism. In this way, hash value declared in HTML elements (for now only script and link elements are supported) will be compared with the hash value of the resource hosted by third party.

Use of SRI is recommended as a best-practice, whenever libraries are loaded from a third-party source.

Vulnerabilities

10.1. https://cdr-stage.cancer.gov/cgi-bin/cdr/GuestUsers.py

Identified Sub Resource(s)

- https://ajax.googleapis.com/ajax/libs/jqueryui/1.12.1/themes/smoothness/jquery-ui.css
- https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js
- https://ajax.googleapis.com/ajax/libs/jqueryui/1.12.1/jquery-ui.min.js

Certainty

Request

```
GET /cgi-bin/cdr/GuestUsers.py HTTP/1.1
Host: cdr-stage.cancer.gov
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; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/79.0.39
45.0 Safari/537.36
X-Scanner: NCI-Netsparker
```

Response

Response Time (ms) : 1346.6344 Total Bytes Received : 1016 Body Length : 787 Is Compressed : No

```
HTTP/1.1 200 OK
Server: Microsoft-IIS/10.0
X-Powered-By: ASP.NET
Connection: close
Content-Length: 521
Content-Type: text/html;charset=utf-8
Content-Encoding:
Date: Tue, 22 Dec 2020 21:02:16 GMT
Vary: Accept-Encoding

<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<title>CDR Administration</title>
<link href="/favicon.ico" rel="icon">
<link href="https://ajax.googleapis.com/ajax/libs/jqueryui/1.12.1/themes/smoothness/jquery-ui.css" r
el="stylesheet">
<link href="../../stylesheets/cdr.css?v=201909071039" rel="stylesheet">
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script><script src=
"https://ajax.googleapis.com/ajax/libs/jqueryui/1.12.1/jquery-ui.min.js"></script>
</head>
<body id="cdr-page" class="admin-menu">
<header><h1>CDR Administration</h1>
<h2>Guest Users</h2></header><ol>
<li><a href="AdvancedSearch.py?Session=guest">Advanced Search</a></li>
<li><a href="TerminologyReports.py?Session=guest">Terminology Reports</a></li>
</ol>
</body>
</html>
```

Remedy

Using Subresource Integrity is simply to add *integrity* attribute to the *script* tag along with a base64 encoded cryptographic hash value.

```
<script src="https://code.jquery.com/jquery-2.1.4.min.js" integrity="sha384-
R4/ztc4ZlRqWjqIuvf6RX5yb/v90qNGx6fS48N0tRxiGkqveZETq72KgDVJCp2TC" crossorigin="anonymous"></script>
```

The hash algorithm must be one of **sha256**, **sha384** or **sha512**, followed by a '-' character.

External References

- Subresource Integrity
- Do not let your CDN betray you: Use Subresource Integrity
- Web Application Security with Subresource Integrity
- SRI Hash Generator

CLASSIFICATION

CWE	16
WASC	15
ISO27001	A.14.2.5

11. ASP.NET Identified

INFORMATION 1

Netsparker 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

11.1. https://cdr-stage.cancer.gov/

Certainty

Request

```
GET / HTTP/1.1
Host: cdr-stage.cancer.gov
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; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/79.0.39
45.0 Safari/537.36
X-Scanner: NCI-Netsparker
```

Response

Response Time (ms) : 181.434 Total Bytes Received : 413 Body Length : 175 Is Compressed : No

```
HTTP/1.1 302 Redirect
Server: Microsoft-IIS/10.0
X-Powered-By: ASP.NET
Content-Length: 175
Content-Type: text/html; charset=UTF-8
Location: https://cdr-stage.cancer.gov/cgi-bin/secure/admin.py
Date: Tue, 22 Dec 2020 21:01:27 GMT

<head><title>Document Moved</title></head>
<body><h1>Object Moved</h1>This document may be found <a HREF="https://cdr-stage.cancer.gov/cgi-bin/
secure/admin.py">here</a></body>
```

CLASSIFICATION

CWE	200
WASC	13
OWASP Proactive Controls	C7
ISO27001	A.8.1.1

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

12. Authorization Required

INFORMATION 1 CONFIRMED 1

Netsparker detected that authorization is required.

Impact

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

Vulnerabilities

12.1. https://cdr-stage.cancer.gov/cgi-bin/secure/admin.py

CONFIRMED

Type

- Digest

Request

```
GET /cgi-bin/secure/admin.py HTTP/1.1
Host: cdr-stage.cancer.gov
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Accept-Language: en-us,en;q=0.5
Cache-Control: no-cache
Referer: https://cdr-stage.cancer.gov/
User-Agent: Mozilla/5.0 (Windows NT 10.0; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/79.0.39
45.0 Safari/537.36
X-Scanner: NCI-Netsparker
```

Response

Response Time (ms) : 293.8924 Total Bytes Received : 1694 Body Length : 1293 Is Compressed : No

```
HTTP/1.1 401 Unauthorized
WWW-Authenticate: Digest qop="auth",algorithm=MD5-sess,nonce="+Upgraded+v13f44ee0aaf62a87f4ca72c374e083d294bbaaf04a6d8d601c9158895bce6fab94e967c0effc35d98dd8d76d8c5e1ac7704c043e62209545f",charset=utf-8,realm="Digest"

Server: Microsoft-IIS/10.0
X-Powered-By: ASP.NET
Connection: close
Content-Length: 1293
Content-Type: text/html
Date: Tue, 22 Dec 2020 21:04:20 GMT

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1"/>
<title>401 - Unauthorized: Access is denied due to invalid credentials.</title>
<style type="text/css">
<!--
body{margin:0;font-size:.7em;font-family:Verdana, Arial, Helvetica, sans-serif;background:#EEEEEE;}
fieldset{padding:0 15px 10px 15px;}
h1{font-size:2.4em;margin:0;color:#FFF;}
h2{font-size:1.7em;margin:0;color:#CC0000;}
h3{font-size:1.2em;margin:10px 0 0 0;color:#000000;}
#header{width:96%;margin:0 0 0 0;padding:6px 2% 6px 2%;font-family:"trebuchet MS", Verdana, sans-serif;color:#FFF;
background-color:#555555;}
#content{margin:0 0 0 2%;position:relative;}
.content-container{background:#FFF;width:96%;margin-top:8px;padding:10px;position:relative;}
-->
</style>
</head>
<body>
<div id="header"><h1>Server Error</h1></div>
<div id="content">
 <div class="content-container"><fieldset>
  <h2>401 - Unauthorized: Access is denied due to invalid credentials.</h2>
  <h3>You do not have permission to view this directory or page using the credentials that you supplied.</h3>
 </fieldset></div>
</div>
</body>
</html>
```

CLASSIFICATION

ISO27001	A.9.4.1

13. Forbidden Resource

INFORMATION 1 CONFIRMED 1

Netsparker identified a forbidden resource.

Access to this resource has been denied by the web server. This is generally not a security issue, and is reported here for informational purposes.

Impact

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

Vulnerabilities

13.1. https://cdr-stage.cancer.gov/cgi-bin/

CONFIRMED

Request

```
GET /cgi-bin/ HTTP/1.1
Host: cdr-stage.cancer.gov
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; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/79.0.39
45.0 Safari/537.36
X-Scanner: NCI-Netsparker
```

Response

Response Time (ms) : 0 Total Bytes Received : 1394 Body Length : 1233 Is Compressed : No

```
HTTP/1.1 403 Forbidden

Server: Microsoft-IIS/10.0
X-Powered-By: ASP.NET
Content-Length: 1233
Content-Type: text/html
Date: Tue, 22 Dec 2020 21:01:37 GMT

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-str
ict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1"/>
<title>403 - Forbidden: Access is denied.</title>
<style type="text/css">
<!--
body{margin:0;font-size:.7em;font-family:Verdana, Arial, Helvetica, sans-serif;background:#EEEEEE;}
fieldset{padding:0 15px 10px 15px;}
h1{font-size:2.4em;margin:0;color:#FFF;}
h2{font-size:1.7em;margin:0;color:#CC0000;}
h3{font-size:1.2em;margin:10px 0 0 0;color:#000000;}
#header{width:96%;margin:0 0 0 0;padding:6px 2% 6px 2%;font-family:"trebuchet MS", Verdana, sans-ser
if;color:#FFF;
background-color:#555555;}
#content{margin:0 0 0 2%;position:relative;}
.content-container{background:#FFF;width:96%;margin-top:8px;padding:10px;position:relative;}
-->
</style>
</head>
<body>
<div id="header"><h1>Server Error</h1></div>
<div id="content">
 <div class="content-container"><fieldset>
  <h2>403 - Forbidden: Access is denied.</h2>
  <h3>You do not have permission to view this directory or page using the credentials that you suppl
ied.</h3>
 </fieldset></div>
</div>
</body>
</html>
```

CLASSIFICATION

OWASP Proactive Controls	C8

ISO27001	A.8.1.1

14. Version Disclosure (IIS)

INFORMATION 1

Netsparker identified a version disclosure (IIS) in 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

14.1. https://cdr-stage.cancer.gov/

Extracted Version

- 10.0

Certainty

Request

```
GET / HTTP/1.1
Host: cdr-stage.cancer.gov
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; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/79.0.39
45.0 Safari/537.36
X-Scanner: NCI-Netsparker
```

Response

Response Time (ms) : 181.434 Total Bytes Received : 413 Body Length : 175 Is Compressed : No

```
HTTP/1.1 302 Redirect
Server: Microsoft-IIS/10.0

X-Powered-By: ASP.NET
Content-Length: 175
Content-Type: text/html; charset=UTF-8
Location: https://cdr-stage.cancer.gov/cgi-bin/secure/admin.py
Date: Tue, 22 Dec 2020 21:01:27 GMT

<head><title>Document Moved</title></head>
<body><h1>Object Moved</h1>This document may be found <a HREF="https://cdr-stage.cancer.gov/cgi-bin/
secure/admin.py">here</a></body>
```

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	45
HIPAA	164.306(A), 164.308(A)
OWASP Proactive Controls	C7
ISO27001	A.18.1.3

Show Scan Detail

Enabled Security Checks	: BREACH Attack, Code Evaluation, Code Evaluation (Out of Band), Command Injection, Command Injection (Blind), 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), Expect Certificate Transparency (Expect-CT), File Upload, Header Analyzer, Heartbleed, HSTS, HTML Content, HTTP Header Injection, HTTP Methods, HTTP Status, HTTP.sys (CVE-2015-1635), IFrame Security, Insecure JSONP Endpoint, Insecure Reflected Content, JavaScript Libraries, Local File Inclusion, Login Page Identifier, Mixed Content, Open Redirection, Referrer Policy, Reflected File Download, Remote File Inclusion, Reverse Proxy Detection, Server-Side Request Forgery (DNS), Server-Side Request Forgery (IP Combinations), Server-Side Request Forgery (Pattern Based), Server-Side Template Injection, Signatures, SQL Injection (Blind), SQL Injection (Boolean), SQL Injection (Error Based), SQL Injection (Out of Band), SSL, Static Resources (All Paths), Unicode Transformation (Best-Fit Mapping), WAF Identifier, Web App Fingerprint, Web Cache Deception, WebDAV, Windows Short Filename,

	XML External Entity, XML External Entity (Out of Band)
URL Rewrite Mode	: Heuristic
Detected URL Rewrite Rule(s)	: None
Excluded URL Patterns	: gtm\.js WebResource\.axd ScriptResource\.axd
Authentication	: Form Authentication
Scheduled	: No
Additional Website(s)	: None

This report created with 5.9.1.29030-master-d201dea
https://www.netsparker.com