Acunetix
by Invicti

Affected Items Report

Acunetix Security Audit

2025-11-17

Generated by Acunetix

Scan of 192.168.1.8

Scan details

Scan information	
Start time	2025-11-17T05:02:08.611868+00:00
Start url	http://192.168.1.8/site/
Host	192.168.1.8
Scan time	5 minutes, 21 seconds
Profile	Full Scan
Server information	Apache/2.4.29 (Ubuntu)
Responsive	True
Server OS	Unix
Application build	24.6.240626115

Threat level

Acunetix Threat Level 2

One or more medium-severity type vulnerabilities have been discovered by the scanner. You should investigate each of these vulnerabilities to ensure they will not escalate to more severe problems.

Alerts distribution

Total alerts found	12
Critical	0
High	0
Medium	8
Low	0
Informational	4

Affected items

Web Server	
Alert group	**Directory listings (verified)**
Severity	Medium
Description	Directory listing is a web server function that displays the directory contents when there is no index file in a specific website directory. It is dangerous to leave this function turned on for the web server because it leads to information disclosure.
Recommendations	You should make sure no sensitive information is disclosed or you may want to restrict directory listings from the web server configuration.
Alert variants	
Details	Folders with directory listing enabled: • http://192.168.1.8/ • http://192.168.1.8/site/css/ • http://192.168.1.8/site/js/ • http://192.168.1.8/site/images/

```
GET / HTTP/1.1
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Encoding: gzip,deflate,br
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/125.0.0.0 Safari/537.36
Host: 192.168.1.8
Connection: Keep-alive
```

Web Server	
Alert group	**Insecure HTTP Usage**
Severity	Medium
Description	It was detected that your web application uses HTTP protocol, but doesn't automatically redirect users to HTTPS.
Recommendations	It's recommended to implement best practices of HTTP Redirection into your web application. Consult web references for more information
Alert variants	
Details	

```
GET / HTTP/1.1
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Encoding: gzip,deflate,br
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/125.0.0.0 Safari/537.36
Host: 192.168.1.8
Connection: Keep-alive
```

Web Server	
Alert group	**SSL/TLS Not Implemented (verified)**
Severity	Medium
Description	This scan target was connected to over an unencrypted connection. A potential attacker can intercept and modify data sent and received from this site.
Recommendations	The site should send and receive data over a secure (HTTPS) connection.
Alert variants	
Details	

```
GET /site/ HTTP/1.1
Referer: http://192.168.1.8/site/
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Encoding: gzip,deflate,br
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/125.0.0.0 Safari/537.36
Host: 192.168.1.8
Connection: Keep-alive
```

Web Server	
Alert group	**Virtual host directory listing**
Severity	Medium
Description	This web server is responding with a directory listing when the Host header is manipulated and various common virtual hosts and/or IP addresses are tested. This is a web server misconfiguration and should be fixed as it may disclose sensitive information to an attacker. Consult Attack details for more information.
Recommendations	Fix virtual hosts configuration to resolve this problem.
Alert variants	
Details	Virtual host: **kTuodyCw** Response: `Last modified</a>`

```
GET / HTTP/1.1
Host: kTuodyCw
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Encoding: gzip,deflate,br
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/125.0.0.0 Safari/537.36
Connection: Keep-alive
```

Web Server	
Alert group	**Vulnerable JavaScript libraries**
Severity	Medium
Description	You are using one or more vulnerable JavaScript libraries. One or more vulnerabilities were reported for this version of the library. Consult Attack details and Web References for more information about the affected library and the vulnerabilities that were reported.
Recommendations	Upgrade to the latest version.
Alert variants	

Details	• **jQuery 3.4.1** ◦ URL: http://192.168.1.8/site/ ◦ Detection method: The library's name and version were determined based on its dynamic behavior. ◦ CVE-ID: CVE-2020-11022, CVE-2020-11023 ◦ Description: In jQuery versions greater than or equal to 1.2 and before 3.5.0, passing HTML from untrusted sources - even after sanitizing it - to one of jQuery's DOM manipulation methods (i.e. .html(), .append(), and others) may execute untrusted code. This problem is patched in jQuery 3.5.0. / In jQuery versions greater than or equal to 1.0.3 and before 3.5.0, passing HTML containing option elements from untrusted sources - even after sanitizing it - to one of jQuery's DOM manipulation methods (i.e. .html(), .append(), and others) may execute untrusted code. This problem is patched in jQuery 3.5.0. ◦ References: ▪ https://blog.jquery.com/2020/04/10/jquery-3-5-0-released/ ▪ https://mksben.l0.cm/2020/05/jquery3.5.0-xss.html ▪ https://jquery.com/upgrade-guide/3.5/ ▪ https://api.jquery.com/jQuery.htmlPrefilter/ ▪ https://www.cvedetails.com/cve/CVE-2020-11022/ ▪ https://github.com/advisories/GHSA-gxr4-xjj5-5px2 ▪ https://www.cvedetails.com/cve/CVE-2020-11023/ ▪ https://github.com/advisories/GHSA-jpcq-cgw6-v4j6

```
GET /site/ HTTP/1.1
Referer: http://192.168.1.8/site/
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Encoding: gzip,deflate,br
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/125.0.0.0 Safari/537.36
Host: 192.168.1.8
Connection: Keep-alive
```

Web Server	
Alert group	**jQuery Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting') Vulnerability**
Severity	Medium
Description	In jQuery versions greater than or equal to 1.0.3 and before 3.5.0, passing HTML containing <option> elements from untrusted sources - even after sanitizing it - to one of jQuery's DOM manipulation methods (i.e. .html(), .append(), and others) may execute untrusted code. This problem is patched in jQuery 3.5.0.
Recommendations	
Alert variants	
Details	jquery v3.4.1-3.4.1

Web Server	
Alert group	**jQuery Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting') Vulnerability**
Severity	Medium
Description	Cross Site Scripting vulnerability in jQuery 2.2.0 through 3.x before 3.5.0 allows a remote attacker to execute arbitrary code via the <options> element.
Recommendations	
Alert variants	
Details	jquery v3.4.1-3.4.1

Web Server	
Alert group	**jQuery Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting') Vulnerability**
Severity	Medium
Description	In jQuery versions greater than or equal to 1.2 and before 3.5.0, passing HTML from untrusted sources - even after sanitizing it - to one of jQuery's DOM manipulation methods (i.e. .html(), .append(), and others) may execute untrusted code. This problem is patched in jQuery 3.5.0.
Recommendations	
Alert variants	
Details	jquery v3.4.1-3.4.1

Web Server	
Alert group	**Content Security Policy (CSP) Not Implemented**
Severity	Informational

Description	Content Security Policy (CSP) is an added layer of security that helps to detect and mitigate certain types of attacks, including Cross Site Scripting (XSS) and data injection attacks.
	
	Content Security Policy (CSP) can be implemented by adding a **Content-Security-Policy** header. The value of this header is a string containing the policy directives describing your Content Security Policy. To implement CSP, you should define lists of allowed origins for the all of the types of resources that your site utilizes. For example, if you have a simple site that needs to load scripts, stylesheets, and images hosted locally, as well as from the jQuery library from their CDN, the CSP header could look like the following:
	
	`Content-Security-Policy:`
	`  default-src 'self';`
	`  script-src 'self' https://code.jquery.com;`
	
	It was detected that your web application doesn't implement Content Security Policy (CSP) as the CSP header is missing from the response. It's recommended to implement Content Security Policy (CSP) into your web application.

Recommendations	It's recommended to implement Content Security Policy (CSP) into your web application. Configuring Content Security Policy involves adding the **Content-Security-Policy** HTTP header to a web page and giving it values to control resources the user agent is allowed to load for that page.
Alert variants	
Details	Paths without CSP header: • http://192.168.1.8/site/ • http://192.168.1.8/site/images/ • http://192.168.1.8/site/css/ • http://192.168.1.8/site/js/

```
GET /site/ HTTP/1.1
Referer: http://192.168.1.8/site/
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Encoding: gzip,deflate,br
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/125.0.0.0 Safari/537.36
Host: 192.168.1.8
Connection: Keep-alive
```

Web Server	
Alert group	**Error page web server version disclosure**
Severity	Informational
Description	Application errors or warning messages may disclose sensitive information about an application's internal workings to an attacker. Acunetix found the web server version number and a list of modules enabled on the target server. Consult the 'Attack details' section for more information about the affected page.
Recommendations	Properly configure the web server not to disclose information about an application's internal workings to the user. Consult the 'Web references' section for more information.
Alert variants	
Details	

```
GET /qmE1aSk7rO HTTP/1.1
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Encoding: gzip,deflate,br
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/125.0.0.0 Safari/537.36
Host: 192.168.1.8
Connection: Keep-alive
```

Web Server	
Alert group	**Permissions-Policy header not implemented**
Severity	Informational
Description	The Permissions-Policy header allows developers to selectively enable and disable use of various browser features and APIs.
Recommendations	
Alert variants	
Details	Locations without Permissions-Policy header: • http://192.168.1.8/site/ • http://192.168.1.8/site/images/ • http://192.168.1.8/site/css/ • http://192.168.1.8/site/js/

```
GET /site/ HTTP/1.1
Referer: http://192.168.1.8/site/
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Encoding: gzip,deflate,br
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/125.0.0.0 Safari/537.36
Host: 192.168.1.8
Connection: Keep-alive
```

/site/	
Alert group	**Subresource Integrity (SRI) Not Implemented**
Severity	Informational
Description	Subresource Integrity (SRI) is a security feature that enables browsers to verify that third-party resources they fetch (for example, from a CDN) are delivered without unexpected manipulation. It works by allowing developers to provide a cryptographic hash that a fetched file must match. Third-party resources (such as scripts and stylesheets) can be manipulated. An attacker that has access or has hacked the hosting CDN can manipulate or replace the files. SRI allows developers to specify a base64-encoded cryptographic hash of the resource to be loaded. The integrity attribute containing the hash is then added to the <script> HTML element tag. The integrity string consists of a base64-encoded hash, followed by a prefix that depends on the hash algorithm. This prefix can either be sha256, sha384 or sha512. The script loaded from the external URL specified in the Details section doesn't implement Subresource Integrity (SRI). It's recommended to implement Subresource Integrity (SRI) for all the scripts loaded from external hosts.
Recommendations	Use the SRI Hash Generator link (from the References section) to generate a <script> element that implements Subresource Integrity (SRI). For example, you can use the following <script> element to tell a browser that before executing the https://example.com/example-framework.js script, the browser must first compare the script to the expected hash, and verify that there's a match. `<script src="https://example.com/example-framework.js" integrity="sha384-oqVuAfXRKap7fdgcCY5uykM6+R9GqQ8K/uxy9rx7HN crossorigin="anonymous"></script>`
Alert variants	
Details	Pages where SRI is not implemented: • http://192.168.1.8/site/ Script SRC: **https://ajax.googleapis.com/ajax/libs/webfont/1.6.26/webfont.js**

```
GET /site/ HTTP/1.1
Referer: http://192.168.1.8/site/
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Encoding: gzip,deflate,br
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/125.0.0.0 Safari/537.36
Host: 192.168.1.8
Connection: Keep-alive
```

Scanned items (coverage report)

http://192.168.1.8/
http://192.168.1.8/site/
http://192.168.1.8/site/css/
http://192.168.1.8/site/css/normalize.css
http://192.168.1.8/site/css/split-opl.webflow.css
http://192.168.1.8/site/css/webflow.css
http://192.168.1.8/site/images/
http://192.168.1.8/site/js/
http://192.168.1.8/site/js/webflow.js