Acunetix
by Invicti

Scan Comparison

Acunetix Security Audit

2024-06-09

Generated by Acunetix

Scan comparison

Scan details

Start URL	
First scan	http://192.168.193.130:3000/#/
Second scan	http://192.168.193.130:3000/#/

Threat levels

First scan	Second scan
Acunetix Threat Level 4	Acunetix Threat Level 4

Alert counts

First scan		Second scan	
Total alerts found	25	Total alerts found	34
Critical	2	Critical	2
High	0	High	6
Medium	8	Medium	8
Low	2	Low	3
Informational	13	Informational	15

Newly discovered issues

API Sensitive Info(PII) accessible without authentication

Severity	**High**
Reported by module	/httpdata/api_sensitive_info_exposure.js

Description

The API exposes sensitive information (Personally Identifiable Information (PII)) due to a vulnerability in the authorization process. An unauthenticated attacker can gain access to the personal data.

Impact

Sensitive info exposure

Affected items

/rest/memories
Details
Field name: email **Sensitive value:** bjoern@owasp.org **Description:** Email
/rest/memories/
Details
Field name: email **Sensitive value:** bjoern@owasp.org **Description:** Email
/rest/products/1/reviews
Details
Field name: author **Sensitive value:** admin@juice-sh.op **Description:** Email
/rest/products/30/reviews
Details
Field name: author **Sensitive value:** uvogin@juice-sh.op **Description:** Email
/rest/products/42/reviews
Details
Field name: author **Sensitive value:** stan@juice-sh.op **Description:** Email
/rest/products/6/reviews
Details
Field name: author **Sensitive value:** bender@juice-sh.op **Description:** Email

[Possible] Internal IP Address Disclosure

Severity	**Low**
Reported by module	/httpdata/text_search.js

Description

One or more strings matching an internal IPv4 address were found. These IPv4 addresses may disclose information about the IP addressing scheme of the internal network. This information can be used to conduct further attacks.

The significance of this finding should be confirmed manually.

Impact

Possible sensitive information disclosure.

Affected items

Web Server
Details
Pages with internal IPs: • http://192.168.193.130:3000/profile **192.168.193.1**

(Possible) Cross site scripting

Severity	**Informational**
Reported by module	/Scripts/PerScheme/XSS.script

Description

This script is possibly vulnerable to Cross Site Scripting (XSS) attacks.

Due to the Content-type header of the response (JSON), exploitation of this vulnerability is not possible. The issue might be indirectly exploitable if a client-side script processes the response and embeds it into an HTML context. Manual confirmation is required.

Cross site scripting (also referred to as XSS) is a vulnerability that allows an attacker to send malicious code (usually in the form of Javascript) to another user. Because a browser cannot know if the script should be trusted or not, it will execute the script in the user context allowing the attacker to access any cookies or session tokens retained by the browser.

Impact

Malicious users may inject JavaScript, VBScript, ActiveX, HTML or Flash into a vulnerable application to fool a user in order to gather data from them. An attacker can steal the session cookie and take over the account, impersonating the user. It is also possible to modify the content of the page presented to the user.

Affected items

/api/BasketItems
Details

/api/Complaints
Details

/api/Users
Details

Undetected issues

(Possible) Cross site scripting

Severity	**Informational**
Reported by module	/Scripts/PerScheme/XSS.script

Description

This script is possibly vulnerable to Cross Site Scripting (XSS) attacks.

Due to the Content-type header of the response (JSON), exploitation of this vulnerability is not possible. The issue might be indirectly exploitable if a client-side script processes the response and embeds it into an HTML context. Manual confirmation is required.

Cross site scripting (also referred to as XSS) is a vulnerability that allows an attacker to send malicious code (usually in the form of Javascript) to another user. Because a browser cannot know if the script should be trusted or not, it will execute the script in the user context allowing the attacker to access any cookies or session tokens retained by the browser.

Impact

Malicious users may inject JavaScript, VBScript, ActiveX, HTML or Flash into a vulnerable application to fool a user in order to gather data from them. An attacker can steal the session cookie and take over the account, impersonating the user. It is also possible to modify the content of the page presented to the user.

Affected items

/api/SecurityQuestions/
Details

Unchanged issues

SQL Injection

Severity	**Critical**
Reported by module	/Scripts/PerScheme/Sql_Injection.script

Description

SQL injection (SQLi) refers to an injection attack wherein an attacker can execute malicious SQL statements that control a web application's database server.

Impact

An attacker can use SQL injection to bypass a web application's authentication and authorization mechanisms and retrieve the contents of an entire database. SQLi can also be used to add, modify and delete records in a database, affecting data integrity. Under the right circumstances, SQLi can also be used by an attacker to execute OS commands, which may then be used to escalate an attack even further.

Affected items

/rest/products/search

Details

URL encoded GET input **q** was set to **'"**

Error message found:

```
near \"\"%' OR description LIKE '%'\"\": syntax error
```

/rest/user/login

Details

JSON input **email** was set to **-1' OR 3*2*1=6 AND 000149=000149 --**

Tests performed:

- -1' OR 2+149-149-1=0+0+0+1 -- => **TRUE**
- -1' OR 3+149-149-1=0+0+0+1 -- => **FALSE**
- -1' OR 3*2<(0+5+149-149) -- => **FALSE**
- -1' OR 3*2>(0+5+149-149) -- => **FALSE**
- -1' OR 2+1-1+1=1 AND 000149=000149 -- => **FALSE**
- -1' OR 3*2=5 AND 000149=000149 -- => **FALSE**
- -1' OR 3*2=6 AND 000149=000149 -- => **TRUE**
- -1' OR 3*2*0=6 AND 000149=000149 -- => **FALSE**
- -1' OR 3*2*1=6 AND 000149=000149 -- => **TRUE**

Original value: **1**

Insecure HTTP Usage

Severity	**Medium**
Reported by module	/target/http_redirections.js

Description

It was detected that your web application uses HTTP protocol, but doesn't automatically redirect users to HTTPS.

Impact

In some circumstances, it could be used for a man-in-the-middle (MitM) attack

Affected items

Web Server
Details

jQuery Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting') Vulnerability

Severity	**Medium**
Reported by module	/httpdata/javascript_library_audit_external.js

Description

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.

Impact

Affected items

Web Server
Details
jquery v2.2.4-2.2.4

jQuery Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting') Vulnerability

Severity	**Medium**
Reported by module	/httpdata/javascript_library_audit_external.js

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.

Impact

Affected items

Web Server
Details
jquery v2.2.4-2.2.4

jQuery Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting') Vulnerability

Severity	**Medium**
Reported by module	/httpdata/javascript_library_audit_external.js

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.

Impact

Affected items

Web Server
Details
jquery v2.2.4-2.2.4

jQuery Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting') Vulnerability

Severity	**Medium**
Reported by module	/httpdata/javascript_library_audit_external.js

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.

Impact

Affected items

Web Server
Details
jquery v2.2.4-2.2.4

JQuery Prototype Pollution Vulnerability

Severity	**Medium**
Reported by module	/httpdata/javascript_library_audit_external.js

Description

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.

Impact

Affected items

Web Server
Details
jquery v2.2.4-2.2.4

SSL/TLS Not Implemented (verified)

Severity	**Medium**
Reported by module	/RPA/no_https.js

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.

Impact

Possible information disclosure.

Affected items

Web Server
Details

Vulnerable JavaScript libraries

Severity	**Medium**
Reported by module	/httpdata/javascript_library_audit_external.js

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.

Impact

Consult References for more information.

Affected items

Web Server
Details

- **jQuery 2.2.4**
  - URL: //cdnjs.cloudflare.com/ajax/libs/jquery/2.2.4/jquery.min.js
  - Detection method: The library's name and version were determined based on the file's CDN URI.
  - CVE-ID: CVE-2015-9251, CVE-2020-11022, CVE-2020-11023, CVE-2019-11358
  - Description: Possible Cross Site Scripting via third-party text/javascript responses (1.12.0-1.12.2 mitigation reverted) / 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. / jQuery 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.
  - References:
    - https://github.com/jquery/jquery/issues/2432
    - 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
    - https://github.com/jquery/jquery/pull/4333
    - https://nvd.nist.gov/vuln/detail/CVE-2019-11358
    - https://nvd.nist.gov/vuln/detail/CVE-2019-5428
    - https://blog.jquery.com/2019/04/10/jquery-3-4-0-released/

Programming Error Messages

Severity	**Low**
Reported by module	/Scripts/PerScheme/Error_Message.script

Description

This alert requires manual confirmation

Acunetix found one or more error/warning messages. Application error or warning messages may expose sensitive information about an application's internal workings to an attacker.
These messages may also contain the location of the file that produced an unhandled exception.
Consult the 'Attack details' section for more information about the affected page(s).

Impact

Error messages may disclose sensitive information which can be used to escalate attacks.

Affected items

Web Server
Details

Application error messages:

- http://192.168.193.130:3000/rest/user/reset-password
  SyntaxError: Unexpected token
- http://192.168.193.130:3000/api/Feedbacks/
  SyntaxError: Unexpected token
- http://192.168.193.130:3000/rest/user/login
  SyntaxError: Unexpected token

Unrestricted access to Prometheus Metrics

Severity	**Low**
Reported by module	/target/open_prometheus.js

Description

Prometheus is a monitoring system and time series database

Acunetix determined that it was possible to access without authentication a web application's metrics exposed for Prometheus.

Impact

Possible sensitive information disclosure

Affected items

Web Server
Details

(Possible) Cross site scripting

Severity	**Informational**
Reported by module	/Scripts/PerScheme/XSS.script

Description

This script is possibly vulnerable to Cross Site Scripting (XSS) attacks.

Due to the Content-type header of the response (JSON), exploitation of this vulnerability is not possible. The issue might be indirectly exploitable if a client-side script processes the response and embeds it into an HTML context. Manual confirmation is required.

Cross site scripting (also referred to as XSS) is a vulnerability that allows an attacker to send malicious code (usually in the form of Javascript) to another user. Because a browser cannot know if the script should be trusted or not, it will execute the script in the user context allowing the attacker to access any cookies or session tokens retained by the browser.

Impact

Malicious users may inject JavaScript, VBScript, ActiveX, HTML or Flash into a vulnerable application to fool a user in order to gather data from them. An attacker can steal the session cookie and take over the account, impersonating the user. It is also possible to modify the content of the page presented to the user.

Affected items

/api/Challenges
Details
/api/Challenges/
Details
/api/Feedbacks
Details
/api/Feedbacks/
Details
/api/Products
Details
/api/Quantitys
Details
/api/Quantitys/
Details
/api/SecurityQuestions
Details

[Possible] Internal Path Disclosure (*nix)

Severity	**Informational**
Reported by module	/httpdata/text_search.js

Description

One or more fully qualified path names were found. From this information the attacker may learn the file system structure from the web server. This information can be used to conduct further attacks.

This alert may be a false positive, manual confirmation is required.

Impact

Possible sensitive information disclosure.

Affected items

Web Server
Details

Pages with paths being disclosed:

- http://192.168.193.130:3000/redirect
  (/home/huy/juice
- http://192.168.193.130:3000/api/
  /home/huy/juice
- http://192.168.193.130:3000/rest/
  /home/huy/juice
- http://192.168.193.130:3000/rest/products/
  /home/huy/juice
- http://192.168.193.130:3000/rest/admin/
  /home/huy/juice
- http://192.168.193.130:3000/rest/user/
  /home/huy/juice
- http://192.168.193.130:3000/api/Feedbacks/
  (/home/huy/juice
- http://192.168.193.130:3000/rest/user/security-question
  (/home/huy/juice

Access-Control-Allow-Origin header with wildcard (*) value

Severity	**Informational**
Reported by module	/location/cors_origin_validation.js

Description

Cross-origin resource sharing (CORS) is a mechanism that allows restricted resources (e.g. fonts) on a web page to be requested from another domain outside the domain from which the resource originated. The Access-Control-Allow-Origin header indicates whether a resource can be shared based on the value of the Origin request header, "*****", or "null" in the response.

If a website responds with Access-Control-Allow-Origin: ***** the requested resource allows sharing with every origin. Therefore, any website can make XHR (XMLHTTPRequest) requests to the site and access the responses.

Impact

Any website can make XHR requests to the site and access the responses.

Affected items

Web Server
Details

Affected paths (max. 25):

- /
- /assets/public/
- /clientaccesspolicy.xml
- /assets/
- /ftp
- /ftp/coupons_2013.md.bak
- /crossdomain.xml
- /ftp/eastere.gg
- /ftp/encrypt.pyc
- /ftp/
- /ftp/suspicious_errors.yml
- /ftp/package.json.bak
- /ftp/quarantine
- /ftp/quarantine/
- /rest/
- /rest/products/search
- /rest/admin/application-configuration
- /rest/user/reset-password
- /assets/i18n/en.json
- /rest/user/security-question
- /api/

Content Security Policy (CSP) Not Implemented

Severity	**Informational**
Reported by module	/httpdata/CSP_not_implemented.js

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.

Impact

CSP can be used to prevent and/or mitigate attacks that involve content/code injection, such as cross-site scripting/XSS attacks, attacks that require embedding a malicious resource, attacks that involve malicious use of iframes, such as clickjacking attacks, and others.

Affected items

Web Server
Details
Paths without CSP header: • http://192.168.193.130:3000/ • http://192.168.193.130:3000/assets/public/ • http://192.168.193.130:3000/clientaccesspolicy.xml • http://192.168.193.130:3000/ftp • http://192.168.193.130:3000/crossdomain.xml • http://192.168.193.130:3000/assets/ • http://192.168.193.130:3000/ftp/quarantine • http://192.168.193.130:3000/ftp/ • http://192.168.193.130:3000/ftp/quarantine/ • http://192.168.193.130:3000/assets/public/images/uploads/%F0%9F%98%BC- • http://192.168.193.130:3000/assets/public/images/uploads/ • http://192.168.193.130:3000/assets/public/images/ • http://192.168.193.130:3000/assets/i18n/

Permissions-Policy header not implemented

Severity	**Informational**
Reported by module	/httpdata/permissions_policy.js

Description

The Permissions-Policy header allows developers to selectively enable and disable use of various browser features and APIs.

Impact

Affected items

Web Server
Details

Locations without Permissions-Policy header:

- http://192.168.193.130:3000/
- http://192.168.193.130:3000/assets/public/
- http://192.168.193.130:3000/clientaccesspolicy.xml
- http://192.168.193.130:3000/ftp
- http://192.168.193.130:3000/ftp/coupons_2013.md.bak
- http://192.168.193.130:3000/crossdomain.xml
- http://192.168.193.130:3000/assets/
- http://192.168.193.130:3000/ftp/eastere.gg
- http://192.168.193.130:3000/ftp/encrypt.pyc
- http://192.168.193.130:3000/ftp/suspicious_errors.yml
- http://192.168.193.130:3000/ftp/quarantine
- http://192.168.193.130:3000/ftp/
- http://192.168.193.130:3000/ftp/package.json.bak
- http://192.168.193.130:3000/ftp/quarantine/
- http://192.168.193.130:3000/socket.io/
- http://192.168.193.130:3000/api/Feedbacks/
- http://192.168.193.130:3000/assets/public/images/uploads/%F0%9F%98%BC-
- http://192.168.193.130:3000/rest/user/login
- http://192.168.193.130:3000/assets/public/images/uploads/
- http://192.168.193.130:3000/assets/public/images/
- http://192.168.193.130:3000/assets/i18n/