

Website Vulnerability Scanner Report

✓ <https://sumasys.analyticalcenter.id/>

Summary

Overall risk level:

Medium

Risk ratings:

Critical: 0

High: 0

Medium: 1

Low: 11

Info: 0

Scan information:

Start time: Apr 06, 2026 / 09:54:01 UTC+07

Finish time: Apr 06, 2026 / 18:46:05 UTC+07

Scan duration: 8 hrs, 52 min, 4 sec

Tests performed: 75

VPN Profile: KN VPN

Scan status: **Finished**

Findings

CORS misconfiguration

port 443/tcp

CONFIRMED

URL	Method	Summary
https://sumasys.analyticalcenter.id/	GET	We injected the value <code>null</code> inside the Origin header. The server responded with a Access-Control-Allow-Origin header with a value of <code>*</code> , indicating that it accepts CORS requests from arbitrary origins.

▼ Details

Risk description:

The risk is that any sensitive content hosted on the application can be read in the browser by JavaScript on any attacker controlled domain. In combination with a true value for the **Access-Control-Allow-Credentials** header, an attacker could read the data of a user authenticated on your application. As an example, if Gmail was affected by this vulnerability, it would allow an attacker to read all the emails of any user who visits his malicious website.

Recommendation:

We recommend that instead of parsing the **Origin** header yourself you compare it against a whitelist of allowed domain names that you define. If this is not possible, and your application requires the ability to dynamically generate the **Access-Control-Allow-Origin** header, we recommend that you use the standard URL parsing library that comes bundled with your backend programming language. These libraries have been battle tested and are generally less prone to parsing errors than a custom built regex. Additionally, we recommend that you never allow cross-origin requests on pages containing sensitive data. At the very least, if a page contains sensitive information, put it behind authentication and set the value of the **Access-Control-Allow-Credentials** header to `false`. This will tell browsers not to send any cookies or authentication headers with XHR requests.

References:

<https://developer.mozilla.org/en-US/docs/Web/HTTP/CORS>

<https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Access-Control-Allow-Origin>

<https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Access-Control-Allow-Credentials>

Classification:

CWE : CWE-942

OWASP Top 10 - 2017 : **A6 - Security Misconfiguration**

OWASP Top 10 - 2021 : **A5 - Security Misconfiguration**

OWASP Top 10 - 2025 : **A02 - Security Misconfiguration**

Unsafe security header: Content-Security-Policy

port 443/tcp

CONFIRMED

URL	Evidence
-----	----------

https://sumasys.analyticalcenter.id/	Response headers include the HTTP Content-Security-Policy security header with the following security issues: <pre>default-src: data: URI in default-src allows the execution of unsafe scripts. script-src: 'unsafe-eval' allows the execution of code injected into DOM APIs such as eval(). script-src: 'self' can be problematic if you host JSONP, Angular or user uploaded files. default-src: https: URI in default-src allows the execution of unsafe scripts. script-src: 'unsafe-inline' allows the execution of unsafe in-page scripts and event handler s. object-src: We recommend restricting object-src to 'none'.</pre> Request / Response
---	--

▼ Details

Risk description: For example, if the unsafe-inline directive is present in the CSP header, the execution of inline scripts and event handlers is allowed. This can be exploited by an attacker to execute arbitrary JavaScript code in the context of the vulnerable application. Recommendation: Remove the unsafe values from the directives, adopt nonces or hashes for safer inclusion of inline scripts if they are needed, and explicitly define the sources from which scripts, styles, images or other resources can be loaded. References: https://cheatsheetseries.owasp.org/cheatsheets/Content_Security_Policy_Cheat_Sheet.html https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Content-Security-Policy Classification: CWE : CWE-1021 OWASP Top 10 - 2017 : A6 - Security Misconfiguration OWASP Top 10 - 2021 : A5 - Security Misconfiguration OWASP Top 10 - 2025 : A02 - Security Misconfiguration

 Robots.txt file found CONFIRMED

port 443/tcp

URL
https://sumasys.analyticalcenter.id/robots.txt

▼ Details

Risk description: There is no particular security risk in having a robots.txt file. However, it's important to note that adding endpoints in it should not be considered a security measure, as this file can be directly accessed and read by anyone. Recommendation: We recommend you to manually review the entries from robots.txt and remove the ones which lead to sensitive locations in the website (ex. administration panels, configuration files, etc). References: https://www.theregister.co.uk/2015/05/19/robotstxt/ Classification: OWASP Top 10 - 2017 : A6 - Security Misconfiguration OWASP Top 10 - 2021 : A5 - Security Misconfiguration OWASP Top 10 - 2025 : A06 - Insecure Design Screenshot:
--

User-agent: *
Disallow: *

Figure 1. robots.txt

Missing security header: Strict-Transport-Security

CONFIRMED

port 443/tcp

URL	Evidence
https://sumasys.analyticalcenter.id/import-price-item	Response headers do not include the HTTP Strict-Transport-Security header Request / Response

Details

Risk description:

The risk is that lack of this header permits an attacker to force a victim user to initiate a clear-text HTTP connection to the server, thus opening the possibility to eavesdrop on the network traffic and extract sensitive information (e.g. session cookies).

Recommendation:

The Strict-Transport-Security HTTP header should be sent with each HTTPS response. The syntax is as follows:

```
Strict-Transport-Security: max-age=<seconds>[; includeSubDomains]
```

The parameter `max-age` gives the time frame for requirement of HTTPS in seconds and should be chosen quite high, e.g. several months. A value below 7776000 is considered as too low by this scanner check.

The flag `includeSubDomains` defines that the policy applies also for sub domains of the sender of the response.

Classification:

CWE : [CWE-693](#)

OWASP Top 10 - 2017 : [A6 - Security Misconfiguration](#)

OWASP Top 10 - 2021 : [A5 - Security Misconfiguration](#)

OWASP Top 10 - 2025 : [A02 - Security Misconfiguration](#)

Missing security header: Referrer-Policy

CONFIRMED

port 443/tcp

URL	Evidence
https://sumasys.analyticalcenter.id/import-price-item	Response headers do not include the Referrer-Policy HTTP security header as well as the <meta> tag with name 'referrer' is not present in the response. Request / Response

Details

Risk description:

The risk is that if a user visits a web page (e.g. "http://example.com/pricing/") and clicks on a link from that page going to e.g. "https://www.google.com", the browser will send to Google the full originating URL in the `Referer` header, assuming the Referrer-Policy header is not set. The originating URL could be considered sensitive information and it could be used for user tracking.

Recommendation:

The Referrer-Policy header should be configured on the server side to avoid user tracking and inadvertent information leakage. The value `no-referrer` of this header instructs the browser to omit the Referer header entirely.

References:

https://developer.mozilla.org/en-US/docs/Web/Security/Referer_header:_privacy_and_security_concerns

Classification:

CWE : [CWE-693](#)

OWASP Top 10 - 2017 : [A6 - Security Misconfiguration](#)

OWASP Top 10 - 2021 : [A5 - Security Misconfiguration](#)

OWASP Top 10 - 2025 : [A02 - Security Misconfiguration](#)

 Missing security header: X-Content-Type-Options**CONFIRMED**

port 443/tcp

URL	Evidence
https://sumasys.analyticalcenter.id/import-price-item	Response headers do not include the X-Content-Type-Options HTTP security header Request / Response

 Details**Risk description:**

The risk is that lack of this header could make possible attacks such as Cross-Site Scripting or phishing in Internet Explorer browsers.

Recommendation:

We recommend setting the X-Content-Type-Options header such as `X-Content-Type-Options: nosniff`.

References:

<https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Content-Type-Options>

Classification:

CWE : [CWE-693](#)

OWASP Top 10 - 2017 : [A6 - Security Misconfiguration](#)

OWASP Top 10 - 2021 : [A5 - Security Misconfiguration](#)

OWASP Top 10 - 2025 : [A02 - Security Misconfiguration](#)

 Missing security header: Content-Security-Policy**CONFIRMED**

port 443/tcp

URL	Evidence
https://sumasys.analyticalcenter.id/import-price-item	Response does not include the HTTP Content-Security-Policy security header or meta tag Request / Response

 Details**Risk description:**

The risk is that if the target application is vulnerable to XSS, lack of this header makes it easily exploitable by attackers.

Recommendation:

Configure the Content-Security-Header to be sent with each HTTP response in order to apply the specific policies needed by the application.

References:

https://cheatsheetseries.owasp.org/cheatsheets/Content_Security_Policy_Cheat_Sheet.html

<https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Content-Security-Policy>

Classification:

CWE : [CWE-1021](#)

OWASP Top 10 - 2017 : [A6 - Security Misconfiguration](#)

OWASP Top 10 - 2021 : [A5 - Security Misconfiguration](#)

OWASP Top 10 - 2025 : [A02 - Security Misconfiguration](#)

 Internal Server Error Found**CONFIRMED**

port 443/tcp

URL	Method	Parameters	Evidence
-----	--------	------------	----------

https://sumasys.analyticalcenter.id/inline-staging	GET	Query: search=1d3d2d231d2dd4 Headers: User-Agent=Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/108.0.0.0 Safari/537.36 Cookies: XSRF- TOKEN=eyJpdil6lkgl1MzNYTCTYcERIOE40d2xIMzlt2c9PSIsInZhbHVlIjoVDFoeko5OFFOWFp4UDJoZGtIUvhaRTIPZlFUanB...	Response has an internal server error status code: 500
---	-----	--	--

Details

Risk description:

The risk exists that attackers could utilize information revealed in Internal Server Error messages to mount more targeted and effective attacks. Detailed error messages could, for example, expose a path traversal weakness (CWE-22) or other exploitable system vulnerabilities.

Recommendation:

Ensure that error messages only contain minimal details that are useful to the intended audience, and nobody else. The messages need to strike the balance between being too cryptic and not being cryptic enough. They should not necessarily reveal the methods that were used to determine the error. Such detailed information can be used to refine the original attack to increase the chances of success. If errors must be tracked in some detail, capture them in log messages - but consider what could occur if the log messages can be viewed by attackers. Avoid recording highly sensitive information such as passwords in any form. Avoid inconsistent messaging that might accidentally tip off an attacker about internal state, such as whether a username is valid or not.

Classification:

CWE : [CWE-209](#)

OWASP Top 10 - 2017 : [A6 - Security Misconfiguration](#)

OWASP Top 10 - 2021 : [A5 - Security Misconfiguration](#)

OWASP Top 10 - 2025 : [A10 - Mishandling of Exceptional Conditions](#)

Screenshot:

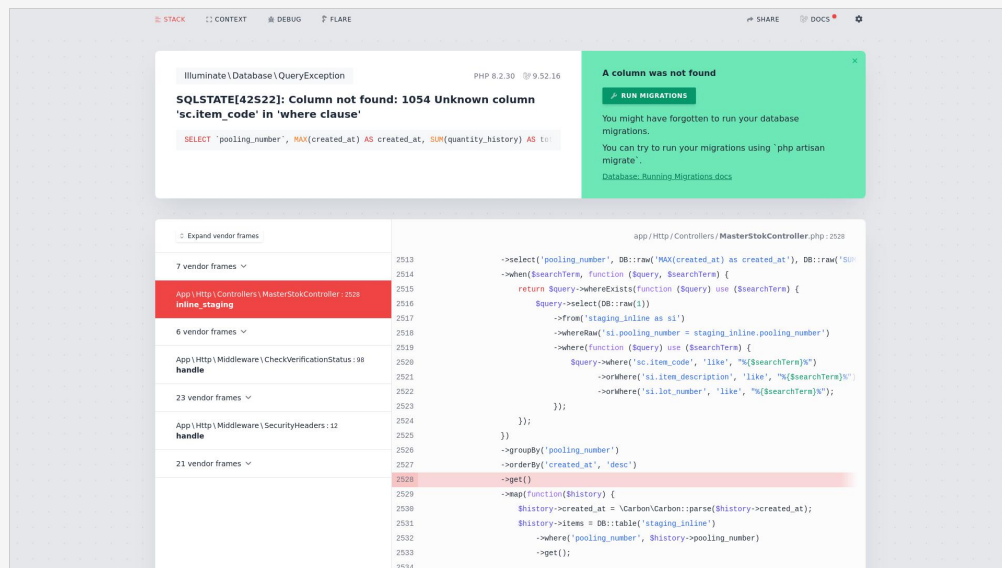


Figure 2. Internal Error

Error message containing sensitive information

UNCONFIRMED

port 443/tcp

URL	Method	Parameters	Evidence
https://sumasys.analyticalcenter.id/import-price-item	GET	Headers: User-Agent=Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/108.0.0.0 Safari/537.36 Cookies: XSRF- TOKEN=eyJpdil6lINEK3IQVkdQMHBIVVRvb1Z5WGJHaHc9PSIsInZhbHVlIjoR1F2ckczdDN1K21SZ3k1NVRqd1JwOHJ2ZFZBM2tMeCtub3gwbXhJbHAYWXpvrHZVZVZW90a2d5M0J...	Error message error report found in: <pre>block -mt-1 ml-1 mr-px}},"flare"}," , the laravel error reporting service."))}}function dv(e){var t=e.erroroccur</pre>

▼ Details

Risk description:

The risk is that an attacker may use the contents of error messages to help launch another, more focused attack. For example, an attempt to exploit a path traversal weakness (CWE-22) might yield the full pathname of the installed application.

Recommendation:

It is recommended treating all exceptions of the application flow. Ensure that error messages only contain minimal details.

Classification:

CWE : [CWE-209](#)

OWASP Top 10 - 2017 : A6 - Security Misconfiguration

OWASP Top 10 - 2021: [A4 - Insecure Design](#)

OWASP Top 10 - 2025 : A10 - Mishandling of Exceptional Conditions

 Suspicious Comment

port 443/tcp

UNCONFIRMED

URL	Method	Parameters	Evidence
https://sumasys.analyticalcenter.id/admin/change-role/40	GET	Headers: User-Agent=Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/108.0.0.0 Safari/537.36 Cookies: XSRF-TOKEN=eyJpdil6lk9MzNYTCTYcERIOE40d2xIMzltId2c9PSIsInZhbHVlIjoieVDFoeko5OFFOWFp4UDJoZGtIUvhaRTIPZlFUanBlc0hUeUdZYWE3aEUrUmxzZWw4Tm1nRVhuOFI...	Suspicious comment admin found in: eption: The GET method is not supported for route admin/change-role/40. Supported methods: POST. in file

▼ Details

Risk description:

The risk exists that attackers could analyze these comments to identify vulnerabilities or weaknesses in the application. While comments themselves do not directly lead to security breaches, they may guide attackers to focus their efforts on specific parts of the application, potentially uncovering and exploiting vulnerabilities.

Recommendation:

Remove comments that suggest the presence of bugs, incomplete functionality, or weaknesses, before deploying the application.

Classification:

CWE : [CWE-546](#)

OWASP Top 10 - 2017 : A6 - Security Misconfiguration

OWASP Top 10 - 2021: **A4 - Insecure Design**

OWASP Top 10 - 2025 : [A06 - Insecure Design](#)

Server Information disclosure

port 443/tcp

UNCONFIRMED

URL	Page Title	Page Size	Summary
https://sumasys.analyticalcenter.id/.htaccess		768 B	Contains configuration and/or authorization information
https://sumasys.analyticalcenter.id/.gitignore		254 B	.gitignore file found. It is possible to grasp the directory structure.

▼ Details

Risk description:

The risk is that an attacker could use these files to find information about the backend application, server software and their specific versions. This information could be further used to mount targeted attacks against the server.

Recommendation:

We recommend you to remove these scripts if they are not needed for business purposes.

References:

<http://projects.webappsec.org/w/page/13246936/Information%20Leakage>

Classification:

🚩 Enumerable Parameter

port 443/tcp

UNCONFIRMED

URL	Method	Vulnerable Parameter	Evidence
https://sumasys.analyticalcenter.id/item-ed	GET	id (Query Parameter)	The id query parameter appears to contain an enumerable numeric part. We modified its initial value 1123123 to 1123122 and the two responses were 99% similar. The parameter may introduce an Insecure Direct Object Reference (IDOR) vulnerability. Request / Response
https://sumasys.analyticalcenter.id/item-ed	GET	id (Query Parameter)	The id query parameter appears to contain an enumerable numeric part. We modified its initial value 1123123 to 1123122 and the two responses were 99% similar. The parameter may introduce an Insecure Direct Object Reference (IDOR) vulnerability. Request / Response
https://sumasys.analyticalcenter.id/item-ed	GET	id (Query Parameter)	The id query parameter appears to contain an enumerable numeric part. We modified its initial value 1123123 to 1123122 and the two responses were 99% similar. The parameter may introduce an Insecure Direct Object Reference (IDOR) vulnerability. Request / Response

▼ Details

Risk description:

The vulnerability allows attackers to brute-force parameter values to uncover and access unauthorized resources and functionalities.

Recommendation:

Ensure that parameter values would not reveal sensitive information and that the application properly checks the user's authorization to access the resource. Also, the resource IDs should not be predictable.

References:

[Testing for Insecure Direct Object References](#)

Classification:

CWE : [CWE-284](#)
OWASP Top 10 - 2017 : [A5 - Broken Access Control](#)
OWASP Top 10 - 2021 : [A1 - Broken Access Control](#)
OWASP Top 10 - 2025 : [A01 - Broken Access Control](#)

Scan coverage information

List of tests performed (75)

Port 443

- ✓ Scanned for CORS misconfiguration
- ✓ Scanned for Insecure Direct Object Reference
- ✓ Scanned for error messages
- ✓ Scanned for sensitive data
- ✓ Scanned for internal error code
- ✓ Scanned for missing HTTP header - Content Security Policy
- ✓ Scanned for missing HTTP header - Referrer
- ✓ Scanned for missing HTTP header - Strict-Transport-Security
- ✓ Scanned for missing HTTP header - X-Content-Type-Options
- ✓ Scanned for robots.txt file
- ✓ Scanned for information disclosure
- ✓ Scanned for website technologies
- ✓ Scanned for code comments
- ✓ Scanned for unsafe HTTP header Content Security Policy

- ✓ Scanned for API endpoints
- ✓ Obtained an authenticated session
- ✓ Scanned for file upload
- ✓ Scanned for enabled HTTP OPTIONS method
- ✓ Scanned for login interfaces
- ✓ Scanned for misconfigurations
- ✓ Scanned for Path Disclosure
- ✓ Scanned for absence of the security.txt file
- ✓ Performed web-crawling on target
- ✓ Scanned for version-based vulnerabilities of server-side software
- ✓ Scanned for client access policies
- ✓ Scanned for use of untrusted certificates
- ✓ Scanned for enabled HTTP debug methods
- ✓ Scanned for administration consoles
- ✓ Scanned for software identification
- ✓ Scanned for sensitive files
- ✓ Scanned for interesting files
- ✓ Scanned for GraphQL endpoints
- ✓ Performed fuzzing for OpenAPI files
- ✓ Scanned for secure communication
- ✓ Scanned for directory listing
- ✓ Scanned for passwords submitted unencrypted
- ✓ Scanned for Cross-Site Scripting
- ✓ Scanned for SQL Injection
- ✓ Scanned for Local File Inclusion
- ✓ Scanned for OS Command Injection
- ✓ Scanned for debug messages
- ✓ Scanned for XML External Entity Injection
- ✓ Scanned for passwords submitted in URLs
- ✓ Scanned for JWT weaknesses
- ✓ Scanned for domain too loose set for cookies
- ✓ Scanned for mixed content between HTTP and HTTPS
- ✓ Scanned for cross domain file inclusion
- ✓ Scanned for HttpOnly flag of cookie
- ✓ Scanned for Secure flag of cookie
- ✓ Scanned for secure password submission
- ✓ Scanned for Server Side Request Forgery
- ✓ Scanned for Open Redirect
- ✓ Scanned for PHP Code Injection
- ✓ Scanned for JavaScript Code Injection
- ✓ Scanned for Broken Authentication
- ✓ Scanned for Ruby Code Injection
- ✓ Scanned for Python Code Injection
- ✓ Scanned for Perl Code Injection
- ✓ Scanned for Remote Code Execution through Log4j
- ✓ Scanned for Server Side Template Injection
- ✓ Scanned for Remote Code Execution through VIEWSTATE
- ✓ Scanned for Prototype Pollution
- ✓ Scanned for Exposed Backup Files
- ✓ Scanned for Request URL Override
- ✓ Scanned for CSRF
- ✓ Scanned for NoSQL Injection
- ✓ Scanned for Insecure Deserialization
- ✓ Scanned for Session Fixation
- ✓ Scanned for OpenAPI files
- ✓ Scanned for SQL statement in request parameter
- ✓ Scanned for password returned in later response
- ✓ Scanned for Session Token in URL
- ✓ Scanned for Response Header Injection
- ✓ Scanned for emails
- ✓ Scanned for missing HTTP header - Rate Limit

Scan parameters

Target:	https://sumasys.analyticalcenter.id/
Scan type:	Deep_scan_default
Authentication:	True
fingerprint:	True
software_vulnerabilities:	True
check_robots:	True
outdated_js:	True
untrusted_certificates:	True
client_access_policies:	True
http_debug_methods:	True
security_txt:	True
cors_misconfiguration:	True
resource_discovery:	True
sensitive_files:	True
admin_consoles:	True
interesting_files:	True
server_info_disc:	True
server_software:	True
misconfigurations:	True
graphql_endpoint:	True
fuzz_openapi_locations:	True
ai_endpoint_discovery:	True
approach:	Auto
depth:	10
max_time:	3600
requests_per_second:	100
xss:	True
sqli:	True
lfi:	True
oscmdi:	True
ssrf:	True
open_redirect:	True
broken_authentication:	True
php_code_injection:	True
js_code_injection:	True
ruby_code_injection:	True
python_code_injection:	True
perl_code_injection:	True
log4j_rce:	True
ssti:	True
xxe:	True
viewstate_rce:	True
prototype_pollution:	True
backup_files:	True
request_url_override:	True
http_request_smuggling:	True
csrf:	True
insecure_deserialization:	True
nosqli:	True
session_fixation:	True
idor:	True
jwt:	True
response_header_injection:	True
security_headers:	True
cookie_security:	True
directory_listing:	True
secure_communication:	True
weak_password_submission:	True
error_debug_messages:	True
password_cleartext:	True
cross_domain_source:	True
mixed_content:	True
sensitive_data:	True
login_interfaces:	True
file_upload:	True
openapi_documents:	True
path_disclosure:	True
sql_statement_in_request:	True
password_in_response:	True
session_token_in_url:	True
api_endpoint:	True

Scan stats

Unique Injection Points Detected: 312

URLs spidered:	401
Total number of HTTP requests:	41052
Average time until a response was received:	2428ms
Total number of HTTP request errors:	3011