

SecureThread OPS

Securing the Digital Future

SECURITY ASSESSMENT REPORT

juice-shop

Repository: dummyhshz/juice-shop

Branch/Commit: master / HEAD

Scan Date: 22 December 2025 12:21

Stats: 863 files / 69,040 LoC

Report ID: ST-180-20251231-1301

SECURITY POSTURE SUMMARY

Score	Grade	Findings	Critical
0	F	1228	269

269 Critical

663 High

241 Medium

55 Low

Report Navigation Guide

This report is organized into the following main sections:

- Section 1:** **Executive Summary** - High-level overview for management and stakeholders
- Section 2:** **Compliance Mapping** - Regulatory and framework alignment (OWASP, PCI DSS, GDPR)
- Section 3:** **Detailed Findings** - Technical vulnerability details for development teams
- Section 4:** **Recommendations** - Prioritized remediation roadmap and best practices
- Appendices:** **Reference Materials** - Methodology, glossary, CWE reference, and resources

Quick Start:

- Executives: Read Section 1 (Executive Summary)
- Developers: Focus on Section 3 (Detailed Findings) and Section 4 (Recommendations)
- Compliance: Review Section 2 (Compliance Mapping)
- Security Team: Review all sections

Table of Contents

List of Figures & Visualizations

Severity Rating Legend

This report uses a standardized severity rating system based on CVSS (Common Vulnerability Scoring System) and impact assessment. Use this legend to quickly understand the urgency of each finding.

Icon	Severity	Description	Action Timeline
C	CRITICAL CVSS: 9.0-10.0	Immediate threat of system compromise, data breach, or complete service disruption. Exploitable remotely without authentication.	0-72 hours Immediate action required
H	HIGH CVSS: 7.0-8.9	Significant security risk allowing unauthorized access, data exposure, or privilege escalation. May require some user interaction.	7-14 days Urgent attention needed
M	MEDIUM CVSS: 4.0-6.9	Moderate security weakness that could lead to information disclosure or limited access. Typically requires specific conditions to exploit.	30-60 days Address in next sprint
L	LOW CVSS: 0.1-3.9	Minor security concern with minimal impact. Difficult to exploit or requires extensive preconditions.	60-90 days Maintenance priority
I	INFO CVSS: 0.0	Informational finding or security best practice recommendation without direct exploitability.	Ongoing Best practice improvement

CVSS Score Calculation

CVSS (Common Vulnerability Scoring System) scores are calculated based on:

Factor	Description
Attack Vector	How the vulnerability can be exploited (Network, Adjacent, Local, Physical)
Attack Complexity	Conditions beyond attacker's control (Low, High)
Privileges Required	Level of access needed (None, Low, High)
User Interaction	Whether user action is required (None, Required)
Confidentiality Impact	Impact on data confidentiality (None, Low, High)
Integrity Impact	Impact on data integrity (None, Low, High)
Availability Impact	Impact on system availability (None, Low, High)

Note: For complete CVSS documentation, visit: <https://www.first.org/cvss/>

Common Abbreviations & Acronyms

Quick reference for technical terms used throughout this report:

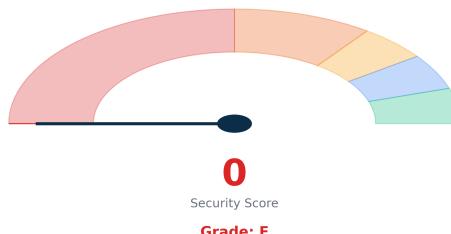
API	Application Programming Interface	NIST	National Institute of Standards and Technology
CORS	Cross-Origin Resource Sharing	OWASP	Open Web Application Security Project
CSRF	Cross-Site Request Forgery	PCI DSS	Payment Card Industry Data Security Standard
CVE	Common Vulnerabilities and Exposures	PII	Personally Identifiable Information
CVSS	Common Vulnerability Scoring System	REST	Representational State Transfer
CWE	Common Weakness Enumeration	RCE	Remote Code Execution
DAST	Dynamic Application Security Testing	SANS	SysAdmin, Audit, Network, Security
DoS	Denial of Service	SAST	Static Application Security Testing
GDPR	General Data Protection Regulation	SCA	Software Composition Analysis
HIPAA	Health Insurance Portability and Accountability Act	SDLC	Software Development Lifecycle
HTTPS	Hypertext Transfer Protocol Secure	SOC	Security Operations Center
IAM	Identity and Access Management	SQL	Structured Query Language
IaC	Infrastructure as Code	SSRF	Server-Side Request Forgery
JSON	JavaScript Object Notation	SSL/TLS	Secure Sockets Layer / Transport Layer Security
JWT	JSON Web Token	XSS	Cross-Site Scripting
LOC	Lines of Code	XXE	XML External Entity
MFA	Multi-Factor Authentication		

Need More Detail? A comprehensive glossary is available in Appendix D (page ??).

Executive Summary

Assessment Overview

SecureThread OPS has completed a comprehensive automated security assessment of the **juice-shop** repository using Static Application Security Testing (SAST) techniques combined with AI-powered analysis.



Key Statistics:

Scan Date:	22 December 2025 12:21
Branch/Commit:	master / HEAD
Files Analyzed:	863
Lines of Code:	69,040
Total Findings:	1228
Vulnerability Density:	17.79 per 1k LOC
Auto-Fixable:	10 (1%)

Severity Distribution

The following table summarizes the distribution of vulnerabilities by severity level:

Severity	Count	%	Risk Assessment
CRITICAL	269	21.9%	Immediate remediation required - system compromise risk
HIGH	663	54.0%	Urgent attention needed - significant security risk
MEDIUM	241	19.6%	Address in next sprint - moderate risk
LOW	55	4.5%	Address during maintenance - minimal risk

Business Impact Analysis

This section quantifies the financial and operational impact of identified vulnerabilities:

Metric	Value	Impact Level
Remediation Cost Developer hours to fix	\$1,623,700.00	HIGH
Potential Breach Cost If vulnerabilities exploited	\$24,162,000.00	CRITICAL
Total Estimated Effort Time to complete remediation	7984 hours (998.0 days)	SIGNIFICANT
Total Financial Risk Combined cost exposure	\$25,785,700.00	SEVERE

Executive Action Required

The identified vulnerabilities represent a **significant financial risk** exceeding **\$25,785,700.00**. Immediate executive attention and resource allocation is recommended to address critical security gaps and prevent potential data breaches that could result in regulatory fines, legal liability, and reputational damage.

Security Analysis Visualizations

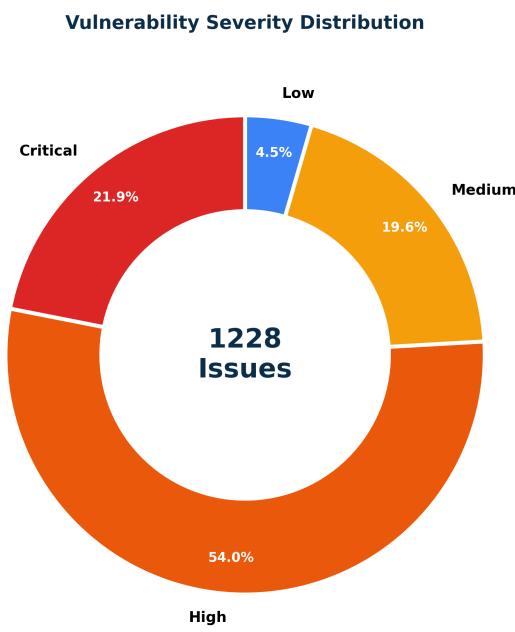


Figure 1: Vulnerability Severity Distribution

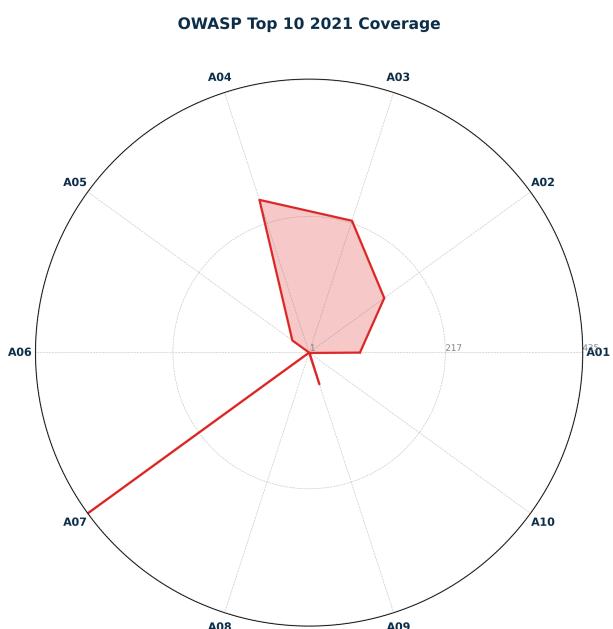


Figure 2: OWASP Top 10 2021 Coverage

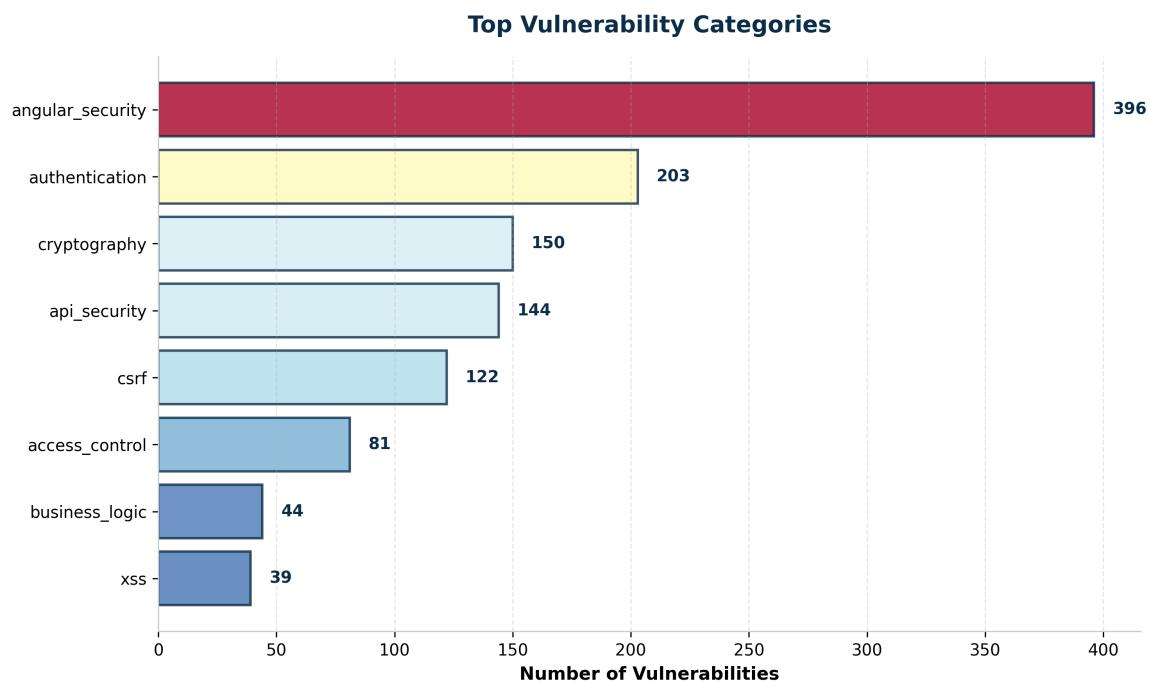


Figure 3: Top Vulnerability Categories Detected

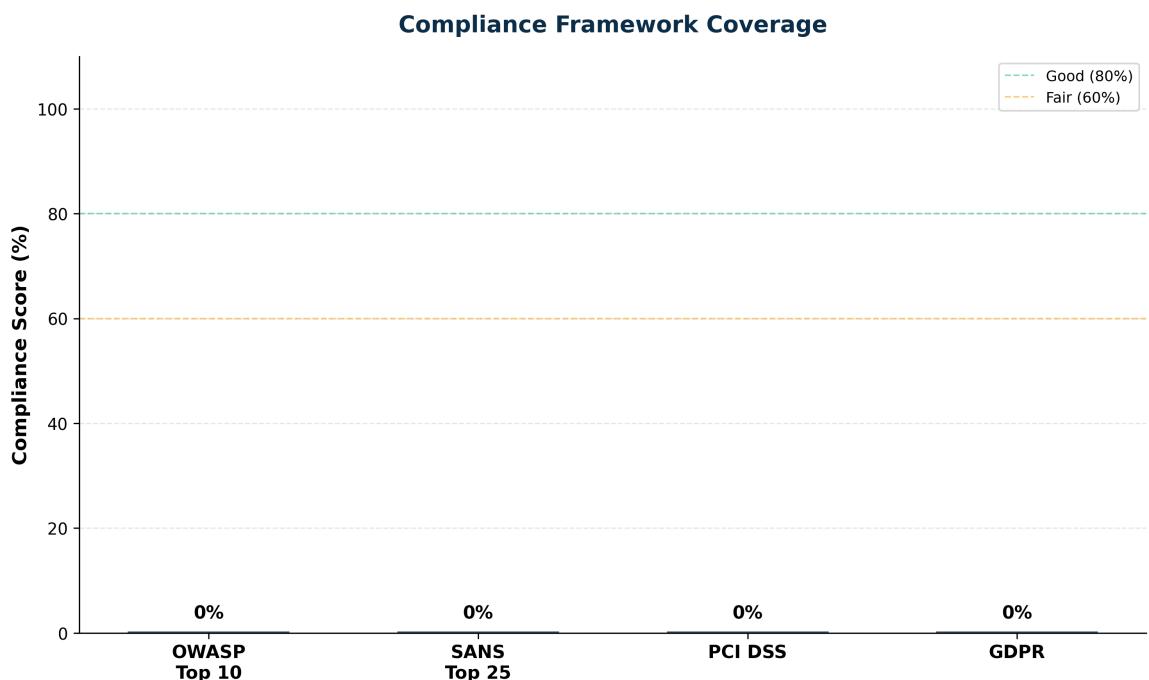


Figure 4: Compliance Framework Coverage Assessment

Critical & High Priority Findings

The following table highlights the most critical security findings requiring immediate attention:

ID	Severity	Title	Location	CVSS
#297	HIGH	CSRF - Missing Origin Validation	data/static/codefixes/changeProductCh	7.5
#298	HIGH	Vertical Access Control - Missing Role...	data/static/codefixes/changeProductCh	7.5
#299	HIGH	Vertical Access Control - Missing Role...	data/static/codefixes/changeProductCh	7.5
#300	HIGH	Vertical Access Control - Missing Role...	data/static/codefixes/changeProductCh	7.5
#301	HIGH	Vertical Access Control - Missing Role...	data/static/codefixes/changeProductCh	7.5
#302	HIGH	Vertical Access Control - Missing Role...	data/static/codefixes/changeProductCh	7.5
#303	HIGH	Vertical Access Control - Missing Role...	data/static/codefixes/changeProductCh	7.5
#304	HIGH	Vertical Access Control - Missing Role...	data/static/codefixes/changeProductCh	7.5
#305	HIGH	CSRF - Missing Origin Validation	data/static/codefixes/changeProductCh	7.5
#306	HIGH	CSRF - Missing Origin Validation	data/static/codefixes/changeProductCh	7.5

Note: 922 additional critical/high severity findings are documented in the Detailed Findings section (Section 4).

Key Recommendations

Based on this assessment, we recommend the following prioritized actions:

1. **#1: Immediate Critical Remediation**

Address all 269 critical vulnerabilities within 7 days to prevent system compromise

2. **#2: High-Severity Sprint Planning**

Allocate development resources to remediate 663 high-severity issues across 2-3 sprints

3. **#3: Compliance Alignment**

Address 16 compliance violations (OWASP, PCI DSS, GDPR) to meet regulatory requirements

4. **#4: Code Quality Improvement**

Vulnerability density (17.79/1k LOC) is elevated. Implement secure coding training and SAST in CI/CD

5. #5: Leverage Auto-Fix Suggestions

10 vulnerabilities have automated fix suggestions - implement these quick wins first

Detailed remediation guidance, sprint planning, and secure coding best practices are provided in Section 5: Strategic Recommendations & Remediation Roadmap.

Industry Benchmark Comparison

Security Posture Rating: Your security score of **0.0/100** is **below industry standards and requires significant improvement.**

Your Score: 0.0/100 (Grade: F)

Industry Average: 72/100 (Grade: C+)

Your Density: 17.79 vulnerabilities per 1k LOC

Industry Average: 3.5 vulnerabilities per 1k LOC

0.1 Compliance \& Regulatory Mapping

This section maps identified vulnerabilities to industry-standard security frameworks and regulatory requirements, helping organizations understand their compliance posture.

0.1.1 OWASP Top 10 2021 Coverage

The **OWASP Top 10** represents the most critical security risks to web applications. This assessment identifies which OWASP categories are present in your codebase.

Category	Description	Findings	Risk
A01	Failures related to enforcing access policies	81	Critical
A02	Weak or missing encryption, exposed sensitive data	148	Critical
A03	SQL, NoSQL, OS command injection attacks	221	Critical
A04	Missing or ineffective security design patterns	256	Critical
A05	Insecure default configurations or settings	33	Critical
A06	Use of outdated or vulnerable libraries	0	None
A07	Authentication and session management flaws	435	Critical
A08	Insecure CI/CD, unsigned code or data	0	None
A09	Insufficient logging and monitoring	53	Critical
A10	Server-Side Request Forgery vulnerabilities	1	Low

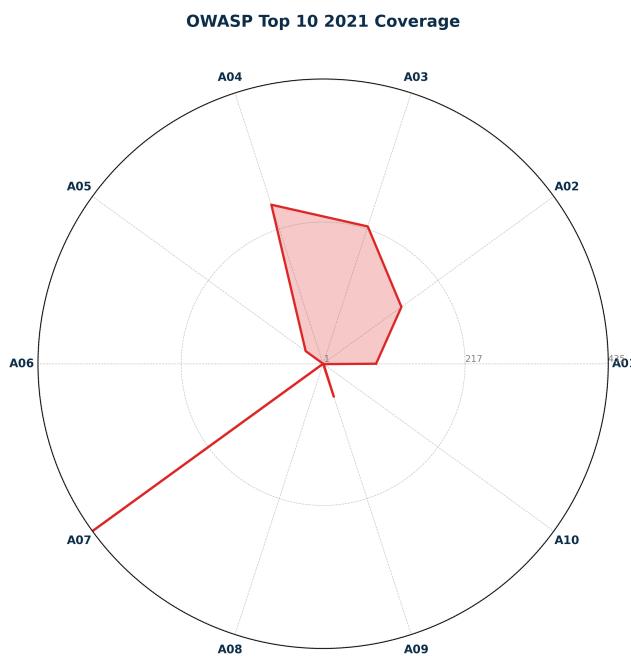


Figure 5: OWASP Top 10 Coverage Visualization

0.1.2 SANS Top 25 Most Dangerous Software Weaknesses

The **SANS Top 25** list identifies the most widespread and critical software security weaknesses based on Common Weakness Enumeration (CWE) identifiers.

CWE ID	Weakness Name	Findings	Rank
CWE-306	Missing Authentication	387	#10
CWE-352	Cross-Site Request Forgery (CSRF)	122	#8
CWE-79	Cross-site Scripting (XSS)	44	#1
CWE-20	Improper Input Validation	4	#3
CWE-434	Unrestricted File Upload	2	#9
CWE-89	SQL Injection	1	#2

0.1.3 PCI DSS v4.0 Requirements

The **Payment Card Industry Data Security Standard (PCI DSS)** is mandatory for organizations that handle credit card information. Requirement 6 focuses on secure software development.

PCI DSS Requirement	Findings	Compliance Status
Req 6.2: Secure Coding	413	Non-Compliant
Req 6.5: Common Vulnerabilities	45	Non-Compliant
Req 8: Strong Access Control	609	Non-Compliant
Req 4: Encryption in Transit	150	Non-Compliant
Req 3: Protect Stored Data	11	Non-Compliant

PCI DSS Compliance Alert

Action Required: This application has 1228 findings that may impact PCI DSS compliance. Organizations processing payment card data must remediate these issues to maintain certification. Consult with your QSA (Qualified Security Assessor).

0.1.4 GDPR Article 32: Security of Processing

GDPR Article 32 requires appropriate technical and organizational measures to ensure a level of security appropriate to the risk when processing personal data.

GDPR Risk Assessment

High-Impact Issues: 91

Medium-Impact Issues: 0

Overall Risk Level: High

GDPR Compliance Actions:

- Conduct a Data Protection Impact Assessment (DPIA) for high-risk vulnerabilities
- Notify your Data Protection Officer (DPO) of these findings
- Document remediation efforts in your Records of Processing Activities (ROPA)
- Consider whether a breach notification (Article 33) may be required if exploited
- Implement "Privacy by Design" principles (Article 25)

0.1.5 Compliance Summary Dashboard

The following dashboard provides an at-a-glance view of your compliance posture across multiple security frameworks and regulatory standards.

Framework	Total Issues	Critical	Status	Priority
OWASP Top 10	1228	269	At Risk	HIGH
SANS Top 25	560	3	At Risk	HIGH
PCI DSS v4.0	1228	5	Non-Compliant	CRITICAL
GDPR Article 32	91	91	High	HIGH
Overall Compliance Grade:				D

0.2 Detailed Vulnerability Findings

This section provides comprehensive technical details for each identified vulnerability, including code snippets, remediation guidance, and compliance mappings. Findings are organized by severity level for prioritized remediation.

Reading Guide:

- **CWE/OWASP:** Industry-standard vulnerability classifications
- **CVSS Score:** Risk rating from 0.0 (low) to 10.0 (critical)
- **Location:** Exact file path and line number(s)
- **Code Snippet:** Vulnerable code section
- **Remediation:** Step-by-step fix guidance
- **Fix Suggestion:** Automated or recommended code changes

0.2.1 Critical Severity Issues (269 found)

IMMEDIATE ACTION REQUIRED

Critical vulnerabilities represent **immediate threats** that could lead to:

- Complete system compromise
- Unauthorized data access or exfiltration
- Remote code execution
- Privilege escalation to administrator level

Recommended Timeline: Fix within 24-72 hours

#960: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	data/static/codefixes/loginBenderChallenge_3.ts:17	
Line(s):	17	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings

Vulnerable Code

```
models.sequelize.query('SELECT * FROM Users WHERE email = :mail AND password =\n' +\n' ${security.hash(req.body.password || "")}' AND deletedAt IS NULL',
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

```
{"json\n{\n"explanation": "The code contains a CRITICAL vulnerability: hardcoded database credentials at\nline 17. While the exact hardcoded password isn't visible in the provided snippet, the detection\nindicates credentials are embedded directly in the source code. This is dangerous because: 1)\nCredentials become exposed in version control systems, 2) Anyone with access to the codebase\ncan extract them, 3) Changing passwords requires code changes and redeployment, 4) Different\nenvironments (d\n\n
```

Suggested Fix (Copy & Apply)

```
ev/staging/prod) cannot use different credentials without code modifications.\nAdditionally, the code shows SQL query construction with string interpolation\n('`${security.hash(...)}''), which could lead to SQL injection if not properly handled\nby the ORM/query builder, though Sequelize's replacements help mitigate this risk.",
```

References & Further Reading:

<https://owasp.org/Top10/A07/>

#961: Horizontal Access Control - Tenant Isolation Bypass

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-566 OWASP: A01:2021 - Broken Access Control
Category:	access_control Confidence: MEDIUM
Location:	data/static/codefixes/loginBenderChallenge_3.ts:17
Line(s):	17
Exploitability:	Unknown Impact: Unknown

Description

Detects multi-tenant isolation bypass **Vulnerable Code**

```
models.sequelize.query('SELECT * FROM Users WHERE email = :mail AND password =\n' +\n' ${security.hash(req.body.password || "")} AND deletedAt IS NULL',
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the access_control issue in data/static/codefixes/loginBenderChallenge_3.ts **References**

& Further Reading:

<https://owasp.org/Top10/A01/>

#962: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798 OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication Confidence: MEDIUM
Location:	data/static/codefixes/loginBenderChallenge_4.ts:17
Line(s):	17
Exploitability:	Unknown Impact: Unknown

Description

Detects hardcoded database passwords in connection strings

Vulnerable Code

```
models.sequelize.query('SELECT * FROM Users WHERE email = '${req.body.email || ''}' AND password = '${security.hash(req.body.password || '')}' AND deletedAt IS NULL', { model: models.User, plain: false })
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in data/static/codefixes/loginBenderChallenge_4.ts

& Further Reading:

<https://owasp.org/Top10/A07/>

#963: Horizontal Access Control - Tenant Isolation Bypass

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-566	OWASP: A01:2021 - Broken Access Control
Category:	access_control	Confidence: MEDIUM
Location:	data/static/codefixes/loginBenderChallenge_4.ts:17	
Line(s):	17	
Exploitability:	Unknown	Impact: Unknown

Description

Detects multi-tenant isolation bypass

Vulnerable Code

```
models.sequelize.query('SELECT * FROM Users WHERE email = '${req.body.email || ''}' AND password = '${security.hash(req.body.password || '')}' AND deletedAt IS NULL', { model: models.User, plain: false })
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the access_control issue in data/static/codefixes/loginBenderChallenge_4.ts

& Further Reading:

<https://owasp.org/Top10/A01/>

#964: Hardcoded Credentials - Database Password

Severity: CRITICAL (CVSS: 9.5/10.0)

CWE ID: CWE-798 **OWASP:** A07:2021 - Identification and Authentication Failures

Category: authentication **Confidence:** MEDIUM

Location: data/static/codefixes/loginJimChallenge_2.ts:17

Line(s): 17

Exploitability: Unknown **Impact:** Unknown

Description

Detects hardcoded database passwords in connection strings

Vulnerable Code

```
models.sequelize.query('SELECT * FROM Users WHERE email = '${req.body.email || ''} AND password = '${security.hash(req.body.password || '')}' AND deletedAt IS NULL', { model: models.User, plain: false })
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation GuidanceReview and fix the authentication issue in data/static/codefixes/loginJimChallenge_2.ts **References****& Further Reading:**<https://owasp.org/Top10/A07/>**#965: Horizontal Access Control - Tenant Isolation Bypass**

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-566	OWASP: A01:2021 - Broken Access Control
Category:	access_control	Confidence: MEDIUM
Location:	data/static/codefixes/loginJimChallenge_2.ts:17	
Line(s):	17	
Exploitability:	Unknown	Impact: Unknown

DescriptionDetects multi-tenant isolation bypass **Vulnerable Code**

```
models.sequelize.query('SELECT * FROM Users WHERE email = '${req.body.email || ''} AND password = '${security.hash(req.body.password || '')}' AND deletedAt IS NULL', { model: models.User, plain: false })
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation GuidanceReview and fix the access_control issue in data/static/codefixes/loginJimChallenge_2.ts **References****& Further Reading:**<https://owasp.org/Top10/A01/>

#966: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	data/static/codefixes/loginJimChallenge_4.ts:20	
Line(s):	20	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings

Vulnerable Code

```
models.sequelize.query('SELECT * FROM Users WHERE email = '${req.body.email || ''} AND password = '${security.hash(req.body.password || '')}' AND deletedAt IS NULL', { model: models.User, plain: true })
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in data/static/codefixes/loginJimChallenge_4.ts

& Further Reading:

<https://owasp.org/Top10/A07/>

#967: Horizontal Access Control - Tenant Isolation Bypass

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-566
Category:	access_control
Location:	data/static/codefixes/loginJimChallenge_4.ts:20
Line(s):	20
Exploitability:	Unknown
Impact:	Unknown

Description

Detects multi-tenant isolation bypass **Vulnerable Code**

```
models.sequelize.query('SELECT * FROM Users WHERE email = '${req.body.email || ''} AND password = '${security.hash(req.body.password || '')}' AND deletedAt IS NULL', { model: models.User, plain: true })
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the access_control issue in data/static/codefixes/loginJimChallenge_4.ts **References**

& Further Reading:

<https://owasp.org/Top10/A01/>

#968: Angular - bypassSecurityTrust XSS

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-79 OWASP: A03:2021 - Injection
Category:	angular_security Confidence: MEDIUM
Location:	frontend/src/app/administration/administration.component.ts:57
Line(s):	57
Exploitability:	Unknown Impact: Unknown

Description

Detects bypassSecurityTrust methods with user input **Vulnerable Code**

```
user.email = this.sanitizer.bypassSecurityTrustHtml('<span class="${this.doesUserHaveAnActiveSession(user)} ? \'confirmation\' : \'error\'>"${user.email}</span>')
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the angular_security issue in frontend/src/app/administration/administration.component.ts **References**

& Further Reading:

<https://owasp.org/Top10/A03/>

#969: XSS - Angular bypassSecurityTrustHtml

Severity:	CRITICAL	(CVSS: 9.5/10.0)
CWE ID:	CWE-79	OWASP: A03:2021 - Injection
Category:	xss	Confidence: MEDIUM
Location:	frontend/src/app/administration/administration.component.ts:57	
Line(s):	57	
Exploitability:	Unknown	Impact: Unknown

Description

Detects Angular DomSanitizer security bypass **Vulnerable Code**

```
user.email = this.sanitizer.bypassSecurityTrustHtml('<span class="${this.doesUserHaveAnActiveSession(user)} ? \'confirmation\' : \'error\'>"${user.email}</span>')
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the xss issue in frontend/src/app/administration/administration.component.ts **References**

& Further Reading:

<https://owasp.org/Top10/A03/>

#970: XSS - Angular bypassSecurityTrustHtml

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-79 OWASP: A03:2021 - Injection
Category:	xss Confidence: MEDIUM
Location:	frontend/src/app/administration/administration.component.ts:72
Line(s):	72
Exploitability:	Unknown
	Impact: Unknown

Description

Detects Angular DomSanitizer security bypass **Vulnerable Code**

```
feedback.comment = this.sanitizer.bypassSecurityTrustHtml(feedback.comment)
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the xss issue in frontend/src/app/administration/administration.component.ts **References**

& Further Reading:

<https://owasp.org/Top10/A03/>

#971: XSS - Angular bypassSecurityTrustHtml

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-79 OWASP: A03:2021 - Injection
Category:	xss Confidence: MEDIUM
Location:	frontend/src/app/administration/administration.component.ts:57
Line(s):	57
Exploitability:	Unknown
	Impact: Unknown

Description

Detects Angular DomSanitizer security bypass **Vulnerable Code**

```
user.email = this.sanitizer.bypassSecurityTrustHtml(`<span class="${this.doesUserHaveAnActiveSession(user) ? 'confirmation' : 'error'}">${user.email}</span>`)
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the XSS issue in frontend/src/app/administration/administration.component.ts **References**

& Further Reading:

<https://owasp.org/Top10/A03/>

#972: XSS - Angular bypassSecurityTrustHtml

Severity: CRITICAL (CVSS: 9.5/10.0)

CWE ID: CWE-79 **OWASP:** A03:2021 - Injection

Category: XSS **Confidence:** MEDIUM

Location: frontend/src/app/administration/administration.component.ts:72

Line(s): 72

Exploitability: Unknown **Impact:** Unknown

Description

Detects Angular DomSanitizer security bypass **Vulnerable Code**

```
feedback.comment = this.sanitizer.bypassSecurityTrustHtml(feedback.comment)
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the xss issue in frontend/src/app/administration/administration.component.ts [References](#)

& Further Reading:

<https://owasp.org/Top10/A03/>

#973: CSRF - Password Change Without CSRF

Severity: CRITICAL (CVSS: 9.5/10.0)

CWE ID: CWE-352 **OWASP:** A07:2021 - Identification and Authentication Failures

Category: csrf **Confidence:** MEDIUM

Location: frontend/src/app/change-password/change-password.component.spec.ts:27

Line(s): 27

Exploitability: Unknown **Impact:** Unknown

Description

Detects password change without CSRF protection **Vulnerable Code**

```
userService = jasmine.createSpyObj('UserService', ['changePassword'])
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the csrf issue in frontend/src/app/change-password/change-password.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#974: CSRF - Password Change Without CSRF

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-352 OWASP: A07:2021 - Identification and Authentication Failures
Category:	csrf Confidence: MEDIUM
Location:	frontend/src/app/change-password/change-password.component.spec.ts:97
Line(s):	97
Exploitability:	Unknown Impact: Unknown

Description

Detects password change without CSRF protection **Vulnerable Code**

```
userService.changePassword.and.returnValue(of({}))
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the csrf issue in frontend/src/app/change-password/change-password.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#975: CSRF - Password Change Without CSRF

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-352 OWASP: A07:2021 - Identification and Authentication Failures
Category:	csrf Confidence: MEDIUM
Location:	frontend/src/app/change-password/change-password.component.spec.ts:109
Line(s):	109
Exploitability:	Unknown
	Impact: Unknown

Description

Detects password change without CSRF protection

```
userService.changePassword.and.returnValue throwError('Error'))
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the csrf issue in frontend/src/app/change-password/change-password.component.spec.ts

References & Further Reading:

<https://owasp.org/Top10/A07/>

#976: CSRF - Password Change Without CSRF

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-352 OWASP: A07:2021 - Identification and Authentication Failures
Category:	csrf Confidence: MEDIUM
Location:	frontend/src/app/change-password/change-password.component.ts:39
Line(s):	39
Exploitability:	Unknown
	Impact: Unknown

Description

Detects password change without CSRF protection **Vulnerable Code**

```
this.formSubmitService.attachEnterKeyHandler('password-form', 'changeButton', () => {
  this.changePassword() })
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the csrf issue in frontend/src/app/change-password/change-password.component.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#977: XSS - Angular bypassSecurityTrustHtml

Severity: CRITICAL (CVSS: 9.5/10.0)

CWE ID: CWE-79 **OWASP:** A03:2021 - Injection

Category: xss **Confidence:** MEDIUM

Location: frontend/src/app/last-login-ip/last-login-ip.component.spec.ts:21

Line(s): 21

Exploitability: Unknown **Impact:** Unknown

Description

Detects Angular DomSanitizer security bypass **Vulnerable Code**

```
sanitizer = jasmine.createSpyObj('DomSanitizer', ['bypassSecurityTrustHtml', 'sanitize'])
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the XSS issue in frontend/src/app/last-login-ip/last-login-ip.component.spec.ts

References & Further Reading:

<https://owasp.org/Top10/A03/>

#978: XSS - Angular bypassSecurityTrustHtml

Severity: CRITICAL (CVSS: 9.5/10.0)

CWE ID: CWE-79 **OWASP:** A03:2021 - Injection

Category: XSS **Confidence:** MEDIUM

Location: frontend/src/app/last-login-ip/last-login-ip.component.spec.ts:22

Line(s): 22

Exploitability: Unknown **Impact:** Unknown

Description

Detects Angular DomSanitizer security bypass

Vulnerable Code

```
sanitizer.bypassSecurityTrustHtml.and.callFake((args: any) => args)
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the xss issue in frontend/src/app/last-login-ip/last-login-ip.component.spec.ts

References & Further Reading:

<https://owasp.org/Top10/A03/>

#979: XSS - Angular bypassSecurityTrustHtml

Severity: CRITICAL (CVSS: 9.5/10.0)

CWE ID: CWE-79 **OWASP:** A03:2021 - Injection

Category: xss **Confidence:** MEDIUM

Location: frontend/src/app/last-login-ip/last-login-ip.component.spec.ts:63

Line(s): 63

Exploitability: Unknown **Impact:** Unknown

Description

Detects Angular DomSanitizer security bypass

Vulnerable Code

```
expect(sanitizer.bypassSecurityTrustHtml).toHaveBeenCalledWith('<small>1.2.3.4</small>')
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the xss issue in frontend/src/app/last-login-ip/last-login-ip.component.spec.ts

References & Further Reading:

<https://owasp.org/Top10/A03/>

#980: XSS - Angular bypassSecurityTrustHtml

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-79	OWASP: A03:2021 - Injection
Category:	xss	Confidence: MEDIUM
Location:	frontend/src/app/last-login-ip/last-login-ip.component.spec.ts:69	
Line(s):	69	
Exploitability:	Unknown	Impact: Unknown

Description

Detects Angular DomSanitizer security bypass **Vulnerable Code**

```
expect(sanitizer.bypassSecurityTrustHtml).not.toHaveBeenCalled()
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the xss issue in frontend/src/app/last-login-ip/last-login-ip.component.spec.ts

References & Further Reading:

<https://owasp.org/Top10/A03/>

#981: XSS - Angular bypassSecurityTrustHtml

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-79	OWASP: A03:2021 - Injection
Category:	xss	Confidence: MEDIUM
Location:	frontend/src/app/last-login-ip/last-login-ip.component.ts:38	
Line(s):	38	
Exploitability:	Unknown	Impact: Unknown

Description

Detects Angular DomSanitizer security bypass **Vulnerable Code**

```
this.lastLoginIp = this.sanitizer.bypassSecurityTrustHtml('<small>${payload.data.lastLoginIp}</small>')
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the XSS issue in frontend/src/app/last-login-ip/last-login-ip.component.ts **References & Further Reading:**

<https://owasp.org/Top10/A03/>

#982: XSS - Angular bypassSecurityTrustHtml

Severity: CRITICAL (CVSS: 9.5/10.0)

CWE ID: CWE-79 **OWASP:** A03:2021 - Injection

Category: XSS **Confidence:** MEDIUM

Location: frontend/src/app/last-login-ip/last-login-ip.component.ts:38

Line(s): 38

Exploitability: Unknown **Impact:** Unknown

Description

Detects Angular DomSanitizer security bypass **Vulnerable Code**

```
this.lastLoginIp = this.sanitizer.bypassSecurityTrustHtml('<small>${payload.data.lastLoginIp}</small>')
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the XSS issue in frontend/src/app/last-login-ip/last-login-ip.component.ts

References & Further Reading:

<https://owasp.org/Top10/A03/>

#983: Hardcoded Credentials - Default Passwords

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/api/2faSpec.ts:311	
Line(s):	311	
Exploitability:	Unknown	Impact: Unknown

Description

Detects common default passwords in code

```
const password = '123456'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/2faSpec.ts

References & Further Reading:

<https://owasp.org/Top10/A07/>

#984: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/api/2faSpec.ts:174	
Line(s):	174	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'EinBelegtesBrotMitSchinkenSCHINKEN!',
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/2faSpec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#985: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/2faSpec.ts:200
Line(s):	200
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: '0Y8rMnww$*9VFYE 59-!Fg1L6t&61B'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in test/api/2faSpec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#986: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/2faSpec.ts:369
Line(s):	369
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
const password = 'EinBelegtesBrotMitSchinkenSCHINKEN!'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/2faSpec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#987: Hardcoded Credentials - Default Passwords

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/api/2faSpec.ts:236	
Line(s):	236	
Exploitability:	Unknown	Impact: Unknown

Description

Detects common default passwords in code **Vulnerable Code**

```
const password = '123456'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data

- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in test/api/2faSpec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#988: Hardcoded Credentials - Default Passwords

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/api/2faSpec.ts:282	
Line(s):	282	
Exploitability:	Unknown	Impact: Unknown

Description

Detects common default passwords in code **Vulnerable Code**

```
const password = '123456'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in test/api/2faSpec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#989: Hardcoded Credentials - Private Keys

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798 OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication Confidence: MEDIUM
Location:	lib/insecurity.ts:23
Line(s):	23
Exploitability:	Unknown Impact: Unknown

Description

Detects hardcoded private keys in code **Vulnerable Code**

```
const privateKey = '-----BEGIN RSA PRIVATE KEY-----\r\nMIICXAIBAAKBgQDNwqLEe9wgTXCbC7+RPdDbBbeqdjdb4kOPOIGzdLp...
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in lib/insecurity.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#990: CSRF - Password Change Without CSRF

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-352 OWASP: A07:2021 - Identification and Authentication Failures
Category:	csrf Confidence: MEDIUM
Location:	routes/changePassword.ts:12
Line(s):	12
Exploitability:	Unknown Impact: Unknown

Description

Detects password change without CSRF protection **Vulnerable Code**

```
export function changePassword () {
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the csrf issue in routes/changePassword.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#991: CSRF - Password Change Without CSRF

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-352
Category:	csrf
Location:	routes/changePassword.ts:53
Line(s):	53
Exploitability:	Unknown
	Impact: Unknown

Description

Detects password change without CSRF protection **Vulnerable Code**

```
challenges.changePasswordBenderChallenge,
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the csrf issue in routes/changePassword.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#992: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	routes/login.ts:34	
Line(s):	34	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
models.sequelize.query('SELECT * FROM Users WHERE email = '${req.body.email || ''}'  
AND password = '${security.hash(req.body.password || '')}' AND deletedAt IS NULL', {  
model: UserModel, plain: true }) // vuln-code-snippet vuln-line loginAdminChallenge  
loginBenderChallenge loginJimChallenge
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in routes/login.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#993: Horizontal Access Control - Tenant Isolation Bypass

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-566	OWASP: A01:2021 - Broken Access Control
Category:	access_control	Confidence: MEDIUM
Location:	routes/login.ts:34	
Line(s):	34	
Exploitability:	Unknown	Impact: Unknown

Description

Detects multi-tenant isolation bypass **Vulnerable Code**

```
models.sequelize.query('SELECT * FROM Users WHERE email = '${req.body.email || ''}'  
AND password = '${security.hash(req.body.password || '')}' AND deletedAt IS NULL', {  
model: UserModel, plain: true }) // vuln-code-snippet vuln-line loginAdminChallenge  
loginBenderChallenge loginJimChallenge
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the access_control issue in routes/login.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

#994: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/erasureRequestApiSpec.ts:18
Line(s):	18
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'kitten lesser pooch karate buffoonindoors'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in test/api/erasureRequestApiSpec.ts **References &**

Further Reading:

<https://owasp.org/Top10/A07/>

#995: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/erasureRequestApiSpec.ts:37
Line(s):	37
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/erasureRequestApiSpec.ts **References &**

Further Reading:

<https://owasp.org/Top10/A07/>

#996: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/erasureRequestApiSpec.ts:64
Line(s):	64
Exploitability:	Unknown
	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/erasureRequestApiSpec.ts **References &**

Further Reading:

<https://owasp.org/Top10/A07/>

#997: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/api/erasureRequestApiSpec.ts:80	
Line(s):	80	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/erasureRequestApiSpec.ts **References &**

Further Reading:

<https://owasp.org/Top10/A07/>

#998: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798 OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication Confidence: MEDIUM
Location:	test/api/erasureRequestApiSpec.ts:99
Line(s):	99
Exploitability:	Unknown Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/erasureRequestApiSpec.ts **References &**

Further Reading:

<https://owasp.org/Top10/A07/>

#999: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/erasureRequestApiSpec.ts:119
Line(s):	119
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in test/api/erasureRequestApiSpec.ts **References &**

Further Reading:

<https://owasp.org/Top10/A07/>

#1000: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/erasureRequestApiSpec.ts:140
Line(s):	140
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/erasureRequestApiSpec.ts **References &**

Further Reading:

<https://owasp.org/Top10/A07/>

#1001: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/feedbackApiSpec.ts:120
Line(s):	120
Exploitability:	Unknown
	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/feedbackApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1002: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/api/feedbackApiSpec.ts:153	
Line(s):	153	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/feedbackApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1003: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798 OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication Confidence: MEDIUM
Location:	test/api/loginApiSpec.ts:21
Line(s):	21
Exploitability:	Unknown Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'kallliiii'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/loginApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1004: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/loginApiSpec.ts:30
Line(s):	30
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'kallliiii'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in test/api/loginApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1005: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/loginApiSpec.ts:64
Line(s):	64
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'admin123'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/loginApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1006: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/loginApiSpec.ts:79
Line(s):	79
Exploitability:	Unknown
	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'J6aVjTgOpRs@?51!Zkq2AYnCE@RF$P'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/loginApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1007: Hardcoded Credentials - Default Passwords

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/api/2faSpec.ts:340	
Line(s):	340	
Exploitability:	Unknown	Impact: Unknown

Description

Detects common default passwords in code **Vulnerable Code**

```
const password = '123456'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/2faSpec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#1008: Hardcoded Credentials - Default Passwords

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/api/2faSpec.ts:398	
Line(s):	398	
Exploitability:	Unknown	Impact: Unknown

Description

Detects common default passwords in code **Vulnerable Code**

```
const password = '123456'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/2faSpec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#1009: Hardcoded Credentials - Default Passwords

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/2faSpec.ts:435
Line(s):	435
Exploitability:	Unknown
Impact:	Unknown

Description

Detects common default passwords in code **Vulnerable Code**

```
const password = '123456'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in test/api/2faSpec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#1010: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/addressApiSpec.ts:20
Line(s):	20
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'ncc-1701'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/addressApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1011: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/authenticatedUsersSpec.ts:21
Line(s):	21
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: '*****'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/authenticatedUsersSpec.ts **References &**

Further Reading:

<https://owasp.org/Top10/A07/>

#1012: Hardcoded Credentials - Database Password

Severity: CRITICAL (CVSS: 9.5/10.0)

CWE ID: CWE-798 **OWASP:** A07:2021 - Identification and Authentication Failures

Category: authentication **Confidence:** MEDIUM

Location: test/api/authenticatedUsersSpec.ts:30

Line(s): 30

Exploitability: Unknown **Impact:** Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'ncc-1701'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/authenticatedUsersSpec.ts **References &**

Further Reading:

<https://owasp.org/Top10/A07/>

#1013: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798 OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication Confidence: MEDIUM
Location:	test/api/basketApiSpec.ts:25
Line(s):	25
Exploitability:	Unknown Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'ncc-1701'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/basketApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1014: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/basketApiSpec.ts:101
Line(s):	101
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in test/api/basketApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1015: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/basketItemApiSpec.ts:21
Line(s):	21
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'ncc-1701'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/basketItemApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1016: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/chatBotSpec.ts:56
Line(s):	56
Exploitability:	Unknown
	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: '0Y8rMnww$*9VFYE 59-!Fg1L6t&61B'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/chatBotSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1017: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/api/chatBotSpec.ts:77	
Line(s):	77	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: '0Y8rMnww$*9VFYE 59-!Fg1L6t&61B'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/chatBotSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1018: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798 OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication Confidence: MEDIUM
Location:	test/api/chatBotSpec.ts:108
Line(s):	108
Exploitability:	Unknown Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/chatBotSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1019: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/chatBotSpec.ts:140
Line(s):	140
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in test/api/chatBotSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1020: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/chatBotSpec.ts:174
Line(s):	174
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/chatBotSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1021: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/chatBotSpec.ts:205
Line(s):	205
Exploitability:	Unknown
	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'ship coffin krypt cross estate supply insurance asbestos souvenir'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/chatBotSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1022: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/api/chatBotSpec.ts:250	
Line(s):	250	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/chatBotSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1023: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798 OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication Confidence: MEDIUM
Location:	test/api/chatBotSpec.ts:287
Line(s):	287
Exploitability:	Unknown Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'testtesttest',
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/chatBotSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1024: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/chatBotSpec.ts:295
Line(s):	295
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'testtesttest'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in test/api/chatBotSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1025: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/dataExportApiSpec.ts:21
Line(s):	21
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/dataExportApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1026: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/dataExportApiSpec.ts:48
Line(s):	48
Exploitability:	Unknown
	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/dataExportApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1027: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/api/dataExportApiSpec.ts:77	
Line(s):	77	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/dataExportApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1028: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798 OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication Confidence: MEDIUM
Location:	test/api/dataExportApiSpec.ts:112
Line(s):	112
Exploitability:	Unknown Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'K1f.....'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/dataExportApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1029: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/dataExportApiSpec.ts:152
Line(s):	152
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'ncc-1701'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in test/api/dataExportApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1030: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/dataExportApiSpec.ts:194
Line(s):	194
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'ncc-1701'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/dataExportApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1031: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/dataExportApiSpec.ts:234
Line(s):	234
Exploitability:	Unknown
	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'K1f.....'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/dataExportApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1032: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/api/dataExportApiSpec.ts:282	
Line(s):	282	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

password: 'ncc-1701'

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/dataExportApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1033: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798 OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication Confidence: MEDIUM
Location:	test/api/dataExportApiSpec.ts:332
Line(s):	332
Exploitability:	Unknown Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'ncc-1701'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/dataExportApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1034: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/deliveryApiSpec.ts:23
Line(s):	23
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'ncc-1701'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in test/api/deliveryApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1035: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/deliveryApiSpec.ts:52
Line(s):	52
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'mDLx?94T~1CfVfZMzw@sJ9f?s3L61bMqE70FfI8^54jbNikY5fyrmx7c!YbJb'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/deliveryApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1036: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/deliveryApiSpec.ts:83
Line(s):	83
Exploitability:	Unknown
	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'ncc-1701'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/deliveryApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1037: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/api/deliveryApiSpec.ts:111	
Line(s):	111	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'mDLx?94T~1CfVfZMzw@sJ9f?s3L61bMqE70FFI8^54jbNikY5fymx7c!YbJb'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/deliveryApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1038: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798 OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication Confidence: MEDIUM
Location:	test/api/deluxeApiSpec.ts:35
Line(s):	35
Exploitability:	Unknown Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: '0hG0dPlease1nsertLiquor!'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/deluxeApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1039: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/deluxeApiSpec.ts:53
Line(s):	53
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'mDLx?94T~1CfVfZMzw@sJ9f?s3L61bMqE70FfI8^54jbNikY5fyrmx7c!YbJb'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in test/api/deluxeApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1040: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/deluxeApiSpec.ts:71
Line(s):	71
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'admin123'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/deluxeApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1041: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/deluxeApiSpec.ts:89
Line(s):	89
Exploitability:	Unknown
	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'i am an awesome accountant'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/deluxeApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1042: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/api/deluxeApiSpec.ts:105	
Line(s):	105	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: '0hG0dPlease1nsertLiquor!'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/deluxeApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1043: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798 OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication Confidence: MEDIUM
Location:	test/api/deluxeApiSpec.ts:129
Line(s):	129
Exploitability:	Unknown Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'ncc-1701'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/deluxeApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1044: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/deluxeApiSpec.ts:149
Line(s):	149
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'mDLx?94T~1CfVfZMzw@sJ9f?s3L61bMqE70FfI8^54jbNikY5fyrmx7c!YbJb'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in test/api/deluxeApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1045: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/deluxeApiSpec.ts:170
Line(s):	170
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'admin123'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/deluxeApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1046: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/deluxeApiSpec.ts:191
Line(s):	191
Exploitability:	Unknown
	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'i am an awesome accountant'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/deluxeApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1047: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/api/productReviewApiSpec.ts:112	
Line(s):	112	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/productReviewApiSpec.ts **References &**

Further Reading:

<https://owasp.org/Top10/A07/>

#1048: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798 OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication Confidence: MEDIUM
Location:	test/api/productReviewApiSpec.ts:132
Line(s):	132
Exploitability:	Unknown Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/productReviewApiSpec.ts **References &**

Further Reading:

<https://owasp.org/Top10/A07/>

#1049: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/loginApiSpec.ts:94
Line(s):	94
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'Mr. NOodles'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in test/api/loginApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1050: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/loginApiSpec.ts:109
Line(s):	109
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'K1f.....'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/loginApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1051: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/loginApiSpec.ts:124
Line(s):	124
Exploitability:	Unknown
	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'EinBelegtesBrotMitSchinkenSCHINKEN!'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/loginApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1052: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/api/loginApiSpec.ts:142	
Line(s):	142	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/loginApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1053: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798 OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication Confidence: MEDIUM
Location:	test/api/loginApiSpec.ts:245
Line(s):	245
Exploitability:	Unknown Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/loginApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1054: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/loginApiSpec.ts:266
Line(s):	266
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in test/api/loginApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1055: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/memoryApiSpec.ts:26
Line(s):	26
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'ncc-1701'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/memoryApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1056: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/memoryApiSpec.ts:64
Line(s):	64
Exploitability:	Unknown
	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'ncc-1701'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/memoryApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1057: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/api/memoryApiSpec.ts:86	
Line(s):	86	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'ncc-1701'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/memoryApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1058: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/api/memoryApiSpec.ts:112	
Line(s):	112	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'ncc-1701'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/memoryApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1059: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/memoryApiSpec.ts:142
Line(s):	142
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'ncc-1701'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in test/api/memoryApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1060: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/orderHistoryApiSpec.ts:19
Line(s):	19
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'admin123'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/orderHistoryApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1061: Hardcoded Credentials - Database Password

Severity: **CRITICAL** (CVSS: 9.5/10.0)

CWE ID: CWE-798 **OWASP:** A07:2021 - Identification and Authentication Failures

Category: authentication **Confidence:** MEDIUM

Location: test/api/orderHistoryApiSpec.ts:56

Line(s): 56

Exploitability: Unknown **Impact:** Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'ncc-1701'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/orderHistoryApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1062: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/api/orderHistoryApiSpec.ts:73	
Line(s):	73	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'admin123'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/orderHistoryApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1063: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798 OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication Confidence: MEDIUM
Location:	test/api/orderHistoryApiSpec.ts:90
Line(s):	90
Exploitability:	Unknown Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'i am an awesome accountant'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/orderHistoryApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1064: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/orderHistoryApiSpec.ts:109
Line(s):	109
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'admin123'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in test/api/orderHistoryApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1065: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/orderHistoryApiSpec.ts:129
Line(s):	129
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'ncc-1701'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/orderHistoryApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1066: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/orderHistoryApiSpec.ts:149
Line(s):	149
Exploitability:	Unknown
	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'i am an awesome accountant'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/orderHistoryApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1067: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/api/passwordApiSpec.ts:20	
Line(s):	20	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'kunigunde'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/passwordApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1068: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798 OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication Confidence: MEDIUM
Location:	test/api/passwordApiSpec.ts:29
Line(s):	29
Exploitability:	Unknown Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'kunigunde'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/passwordApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1069: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/passwordApiSpec.ts:47
Line(s):	47
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'monkey summer birthday are all bad passwords but work just fine in a long  
passphrase'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/passwordApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1070: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/passwordApiSpec.ts:93
Line(s):	93
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'OhG0dPlease1nsertLiquor!'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/passwordApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1071: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/paymentApiSpec.ts:20
Line(s):	20
Exploitability:	Unknown
	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'ncc-1701'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/paymentApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1072: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/cypress/e2e/changePassword.spec.ts:6	
Line(s):	6	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'focusOnScienceMorty!focusOnScience'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/cypress/e2e/changePassword.spec.ts **References**

& Further Reading:

<https://owasp.org/Top10/A07/>

#1073: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/cypress/e2e/changePassword.spec.ts:25
Line(s):	25
Exploitability:	Unknown
	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: '0hG0dPlease1nsertLiquor!'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/cypress/e2e/changePassword.spec.ts **References**

& Further Reading:

<https://owasp.org/Top10/A07/>

#1074: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/profileImageUploadSpec.ts:25
Line(s):	25
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'ncc-1701'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in test/api/profileImageUploadSpec.ts **References &**

Further Reading:

<https://owasp.org/Top10/A07/>

#1075: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/profileImageUploadSpec.ts:52
Line(s):	52
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'ncc-1701'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/profileImageUploadSpec.ts **References &**

Further Reading:

<https://owasp.org/Top10/A07/>

#1076: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/profileImageUploadSpec.ts:97
Line(s):	97
Exploitability:	Unknown
	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'ncc-1701'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/profileImageUploadSpec.ts **References &**

Further Reading:

<https://owasp.org/Top10/A07/>

#1077: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/api/profileImageUploadSpec.ts:123	
Line(s):	123	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'ncc-1701'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/profileImageUploadSpec.ts **References &**

Further Reading:

<https://owasp.org/Top10/A07/>

#1078: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798 OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication Confidence: MEDIUM
Location:	test/api/profileImageUploadSpec.ts:164
Line(s):	164
Exploitability:	Unknown Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'ncc-1701'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/profileImageUploadSpec.ts **References &**

Further Reading:

<https://owasp.org/Top10/A07/>

#1079: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/quantityApiSpec.ts:21
Line(s):	21
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'ncc-1701'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in test/api/quantityApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1080: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/quantityApiSpec.ts:38
Line(s):	38
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'admin123'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/quantityApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1081: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/quantityApiSpec.ts:55
Line(s):	55
Exploitability:	Unknown
	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'i am an awesome accountant'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/quantityApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1082: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/api/quantityApiSpec.ts:72	
Line(s):	72	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

password: 'ncc-1701'

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/quantityApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1083: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798 OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication Confidence: MEDIUM
Location:	test/api/quantityApiSpec.ts:93
Line(s):	93
Exploitability:	Unknown Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'admin123'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/quantityApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1084: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/quantityApiSpec.ts:114
Line(s):	114
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'i am an awesome accountant'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in test/api/quantityApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1085: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/quantityApiSpec.ts:137
Line(s):	137
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'ncc-1701'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/quantityApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1086: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/quantityApiSpec.ts:155
Line(s):	155
Exploitability:	Unknown
	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'admin123'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/quantityApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1087: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/api/quantityApiSpec.ts:173	
Line(s):	173	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'i am an awesome accountant'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/quantityApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1088: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798 OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication Confidence: MEDIUM
Location:	test/api/quantityApiSpec.ts:190
Line(s):	190
Exploitability:	Unknown Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'i am an awesome accountant'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/quantityApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1089: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/quantityApiSpec.ts:207
Line(s):	207
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'ncc-1701'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in test/api/quantityApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1090: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/quantityApiSpec.ts:228
Line(s):	228
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'ncc-1701'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/quantityApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1091: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/quantityApiSpec.ts:249
Line(s):	249
Exploitability:	Unknown
	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'i am an awesome accountant'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/quantityApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1092: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/api/quantityApiSpec.ts:269	
Line(s):	269	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'i am an awesome accountant'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/quantityApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1093: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798 OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication Confidence: MEDIUM
Location:	test/api/quantityApiSpec.ts:292
Line(s):	292
Exploitability:	Unknown Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'i am an awesome accountant'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/quantityApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1094: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/quantityApiSpec.ts:309
Line(s):	309
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'admin123'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in test/api/quantityApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1095: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/quantityApiSpec.ts:326
Line(s):	326
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'ncc-1701'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/quantityApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1096: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/userApiSpec.ts:45
Line(s):	45
Exploitability:	Unknown
	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'hooooorst'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/userApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1097: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/api/userApiSpec.ts:63	
Line(s):	63	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'hooooorst',
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/userApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1098: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798 OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication Confidence: MEDIUM
Location:	test/api/userApiSpec.ts:139
Line(s):	139
Exploitability:	Unknown Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'hooooorst',
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/userApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1099: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/userApiSpec.ts:161
Line(s):	161
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'hooooorst',
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in test/api/userApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1100: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/userApiSpec.ts:183
Line(s):	183
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'hooooorst',
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/userApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1101: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/api/userApiSpec.ts:202
Line(s):	202
Exploitability:	Unknown
	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'does.not.matter'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/userApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1102: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/api/userApiSpec.ts:256	
Line(s):	256	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/userApiSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1103: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798 OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication Confidence: MEDIUM
Location:	test/api/userProfileSpec.ts:19
Line(s):	19
Exploitability:	Unknown Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'ncc-1701'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/api/userProfileSpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1104: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/cypress/e2e/administration.spec.ts:5
Line(s):	5
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings [Vulnerable Code](#)

```
password: 'admin123'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in test/cypress/e2e/administration.spec.ts [References & Resources](#)

Further Reading:

<https://owasp.org/Top10/A07/>

#1105: Business Logic - Purchase Without Payment

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-840
Category:	business_logic
Location:	data/datacreator.ts:52
Line(s):	52
Exploitability:	Unknown
Impact:	Unknown

Description

Detects order confirmation without payment verification **Vulnerable Code**

```
createOrders,
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the business_logic issue in data/datacreator.ts **References & Further Reading:**

<https://owasp.org/Top10/A04/>

#1106: Business Logic - Purchase Without Payment

Severity: CRITICAL (CVSS: 9.5/10.0)

CWE ID: CWE-840 **OWASP:** A04:2021 - Insecure Design

Category: business_logic **Confidence:** MEDIUM

Location: data/datacreator.ts:634

Line(s): 634

Exploitability: Unknown **Impact:** Unknown

Description

Detects order confirmation without payment verification **Vulnerable Code**

```
async function createOrders () {
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the business_logic issue in data/datacreator.ts

References & Further Reading:

<https://owasp.org/Top10/A04/>

#1107: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/cypress/e2e/changePassword.spec.ts:31	
Line(s):	31	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings

Vulnerable Code

```
cy.login({ email: 'bender', password: 'slurmCl4ssic' })
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/cypress/e2e/changePassword.spec.ts

References & Further Reading:

<https://owasp.org/Top10/A07/>

#1108: CSRF - Password Change Without CSRF

Severity:	CRITICAL	(CVSS: 9.5/10.0)
CWE ID:	CWE-352	OWASP: A07:2021 - Identification and Authentication Failures
Category:	csrf	Confidence: MEDIUM
Location:	test/cypress/e2e/changePassword.spec.ts:21	
Line(s):	21	
Exploitability:	Unknown	Impact: Unknown

Description

Detects password change without CSRF protection **Vulnerable Code**

```
describe('challenge "changePasswordBenderChallenge"', () => {
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the csrf issue in test/cypress/e2e/changePassword.spec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1109: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798 OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication Confidence: MEDIUM
Location:	test/cypress/e2e/restApi.spec.ts:4
Line(s):	4
Exploitability:	Unknown Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
cy.login({ email: 'admin', password: 'admin123' })
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in test/cypress/e2e/restApi.spec.ts **References &**

Further Reading:

<https://owasp.org/Top10/A07/>

#1110: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798 OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication Confidence: MEDIUM
Location:	test/cypress/e2e/restApi.spec.ts:82
Line(s):	82
Exploitability:	Unknown Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'admin123'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/cypress/e2e/restApi.spec.ts **References &**

Further Reading:

<https://owasp.org/Top10/A07/>

#1111: Vertical Access Control - Unrestricted File Upload Types

Severity: **CRITICAL** (CVSS: 9.5/10.0)

CWE ID: CWE-434 **OWASP:** A04:2021 - Insecure Design

Category: access_control **Confidence:** MEDIUM

Location: server.ts:661

Line(s): 661

Exploitability: Unknown **Impact:** Unknown

Description

Detects file upload without type restrictions **Vulnerable Code**

```
const uploadToMemory = multer({ storage: multer.memoryStorage(), limits: { fileSize: 200000 } })
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the access_control issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A04/>

#1112: CSRF - Password Change Without CSRF

Severity: CRITICAL (CVSS: 9.5/10.0)

CWE ID: CWE-352 **OWASP:** A07:2021 - Identification and Authentication Failures

Category: csrf **Confidence:** MEDIUM

Location: server.ts:95

Line(s): 95

Exploitability: Unknown **Impact:** Unknown

Description

Detects password change without CSRF protection **Vulnerable Code**

```
import { changePassword } from './routes/changePassword'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the csrf issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#1113: CSRF - Password Change Without CSRF

Severity:	CRITICAL	(CVSS: 9.5/10.0)
CWE ID:	CWE-352	OWASP: A07:2021 - Identification and Authentication Failures
Category:	csrf	Confidence: MEDIUM
Location:	server.ts:575	
Line(s):	575	
Exploitability:	Unknown	Impact: Unknown

Description

Detects password change without CSRF protection **Vulnerable Code**

```
app.get('/rest/user/change-password', changePassword())
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the csrf issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#1114: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	data/static/users.yml:3	
Line(s):	3	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'admin123'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in data/static/users.yml **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#1115: Horizontal Access Control - Tenant Isolation Bypass

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-566	OWASP: A01:2021 - Broken Access Control
Category:	access_control	Confidence: MEDIUM
Location:	data/static/codefixes/dbSchemaChallenge_1.ts:5	
Line(s):	5	
Exploitability:	Unknown	Impact: Unknown

Description

Detects multi-tenant isolation bypass **Vulnerable Code**

```
models.sequelize.query("SELECT * FROM Products WHERE ((name LIKE '%"+criteria+"%' OR  
description LIKE '%"+criteria+"%') AND deletedAt IS NULL) ORDER BY name")
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the access_control issue in data/static/codefixes/dbSchemaChallenge_1.ts **References**

& Further Reading:

<https://owasp.org/Top10/A01/>

#1116: SQL Injection - TypeScript ORM Raw Queries

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-89	OWASP: A03:2021 - Injection
Category:	sql_injection	Confidence: MEDIUM
Location:	data/static/codefixes/dbSchemaChallenge_1.ts:5	
Line(s):	5	
Exploitability:	Unknown	Impact: Unknown

Description

Detects raw SQL queries in TypeScript ORM libraries without parameterization **Vulnerable**

Code

```
models.sequelize.query("SELECT * FROM Products WHERE ((name LIKE '%"+criteria+"%' OR  
description LIKE '%"+criteria+"%') AND deletedAt IS NULL) ORDER BY name")
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the sql_injection issue in data/static/codefixes/dbSchemaChallenge_1.ts **References**

& Further Reading:

<https://owasp.org/Top10/A03/>

#1117: Horizontal Access Control - Tenant Isolation Bypass

Severity: CRITICAL (CVSS: 9.5/10.0)

CWE ID: CWE-566 **OWASP:** A01:2021 - Broken Access Control

Category: access_control **Confidence:** MEDIUM

Location: data/static/codefixes/dbSchemaChallenge_3.ts:11

Line(s): 11

Exploitability: Unknown **Impact:** Unknown

Description

Detects multi-tenant isolation bypass **Vulnerable Code**

```
models.sequelize.query('SELECT * FROM Products WHERE ((name LIKE \'%${criteria}%' OR  
description LIKE \'%${criteria}\%') AND deletedAt IS NULL) ORDER BY name')
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the access_control issue in data/static/codefixes/dbSchemaChallenge_3.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

#1118: XSS - Angular bypassSecurityTrustHtml

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-79	OWASP: A03:2021 - Injection
Category:	xss	Confidence: MEDIUM
Location:	data/static/codefixes/restfulXssChallenge_3.ts:45	
Line(s):	45	
Exploitability:	Unknown	Impact: Unknown

Description

Detects Angular DomSanitizer security bypass **Vulnerable Code**

```
tableData[i].description = this.sanitizer.bypassSecurityTrustHtml(tableData[i].description)
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the XSS issue in data/static/codefixes/restfulXssChallenge_3.ts **References & Further Reading:**

<https://owasp.org/Top10/A03/>

#1119: XSS - Angular bypassSecurityTrustHtml

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-79	OWASP: A03:2021 - Injection
Category:	xss	Confidence: MEDIUM
Location:	data/static/codefixes/restfulXssChallenge_3.ts:45	
Line(s):	45	
Exploitability:	Unknown	Impact: Unknown

Description

Detects Angular DomSanitizer security bypass **Vulnerable Code**

```
tableData[i].description = this.sanitizer.bypassSecurityTrustHtml(tableData[i].description)
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the XSS issue in data/static/codefixes/restfulXssChallenge_3.ts **References &**

Further Reading:

<https://owasp.org/Top10/A03/>

#1120: Angular - bypassSecurityTrust XSS

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-79	OWASP: A03:2021 - Injection
Category:	angular_security	Confidence: MEDIUM
Location:	data/static/codefixes/restfulXssChallenge_4.ts:59	
Line(s):	59	
Exploitability:	Unknown	Impact: Unknown

Description

Detects bypassSecurityTrust methods with user input

Vulnerable Code

```
tableData[i].description = this.sanitizer.bypassSecurityTrustScript(tableData[i].description)
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the angular_security issue in data/static/codefixes/restfulXssChallenge_4.ts

& Further Reading:

<https://owasp.org/Top10/A03/>

#1121: XSS - Angular bypassSecurityTrustScript

Severity: CRITICAL (CVSS: 9.5/10.0)

CWE ID: CWE-79 **OWASP:** A03:2021 - Injection

Category: xss **Confidence:** MEDIUM

Location: data/static/codefixes/restfulXssChallenge_4.ts:59

Line(s): 59

Exploitability: Unknown **Impact:** Unknown

Description

Detects Angular script trust bypass

Vulnerable Code

```
tableData[i].description = this.sanitizer.bypassSecurityTrustScript(tableData[i].description)
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the XSS issue in data/static/codefixes/restfulXssChallenge_4.ts **References &**

Further Reading:

<https://owasp.org/Top10/A03/>

#1122: XSS - Angular bypassSecurityTrustScript

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-79	OWASP: A03:2021 - Injection
Category:	xss	Confidence: MEDIUM
Location:	data/static/codefixes/restfulXssChallenge_4.ts:59	
Line(s):	59	
Exploitability:	Unknown	Impact: Unknown

Description

Detects Angular script trust bypass **Vulnerable Code**

```
tableData[i].description = this.sanitizer.bypassSecurityTrustScript(tableData[i].description)
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the XSS issue in data/static/codefixes/restfulXssChallenge_4.ts **References &**

Further Reading:

<https://owasp.org/Top10/A03/>

#1123: Horizontal Access Control - Tenant Isolation Bypass

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-566 OWASP: A01:2021 - Broken Access Control
Category:	access_control Confidence: MEDIUM
Location:	data/static/codefixes/unionSqlInjectionChallenge_1.ts:6
Line(s):	6
Exploitability:	Unknown Impact: Unknown

Description

Detects multi-tenant isolation bypass **Vulnerable Code**

```
models.sequelize.query('SELECT * FROM Products WHERE ((name LIKE \'%${criteria}%' OR  
description LIKE \'%${criteria}\%') AND deletedAt IS NULL) ORDER BY name')
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the access_control issue in data/static/codefixes/unionSqlInjectionChallenge_1.ts **References**

& Further Reading:

<https://owasp.org/Top10/A01/>

#1124: Horizontal Access Control - Tenant Isolation Bypass

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-566
Category:	access_control
Location:	data/static/codefixes/unionSqlInjectionChallenge_3.ts:10
Line(s):	10
Exploitability:	Unknown
Impact:	Unknown

Description

Detects multi-tenant isolation bypass **Vulnerable Code**

```
models.sequelize.query('SELECT * FROM Products WHERE ((name LIKE \'%${criteria}\' OR  
description LIKE \'%${criteria}\') AND deletedAt IS NULL) ORDER BY name')
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the access_control issue in data/static/codefixes/unionSqlInjectionChallenge_3.ts **References**

& Further Reading:

<https://owasp.org/Top10/A01/>

#1125: XSS - Angular bypassSecurityTrustHtml

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-79	OWASP: A03:2021 - Injection
Category:	xss	Confidence: MEDIUM
Location:	frontend/src/app/about/about.component.ts:122	
Line(s):	122	
Exploitability:	Unknown	Impact: Unknown

Description

Detects Angular DomSanitizer security bypass **Vulnerable Code**

```
feedbacks[i].comment = this.sanitizer.bypassSecurityTrustHtml(
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the xss issue in frontend/src/app/about/about.component.ts **References &**

Further Reading:

<https://owasp.org/Top10/A03/>

#1126: XSS - Angular bypassSecurityTrustHtml

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-79	OWASP: A03:2021 - Injection
Category:	xss	Confidence: MEDIUM
Location:	frontend/src/app/about/about.component.ts:122	
Line(s):	122	
Exploitability:	Unknown	Impact: Unknown

Description

Detects Angular DomSanitizer security bypass **Vulnerable Code**

```
feedbacks[i].comment = this.sanitizer.bypassSecurityTrustHtml(
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the XSS issue in frontend/src/app/about/about.component.ts **References &**

Further Reading:

<https://owasp.org/Top10/A03/>

#1127: Angular - bypassSecurityTrust XSS

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-79 OWASP: A03:2021 - Injection
Category:	angular_security Confidence: MEDIUM
Location:	data/static/codefixes/localXssChallenge_1.ts:6
Line(s):	6
Exploitability:	Unknown Impact: Unknown

Description

Detects bypassSecurityTrust methods with user input **Vulnerable Code**

```
this.searchValue = this.sanitizer.bypassSecurityTrustResourceUrl(queryParam)
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the angular_security issue in data/static/codefixes/localXssChallenge_1.ts **References**

& Further Reading:

<https://owasp.org/Top10/A03/>

#1128: Angular - bypassSecurityTrust XSS

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-79	OWASP: A03:2021 - Injection
Category:	angular_security	Confidence: MEDIUM
Location:	data/static/codefixes/localXssChallenge_3.ts:6	
Line(s):	6	
Exploitability:	Unknown	Impact: Unknown

Description

Detects bypassSecurityTrust methods with user input **Vulnerable Code**

```
this.searchValue = this.sanitizer.bypassSecurityTrustScript(queryParam)
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the angular_security issue in data/static/codefixes/localXssChallenge_3.ts **References**

& Further Reading:

<https://owasp.org/Top10/A03/>

#1129: XSS - Angular bypassSecurityTrustScript

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-79	OWASP: A03:2021 - Injection
Category:	xss	Confidence: MEDIUM
Location:	data/static/codefixes/localXssChallenge_3.ts:6	
Line(s):	6	
Exploitability:	Unknown	Impact: Unknown

Description

Detects Angular script trust bypass **Vulnerable Code**

```
this.searchValue = this.sanitizer.bypassSecurityTrustScript(queryParam)
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the XSS issue in data/static/codefixes/localXssChallenge_3.ts **References &**

Further Reading:

<https://owasp.org/Top10/A03/>

#1130: XSS - Angular bypassSecurityTrustScript

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-79	OWASP: A03:2021 - Injection
Category:	xss	Confidence: MEDIUM
Location:	data/static/codefixes/localXssChallenge_3.ts:6	
Line(s):	6	
Exploitability:	Unknown	Impact: Unknown

Description

Detects Angular script trust bypass **Vulnerable Code**

```
this.searchValue = this.sanitizer.bypassSecurityTrustScript(queryParam)
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the xss issue in data/static/codefixes/localXssChallenge_3.ts **References &**

Further Reading:

<https://owasp.org/Top10/A03/>

#1131: Angular - bypassSecurityTrust XSS

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-79	OWASP: A03:2021 - Injection
Category:	angular_security	Confidence: MEDIUM
Location:	data/static/codefixes/xssBonusChallenge_2.ts:6	
Line(s):	6	
Exploitability:	Unknown	Impact: Unknown

Description

Detects bypassSecurityTrust methods with user input

Vulnerable Code

```
this.searchValue = this.sanitizer.bypassSecurityTrustResourceUrl(queryParam)
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the angular_security issue in data/static/codefixes/xssBonusChallenge_2.ts

& Further Reading:

<https://owasp.org/Top10/A03/>

#1132: Business Logic - Withdrawal Limit Bypass

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-840
Category:	business_logic
Location:	OWASP: A04:2021 - Insecure Design
Line(s):	frontend/src/app/faucet/faucet.component.ts:231
Exploitability:	Unknown
Impact:	Unknown

Description

Detects withdrawal limit bypass vulnerabilities

Vulnerable Code

```
const tx = await contract.withdraw(amount)
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the business_logic issue in frontend/src/app/faucet/faucet.component.ts **References**

& Further Reading:

<https://owasp.org/Top10/A04/>

#1133: CSRF - Password Change Without CSRF

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-352	OWASP: A07:2021 - Identification and Authentication Failures
Category:	csrf	Confidence: MEDIUM
Location:	frontend/src/app/Services/user.service.ts:53	
Line(s):	53	
Exploitability:	Unknown	Impact: Unknown

Description

Detects password change without CSRF protection **Vulnerable Code**

```
changePassword (passwords: Passwords) {
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the csrf issue in frontend/src/app/Services/user.service.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1134: Vertical Access Control - Unrestricted File Upload Types

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-434	OWASP: A04:2021 - Insecure Design
Category:	access_control	Confidence: MEDIUM
Location:	frontend/src/app/complaint/complaint.component.ts:81	
Line(s):	81	
Exploitability:	Unknown	Impact: Unknown

Description

Detects file upload without type restrictions **Vulnerable Code**

```
this.uploader.queue[0].upload()
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the access_control issue in frontend/src/app/complaint/complaint.component.ts **References**

& Further Reading:

<https://owasp.org/Top10/A04/>

#1135: XSS - Angular bypassSecurityTrustHtml

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-79	OWASP: A03:2021 - Injection
Category:	xss	Confidence: MEDIUM
Location:	frontend/src/app/data-export/data-export.component.ts:55	
Line(s):	55	
Exploitability:	Unknown	Impact: Unknown

Description

Detects Angular DomSanitizer security bypass **Vulnerable Code**

```
this.captcha = this.sanitizer.bypassSecurityTrustHtml(data.image)
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the xss issue in frontend/src/app/data-export/data-export.component.ts

References & Further Reading:

<https://owasp.org/Top10/A03/>

#1136: XSS - Angular bypassSecurityTrustHtml

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-79	OWASP: A03:2021 - Injection
Category:	xss	Confidence: MEDIUM
Location:	frontend/src/app/data-export/data-export.component.ts:55	
Line(s):	55	
Exploitability:	Unknown	Impact: Unknown

Description

Detects Angular DomSanitizer security bypass **Vulnerable Code**

```
this.captcha = this.sanitizer.bypassSecurityTrustHtml(data.image)
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the XSS issue in frontend/src/app/data-export/data-export.component.ts **References & Further Reading:**

<https://owasp.org/Top10/A03/>

#1137: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798 OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication Confidence: MEDIUM
Location:	frontend/src/app/register/register.component.spec.ts:151
Line(s):	151
Exploitability:	Unknown Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
const user = { email: 'x@x.xx', password: 'password', passwordRepeat: 'password',
  securityQuestion: { id: 1, question: 'Wat is?' }, securityAnswer: 'Answer' }
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in frontend/src/app/register/register.component.spec.ts **References**

& Further Reading:

<https://owasp.org/Top10/A07/>

#1138: Hardcoded Credentials - Default Passwords

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	frontend/src/app/register/register.component.spec.ts:151	
Line(s):	151	
Exploitability:	Unknown	Impact: Unknown

Description

Detects common default passwords in code **Vulnerable Code**

```
const user = { email: 'x@x.xx', password: 'password', passwordRepeat: 'password',
  securityQuestion: { id: 1, question: 'Wat is?' }, securityAnswer: 'Answer' }
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in frontend/src/app/register/register.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#1139: Horizontal Access Control - Tenant Isolation Bypass

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-566 OWASP: A01:2021 - Broken Access Control
Category:	access_control Confidence: MEDIUM
Location:	frontend/src/app/product-details/product-details.component.spec.ts:92
Line(s):	92
Exploitability:	Unknown Impact: Unknown

Description

Detects multi-tenant isolation bypass **Vulnerable Code**

```
const textArea: HTMLTextAreaElement = fixture.debugElement.query(By.css('textare')).nativeElement
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the access_control issue in frontend/src/app/product-details/product-details.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

#1140: Horizontal Access Control - Tenant Isolation Bypass

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-566 OWASP: A01:2021 - Broken Access Control
Category:	access_control Confidence: MEDIUM
Location:	frontend/src/app/product-details/product-details.component.spec.ts:94
Line(s):	94
Exploitability:	Unknown Impact: Unknown

Description

Detects multi-tenant isolation bypass **Vulnerable Code**

```
const buttonDe = fixture.debugElement.query(By.css('#submitButton'))
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the access_control issue in frontend/src/app/product-details/product-details.component.spec.ts

References & Further Reading:

<https://owasp.org/Top10/A01/>

#1141: Horizontal Access Control - Tenant Isolation Bypass

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-566 OWASP: A01:2021 - Broken Access Control
Category:	access_control Confidence: MEDIUM
Location:	frontend/src/app/product-details/product-details.component.spec.ts:106
Line(s):	106
Exploitability:	Unknown Impact: Unknown

Description

Detects multi-tenant isolation bypass **Vulnerable Code**

```
const textArea: HTMLTextAreaElement = fixture.debugElement.query(By.css('textarea')).nativeElement
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the access_control issue in frontend/src/app/product-details/product-details.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

#1142: Horizontal Access Control - Tenant Isolation Bypass

Severity: CRITICAL (CVSS: 9.5/10.0)

CWE ID: CWE-566 **OWASP:** A01:2021 - Broken Access Control

Category: access_control **Confidence:** MEDIUM

Location: frontend/src/app/product-details/product-details.component.spec.ts:108

Line(s): 108

Exploitability: Unknown **Impact:** Unknown

Description

Detects multi-tenant isolation bypass **Vulnerable Code**

```
const buttonDe = fixture.debugElement.query(By.css('#submitButton'))
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the access_control issue in frontend/src/app/product-details/product-details.component.spec.ts

References & Further Reading:

<https://owasp.org/Top10/A01/>

#1143: Horizontal Access Control - Tenant Isolation Bypass

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-566	OWASP: A01:2021 - Broken Access Control
Category:	access_control	Confidence: MEDIUM
Location:	frontend/src/app/product-details/product-details.component.spec.ts:130	
Line(s):	130	
Exploitability:	Unknown	Impact: Unknown

Description

Detects multi-tenant isolation bypass

```
const textArea: HTMLTextAreaElement = fixture.debugElement.query(By.css('textare')) .nativeElement
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the access_control issue in frontend/src/app/product-details/product-

details.component.spec.ts**References & Further Reading:**

<https://owasp.org/Top10/A01/>

#1144: Horizontal Access Control - Tenant Isolation Bypass

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-566
Category:	access_control
Location:	OWASP: A01:2021 - Broken Access Control
Line(s):	frontend/src/app/product-details/product-details.component.spec.ts:132
Exploitability:	Unknown
Impact:	Unknown

Description

Detects multi-tenant isolation bypass**Vulnerable Code**

```
const buttonDe = fixture.debugElement.query(By.css('#submitButton'))
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the access_control issue in frontend/src/app/product-details/product-details.component.spec.ts**References & Further Reading:**

<https://owasp.org/Top10/A01/>

#1145: Horizontal Access Control - Tenant Isolation Bypass

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-566 OWASP: A01:2021 - Broken Access Control
Category:	access_control Confidence: MEDIUM
Location:	frontend/src/app/product-details/product-details.component.spec.ts:145
Line(s):	145
Exploitability:	Unknown Impact: Unknown

Description

Detects multi-tenant isolation bypass **Vulnerable Code**

```
const textArea: HTMLTextAreaElement = fixture.debugElement.query(By.css('textare')).nativeElement
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the access_control issue in frontend/src/app/product-details/product-details.component.spec.ts

References & Further Reading:

<https://owasp.org/Top10/A01/>

#1146: Horizontal Access Control - Tenant Isolation Bypass

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-566 OWASP: A01:2021 - Broken Access Control
Category:	access_control Confidence: MEDIUM
Location:	frontend/src/app/product-details/product-details.component.spec.ts:147
Line(s):	147
Exploitability:	Unknown Impact: Unknown

Description

Detects multi-tenant isolation bypass **Vulnerable Code**

```
const buttonDe = fixture.debugElement.query(By.css('#submitButton'))
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the access_control issue in frontend/src/app/product-details/product-details.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

#1147: Horizontal Access Control - Tenant Isolation Bypass

Severity: CRITICAL (CVSS: 9.5/10.0)

CWE ID: CWE-566 **OWASP:** A01:2021 - Broken Access Control

Category: access_control **Confidence:** MEDIUM

Location: frontend/src/app/product-details/product-details.component.spec.ts:159

Line(s): 159

Exploitability: Unknown **Impact:** Unknown

Description

Detects multi-tenant isolation bypass **Vulnerable Code**

```
const buttonDe = fixture.debugElement.query(By.css('div.review-text'))
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the access_control issue in frontend/src/app/product-details/product-details.component.spec.ts

References & Further Reading:

<https://owasp.org/Top10/A01/>

#1148: Horizontal Access Control - Tenant Isolation Bypass

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-566	OWASP: A01:2021 - Broken Access Control
Category:	access_control	Confidence: MEDIUM
Location:	frontend/src/app/product-details/product-details.component.spec.ts:172	
Line(s):	172	
Exploitability:	Unknown	Impact: Unknown

Description

Detects multi-tenant isolation bypass

```
const buttonDe = fixture.debugElement.query(By.css('div.review-text'))
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the access_control issue in frontend/src/app/product-details/product-

details.component.spec.ts**References & Further Reading:**

<https://owasp.org/Top10/A01/>

#1149: XSS - Angular bypassSecurityTrustHtml

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-79
Category:	xss
Location:	frontend/src/app/score-board/score-board.component.ts:79
Line(s):	79
Exploitability:	Unknown
	Impact: Unknown

Description

Detects Angular DomSanitizer security bypass**Vulnerable Code**

```
description: this.sanitizer.bypassSecurityTrustHtml(challenge.description as string)
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the xss issue in frontend/src/app/score-board/score-board.component.ts**References & Further Reading:**

<https://owasp.org/Top10/A03/>

#1150: XSS - Angular bypassSecurityTrustHtml

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-79	OWASP: A03:2021 - Injection
Category:	xss	Confidence: MEDIUM
Location:	frontend/src/app/score-board/score-board.component.ts:79	
Line(s):	79	
Exploitability:	Unknown	Impact: Unknown

Description

Detects Angular DomSanitizer security bypass **Vulnerable Code**

```
description: this.sanitizer.bypassSecurityTrustHtml(challenge.description as string)
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the xss issue in frontend/src/app/score-board/score-board.component.ts

References & Further Reading:

<https://owasp.org/Top10/A03/>

#1151: XSS - Angular bypassSecurityTrustHtml

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-79	OWASP: A03:2021 - Injection
Category:	xss	Confidence: MEDIUM
Location:	frontend/src/app/search-result/search-result.component.spec.ts:87	
Line(s):	87	
Exploitability:	Unknown	Impact: Unknown

Description

Detects Angular DomSanitizer security bypass **Vulnerable Code**

```
sanitizer = jasmine.createSpyObj('DomSanitizer', ['bypassSecurityTrustHtml', 'sanitize'])
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the XSS issue in frontend/src/app/search-result/search-result.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A03/>

#1152: XSS - Angular bypassSecurityTrustHtml

Severity: CRITICAL (CVSS: 9.5/10.0)

CWE ID: CWE-79 **OWASP:** A03:2021 - Injection

Category: XSS **Confidence:** MEDIUM

Location: frontend/src/app/search-result/search-result.component.spec.ts:88

Line(s): 88

Exploitability: Unknown **Impact:** Unknown

Description

Detects Angular DomSanitizer security bypass **Vulnerable Code**

```
sanitizer.bypassSecurityTrustHtml.and.returnValue(of({}))
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the XSS issue in frontend/src/app/search-result/search-result.component.spec.ts

References & Further Reading:

<https://owasp.org/Top10/A03/>

#1153: XSS - Angular bypassSecurityTrustHtml

Severity: CRITICAL (CVSS: 9.5/10.0)

CWE ID: CWE-79 **OWASP:** A03:2021 - Injection

Category: XSS **Confidence:** MEDIUM

Location: frontend/src/app/search-result/search-result.component.spec.ts:141

Line(s): 141

Exploitability: Unknown **Impact:** Unknown

Description

Detects Angular DomSanitizer security bypass

```
expect(sanitizer.bypassSecurityTrustHtml).toHaveBeenCalledWith('<script>alert("XSS")</script>')
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the XSS issue in frontend/src/app/search-result/search-

result.component.spec.ts**References & Further Reading:**

<https://owasp.org/Top10/A03/>

#1154: XSS - Angular bypassSecurityTrustHtml

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-79 OWASP: A03:2021 - Injection
Category:	xss Confidence: MEDIUM
Location:	frontend/src/app/search-result/search-result.component.spec.ts:191
Line(s):	191
Exploitability:	Unknown Impact: Unknown

Description

Detects Angular DomSanitizer security bypass**Vulnerable Code**

```
expect(sanitizer.bypassSecurityTrustHtml).toHaveBeenCalledWith('<script>scripttag</script>')
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the xss issue in frontend/src/app/search-result/search-result.component.spec.ts**References & Further Reading:**

<https://owasp.org/Top10/A03/>

#1155: XSS - Angular bypassSecurityTrustHtml

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-79 OWASP: A03:2021 - Injection
Category:	xss Confidence: MEDIUM
Location:	frontend/src/app/search-result/search-result.component.ts:133
Line(s):	133
Exploitability:	Unknown
	Impact: Unknown

Description

Detects Angular DomSanitizer security bypass **Vulnerable Code**

```
tableData[i].description = this.sanitizer.bypassSecurityTrustHtml(tableData[i].description)
// vuln-code-snippet vuln-line restfulXssChallenge
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the XSS issue in frontend/src/app/search-result/search-result.component.ts **References & Further Reading:**

<https://owasp.org/Top10/A03/>

#1156: XSS - Angular bypassSecurityTrustHtml

Severity:	CRITICAL	(CVSS: 9.5/10.0)
CWE ID:	CWE-79	OWASP: A03:2021 - Injection
Category:	xss	Confidence: MEDIUM
Location:	frontend/src/app/search-result/search-result.component.ts:159	
Line(s):	159	
Exploitability:	Unknown	Impact: Unknown

Description

Detects Angular DomSanitizer security bypass **Vulnerable Code**

```
this.searchValue = this.sanitizer.bypassSecurityTrustHtml(queryParam) // vuln-code-snippet  
vuln-line localXssChallenge XSSBonusChallenge
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the XSS issue in frontend/src/app/search-result/search-result.component.ts **References & Further Reading:**

<https://owasp.org/Top10/A03/>

#1157: XSS - Angular bypassSecurityTrustHtml

Severity:	CRITICAL	(CVSS: 9.5/10.0)
CWE ID:	CWE-79	OWASP: A03:2021 - Injection
Category:	xss	Confidence: MEDIUM
Location:	frontend/src/app/search-result/search-result.component.ts:133	
Line(s):	133	
Exploitability:	Unknown	Impact: Unknown

Description

Detects Angular DomSanitizer security bypass **Vulnerable Code**

```
tableData[i].description = this.sanitizer.bypassSecurityTrustHtml(tableData[i].description)
// vuln-code-snippet vuln-line restfulXssChallenge
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the xss issue in frontend/src/app/search-result/search-result.component.ts **References & Further Reading:**

<https://owasp.org/Top10/A03/>

#1158: XSS - Angular bypassSecurityTrustHtml

Severity:	CRITICAL	(CVSS: 9.5/10.0)
CWE ID:	CWE-79	OWASP: A03:2021 - Injection
Category:	xss	Confidence: MEDIUM
Location:	frontend/src/app/search-result/search-result.component.ts:159	
Line(s):	159	
Exploitability:	Unknown	Impact: Unknown

Description

Detects Angular DomSanitizer security bypass **Vulnerable Code**

```
this.searchValue = this.sanitizer.bypassSecurityTrustHtml(queryParam) // vuln-code-snippet  
vuln-line localXssChallenge XSSBonusChallenge
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the XSS issue in frontend/src/app/search-result/search-result.component.ts **References & Further Reading:**

<https://owasp.org/Top10/A03/>

#1159: CSRF - Password Change Without CSRF

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-352
Category:	csrf
Location:	frontend/src/app/Services/user.service.spec.ts:90
Line(s):	90
Exploitability:	Unknown
Impact:	Unknown

Description

Detects password change without CSRF protection [Vulnerable Code](#)

```
service.changePassword({ current: 'foo', new: 'bar', repeat: 'bar' }).subscribe((data) =>
(res = data))
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the csrf issue in frontend/src/app/Services/user.service.spec.ts [References &](#)

Further Reading:

<https://owasp.org/Top10/A07/>

#1160: XSS - Angular bypassSecurityTrustHtml

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-79	OWASP: A03:2021 - Injection
Category:	xss	Confidence: MEDIUM
Location:	frontend/src/app/track-result/track-result.component.spec.ts:27	
Line(s):	27	
Exploitability:	Unknown	Impact: Unknown

Description

Detects Angular DomSanitizer security bypass **Vulnerable Code**

```
sanitizer = jasmine.createSpyObj('DomSanitizer', ['bypassSecurityTrustHtml', 'sanitize'])
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the xss issue in frontend/src/app/track-result/track-result.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A03/>

#1161: XSS - Angular bypassSecurityTrustHtml

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-79	OWASP: A03:2021 - Injection
Category:	xss	Confidence: MEDIUM
Location:	frontend/src/app/track-result/track-result.component.spec.ts:28	
Line(s):	28	
Exploitability:	Unknown	Impact: Unknown

Description

Detects Angular DomSanitizer security bypass **Vulnerable Code**

```
sanitizer.bypassSecurityTrustHtml.and.callFake((args: any) => args)
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the XSS issue in frontend/src/app/track-result/track-result.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A03/>

#1162: XSS - Angular bypassSecurityTrustHtml

Severity: CRITICAL (CVSS: 9.5/10.0)

CWE ID: CWE-79 **OWASP:** A03:2021 - Injection

Category: XSS **Confidence:** MEDIUM

Location: frontend/src/app/track-result/track-result.component.spec.ts:62

Line(s): 62

Exploitability: Unknown **Impact:** Unknown

Description

Detects Angular DomSanitizer security bypass **Vulnerable Code**

```
expect(sanitizer.bypassSecurityTrustHtml).toHaveBeenCalledWith('<code><a  
src="link">Link</a></code>')
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the xss issue in frontend/src/app/track-result/track-result.component.spec.ts

References & Further Reading:

<https://owasp.org/Top10/A03/>

#1163: API - Balance/Credits Field in Request

Severity: CRITICAL (CVSS: 9.5/10.0)

CWE ID: CWE-915 **OWASP:** A01:2021 - Broken Access Control

Category: api_security **Confidence:** MEDIUM

Location: routes/nftMint.ts:36

Line(s): 36

Exploitability: Unknown **Impact:** Unknown

Description

Detects balance/credits manipulation via API

```
const metamaskAddress = req.body.walletAddress
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the api_security issue in routes/nftMint.ts

References & Further Reading:

<https://owasp.org/Top10/A01/>

#1164: XSS - Angular bypassSecurityTrustHtml

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-79	OWASP: A03:2021 - Injection
Category:	xss	Confidence: MEDIUM
Location:	frontend/src/app/track-result/track-result.component.ts:47	
Line(s):	47	
Exploitability:	Unknown	Impact: Unknown

Description

Detects Angular DomSanitizer security bypass **Vulnerable Code**

```
this.results.orderNo = this.sanitizer.bypassSecurityTrustHtml('<code>${results.data[0].orderId}</code>')
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the xss issue in frontend/src/app/track-result/track-result.component.ts **References & Further Reading:**

<https://owasp.org/Top10/A03/>

#1165: XSS - Angular bypassSecurityTrustHtml

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-79	OWASP: A03:2021 - Injection
Category:	xss	Confidence: MEDIUM
Location:	frontend/src/app/track-result/track-result.component.ts:47	
Line(s):	47	
Exploitability:	Unknown	Impact: Unknown

Description

Detects Angular DomSanitizer security bypass **Vulnerable Code**

```
this.results.orderNo = this.sanitizer.bypassSecurityTrustHtml('<code>${results.data[0].orderId}</code>')
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the xss issue in frontend/src/app/track-result/track-result.component.ts

References & Further Reading:

<https://owasp.org/Top10/A03/>

#1166: CSRF - Password Change Without CSRF

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-352	OWASP: A07:2021 - Identification and Authentication Failures
Category:	csrf	Confidence: MEDIUM
Location:	models/challenge.ts:30	
Line(s):	30	
Exploitability:	Unknown	Impact: Unknown

Description

Detects password change without CSRF protection **Vulnerable Code**

```
'changePasswordBenderChallenge',
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the csrf issue in models/challenge.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#1167: IDOR - UPDATE Without Authorization

Severity: CRITICAL (CVSS: 9.5/10.0)

CWE ID: CWE-639 **OWASP:** A01:2021 - Broken Access Control

Category: access_control **Confidence:** MEDIUM

Location: routes/chatbot.ts:141

Line(s): 141

Exploitability: Unknown **Impact:** Unknown

Description

Detects UPDATE operations without authorization **Vulnerable Code**

```
const updatedUser = await userModel.update({ username: req.body.query })
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the access_control issue in routes/chatbot.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

#1168: Horizontal Access Control - Tenant Isolation Bypass

Severity: CRITICAL (CVSS: 9.5/10.0)

CWE ID: CWE-566 **OWASP:** A01:2021 - Broken Access Control

Category: access_control **Confidence:** MEDIUM

Location: routes/search.ts:23

Line(s): 23

Exploitability: Unknown **Impact:** Unknown

Description

Detects multi-tenant isolation bypass **Vulnerable Code**

```
models.sequelize.query('SELECT * FROM Products WHERE ((name LIKE \'%${criteria}%' OR description LIKE \'%${criteria}\%') AND deletedAt IS NULL) ORDER BY name') //  
vuln-code-snippet vuln-line unionSqlInjectionChallenge dbSchemaChallenge
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the access_control issue in routes/search.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

#1169: Horizontal Access Control - Tenant Isolation Bypass

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-566	OWASP: A01:2021 - Broken Access Control
Category:	access_control	Confidence: MEDIUM
Location:	routes/search.ts:47	
Line(s):	47	
Exploitability:	Unknown	Impact: Unknown

Description

Detects multi-tenant isolation bypass **Vulnerable Code**

```
void models.sequelize.query('SELECT sql FROM sqlite_master').then(([data]: any) => {
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the access_control issue in routes/search.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

#1170: API - Balance/Credits Field in Request

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-915 OWASP: A01:2021 - Broken Access Control
Category:	api_security Confidence: MEDIUM
Location:	routes/wallet.ts:26
Line(s):	26
Exploitability:	Unknown Impact: Unknown

Description

Detects balance/credits manipulation via API **Vulnerable Code**

```
WalletModel.increment({ balance: req.body.balance }, { where: { UserId: req.body.UserId } })
  .then(() => {
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the api_security issue in routes/wallet.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

#1171: API - Balance/Credits Field in Request

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-915 OWASP: A01:2021 - Broken Access Control
Category:	api_security Confidence: MEDIUM
Location:	routes/wallet.ts:27
Line(s):	27
Exploitability:	Unknown Impact: Unknown

Description

Detects balance/credits manipulation via API **Vulnerable Code**

```
res.status(200).json({ status: 'success', data: req.body.balance })
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the api_security issue in routes/wallet.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

#1172: API - Balance/Credits Field in Request

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-915 OWASP: A01:2021 - Broken Access Control
Category:	api_security Confidence: MEDIUM
Location:	routes/web3Wallet.ts:15
Line(s):	15
Exploitability:	Unknown Impact: Unknown

Description

Detects balance/credits manipulation via API **Vulnerable Code**

```
const metamaskAddress = req.body.walletAddress
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the api_security issue in routes/web3Wallet.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

#1173: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/cypress/e2e/b2bOrder.spec.ts:6	
Line(s):	6	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
cy.login({ email: 'admin', password: 'admin123' })
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/cypress/e2e/b2bOrder.spec.ts **References &**

Further Reading:

<https://owasp.org/Top10/A07/>

#1174: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/cypress/e2e/b2bOrder.spec.ts:37	
Line(s):	37	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings

Vulnerable Code

```
cy.login({ email: 'admin', password: 'admin123' })
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/cypress/e2e/b2bOrder.spec.ts

References &

Further Reading:

<https://owasp.org/Top10/A07/>

#1175: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/cypress/e2e/basket.spec.ts:4	
Line(s):	4	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
cy.login({ email: 'admin', password: 'admin123' })
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in test/cypress/e2e/basket.spec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1176: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/cypress/e2e/basket.spec.ts:76	
Line(s):	76	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
cy.login({ email: 'jim', password: 'ncc-1701' })
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/cypress/e2e/basket.spec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1177: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798 OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication Confidence: MEDIUM
Location:	test/cypress/e2e/chatbot.spec.ts:3
Line(s):	3
Exploitability:	Unknown Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
cy.login({ email: 'admin', password: 'admin123' })
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/cypress/e2e/chatbot.spec.ts **References &**

Further Reading:

<https://owasp.org/Top10/A07/>

#1178: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/cypress/e2e/complain.spec.ts:5	
Line(s):	5	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'admin123'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/cypress/e2e/complain.spec.ts **References &**

Further Reading:

<https://owasp.org/Top10/A07/>

#1179: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798 OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication Confidence: MEDIUM
Location:	test/cypress/e2e/contact.spec.ts:11
Line(s):	11
Exploitability:	Unknown Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
cy.login({ email: 'admin', password: 'admin123' })
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/cypress/e2e/contact.spec.ts **References &**

Further Reading:

<https://owasp.org/Top10/A07/>

#1180: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/cypress/e2e/contact.spec.ts:47	
Line(s):	47	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
cy.login({ email: 'admin', password: 'admin123' })
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in test/cypress/e2e/contact.spec.ts **References &**

Further Reading:

<https://owasp.org/Top10/A07/>

#1181: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/cypress/e2e/dataErasure.spec.ts:3	
Line(s):	3	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
cy.login({ email: 'admin', password: 'admin123' })
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/cypress/e2e/dataErasure.spec.ts **References &**

Further Reading:

<https://owasp.org/Top10/A07/>

#1182: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/cypress/e2e/dataExport.spec.ts:24
Line(s):	24
Exploitability:	Unknown
	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
cy.login({ email: 'admun', password: 'admun123' })
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/cypress/e2e/dataExport.spec.ts **References &**

Further Reading:

<https://owasp.org/Top10/A07/>

#1183: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/cypress/e2e/deluxe.spec.ts:4	
Line(s):	4	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
cy.login({ email: 'jim', password: 'ncc-1701' })
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/cypress/e2e/deluxe.spec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1184: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798 OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication Confidence: MEDIUM
Location:	test/cypress/e2e/deluxe.spec.ts:21
Line(s):	21
Exploitability:	Unknown Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'ncc-1701'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/cypress/e2e/deluxe.spec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1185: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798 OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication Confidence: MEDIUM
Location:	test/cypress/e2e/noSql.spec.ts:8
Line(s):	8
Exploitability:	Unknown Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
cy.login({ email: 'admin', password: 'admin123' })
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in test/cypress/e2e/noSql.spec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1186: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798 OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication Confidence: MEDIUM
Location:	test/cypress/e2e/noSql.spec.ts:53
Line(s):	53
Exploitability:	Unknown Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
cy.login({ email: 'admin', password: 'admin123' })
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/cypress/e2e/noSql.spec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1187: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/cypress/e2e/noSql.spec.ts:76
Line(s):	76
Exploitability:	Unknown
	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
cy.login({ email: 'mc.safesearch', password: 'Mr. N00dles' })
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/cypress/e2e/noSql.spec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1188: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/cypress/e2e/noSql.spec.ts:120	
Line(s):	120	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
cy.login({ email: 'mc.safesearch', password: 'Mr. N00dles' })
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/cypress/e2e/noSql.spec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1189: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798 OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication Confidence: MEDIUM
Location:	test/cypress/e2e/profile.spec.ts:3
Line(s):	3
Exploitability:	Unknown Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
cy.login({ email: 'admin', password: 'admin123' })
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/cypress/e2e/profile.spec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1190: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/cypress/e2e/register.spec.ts:10
Line(s):	10
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'admin123'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in test/cypress/e2e/register.spec.ts **References &**

Further Reading:

<https://owasp.org/Top10/A07/>

#1191: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/cypress/e2e/register.spec.ts:84
Line(s):	84
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'ThereCanBeOnlyOne'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/cypress/e2e/register.spec.ts **References &**

Further Reading:

<https://owasp.org/Top10/A07/>

#1192: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/cypress/e2e/search.spec.ts:56
Line(s):	56
Exploitability:	Unknown
	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'admin123'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/cypress/e2e/search.spec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1193: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	test/cypress/e2e/search.spec.ts:83	
Line(s):	83	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'admin123'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/cypress/e2e/search.spec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A07/>

#1194: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798 OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication Confidence: MEDIUM
Location:	test/cypress/e2e/totpSetup.spec.ts:6
Line(s):	6
Exploitability:	Unknown Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'EinBelegtesBrotMitSchinkenSCHINKEN!',
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in test/cypress/e2e/totpSetup.spec.ts **References &**

Further Reading:

<https://owasp.org/Top10/A07/>

#1195: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	test/cypress/e2e/totpSetup.spec.ts:20
Line(s):	20
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'K1f.....'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in test/cypress/e2e/totpSetup.spec.ts **References &**

Further Reading:

<https://owasp.org/Top10/A07/>

#1196: Crypto - Insecure JWT Algorithm None

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-327
Category:	cryptography
Location:	test/server/verifySpec.ts:260
Line(s):	260
Exploitability:	Unknown
Impact:	Unknown

Description

Detects JWT with "none" algorithm allowed **Vulnerable Code**

```
Header: { "alg": "none", "typ": "JWT" }
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the cryptography issue in test/server/verifySpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A02/>

#1197: Crypto - Insecure JWT Algorithm None

Severity: CRITICAL (CVSS: 9.5/10.0)

CWE ID: CWE-327 **OWASP:** A02:2021 - Cryptographic Failures

Category: cryptography **Confidence:** MEDIUM

Location: test/server/verifySpec.ts:272

Line(s): 272

Exploitability: Unknown **Impact:** Unknown

Description

Detects JWT with "none" algorithm allowed **Vulnerable Code**

```
Header: { "alg": "none", "typ": "JWT" }
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the cryptography issue in test/server/verifySpec.ts **References & Further**

Reading:

<https://owasp.org/Top10/A02/>

#1198: CSRF - Password Change Without CSRF

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-352
Category:	OWASP: A07:2021 - Identification and Authentication Failures
Location:	config.schema.yml:312
Line(s):	312
Exploitability:	Unknown
	Impact: Unknown

Description

Detects password change without CSRF protection **Vulnerable Code**

changePasswordBenderChallenge:

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the csrf issue in config.schema.yml **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#1199: CSRF - Password Change Without CSRF

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-352
Category:	OWASP: A07:2021 - Identification and Authentication Failures
Location:	csrf
Line(s):	config/fbctf.yml:87
Exploitability:	87
Impact:	Unknown

Description

Detects password change without CSRF protection **Vulnerable Code**

changePasswordBenderChallenge:

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the csrf issue in config/fbctf.yml **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#1200: CSRF - Password Change Without CSRF

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-352
Category:	csrf
Location:	data/static/challenges.yml:169
Line(s):	169
Exploitability:	Unknown
Impact:	Unknown

Description

Detects password change without CSRF protection **Vulnerable Code**

```
key: changePasswordBenderChallenge
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the csrf issue in data/static/challenges.yml **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#1201: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	data/static/users.yml:31
Line(s):	31
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'ncc-1701'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in data/static/users.yml **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#1202: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	data/static/users.yml:63	
Line(s):	63	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'OhG0dPlease1nsertLiquor!'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in data/static/users.yml **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#1203: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	data/static/users.yml:88	
Line(s):	88	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'bW9jLmxpYW1nQGhjaW5pbW1pay5ucmVvamI='
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in data/static/users.yml **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#1204: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	data/static/users.yml:107	
Line(s):	107	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings

Vulnerable Code

```
password: 'mDLx?94T~1CfVfZMzw@sJ9f?s3L61bMqE70FfI8^54jbNikY5fyrmx7c!YbJb'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in data/static/users.yml

References & Further Reading:

<https://owasp.org/Top10/A07/>

#1205: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	data/static/users.yml:115
Line(s):	115
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'J6aVjTgOpRs@?51!Zkq2AYnCE@RF$P'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in data/static/users.yml **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#1206: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	data/static/users.yml:123
Line(s):	123
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'focusOnScienceMorty!focusOnScience'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in data/static/users.yml **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#1207: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	data/static/users.yml:132	
Line(s):	132	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'Mr. NOOdles'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data

- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in data/static/users.yml **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#1208: Hardcoded Credentials - Database Password

Severity: CRITICAL (CVSS: 9.5/10.0)

CWE ID: CWE-798 **OWASP:** A07:2021 - Identification and Authentication Failures

Category: authentication **Confidence:** MEDIUM

Location: data/static/users.yml:140

Line(s): 140

Exploitability: Unknown **Impact:** Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: '0Y8rMnww$*9VFYE 59-!Fg1L6t&61B'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in data/static/users.yml **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#1209: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	data/static/users.yml:149
Line(s):	149
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'EinBelegtesBrotMitSchinkenSCHINKEN!'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in data/static/users.yml **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#1210: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	data/static/users.yml:158
Line(s):	158
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'K1f.....'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in data/static/users.yml **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#1211: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	data/static/users.yml:166	
Line(s):	166	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'monkey summer birthday are all bad passwords but work just fine in a long  
passphrase'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in data/static/users.yml **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#1212: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	data/static/users.yml:176	
Line(s):	176	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'kitten lesser pooch karate buffoonindoors'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in data/static/users.yml **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#1213: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	data/static/users.yml:186	
Line(s):	186	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings

Vulnerable Code

```
password: 'uss enterprise'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in data/static/users.yml

References & Further Reading:

<https://owasp.org/Top10/A07/>

#1214: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	data/static/users.yml:195
Line(s):	195
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'i am an awesome accountant'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in data/static/users.yml **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#1215: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	data/static/users.yml:204
Line(s):	204
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'muda-muda > ora-ora' # https://www.youtube.com/watch?v=vnJv8IoLMwc
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in data/static/users.yml **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#1216: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	data/static/users.yml:237	
Line(s):	237	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'y&x5Z#f6W532Z4445#Ae2HkwZVyDb7&oCUaDzFU'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data

- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in data/static/users.yml **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#1217: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	data/static/users.yml:243	
Line(s):	243	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'y&x5Z#f6W532ZUf$q3DsdgfgfgxxUsvoCUaDzFU'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in data/static/users.yml **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#1218: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	data/static/users.yml:249
Line(s):	249
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'ship coffin krypt cross estate supply insurance asbestos souvenir'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability
Remediation Guidance

Review and fix the authentication issue in data/static/users.yml **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#1219: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)
CWE ID:	CWE-798
Category:	authentication
Location:	data/static/users.yml:270
Line(s):	270
Exploitability:	Unknown
Impact:	Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
password: 'IamUsedForTesting'
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

Review and fix the authentication issue in data/static/users.yml **References & Further Reading:**

<https://owasp.org/Top10/A07/>

#1220: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	data/static/codefixes/loginAdminChallenge_2.ts:17	
Line(s):	17	
Exploitability:	Unknown	Impact: Unknown

Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
models.sequelize.query('SELECT * FROM Users WHERE email = $1 AND password = '${security.hash(req.body.password || "")}' AND deletedAt IS NULL',
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

```
“json
{
  “explanation”: “The code contains two critical vulnerabilities:\n\n1. **Hardcoded Database Credentials (Line 17)**: The database connection appears to use hardcoded credentials in the ‘models.sequelize.query()’ call. While the exact password isn’t visible in this snippet, the pattern suggests credentials are embedded in the codebase. This violates the principle of separating configuration from code, making credentials accessible to anyone with code access, including version control”}
```

Suggested Fix (Copy & Apply)

```
history. If compromised, attackers gain direct database access.\n\n2. **Horizontal Access Control / Tenant Isolation Bypass**: The SQL query at line 17 lacks tenant isolation controls. When executed, it retrieves users based only on email and password without verifying the requesting user’s authorization to access that specific tenant’s data. In multi-tenant applications, this allows users to access other tenants’ data by manipulating email parameters, leading to data leakage between organizati
```

References & Further Reading:

<https://owasp.org/Top10/A07/>

#1221: Horizontal Access Control - Tenant Isolation Bypass

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-566	OWASP: A01:2021 - Broken Access Control
Category:	access_control	Confidence: MEDIUM
Location:	data/static/codedefixes/loginAdminChallenge_2.ts:17	
Line(s):	17	
Exploitability:	Unknown	Impact: Unknown

Description

Detects multi-tenant isolation bypass

Vulnerable Code

```
models.sequelize.query('SELECT * FROM Users WHERE email = $1 AND password = ${security.hash(req.body.password || "")} AND deletedAt IS NULL',
```

Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
 - Execute arbitrary code on the server
 - Compromise the entire application infrastructure
 - Exfiltrate confidential information

Business Consequences: Data breach, regulatory fines, reputational damage, legal liability

Remediation Guidance

```
```json
{
 "explanation": "The code contains two critical vulnerabilities:\n\n**Hardcoded Database Credentials (Line 17)**: The database connection appears to use hardcoded credentials in the 'models.sequelize.query()' call. While the exact password isn't visible in this snippet, the pattern suggests credentials are embedded in the codebase. This violates the principle of separating configuration from code, making credentials accessible to anyone with code access, including version control"
}
```

## Suggested Fix (Copy & Apply)

history. If compromised, attackers gain direct database access.\n\n**2. \*\*Horizontal Access Control / Tenant Isolation Bypass\*\*:** The SQL query at line 17 lacks tenant isolation controls. When executed, it retrieves users based only on email and password without verifying the requesting user's authorization to access that specific tenant's data. In multi-tenant applications, this allows users to access other tenants' data by manipulating email parameters, leading to data leakage between organizations.

## **References & Further Reading:**

<https://owasp.org/Top10/A01/>

## #1222: Hardcoded Credentials - Database Password

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-798	OWASP: A07:2021 - Identification and Authentication Failures
Category:	authentication	Confidence: MEDIUM
Location:	data/static/codefixes/loginBenderChallenge_1.ts:20	
Line(s):	20	
Exploitability:	Unknown	Impact: Unknown

### Description

Detects hardcoded database passwords in connection strings

### Vulnerable Code

```
models.sequelize.query('SELECT * FROM Users WHERE email = '${req.body.email || ''} AND password = '${security.hash(req.body.password || '')}' AND deletedAt IS NULL', { model: models.User, plain: true })
```

### Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

**Business Consequences:** Data breach, regulatory fines, reputational damage, legal liability  
**Remediation Guidance**

```
"json
{
 "explanation": "The code contains two critical vulnerabilities:\\n\\n1. **Horizontal Access Control - Tenant Isolation Bypass (Line 20)**: The SQL query at line 20 uses string concatenation with user input ('req.body.email' and 'req.body.password') without proper parameterization. This creates SQL injection vulnerabilities that could allow attackers to bypass authentication, access other users' data, or execute arbitrary SQL commands. The regex-based blacklist filter at line 18 is in
}
```

### Suggested Fix (Copy & Apply)

```
sufficient and can be bypassed with various SQL injection techniques.\\n\\n2. **Hardcoded Credentials - Database Password (Line 20)**: While not literally showing a hardcoded password in this snippet, the pattern of constructing SQL queries with string concatenation suggests the application may be vulnerable to credential exposure through SQL injection. Additionally, the code shows password hashing being applied directly in the SQL query, which could expose password hashes if the query is vulnerab
```

**References & Further Reading:**

<https://owasp.org/Top10/A07/>

**#1223: Horizontal Access Control - Tenant Isolation Bypass**

<b>Severity:</b>	<b>CRITICAL</b> (CVSS: 9.5/10.0)	
<b>CWE ID:</b>	CWE-566	<b>OWASP:</b> A01:2021 - Broken Access Control
<b>Category:</b>	access_control	<b>Confidence:</b> MEDIUM
<b>Location:</b>	data/static/codefixes/loginBenderChallenge_1.ts:20	
<b>Line(s):</b>	20	
<b>Exploitability:</b>	<a href="#">Unknown</a>	<b>Impact:</b> Unknown

**Description**

Detects multi-tenant isolation bypass **Vulnerable Code**

```
models.sequelize.query('SELECT * FROM Users WHERE email = '${req.body.email || ''} AND password = '${security.hash(req.body.password || '')}' AND deletedAt IS NULL', { model: models.User, plain: true })
```

**Security Impact**

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

**Business Consequences:** Data breach, regulatory fines, reputational damage, legal liability

**Remediation Guidance**

```
“json
{
 “explanation”: “The code contains two critical vulnerabilities:\\n\\n1. **Horizontal Access Control - Tenant Isolation Bypass (Line 20)**: The SQL query at line 20 uses string concatenation with user input (‘req.body.email’ and ‘req.body.password’) without proper parameterization. This creates SQL injection vulnerabilities that could allow attackers to bypass authentication, access other users’ data, or execute arbitrary SQL commands. The regex-based blacklist filter at line 18 is in”}
```

### Suggested Fix (Copy & Apply)

sufficient and can be bypassed with various SQL injection techniques.\n\n\*\*Hardcoded Credentials - Database Password (Line 20)\*\*: While not literally showing a hardcoded password in this snippet, the pattern of constructing SQL queries with string concatenation suggests the application may be vulnerable to credential exposure through SQL injection. Additionally, the code shows password hashing being applied directly in the SQL query, which could expose password hashes if the query is vulnerab

### References & Further Reading:

<https://owasp.org/Top10/A01/>

## #1224: Hardcoded Credentials - Database Password

<b>Severity:</b>	<b>CRITICAL</b> (CVSS: 9.5/10.0)	
<b>CWE ID:</b>	CWE-798	<b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	authentication	<b>Confidence:</b> MEDIUM
<b>Location:</b>	data/static/codefixes/loginAdminChallenge_1.ts:20	
<b>Line(s):</b>	20	
<b>Exploitability:</b>	<a href="#">Unknown</a>	<b>Impact:</b> Unknown

### Description

Detects hardcoded database passwords in connection strings **Vulnerable Code**

```
models.sequelize.query('SELECT * FROM Users WHERE email = '${req.body.email || ''} AND password = '${security.hash(req.body.password || '')}', { model: models.User, plain: true })
```

### Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

**Business Consequences:** Data breach, regulatory fines, reputational damage, legal liability  
**Remediation Guidance**

```
“json
{
```

"explanation": "The code contains two critical vulnerabilities:\n\n1. \*\*Hardcoded Database Credentials (Line 20)\*\*: The SQL query directly embeds user input without parameterization, creating SQL injection risk. While there's a basic regex filter, it's insufficient against sophisticated attacks. The real issue is the query construction pattern itself, which could expose database credentials if the application's configuration is compromised.\n\n2. \*\*Horizontal Access Control / Tenant Isolation Bypass\*\*: The query retrieves users based only on email and password without verifying tenant context. In a multi-tenant system, this allows users from one tenant to potentially access data from another tenant if they guess valid credentials. The 'deletedAt IS NULL' check doesn't provide tenant isolation.\n\nAdditionally, there are implicit issues:\n- SQL Injection vulnerability despite the regex filter (incomplete protection)\n- No rate limiting on login attempts\n- Password hashing is present but likely weak or non-compliant."

### Suggested Fix (Copy & Apply)

t Isolation Bypass\*\*: The query retrieves users based only on email and password without verifying tenant context. In a multi-tenant system, this allows users from one tenant to potentially access data from another tenant if they guess valid credentials. The 'deletedAt IS NULL' check doesn't provide tenant isolation.\n\nAdditionally, there are implicit issues:\n- SQL Injection vulnerability despite the regex filter (incomplete protection)\n- No rate limiting on login attempts\n- Password hashing is present but likely weak or non-compliant.

### References & Further Reading:

<https://owasp.org/Top10/A07/>

## #1225: Horizontal Access Control - Tenant Isolation Bypass

Severity:	CRITICAL (CVSS: 9.5/10.0)	
CWE ID:	CWE-566	OWASP: A01:2021 - Broken Access Control
Category:	access_control	Confidence: MEDIUM
Location:	data/static/codefixes/loginAdminChallenge_1.ts:20	
Line(s):	20	
Exploitability:	Unknown	Impact: Unknown

### Description

Detects multi-tenant isolation bypass

```
models.sequelize.query('SELECT * FROM Users WHERE email = '${req.body.email || ''}' AND password = '${security.hash(req.body.password || '')}' AND deletedAt IS NULL', { model: models.User, plain: true })
```

### Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server

- Compromise the entire application infrastructure
- Exfiltrate confidential information

**Business Consequences:** Data breach, regulatory fines, reputational damage, legal liability  
**Remediation Guidance**

```
“json
{
 “explanation”: “The code contains two critical vulnerabilities:
 1. **Hardcoded Database Credentials (Line 20)**: The SQL query directly embeds user input without parameterization, creating SQL injection risk. While there’s a basic regex filter, it’s insufficient against sophisticated attacks. The real issue is the query construction pattern itself, which could expose database credentials if the application’s configuration is compromised.
 2. **Horizontal Access Control / Tenant Isolation Bypass**: The query retrieves users based only on email and password without verifying tenant context. In a multi-tenant system, this allows users from one tenant to potentially access data from another tenant if they guess valid credentials. The ‘deletedAt IS NULL’ check doesn’t provide tenant isolation.
 Additionally, there are implicit issues:
 - SQL Injection vulnerability despite the regex filter (incomplete protection)
 - No rate limiting on login attempts
 - Password hashing
```

### Suggested Fix (Copy & Apply)

```
t Isolation Bypass**:
 The query retrieves users based only on email and password without verifying tenant context. In a multi-tenant system, this allows users from one tenant to potentially access data from another tenant if they guess valid credentials. The ‘deletedAt IS NULL’ check doesn’t provide tenant isolation.
 Additionally, there are implicit issues:
 - SQL Injection vulnerability despite the regex filter (incomplete protection)
 - No rate limiting on login attempts
 - Password hashing
```

### References & Further Reading:

<https://owasp.org/Top10/A01/>

## #1226: Hardcoded Credentials - Database Password

<b>Severity:</b>	<b>CRITICAL</b> (CVSS: 9.5/10.0)
<b>CWE ID:</b>	CWE-798 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	authentication <b>Confidence:</b> MEDIUM
<b>Location:</b>	frontend/src/app/oauth/oauth.component.spec.ts:84
<b>Line(s):</b>	84
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

### Description

Detects hardcoded database passwords in connection strings

### Vulnerable Code

```
expect(userService.save).toHaveBeenCalledWith({ email: ‘test@test.com’, password: ‘bW9jLnRzZXRAdHNldA==’, passwordRepeat: ‘bW9jLnRzZXRAdHNldA==’ })
```

## Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

**Business Consequences:** Data breach, regulatory fines, reputational damage, legal liability

### Remediation Guidance

```
“json
{
 “explanation”: “The code analysis reveals two CRITICAL vulnerabilities related to hardcoded credentials, but upon closer inspection, these are actually false positives in the context of this test file. The ‘passwords’ detected at lines 84 and 91 are not actual database passwords but rather test data representing base64-encoded reversed email addresses used for OAuth user creation testing. However, the analysis tool flagged them because they appear as hardcoded strings that resemble”}
```

### Suggested Fix (Copy & Apply)

credentials. The real security concern here is the pattern of using deterministic password generation from email addresses in the OAuth flow, which could lead to predictable credentials if this logic were implemented in production code. Additionally, the test file demonstrates insecure patterns like storing authentication tokens in localStorage and sessionStorage without proper security considerations.”,

### References & Further Reading:

<https://owasp.org/Top10/A07/>

## #1227: Hardcoded Credentials - Database Password

<b>Severity:</b>	<b>CRITICAL</b> (CVSS: 9.5/10.0)
<b>CWE ID:</b>	CWE-798
<b>Category:</b>	authentication
<b>Location:</b>	frontend/src/app/oauth/oauth.component.spec.ts:91
<b>Line(s):</b>	91
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

### Description

Detects hardcoded database passwords in connection strings

### Vulnerable Code

```
expect(userService.login).toHaveBeenCalledWith({ email: 'test@test.com', password: 'bW9jLnRzZXRAdHNldA==', oauth: true })
```

## Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

**Business Consequences:** Data breach, regulatory fines, reputational damage, legal liability

### Remediation Guidance

```
“json
{
 "explanation": "The code analysis reveals two CRITICAL vulnerabilities related to hardcoded credentials, but upon closer inspection, these are actually false positives in the context of this test file. The 'passwords' detected at lines 84 and 91 are not actual database passwords but rather test data representing base64-encoded reversed email addresses used for OAuth user creation testing. However, the analysis tool flagged them because they appear as hardcoded strings that resemble"
```

### Suggested Fix (Copy & Apply)

```
credentials. The real security concern here is the pattern of using deterministic password generation from email addresses in the OAuth flow, which could lead to predictable credentials if this logic were implemented in production code. Additionally, the test file demonstrates insecure patterns like storing authentication tokens in localStorage and sessionStorage without proper security considerations.",
```

### References & Further Reading:

<https://owasp.org/Top10/A07/>

## #1228: CSRF - Password Change Without CSRF

Severity:	<b>CRITICAL</b> (CVSS: 9.5/10.0)
CWE ID:	CWE-352
Category:	csrf
Location:	frontend/src/app/change-password/change-password.component.html:65
Line(s):	65
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

## Description

Detects password change without CSRF protection **Vulnerable Code**

```
mat-raised-button color="primary" (click)="changePassword()" aria-label="Button to confirm
the change">
```

## Security Impact

This critical vulnerability could allow an attacker to:

- Gain unauthorized access to sensitive systems or data
- Execute arbitrary code on the server
- Compromise the entire application infrastructure
- Exfiltrate confidential information

**Business Consequences:** Data breach, regulatory fines, reputational damage, legal liability

### Remediation Guidance

```
“json
{
 “explanation”: “The vulnerability is a Cross-Site Request Forgery (CSRF) attack vector in the
 password change functionality. The issue occurs because the password change form lacks
 CSRF protection mechanisms. When an authenticated user visits a malicious website, that site
 can forge a request to change the user’s password without their knowledge or consent. This is
 particularly dangerous because: 1) It allows attackers to take over user accounts by changing
 passwords, 2) The attack
```

### Suggested Fix (Copy & Apply)

```
works even if the user is logged into the legitimate application, 3) No user interaction
beyond visiting a malicious page is required, 4) The attack is transparent to the user.
The current implementation relies solely on session authentication without verifying that
the request originated from the legitimate application.”,
```

### References & Further Reading:

<https://owasp.org/Top10/A07/>

## 0.2.2 High Severity Issues (663 found)

### URGENT ATTENTION NEEDED

High severity vulnerabilities pose **significant security risks** including:

- Sensitive data exposure
- Authentication bypass
- Unauthorized access to protected resources
- Injection attacks (SQL, XSS, Command)

**Recommended Timeline:** Fix within **7-14 days**

### #297: CSRF - Missing Origin Validation

**Severity:** **HIGH** (CVSS: **7.5/10.0**)

**CWE ID:** CWE-352      **OWASP:** A01:2021 - Broken Access Control

**Category:** csrf      **Confidence:** **MEDIUM**

**Location:** data/static/codedefixes/changeProductChallenge\_2.ts:64

**Line(s):** 64

**Exploitability:** **Unknown**      **Impact:** Unknown

#### Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post('/api/Addresss', security.appendUserId())
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in data/static/codedefixes/changeProductChallenge\_2.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A01/>

## #298: Vertical Access Control - Missing Role Check

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-862
<b>Category:</b>	access_control
<b>Location:</b>	OWASP: A01:2021 - Broken Access Control
<b>Line(s):</b>	data/static/codefixes/changeProductChallenge_3_correct.ts:32
<b>Exploitability:</b>	32
<b>Impact:</b>	Unknown

### Description

Detects operations without role-based access control

### Vulnerable Code

```
app.put(
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption

**Remediation Guidance**

Review and fix the access\_control issue in data/static/codefixes/changeProductChallenge\_3\_correct.ts

### & Further Reading:

<https://owasp.org/Top10/A01/>

## #299: Vertical Access Control - Missing Role Check

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-862
<b>Category:</b>	access_control
<b>Location:</b>	data/static/codefixes/changeProductChallenge_3_correct.ts:52
<b>Line(s):</b>	52
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects operations without role-based access control [Vulnerable Code](#)

```
app.put('/api/Feedbacks/:id', denyAll())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in data/static/codefixes/changeProductChallenge\_3\_correct.ts [Reference](#)

### & Further Reading:

<https://owasp.org/Top10/A01/>

## #300: Vertical Access Control - Missing Role Check

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-862
<b>Category:</b>	access_control
<b>Location:</b>	data/static/codefixes/changeProductChallenge_3_correct.ts:82
<b>Line(s):</b>	82
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects operations without role-based access control

## Vulnerable Code

```
app.put('/api/Products/:id', denyAll())
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption

## Remediation Guidance

Review and fix the access\_control issue in data/static/codefixes/changeProductChallenge\_3\_correct.ts

## & Further Reading:

<https://owasp.org/Top10/A01/>

## #301: Vertical Access Control - Missing Role Check

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)	
CWE ID:	CWE-862	OWASP: A01:2021 - Broken Access Control
Category:	access_control	Confidence: <b>MEDIUM</b>
Location:	data/static/codefixes/changeProductChallenge_3_correct.ts:121	
Line(s):	121	
Exploitability:	<a href="#">Unknown</a>	Impact: Unknown

## Description

Detects operations without role-based access control

## Vulnerable Code

```
app.put('/api/Recycles/:id', denyAll())
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in data/static/codefixes/changeProductChallenge\_3\_correct.tsReference

#### & Further Reading:

<https://owasp.org/Top10/A01/>

## #302: Vertical Access Control - Missing Role Check

**Severity:**

**HIGH** (CVSS: 7.5/10.0)

**CWE ID:**

CWE-862

**OWASP:** A01:2021 - Broken Access Control

**Category:**

access\_control

**Confidence:** MEDIUM

**Location:**

data/static/codefixes/changeProductChallenge\_3\_correct.ts:160

**Line(s):**

160

**Exploitability:**

**Unknown**

**Impact:** Unknown

### Description

Detects operations without role-based access control **Vulnerable Code**

```
app.delete('/api/Quantitys/:id', denyAll())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in data/static/codefixes/changeProductChallenge\_3\_correct.tsReference

**& Further Reading:**

<https://owasp.org/Top10/A01/>

### #303: Vertical Access Control - Missing Role Check

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-862      **OWASP:** A01:2021 - Broken Access Control

**Category:** access\_control      **Confidence:** MEDIUM

**Location:** data/static/codefixes/changeProductChallenge\_3\_correct.ts:194

**Line(s):** 194

**Exploitability:** Unknown      **Impact:** Unknown

#### Description

Detects operations without role-based access control **Vulnerable Code**

```
app.delete(
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in data/static/codefixes/changeProductChallenge\_3\_correct.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A01/>

## #304: Vertical Access Control - Missing Role Check

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-862
<b>Category:</b>	access_control
<b>Location:</b>	data/static/codefixes/changeProductChallenge_3_correct.ts:225
<b>Line(s):</b>	225
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects operations without role-based access control [Vulnerable Code](#)

```
app.put('/api/Addresss/:id', requireAuth(), security.appendUserId())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in data/static/codefixes/changeProductChallenge\_3\_correct.ts [Reference](#)

### & Further Reading:

<https://owasp.org/Top10/A01/>

## #305: CSRF - Missing Origin Validation

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-352
<b>Category:</b>	csrf
<b>Location:</b>	data/static/codefixes/changeProductChallenge_3_correct.ts:24
<b>Line(s):</b>	24
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post(
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in data/static/codefixes/changeProductChallenge\_3\_correct.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A01/>

## #306: CSRF - Missing Origin Validation

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-352      **OWASP:** A01:2021 - Broken Access Control

**Category:** csrf      **Confidence:** MEDIUM

**Location:** data/static/codefixes/changeProductChallenge\_3\_correct.ts:62

**Line(s):** 62

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post('/api/Users', security.allowRegistration())
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in data/static/codefixes/changeProductChallenge\_3\_correct.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A01/>

## #307: CSRF - Missing Origin Validation

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-352      **OWASP:** A01:2021 - Broken Access Control

**Category:** csrf      **Confidence:** MEDIUM

**Location:** data/static/codefixes/changeProductChallenge\_3\_correct.ts:92

**Line(s):** 92

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post('/api/Challenges', denyAll())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in data/static/codefixes/changeProductChallenge\_3\_correct.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A01/>

## #308: CSRF - Missing Origin Validation

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-352      **OWASP:** A01:2021 - Broken Access Control

**Category:** csrf      **Confidence:** MEDIUM

**Location:** data/static/codefixes/changeProductChallenge\_3\_correct.ts:131

**Line(s):** 131

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post('/api/SecurityQuestions', denyAll())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in data/static/codefixes/changeProductChallenge\_3\_correct.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A01/>

## #309: CSRF - Missing Origin Validation

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-352
<b>Category:</b>	csrf
<b>Location:</b>	data/static/codeworks/changeProductChallenge_3_correct.ts:159
<b>Line(s):</b>	159
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post('/api/Quantitys', denyAll())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in data/static/codeworks/changeProductChallenge\_3\_correct.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A01/>

## #310: CSRF - Missing Origin Validation

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-352
<b>Category:</b>	csrf
<b>Location:</b>	data/static/codeworks/changeProductChallenge_3_correct.ts:209
<b>Line(s):</b>	209
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post('/api/Addressss', requireAuth(), security.appendUserId())
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in data/static/codefixes/changeProductChallenge\_3\_correct.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A01/>

## #311: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** test/api/feedbackApiSpec.ts:77

**Line(s):** 77

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
comment: 'Lousy crap! You use sequelize 1.7.x? Welcome to SQL Injection-land, morons!
As if that is not bad enough, you use z85/base85 and hashids for crypto? Even MD5 to hash
passwords! Srsly?!?!',
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in test/api/feedbackApiSpec.ts **References & Further**

### Reading:

<https://owasp.org/Top10/A02/>

## #312: Vertical Access Control - Missing Role Check

**Severity:** **HIGH** (CVSS: **7.5/10.0**)

**CWE ID:** CWE-862      **OWASP:** A01:2021 - Broken Access Control

**Category:** access\_control      **Confidence:** **MEDIUM**

**Location:** data/static/codefixes/changeProductChallenge\_4.ts:17

**Line(s):** 17

**Exploitability:** **Unknown**      **Impact:** Unknown

### Description

Detects operations without role-based access control **Vulnerable Code**

```
app.put('/api/Products/:id', security.isAuthorized())
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

## tion Guidance

Review and fix the access\_control issue in data/static/codefixes/changeProductChallenge\_4.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A01/>

## #313: Vertical Access Control - Missing Role Check

**Severity:** HIGH (CVSS: 7.5/10.0)

**CWE ID:** CWE-862      **OWASP:** A01:2021 - Broken Access Control

**Category:** access\_control      **Confidence:** MEDIUM

**Location:** data/static/codefixes/changeProductChallenge\_4.ts:31

**Line(s):** 31

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects operations without role-based access control **Vulnerable Code**

```
app.put('/api/Recycles/:id', security.denyAll())
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in data/static/codefixes/changeProductChallenge\_4.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A01/>

## #314: Vertical Access Control - Missing Role Check

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-862
<b>Category:</b>	access_control
<b>Location:</b>	data/static/codefixes/changeProductChallenge_4.ts:46
<b>Line(s):</b>	46
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects operations without role-based access control

### Vulnerable Code

```
app.put('/api/BasketItems/:id', security.appendUserId(), basketItems.quantityCheckBeforeBasketItemUpdate())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption  
**Remediation Guidance:**

Review and fix the access\_control issue in data/static/codefixes/changeProductChallenge\_4.ts

### & Further Reading:

<https://owasp.org/Top10/A01/>

## #315: Vertical Access Control - Missing Role Check

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-862
<b>Category:</b>	access_control
<b>Location:</b>	data/static/codefixes/changeProductChallenge_4.ts:53
<b>Line(s):</b>	53
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects operations without role-based access control **Vulnerable Code**

```
app.put('/api/Feedbacks/:id', security.denyAll())
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in data/static/codefixes/changeProductChallenge\_4.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A01/>

## #316: Vertical Access Control - Missing Role Check

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-862      **OWASP:** A01:2021 - Broken Access Control

**Category:** access\_control      **Confidence:** MEDIUM

**Location:** data/static/codefixes/changeProductChallenge\_4.ts:60

**Line(s):** 60

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects operations without role-based access control **Vulnerable Code**

```
app.put('/api/Cards/:id', security.denyAll())
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in data/static/codefixes/changeProductChallenge\_4.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A01/>

## #317: Vertical Access Control - Missing Role Check

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-862 **OWASP:** A01:2021 - Broken Access Control

**Category:** access\_control **Confidence:** MEDIUM

**Location:** data/static/codefixes/changeProductChallenge\_4.ts:70

**Line(s):** 70

**Exploitability:** Unknown **Impact:** Unknown

### Description

Detects operations without role-based access control **Vulnerable Code**

```
app.put('/api/Addresss/:id', security.appendUserId())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in data/static/codefixes/changeProductChallenge\_4.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A01/>

## #318: CSRF - Missing Origin Validation

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-352      **OWASP:** A01:2021 - Broken Access Control

**Category:** csrf      **Confidence:** MEDIUM

**Location:** data/static/codefixes/changeProductChallenge\_4.ts:16

**Line(s):** 16

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post('/api/Products', security.isAuthorized())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in data/static/codefixes/changeProductChallenge\_4.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A01/>

## #319: CSRF - Missing Origin Validation

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-352
<b>Category:</b>	csrf
<b>Location:</b>	data/static/codefixes/changeProductChallenge_4.ts:28
<b>Line(s):</b>	28
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post('/api/Recycles', security.isAuthorized())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in data/static/codefixes/changeProductChallenge\_4.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A01/>

## #320: CSRF - Missing Origin Validation

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-352
<b>Category:</b>	csrf
<b>Location:</b>	data/static/codefixes/changeProductChallenge_4.ts:47
<b>Line(s):</b>	47
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post('/api/BasketItems', security.appendUserId(), basketItems.quantityCheckBeforeBasketItemAddition(), basketItems.addBasketItem())
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in data/static/codefixes/changeProductChallenge\_4.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A01/>

## #321: CSRF - Missing Origin Validation

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-352      **OWASP:** A01:2021 - Broken Access Control

**Category:** csrf      **Confidence:** MEDIUM

**Location:** data/static/codefixes/changeProductChallenge\_4.ts:58

**Line(s):** 58

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post('/api/Cards', security.appendUserId())
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in data/static/codefixes/changeProductChallenge\_4.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A01/>

## #322: CSRF - Missing Origin Validation

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-352      **OWASP:** A01:2021 - Broken Access Control

**Category:** csrf      **Confidence:** MEDIUM

**Location:** data/static/codefixes/changeProductChallenge\_4.ts:68

**Line(s):** 68

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post('/api/Addresss', security.appendUserId())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in data/static/codefixes/changeProductChallenge\_4.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A01/>

## #323: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327 **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/az\_AZ.json:189

**Line(s):** 189

**Exploitability:** Unknown **Impact:** Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": "Veril nl rin ixrac n t l b et",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/az\_AZ.json **References &**

**Further Reading:**

<https://owasp.org/Top10/A02/>

## #324: API - No Rate Limit on Email Sending

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-770 <b>OWASP:</b> A04:2021 - Insecure Design
<b>Category:</b>	api_security <b>Confidence:</b> MEDIUM
<b>Location:</b>	data/static/codefixes/loginJimChallenge_4.ts:36
<b>Line(s):</b>	36
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects email endpoints without rate limiting **Vulnerable Code**

```
res.status(401).send(res.__('Invalid email or password.'))
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the api\_security issue in data/static/codefixes/loginJimChallenge\_4.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A04/>

## #325: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> MEDIUM
<b>Location:</b>	frontend/src/app/administration/administration.component.spec.ts:15
<b>Line(s):</b>	15
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/administration/administration.component.spec.ts

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #326: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/administration/administration.component.spec.ts:24

**Line(s):** 24

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/administration/administration.component.spec.ts

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #327: Angular - HttpClient Without Interceptor Auth

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-306
Category:	angular_security
Location:	frontend/src/app/administration/administration.component.spec.ts:55
Line(s):	55
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

#### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/administration/administration.component.spec.ts

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #328: CSRF - Missing Origin Validation

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-352      **OWASP:** A01:2021 - Broken Access Control

**Category:** csrf      **Confidence:** MEDIUM

**Location:** server.ts:574

**Line(s):** 574

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post('/rest/user/login', login())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

## #329: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/administration/administration.component.spec.ts:56
<b>Line(s):</b>	56
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClientTesting()
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/administration/administration.component.spec.ts

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #330: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/about/about.component.spec.ts:9
<b>Line(s):</b>	9
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/about/about.component.spec.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #331: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/about/about.component.spec.ts:17

**Line(s):** 17

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/about/about.component.spec.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #332: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	Confidence: MEDIUM
<b>Exploitability:</b>	frontend/src/app/about/about.component.spec.ts:43
<b>Impact:</b>	43
<b>Vulnerable Code</b>	<a href="#">Unknown</a>

#### Description

Detects HttpClient calls without auth interceptor

```
provideHttpClient(withInterceptorsFromDi()),
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/about/about.component.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #333: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/about/about.component.spec.ts:44

**Line(s):** 44

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClientTesting()
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/about/about.component.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #334: Vertical Access Control - Test Account in Production

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-489 <b>OWASP:</b> A05:2021 - Security Misconfiguration
<b>Category:</b>	access_control <b>Confidence:</b> MEDIUM
<b>Location:</b>	test/api/authenticatedUsersSpec.ts:14
<b>Line(s):</b>	14
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

## Description

Detects test or demo accounts in production code **Vulnerable Code**

```
const authHeader = { Authorization: 'Bearer ${security.authorize({ data: { email: \'admin@juice-sh.op\' } })}', 'content-type': 'application/json' }
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in test/api/authenticatedUsersSpec.ts **References & Further Reading:**

### Further Reading:

<https://owasp.org/Top10/A05/>

## #335: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> MEDIUM
<b>Location:</b>	frontend/src/app/accounting/accounting.component.spec.ts:8
<b>Line(s):</b>	8
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/accounting/accounting.component.spec.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #336: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/accounting/accounting.component.spec.ts:26

**Line(s):** 26

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/accounting/accounting.component.spec.ts **References & Further Reading:**

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #337: CSRF - Missing Origin Validation

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-352      **OWASP:** A01:2021 - Broken Access Control

**Category:** csrf      **Confidence:** MEDIUM

**Location:** server.ts:418

**Line(s):** 418

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post('/api/Quantitys', security.denyAll())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

## #338: XSS - Angular bypassSecurityTrustResourceUrl

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-79 <b>OWASP:</b> A03:2021 - Injection
<b>Category:</b>	xss <b>Confidence:</b> MEDIUM
<b>Location:</b>	data/static/codefixes/localXssChallenge_1.ts:6
<b>Line(s):</b>	6
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects Angular resource URL trust bypass **Vulnerable Code**

```
this.searchValue = this.sanitizer.bypassSecurityTrustResourceUrl(queryParam)
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the xss issue in data/static/codefixes/localXssChallenge\_1.ts **References &**

### Further Reading:

<https://owasp.org/Top10/A03/>

## #339: XSS - Angular bypassSecurityTrustResourceUrl

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-79 <b>OWASP:</b> A03:2021 - Injection
<b>Category:</b>	xss <b>Confidence:</b> MEDIUM
<b>Location:</b>	data/static/codefixes/localXssChallenge_1.ts:6
<b>Line(s):</b>	6
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

## Description

Detects Angular resource URL trust bypass **Vulnerable Code**

```
this.searchValue = this.sanitizer.bypassSecurityTrustResourceUrl(queryParam)
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the xss issue in data/static/codefixes/localXssChallenge\_1.ts **References &**

## Further Reading:

<https://owasp.org/Top10/A03/>

## #340: XSS - Angular bypassSecurityTrustResourceUrl

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-79 <b>OWASP:</b> A03:2021 - Injection
<b>Category:</b>	xss <b>Confidence:</b> MEDIUM
<b>Location:</b>	data/static/codefixes/xssBonusChallenge_2.ts:6
<b>Line(s):</b>	6
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

## Description

Detects Angular resource URL trust bypass **Vulnerable Code**

```
this.searchValue = this.sanitizer.bypassSecurityTrustResourceUrl(queryParam)
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the XSS issue in data/static/codefixes/xssBonusChallenge\_2.ts **References & Further Reading:**

<https://owasp.org/Top10/A03/>

## #341: XSS - Angular bypassSecurityTrustResourceUrl

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-79
Category:	xss
Location:	data/static/codefixes/xssBonusChallenge_2.ts:6
Line(s):	6
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

## Description

Detects Angular resource URL trust bypass **Vulnerable Code**

```
this.searchValue = this.sanitizer.bypassSecurityTrustResourceUrl(queryParam)
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the XSS issue in data/static/codefixes/xssBonusChallenge\_2.ts **References &**

#### Further Reading:

<https://owasp.org/Top10/A03/>

## #342: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/main.ts:85

**Line(s):** 85

**Exploitability:** Unknown      **Impact:** Unknown

#### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HTTP_INTERCEPTORS, HttpClient, withInterceptorsFromDi, provideHttpClient } from
'@angular/common/http';
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/main.ts **References & Further**

#### Reading:

<https://owasp.org/Top10/A07/>

## #343: Angular - HttpClient Without Interceptor Auth

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)	
CWE ID:	CWE-306	OWASP: A07:2021 - Identification and Authentication Failures
Category:	angular_security	Confidence: <b>MEDIUM</b>
Location:	frontend/src/main.ts:85	
Line(s):	85	
Exploitability:	<b>Unknown</b>	Impact: Unknown

#### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HTTP_INTERCEPTORS, HttpClient, withInterceptorsFromDi, provideHttpClient } from
'@angular/common/http';
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/main.ts **References & Further**

#### Reading:

<https://owasp.org/Top10/A07/>

## #344: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/main.ts:97
<b>Line(s):</b>	97
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
deps: [HttpClient]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/main.ts **References & Further**

### Reading:

<https://owasp.org/Top10/A07/>

## #345: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/main.ts:148
<b>Line(s):</b>	148
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/main.ts **References & Further Reading:**

### Reading:

<https://owasp.org/Top10/A07/>

## #346: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/app.component.spec.ts:9

**Line(s):** 9

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/app.component.spec.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #347: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/app.component.spec.ts:37

**Line(s):** 37

**Exploitability:** **Unknown**      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/app.component.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #348: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/app.component.spec.ts:71

**Line(s):** 71

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [TranslateService, LoginGuard, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/app.component.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #349: Angular - HttpClient Without Interceptor Auth

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-306
Category:	angular_security
Location:	frontend/src/app/app.component.spec.ts:71
Line(s):	71
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [TranslateService, LoginGuard, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/app.component.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #350: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/app.guard.spec.ts:8
<b>Line(s):</b>	8
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/app.guard.spec.ts **References &**

### Further Reading:

<https://owasp.org/Top10/A07/>

## #351: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/app.guard.spec.ts:11
<b>Line(s):</b>	11
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/app.guard.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #352: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/app.guard.spec.ts:19

**Line(s):** 19

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [LoginGuard, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/app.guard.spec.ts **References &**

#### Further Reading:

<https://owasp.org/Top10/A07/>

## #353: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/app.guard.spec.ts:19

**Line(s):** 19

**Exploitability:** Unknown      **Impact:** Unknown

#### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [LoginGuard, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/app.guard.spec.ts **References &**

#### Further Reading:

<https://owasp.org/Top10/A07/>

## #354: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/app.guard.spec.ts:70

**Line(s):** 70

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/app.guard.spec.ts **References &**

#### Further Reading:

<https://owasp.org/Top10/A07/>

## #355: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/app.guard.spec.ts:71
<b>Line(s):</b>	71
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClientTesting()
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/app.guard.spec.ts **References &**

### Further Reading:

<https://owasp.org/Top10/A07/>

## #356: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/app.guard.spec.ts:117
<b>Line(s):</b>	117
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/app.guard.spec.ts **References & Further Reading:**

## Further Reading:

<https://owasp.org/Top10/A07/>

## #357: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/app.guard.spec.ts:118

**Line(s):** 118

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClientTesting()
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/app.guard.spec.ts **References &**

#### Further Reading:

<https://owasp.org/Top10/A07/>

## #358: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/app.guard.spec.ts:164

**Line(s):** 164

**Exploitability:** [Unknown](#)      **Impact:** Unknown

#### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/app.guard.spec.ts **References &**

**Further Reading:**

<https://owasp.org/Top10/A07/>

## #359: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/app.guard.spec.ts:165

**Line(s):** 165

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClientTesting()
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/app.guard.spec.ts **References &**

### Further Reading:

<https://owasp.org/Top10/A07/>

## #360: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/app.module.ts:7
<b>Line(s):</b>	7
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { type HttpClient } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/app.module.ts **References &**

### Further Reading:

<https://owasp.org/Top10/A07/>

## #361: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/app.module.ts:12
<b>Line(s):</b>	12
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
export function HttpLoaderFactory (http: HttpClient) {
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/app.module.ts **References &**

## Further Reading:

<https://owasp.org/Top10/A07/>

## #362: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** **MEDIUM**

**Location:** frontend/src/app/app.module.ts:26

**Line(s):** 26

**Exploitability:** [Unknown](#)      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
deps: [HttpClient]
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/app.module.ts **References &**

#### Further Reading:

<https://owasp.org/Top10/A07/>

## #363: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/app.module.ts:32

**Line(s):** 32

**Exploitability:** Unknown      **Impact:** Unknown

#### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
HttpClientModule,
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/app.module.ts **References &**

**Further Reading:**

<https://owasp.org/Top10/A07/>

## #364: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> MEDIUM
<b>Location:</b>	frontend/src/app/accounting/accounting.component.spec.ts:69
<b>Line(s):</b>	69
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/accounting/accounting.component.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #365: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/accounting/accounting.component.spec.ts:70
<b>Line(s):</b>	70
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClientTesting()
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/accounting/accounting.component.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #366: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/address/address.component.spec.ts:6
<b>Line(s):</b>	6
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/address/address.component.spec.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #367: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/address/address.component.spec.ts:30

**Line(s):** 30

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/address/address.component.spec.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #368: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	Confidence: MEDIUM
<b>Exploitability:</b>	frontend/src/app/address/address.component.spec.ts:73
<b>Impact:</b>	73
<b>Line(s):</b>	Unknown
<b>Exploitability:</b>	Impact: Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/address/address.component.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #369: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/address/address.component.spec.ts:74

**Line(s):** 74

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClientTesting()
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/address/address.component.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #370: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/address-create/address-create.component.spec.ts:7
<b>Line(s):</b>	7
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/address-create/address-create.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #371: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/address-create/address-create.component.spec.ts:23
<b>Line(s):</b>	23
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/address-create/address-create.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #372: Angular - HttpClient Without Interceptor Auth

**Severity:** HIGH (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/address-create/address-create.component.spec.ts:61

**Line(s):** 61

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/address-create/address-create.component.spec.ts

#### References & Further Reading:

<https://owasp.org/Top10/A07/>

## #373: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/address-create/address-create.component.spec.ts:62

**Line(s):** 62

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor

### Vulnerable Code

```
provideHttpClientTesting()
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/address-create/address-

create.component.spec.ts**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #374: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	frontend/src/app/address-select/address-select.component.spec.ts:7
<b>Exploitability:</b>	7
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor**Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/address-select/address-select.component.spec.ts**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #375: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/address-select/address-select.component.spec.ts:30
<b>Line(s):</b>	30
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/address-select/address-select.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #376: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/address-select/address-select.component.spec.ts:67
<b>Line(s):</b>	67
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor

### Vulnerable Code

```
{ provide: MatSnackBar, useValue: snackBar }, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption  
**Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/address-select/address-select.component.spec.ts

### References & Further Reading:

<https://owasp.org/Top10/A07/>

## #377: Angular - HttpClient Without Interceptor Auth

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-306
Category:	angular_security
Location:	OWASP: A07:2021 - Identification and Authentication Failures
Line(s):	Confidence: MEDIUM
Exploitability:	frontend/src/app/address-select/address-select.component.spec.ts:67
	67
	Impact: Unknown

## Description

Detects HttpClient calls without auth interceptor

### Vulnerable Code

```
{ provide: MatSnackBar, useValue: snackBar }, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/address-select/address-select.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #378: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)	
<b>CWE ID:</b>	CWE-306	<b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security	<b>Confidence:</b> MEDIUM
<b>Location:</b>	frontend/src/app/basket/basket.component.spec.ts:17	
<b>Line(s):</b>	17	
<b>Exploitability:</b>	<a href="#">Unknown</a>	<b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

## tion Guidance

Review and fix the angular\_security issue in frontend/src/app/basket/basket.component.spec.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #379: Angular - HttpClient Without Interceptor Auth

**Severity:** HIGH (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/basket/basket.component.spec.ts:25

**Line(s):** 25

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/basket/basket.component.spec.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #380: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/basket/basket.component.spec.ts:52
<b>Line(s):</b>	52
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/basket/basket.component.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #381: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/basket/basket.component.spec.ts:53
<b>Line(s):</b>	53
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClientTesting()
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/basket/basket.component.spec.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #382: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/challenge-solved-notification/challenge-solved-notif...

**Line(s):** 14

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/challenge-solved-notification/challenge-solved-notification.component.spec.ts

**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #383: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/challenge-solved-notification/challenge-solved-notif...

**Line(s):** 22

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor

### Vulnerable Code

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/challenge-solved-

notification/challenge-solved-notification.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #384: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	frontend/src/app/challenge-solved-notification/challenge-solved-notif... 73
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/challenge-solved-notification/challenge-solved-notification.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #385: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/challenge-solved-notification/challenge-solved-notif...
<b>Line(s):</b>	74
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClientTesting()
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/challenge-solved-notification/challenge-solved-notification.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #386: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/challenge-status-badge/challenge-status-badge.compon...
<b>Line(s):</b>	13
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/challenge-status-badge/challenge-status-badge.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #387: Angular - HttpClient Without Interceptor Auth

**Severity:** HIGH (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/challenge-status-badge/challenge-status-badge.component.ts...

**Line(s):** 19

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/challenge-status-badge/challenge-status-badge.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #388: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	Confidence: MEDIUM
<b>Exploitability:</b>	frontend/src/app/challenge-status-badge/challenge-status-badge.component.spec.ts
<b>Impact:</b>	47
<b>Line(s):</b>	Unknown
<b>Exploitability:</b>	Impact: Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/challenge-status-badge/challenge-

status-badge.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #389: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	fronted/src/app/challenge-status-badge/challenge-status-badge.compon...
<b>Exploitability:</b>	48
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClientTesting()
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in fronted/src/app/challenge-status-badge/challenge-status-badge.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #390: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/chatbot/chatbot.component.spec.ts:19
<b>Line(s):</b>	19
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/chatbot/chatbot.component.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #391: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/chatbot/chatbot.component.spec.ts:22
<b>Line(s):</b>	22
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/chatbot/chatbot.component.spec.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #392: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/chatbot/chatbot.component.spec.ts:64

**Line(s):** 64

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/chatbot/chatbot.component.spec.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #393: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	Confidence: MEDIUM
<b>Exploitability:</b>	frontend/src/app/chatbot/chatbot.component.spec.ts:65
<b>Impact:</b>	65
<b>Line(s):</b>	Unknown
<b>Exploitability:</b>	Impact: Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClientTesting()
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/chatbot/chatbot.component.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #394: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> MEDIUM
<b>Location:</b>	frontend/src/app/feedback-details/feedback-details.component.spec.ts:9
<b>Line(s):</b>	9
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/feedback-details/feedback-details.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #395: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/feedback-details/feedback-details.component.spec.ts:14
<b>Line(s):</b>	14
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/feedback-details/feedback-details.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #396: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/feedback-details/feedback-details.component.spec.ts:30
<b>Line(s):</b>	30
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/feedback-details/feedback-details.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #397: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/feedback-details/feedback-details.component.spec.ts:31

**Line(s):** 31

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClientTesting()
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/feedback-details/feedback-details.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #398: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	Confidence: MEDIUM
<b>Exploitability:</b>	frontend/src/app/navbar/navbar.component.spec.ts:12
<b>Impact:</b>	12
<b>Line(s):</b>	Unknown
<b>Exploitability:</b>	Impact: Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/navbar/navbar.component.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #399: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/navbar/navbar.component.spec.ts:40

**Line(s):** 40

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/navbar/navbar.component.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #400: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/navbar/navbar.component.spec.ts:115
<b>Line(s):</b>	115
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/navbar/navbar.component.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #401: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/navbar/navbar.component.spec.ts:116
<b>Line(s):</b>	116
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClientTesting()
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/navbar/navbar.component.spec.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #402: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/order-completion/order-completion.component.spec.ts:8

**Line(s):** 8

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/order-completion/order-completion.component.spec.ts

#### References & Further Reading:

<https://owasp.org/Top10/A07/>

## #403: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	Confidence: MEDIUM
<b>Exploitability:</b>	frontend/src/app/order-completion/order-completion.component.spec.ts:26
<b>Impact:</b>	26
<b>Impact:</b>	Unknown
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor

### Vulnerable Code

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/order-completion/order-

completion.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #404: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	frontend/src/app/order-completion/order-completion.component.spec.ts:71
<b>Exploitability:</b>	71
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/order-completion/order-completion.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #405: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/code-snippet/code-snippet.component.spec.ts:8
<b>Line(s):</b>	8
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/code-snippet/code-snippet.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #406: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/code-snippet/code-snippet.component.spec.ts:21
<b>Line(s):</b>	21
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/code-snippet/code-

snippet.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #407: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/code-snippet/code-snippet.component.spec.ts:62

**Line(s):** 62

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/code-snippet/code-snippet.component.spec.ts

#### References & Further Reading:

<https://owasp.org/Top10/A07/>

## #408: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	Confidence: MEDIUM
<b>Exploitability:</b>	frontend/src/app/code-snippet/code-snippet.component.spec.ts:63
<b>Impact:</b>	63
<b>Line(s):</b>	Unknown
<b>Exploitability:</b>	Impact: Unknown

### Description

Detects HttpClient calls without auth interceptor

### Vulnerable Code

```
provideHttpClientTesting()
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/code-snippet/code-

snippet.component.spec.ts**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #409: Crypto - Insecure Random Number Generator

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-338
<b>Category:</b>	cryptography
<b>Location:</b>	frontend/src/app/code-snippet/code-snippet.component.ts:159
<b>Line(s):</b>	159
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects usage of insecure random number generators **Vulnerable Code**

```
.map((fix, index) => ({ fix, index, sort: Math.random() }))
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/app/code-snippet/code-snippet.component.ts **References & Further Reading:**

<https://owasp.org/Top10/A02/>

## #410: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/complaint/complaint.component.spec.ts:21
<b>Line(s):</b>	21
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/complaint/complaint.component.spec.ts **Reference**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #411: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/complaint/complaint.component.spec.ts:23
<b>Line(s):</b>	23
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/complaint/complaint.component.spec.ts **Reference**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #412: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/complaint/complaint.component.spec.ts:57

**Line(s):** 57

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/complaint/complaint.component.spec.ts **Reference**

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #413: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	Confidence: MEDIUM
<b>Exploitability:</b>	frontend/src/app/complaint/complaint.component.spec.ts:58
<b>Impact:</b>	58
<b>Line(s):</b>	Unknown
<b>Exploitability:</b>	Impact: Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClientTesting()
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/complaint/complaint.component.spec.ts **Reference**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #414: API - No Rate Limit on Email Sending

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-770      **OWASP:** A04:2021 - Insecure Design

**Category:** api\_security      **Confidence:** MEDIUM

**Location:** data/static/codeworks/loginAdminChallenge\_2.ts:34

**Line(s):** 34

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects email endpoints without rate limiting **Vulnerable Code**

```
res.status(401).send(res._('Invalid email or password.'))
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the api\_security issue in data/static/codeworks/loginAdminChallenge\_2.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A04/>

## #415: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/contact/contact.component.spec.ts:8
<b>Line(s):</b>	8
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/contact/contact.component.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #416: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/contact/contact.component.spec.ts:22
<b>Line(s):</b>	22
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/contact/contact.component.spec.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #417: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/contact/contact.component.spec.ts:63

**Line(s):** 63

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/contact/contact.component.spec.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #418: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	Confidence: MEDIUM
<b>Line(s):</b>	frontend/src/app/contact/contact.component.spec.ts:64
<b>Exploitability:</b>	64
<b>Impact:</b>	Unknown

#### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClientTesting()
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/contact/contact.component.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #419: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/data-export/data-export.component.spec.ts:10

**Line(s):** 10

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/data-export/data-export.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #420: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/order-completion/order-completion.component.spec.ts:72
<b>Line(s):</b>	72
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClientTesting()
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/order-completion/order-completion.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #421: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/data-export/data-export.component.spec.ts:22
<b>Line(s):</b>	22
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/data-export/data-export.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #422: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/data-export/data-export.component.spec.ts:50

**Line(s):** 50

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/data-export/data-export.component.spec.ts

#### References & Further Reading:

<https://owasp.org/Top10/A07/>

## #423: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	Confidence: MEDIUM
<b>Exploitability:</b>	frontend/src/app/data-export/data-export.component.spec.ts:51
<b>Impact:</b>	51
<b>Line(s):</b>	Unknown
<b>Exploitability:</b>	Impact: Unknown

### Description

Detects HttpClient calls without auth interceptor

### Vulnerable Code

```
provideHttpClientTesting()
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/data-export/data-

export.component.spec.ts**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #424: API - No Rate Limit on Email Sending

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-770
<b>Category:</b>	api_security
<b>Location:</b>	OWASP: A04:2021 - Insecure Design
<b>Line(s):</b>	api_security
<b>Exploitability:</b>	Confidence: MEDIUM
	data/static/codefixes/loginAdminChallenge_3.ts:34
<b>Line(s):</b>	34
<b>Exploitability:</b>	<b>Unknown</b>
	<b>Impact:</b> Unknown

### Description

Detects email endpoints without rate limiting **Vulnerable Code**

```
res.status(401).send(res.__('Invalid email or password.'))
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the api\_security issue in data/static/codefixes/loginAdminChallenge\_3.ts **References & Further Reading:**

<https://owasp.org/Top10/A04/>

## #425: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/delivery-method/delivery-method.component.spec.ts:13
<b>Line(s):</b>	13
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/delivery-method/delivery-method.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #426: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/delivery-method/delivery-method.component.spec.ts:29
<b>Line(s):</b>	29
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/delivery-method/delivery-method.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #427: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/delivery-method/delivery-method.component.spec.ts:64

**Line(s):** 64

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/delivery-method/delivery-method.component.spec.ts

#### References & Further Reading:

<https://owasp.org/Top10/A07/>

## #428: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/delivery-method/delivery-method.component.spec.ts:65

**Line(s):** 65

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor

#### Vulnerable Code

```
provideHttpClientTesting()
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/delivery-method/delivery-

method.component.spec.ts**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #429: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	Confidence: MEDIUM
<b>Exploitability:</b>	frontend/src/app/deluxe-user/deluxe-user.component.spec.ts:7
<b>Impact:</b>	7
<b>Line(s):</b>	Unknown
<b>Exploitability:</b>	Impact: Unknown

### Description

Detects HttpClient calls without auth interceptor**Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/deluxe-user/deluxe-user.component.spec.ts**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #430: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/deluxe-user/deluxe-user.component.spec.ts:32
<b>Line(s):</b>	32
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/deluxe-user/deluxe-user.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #431: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/deluxe-user/deluxe-user.component.spec.ts:75
<b>Line(s):</b>	75
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/deluxe-user/deluxe-

user.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #432: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/deluxe-user/deluxe-user.component.spec.ts:76

**Line(s):** 76

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClientTesting()
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/deluxe-user/deluxe-user.component.spec.ts

#### References & Further Reading:

<https://owasp.org/Top10/A07/>

## #433: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	Confidence: MEDIUM
<b>Exploitability:</b>	frontend/src/app/order-summary/order-summary.component.spec.ts:13
<b>Impact:</b>	13
<b>Line(s):</b>	Unknown
<b>Exploitability:</b>	Impact: Unknown

### Description

Detects HttpClient calls without auth interceptor

### Vulnerable Code

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/order-summary/order-

summary.component.spec.ts**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #434: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	frontend/src/app/order-summary/order-summary.component.spec.ts:30
<b>Exploitability:</b>	30
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor**Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/order-summary/order-summary.component.spec.ts**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #435: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/order-history/order-history.component.spec.ts:8
<b>Line(s):</b>	8
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/order-history/order-history.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #436: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/order-history/order-history.component.spec.ts:28
<b>Line(s):</b>	28
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/order-history/order-history.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #437: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/order-history/order-history.component.spec.ts:64

**Line(s):** 64

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/order-history/order-history.component.spec.ts

#### References & Further Reading:

<https://owasp.org/Top10/A07/>

## #438: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	Confidence: MEDIUM
<b>Exploitability:</b>	frontend/src/app/order-history/order-history.component.spec.ts:65
<b>Impact:</b>	65
<b>Line(s):</b>	Unknown
<b>Exploitability:</b>	Impact: Unknown

### Description

Detects HttpClient calls without auth interceptor

### Vulnerable Code

```
provideHttpClientTesting()
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/order-history/order-

history.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #439: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	Confidence: MEDIUM
<b>Exploitability:</b>	frontend/src/app/recycle/recycle.component.spec.ts:11
<b>Line(s):</b>	11
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/recycle/recycle.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #440: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/recycle/recycle.component.spec.ts:35
<b>Line(s):</b>	35
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/recycle/recycle.component.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #441: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/recycle/recycle.component.spec.ts:87
<b>Line(s):</b>	87
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/recycle/recycle.component.spec.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #442: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/recycle/recycle.component.spec.ts:88

**Line(s):** 88

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClientTesting()
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/recycle/recycle.component.spec.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #443: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/change-password/change-password.component.spec.ts:8

**Line(s):** 8

**Exploitability:** **Unknown**      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/change-password/change-

password.component.spec.ts**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #444: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	frontend/src/app/change-password/change-password.component.spec.ts:19
<b>Exploitability:</b>	19
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor**Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/change-password/change-password.component.spec.ts**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #445: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/register/register.component.spec.ts:10
<b>Line(s):</b>	10
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/register/register.component.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #446: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/register/register.component.spec.ts:29
<b>Line(s):</b>	29
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/register/register.component.spec.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #447: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/register/register.component.spec.ts:69

**Line(s):** 69

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/register/register.component.spec.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #448: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	Confidence: MEDIUM
<b>Exploitability:</b>	frontend/src/app/register/register.component.spec.ts:70
<b>Impact:</b>	70
<b>Line(s):</b>	Unknown
<b>Exploitability:</b>	Impact: Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClientTesting()
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/register/register.component.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #449: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> MEDIUM
<b>Location:</b>	frontend/src/app/saved-address/saved-address.component.spec.ts:7
<b>Line(s):</b>	7
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/saved-address/saved-address.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #450: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/order-summary/order-summary.component.spec.ts:80
<b>Line(s):</b>	80
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/order-summary/order-summary.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #451: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/order-summary/order-summary.component.spec.ts:81
<b>Line(s):</b>	81
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClientTesting()
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/order-summary/order-summary.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #452: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/payment/payment.component.spec.ts:7

**Line(s):** 7

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/payment/payment.component.spec.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #453: Angular - HttpClient Without Interceptor Auth

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-306
Category:	angular_security
Location:	frontend/src/app/payment/payment.component.spec.ts:41
Line(s):	41
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

#### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/payment/payment.component.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #454: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/payment/payment.component.spec.ts:113

**Line(s):** 113

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/payment/payment.component.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #455: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/payment/payment.component.spec.ts:114
<b>Line(s):</b>	114
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClientTesting()
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/payment/payment.component.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #456: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/payment-method/payment-method.component.spec.ts:7
<b>Line(s):</b>	7
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/payment-method/payment-method.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #457: Angular - HttpClient Without Interceptor Auth

**Severity:** HIGH (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/payment-method/payment-method.component.spec.ts:25

**Line(s):** 25

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/payment-method/payment-method.component.spec.ts

#### References & Further Reading:

<https://owasp.org/Top10/A07/>

## #458: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	frontend/src/app/payment-method/payment-method.component.spec.ts:63
<b>Exploitability:</b>	Confidence: MEDIUM
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor

#### Vulnerable Code

```
provideHttpClient(withInterceptorsFromDi()),
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/payment-method/payment-

method.component.spec.ts**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #459: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	frontend/src/app/payment-method/payment-method.component.spec.ts:64
<b>Exploitability:</b>	64
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor**Vulnerable Code**

```
provideHttpClientTesting()
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/payment-method/payment-method.component.spec.ts**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #460: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/photo-wall/photo-wall.component.spec.ts:8
<b>Line(s):</b>	8
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/photo-wall/photo-wall.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #461: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/photo-wall/photo-wall.component.spec.ts:30
<b>Line(s):</b>	30
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/photo-wall/photo-wall.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #462: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/photo-wall/photo-wall.component.spec.ts:76

**Line(s):** 76

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/photo-wall/photo-wall.component.spec.ts

#### References & Further Reading:

<https://owasp.org/Top10/A07/>

## #463: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/photo-wall/photo-wall.component.spec.ts:77

**Line(s):** 77

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor

#### Vulnerable Code

```
provideHttpClientTesting()
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/photo-wall/photo-

wall.component.spec.ts**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #464: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	privacy-policy.component.spec.ts:8
<b>Exploitability:</b>	Unknown
<b>Impact:</b>	Medium

### Description

Detects HttpClient calls without auth interceptor**Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/privacy-policy/privacy-policy.component.spec.ts**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #465: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/privacy-policy/privacy-policy.component.spec.ts:16
<b>Line(s):</b>	16
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/privacy-policy/privacy-policy.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #466: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/privacy-policy/privacy-policy.component.spec.ts:41
<b>Line(s):</b>	41
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/privacy-policy/privacy-

policy.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #467: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/privacy-policy/privacy-policy.component.spec.ts:42

**Line(s):** 42

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClientTesting()
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/privacy-policy/privacy-policy.component.spec.ts

#### References & Further Reading:

<https://owasp.org/Top10/A07/>

## #468: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/change-password/change-password.component.spec.ts:39

**Line(s):** 39

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor

### Vulnerable Code

```
providers: [{ provide: UserService, useValue: userService }, provideHttpClient(withInterceptorsFromDi()), provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/change-password/change-password.component.spec.ts**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #469: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/change-password/change-password.component.spec.ts:39
<b>Line(s):</b>	39
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor**Vulnerable Code**

```
providers: [{ provide: UserService, useValue: userService }, provideHttpClient(withInterceptorsFromDi()), provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/change-password/change-password.component.spec.ts**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #470: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/forgot-password/forgot-password.component.spec.ts:7
<b>Line(s):</b>	7
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/forgot-password/forgot-password.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #471: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/forgot-password/forgot-password.component.spec.ts:23
<b>Line(s):</b>	23
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/forgot-password/forgot-password.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #472: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/forgot-password/forgot-password.component.spec.ts:52

**Line(s):** 52

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/forgot-password/forgot-password.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #473: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	fronted/src/app/forgot-password/forgot-password.component.spec.ts:53
<b>Exploitability:</b>	53
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClientTesting()
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/forgot-password/forgot-

password.component.spec.ts**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #474: Business Logic - Password Reset Token Reuse

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-640
<b>Category:</b>	business_logic
<b>Location:</b>	frontend/src/app/forgot-password/forgot-password.component.ts:76
<b>Line(s):</b>	76
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

### Description

Detects password reset token reuse vulnerabilities **Vulnerable Code**

```
this.userService.resetPassword({
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the business\_logic issue in frontend/src/app/forgot-password/forgot-password.component.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #475: API - No Rate Limit on Email Sending

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-770 <b>OWASP:</b> A04:2021 - Insecure Design
<b>Category:</b>	api_security <b>Confidence:</b> MEDIUM
<b>Location:</b>	data/static/codfixes/loginAdminChallenge_4_correct.ts:34
<b>Line(s):</b>	34
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects email endpoints without rate limiting **Vulnerable Code**

```
res.status(401).send(res.__('Invalid email or password.'))
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the api\_security issue in data/static/codfixes/loginAdminChallenge\_4\_correct.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A04/>

## #476: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> MEDIUM
<b>Location:</b>	frontend/src/app/product-details/product-details.component.spec.ts:12
<b>Line(s):</b>	12
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/product-details/product-details.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #477: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/product-details/product-details.component.spec.ts:29

**Line(s):** 29

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/product-details/product-details.component.spec.ts

#### References & Further Reading:

<https://owasp.org/Top10/A07/>

## #478: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	frontend/src/app/product-details/product-details.component.spec.ts:71
<b>Exploitability:</b>	71
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor

#### Vulnerable Code

```
provideHttpClient(withInterceptorsFromDi()),
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/product-details/product-

details.component.spec.ts**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #479: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	frontend/src/app/product-details/product-details.component.spec.ts:72
<b>Exploitability:</b>	72
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor**Vulnerable Code**

```
provideHttpClientTesting()
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/product-details/product-details.component.spec.ts**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #480: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/purchase-basket/purchase-basket.component.spec.ts:14
<b>Line(s):</b>	14
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/purchase-basket/purchase-basket.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #481: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/purchase-basket/purchase-basket.component.spec.ts:24
<b>Line(s):</b>	24
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/purchase-basket/purchase-basket.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #482: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/purchase-basket/purchase-basket.component.spec.ts:70

**Line(s):** 70

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/purchase-basket/purchase-basket.component.spec.ts

#### References & Further Reading:

<https://owasp.org/Top10/A07/>

## #483: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/purchase-basket/purchase-basket.component.spec.ts:71

**Line(s):** 71

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor

### Vulnerable Code

```
provideHttpClientTesting()
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/purchase-basket/purchase-

basket.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #484: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	Confidence: MEDIUM
<b>Exploitability:</b>	frontend/src/app/Services/chatbot.service.spec.ts:6
<b>Line(s):</b>	6
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpTestingController, provideHttpClientTesting } from
'@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/chatbot.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #485: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/chatbot.service.spec.ts:10
<b>Line(s):</b>	10
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/chatbot.service.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #486: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/saved-address/saved-address.component.spec.ts:27
<b>Line(s):</b>	27
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor

### Vulnerable Code

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption

## Remediation Guidance

Review and fix the angular\_security issue in frontend/src/app/saved-address/saved-address.component.spec.ts

### References & Further Reading:

<https://owasp.org/Top10/A07/>

## #487: Angular - HttpClient Without Interceptor Auth

**Severity:** HIGH (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/saved-address/saved-address.component.spec.ts:52

**Line(s):** 52

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor

### Vulnerable Code

```
providers: [{ provide: MatSnackBar, useValue: snackBar }, provideHttpClient(withInterceptorsFromDi()), provideHttpClientTesting()]
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/saved-address/saved-address.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #488: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)	
<b>CWE ID:</b>	CWE-306	<b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security	<b>Confidence:</b> MEDIUM
<b>Location:</b>	frontend/src/app/saved-address/saved-address.component.spec.ts:52	
<b>Line(s):</b>	52	
<b>Exploitability:</b>	<a href="#">Unknown</a>	<b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [{ provide: MatSnackBar, useValue: snackBar }, provideHttpClient(withInterceptorsFromDi()), provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/saved-address/saved-address.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #489: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/saved-payment-methods/saved-payment-methods.componen...
<b>Line(s):</b>	7
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/saved-payment-methods/saved-payment-methods.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #490: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/saved-payment-methods/saved-payment-methods.component.ts
<b>Line(s):</b>	25
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/saved-payment-methods/saved-payment-methods.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #491: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/saved-payment-methods/saved-payment-methods.component.ts
<b>Line(s):</b>	57
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects HttpClient calls without auth interceptor

### Vulnerable Code

```
provideHttpClient(withInterceptorsFromDi()),
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption

## Remediation Guidance

Review and fix the angular\_security issue in frontend/src/app/saved-payment-methods/saved-payment-methods.component.spec.ts

### References & Further Reading:

<https://owasp.org/Top10/A07/>

## #492: Angular - HttpClient Without Interceptor Auth

**Severity:** HIGH (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/saved-payment-methods/saved-payment-methods.component.ts

**Line(s):** 58

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor

### Vulnerable Code

```
provideHttpClientTesting()
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/saved-payment-methods/saved-payment-methods.component.spec.ts

#### References & Further Reading:

<https://owasp.org/Top10/A07/>

## #493: Angular - HttpClient Without Interceptor Auth

**Severity:** HIGH (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/score-board/score-board.component.spec.ts:2

**Line(s):** 2

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor

#### Vulnerable Code

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/score-board/score-

board.component.spec.ts**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #494: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	frontend/src/app/score-board/score-board.component.spec.ts:23
<b>Exploitability:</b>	Confidence: MEDIUM
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor**Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/score-board/score-board.component.spec.ts**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #495: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/score-board/score-board.component.spec.ts:82
<b>Line(s):</b>	82
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/score-board/score-board.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #496: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/score-board/score-board.component.spec.ts:83
<b>Line(s):</b>	83
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClientTesting()
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/score-board/score-

board.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #497: API - No Rate Limit on Email Sending

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-770      **OWASP:** A04:2021 - Insecure Design

**Category:** api\_security      **Confidence:** **MEDIUM**

**Location:** data/static/codeworks/loginAdminChallenge\_1.ts:36

**Line(s):** 36

**Exploitability:** [Unknown](#)      **Impact:** Unknown

## Description

Detects email endpoints without rate limiting **Vulnerable Code**

```
res.status(401).send(res.__('Invalid email or password.'))
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the api\_security issue in data/static/codefixes/loginAdminChallenge\_1.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A04/>

## #498: CSRF - Missing Origin Validation

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-352      **OWASP:** A01:2021 - Broken Access Control

**Category:** csrf      **Confidence:** MEDIUM

**Location:** server.ts:432

**Line(s):** 432

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post('/api/PrivacyRequests', security.isAuthorized())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

## #499: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> MEDIUM
<b>Location:</b>	frontend/src/app/search-result/search-result.component.spec.ts:8
<b>Line(s):</b>	8
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/search-result/search-result.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #500: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/search-result/search-result.component.spec.ts:32
<b>Line(s):</b>	32
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/search-result/search-result.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #501: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/search-result/search-result.component.spec.ts:119
<b>Line(s):</b>	119
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/search-result/search-result.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #502: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/search-result/search-result.component.spec.ts:120

**Line(s):** 120

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClientTesting()
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/search-result/search-result.component.spec.ts

#### References & Further Reading:

<https://owasp.org/Top10/A07/>

## #503: API - No Rate Limit on Email Sending

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-770
<b>Category:</b>	api_security
<b>Location:</b>	OWASP: A04:2021 - Insecure Design
<b>Line(s):</b>	data/static/codfixes/loginBenderChallenge_1.ts:36
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects email endpoints without rate limiting

#### Vulnerable Code

```
res.status(401).send(res._('Invalid email or password.'))
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the api\_security issue in data/static/codfixes/loginBenderChallenge\_1.ts

**& Further Reading:**

<https://owasp.org/Top10/A04/>

## #504: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/login/login.component.spec.ts:8

**Line(s):** 8

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/login/login.component.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #505: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/login/login.component.spec.ts:31
<b>Line(s):</b>	31
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/login/login.component.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #506: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/login/login.component.spec.ts:69
<b>Line(s):</b>	69
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/login/login.component.spec.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #507: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/server-started-notification/server-started-notificat...

**Line(s):** 7

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/server-started-notification/server-started-notification.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #508: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/server-started-notification/server-started-notificat...
<b>Line(s):</b>	19
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/server-started-notification/server-

started-notification.component.spec.ts**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #509: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/server-started-notification/server-started-notificat...
<b>Line(s):</b>	60
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor**Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/server-started-notification/server-started-notification.component.spec.ts**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #510: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/server-started-notification/server-started-notificat...
<b>Line(s):</b>	61
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClientTesting()
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/server-started-notification/server-started-notification.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #511: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/address.service.spec.ts:6
<b>Line(s):</b>	6
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpTestingController, provideHttpClientTesting } from
'@angular/common/http/testing'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/address.service.spec.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #512: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/address.service.spec.ts:9
<b>Line(s):</b>	9
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/address.service.spec.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #513: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)	
<b>CWE ID:</b>	CWE-306	<b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security	<b>Confidence:</b> MEDIUM
<b>Location:</b>	frontend/src/app/Services/address.service.spec.ts:15	
<b>Line(s):</b>	15	
<b>Exploitability:</b>	<a href="#">Unknown</a>	<b>Impact:</b> Unknown

#### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [AddressService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/address.service.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #514: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/address.service.spec.ts:15
<b>Line(s):</b>	15
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [AddressService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/address.service.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #515: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/address.service.ts:8
<b>Line(s):</b>	8
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpClient } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/address.service.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #516: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/address.service.ts:18
<b>Line(s):</b>	18
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
constructor (private readonly http: HttpClient) { }
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/address.service.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #517: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/basket.service.spec.ts:6

**Line(s):** 6

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpTestingController, provideHttpClientTesting } from
'@angular/common/http/testing'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/basket.service.spec.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #518: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/basket.service.spec.ts:10

**Line(s):** 10

**Exploitability:** Unknown      **Impact:** Unknown

#### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/basket.service.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #519: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/basket.service.spec.ts:16
<b>Line(s):</b>	16
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [BasketService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/basket.service.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #520: Angular - HttpClient Without Interceptor Auth

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-306
Category:	angular_security
Location:	frontend/src/app/Services/basket.service.spec.ts:16
Line(s):	16
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [BasketService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/basket.service.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #521: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/basket.service.ts:8
<b>Line(s):</b>	8
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpClient } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/basket.service.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #522: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/basket.service.ts:26
<b>Line(s):</b>	26
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
constructor (private readonly http: HttpClient) { }
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/basket.service.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #523: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/captcha.service.spec.ts:6

**Line(s):** 6

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpTestingController, provideHttpClientTesting } from
'@angular/common/http/testing'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/captcha.service.spec.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #524: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/captcha.service.spec.ts:10
<b>Line(s):</b>	10
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

#### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/captcha.service.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #525: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/captcha.service.spec.ts:16
<b>Line(s):</b>	16
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [CaptchaService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/captcha.service.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #526: Angular - HttpClient Without Interceptor Auth

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-306
Category:	angular_security
Location:	frontend/src/app/Services/captcha.service.spec.ts:16
Line(s):	16
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [CaptchaService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/captcha.service.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #527: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/captcha.service.ts:6
<b>Line(s):</b>	6
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpClient } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/captcha.service.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #528: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/captcha.service.ts:18
<b>Line(s):</b>	18
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
constructor (private readonly http: HttpClient) { }
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/captcha.service.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #529: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/challenge.service.spec.ts:6

**Line(s):** 6

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpTestingController, provideHttpClientTesting } from
'@angular/common/http/testing'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/challenge.service.spec.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #530: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/challenge.service.spec.ts:10
<b>Line(s):</b>	10
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

#### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/challenge.service.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #531: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/challenge.service.spec.ts:16
<b>Line(s):</b>	16
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [ChallengeService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/challenge.service.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #532: Angular - HttpClient Without Interceptor Auth

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-306
Category:	angular_security
Location:	frontend/src/app/Services/challenge.service.spec.ts:16
Line(s):	16
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [ChallengeService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/challenge.service.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #533: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/challenge.service.ts:7
<b>Line(s):</b>	7
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpClient } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/challenge.service.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #534: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/challenge.service.ts:19
<b>Line(s):</b>	19
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
constructor (private readonly http: HttpClient) { }
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/challenge.service.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #535: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/delivery.service.ts:8

**Line(s):** 8

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpClient } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/delivery.service.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #536: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	Confidence: MEDIUM
<b>Exploitability:</b>	frontend/src/app/Services/delivery.service.ts:29
<b>Impact:</b>	29
<b>Exploitability:</b>	Unknown
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
constructor (private readonly http: HttpClient) { }
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/delivery.service.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #537: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/feedback.service.spec.ts:6

**Line(s):** 6

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpTestingController, provideHttpClientTesting } from
'@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/feedback.service.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #538: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/feedback.service.spec.ts:10
<b>Line(s):</b>	10
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/feedback.service.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #539: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/feedback.service.spec.ts:16
<b>Line(s):</b>	16
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [FeedbackService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/feedback.service.spec.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #540: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	Confidence: MEDIUM
<b>Exploitability:</b>	frontend/src/app/Services/chatbot.service.spec.ts:16
<b>Impact:</b>	16
<b>Impact:</b>	Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [ChatbotService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/chatbot.service.spec.ts

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #541: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: **7.5/10.0**)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/chatbot.service.spec.ts:16

**Line(s):** 16

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [ChatbotService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

## tion Guidance

Review and fix the angular\_security issue in frontend/src/app/Services/chatbot.service.spec.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #542: Angular - HttpClient Without Interceptor Auth

**Severity:** HIGH (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/chatbot.service.ts:7

**Line(s):** 7

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpClient } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/chatbot.service.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #543: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/chatbot.service.ts:18
<b>Line(s):</b>	18
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
constructor (private readonly http: HttpClient) { }
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/chatbot.service.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #544: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/code-fixes.service.spec.ts:2
<b>Line(s):</b>	2
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpTestingController, provideHttpClientTesting } from
'@angular/common/http/testing'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/code-fixes.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #545: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/code-fixes.service.spec.ts:4
<b>Line(s):</b>	4
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/code-fixes.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #546: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)	
<b>CWE ID:</b>	CWE-306	<b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security	<b>Confidence:</b> MEDIUM
<b>Location:</b>	frontend/src/app/Services/code-fixes.service.spec.ts:12	
<b>Line(s):</b>	12	
<b>Exploitability:</b>	<a href="#">Unknown</a>	<b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [CodeFixesService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/code-fixes.service.spec.ts  
**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #547: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/code-fixes.service.spec.ts:12
<b>Line(s):</b>	12
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [CodeFixesService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/code-fixes.service.spec.ts  
**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #548: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/code-fixes.service.ts:3
<b>Line(s):</b>	3
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpClient } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/code-fixes.service.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #549: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/code-fixes.service.ts:26
<b>Line(s):</b>	26
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
constructor (private readonly http: HttpClient) { }
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/code-fixes.service.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #550: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/code-snippet.service.spec.ts:6

**Line(s):** 6

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpTestingController, provideHttpClientTesting } from
'@angular/common/http/testing'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/code-snippet.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #551: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)	
<b>CWE ID:</b>	CWE-306	<b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security	<b>Confidence:</b> MEDIUM
<b>Location:</b>	frontend/src/app/Services/code-snippet.service.spec.ts:10	
<b>Line(s):</b>	10	
<b>Exploitability:</b>	<a href="#">Unknown</a>	<b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/code-snippet.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #552: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/code-snippet.service.spec.ts:16
<b>Line(s):</b>	16
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [CodeSnippetService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/code-snippet.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #553: Angular - HttpClient Without Interceptor Auth

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-306
Category:	angular_security
Location:	frontend/src/app/Services/code-snippet.service.spec.ts:16
Line(s):	16
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [CodeSnippetService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/code-snippet.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #554: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/code-snippet.service.ts:8
<b>Line(s):</b>	8
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpClient } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/code-snippet.service.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #555: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/code-snippet.service.ts:28
<b>Line(s):</b>	28
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
constructor (private readonly http: HttpClient) { }
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/code-snippet.service.ts

**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #556: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/complaint.service.spec.ts:6

**Line(s):** 6

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpTestingController, provideHttpClientTesting } from
'@angular/common/http/testing'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/complaint.service.spec.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #557: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/complaint.service.spec.ts:10

**Line(s):** 10

**Exploitability:** Unknown      **Impact:** Unknown

#### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/complaint.service.spec.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #558: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/complaint.service.spec.ts:16
<b>Line(s):</b>	16
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [ComplaintService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/complaint.service.spec.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #559: Angular - HttpClient Without Interceptor Auth

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-306
Category:	angular_security
Location:	frontend/src/app/Services/complaint.service.spec.ts:16
Line(s):	16
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [ComplaintService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/complaint.service.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #560: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/complaint.service.ts:7
<b>Line(s):</b>	7
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpClient } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/complaint.service.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #561: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/complaint.service.ts:17
<b>Line(s):</b>	17
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
constructor (private readonly http: HttpClient) { }
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/complaint.service.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #562: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/configuration.service.spec.ts:7

**Line(s):** 7

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpTestingController, provideHttpClientTesting } from
'@angular/common/http/testing'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/configuration.service.spec.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #563: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)	
<b>CWE ID:</b>	CWE-306	<b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security	<b>Confidence:</b> MEDIUM
<b>Location:</b>	frontend/src/app/Services/configuration.service.spec.ts:9	
<b>Line(s):</b>	9	
<b>Exploitability:</b>	<a href="#">Unknown</a>	<b>Impact:</b> Unknown

#### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/configuration.service.spec.ts [References](#)

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #564: Angular - HttpClient Without Interceptor Auth

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-306
Category:	angular_security
Location:	frontend/src/app/Services/configuration.service.spec.ts:15
Line(s):	15
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

### Description

Detects HttpClient calls without auth interceptor [Vulnerable Code](#)

```
providers: [ConfigurationService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption [Remediation Guidance](#)

Review and fix the angular\_security issue in frontend/src/app/Services/configuration.service.spec.ts [References](#)

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #565: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/configuration.service.spec.ts:15
<b>Line(s):</b>	15
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [ConfigurationService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/configuration.service.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #566: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/configuration.service.ts:8
<b>Line(s):</b>	8
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpClient } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/configuration.service.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #567: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/configuration.service.ts:105
<b>Line(s):</b>	105
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
constructor (private readonly http: HttpClient) { }
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/configuration.service.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #568: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/country-mapping.service.spec.ts:6

**Line(s):** 6

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpTestingController, provideHttpClientTesting } from
'@angular/common/http/testing'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/country-mapping.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #569: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/country-mapping.service.spec.ts:10
<b>Line(s):</b>	10
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/country-mapping.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #570: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/country-mapping.service.spec.ts:16
<b>Line(s):</b>	16
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [CountryMappingService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/country-mapping.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #571: Angular - HttpClient Without Interceptor Auth

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-306
Category:	angular_security
Location:	frontend/src/app/Services/country-mapping.service.spec.ts:16
Line(s):	16
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [CountryMappingService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/country-mapping.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #572: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/country-mapping.service.ts:7
<b>Line(s):</b>	7
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpClient } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/country-mapping.service.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #573: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/country-mapping.service.ts:16
<b>Line(s):</b>	16
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
constructor (private readonly http: HttpClient) { }
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/country-mapping.service.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #574: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/data-subject.service.spec.ts:6

**Line(s):** 6

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpTestingController, provideHttpClientTesting } from
'@angular/common/http/testing'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/data-subject.service.spec.ts

#### References & Further Reading:

<https://owasp.org/Top10/A07/>

## #575: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/data-subject.service.spec.ts:9

**Line(s):** 9

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/data-subject.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #576: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/data-subject.service.spec.ts:15
<b>Line(s):</b>	15
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [DataSubjectService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/data-subject.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #577: Angular - HttpClient Without Interceptor Auth

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-306
Category:	angular_security
Location:	frontend/src/app/Services/data-subject.service.spec.ts:15
Line(s):	15
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [DataSubjectService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/data-subject.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #578: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/data-subject.service.ts:7
<b>Line(s):</b>	7
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpClient } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/data-subject.service.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #579: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/data-subject.service.ts:19
<b>Line(s):</b>	19
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
constructor (private readonly http: HttpClient) { }
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/data-

subject.service.ts

**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #580: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/delivery.service.spec.ts:6

**Line(s):** 6

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpTestingController, provideHttpClientTesting } from
'@angular/common/http/testing'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/delivery.service.spec.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #581: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)	
<b>CWE ID:</b>	CWE-306	<b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security	<b>Confidence:</b> MEDIUM
<b>Location:</b>	frontend/src/app/Services/delivery.service.spec.ts:9	
<b>Line(s):</b>	9	
<b>Exploitability:</b>	<a href="#">Unknown</a>	<b>Impact:</b> Unknown

#### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/delivery.service.spec.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #582: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/delivery.service.spec.ts:15
<b>Line(s):</b>	15
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [DeliveryService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/delivery.service.spec.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #583: Angular - HttpClient Without Interceptor Auth

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-306
Category:	angular_security
Location:	frontend/src/app/Services/delivery.service.spec.ts:15
Line(s):	15
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [DeliveryService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/delivery.service.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #584: Angular - HttpClient Without Interceptor Auth

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-306
Category:	angular_security
Location:	frontend/src/app/Services/product.service.spec.ts:6
Line(s):	6
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpTestingController, provideHttpClientTesting } from
'@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/product.service.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #585: Angular - HttpClient Without Interceptor Auth

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-306
Category:	angular_security
Location:	frontend/src/app/Services/feedback.service.spec.ts:16
Line(s):	16
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [FeedbackService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/feedback.service.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #586: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/feedback.service.ts:8
<b>Line(s):</b>	8
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpClient } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/feedback.service.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #587: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/feedback.service.ts:18
<b>Line(s):</b>	18
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
constructor (private readonly http: HttpClient) { }
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/feedback.service.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #588: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/image-captcha.service.spec.ts:6

**Line(s):** 6

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpTestingController, provideHttpClientTesting } from
'@angular/common/http/testing'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/image-captcha.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #589: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)	
<b>CWE ID:</b>	CWE-306	<b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security	<b>Confidence:</b> MEDIUM
<b>Location:</b>	frontend/src/app/Services/image-captcha.service.spec.ts:9	
<b>Line(s):</b>	9	
<b>Exploitability:</b>	<a href="#">Unknown</a>	<b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/image-captcha.service.spec.ts

**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #590: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/image-captcha.service.spec.ts:15
<b>Line(s):</b>	15
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor

### Vulnerable Code

```
providers: [ImageCaptchaService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption

**Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/image-captcha.service.spec.ts

**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #591: Angular - HttpClient Without Interceptor Auth

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-306
Category:	angular_security
Location:	frontend/src/app/Services/image-captcha.service.spec.ts:15
Line(s):	15
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [ImageCaptchaService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/image-captcha.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #592: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/image-captcha.service.ts:6
<b>Line(s):</b>	6
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpClient } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/image-captcha.service.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #593: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/image-captcha.service.ts:17
<b>Line(s):</b>	17
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
constructor (private readonly http: HttpClient) { }
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/image-

captcha.service.ts

**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #594: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/keys.service.ts:2

**Line(s):** 2

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpClient } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/keys.service.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #595: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/keys.service.ts:13

**Line(s):** 13

**Exploitability:** **Unknown**      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
constructor (private readonly http: HttpClient) {}
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/keys.service.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #596: Angular - HttpClient Without Interceptor Auth

**Severity:** HIGH (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/languages.service.spec.ts:6

**Line(s):** 6

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpTestingController, provideHttpClientTesting } from
'@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/languages.service.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #597: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/languages.service.spec.ts:10
<b>Line(s):</b>	10
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/languages.service.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #598: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/languages.service.spec.ts:15
<b>Line(s):</b>	15
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [LanguagesService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/languages.service.spec.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #599: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/languages.service.spec.ts:15
<b>Line(s):</b>	15
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [LanguagesService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/languages.service.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #600: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: **7.5/10.0**)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** **MEDIUM**

**Location:** frontend/src/app/Services/languages.service.ts:8

**Line(s):** 8

**Exploitability:** **Unknown**      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpClient } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

## tion Guidance

Review and fix the angular\_security issue in frontend/src/app/Services/languages.service.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #601: Angular - HttpClient Without Interceptor Auth

**Severity:** HIGH (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/languages.service.ts:16

**Line(s):** 16

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
constructor (private readonly http: HttpClient) { }
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/languages.service.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #602: Angular - HttpClient Without Interceptor Auth

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-306
Category:	angular_security
Location:	frontend/src/app/Services/order-history.service.spec.ts:6
Line(s):	6
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpTestingController, provideHttpClientTesting } from
'@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/order-history.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #603: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/order-history.service.spec.ts:9
<b>Line(s):</b>	9
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/order-history.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #604: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/order-history.service.spec.ts:15
<b>Line(s):</b>	15
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [OrderHistoryService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/order-history.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #605: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/order-history.service.spec.ts:15
<b>Line(s):</b>	15
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [OrderHistoryService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/order-history.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #606: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)	
<b>CWE ID:</b>	CWE-306	<b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security	<b>Confidence:</b> MEDIUM
<b>Location:</b>	frontend/src/app/Services/order-history.service.ts:8	
<b>Line(s):</b>	8	
<b>Exploitability:</b>	<a href="#">Unknown</a>	<b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpClient } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

## tion Guidance

Review and fix the angular\_security issue in frontend/src/app/Services/order-history.service.ts

**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #607: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/order-history.service.ts:18
<b>Line(s):</b>	18
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor

## Vulnerable Code

```
constructor (private readonly http: HttpClient) { }
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption

**Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/order-history.service.ts

**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #608: Angular - HttpClient Without Interceptor Auth

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-306
Category:	angular_security
Location:	frontend/src/app/Services/payment.service.spec.ts:6
Line(s):	6
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpTestingController, provideHttpClientTesting } from
'@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/payment.service.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #609: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/payment.service.spec.ts:9
<b>Line(s):</b>	9
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/payment.service.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #610: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/payment.service.spec.ts:15
<b>Line(s):</b>	15
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [PaymentService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/payment.service.spec.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #611: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/payment.service.spec.ts:15
<b>Line(s):</b>	15
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [PaymentService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/payment.service.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #612: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: **7.5/10.0**)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** **MEDIUM**

**Location:** frontend/src/app/Services/payment.service.ts:8

**Line(s):** 8

**Exploitability:** **Unknown**      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpClient } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

## tion Guidance

Review and fix the angular\_security issue in frontend/src/app/Services/payment.service.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #613: Angular - HttpClient Without Interceptor Auth

**Severity:** HIGH (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/payment.service.ts:18

**Line(s):** 18

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
constructor (private readonly http: HttpClient) { }
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/payment.service.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #614: Angular - HttpClient Without Interceptor Auth

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-306
Category:	angular_security
Location:	frontend/src/app/Services/photo-wall.service.spec.ts:6
Line(s):	6
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpTestingController, provideHttpClientTesting } from
'@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/photo-wall.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #615: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/photo-wall.service.spec.ts:9
<b>Line(s):</b>	9
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/photo-wall.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #616: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/photo-wall.service.spec.ts:15
<b>Line(s):</b>	15
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [PhotoWallService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/photo-wall.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #617: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/photo-wall.service.spec.ts:15
<b>Line(s):</b>	15
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [PhotoWallService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/photo-wall.service.spec.ts

**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #618: Angular - HttpClient Without Interceptor Auth

**Severity:** HIGH (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/photo-wall.service.ts:8

**Line(s):** 8

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpClient } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

## tion Guidance

Review and fix the angular\_security issue in frontend/src/app/Services/photo-wall.service.ts

**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #619: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/photo-wall.service.ts:18
<b>Line(s):</b>	18
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor

## Vulnerable Code

```
constructor (private readonly http: HttpClient) { }
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption

**Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/photo-wall.service.ts

**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #620: Angular - HttpClient Without Interceptor Auth

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-306
Category:	angular_security
Location:	frontend/src/app/Services/product-review.service.spec.ts:6
Line(s):	6
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpTestingController, provideHttpClientTesting } from
'@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/product-review.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #621: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/product-review.service.spec.ts:10
<b>Line(s):</b>	10
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/product-review.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #622: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/product-review.service.spec.ts:16
<b>Line(s):</b>	16
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [ProductReviewService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/product-review.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #623: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/product-review.service.spec.ts:16
<b>Line(s):</b>	16
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [ProductReviewService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/product-review.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #624: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)	
<b>CWE ID:</b>	CWE-306	<b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security	<b>Confidence:</b> MEDIUM
<b>Location:</b>	frontend/src/app/Services/product-review.service.ts:8	
<b>Line(s):</b>	8	
<b>Exploitability:</b>	<a href="#">Unknown</a>	<b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpClient } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

## tion Guidance

Review and fix the angular\_security issue in frontend/src/app/Services/product-review.service.ts

**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #625: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/product-review.service.ts:18
<b>Line(s):</b>	18
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor

### Vulnerable Code

```
constructor (private readonly http: HttpClient) { }
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption

**Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/product-review.service.ts

**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #626: Angular - HttpClient Without Interceptor Auth

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-306
Category:	angular_security
Location:	frontend/src/app/Services/vuln-lines.service.spec.ts:2
Line(s):	2
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpTestingController, provideHttpClientTesting } from
'@angular/common/http/testing'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/vuln-lines.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #627: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/vuln-lines.service.spec.ts:4
<b>Line(s):</b>	4
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/vuln-lines.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #628: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/product.service.spec.ts:10
<b>Line(s):</b>	10
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/product.service.spec.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #629: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/product.service.spec.ts:16

**Line(s):** 16

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [ProductService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/product.service.spec.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #630: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/product.service.spec.ts:16

**Line(s):** 16

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [ProductService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/product.service.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #631: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/product.service.ts:8
<b>Line(s):</b>	8
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpClient } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/product.service.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #632: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/product.service.ts:18
<b>Line(s):</b>	18
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
constructor (private readonly http: HttpClient) { }
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/product.service.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #633: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/quantity.service.spec.ts:6
<b>Line(s):</b>	6
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpTestingController, provideHttpClientTesting } from
'@angular/common/http/testing'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/quantity.service.spec.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #634: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	Confidence: MEDIUM
<b>Exploitability:</b>	frontend/src/app/Services/quantity.service.spec.ts:10
<b>Impact:</b>	10
<b>Impact:</b>	Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/quantity.service.spec.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #635: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)	
<b>CWE ID:</b>	CWE-306	<b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security	<b>Confidence:</b> MEDIUM
<b>Location:</b>	frontend/src/app/Services/quantity.service.spec.ts:16	
<b>Line(s):</b>	16	
<b>Exploitability:</b>	<a href="#">Unknown</a>	<b>Impact:</b> Unknown

#### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [QuantityService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/quantity.service.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #636: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/quantity.service.spec.ts:16
<b>Line(s):</b>	16
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [QuantityService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/quantity.service.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #637: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/quantity.service.ts:8
<b>Line(s):</b>	8
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpClient } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/quantity.service.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #638: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/quantity.service.ts:18
<b>Line(s):</b>	18
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
constructor (private readonly http: HttpClient) { }
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/quantity.service.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #639: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/recycle.service.spec.ts:6

**Line(s):** 6

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpTestingController, provideHttpClientTesting } from
'@angular/common/http/testing'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/recycle.service.spec.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #640: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/recycle.service.spec.ts:10
<b>Line(s):</b>	10
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

#### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/recycle.service.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #641: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/recycle.service.spec.ts:16
<b>Line(s):</b>	16
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [RecycleService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/recycle.service.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #642: Angular - HttpClient Without Interceptor Auth

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-306
Category:	angular_security
Location:	frontend/src/app/Services/recycle.service.spec.ts:16
Line(s):	16
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [RecycleService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/recycle.service.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #643: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/recycle.service.ts:7
<b>Line(s):</b>	7
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpClient } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/recycle.service.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #644: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/recycle.service.ts:18
<b>Line(s):</b>	18
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
constructor (private readonly http: HttpClient) { }
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/recycle.service.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #645: Angular - HTTP Interceptor Missing Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/request.interceptor.ts:6

**Line(s):** 6

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HTTP interceptors without authentication headers **Vulnerable Code**

```
import { type HttpEvent, type HttpHandler, type HttpInterceptor, type HttpRequest } from
'@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/request.interceptor.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #646: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/track-order.service.spec.ts:6

**Line(s):** 6

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpTestingController, provideHttpClientTesting } from
'@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/track-order.service.spec.ts  
**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #647: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/track-order.service.spec.ts:10
<b>Line(s):</b>	10
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor

### Vulnerable Code

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption

### Remediation Guidance

Review and fix the angular\_security issue in frontend/src/app/Services/track-order.service.spec.ts  
**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #648: Angular - HttpClient Without Interceptor Auth

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-306
Category:	angular_security
Location:	frontend/src/app/Services/track-order.service.spec.ts:16
Line(s):	16
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

## Description

Detects HttpClient calls without auth interceptor [Vulnerable Code](#)

```
providers: [TrackOrderService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/track-order.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #649: Angular - HttpClient Without Interceptor Auth

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-306
Category:	angular_security
Location:	frontend/src/app/Services/track-order.service.spec.ts:16
Line(s):	16
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [TrackOrderService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/track-order.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #650: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/track-order.service.ts:7
<b>Line(s):</b>	7
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpClient } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/track-order.service.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #651: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/track-order.service.ts:18
<b>Line(s):</b>	18
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
constructor (private readonly http: HttpClient) { }
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/track-

order.service.ts

**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #652: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/login/login.component.spec.ts:70

**Line(s):** 70

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClientTesting()
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/login/login.component.spec.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #653: Business Logic - Password Reset Token Reuse

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-640
<b>Category:</b>	business_logic
<b>Location:</b>	frontend/src/app/Services/user.service.spec.ts:121
<b>Line(s):</b>	121
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects password reset token reuse vulnerabilities **Vulnerable Code**

```
service.resetPassword(mockObject).subscribe((data) => (res = data))
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the business\_logic issue in frontend/src/app/Services/user.service.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #654: Angular - HttpClient Without Interceptor Auth

**Severity:** HIGH (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/user.service.spec.ts:6

**Line(s):** 6

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpTestingController, provideHttpClientTesting } from
'@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/user.service.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #655: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/user.service.spec.ts:10
<b>Line(s):</b>	10
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/user.service.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #656: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/user.service.spec.ts:16
<b>Line(s):</b>	16
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [UserService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/user.service.spec.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #657: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/user.service.spec.ts:16
<b>Line(s):</b>	16
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [UserService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/user.service.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #658: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: **7.5/10.0**)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** **MEDIUM**

**Location:** frontend/src/app/Services/user.service.ts:8

**Line(s):** 8

**Exploitability:** **Unknown**      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpClient } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

## tion Guidance

Review and fix the angular\_security issue in frontend/src/app/Services/user.service.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #659: Angular - HttpClient Without Interceptor Auth

**Severity:** HIGH (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/user.service.ts:26

**Line(s):** 26

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
constructor (private readonly http: HttpClient) { }
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/user.service.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #660: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/az_AZ.json:203
<b>Line(s):</b>	203
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"EXPORT_LABEL": " xrac format ",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/az\_AZ.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #661: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/az_AZ.json:287
<b>Line(s):</b>	287
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "(Veril nl riniz ixrac , yeni S yyah p nc r sind a lacaq.)",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/az\_AZ.json **References &**

## Further Reading:

<https://owasp.org/Top10/A02/>

## #662: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/bg\_BG.json:189

**Line(s):** 189

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": "Request Data Export",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/bg\_BG.json **References &**

#### Further Reading:

<https://owasp.org/Top10/A02/>

## #663: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: **7.5/10.0**)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** **MEDIUM**

**Location:** frontend/src/assets/i18n/bg\_BG.json:287

**Line(s):** 287

**Exploitability:** **Unknown**      **Impact:** Unknown

#### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "(Your data export will open in a new Browser window.)",
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/bg\_BG.json **References &**

**Further Reading:**

<https://owasp.org/Top10/A02/>

## #664: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/bg\_BG.json:203

**Line(s):** 203

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"EXPORT_LABEL": "Export Format",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/bg\_BG.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #665: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/bn_BD.json:189
<b>Line(s):</b>	189
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects weak or export-grade cipher suites

#### Vulnerable Code

```
"TITLE_REQUEST_DATA_EXPORT": "Request Data Export",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption  
**Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/bn\_BD.json

### Further Reading:

<https://owasp.org/Top10/A02/>

## #666: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/bn_BD.json:203
<b>Line(s):</b>	203
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"EXPORT_LABEL": "Export Format",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/bn\_BD.json **References &**

## Further Reading:

<https://owasp.org/Top10/A02/>

## #667: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/bn\_BD.json:287

**Line(s):** 287

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "(Your data export will open in a new Browser window.)",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/bn\_BD.json **References &**

#### Further Reading:

<https://owasp.org/Top10/A02/>

## #668: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: **7.5/10.0**)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** **MEDIUM**

**Location:** frontend/src/assets/i18n/ca\_ES.json:189

**Line(s):** 189

**Exploitability:** **Unknown**      **Impact:** Unknown

#### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": "Request Data Export",
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/ca\_ES.json **References &**

**Further Reading:**

<https://owasp.org/Top10/A02/>

## #669: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: **7.5/10.0**)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** **MEDIUM**

**Location:** frontend/src/assets/i18n/ca\_ES.json:203

**Line(s):** 203

**Exploitability:** **Unknown**      **Impact:** Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"EXPORT_LABEL": "Exportar formato",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/ca\_ES.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #670: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/ca_ES.json:287
<b>Line(s):</b>	287
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "(Your data export will open in a new Browser window.)",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/ca\_ES.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #671: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/cs_CZ.json:189
<b>Line(s):</b>	189
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": "dost o Export Dat",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/cs\_CZ.json **References &**

## Further Reading:

<https://owasp.org/Top10/A02/>

## #672: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327 **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/cs\_CZ.json:203

**Line(s):** 203

**Exploitability:** Unknown **Impact:** Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"EXPORT_LABEL": "Form t pro Export",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/cs\_CZ.json **References &**

#### Further Reading:

<https://owasp.org/Top10/A02/>

## #673: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327 **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/cs\_CZ.json:287

**Line(s):** 287

**Exploitability:** Unknown **Impact:** Unknown

#### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "(Export va ich dat bude otev en v nov m okn Prohl e e)",
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/cs\_CZ.json **References &**

**Further Reading:**

<https://owasp.org/Top10/A02/>

## #674: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/vuln-lines.service.spec.ts:10
<b>Line(s):</b>	10
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [VulnLinesService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/vuln-lines.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #675: Angular - HttpClient Without Interceptor Auth

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-306
Category:	angular_security
Location:	frontend/src/app/Services/vuln-lines.service.spec.ts:10
Line(s):	10
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [VulnLinesService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption  
**Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/vuln-lines.service.spec.ts  
**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #676: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/vuln-lines.service.ts:3
<b>Line(s):</b>	3
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpClient } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/vuln-lines.service.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #677: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/vuln-lines.service.ts:18
<b>Line(s):</b>	18
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
constructor (private readonly http: HttpClient) { }
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/vuln-lines.service.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #678: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/wallet.service.spec.ts:6

**Line(s):** 6

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpTestingController, provideHttpClientTesting } from
'@angular/common/http/testing'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/wallet.service.spec.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #679: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/wallet.service.spec.ts:9

**Line(s):** 9

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/wallet.service.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #680: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/wallet.service.spec.ts:15

**Line(s):** 15

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [WalletService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/wallet.service.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #681: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/wallet.service.spec.ts:15
<b>Line(s):</b>	15
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [WalletService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/wallet.service.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #682: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/wallet.service.ts:8
<b>Line(s):</b>	8
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpClient } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/wallet.service.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #683: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/wallet.service.ts:18
<b>Line(s):</b>	18
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
constructor (private readonly http: HttpClient) { }
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/wallet.service.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #684: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/sidenav/sidenav.component.spec.ts:13

**Line(s):** 13

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/sidenav/sidenav.component.spec.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #685: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/sidenav/sidenav.component.spec.ts:87

**Line(s):** 87

**Exploitability:** **Unknown**      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/sidenav/sidenav.component.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #686: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> MEDIUM
<b>Location:</b>	frontend/src/app/two-factor-auth-enter/two-factor-auth-enter.component.ts:14
<b>Line(s):</b>	14
<b>Exploitability:</b>	<b>Unknown</b>
	<b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/two-factor-auth-enter/two-factor-auth-enter.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #687: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/two-factor-auth-enter/two-factor-auth-enter.component.spec.ts:37
<b>Line(s):</b>	37
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/two-factor-auth-enter/two-factor-auth-enter.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #688: Vertical Access Control - Test Account in Production

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-489
<b>Category:</b>	access_control
<b>Location:</b>	frontend/src/app/oauth/oauth.component.spec.ts:82
<b>Line(s):</b>	82
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects test or demo accounts in production code **Vulnerable Code**

```
userService.oauthLogin.and.returnValue(of({ email: 'test@test.com' }))
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in frontend/src/app/oauth/oauth.component.spec.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A05/>

## #689: Vertical Access Control - Test Account in Production

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-489      **OWASP:** A05:2021 - Security Misconfiguration

**Category:** access\_control      **Confidence:** MEDIUM

**Location:** frontend/src/app/oauth/oauth.component.spec.ts:84

**Line(s):** 84

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects test or demo accounts in production code **Vulnerable Code**

```
expect(userService.save).toHaveBeenCalledWith({ email: 'test@test.com', password: 'bW9jLnRzZXRAdHNldA==', passwordRepeat: 'bW9jLnRzZXRAdHNldA==' })
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in frontend/src/app/oauth/oauth.component.spec.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A05/>

## #690: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/track-result/track-result.component.spec.ts:6

**Line(s):** 6

**Exploitability:** Unknown      **Impact:** Unknown

#### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/track-result/track-result.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #691: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/track-result/track-result.component.spec.ts:16
<b>Line(s):</b>	16
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/track-result/track-result.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #692: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/track-result/track-result.component.spec.ts:40
<b>Line(s):</b>	40
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/track-result/track-result.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #693: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/track-result/track-result.component.spec.ts:41
<b>Line(s):</b>	41
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClientTesting()
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/track-result/track-result.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #694: Vertical Access Control - Test Account in Production

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-489 <b>OWASP:</b> A05:2021 - Security Misconfiguration
<b>Category:</b>	access_control <b>Confidence:</b> MEDIUM
<b>Location:</b>	frontend/src/app/oauth/oauth.component.spec.ts:88
<b>Line(s):</b>	88
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects test or demo accounts in production code **Vulnerable Code**

```
userService.oauthLogin.and.returnValue(of({ email: 'test@test.com' }))
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in frontend/src/app/oauth/oauth.component.spec.ts

#### & Further Reading:

<https://owasp.org/Top10/A05/>

## #695: Vertical Access Control - Test Account in Production

**Severity:** HIGH (CVSS: 7.5/10.0)

**CWE ID:** CWE-489      **OWASP:** A05:2021 - Security Misconfiguration

**Category:** access\_control      **Confidence:** MEDIUM

**Location:** frontend/src/app/oauth/oauth.component.spec.ts:91

**Line(s):** 91

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects test or demo accounts in production code **Vulnerable Code**

```
expect(userService.login).toHaveBeenCalledWith({ email: 'test@test.com', password: 'bW9jLnRzZXRAdHNldA==', oauth: true })
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in frontend/src/app/oauth/oauth.component.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A05/>

## #696: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	frontend/src/app/user-details/user-details.component.spec.ts:9
<b>Exploitability:</b>	Confidence: MEDIUM
<b>Line(s):</b>	9
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/user-details/user-details.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #697: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/user-details/user-details.component.spec.ts:15
<b>Line(s):</b>	15
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/user-details/user-details.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #698: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/user-details/user-details.component.spec.ts:35
<b>Line(s):</b>	35
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/user-details/user-details.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #699: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/user-details/user-details.component.spec.ts:36

**Line(s):** 36

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClientTesting()
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/user-details/user-details.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #700: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	Confidence: MEDIUM
<b>Exploitability:</b>	frontend/src/app/wallet/wallet.component.spec.ts:7
<b>Impact:</b>	7
<b>Line(s):</b>	Unknown
<b>Exploitability:</b>	Impact: Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/wallet/wallet.component.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #701: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/wallet/wallet.component.spec.ts:22

**Line(s):** 22

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/wallet/wallet.component.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #702: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/wallet/wallet.component.spec.ts:56
<b>Line(s):</b>	56
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/wallet/wallet.component.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #703: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/wallet/wallet.component.spec.ts:57
<b>Line(s):</b>	57
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClientTesting()
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/wallet/wallet.component.spec.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #704: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/welcome/welcome.component.spec.ts:7

**Line(s):** 7

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/welcome/welcome.component.spec.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #705: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	Confidence: MEDIUM
<b>Exploitability:</b>	frontend/src/app/welcome/welcome.component.spec.ts:16
<b>Impact:</b>	16
<b>Impact:</b>	Unknown

#### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/welcome/welcome.component.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #706: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/welcome/welcome.component.spec.ts:40

**Line(s):** 40

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/welcome/welcome.component.spec.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #707: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/welcome/welcome.component.spec.ts:41
<b>Line(s):</b>	41
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClientTesting()
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/welcome/welcome.component.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #708: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/welcome-banner/welcome-banner.component.spec.ts:7
<b>Line(s):</b>	7
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/welcome-banner/welcome-

banner.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #709: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/welcome-banner/welcome-banner.component.spec.ts:18

**Line(s):** 18

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/welcome-banner/welcome-banner.component.spec.ts

#### References & Further Reading:

<https://owasp.org/Top10/A07/>

## #710: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/welcome-banner/welcome-banner.component.spec.ts:41

**Line(s):** 41

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor

#### Vulnerable Code

```
provideHttpClient(withInterceptorsFromDi()),
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/welcome-banner/welcome-

banner.component.spec.ts**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #711: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	frontend/src/app/welcome-banner/welcome-banner.component.spec.ts:42
<b>Exploitability:</b>	Confidence: MEDIUM
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor**Vulnerable Code**

```
provideHttpClientTesting()
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/welcome-banner/welcome-banner.component.spec.ts**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #712: Data Exposure - PII in Logs

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-532
Category:	data_exposure
Location:	frontend/src/app/oauth/oauth.component.spec.ts:91
Line(s):	91
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

### Description

Detects Personally Identifiable Information in log statements [Vulnerable Code](#)

```
expect(userService.login).toHaveBeenCalledWith({ email: 'test@test.com', password: 'bW9jLnRzZXRAdHNldA==', oauth: true })
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the data\_exposure issue in frontend/src/app/oauth/oauth.component.spec.ts [References](#)

### & Further Reading:

<https://owasp.org/Top10/A09/>

## #713: Crypto - Insecure Random Number Generator

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-338
<b>Category:</b>	cryptography
<b>Location:</b>	routes/captcha.ts:15
<b>Line(s):</b>	15
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects usage of insecure random number generators **Vulnerable Code**

```
const firstTerm = Math.floor((Math.random() * 10) + 1)
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in routes/captcha.ts **References & Further Reading:**

<https://owasp.org/Top10/A02/>

## #714: Crypto - Insecure Random Number Generator

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-338
<b>Category:</b>	cryptography
<b>Location:</b>	routes/captcha.ts:16
<b>Line(s):</b>	16
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects usage of insecure random number generators **Vulnerable Code**

```
const secondTerm = Math.floor((Math.random() * 10) + 1)
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption

## Remediation Guidance

Review and fix the cryptography issue in routes/captcha.ts **References & Further Reading:**

<https://owasp.org/Top10/A02/>

## #715: Crypto - Insecure Random Number Generator

**Severity:** HIGH (CVSS: 7.5/10.0)

**CWE ID:** CWE-338      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** routes/captcha.ts:17

**Line(s):** 17

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects usage of insecure random number generators **Vulnerable Code**

```
const thirdTerm = Math.floor((Math.random() * 10) + 1)
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in routes/captcha.ts **References & Further Reading:**

<https://owasp.org/Top10/A02/>

## #716: Crypto - Insecure Random Number Generator

**Severity:** **HIGH** (CVSS: **7.5/10.0**)

**CWE ID:** CWE-338 **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography **Confidence:** **MEDIUM**

**Location:** routes/captcha.ts:19

**Line(s):** 19

**Exploitability:** **Unknown** **Impact:** Unknown

### Description

Detects usage of insecure random number generators **Vulnerable Code**

```
const firstOperator = operators[Math.floor((Math.random() * 3))]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in routes/captcha.ts **References & Further Reading:**

<https://owasp.org/Top10/A02/>

## #717: Crypto - Insecure Random Number Generator

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)	
CWE ID:	CWE-338	OWASP: A02:2021 - Cryptographic Failures
Category:	cryptography	Confidence: <b>MEDIUM</b>
Location:	routes/captcha.ts:20	
Line(s):	20	
Exploitability:	<a href="#">Unknown</a>	Impact: Unknown

### Description

Detects usage of insecure random number generators **Vulnerable Code**

```
const secondOperator = operators[Math.floor((Math.random() * 3))]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in routes/captcha.ts **References & Further Reading:**

<https://owasp.org/Top10/A02/>

## #718: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/oauth/oauth.component.spec.ts:14
<b>Line(s):</b>	14
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/oauth/oauth.component.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #719: Vertical Access Control - Test Account in Production

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-489
<b>Category:</b>	access_control
<b>Location:</b>	test/api/deluxeApiSpec.ts:70
<b>Line(s):</b>	70
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects test or demo accounts in production code **Vulnerable Code**

```
email: 'admin@' + config.get<string>('application.domain'),
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in test/api/deluxeApiSpec.ts **References & Further Reading:**

### Reading:

<https://owasp.org/Top10/A05/>

## #720: Vertical Access Control - Test Account in Production

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-489      **OWASP:** A05:2021 - Security Misconfiguration

**Category:** access\_control      **Confidence:** MEDIUM

**Location:** test/api/deluxeApiSpec.ts:169

**Line(s):** 169

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects test or demo accounts in production code **Vulnerable Code**

```
email: 'admin@' + config.get<string>('application.domain'),
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in test/api/deluxeApiSpec.ts **References & Further**

#### Reading:

<https://owasp.org/Top10/A05/>

## #721: Session Management - Cookie Without Secure Flag

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-614 **OWASP:** A05:2021 - Security Misconfiguration

**Category:** authentication **Confidence:** MEDIUM

**Location:** routes/updateUserProfile.ts:40

**Line(s):** 40

**Exploitability:** Unknown **Impact:** Unknown

### Description

Detects cookies without secure flag **Vulnerable Code**

```
res.cookie('token', updatedToken)
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the authentication issue in routes/updateUserProfile.ts **References & Further**

**Reading:**

<https://owasp.org/Top10/A05/>

## #722: Misconfig - Weak Content Security Policy

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-1021      **OWASP:** A05:2021 - Security Misconfiguration

**Category:** misconfiguration      **Confidence:** MEDIUM

**Location:** routes/userProfile.ts:90

**Line(s):** 90

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects weak or missing Content Security Policy **Vulnerable Code**

```
return username && user?.profileImage.match(/;[]*script-src(.*)*'unsafe-inline'/g) !== null
&& utils.contains(username, '<script>alert('xss')</script>')
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the misconfiguration issue in routes/userProfile.ts **References & Further**

**Reading:**

<https://owasp.org/Top10/A05/>

## #723: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/oauth/oauth.component.spec.ts:25
<b>Line(s):</b>	25
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/oauth/oauth.component.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #724: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/oauth/oauth.component.spec.ts:57
<b>Line(s):</b>	57
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/oauth/oauth.component.spec.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A07/>

## #725: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/oauth/oauth.component.spec.ts:58

**Line(s):** 58

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClientTesting()
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/oauth/oauth.component.spec.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A07/>

## #726: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	Confidence: MEDIUM
<b>Line(s):</b>	frontend/src/app/Services/administration.service.spec.ts:6
<b>Exploitability:</b>	6
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpTestingController, provideHttpClientTesting } from
'@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/administration.service.spec.ts [References](#)

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #727: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/administration.service.spec.ts:10
<b>Line(s):</b>	10
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor [Vulnerable Code](#)

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption [Remediation Guidance](#)

### Remediation Guidance

Review and fix the angular\_security issue in frontend/src/app/Services/administration.service.spec.ts [References](#)

**& Further Reading:**

<https://owasp.org/Top10/A07/>

## #728: Angular - HttpClient Without Interceptor Auth

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-306
Category:	angular_security
Location:	frontend/src/app/Services/administration.service.spec.ts:16
Line(s):	16
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [AdministrationService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/administration.service.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #729: Angular - HttpClient Without Interceptor Auth

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-306
Category:	angular_security
Location:	frontend/src/app/Services/administration.service.spec.ts:16
Line(s):	16
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [AdministrationService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/administration.service.spec.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #730: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/administration.service.ts:7
<b>Line(s):</b>	7
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpClient } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/administration.service.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A07/>

## #731: Vertical Access Control - Test Account in Production

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-489
<b>Category:</b>	access_control
<b>Location:</b>	test/api/loginApiSpec.ts:63
<b>Line(s):</b>	63
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects test or demo accounts in production code **Vulnerable Code**

```
email: 'admin@' + config.get<string>('application.domain'),
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in test/api/loginApiSpec.ts **References & Further**

## Reading:

<https://owasp.org/Top10/A05/>

## #732: Vertical Access Control - Test Account in Production

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-489      **OWASP:** A05:2021 - Security Misconfiguration

**Category:** access\_control      **Confidence:** MEDIUM

**Location:** test/api/loginApiSpec.ts:167

**Line(s):** 167

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects test or demo accounts in production code **Vulnerable Code**

```
it('POST login with known email "admin@juice-sh.op" in SQL injection attack', () => {
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in test/api/loginApiSpec.ts **References & Further**

#### Reading:

<https://owasp.org/Top10/A05/>

## #733: Vertical Access Control - Test Account in Production

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-489      **OWASP:** A05:2021 - Security Misconfiguration

**Category:** access\_control      **Confidence:** MEDIUM

**Location:** test/api/loginApiSpec.ts:171

**Line(s):** 171

**Exploitability:** Unknown      **Impact:** Unknown

#### Description

Detects test or demo accounts in production code **Vulnerable Code**

```
email: 'admin@' + config.get<string>('application.domain') + '\>--',
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in test/api/loginApiSpec.ts **References & Further**

**Reading:**

<https://owasp.org/Top10/A05/>

## #734: Data Exposure - PII in Logs

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-532      **OWASP:** A09:2021 - Security Logging and Monitoring Failures

**Category:** data\_exposure      **Confidence:** MEDIUM

**Location:** test/api/loginApiSpec.ts:167

**Line(s):** 167

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects Personally Identifiable Information in log statements **Vulnerable Code**

```
it('POST login with known email "admin@juice-sh.op" in SQL injection attack', () => {
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the data\_exposure issue in test/api/loginApiSpec.ts **References & Further**

**Reading:**

<https://owasp.org/Top10/A09/>

## #735: Data Exposure - PII in Logs

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-532
Category:	data_exposure
Location:	test/api/loginApiSpec.ts:182
Line(s):	182
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

### Description

Detects Personally Identifiable Information in log statements **Vulnerable Code**

```
it('POST login with known email "jim@juice-sh.op" in SQL injection attack', () => {
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the data\_exposure issue in test/api/loginApiSpec.ts **References & Further**

### Reading:

<https://owasp.org/Top10/A09/>

## #736: Data Exposure - PII in Logs

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-532
Category:	data_exposure
Location:	test/api/loginApiSpec.ts:197
Line(s):	197
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

## Description

Detects Personally Identifiable Information in log statements **Vulnerable Code**

```
it('POST login with known email "bender@juice-sh.op" in SQL injection attack', () => {
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the data\_exposure issue in test/api/loginApiSpec.ts **References & Further Reading:**

### Reading:

<https://owasp.org/Top10/A09/>

## #737: Data Exposure - PII in Logs

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-532      **OWASP:** A09:2021 - Security Logging and Monitoring Failures

**Category:** data\_exposure      **Confidence:** MEDIUM

**Location:** test/api/loginApiSpec.ts:212

**Line(s):** 212

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects Personally Identifiable Information in log statements **Vulnerable Code**

```
it('POST login with non-existing email "acc0unt4nt@juice-sh.op" via UNION SELECT injection attack', () => {
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the data\_exposure issue in test/api/loginApiSpec.ts **References & Further**

#### Reading:

<https://owasp.org/Top10/A09/>

## #738: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/administration.service.ts:18

**Line(s):** 18

**Exploitability:** Unknown      **Impact:** Unknown

#### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
constructor (private readonly http: HttpClient) { }
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/administration.service.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #739: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/security-answer.service.spec.ts:6
<b>Line(s):</b>	6
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpTestingController, provideHttpClientTesting } from
'@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/security-answer.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #740: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/security-answer.service.spec.ts:10
<b>Line(s):</b>	10
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/security-answer.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #741: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/security-answer.service.spec.ts:16
<b>Line(s):</b>	16
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [SecurityAnswerService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/security-answer.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #742: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	<b>MEDIUM</b>
<b>Exploitability:</b>	frontend/src/app/Services/security-answer.service.spec.ts:16
<b>Impact:</b>	16
<b>Impact:</b>	Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [SecurityAnswerService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/security-answer.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #743: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)	
<b>CWE ID:</b>	CWE-306	<b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security	<b>Confidence:</b> MEDIUM
<b>Location:</b>	frontend/src/app/Services/security-answer.service.ts:7	
<b>Line(s):</b>	7	
<b>Exploitability:</b>	<a href="#">Unknown</a>	<b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpClient } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

## tion Guidance

Review and fix the angular\_security issue in frontend/src/app/Services/security-answer.service.ts

**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #744: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/security-answer.service.ts:18
<b>Line(s):</b>	18
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor

## Vulnerable Code

```
constructor (private readonly http: HttpClient) { }
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption

**Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/security-answer.service.ts

**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #745: Angular - HttpClient Without Interceptor Auth

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-306
Category:	angular_security
Location:	frontend/src/app/Services/security-question.service.spec.ts:6
Line(s):	6
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpTestingController, provideHttpClientTesting } from
'@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/security-question.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #746: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/security-question.service.spec.ts:10
<b>Line(s):</b>	10
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/security-question.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #747: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/Services/security-question.service.spec.ts:16
<b>Line(s):</b>	16
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [SecurityQuestionService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/security-question.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #748: Vertical Access Control - Test Account in Production

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-489      **OWASP:** A05:2021 - Security Misconfiguration

**Category:** access\_control      **Confidence:** MEDIUM

**Location:** test/server/currentUserSpec.ts:37

**Line(s):** 37

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects test or demo accounts in production code **Vulnerable Code**

```
{ data: { id: 1, email: 'admin@juice-sh.op', lastLoginIp: '0.0.0.0', profileImage: '/assets/public/images/uploads/default.svg' } as unknown as UserModel }
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in test/server/currentUserSpec.ts **References & Further**

### Reading:

<https://owasp.org/Top10/A05/>

## #749: Vertical Access Control - Test Account in Production

**Severity:** **HIGH** (CVSS: **7.5/10.0**)

**CWE ID:** CWE-489      **OWASP:** A05:2021 - Security Misconfiguration

**Category:** access\_control      **Confidence:** **MEDIUM**

**Location:** test/server/currentUserSpec.ts:41

**Line(s):** 41

**Exploitability:** **Unknown**      **Impact:** Unknown

### Description

Detects test or demo accounts in production code **Vulnerable Code**

```
expect(res.json).to.have.been.calledWith({ user: { id: 1, email: 'admin@juice-sh.op', lastLoginIp: '0.0.0.0', profileImage: '/assets/public/images/uploads/default.svg' } })
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation**

## tion Guidance

Review and fix the access\_control issue in test/server/currentUserSpec.ts **References & Further Reading:**

### Reading:

<https://owasp.org/Top10/A05/>

## #750: Angular - HttpClient Without Interceptor Auth

**Severity:** HIGH (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/security-question.service.spec.ts:16

**Line(s):** 16

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [SecurityQuestionService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/security-question.service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #751: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/security-question.service.ts:6
<b>Line(s):</b>	6
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpClient } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/security-question.service.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #752: Vertical Access Control - Test Account in Production

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-489
<b>Category:</b>	access_control
<b>Location:</b>	test/server/verifySpec.ts:42
<b>Line(s):</b>	42
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects test or demo accounts in production code **Vulnerable Code**

```
email: 'test@juice-sh.op'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in test/server/verifySpec.ts **References & Further**

## Reading:

<https://owasp.org/Top10/A05/>

## #753: Vertical Access Control - Test Account in Production

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-489      **OWASP:** A05:2021 - Security Misconfiguration

**Category:** access\_control      **Confidence:** MEDIUM

**Location:** test/api/orderHistoryApiSpec.ts:18

**Line(s):** 18

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects test or demo accounts in production code **Vulnerable Code**

```
email: 'admin@' + config.get<string>('application.domain'),
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in test/api/orderHistoryApiSpec.ts **References & Further Reading:**

#### Further Reading:

<https://owasp.org/Top10/A05/>

## #754: SSRF - Grafana Dashboard

**Severity:** HIGH (CVSS: 7.5/10.0)

**CWE ID:** CWE-918      **OWASP:** A10:2021 - Server-Side Request Forgery

**Category:** ssrf      **Confidence:** MEDIUM

**Location:** config/default.yml:2

**Line(s):** 2

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects SSRF to Grafana instances **Vulnerable Code**

port: 3000

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the ssrf issue in config/default.yml **References & Further Reading:**

## #755: Data Exposure - Backup Files Accessible

Severity:	HIGH (CVSS: 7.5/10.0)
CWE ID:	CWE-530
Category:	data_exposure
Location:	config/default.yml:464
Line(s):	464
Exploitability:	Unknown
Impact:	Unknown

### Description

Detects accessible backup file patterns **Vulnerable Code**

```
countryMapping: ~
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the data\_exposure issue in config/default.yml **References & Further Reading:**

<https://owasp.org/Top10/A05/>

## #756: Vertical Access Control - Test Account in Production

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-489 <b>OWASP:</b> A05:2021 - Security Misconfiguration
<b>Category:</b>	access_control <b>Confidence:</b> MEDIUM
<b>Location:</b>	test/api/orderHistoryApiSpec.ts:72
<b>Line(s):</b>	72
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

### Description

Detects test or demo accounts in production code

```
email: 'admin@' + config.get<string>('application.domain'),
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption

**Remediation Guidance**

Review and fix the access\_control issue in test/api/orderHistoryApiSpec.ts

### Further Reading:

<https://owasp.org/Top10/A05/>

## #757: Vertical Access Control - Test Account in Production

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-489 <b>OWASP:</b> A05:2021 - Security Misconfiguration
<b>Category:</b>	access_control <b>Confidence:</b> MEDIUM
<b>Location:</b>	test/api/orderHistoryApiSpec.ts:108
<b>Line(s):</b>	108
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

## Description

Detects test or demo accounts in production code **Vulnerable Code**

```
email: 'admin@' + config.get<string>('application.domain'),
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in test/api/orderHistoryApiSpec.ts **References &**

## Further Reading:

<https://owasp.org/Top10/A05/>

## #758: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/security-question.service.ts:18

**Line(s):** 18

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
constructor (private readonly http: HttpClient) { }
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/security-question.service.ts

#### References & Further Reading:

<https://owasp.org/Top10/A07/>

## #759: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/Services/two-factor-auth-service.spec.ts:6

**Line(s):** 6

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects HttpClient calls without auth interceptor

### Vulnerable Code

```
import { HttpTestingController, provideHttpClientTesting } from
'@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/two-factor-auth-service.spec.ts  
**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #760: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/two-factor-auth-service.spec.ts:10
<b>Line(s):</b>	10
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor

### Vulnerable Code

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption

### Remediation Guidance

Review and fix the angular\_security issue in frontend/src/app/Services/two-factor-auth-service.spec.ts  
**References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #761: Angular - HttpClient Without Interceptor Auth

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-306
Category:	angular_security
Location:	frontend/src/app/Services/two-factor-auth-service.spec.ts:15
Line(s):	15
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [TwoFactorAuthService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/two-factor-auth-service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #762: Angular - HttpClient Without Interceptor Auth

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-306
Category:	angular_security
Location:	frontend/src/app/Services/two-factor-auth-service.spec.ts:15
Line(s):	15
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
providers: [TwoFactorAuthService, provideHttpClient(withInterceptorsFromDi()),
provideHttpClientTesting()]
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/two-factor-auth-service.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #763: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	angular_security
<b>Location:</b>	frontend/src/app/Services/two-factor-auth-service.ts:7
<b>Line(s):</b>	7
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { HttpClient } from '@angular/common/http'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/two-factor-auth-service.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #764: CSRF - Missing Origin Validation

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-352
<b>Category:</b>	csrf
<b>Location:</b>	server.ts:598
<b>Line(s):</b>	598
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post('/rest/user/data-export', security.appendUserId(), verifyImageCaptcha())
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

## #765: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: **7.5/10.0**)

**CWE ID:** **CWE-306** **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security **Confidence:** **MEDIUM**

**Location:** frontend/src/app/Services/two-factor-auth-service.ts:33

**Line(s):** 33

**Exploitability:** **Unknown** **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
constructor (private readonly http: HttpClient) {}
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/Services/two-factor-auth-service.ts

#### References & Further Reading:

<https://owasp.org/Top10/A07/>

## #766: API - Private Field Exposure

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-200      **OWASP:** A01:2021 - Broken Access Control

**Category:** api\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/two-factor-auth/two-factor-auth.component.spec.ts:84

**Line(s):** 84

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects private/internal fields in responses

### Vulnerable Code

```
twoFactorAuthService.status.and.returnValue(of({ setup: false, email: 'email', secret: 'secret', setupToken: '12345' }))
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the api\_security issue in frontend/src/app/two-factor-auth/two-factor-auth.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

## #767: API - Private Field Exposure

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-200 <b>OWASP:</b> A01:2021 - Broken Access Control
<b>Category:</b>	api_security <b>Confidence:</b> MEDIUM
<b>Location:</b>	frontend/src/app/two-factor-auth/two-factor-auth.component.spec.ts:95
<b>Line(s):</b>	95
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects private/internal fields in responses **Vulnerable Code**

```
twoFactorAuthService.status.and.returnValue(of({ setup: true, email: 'email', secret: 'secret', setupToken: '12345' }))
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the api\_security issue in frontend/src/app/two-factor-auth/two-factor-auth.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

## #768: API - Private Field Exposure

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-200
<b>Category:</b>	api_security
<b>Location:</b>	frontend/src/app/two-factor-auth/two-factor-auth.component.spec.ts:134
<b>Line(s):</b>	134
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects private/internal fields in responses

### Vulnerable Code

```
twoFactorAuthService.status.and.returnValue(of({ setup: true, email: 'email', secret: 'secret', setupToken: '12345' }))
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption  
**Remediation Guidance**

Review and fix the api\_security issue in frontend/src/app/two-factor-auth/two-factor-auth.component.spec.ts

### References & Further Reading:

<https://owasp.org/Top10/A01/>

## #769: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/two-factor-auth/two-factor-auth.component.spec.ts:10
<b>Line(s):</b>	10
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClientTesting } from '@angular/common/http/testing'
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/two-factor-auth/two-factor-auth.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #770: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-306 <b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	angular_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/two-factor-auth/two-factor-auth.component.spec.ts:33
<b>Line(s):</b>	33
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
import { provideHttpClient, withInterceptorsFromDi } from '@angular/common/http'
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/two-factor-auth/two-factor-auth.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #771: Angular - HttpClient Without Interceptor Auth

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-306      **OWASP:** A07:2021 - Identification and Authentication Failures

**Category:** angular\_security      **Confidence:** MEDIUM

**Location:** frontend/src/app/two-factor-auth/two-factor-auth.component.spec.ts:66

**Line(s):** 66

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/two-factor-auth/two-factor-auth.component.spec.ts

#### References & Further Reading:

<https://owasp.org/Top10/A07/>

## #772: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	Confidence: MEDIUM
<b>Exploitability:</b>	frontend/src/app/two-factor-auth/two-factor-auth.component.spec.ts:67
<b>Impact:</b>	67
<b>Line(s):</b>	Unknown
<b>Exploitability:</b>	Impact: Unknown

### Description

Detects HttpClient calls without auth interceptor

#### Vulnerable Code

```
provideHttpClientTesting()
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/two-factor-auth/two-factor-

auth.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #773: API - No Rate Limit on Email Sending

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-770
<b>Category:</b>	api_security
<b>Location:</b>	data/static/codefixes/loginBenderChallenge_2_correct.ts:34
<b>Line(s):</b>	34
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

### Description

Detects email endpoints without rate limiting **Vulnerable Code**

```
res.status(401).send(res.__('Invalid email or password.'))
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the api\_security issue in data/static/codefixes/loginBenderChallenge\_2\_correct.ts **References & Further Reading:**

<https://owasp.org/Top10/A04/>

## #774: CSRF - Missing Origin Validation

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-352
<b>Category:</b>	csrf
<b>Location:</b>	server.ts:380
<b>Line(s):</b>	380
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post('/api/SecurityQuestions', security.denyAll())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

## #775: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327
<b>Category:</b>	cryptography
<b>Location:</b>	frontend/src/assets/i18n/da_DK.json:189
<b>Line(s):</b>	189
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": "Anmod om Data Eksport",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/da\_DK.json **References &**

## Further Reading:

<https://owasp.org/Top10/A02/>

## #776: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327 **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/da\_DK.json:203

**Line(s):** 203

**Exploitability:** Unknown **Impact:** Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"EXPORT_LABEL": "Eksport Format",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/da\_DK.json **References &**

#### Further Reading:

<https://owasp.org/Top10/A02/>

## #777: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: **7.5/10.0**)

**CWE ID:** CWE-327 **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography **Confidence:** **MEDIUM**

**Location:** frontend/src/assets/i18n/da\_DK.json:287

**Line(s):** 287

**Exploitability:** **Unknown** **Impact:** Unknown

#### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "(Dit dataekspor til bne i et nyt browservindue.)",
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/da\_DK.json **References &**

**Further Reading:**

<https://owasp.org/Top10/A02/>

## #778: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/de\_CH.json:189

**Line(s):** 189

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": "Datenexport anfordern",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/de\_CH.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #779: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327
<b>Category:</b>	cryptography
<b>Location:</b>	frontend/src/assets/i18n/de_CH.json:203
<b>Line(s):</b>	203
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"EXPORT_LABEL": "Export-Format",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/de\_CH.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #780: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327
<b>Category:</b>	cryptography
<b>Location:</b>	frontend/src/assets/i18n/de_CH.json:287
<b>Line(s):</b>	287
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "(Ihr Datenexport wird in einem neuen Browser-Fenster ge ffnet.)",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/de\_CH.json **References & Further Reading:**

<https://owasp.org/Top10/A02/>

## #781: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/de\_DE.json:189

**Line(s):** 189

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": "Datenexport anfordern",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/de\_DE.json **References &**

#### Further Reading:

<https://owasp.org/Top10/A02/>

## #782: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: **7.5/10.0**)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** **MEDIUM**

**Location:** frontend/src/assets/i18n/de\_DE.json:203

**Line(s):** 203

**Exploitability:** **Unknown**      **Impact:** Unknown

#### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"EXPORT_LABEL": "Export-Format",
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/de\_DE.json **References &**

**Further Reading:**

<https://owasp.org/Top10/A02/>

## #783: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: **7.5/10.0**)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** **MEDIUM**

**Location:** frontend/src/assets/i18n/de\_DE.json:287

**Line(s):** 287

**Exploitability:** **Unknown**      **Impact:** Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "(Ihr Datenexport wird in einem neuen Browser-Fenster ge ffnet.)",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/de\_DE.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #784: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327
<b>Category:</b>	cryptography
<b>Location:</b>	frontend/src/assets/i18n/el_GR.json:189
<b>Line(s):</b>	189
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects weak or export-grade cipher suites

### Vulnerable Code

```
"TITLE_REQUEST_DATA_EXPORT": " ",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption

**Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/el\_GR.json

### Further Reading:

<https://owasp.org/Top10/A02/>

## #785: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327
<b>Category:</b>	cryptography
<b>Location:</b>	frontend/src/assets/i18n/el_GR.json:203
<b>Line(s):</b>	203
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects weak or export-grade cipher suites

## Vulnerable Code

```
"EXPORT_LABEL": " ",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption

## Remediation Guidance

Review and fix the cryptography issue in frontend/src/assets/i18n/el\_GR.json

## References &

## Further Reading:

<https://owasp.org/Top10/A02/>

## #786: TLS - Weak Cipher Suite Configuration

**Severity:** HIGH (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/el\_GR.json:287

**Line(s):** 287

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects weak or export-grade cipher suites

## Vulnerable Code

```
"DATA_EXPORT_HINT": "(Your data export will open in a new Browser window.)",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/en.json **References & Further**

#### Further Reading:

<https://owasp.org/Top10/A02/>

## #787: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327 **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/en.json:189

**Line(s):** 189

**Exploitability:** Unknown **Impact:** Unknown

#### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": "Request Data Export",
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/en.json **References & Further**

**Reading:**

<https://owasp.org/Top10/A02/>

## #788: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/en.json:203

**Line(s):** 203

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"EXPORT_LABEL": "Export Format",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/en.json **References & Further**

**Reading:**

<https://owasp.org/Top10/A02/>

## #789: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/en.json:287
<b>Line(s):</b>	287
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "(Your data export will open in a new Browser window.)",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/en.json **References & Further**

### Reading:

<https://owasp.org/Top10/A02/>

## #790: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/es_ES.json:189
<b>Line(s):</b>	189
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": "Enviar solicitud para exportar los datos",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/es\_ES.json **References &**

## Further Reading:

<https://owasp.org/Top10/A02/>

## #791: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/es\_ES.json:203

**Line(s):** 203

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"EXPORT_LABEL": "Exportar en Formato",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/es\_ES.json **References &**

#### Further Reading:

<https://owasp.org/Top10/A02/>

## #792: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/es\_ES.json:287

**Line(s):** 287

**Exploitability:** Unknown      **Impact:** Unknown

#### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "(Los datos exportados se abren en una nueva ventana del Browser.)",
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/es\_ES.json **References &**

**Further Reading:**

<https://owasp.org/Top10/A02/>

## #793: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: **7.5/10.0**)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** **MEDIUM**

**Location:** frontend/src/assets/i18n/et\_EE.json:189

**Line(s):** 189

**Exploitability:** **Unknown**      **Impact:** Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": "Andmete eksport",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/et\_EE.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #794: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327
<b>Category:</b>	cryptography
<b>Location:</b>	frontend/src/assets/i18n/et_EE.json:203
<b>Line(s):</b>	203
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects weak or export-grade cipher suites

### Vulnerable Code

```
"EXPORT_LABEL": "Ekspordivorming",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption  
**Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/et\_EE.json

### Further Reading:

<https://owasp.org/Top10/A02/>

## #795: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327
<b>Category:</b>	cryptography
<b>Location:</b>	frontend/src/assets/i18n/et_EE.json:287
<b>Line(s):</b>	287
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "(Andmete eksport avaneb uues veeblehitseja aknas)",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/et\_EE.json **References & Further Reading:**

<https://owasp.org/Top10/A02/>

## #796: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/fi\_FI.json:189

**Line(s):** 189

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": "Pyyd tietojasi j rjestelm st ",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/fi\_FI.json **References &**

#### Further Reading:

<https://owasp.org/Top10/A02/>

## #797: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/fi\_FI.json:203

**Line(s):** 203

**Exploitability:** Unknown      **Impact:** Unknown

#### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"EXPORT_LABEL": "Tietojen tuontiasetukset",
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/fi\_FI.json **References &**

**Further Reading:**

<https://owasp.org/Top10/A02/>

## #798: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: **7.5/10.0**)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** **MEDIUM**

**Location:** frontend/src/assets/i18n/fi\_FI.json:287

**Line(s):** 287

**Exploitability:** **Unknown**      **Impact:** Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "(Tiedot avautuu uudessa selainikunassa.)",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/fi\_FI.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #799: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/fr_FR.json:189
<b>Line(s):</b>	189
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects weak or export-grade cipher suites

### Vulnerable Code

```
"TITLE_REQUEST_DATA_EXPORT": "Demande d'exportation de données",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption  
**Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/fr\_FR.json

### Further Reading:

<https://owasp.org/Top10/A02/>

## #800: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/fr_FR.json:203
<b>Line(s):</b>	203
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"EXPORT_LABEL": "Format d'export",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/fr\_FR.json **References & Further Reading:**

<https://owasp.org/Top10/A02/>

## #801: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/fr\_FR.json:287

**Line(s):** 287

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "(Votre export de données s'ouvrira dans une nouvelle fenêtre de navigateur.)",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/fr\_FR.json **References &**

#### Further Reading:

<https://owasp.org/Top10/A02/>

## #802: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327 **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/ga\_IE.json:189

**Line(s):** 189

**Exploitability:** Unknown **Impact:** Unknown

#### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": "Request Data Export",
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/ga\_IE.json **References &**

**Further Reading:**

<https://owasp.org/Top10/A02/>

## #803: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/ga\_IE.json:203

**Line(s):** 203

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"EXPORT_LABEL": "Export Format",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/ga\_IE.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #804: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/ga_IE.json:287
<b>Line(s):</b>	287
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "(Your data export will open in a new Browser window.)",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/ga\_IE.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #805: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/he_IL.json:189
<b>Line(s):</b>	189
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects weak or export-grade cipher suites

## Vulnerable Code

```
"TITLE_REQUEST_DATA_EXPORT": " ",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption

## Remediation Guidance

Review and fix the cryptography issue in frontend/src/assets/i18n/he\_IL.json

## References &

## Further Reading:

<https://owasp.org/Top10/A02/>

## #806: TLS - Weak Cipher Suite Configuration

**Severity:** HIGH (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/he\_IL.json:203

**Line(s):** 203

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects weak or export-grade cipher suites

## Vulnerable Code

```
"EXPORT_LABEL": " ",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/he\_IL.json **References &**

#### Further Reading:

<https://owasp.org/Top10/A02/>

## #807: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327 **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/he\_IL.json:287

**Line(s):** 287

**Exploitability:** Unknown **Impact:** Unknown

#### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "(...)",
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/he\_IL.json **References &**

**Further Reading:**

<https://owasp.org/Top10/A02/>

## #808: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: **7.5/10.0**)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** **MEDIUM**

**Location:** frontend/src/assets/i18n/hi\_IN.json:189

**Line(s):** 189

**Exploitability:** **Unknown**      **Impact:** Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": "Request Data Export",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/hi\_IN.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #809: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/hi_IN.json:203
<b>Line(s):</b>	203
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"EXPORT_LABEL": "Export Format",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/hi\_IN.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #810: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/hi_IN.json:287
<b>Line(s):</b>	287
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "(Your data export will open in a new Browser window.)",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/hi\_IN.json **References & References**

## Further Reading:

<https://owasp.org/Top10/A02/>

## #811: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/hu\_HU.json:189

**Line(s):** 189

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": "Request Data Export",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/hu\_HU.json **References &**

#### Further Reading:

<https://owasp.org/Top10/A02/>

## #812: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327 **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/hu\_HU.json:203

**Line(s):** 203

**Exploitability:** Unknown **Impact:** Unknown

#### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"EXPORT_LABEL": "Export form tum",
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/hu\_HU.json **References &**

**Further Reading:**

<https://owasp.org/Top10/A02/>

## #813: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: **7.5/10.0**)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** **MEDIUM**

**Location:** frontend/src/assets/i18n/hu\_HU.json:287

**Line(s):** 287

**Exploitability:** **Unknown**      **Impact:** Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "(Your data export will open in a new Browser window.)",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/hu\_HU.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #814: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327
<b>Category:</b>	cryptography
<b>Location:</b>	frontend/src/assets/i18n/id_ID.json:189
<b>Line(s):</b>	189
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": "Request Data Export",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/id\_ID.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #815: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327
<b>Category:</b>	cryptography
<b>Location:</b>	frontend/src/assets/i18n/id_ID.json:203
<b>Line(s):</b>	203
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects weak or export-grade cipher suites

## Vulnerable Code

```
"EXPORT_LABEL": "Export Format",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption

## Remediation Guidance

Review and fix the cryptography issue in frontend/src/assets/i18n/id\_ID.json

## References &

## Further Reading:

<https://owasp.org/Top10/A02/>

## #816: TLS - Weak Cipher Suite Configuration

**Severity:** HIGH (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/id\_ID.json:287

**Line(s):** 287

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects weak or export-grade cipher suites

## Vulnerable Code

```
"DATA_EXPORT_HINT": "(Your data export will open in a new Browser window.)",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/id\_ID.json **References &**

#### Further Reading:

<https://owasp.org/Top10/A02/>

## #817: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327 **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/it\_IT.json:189

**Line(s):** 189

**Exploitability:** Unknown **Impact:** Unknown

#### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": "Richiedi esportazione dati",
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/it\_IT.json **References &**

**Further Reading:**

<https://owasp.org/Top10/A02/>

## #818: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/it\_IT.json:203

**Line(s):** 203

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"EXPORT_LABEL": "Formato di esportazione",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/it\_IT.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #819: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/it_IT.json:287
<b>Line(s):</b>	287
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "(Your data export will open in a new Browser window.)",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/it\_IT.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #820: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/ja_JP.json:189
<b>Line(s):</b>	189
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": " ",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/ja\_JP.json **References & Further Reading:**

## Further Reading:

<https://owasp.org/Top10/A02/>

## #821: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327 **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/ja\_JP.json:203

**Line(s):** 203

**Exploitability:** Unknown **Impact:** Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"EXPORT_LABEL": " ",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/ja\_JP.json **References &**

#### Further Reading:

<https://owasp.org/Top10/A02/>

## #822: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: **7.5/10.0**)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** **MEDIUM**

**Location:** frontend/src/assets/i18n/ja\_JP.json:287

**Line(s):** 287

**Exploitability:** **Unknown**      **Impact:** Unknown

#### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "()",
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/ja\_JP.json **References &**

**Further Reading:**

<https://owasp.org/Top10/A02/>

## #823: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/ka\_GE.json:189

**Line(s):** 189

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": "Request Data Export",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/ka\_GE.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #824: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327
<b>Category:</b>	cryptography
<b>Location:</b>	frontend/src/assets/i18n/ka_GE.json:203
<b>Line(s):</b>	203
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects weak or export-grade cipher suites

### Vulnerable Code

```
"EXPORT_LABEL": "Export Format",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption  
**Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/ka\_GE.json

### Further Reading:

<https://owasp.org/Top10/A02/>

## #825: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327
<b>Category:</b>	cryptography
<b>Location:</b>	frontend/src/assets/i18n/ka_GE.json:287
<b>Line(s):</b>	287
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "(Your data export will open in a new Browser window.)",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/ka\_GE.json **References & Further Reading:**

## Further Reading:

<https://owasp.org/Top10/A02/>

## #826: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/ko\_KR.json:189

**Line(s):** 189

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": "",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/ko\_KR.json **References &**

#### Further Reading:

<https://owasp.org/Top10/A02/>

## #827: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327 **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/ko\_KR.json:203

**Line(s):** 203

**Exploitability:** Unknown **Impact:** Unknown

#### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"EXPORT_LABEL": " ",
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/ko\_KR.json **References &**

**Further Reading:**

<https://owasp.org/Top10/A02/>

## #828: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/ko\_KR.json:287

**Line(s):** 287

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "((.)",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/ko\_KR.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #829: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/lv_LV.json:189
<b>Line(s):</b>	189
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": "Request Data Export",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/lv\_LV.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #830: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/lv_LV.json:203
<b>Line(s):</b>	203
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

## Description

Detects weak or export-grade cipher suites

## Vulnerable Code

```
"EXPORT_LABEL": "Eksporta Form ts",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption

## Remediation Guidance

Review and fix the cryptography issue in frontend/src/assets/i18n/lv\_LV.json

## References &

## Further Reading:

<https://owasp.org/Top10/A02/>

## #831: TLS - Weak Cipher Suite Configuration

**Severity:** HIGH (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/lv\_LV.json:287

**Line(s):** 287

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects weak or export-grade cipher suites

## Vulnerable Code

```
"DATA_EXPORT_HINT": "(Your data export will open in a new Browser window.)",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/lv\_LV.json **References &**

#### Further Reading:

<https://owasp.org/Top10/A02/>

## #832: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327 **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/my\_MM.json:189

**Line(s):** 189

**Exploitability:** Unknown **Impact:** Unknown

#### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": "Request Data Export",
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/my\_MM.json **References &**

**Further Reading:**

<https://owasp.org/Top10/A02/>

## #833: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/my\_MM.json:203

**Line(s):** 203

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"EXPORT_LABEL": "Export Format",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/my\_MM.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #834: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/my_MM.json:287
<b>Line(s):</b>	287
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "(Your data export will open in a new Browser window.)",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/my\_MM.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #835: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/nl_NL.json:189
<b>Line(s):</b>	189
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": "Start data export",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/nl\_NL.json **References & Further Reading:**

## Further Reading:

<https://owasp.org/Top10/A02/>

## #836: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327 **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/nl\_NL.json:203

**Line(s):** 203

**Exploitability:** Unknown **Impact:** Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"EXPORT_LABEL": "Exportformaat",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/nl\_NL.json **References &**

#### Further Reading:

<https://owasp.org/Top10/A02/>

## #837: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327 **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/nl\_NL.json:287

**Line(s):** 287

**Exploitability:** Unknown **Impact:** Unknown

#### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "(Uw data export wordt geopend in een nieuw browservenster.)",
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/nl\_NL.json **References &**

**Further Reading:**

<https://owasp.org/Top10/A02/>

## #838: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: **7.5/10.0**)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** **MEDIUM**

**Location:** frontend/src/assets/i18n/no\_NO.json:189

**Line(s):** 189

**Exploitability:** **Unknown**      **Impact:** Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": "Bemerk eksport av data",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/no\_NO.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #839: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327
<b>Category:</b>	cryptography
<b>Location:</b>	frontend/src/assets/i18n/no_NO.json:203
<b>Line(s):</b>	203
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"EXPORT_LABEL": "Eksportformat",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/no\_NO.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #840: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327
<b>Category:</b>	cryptography
<b>Location:</b>	frontend/src/assets/i18n/no_NO.json:287
<b>Line(s):</b>	287
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "(Dataeksporten din vil pnes i et nytt nettleservindu.)",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/no\_NO.json **References & Further Reading:**

<https://owasp.org/Top10/A02/>

## #841: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327 **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/pl\_PL.json:189

**Line(s):** 189

**Exploitability:** Unknown **Impact:** Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": "Eksport Danych",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/pl\_PL.json **References &**

#### Further Reading:

<https://owasp.org/Top10/A02/>

## #842: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327 **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/pl\_PL.json:203

**Line(s):** 203

**Exploitability:** Unknown **Impact:** Unknown

#### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"EXPORT_LABEL": "Format exportu",
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/pl\_PL.json **References &**

**Further Reading:**

<https://owasp.org/Top10/A02/>

## #843: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/pl\_PL.json:287

**Line(s):** 287

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "(Twój eksport danych otworzy się w nowym oknie przeglądarki.)",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/pl\_PL.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #844: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/pt_BR.json:189
<b>Line(s):</b>	189
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": "Solicitar Exporta o de Dados",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/pt\_BR.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #845: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/pt_BR.json:203
<b>Line(s):</b>	203
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"EXPORT_LABEL": "Formato de exportação",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/pt\_BR.json **References &**

## Further Reading:

<https://owasp.org/Top10/A02/>

## #846: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327 **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/pt\_BR.json:287

**Line(s):** 287

**Exploitability:** Unknown **Impact:** Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "(Seus dados exportados serão abertos em uma nova janela do navegador.)",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/pt\_BR.json **References &**

#### Further Reading:

<https://owasp.org/Top10/A02/>

## #847: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327 **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/pt\_PT.json:189

**Line(s):** 189

**Exploitability:** Unknown **Impact:** Unknown

#### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": "Pedido de Obtenção de Dados",
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/pt\_PT.json **References &**

**Further Reading:**

<https://owasp.org/Top10/A02/>

## #848: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/pt\_PT.json:203

**Line(s):** 203

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"EXPORT_LABEL": "Formato de Exportação",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/pt\_PT.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #849: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327
<b>Category:</b>	cryptography
<b>Location:</b>	frontend/src/assets/i18n/pt_PT.json:287
<b>Line(s):</b>	287
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "(O teu pedido de visualiza o dos teus dados abrir numa nova janela do Browser que est s a usar.)",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/pt\_PT.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #850: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/ro_RO.json:189
<b>Line(s):</b>	189
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": "Solicit export de date",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/ro\_RO.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #851: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/ro_RO.json:203
<b>Line(s):</b>	203
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"EXPORT_LABEL": "Export formatul",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/ro\_RO.json **References & Further Reading:**

<https://owasp.org/Top10/A02/>

## #852: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: **7.5/10.0**)

**CWE ID:** CWE-327 **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography **Confidence:** **MEDIUM**

**Location:** frontend/src/assets/i18n/ro\_RO.json:287

**Line(s):** 287

**Exploitability:** [Unknown](#) **Impact:** Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "(Exportarea datelor se va deschide ntr-o fereastr nou de browser.)",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/ro\_RO.json **References &**

#### Further Reading:

<https://owasp.org/Top10/A02/>

## #853: TLS - Weak Cipher Suite Configuration

**Severity:** HIGH (CVSS: 7.5/10.0)

**CWE ID:** CWE-327 **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/ru\_RU.json:189

**Line(s):** 189

**Exploitability:** Unknown **Impact:** Unknown

#### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": " ",
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/ru\_RU.json **References &**

**Further Reading:**

<https://owasp.org/Top10/A02/>

## #854: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/ru\_RU.json:203

**Line(s):** 203

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"EXPORT_LABEL": " ",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/ru\_RU.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #855: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/ru_RU.json:287
<b>Line(s):</b>	287
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "(...)",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/ru\_RU.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #856: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/si_LK.json:189
<b>Line(s):</b>	189
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": "Request Data Export",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/si\_LK.json **References &**

## Further Reading:

<https://owasp.org/Top10/A02/>

## #857: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327 **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/si\_LK.json:203

**Line(s):** 203

**Exploitability:** Unknown **Impact:** Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"EXPORT_LABEL": "Export Format",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/si\_LK.json **References &**

#### Further Reading:

<https://owasp.org/Top10/A02/>

## #858: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: **7.5/10.0**)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** **MEDIUM**

**Location:** frontend/src/assets/i18n/si\_LK.json:287

**Line(s):** 287

**Exploitability:** **Unknown**      **Impact:** Unknown

#### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "(Your data export will open in a new Browser window.)",
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/si\_LK.json **References &**

**Further Reading:**

<https://owasp.org/Top10/A02/>

## #859: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/sv\_SE.json:189

**Line(s):** 189

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": "Beg r dataexport",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/sv\_SE.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #860: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/sv_SE.json:203
<b>Line(s):</b>	203
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"EXPORT_LABEL": "Format f r export",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/sv\_SE.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #861: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/sv_SE.json:287
<b>Line(s):</b>	287
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "(Your data export will open in a new Browser window.)",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/sv\_SE.json **References & Further Reading:**

## Further Reading:

<https://owasp.org/Top10/A02/>

## #862: TLS - Weak Cipher Suite Configuration

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-327
Category:	cryptography
Location:	frontend/src/assets/i18n/th_TH.json:189
Line(s):	189
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": " ",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/th\_TH.json **References &**

#### Further Reading:

<https://owasp.org/Top10/A02/>

## #863: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327 **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/th\_TH.json:203

**Line(s):** 203

**Exploitability:** Unknown **Impact:** Unknown

#### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"EXPORT_LABEL": " ",
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/th\_TH.json **References &**

**Further Reading:**

<https://owasp.org/Top10/A02/>

## #864: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: **7.5/10.0**)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** **MEDIUM**

**Location:** frontend/src/assets/i18n/th\_TH.json:287

**Line(s):** 287

**Exploitability:** **Unknown**      **Impact:** Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "(Your data export will open in a new Browser window.)",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/th\_TH.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #865: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/tlh_AA.json:189
<b>Line(s):</b>	189
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": "Request Data Export",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/tlh\_AA.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #866: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/tlh_AA.json:203
<b>Line(s):</b>	203
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects weak or export-grade cipher suites

## Vulnerable Code

```
"EXPORT_LABEL": "Export Format",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption

## Remediation Guidance

Review and fix the cryptography issue in frontend/src/assets/i18n/tlh\_AA.json

## References &

## Further Reading:

<https://owasp.org/Top10/A02/>

## #867: TLS - Weak Cipher Suite Configuration

**Severity:** HIGH (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/tlh\_AA.json:287

**Line(s):** 287

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects weak or export-grade cipher suites

## Vulnerable Code

```
"DATA_EXPORT_HINT": "(Your data export will open in a new Browser window.)",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/tlh\_AA.json **References &**

#### Further Reading:

<https://owasp.org/Top10/A02/>

## #868: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: **7.5/10.0**)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** **MEDIUM**

**Location:** frontend/src/assets/i18n/tr\_TR.json:189

**Line(s):** 189

**Exploitability:** **Unknown**      **Impact:** Unknown

#### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": "Veri D a Aktar m ste",
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/tr\_TR.json **References &**

**Further Reading:**

<https://owasp.org/Top10/A02/>

## #869: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/tr\_TR.json:203

**Line(s):** 203

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"EXPORT_LABEL": "D a Aktarma Bi imi",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/tr\_TR.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #870: TLS - Weak Cipher Suite Configuration

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-327
Category:	cryptography
Location:	frontend/src/assets/i18n/tr_TR.json:287
Line(s):	287
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

### Description

Detects weak or export-grade cipher suites

### Vulnerable Code

```
"DATA_EXPORT_HINT": "(Veri d a aktarma i leminiz yeni bir Taray c penceresinde a lacakt r.)",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption  
**Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/tr\_TR.json

### Further Reading:

<https://owasp.org/Top10/A02/>

## #871: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/uk_UA.json:189
<b>Line(s):</b>	189
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": "Request Data Export",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/uk\_UA.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #872: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/uk_UA.json:203
<b>Line(s):</b>	203
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"EXPORT_LABEL": "Export Format",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/uk\_UA.json **References &**

## Further Reading:

<https://owasp.org/Top10/A02/>

## #873: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/uk\_UA.json:287

**Line(s):** 287

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "(Your data export will open in a new Browser window.)",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/uk\_UA.json **References &**

#### Further Reading:

<https://owasp.org/Top10/A02/>

## #874: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/zh\_CN.json:189

**Line(s):** 189

**Exploitability:** Unknown      **Impact:** Unknown

#### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": " ",
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/zh\_CN.json **References &**

**Further Reading:**

<https://owasp.org/Top10/A02/>

## #875: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/zh\_CN.json:203

**Line(s):** 203

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"EXPORT_LABEL": " ",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/zh\_CN.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #876: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327
<b>Category:</b>	cryptography
<b>Location:</b>	frontend/src/assets/i18n/zh_CN.json:287
<b>Line(s):</b>	287
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects weak or export-grade cipher suites

### Vulnerable Code

```
"DATA_EXPORT_HINT": "()",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption  
**Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/zh\_CN.json

### References &

### Further Reading:

<https://owasp.org/Top10/A02/>

## #877: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327
<b>Category:</b>	cryptography
<b>Location:</b>	frontend/src/assets/i18n/zh_HK.json:189
<b>Line(s):</b>	189
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": "Request Data Export",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/zh\_HK.json **References &**

## Further Reading:

<https://owasp.org/Top10/A02/>

## #878: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/zh\_HK.json:203

**Line(s):** 203

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"EXPORT_LABEL": "Export Format",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/zh\_HK.json **References &**

#### Further Reading:

<https://owasp.org/Top10/A02/>

## #879: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: **7.5/10.0**)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** **MEDIUM**

**Location:** frontend/src/assets/i18n/zh\_HK.json:287

**Line(s):** 287

**Exploitability:** **Unknown**      **Impact:** Unknown

#### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "(Your data export will open in a new Browser window.)",
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/zh\_HK.json **References &**

**Further Reading:**

<https://owasp.org/Top10/A02/>

## #880: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** frontend/src/assets/i18n/zh\_TW.json:189

**Line(s):** 189

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": " ",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/zh\_TW.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #881: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/zh_TW.json:203
<b>Line(s):</b>	203
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects weak or export-grade cipher suites

#### Vulnerable Code

```
"EXPORT_LABEL": " ",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption  
**Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/zh\_TW.json

### References & Further Reading:

<https://owasp.org/Top10/A02/>

## #882: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/zh_TW.json:287
<b>Line(s):</b>	287
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"DATA_EXPORT_HINT": "()",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/zh\_TW.json **References & Further Reading:**

## Further Reading:

<https://owasp.org/Top10/A02/>

## #883: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** MEDIUM

**Location:** frontend/src/app/data-export/data-export.component.html:9

**Line(s):** 9

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
<h1 translate>TITLE_REQUEST_DATA_EXPORT</h1>
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/app/data-export/data-export.component.html

#### References & Further Reading:

<https://owasp.org/Top10/A02/>

## #884: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-327 **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography **Confidence:** MEDIUM

**Location:** frontend/src/app/data-export/data-export.component.html:17

**Line(s):** 17

**Exploitability:** Unknown **Impact:** Unknown

### Description

Detects weak or export-grade cipher suites

#### Vulnerable Code

```
<mat-label class="radio-label" translate>EXPORT_LABEL :</mat-label>
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/app/data-export/data-

export.component.html**References & Further Reading:**

<https://owasp.org/Top10/A02/>

## #885: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-327
<b>Category:</b>	cryptography
<b>Location:</b>	frontend/src/app/data-export/data-export.component.html:48
<b>Line(s):</b>	48
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

<span translate>DATA\_EXPORT\_HINT</span>

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/app/data-export/data-export.component.html **References & Further Reading:**

<https://owasp.org/Top10/A02/>

## #886: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/navbar/navbar.component.html:149
<b>Line(s):</b>	149
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
TITLE_REQUEST_DATA_EXPORT
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/app/navbar/navbar.component.html **References**

### & Further Reading:

<https://owasp.org/Top10/A02/>

## #887: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/app/sidenav/sidenav.component.html:133
<b>Line(s):</b>	133
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
TITLE_REQUEST_DATA_EXPORT
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/app/sidenav/sidenav.component.html **References**

## & Further Reading:

<https://owasp.org/Top10/A02/>

## #888: API - No Rate Limit on Email Sending

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-770      **OWASP:** A04:2021 - Insecure Design

**Category:** api\_security      **Confidence:** **MEDIUM**

**Location:** data/static/codeworks/loginBenderChallenge\_3.ts:34

**Line(s):** 34

**Exploitability:** **Unknown**      **Impact:** Unknown

## Description

Detects email endpoints without rate limiting **Vulnerable Code**

```
res.status(401).send(res._('Invalid email or password.'))
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the api\_security issue in data/static/codefixes/loginBenderChallenge\_3.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A04/>

## #889: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	Confidence: MEDIUM
<b>Exploitability:</b>	frontend/src/app/two-factor-auth-enter/two-factor-auth-enter.comp...
82	
<b>Impact:</b>	Unknown

#### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClient(withInterceptorsFromDi()),
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in frontend/src/app/two-factor-auth-enter/two-factor-

auth-enter.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #890: Angular - HttpClient Without Interceptor Auth

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-306
<b>Category:</b>	OWASP: A07:2021 - Identification and Authentication Failures
<b>Location:</b>	angular_security
<b>Line(s):</b>	fronted/src/app/two-factor-auth-enter/two-factor-auth-enter.comp... 83
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects HttpClient calls without auth interceptor **Vulnerable Code**

```
provideHttpClientTesting()
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the angular\_security issue in fronted/src/app/two-factor-auth-enter/two-factor-auth-enter.component.spec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #891: Vertical Access Control - Test Account in Production

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-489 <b>OWASP:</b> A05:2021 - Security Misconfiguration
<b>Category:</b>	access_control <b>Confidence:</b> MEDIUM
<b>Location:</b>	frontend/src/hacking-instructor/challenges/passwordStrength.ts:43
<b>Line(s):</b>	43
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

### Description

Detects test or demo accounts in production code

### Vulnerable Code

```
resolved: waitForInputToHaveValue('#email', 'admin@juice-sh.op') // TODO Use domain from config instead
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption

### Remediation Guidance

Review and fix the access\_control issue in frontend/src/hacking-instructor/challenges/passwordStrength.ts

### References & Further Reading:

<https://owasp.org/Top10/A05/>

## #892: CSRF - Missing Origin Validation

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-352 <b>OWASP:</b> A01:2021 - Broken Access Control
<b>Category:</b>	csrf <b>Confidence:</b> MEDIUM
<b>Location:</b>	server.ts:390
<b>Line(s):</b>	390
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

## Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post('/api/Feedbacks', verify.forgedFeedbackChallenge())
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

## #893: Weak Authentication - MD5 Password Hashing

**Severity:** **HIGH** (CVSS: **7.5/10.0**)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** authentication      **Confidence:** **MEDIUM**

**Location:** lib/insecurity.ts:43

**Line(s):** 43

**Exploitability:** [Unknown](#)      **Impact:** Unknown

## Description

Detects MD5 usage for password hashing **Vulnerable Code**

```
export const hash = (data: string) => crypto.createHash('md5').update(data).digest('hex')
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the authentication issue in lib/insecurity.ts **References & Further Reading:**

<https://owasp.org/Top10/A02/>

## #894: Session Management - Cookie Without Secure Flag

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-614 **OWASP:** A05:2021 - Security Misconfiguration

**Category:** authentication **Confidence:** MEDIUM

**Location:** lib/insecurity.ts:195

**Line(s):** 195

**Exploitability:** Unknown **Impact:** Unknown

### Description

Detects cookies without secure flag **Vulnerable Code**

```
res.cookie('token', token)
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the authentication issue in lib/insecurity.ts **References & Further Reading:**

<https://owasp.org/Top10/A05/>

## #895: Crypto - Insecure Random Number Generator

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-338 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	lib/insecurity.ts:55
<b>Line(s):</b>	55
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects usage of insecure random number generators **Vulnerable Code**

```
export const denyAll = () => expressJwt({ secret: " + Math.random() } as any)
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in lib/insecurity.ts **References & Further Reading:**

<https://owasp.org/Top10/A02/>

## #896: CSRF - Missing Origin Validation

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-352 <b>OWASP:</b> A01:2021 - Broken Access Control
<b>Category:</b>	csrf <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	server.ts:370
<b>Line(s):</b>	370
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post('/api/Complaints', security.isAuthorized())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

## #897: CSRF - Missing Origin Validation

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-352 <b>OWASP:</b> A01:2021 - Broken Access Control
<b>Category:</b>	csrf <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	server.ts:408
<b>Line(s):</b>	408
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post('/api/Users', verify.registerAdminChallenge())
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

## #898: API - No Rate Limit on Email Sending

**Severity:** HIGH (CVSS: 7.5/10.0)

**CWE ID:** CWE-770 **OWASP:** A04:2021 - Insecure Design

**Category:** api\_security **Confidence:** MEDIUM

**Location:** data/static/codeworks/loginBenderChallenge\_4.ts:33

**Line(s):** 33

**Exploitability:** Unknown **Impact:** Unknown

## Description

Detects email endpoints without rate limiting **Vulnerable Code**

```
res.status(401).send(res.__('Invalid email or password.'))
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the api\_security issue in data/static/codefixes/loginBenderChallenge\_4.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A04/>

## #899: CSRF - Missing Origin Validation

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-352      **OWASP:** A01:2021 - Broken Access Control

**Category:** csrf      **Confidence:** MEDIUM

**Location:** server.ts:446

**Line(s):** 446

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post('/rest/2fa/verify',
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

## #900: Vertical Access Control - Test Account in Production

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-489 <b>OWASP:</b> A05:2021 - Security Misconfiguration
<b>Category:</b>	access_control <b>Confidence:</b> MEDIUM
<b>Location:</b>	routes/login.ts:60
<b>Line(s):</b>	60
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects test or demo accounts in production code **Vulnerable Code**

```
challengeUtils.solveIf(challenges.weakPasswordChallenge, () => { return req.body.email ===
 'admin@' + config.get<string>('application.domain') && req.body.password === 'admin123' })
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in routes/login.ts **References & Further Reading:**

<https://owasp.org/Top10/A05/>

## #901: Data Exposure - PII in Logs

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-532 <b>OWASP:</b> A09:2021 - Security Logging and Monitoring Failures
<b>Category:</b>	data_exposure <b>Confidence:</b> MEDIUM
<b>Location:</b>	routes/login.ts:61
<b>Line(s):</b>	61
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects Personally Identifiable Information in log statements **Vulnerable Code**

```
challengeUtils.solveIf(challenges.loginSupportChallenge, () => { return req.body.email
 === 'support@' + config.get<string>('application.domain') && req.body.password ===
 'J6aVjTgOpRs@51!Zkq2AYnCE@RF$P' })
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the data\_exposure issue in routes/login.ts **References & Further Reading:**

<https://owasp.org/Top10/A09/>

## #902: Data Exposure - PII in Logs

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-532 <b>OWASP:</b> A09:2021 - Security Logging and Monitoring Failures
<b>Category:</b>	data_exposure <b>Confidence:</b> MEDIUM
<b>Location:</b>	routes/login.ts:62
<b>Line(s):</b>	62
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects Personally Identifiable Information in log statements **Vulnerable Code**

```
challengeUtils.solveIf(challenges.loginRapperChallenge, () => { return req.body.email === 'mc.safesearch@' + config.get<string>('application.domain') && req.body.password === 'Mr. N00dles' })
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the data\_exposure issue in routes/login.ts **References & Further Reading:**

<https://owasp.org/Top10/A09/>

## #903: Data Exposure - PII in Logs

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-532 <b>OWASP:</b> A09:2021 - Security Logging and Monitoring Failures
<b>Category:</b>	data_exposure <b>Confidence:</b> MEDIUM
<b>Location:</b>	routes/login.ts:63
<b>Line(s):</b>	63
<b>Exploitability:</b>	<b>Unknown</b> <b>Impact:</b> Unknown

## Description

Detects Personally Identifiable Information in log statements **Vulnerable Code**

```
challengeUtils.solveIf(challenges.loginAmyChallenge, () => { return req.body.email === 'amy@' + config.get<string>('application.domain') && req.body.password === 'K1f.....' })
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the data\_exposure issue in routes/login.ts **References & Further Reading:**

<https://owasp.org/Top10/A09/>

## #904: API - No Rate Limit on Email Sending

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-770
<b>Category:</b>	api_security
<b>Location:</b>	routes/login.ts:51
<b>Line(s):</b>	51
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

### Description

Detects email endpoints without rate limiting **Vulnerable Code**

```
res.status(401).send(res.___('Invalid email or password.'))
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the api\_security issue in routes/login.ts**References & Further Reading:**

<https://owasp.org/Top10/A04/>

## #905: API - No Rate Limit on Email Sending

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-770
<b>Category:</b>	api_security
<b>Location:</b>	OWASP: A04:2021 - Insecure Design
<b>Line(s):</b>	data/static/codefixes/loginJimChallenge_1_correct.ts:34
<b>Exploitability:</b>	34
<b>Impact:</b>	Unknown

### Description

Detects email endpoints without rate limiting**Vulnerable Code**

```
res.status(401).send(res.__('Invalid email or password.'))
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the api\_security issue in data/static/codefixes/loginJimChallenge\_1\_correct.ts**References**

**& Further Reading:**

<https://owasp.org/Top10/A04/>

## #906: Crypto - Insecure Random Number Generator

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-338
<b>Category:</b>	cryptography
<b>Location:</b>	data/datacreator.ts:231
<b>Line(s):</b>	231
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects usage of insecure random number generators **Vulnerable Code**

```
for (let i = 0; i < length; i++) { text += possible.charAt(Math.floor(Math.random() * possible.length)) }
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in data/datacreator.ts **References & Further Reading:**

<https://owasp.org/Top10/A02/>

## #907: Crypto - Insecure Random Number Generator

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-338
<b>Category:</b>	cryptography
<b>Location:</b>	data/datacreator.ts:249
<b>Line(s):</b>	249
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects usage of insecure random number generators **Vulnerable Code**

```
quantity: product.quantity ?? Math.floor(Math.random() * 70 + 30),
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in data/datacreator.ts **References & Further Reading:**

<https://owasp.org/Top10/A02/>

## #908: Crypto - Insecure Random Number Generator

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-338 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	data/datacreator.ts:307
<b>Line(s):</b>	307
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects usage of insecure random number generators **Vulnerable Code**

```
product.price = product.price ?? Math.floor(Math.random() * 9 + 1)
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in data/datacreator.ts **References & Further Reading:**

<https://owasp.org/Top10/A02/>

## #909: Crypto - Insecure Random Number Generator

**Severity:** **HIGH** (CVSS: **7.5/10.0**)

**CWE ID:** CWE-338      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** **MEDIUM**

**Location:** data/datacreator.ts:693

**Line(s):** 693

**Exploitability:** [Unknown](#)      **Impact:** Unknown

### Description

Detects usage of insecure random number generators **Vulnerable Code**

```
eta: Math.floor((Math.random() * 5) + 1).toString(),
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in data/datacreator.ts **References & Further Reading:**

<https://owasp.org/Top10/A02/>

## #910: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/ar_SA.json:189
<b>Line(s):</b>	189
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
"TITLE_REQUEST_DATA_EXPORT": " ",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/ar\_SA.json **References &**

### Further Reading:

<https://owasp.org/Top10/A02/>

## #911: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	frontend/src/assets/i18n/ar_SA.json:203
<b>Line(s):</b>	203
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects weak or export-grade cipher suites

#### Vulnerable Code

```
"EXPORT_LABEL": " ",
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption  
**Remediation Guidance**

Review and fix the cryptography issue in frontend/src/assets/i18n/ar\_SA.json

### Further Reading:

<https://owasp.org/Top10/A02/>

## #912: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-327 <b>OWASP:</b> A02:2021 - Cryptographic Failures
<b>Category:</b>	cryptography <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	test/cypress/e2e/login.spec.ts:148
<b>Line(s):</b>	148
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects weak or export-grade cipher suites

## Vulnerable Code

```
cy.get('#email').type(" or deletedAt IS NOT NULL--")
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption

## Remediation Guidance

Review and fix the cryptography issue in test/cypress/e2e/login.spec.ts

## References & Further

### Reading:

<https://owasp.org/Top10/A02/>

## #913: Vertical Access Control - Test Account in Production

**Severity:** HIGH (CVSS: 7.5/10.0)

**CWE ID:** CWE-489      **OWASP:** A05:2021 - Security Misconfiguration

**Category:** access\_control      **Confidence:** MEDIUM

**Location:** test/cypress/e2e/login.spec.ts:13

**Line(s):** 13

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects test or demo accounts in production code

## Vulnerable Code

```
it('should log in Admin with SQLI attack on email field using "admin@<juice-sh.op>\''--'', () => {
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in test/cypress/e2e/login.spec.ts **References & Further**

#### Reading:

<https://owasp.org/Top10/A05/>

## #914: Data Exposure - PII in Logs

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-532
<b>Category:</b>	data_exposure
<b>Location:</b>	test/cypress/e2e/login.spec.ts:13
<b>Line(s):</b>	13
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

#### Description

Detects Personally Identifiable Information in log statements **Vulnerable Code**

```
it('should log in Admin with SQLI attack on email field using "admin@<juice-sh.op>\'--"', () => {
```

#### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the data\_exposure issue in test/cypress/e2e/login.spec.ts **References & Further**

#### Reading:

<https://owasp.org/Top10/A09/>

## #915: Data Exposure - PII in Logs

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-532 <b>OWASP:</b> A09:2021 - Security Logging and Monitoring Failures
Category:	data_exposure <b>Confidence:</b> MEDIUM
Location:	test/cypress/e2e/login.spec.ts:26
Line(s):	26
Exploitability:	<b>Unknown</b> <b>Impact:</b> Unknown

### Description

Detects Personally Identifiable Information in log statements **Vulnerable Code**

```
it('should log in Jim with SQLI attack on email field using "jim@<juice-sh.op>\'--"', () =>
{
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the data\_exposure issue in test/cypress/e2e/login.spec.ts **References & Further**

#### Reading:

<https://owasp.org/Top10/A09/>

## #916: Data Exposure - PII in Logs

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-532
Category:	data_exposure
Location:	test/cypress/e2e/login.spec.ts:39
Line(s):	39
Exploitability:	<a href="#">Unknown</a>
Impact:	Unknown

### Description

Detects Personally Identifiable Information in log statements [Vulnerable Code](#)

```
it('should log in Bender with SQLI attack on email field using "bender@<juice-sh.op>\'--"',
() => {
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption [Remediation Guidance](#)

Review and fix the data\_exposure issue in test/cypress/e2e/login.spec.ts [References & Further Reading](#)

### Reading:

<https://owasp.org/Top10/A09/>

## #917: API - No Rate Limit on Email Sending

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-770
<b>Category:</b>	api_security
<b>Location:</b>	data/static/codefixes/loginJimChallenge_2.ts:33
<b>Line(s):</b>	33
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects email endpoints without rate limiting **Vulnerable Code**

```
res.status(401).send(res.__('Invalid email or password.'))
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the api\_security issue in data/static/codefixes/loginJimChallenge\_2.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A04/>

## #918: CSRF - Missing Origin Validation

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-352
<b>Category:</b>	csrf
<b>Location:</b>	server.ts:303
<b>Line(s):</b>	303
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post('/file-upload', uploadToMemory.single('file'), ensureFileIsPassed,
metrics.observeFileUploadMetricsMiddleware(), checkUploadSize, checkFileType,
handleZipFileUpload, handleXmlUpload, handleYamlUpload)
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

## #919: TLS - Weak Cipher Suite Configuration

**Severity:** **HIGH** (CVSS: **7.5/10.0**)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** cryptography      **Confidence:** **MEDIUM**

**Location:** test/server/insecuritySpec.ts:192

**Line(s):** 192

**Exploitability:** [Unknown](#)      **Impact:** Unknown

## Description

Detects weak or export-grade cipher suites **Vulnerable Code**

```
it('returns MD5 hash for any input string', () => {
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the cryptography issue in test/server/insecuritySpec.ts **References & Further**

#### Reading:

<https://owasp.org/Top10/A02/>

## #920: Vertical Access Control - Test Account in Production

**Severity:** **HIGH** (CVSS: **7.5/10.0**)

**CWE ID:** CWE-489      **OWASP:** A05:2021 - Security Misconfiguration

**Category:** access\_control      **Confidence:** **MEDIUM**

**Location:** test/server/insecuritySpec.ts:27

**Line(s):** 27

**Exploitability:** [Unknown](#)      **Impact:** Unknown

### Description

Detects test or demo accounts in production code **Vulnerable Code**

```
expect(security.userEmailFrom({ headers: { 'x-user-email': 'test@bla.blubb' } })) .to.equal('test@bla.blubb')
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in test/server/insecuritySpec.ts **References & Further Reading:**

#### Reading:

<https://owasp.org/Top10/A05/>

## #921: Weak Authentication - MD5 Password Hashing

**Severity:** HIGH (CVSS: 7.5/10.0)

**CWE ID:** CWE-327      **OWASP:** A02:2021 - Cryptographic Failures

**Category:** authentication      **Confidence:** MEDIUM

**Location:** Gruntfile.js:77

**Line(s):** 77

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects MD5 usage for password hashing **Vulnerable Code**

```
const md5 = crypto.createHash('md5')
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the authentication issue in Gruntfile.js **References & Further Reading:**

<https://owasp.org/Top10/A02/>

## #922: API - No Rate Limit on Email Sending

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-770
<b>Category:</b>	api_security
<b>Location:</b>	data/static/codefixes/loginJimChallenge_3.ts:34
<b>Line(s):</b>	34
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects email endpoints without rate limiting **Vulnerable Code**

```
res.status(401).send(res.__('Invalid email or password.'))
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the api\_security issue in data/static/codefixes/loginJimChallenge\_3.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A04/>

## #923: TLS - Weak Cipher Suite Configuration

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-327
<b>Category:</b>	cryptography
<b>Location:</b>	frontend/src/assets/i18n/ar_SA.json:287
<b>Line(s):</b>	287
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects weak or export-grade cipher suites

## Vulnerable Code

```
"DATA_EXPORT_HINT": "(...)",
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption

## Remediation Guidance

Review and fix the cryptography issue in frontend/src/assets/i18n/ar\_SA.json

## References &

## Further Reading:

<https://owasp.org/Top10/A02/>

## #924: Vertical Access Control - Missing Role Check

**Severity:** HIGH (CVSS: 7.5/10.0)

**CWE ID:** CWE-862      **OWASP:** A01:2021 - Broken Access Control

**Category:** access\_control      **Confidence:** MEDIUM

**Location:** server.ts:363

**Line(s):** 363

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects operations without role-based access control

## Vulnerable Code

```
// app.put('/api/Products/:id', security.isAuthorized()) // vuln-code-snippet vuln-line
changeProductChallenge
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

## #925: Vertical Access Control - Missing Role Check

**Severity:** HIGH (CVSS: 7.5/10.0)

**CWE ID:** CWE-862 **OWASP:** A01:2021 - Broken Access Control

**Category:** access\_control **Confidence:** MEDIUM

**Location:** server.ts:377

**Line(s):** 377

**Exploitability:** Unknown **Impact:** Unknown

### Description

Detects operations without role-based access control **Vulnerable Code**

```
app.put('/api/Recycles/:id', security.denyAll())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

## #926: Vertical Access Control - Missing Role Check

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)	
CWE ID:	CWE-862	OWASP: A01:2021 - Broken Access Control
Category:	access_control	Confidence: <b>MEDIUM</b>
Location:	server.ts:414	
Line(s):	414	
Exploitability:	<a href="#">Unknown</a>	Impact: Unknown

### Description

Detects operations without role-based access control

### Vulnerable Code

```
app.put('/api/BasketItems/:id', security.appendUserId(), basketItems.quantityCheckBeforeBasketItemUpdate())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption

**Remediation Guidance**

Review and fix the access\_control issue in server.ts

### References & Further Reading:

<https://owasp.org/Top10/A01/>

## #927: Vertical Access Control - Missing Role Check

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-862
<b>Category:</b>	access_control
<b>Location:</b>	server.ts:421
<b>Line(s):</b>	421
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects operations without role-based access control [Vulnerable Code](#)

```
app.put('/api/Feedbacks/:id', security.denyAll())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

## #928: Vertical Access Control - Missing Role Check

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-862
<b>Category:</b>	access_control
<b>Location:</b>	server.ts:428
<b>Line(s):</b>	428
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects operations without role-based access control **Vulnerable Code**

```
app.put('/api/Cards/:id', security.denyAll())
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

## #929: Vertical Access Control - Missing Role Check

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-862
<b>Category:</b>	access_control
<b>Location:</b>	server.ts:438
<b>Line(s):</b>	438
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects operations without role-based access control **Vulnerable Code**

```
app.put('/api/Addresss/:id', security.appendUserId())
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

## #930: Vertical Access Control - Missing Role Check

**Severity:** **HIGH** (CVSS: **7.5/10.0**)

**CWE ID:** CWE-862 **OWASP:** A01:2021 - Broken Access Control

**Category:** access\_control **Confidence:** **MEDIUM**

**Location:** server.ts:583

**Line(s):** 583

**Exploitability:** [Unknown](#) **Impact:** Unknown

### Description

Detects operations without role-based access control **Vulnerable Code**

```
app.put('/rest/basket/:id/coupon/:coupon', applyCoupon())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

## #931: Vertical Access Control - Missing Role Check

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)	
CWE ID:	CWE-862	OWASP: A01:2021 - Broken Access Control
Category:	access_control	Confidence: <b>MEDIUM</b>
Location:	server.ts:590	
Line(s):	590	
Exploitability:	<a href="#">Unknown</a>	Impact: Unknown

### Description

Detects operations without role-based access control **Vulnerable Code**

```
app.put('/rest/continue-code-findIt/apply/:continueCode', restoreProgress.restoreProgressFindIt())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

## #932: Vertical Access Control - Missing Role Check

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-862
<b>Category:</b>	access_control
<b>Location:</b>	server.ts:603
<b>Line(s):</b>	603
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects operations without role-based access control [Vulnerable Code](#)

```
app.put('/rest/order-history/:id/delivery-status', security.isAccounting(),
 toggleDeliveryStatus())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

## #933: Vertical Access Control - Missing Role Check

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-862
<b>Category:</b>	access_control
<b>Location:</b>	server.ts:613
<b>Line(s):</b>	613
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects operations without role-based access control **Vulnerable Code**

```
app.put('/rest/products/:id/reviews', createProductReviews())
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

## #934: API - No Rate Limit on Email Sending

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-770
<b>Category:</b>	api_security
<b>Location:</b>	server.ts:403
<b>Line(s):</b>	403
<b>Exploitability:</b>	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

## Description

Detects email endpoints without rate limiting **Vulnerable Code**

```
res.status(400).send(res.__('Invalid email/password cannot be empty'))
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the api\_security issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A04/>

## #935: CSRF - Missing Origin Validation

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-352 **OWASP:** A01:2021 - Broken Access Control

**Category:** csrf **Confidence:** MEDIUM

**Location:** server.ts:362

**Line(s):** 362

**Exploitability:** Unknown **Impact:** Unknown

### Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post('/api/Products', security.isAuthorized() // vuln-code-snippet neutral-line
changeProductChallenge
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

## #936: CSRF - Missing Origin Validation

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)	
CWE ID:	CWE-352	OWASP: A01:2021 - Broken Access Control
Category:	csrf	Confidence: <b>MEDIUM</b>
Location:	server.ts:607	
Line(s):	607	
Exploitability:	<a href="#">Unknown</a>	Impact: Unknown

### Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post('/rest/deluxe-membership', security.appendUserId(), upgradeToDeluxe())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

## #937: CSRF - Missing Origin Validation

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-352 <b>OWASP:</b> A01:2021 - Broken Access Control
<b>Category:</b>	csrf <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	server.ts:618
<b>Line(s):</b>	618
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post('/rest/web3/submitKey', checkKeys())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

## #938: CSRF - Missing Origin Validation

<b>Severity:</b>	<b>HIGH</b> (CVSS: <b>7.5/10.0</b> )
<b>CWE ID:</b>	CWE-352 <b>OWASP:</b> A01:2021 - Broken Access Control
<b>Category:</b>	csrf <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	server.ts:644
<b>Line(s):</b>	644
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post('/profile', updateUserProfile())
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

## #939: CSRF - File Upload Without CSRF

**Severity:** HIGH (CVSS: 7.5/10.0)

**CWE ID:** CWE-352 **OWASP:** A01:2021 - Broken Access Control

**Category:** csrf **Confidence:** MEDIUM

**Location:** server.ts:661

**Line(s):** 661

**Exploitability:** Unknown **Impact:** Unknown

## Description

Detects file upload without CSRF protection **Vulnerable Code**

```
const uploadToMemory = multer({ storage: multer.memoryStorage(), limits: { fileSize: 200000 } })
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A01/>

## #940: Vertical Access Control - Missing Role Check

**Severity:** **HIGH** (CVSS: **7.5/10.0**)

**CWE ID:** CWE-862 **OWASP:** A01:2021 - Broken Access Control

**Category:** access\_control **Confidence:** **MEDIUM**

**Location:** data/static/codefixes/changeProductChallenge\_1.ts:17

**Line(s):** 17

**Exploitability:** [Unknown](#) **Impact:** Unknown

### Description

Detects operations without role-based access control **Vulnerable Code**

```
app.delete('/api/Products/:id', security.denyAll())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in data/static/codefixes/changeProductChallenge\_1.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A01/>

## #941: Vertical Access Control - Missing Role Check

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-862      **OWASP:** A01:2021 - Broken Access Control

**Category:** access\_control      **Confidence:** MEDIUM

**Location:** data/static/codefixes/changeProductChallenge\_1.ts:30

**Line(s):** 30

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects operations without role-based access control **Vulnerable Code**

```
app.put('/api/Recycles/:id', security.denyAll())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in data/static/codefixes/changeProductChallenge\_1.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A01/>

## #942: Vertical Access Control - Missing Role Check

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-862
<b>Category:</b>	access_control
<b>Location:</b>	data/static/codefixes/changeProductChallenge_1.ts:45
<b>Line(s):</b>	45
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects operations without role-based access control [Vulnerable Code](#)

```
app.put('/api/BasketItems/:id', security.appendUserId(), basketItems.quantityCheckBeforeBasketItemUpdate())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in data/static/codefixes/changeProductChallenge\_1.ts [References](#)

### & Further Reading:

<https://owasp.org/Top10/A01/>

## #943: Vertical Access Control - Missing Role Check

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-862
<b>Category:</b>	access_control
<b>Location:</b>	data/static/codefixes/changeProductChallenge_1.ts:52
<b>Line(s):</b>	52
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects operations without role-based access control **Vulnerable Code**

```
app.put('/api/Feedbacks/:id', security.denyAll())
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in data/static/codefixes/changeProductChallenge\_1.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A01/>

## #944: Vertical Access Control - Missing Role Check

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-862      **OWASP:** A01:2021 - Broken Access Control

**Category:** access\_control      **Confidence:** MEDIUM

**Location:** data/static/codefixes/changeProductChallenge\_1.ts:59

**Line(s):** 59

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects operations without role-based access control **Vulnerable Code**

```
app.put('/api/Cards/:id', security.denyAll())
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in data/static/codefixes/changeProductChallenge\_1.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A01/>

## #945: Vertical Access Control - Missing Role Check

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-862
<b>Category:</b>	access_control
<b>Location:</b>	OWASP: A01:2021 - Broken Access Control
<b>Line(s):</b>	data/static/codefixes/changeProductChallenge_1.ts:69
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects operations without role-based access control **Vulnerable Code**

```
app.put('/api/Addresss/:id', security.appendUserId())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in data/static/codefixes/changeProductChallenge\_1.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A01/>

## #946: CSRF - Missing Origin Validation

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-352      **OWASP:** A01:2021 - Broken Access Control

**Category:** csrf      **Confidence:** MEDIUM

**Location:** data/static/codefixes/changeProductChallenge\_1.ts:16

**Line(s):** 16

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post('/api/Products', security.isAuthorized())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in data/static/codefixes/changeProductChallenge\_1.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A01/>

## #947: CSRF - Missing Origin Validation

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-352
<b>Category:</b>	csrf
<b>Location:</b>	data/static/codefixes/changeProductChallenge_1.ts:27
<b>Line(s):</b>	27
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post('/api/Recycles', security.isAuthorized())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in data/static/codefixes/changeProductChallenge\_1.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A01/>

## #948: CSRF - Missing Origin Validation

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-352
<b>Category:</b>	csrf
<b>Location:</b>	data/static/codefixes/changeProductChallenge_1.ts:46
<b>Line(s):</b>	46
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post('/api/BasketItems', security.appendUserId(), basketItems.quantityCheckBeforeBasketItemAddition(), basketItems.addBasketItem())
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in data/static/codefixes/changeProductChallenge\_1.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A01/>

## #949: CSRF - Missing Origin Validation

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-352      **OWASP:** A01:2021 - Broken Access Control

**Category:** csrf      **Confidence:** MEDIUM

**Location:** data/static/codefixes/changeProductChallenge\_1.ts:57

**Line(s):** 57

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post('/api/Cards', security.appendUserId())
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in data/static/codefixes/changeProductChallenge\_1.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A01/>

## #950: CSRF - Missing Origin Validation

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-352      **OWASP:** A01:2021 - Broken Access Control

**Category:** csrf      **Confidence:** MEDIUM

**Location:** data/static/codefixes/changeProductChallenge\_1.ts:67

**Line(s):** 67

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post('/api/Addresss', security.appendUserId())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in data/static/codefixes/changeProductChallenge\_1.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A01/>

## #951: Vertical Access Control - Missing Role Check

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-862      **OWASP:** A01:2021 - Broken Access Control

**Category:** access\_control      **Confidence:** MEDIUM

**Location:** data/static/codefixes/changeProductChallenge\_2.ts:27

**Line(s):** 27

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects operations without role-based access control **Vulnerable Code**

```
app.put('/api/Recycles/:id', security.denyAll())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in data/static/codefixes/changeProductChallenge\_2.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A01/>

## #952: Vertical Access Control - Missing Role Check

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-862
<b>Category:</b>	access_control
<b>Location:</b>	data/static/codefixes/changeProductChallenge_2.ts:42
<b>Line(s):</b>	42
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects operations without role-based access control [Vulnerable Code](#)

```
app.put('/api/BasketItems/:id', security.appendUserId(), basketItems.quantityCheckBeforeBasketItemUpdate())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in data/static/codefixes/changeProductChallenge\_2.ts [References](#)

### & Further Reading:

<https://owasp.org/Top10/A01/>

## #953: Vertical Access Control - Missing Role Check

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-862
<b>Category:</b>	access_control
<b>Location:</b>	data/static/codefixes/changeProductChallenge_2.ts:49
<b>Line(s):</b>	49
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects operations without role-based access control **Vulnerable Code**

```
app.put('/api/Feedbacks/:id', security.denyAll())
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in data/static/codefixes/changeProductChallenge\_2.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A01/>

## #954: Vertical Access Control - Missing Role Check

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-862      **OWASP:** A01:2021 - Broken Access Control

**Category:** access\_control      **Confidence:** MEDIUM

**Location:** data/static/codefixes/changeProductChallenge\_2.ts:56

**Line(s):** 56

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects operations without role-based access control **Vulnerable Code**

```
app.put('/api/Cards/:id', security.denyAll())
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in data/static/codefixes/changeProductChallenge\_2.ts **References**

#### & Further Reading:

<https://owasp.org/Top10/A01/>

## #955: Vertical Access Control - Missing Role Check

Severity:	<b>HIGH</b> (CVSS: 7.5/10.0)
CWE ID:	CWE-862
Category:	access_control
Location:	OWASP: A01:2021 - Broken Access Control
Line(s):	data/static/codefixes/changeProductChallenge_2.ts:66
Exploitability:	<a href="#">Unknown</a>
	<b>Impact:</b> Unknown

### Description

Detects operations without role-based access control **Vulnerable Code**

```
app.put('/api/Addresss/:id', security.appendUserId())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the access\_control issue in data/static/codefixes/changeProductChallenge\_2.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A01/>

## #956: CSRF - Missing Origin Validation

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-352      **OWASP:** A01:2021 - Broken Access Control

**Category:** csrf      **Confidence:** MEDIUM

**Location:** data/static/codefixes/changeProductChallenge\_2.ts:16

**Line(s):** 16

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post('/api/Challenges', security.denyAll())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in data/static/codefixes/changeProductChallenge\_2.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A01/>

## #957: CSRF - Missing Origin Validation

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-352
<b>Category:</b>	csrf
<b>Location:</b>	data/static/codefixes/changeProductChallenge_2.ts:30
<b>Line(s):</b>	30
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post('/api/SecurityQuestions', security.denyAll())
```

### Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in data/static/codefixes/changeProductChallenge\_2.ts **References**

### & Further Reading:

<https://owasp.org/Top10/A01/>

## #958: CSRF - Missing Origin Validation

<b>Severity:</b>	<b>HIGH</b> (CVSS: 7.5/10.0)
<b>CWE ID:</b>	CWE-352
<b>Category:</b>	csrf
<b>Location:</b>	data/static/codefixes/changeProductChallenge_2.ts:43
<b>Line(s):</b>	43
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

## Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post('/api/BasketItems', security.appendUserId(), basketItems.quantityCheckBeforeBasketItemAddition(), basketItems.addBasketItem())
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in data/static/codefixes/changeProductChallenge\_2.ts **References**

## & Further Reading:

<https://owasp.org/Top10/A01/>

## #959: CSRF - Missing Origin Validation

**Severity:** **HIGH** (CVSS: 7.5/10.0)

**CWE ID:** CWE-352      **OWASP:** A01:2021 - Broken Access Control

**Category:** csrf      **Confidence:** MEDIUM

**Location:** data/static/codefixes/changeProductChallenge\_2.ts:54

**Line(s):** 54

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects missing Origin header validation **Vulnerable Code**

```
app.post('/api/Cards', security.appendUserId())
```

## Security Impact

This high-severity issue could enable an attacker to:

- Bypass authentication or authorization controls
- Access or modify sensitive data
- Inject malicious payloads
- Escalate privileges

**Business Consequences:** Data exposure, compliance violations, service disruption **Remediation Guidance**

Review and fix the csrf issue in data/static/codefixes/changeProductChallenge\_2.ts **References**

**& Further Reading:**

<https://owasp.org/Top10/A01/>

## 0.2.3 Medium Severity Issues (241 found)

### Address in Next Sprint

Medium severity issues represent **moderate security risks** that should be addressed but may not pose immediate threats. Include these in upcoming development cycles.

**Recommended Timeline:** Fix within **30-60 days**

### #1: 2FA - Missing Two-Factor Authentication

<b>Severity:</b>	<b>MEDIUM</b> (CVSS: 5.0/10.0)	
<b>CWE ID:</b>	CWE-308	<b>OWASP:</b> A07:2021 - Identification and Authentication Failures
<b>Category:</b>	authentication	<b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	test/api/2faSpec.ts:20	
<b>Line(s):</b>	20	
<b>Exploitability:</b>	<a href="#">Unknown</a>	<b>Impact:</b> Unknown

### Description

Detects login without 2FA implementation **Vulnerable Code**

```
async function login ({ email, password, totpSecret }: { email: string, password: string, totpSecret?: string }) {
```

### Security Impact

This moderate security weakness could be exploited to:

- Gather information about the system
- Perform limited unauthorized actions
- Facilitate further exploitation attempts

**Business Consequences:** Increased attack surface, potential stepping stone for advanced attacks  
**Remediation Guidance**

Review and fix the authentication issue in test/api/2faSpec.ts **References & Further Reading:**

<https://owasp.org/Top10/A07/>

## #2: Misconfig - Directory Listing Enabled

<b>Severity:</b>	<b>MEDIUM</b> (CVSS: 5.0/10.0)
<b>CWE ID:</b>	CWE-548 <b>OWASP:</b> A05:2021 - Security Misconfiguration
<b>Category:</b>	misconfiguration <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	data/static/codefixes/directoryListingChallenge_4.ts:14
<b>Line(s):</b>	14
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

### Description

Detects enabled directory listing

#### Vulnerable Code

```
app.use('/support/logs', serveIndexMiddleware, serveIndex('logs', { icons: true, view: 'details' }))
```

### Security Impact

This moderate security weakness could be exploited to:

- Gather information about the system
- Perform limited unauthorized actions
- Facilitate further exploitation attempts

**Business Consequences:** Increased attack surface, potential stepping stone for advanced attacks

#### Remediation Guidance

Review and fix the misconfiguration issue in data/static/codefixes/directoryListingChallenge\_4.ts

#### & Further Reading:

<https://owasp.org/Top10/A05/>

## #3: API - Idempotency Not Enforced

<b>Severity:</b>	<b>MEDIUM</b> (CVSS: 5.0/10.0)
<b>CWE ID:</b>	CWE-362 <b>OWASP:</b> A04:2021 - Insecure Design
<b>Category:</b>	api_security <b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	server.ts:603
<b>Line(s):</b>	603
<b>Exploitability:</b>	<a href="#">Unknown</a> <b>Impact:</b> Unknown

## Description

Detects PUT/DELETE without idempotency checks **Vulnerable Code**

```
app.put('/rest/order-history/:id/delivery-status', security.isAccounting(),
toggleDeliveryStatus())
```

## Security Impact

This moderate security weakness could be exploited to:

- Gather information about the system
- Perform limited unauthorized actions
- Facilitate further exploitation attempts

**Business Consequences:** Increased attack surface, potential stepping stone for advanced attacks  
**Remediation Guidance**

Review and fix the api\_security issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A04/>

## #4: API - Idempotency Not Enforced

<b>Severity:</b>	<b>MEDIUM</b> (CVSS: 5.0/10.0)	
<b>CWE ID:</b>	CWE-362	<b>OWASP:</b> A04:2021 - Insecure Design
<b>Category:</b>	api_security	<b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	server.ts:613	
<b>Line(s):</b>	613	
<b>Exploitability:</b>	<a href="#">Unknown</a>	<b>Impact:</b> Unknown

## Description

Detects PUT/DELETE without idempotency checks **Vulnerable Code**

```
app.put('/rest/products/:id/reviews', createProductReviews())
```

## Security Impact

This moderate security weakness could be exploited to:

- Gather information about the system
- Perform limited unauthorized actions
- Facilitate further exploitation attempts

**Business Consequences:** Increased attack surface, potential stepping stone for advanced attacks  
**Remediation Guidance**

Review and fix the api\_security issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A04/>

## #5: Business Logic - Minimum Order Value Bypass

<b>Severity:</b>	<b>MEDIUM</b> (CVSS: 5.0/10.0)	
<b>CWE ID:</b>	CWE-840	<b>OWASP:</b> A04:2021 - Insecure Design
<b>Category:</b>	business_logic	<b>Confidence:</b> MEDIUM
<b>Location:</b>	server.ts:582	
<b>Line(s):</b>	582	
<b>Exploitability:</b>	<a href="#">Unknown</a>	<b>Impact:</b> Unknown

### Description

Detects minimum order value bypass **Vulnerable Code**

```
app.post('/rest/basket/:id/checkout', placeOrder())
```

### Security Impact

This moderate security weakness could be exploited to:

- Gather information about the system
- Perform limited unauthorized actions
- Facilitate further exploitation attempts

**Business Consequences:** Increased attack surface, potential stepping stone for advanced attacks  
**Remediation Guidance**

Review and fix the business\_logic issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A04/>

## #6: Business Logic - Minimum Order Value Bypass

Severity:	MEDIUM (CVSS: 5.0/10.0)	
CWE ID:	CWE-840	OWASP: A04:2021 - Insecure Design
Category:	business_logic	Confidence: MEDIUM
Location:	test/api/basketApiSpec.ts:114	
Line(s):	114	
Exploitability:	Unknown	Impact: Unknown

### Description

Detects minimum order value bypass **Vulnerable Code**

```
describe('/rest/basket/:id/checkout', () => {
```

### Security Impact

This moderate security weakness could be exploited to:

- Gather information about the system
- Perform limited unauthorized actions
- Facilitate further exploitation attempts

**Business Consequences:** Increased attack surface, potential stepping stone for advanced attacks  
**Remediation Guidance**

Review and fix the business\_logic issue in test/api/basketApiSpec.ts **References & Further**

### Reading:

<https://owasp.org/Top10/A04/>

## #7: Business Logic - Minimum Order Value Bypass

Severity:	MEDIUM (CVSS: 5.0/10.0)	
CWE ID:	CWE-840	OWASP: A04:2021 - Insecure Design
Category:	business_logic	Confidence: MEDIUM
Location:	test/api/basketApiSpec.ts:129	
Line(s):	129	
Exploitability:	Unknown	Impact: Unknown

## Description

Detects minimum order value bypass **Vulnerable Code**

```
return frisby.post(REST_URL + '/basket/42/checkout', { headers: authHeader })
```

## Security Impact

This moderate security weakness could be exploited to:

- Gather information about the system
- Perform limited unauthorized actions
- Facilitate further exploitation attempts

**Business Consequences:** Increased attack surface, potential stepping stone for advanced attacks  
**Remediation Guidance**

Review and fix the business\_logic issue in test/api/basketApiSpec.ts **References & Further**

### Reading:

<https://owasp.org/Top10/A04/>

## #8: Business Logic - Minimum Order Value Bypass

**Severity:** MEDIUM (CVSS: 5.0/10.0)

**CWE ID:** CWE-840      **OWASP:** A04:2021 - Insecure Design

**Category:** business\_logic      **Confidence:** MEDIUM

**Location:** test/api/basketApiSpec.ts:141

**Line(s):** 141

**Exploitability:** Unknown      **Impact:** Unknown

## Description

Detects minimum order value bypass **Vulnerable Code**

```
return frisby.post(REST_URL + '/basket/3/checkout', { headers: authHeader })
```

## Security Impact

This moderate security weakness could be exploited to:

- Gather information about the system
- Perform limited unauthorized actions
- Facilitate further exploitation attempts

**Business Consequences:** Increased attack surface, potential stepping stone for advanced attacks  
**Remediation Guidance**

Review and fix the business\_logic issue in test/api/basketApiSpec.ts **References & Further**

#### Reading:

<https://owasp.org/Top10/A04/>

## #9: API - POST Without Content-Type Check

Severity:	MEDIUM (CVSS: 5.0/10.0)	
CWE ID:	CWE-436	OWASP: A04:2021 - Insecure Design
Category:	api_security	Confidence: MEDIUM
Location:	server.ts:303	
Line(s):	303	
Exploitability:	Unknown	Impact: Unknown

#### Description

Detects POST endpoints without Content-Type validation **Vulnerable Code**

```
app.post('/file-upload', uploadToMemory.single('file'), ensureFileIsPassed,
metrics.observeFileUploadMetricsMiddleware(), checkUploadSize, checkFileType,
handleZipFileUpload, handleXmlUpload, handleYamlUpload)
```

#### Security Impact

This moderate security weakness could be exploited to:

- Gather information about the system
- Perform limited unauthorized actions
- Facilitate further exploitation attempts

**Business Consequences:** Increased attack surface, potential stepping stone for advanced attacks  
**Remediation Guidance**

Review and fix the api\_security issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A04/>

## #10: API - POST Without Content-Type Check

Severity:	MEDIUM (CVSS: 5.0/10.0)	
CWE ID:	CWE-436	OWASP: A04:2021 - Insecure Design
Category:	api_security	Confidence: MEDIUM
Location:	server.ts:306	
Line(s):	306	
Exploitability:	Unknown	Impact: Unknown

### Description

Detects POST endpoints without Content-Type validation **Vulnerable Code**

```
app.post('/rest/memories', uploadToDisk.single('image'), ensureFileIsPassed,
security.appendUserId(), metrics.observeFileUploadMetricsMiddleware(), addMemory())
```

### Security Impact

This moderate security weakness could be exploited to:

- Gather information about the system
- Perform limited unauthorized actions
- Facilitate further exploitation attempts

**Business Consequences:** Increased attack surface, potential stepping stone for advanced attacks **Remediation Guidance**

Review and fix the api\_security issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A04/>

## #11: API - POST Without Content-Type Check

<b>Severity:</b>	<b>MEDIUM</b> (CVSS: 5.0/10.0)
<b>CWE ID:</b>	CWE-436
<b>Category:</b>	api_security
<b>Location:</b>	server.ts:362
<b>Line(s):</b>	362
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects POST endpoints without Content-Type validation [Vulnerable Code](#)

```
app.post('/api/Products', security.isAuthorized()) // vuln-code-snippet neutral-line
changeProductChallenge
```

### Security Impact

This moderate security weakness could be exploited to:

- Gather information about the system
- Perform limited unauthorized actions
- Facilitate further exploitation attempts

**Business Consequences:** Increased attack surface, potential stepping stone for advanced attacks **Remediation Guidance**

Review and fix the api\_security issue in server.ts [References & Further Reading](#):

<https://owasp.org/Top10/A04/>

## #12: API - POST Without Content-Type Check

<b>Severity:</b>	<b>MEDIUM</b> (CVSS: 5.0/10.0)
<b>CWE ID:</b>	CWE-436
<b>Category:</b>	api_security
<b>Location:</b>	server.ts:370
<b>Line(s):</b>	370
<b>Exploitability:</b>	<a href="#">Unknown</a>
<b>Impact:</b>	Unknown

### Description

Detects POST endpoints without Content-Type validation **Vulnerable Code**

```
app.post('/api/Complaints', security.isAuthorized())
```

## Security Impact

This moderate security weakness could be exploited to:

- Gather information about the system
- Perform limited unauthorized actions
- Facilitate further exploitation attempts

**Business Consequences:** Increased attack surface, potential stepping stone for advanced attacks  
**Remediation Guidance**

Review and fix the api\_security issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A04/>

## #13: API - POST Without Content-Type Check

<b>Severity:</b>	<b>MEDIUM</b> (CVSS: 5.0/10.0)	
<b>CWE ID:</b>	CWE-436	<b>OWASP:</b> A04:2021 - Insecure Design
<b>Category:</b>	api_security	<b>Confidence:</b> <b>MEDIUM</b>
<b>Location:</b>	server.ts:380	
<b>Line(s):</b>	380	
<b>Exploitability:</b>	<b>Unknown</b>	<b>Impact:</b> Unknown

## Description

Detects POST endpoints without Content-Type validation **Vulnerable Code**

```
app.post('/api/SecurityQuestions', security.denyAll())
```

## Security Impact

This moderate security weakness could be exploited to:

- Gather information about the system
- Perform limited unauthorized actions

- Facilitate further exploitation attempts

**Business Consequences:** Increased attack surface, potential stepping stone for advanced attacks **Remediation Guidance**

Review and fix the api\_security issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A04/>

## #14: API - POST Without Content-Type Check

**Severity:** MEDIUM (CVSS: 5.0/10.0)

**CWE ID:** CWE-436      **OWASP:** A04:2021 - Insecure Design

**Category:** api\_security      **Confidence:** MEDIUM

**Location:** server.ts:390

**Line(s):** 390

**Exploitability:** Unknown      **Impact:** Unknown

### Description

Detects POST endpoints without Content-Type validation **Vulnerable Code**

```
app.post('/api/Feedbacks', verify.forgedFeedbackChallenge())
```

### Security Impact

This moderate security weakness could be exploited to:

- Gather information about the system
- Perform limited unauthorized actions
- Facilitate further exploitation attempts

**Business Consequences:** Increased attack surface, potential stepping stone for advanced attacks **Remediation Guidance**

Review and fix the api\_security issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A04/>

## #15: API - POST Without Content-Type Check

Severity:	MEDIUM (CVSS: 5.0/10.0)	
CWE ID:	CWE-436	OWASP: A04:2021 - Insecure Design
Category:	api_security	Confidence: MEDIUM
Location:	server.ts:408	
Line(s):	408	
Exploitability:	Unknown	Impact: Unknown

## Description

Detects POST endpoints without Content-Type validation **Vulnerable Code**

```
app.post('/api/Users', verify.registerAdminChallenge())
```

## Security Impact

This moderate security weakness could be exploited to:

- Gather information about the system
- Perform limited unauthorized actions
- Facilitate further exploitation attempts

**Business Consequences:** Increased attack surface, potential stepping stone for advanced attacks  
**Remediation Guidance**

Review and fix the api\_security issue in server.ts **References & Further Reading:**

<https://owasp.org/Top10/A04/>

**Additional Medium Severity Findings:** 226 more medium severity issues are documented in the full scan results. Access the complete report via the SecureThread OPS dashboard for detailed remediation guidance.

## 0.2.4 Low Severity Issues (55 found)

### Maintenance Priority

Low severity issues have **minimal immediate risk** but should be addressed during regular maintenance to improve overall code quality and security posture.

**Recommended Timeline:** Fix during **regular maintenance cycles**

The following table summarizes low-severity findings. Full details are available in the SecureThread OPS dashboard.

ID	Title	Category	Location
#242	API - Conditional Request Headers Ignored	api_security	server.ts:429
#243	API - Conditional Request Headers Ignored	api_security	server.ts:438
#244	API - Conditional Request Headers Ignored	api_security	server.ts:583
#245	API - Conditional Request Headers Ignored	api_security	data/static/codefixes/changeProductChall...
#246	API - Conditional Request Headers Ignored	api_security	data/static/codefixes/changeProductChall...
#247	API - Conditional Request Headers Ignored	api_security	server.ts:591
#248	API - Conditional Request Headers Ignored	api_security	server.ts:603
#249	API - Conditional Request Headers Ignored	api_security	server.ts:613
#250	API - No Gradual Backoff	api_security	server.ts:337
#251	API - No Gradual Backoff	api_security	server.ts:447
#252	API - No Gradual Backoff	api_security	server.ts:454
#253	API - No Gradual Backoff	api_security	server.ts:460
#254	API - Conditional Request Headers Ignored	api_security	data/static/codefixes/changeProductChall...
#255	API - Conditional Request Headers Ignored	api_security	data/static/codefixes/changeProductChall...
#256	API - Conditional Request Headers Ignored	api_security	data/static/codefixes/changeProductChall...
#257	API - Conditional Request Headers Ignored	api_security	data/static/codefixes/changeProductChall...

ID	Title	Category	Location
#258	API - No Gradual Backoff	api_security	data/static/codefixes/resetPasswordMorty...
#259	API - Conditional Request Headers Ignored	api_security	data/static/codefixes/changeProductChall...
#260	API - Conditional Request Headers Ignored	api_security	data/static/codefixes/changeProductChall...
#261	API - Conditional Request Headers Ignored	api_security	data/static/codefixes/changeProductChall...
#262	API - Conditional Request Headers Ignored	api_security	data/static/codefixes/changeProductChall...
#263	API - Conditional Request Headers Ignored	api_security	data/static/codefixes/changeProductChall...
#264	API - Conditional Request Headers Ignored	api_security	data/static/codefixes/changeProductChall...
#265	API - Conditional Request Headers Ignored	api_security	data/static/codefixes/changeProductChall...
#266	API - Conditional Request Headers Ignored	api_security	data/static/codefixes/changeProductChall...

*Note: 30 additional low-severity findings are available in the complete scan results.*

## 0.3 Strategic Recommendations \& Remediation Roadmap

This section provides actionable recommendations prioritized by risk, impact, and remediation effort. Follow this roadmap to systematically improve your application's security posture.

### 0.3.1 Executive Action Plan: Top 5 Priorities

The following actions should be prioritized for immediate executive attention based on risk severity, business impact, and compliance requirements.

### Priority #1: Address Critical Security Vulnerabilities

**Rationale:** Found 269 critical vulnerabilities that pose immediate risk of data breach or system compromise.

**Business Impact:** Prevents potential data breaches, system compromise, and regulatory penalties

**Estimated Effort:** 3228 hours

**Recommended Timeline:** **0-7 days (Immediate)**

### Priority #2: Achieve Regulatory Compliance

**Rationale:** Identified 16 compliance violations across OWASP, PCI DSS, and GDPR frameworks.

**Business Impact:** Ensures regulatory compliance, avoids fines (up to 4% revenue for GDPR)

**Estimated Effort:** Legal and compliance review required

**Recommended Timeline:** **7-30 days**

### Priority #3: Remediate High-Severity Vulnerabilities

**Rationale:** 663 high-severity issues require attention before production deployment.

**Business Impact:** Reduces attack surface and exploitation risk significantly

**Estimated Effort:** 3978 hours

**Recommended Timeline:** **14-30 days**

### Priority #4: Refactor Security Hotspots

**Rationale:** Multiple vulnerabilities concentrated in server.ts and other high-risk files.

**Business Impact:** Improves overall code quality and reduces maintenance burden

**Estimated Effort:** 40-80 hours (architectural review)

**Recommended Timeline:** **30-60 days**

**Priority #5: Integrate Security into Development Lifecycle**

**Rationale:** Establish secure coding practices, automated scanning, and security training.

**Business Impact:** Prevents future vulnerabilities, reduces long-term security costs

**Estimated Effort:** Ongoing process improvement

**Recommended** **60-90 days**

**Timeline:**

### 0.3.2 Remediation Roadmap: Sprint Planning

This roadmap organizes remediation activities into 2-week sprints, allowing for iterative security improvements while maintaining development velocity.

Sprint	Focus Areas	Issues	Effort	Team
<b>Sprint 1</b> (Week 1-2)	<b>Critical Vulnerabilities</b> <ul style="list-style-type: none"><li>All critical security issues</li><li>Top 3 high-severity issues</li><li>Emergency patches</li></ul>	<b>10</b>	3246 hrs (405.8 days)	2-3 devs
<b>Sprint 2</b> (Week 3-4)	<b>High &amp; Medium Issues</b> <ul style="list-style-type: none"><li>Remaining high-severity</li><li>Top medium-severity</li><li>Compliance gaps</li></ul>	<b>665</b>	3975 hrs (496.9 days)	2 devs
<b>Sprint 3</b> (Week 5-6)	<b>Medium &amp; Low Issues</b> <ul style="list-style-type: none"><li>Remaining medium issues</li><li>Quick-win low issues</li><li>Code quality improvements</li></ul>	<b>246</b>	718 hrs (89.8 days)	1-2 devs
<b>Sprint 4+</b> (Ongoing)	<b>Continuous Improvement</b> <ul style="list-style-type: none"><li>Security training</li><li>Automated testing integration</li><li>Security documentation</li></ul>	<b>Ongoing</b>	Ongoing	Full team

#### Remediation Timeline Summary

<b>Total Sprints:</b>	3 (6 weeks)
<b>Total Effort:</b>	7984 hours (998.0 days)
<b>Estimated Cost:</b>	\$1,623,700.00
<b>Team Size:</b>	2-3 developers (recommended)
<b>Completion Target:</b>	6-8 weeks from start

### 0.3.3 Quick Wins: Low Effort, High Impact

The following improvements can be implemented quickly (< 4 hours each) but provide significant security value. Prioritize these for immediate wins.

Most vulnerabilities require moderate to significant effort. Focus on the sprint roadmap above.

### 0.3.4 Secure Coding Best Practices

Implement these secure coding standards to prevent future vulnerabilities:

#### Error Handling & Logging

- Never expose stack traces or internal paths to users
- Log all security-relevant events (auth, access, changes)
- Implement centralized logging with tamper-proof storage
- Set up real-time alerting for security anomalies

#### Dependency & Supply Chain Security

- Keep all dependencies updated to latest stable versions
- Use automated tools (Dependabot, Snyk) for vulnerability scanning
- Verify package integrity using checksums or signatures
- Minimize dependency count and audit third-party code

### 0.3.5 Recommended Security Tooling

Integrate the following tools into your development pipeline for continuous security:

Tool Category	Recommended Tools	Integration Point
<b>SAST</b> (Static Analysis)	SecureThread OPS, SonarQube, Semgrep	Pre-commit hooks, CI/CD pipeline
<b>DAST</b> (Dynamic Analysis)	OWASP ZAP, Burp Suite, Acunetix	Staging environment, nightly scans
<b>SCA</b> (Dependency Scanning)	Snyk, Dependabot, WhiteSource	PR checks, scheduled scans
<b>Secrets Detection</b>	GitGuardian, TruffleHog, detect-secrets	Pre-commit, repository scanning
<b>Container Security</b>	Trivy, Clair, Anchore	Docker build, registry scanning
<b>IaC Security</b>	Checkov, Terrascan, tfsec	Terraform/CloudFormation validation

### 0.3.6 Security Metrics & KPIs

Track these key performance indicators to measure security improvement over time:

#### Current Baseline Metrics

**Security Score:** 0.0/100

**Vulnerability Density:** 17.79/1k LOC

**Critical Issues:** 269

**High Issues:** 663

**Compliance Grade:** N/A

**Fixable Issues:** 1228/1228

#### Target Metrics (After Remediation):

- Security Score: **25.0+/100** (improve by 25+ points)
- Vulnerability Density: **< 5.3/1k LOC** (reduce by 70%)
- Critical Issues: **0** (complete elimination)
- High Issues: **< 3** (minimal acceptable risk)
- Time to Remediate: **< 7 days** (for critical issues)

# .1 Scanning Methodology

This appendix describes the methodology, tools, and techniques used to conduct this security assessment.

## .1.1 Assessment Approach

**Assessment Type:** Static Application Security Testing (SAST)

**Scope:** Full repository source code analysis including all branches and commits.

**Methodology:** Hybrid approach combining:

- **Pattern-based Detection:** Regular expression and syntax tree analysis
- **Rule-based Analysis:** Custom security rules (OWASP, SANS, CWE)
- **Data Flow Analysis:** Taint tracking for injection vulnerabilities
- **LLM Enhancement:** AI-powered context analysis for false positive reduction
- **Semantic Analysis:** Understanding code intent and business logic flaws

## .1.2 Scan Configuration

Configuration Item	Value
Scan Engine	SecureThread OPS v4.0
Repository	dummyhshz/juice-shop
Branch	master
Commit Hash	HEAD
Primary Language	Multi-language
Files Scanned	863
Estimated Lines of Code	69,040
Scan Duration	4m 42s
Scan Date	22 December 2025 12:21
Rules Applied	Default ruleset
LLM Enhancement	Disabled

## .1.3 Severity Classification

Vulnerabilities are classified using a risk-based severity model:

Severity	CVSS Range	Description
CRITICAL	9.0 - 10.0	Immediate threat of complete system compromise, data breach, or service disruption. Exploitable remotely without authentication.
HIGH	7.0 - 8.9	Significant security risk allowing unauthorized access, data exposure, or privilege escalation. May require some user interaction.
MEDIUM	4.0 - 6.9	Moderate security weakness that could lead to information disclosure or limited access. Typically requires specific conditions to exploit.
LOW	0.1 - 3.9	Minor security concern with minimal impact. Difficult to exploit or requires extensive preconditions.
INFO	0.0	Informational finding or security best practice recommendation without direct exploitability.

## .1.4 False Positive Management

SecureThread OPS employs multiple techniques to minimize false positives:

- Context-Aware Analysis:** Understanding code flow and business logic
- Framework Detection:** Recognizing security features in frameworks (Django, Spring, etc.)
- Confidence Scoring:** Each finding includes a confidence level (High/Medium/Low)
- LLM Validation:** AI review of potential findings for contextual accuracy
- Community Rules:** Continuously updated rules based on real-world feedback

**Note:** While we strive for accuracy, manual review by security experts is recommended for production deployments.

## .2 Security Hotspots Analysis

Security hotspots are files or modules with concentrated vulnerabilities, indicating areas requiring architectural review or refactoring.

### .2.1 File-Level Hotspots

The following files contain the highest concentration of security issues:

Rank	File Path	Risk Score	Priority
1	server.ts	297	URGENT
2	data/static/users.yml	200	URGENT
3	test/api/quantityApiSpec.ts	170	URGENT
4	test/api/loginApiSpec.ts	135	URGENT
5	data/static/codedefixes/changeProductChallenge_3_correct.ts	124	URGENT
6	...src/app/product-details/product-details.component.spec.ts	120	URGENT
7	data/static/codedefixes/changeProductChallenge_4.ts	105	URGENT
8	data/static/codedefixes/changeProductChallenge_1.ts	103	URGENT
9	test/api/deluxeApiSpec.ts	102	URGENT
10	data/static/codedefixes/changeProductChallenge_2.ts	95	URGENT

**Recommendation:** Files with Risk Score  $\geq 15$  should undergo comprehensive security review and potential refactoring.

### .2.2 Directory-Level Hotspots

The following directories contain the highest concentration of security issues:

Rank	Directory	Risk Score	Action
1	test/api	1228	Refactor
2	frontend/src/app/Services	950	Refactor
3	data/static/codefixes	879	Refactor
4	frontend/src/assets/i18n	652	Refactor
5	test/cypress/e2e	333	Refactor

## .3 CWE Reference Guide

This section provides detailed information about the Common Weakness Enumeration (CWE) identifiers found in this assessment.

CWE ID	Description	Count	Impact
CWE-306	<b>Missing Authentication</b>	387	Critical functions lack auth checks
CWE-798	<b>Hard-coded Credentials</b>	176	Embedded passwords/keys in code
CWE-327	<b>Broken Crypto</b>	138	Use of weak cryptographic algorithms
CWE-352	<b>CSRF</b>	122	Cross-Site Request Forgery attacks
CWE-362	<b>Security Weakness</b>	68	Potential security risk
CWE-79	<b>Cross-site Scripting (XSS)</b>	44	Code injection via untrusted user input
CWE-436	<b>Security Weakness</b>	41	Potential security risk
CWE-840	<b>Security Weakness</b>	34	Potential security risk
CWE-862	<b>Security Weakness</b>	34	Potential security risk
CWE-548	<b>Security Weakness</b>	31	Potential security risk

**Reference:** For complete CWE descriptions and mitigation guidance, visit:

<https://cwe.mitre.org/>

## .4 Glossary of Terms

### Common security and technical terms used in this report:

<b>SAST</b>	Static Application Security Testing - Analysis of source code without execution
<b>DAST</b>	Dynamic Application Security Testing - Analysis of running applications
<b>CVSS</b>	Common Vulnerability Scoring System - Standardized severity rating (0-10)
<b>CWE</b>	Common Weakness Enumeration - Dictionary of software weakness types
<b>CVE</b>	Common Vulnerabilities and Exposures - Public vulnerability database
<b>OWASP</b>	Open Web Application Security Project - Leading security standards body
<b>XSS</b>	Cross-Site Scripting - Code injection through untrusted web input
<b>SQL Injection</b>	Database attack via unsanitized SQL queries
<b>CSRF</b>	Cross-Site Request Forgery - Unauthorized actions via authenticated user
<b>SSRF</b>	Server-Side Request Forgery - Unauthorized internal resource access
<b>XXE</b>	XML External Entity - XML parser exploitation vulnerability
<b>PCI DSS</b>	Payment Card Industry Data Security Standard - Payment security requirements
<b>GDPR</b>	General Data Protection Regulation - EU data privacy law
<b>Zero-Day</b>	Vulnerability unknown to vendor with no available patch
<b>Supply Chain Attack</b>	Compromise through third-party dependencies
<b>Threat Modeling</b>	Systematic identification of security threats
<b>Attack Surface</b>	Total exposure points vulnerable to exploitation
<b>Least Privilege</b>	Minimal access rights principle
<b>Defense in Depth</b>	Layered security approach
<b>Shift Left</b>	Early integration of security in development lifecycle

## .5 References \& Resources

### Industry Standards and Frameworks:

- OWASP Top 10 2021  
<https://owasp.org/Top10/>
- SANS Top 25 Most Dangerous Software Weaknesses  
<https://www.sans.org/top25-software-errors/>
- CWE/SANS Top 25  
<https://cwe.mitre.org/top25/>
- NIST Secure Software Development Framework (SSDF)  
<https://csrc.nist.gov/Projects/ssdf>
- PCI DSS v4.0 Requirements  
<https://www.pcisecuritystandards.org/>
- GDPR Security Requirements (Article 32)  
<https://gdpr-info.eu/art-32-gdpr/>

### Vulnerability Databases:

- National Vulnerability Database (NVD)  
<https://nvd.nist.gov/>
- MITRE CVE  
<https://cve.mitre.org/>
- Exploit Database  
<https://www.exploit-db.com/>
- Snyk Vulnerability Database  
<https://snyk.io/vuln/>

### Secure Coding Guidelines:

- OWASP Secure Coding Practices  
<https://owasp.org/www-project-secure-coding-practices-quick-reference-guide/>
- SEI CERT Coding Standards  
<https://wiki.sei.cmu.edu/confluence/display/seccode/>
- Microsoft Security Development Lifecycle  
<https://www.microsoft.com/en-us/securityengineering/sdl/>
- Google Security Best Practices  
<https://cloud.google.com/security/best-practices>

### Security Tools and Services:

- **SecureThread OPS Documentation**

<https://securethread.io/docs>

- **OWASP ZAP (Dynamic Scanner)**

<https://www.zaproxy.org/>

- **SonarQube (Code Quality)**

<https://www.sonarqube.org/>

- **Snyk (Dependency Scanner)**

<https://snyk.io/>

- **GitHub Security Features**

<https://github.com/security>

## .6 Report Metadata \& Legal Disclaimer

### Report Information

**Report ID:** ST-180-20251231-1301  
**Generated:** 31 December 2025 13:01 UTC  
**Analyst:** Dummy Test  
**Tool Version:** SecureThread OPS v4.0  
**Report Version:** 1.0  
**Classification:** CONFIDENTIAL

### Legal Disclaimer

This security assessment report is provided "as-is" for informational purposes only. While SecureThread OPS employs industry-leading detection techniques, no automated security tool can guarantee 100% accuracy or detect all vulnerabilities.

#### Limitations:

- This report reflects a point-in-time assessment of the scanned codebase
- Static analysis cannot detect runtime or configuration vulnerabilities
- Business logic flaws may require manual security review
- False positives may occur and require expert validation
- New vulnerabilities may be discovered after report generation

#### Recommendations:

- Conduct manual penetration testing for production systems
- Perform regular security assessments (quarterly recommended)
- Implement defense-in-depth security controls
- Train development teams on secure coding practices
- Maintain an incident response plan

## Confidentiality Notice

This document contains confidential and proprietary security information. Distribution is restricted to authorized personnel only. Unauthorized disclosure, copying, or distribution may result in legal liability.

For questions or concerns regarding this report, contact:

**SecureThread Security Operations Center (SOC)**

Email: [security@securethread.io](mailto:security@securethread.io)

Support Portal: <https://support.securethread.io>

---

*End of Report*

**SecureThread OPS - Securing the Digital Future**