

API PENETRATION TESTING REPORT VERSION 1.3

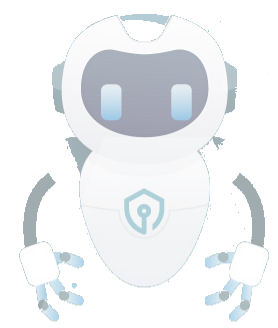


APIsec

www.apisec.ai

TABLE OF CONTENTS

Report Summary	3
Introduction	4
Business Risk	4
Coverage Overview	5
Detected Vulnerabilities	6
Closed Vulnerabilities	7
Review/False-Positives	8
Tested/Discovered Endpoints	10
Tested Categories	13
Remediations	14
About APIsec Inc.	15



Report Summary

Endpoints: 75

Security Tests: 583

Passed Tests: 508

Failed Tests: 75

Ignored/Review-required/False-positives: 47

Total Vulnerabilities: 21

ACTIVE



Count of all active vulnerabilities in the project

NEW THIS MONTH



Count of vulnerabilities have been logged within the past 30 days

CLOSED



Count of all closed vulnerabilities in the project

CLOSED THIS MONTH



Count of vulnerabilities have been closed within the past 30 days

Est. Bug Bounty

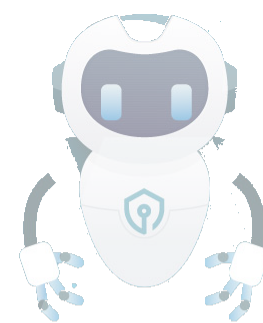


Estimated savings in the identification and discovery of security vulnerabilities

Est. Fix Time (Hours)



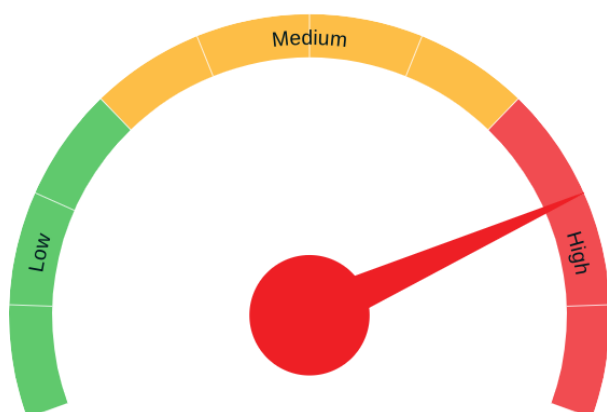
Estimated time required to fix the active vulnerabilities.






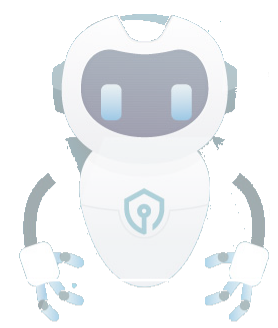
Introduction

Report Title : API Penetration Testing Report
Report By : APIsec Bot
Report For : EthicalCheck
Project Name : Online Banking REST API rSaF
Date : September, 13, 2022

Business Risk:

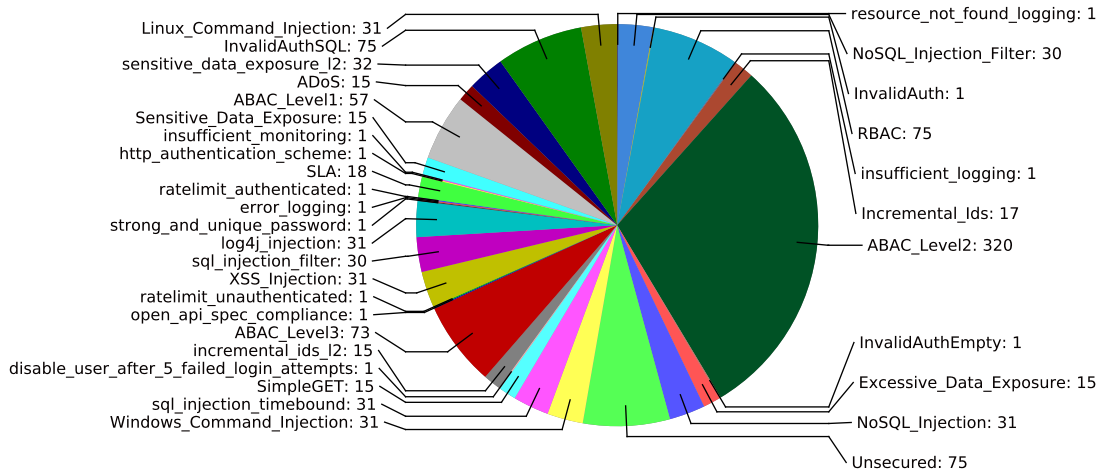


-  High-security risk. Several critical and high severity and commonly exploited vulnerabilities are active.
-  Medium-security risk. Several high-severity and commonly exploited vulnerabilities are active.
-  Low-security risk. Several medium and low severity vulnerabilities that impact API availability and security are active.



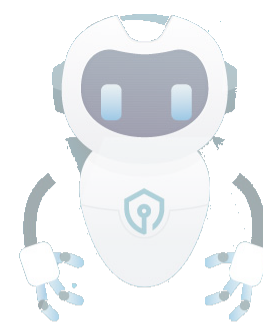
Coverage Overview

Coverage Overview



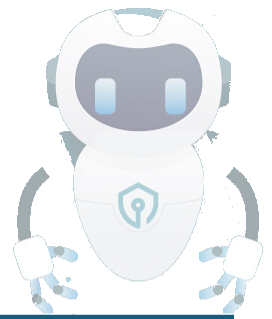
- resource_not_found_logging ● NoSQL_Injection_Filter ● InvalidAuth ● RBAC ● insufficient_logging
- Incremental_Ids ● ABAC_Level2 ● InvalidAuthEmpty ● Excessive_Data_Exposure ● NoSQL_Injection
- Unsecured ● Windows_Command_Injection ● sql_injection_timebound ● SimpleGET
- disable_user_after_5_failed_login_attempts ● incremental_ids_I2 ● ABAC_Level3
- open_api_spec_compliance ● ratelimit_unauthenticated ● XSS_Injection ● sql_injection_filter
- log4j_injection ● strong_and_unique_password ● error_logging ● ratelimit_authenticated ● SLA
- http_authentication_scheme ● insufficient_monitoring ● Sensitive_Data_Exposure ● ABAC_Level1
- ADoS ● sensitive_data_exposure_I2 ● InvalidAuthSQL ● Linux_Command_Injection

This chart aims in creating awareness for the project's risk coverage and test areas completion. APIsec aims at covering all the OWASP Top 10 security risks in its automated test cycles and being a stepping stone for development teams' cultural changes to ensure secure coding as a continuous process.



Detected Vulnerabilities

S/N	OWASP	Category	Endpoint		CVSS 3.1	Severity	Logged on
1	#7	<u>http_authentication_scheme</u>	PUT:/api/v1/primary-account/deposit-amount/{id}	NEW	9.1	Critical	Sep 13 2022
2	#2	<u>Unsecured</u>	DELETE:/api/v1/primary-transaction/{id}	NEW	9.1	Critical	Sep 13 2022
3	#2	<u>Unsecured</u>	GET:/api/v1/primary-transaction	NEW	9.1	Critical	Sep 13 2022
4	#2	<u>Unsecured</u>	GET:/api/v1/primary-transaction/{id}	NEW	9.1	Critical	Sep 13 2022
5	#2	<u>Unsecured</u>	PUT:/api/v1/primary-transaction	NEW	9.1	Critical	Sep 13 2022
6	#1	<u>ABAC_Level1</u>	POST:/api/v1/users/team-sign-up	NEW	8.1	High	Sep 13 2022
7	#4	<u>ADoS</u>	GET:/api/v1/primary-account/primary-account	NEW	6.5	Medium	Sep 13 2022
8	#4	<u>ADoS</u>	GET:/api/v1/issues/product/{projectId}	NEW	6.5	Medium	Sep 13 2022
9	#4	<u>ADoS</u>	GET:/api/v1/orgs	NEW	6.5	Medium	Sep 13 2022
10	#4	<u>ADoS</u>	GET:/api/v1/orgs/{id}/users	NEW	6.5	Medium	Sep 13 2022
11	#4	<u>ADoS</u>	GET:/api/v1/transfers	NEW	6.5	Medium	Sep 13 2022
12	#4	<u>ADoS</u>	GET:/api/v1/orgs/by-user	NEW	6.5	Medium	Sep 13 2022
13	#4	<u>ADoS</u>	GET:/api/v1/receipient	NEW	6.5	Medium	Sep 13 2022
14	#4	<u>ADoS</u>	GET:/api/v1/bank-account	NEW	6.5	Medium	Sep 13 2022
15	#4	<u>ADoS</u>	GET:/api/v1/primary-transaction	NEW	6.5	Medium	Sep 13 2022
16	#4	<u>ADoS</u>	GET:/api/v1/orgs/allorgs	NEW	6.5	Medium	Sep 13 2022
17	#4	<u>ADoS</u>	GET:/api/v1/savings-account/savings-account	NEW	6.5	Medium	Sep 13 2022
18	#4	<u>ADoS</u>	GET:/api/v1/savings-transaction	NEW	6.5	Medium	Sep 13 2022
19	#4	<u>ADoS</u>	GET:/api/v1/orgs/find-by-name/{name}	NEW	6.5	Medium	Sep 13 2022
20	#4	<u>ADoS</u>	GET:/api/v1/orgs/search	NEW	6.5	Medium	Sep 13 2022
21	#4	<u>ADoS</u>	GET:/api/v1/products	NEW	6.5	Medium	Sep 13 2022



Closed Vulnerabilities

OWASP	Category	Endpoint	CVSS 3.1	Severity	Closed on
-------	----------	----------	----------	----------	-----------

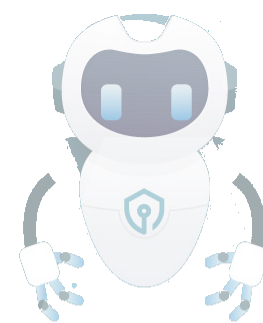
No items to show



Review/False-Positives

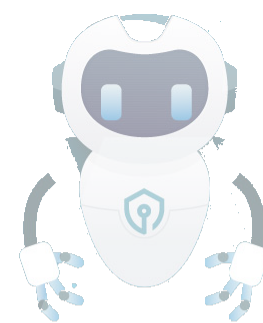
False-Positives occur when an AI Bot, flags a security vulnerability that product do not have. A Security Researcher needs to validate the true False-Positive

S/N	OWASP	Category	Endpoint	AI comment	Logged on
1	#Top 25	SLA	GET:/api/v1/savings-account/savings-account	Failed	Sep 13 2022
2	#3	Incremental_Ids	POST:/api/v1/savings-account/savings-account	Failed	Sep 13 2022
3	#3	Incremental_Ids	POST:/api/v1/users/team-sign-up	Failed	Sep 13 2022
4	#3	Incremental_Ids	POST:/api/v1/orgs	Failed	Sep 13 2022
5	#3	Incremental_Ids	POST:/api/v1/savings-transaction	Failed	Sep 13 2022
6	#2	InvalidAuthSQL	POST:/api/v1/users/enterprise-sign-up	3 Failed	Sep 13 2022
7	#3	Incremental_Ids	POST:/api/v1/products	Failed	Sep 13 2022
8	#3	Incremental_Ids	POST:/api/v1/users/enterprise-sign-up	Failed	Sep 13 2022
9	#Top 25	SLA	GET:/api/v1/savings-transaction	Failed	Sep 13 2022
10	#Top 25	SLA	GET:/api/v1/users/status	Failed	Sep 13 2022
11	#Top 25	SLA	GET:/api/v1/receipient	Failed	Sep 13 2022
12	#Top 25	SLA	GET:/api/v1/orgs/search	Failed	Sep 13 2022
13	#Top 25	SLA	GET:/api/v1/orgs	Failed	Sep 13 2022
14	#Top 25	SLA	GET:/api/v1/orgs/login-status	Failed	Sep 13 2022
15	#Top 25	SLA	GET:/api/v1/orgs/allorgs	Failed	Sep 13 2022
16	#Top 25	SLA	GET:/api/v1/primary-transaction	Failed	Sep 13 2022
17	#3	Incremental_Ids	POST:/api/v1/users/personal-sign-up	Failed	Sep 13 2022
18	#3	Incremental_Ids	POST:/api/v1/branches	Failed	Sep 13 2022
19	#Top 25	SLA	GET:/api/v1/branches	Failed	Sep 13 2022
20	#2	Unsecured	POST:/api/v1/users/enterprise-sign-up	Failed	Sep 13 2022
21	#3	Incremental_Ids	POST:/api/v1/bank-account	Failed	Sep 13 2022
22	#Top 25	SLA	GET:/api/v1/orgs/find-by-name/{name}	Failed	Sep 13 2022
23	#Top 25	SLA	GET:/api/v1/issues/product/{projectId}	Failed	Sep 13 2022
24	#3	Incremental_Ids	POST:/api/v1/orgs/{branchId}/users/{userId}/reset-password	Failed	Sep 13 2022
25	#3	Incremental_Ids	POST:/api/v1/primary-account/primary-account	Failed	Sep 13 2022
26	#2	Unsecured	POST:/api/v1/users/team-sign-up	Failed	Sep 13 2022
27	#3	Incremental_Ids	POST:/api/v1/issues/ui	Failed	Sep 13 2022
28	#3	Incremental_Ids	POST:/api/v1/receipient	Failed	Sep 13 2022
29	#2	Unsecured	POST:/api/v1/users/personal-sign-up	Failed	Sep 13 2022
30	#2	InvalidAuthSQL	POST:/api/v1/users/team-sign-up	3 Failed	Sep 13 2022
31	#3	Incremental_Ids	POST:/api/v1/transfers	Failed	Sep 13 2022



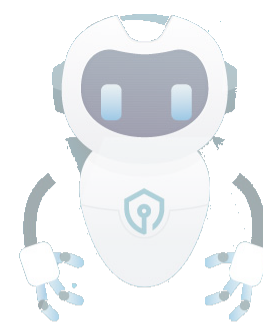
Review/False-Positives

S/N	OWASP	Category	Endpoint	AI comment	Logged on
32	#2	InvalidAuthSQL	POST:/api/v1/users/personal-sign-up	3 Failed	Sep 13 2022
33	#Top 25	SLA	GET:/api/v1/products	Failed	Sep 13 2022
34	#Top 25	SLA	GET:/api/v1/transfers	Failed	Sep 13 2022
35	#Top 25	SLA	GET:/api/v1/primary-account/primary-account	Failed	Sep 13 2022
36	#3	Incremental_Ids	POST:/api/v1/primary-transaction	Failed	Sep 13 2022
37	#3	Incremental_Ids	POST:/api/v1/issues	Failed	Sep 13 2022
38	#Top 25	SLA	GET:/api/v1/bank-account	Failed	Sep 13 2022
39	#Top 25	SLA	GET:/api/v1/orgs/{id}/users	Failed	Sep 13 2022
40	#Top 25	SLA	GET:/api/v1/orgs/by-user	Failed	Sep 13 2022
41	#3	Incremental_Ids	POST:/api/v1/orgs/{branchId}/users/add-member	Failed	Sep 13 2022



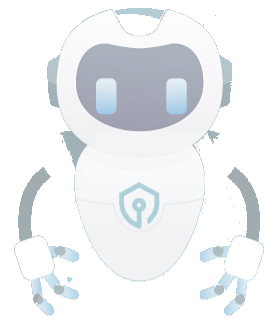
Tested/Discovered Endpoints

S/N	Endpoint	Tested
1	PUT : /api/v1/bank-account	OK
2	POST : /api/v1/bank-account	OK
3	GET : /api/v1/bank-account	X
4	PUT : /api/v1/bank-account/deposit-amount	OK
5	PUT : /api/v1/bank-account/withdraw-	OK
6	DELETE : /api/v1/bank-account/{id}	OK
7	GET : /api/v1/bank-account/{id}	OK
8	GET : /api/v1/branches	OK
9	POST : /api/v1/branches	OK
10	PUT : /api/v1/branches/update	OK
11	GET : /api/v1/branches/{id}	OK
12	DELETE : /api/v1/branches/{id}	OK
13	PUT : /api/v1/issues	OK
14	POST : /api/v1/issues	OK
15	GET : /api/v1/issues/product/{projectId}	X
16	DELETE : /api/v1/issues/product/{projectId}	OK
17	POST : /api/v1/issues/ui	OK
18	PUT : /api/v1/issues/ui	OK
19	GET : /api/v1/issues/{id}	OK
20	GET : /api/v1/orgs	X
21	POST : /api/v1/orgs	OK
22	GET : /api/v1/orgs/allorgs	X
23	GET : /api/v1/orgs/by-user	X
24	GET : /api/v1/orgs/find-by-name/{name}	X
25	GET : /api/v1/orgs/login-status	OK
26	GET : /api/v1/orgs/search	X
27	GET : /api/v1/orgs/{branchId}/branch-user/	OK
28	POST : /api/v1/orgs/{branchId}/users/add-	OK
29	PUT : /api/v1/orgs/{branchId}/users/	OK
30	POST : /api/v1/orgs/{branchId}/users/	OK



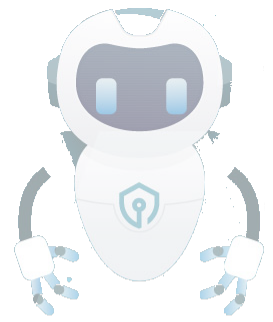
Tested/Discovered Endpoints

S/N	Endpoint	Tested
31	GET : /api/v1/orgs/{id}	OK
32	DELETE : /api/v1/orgs/{id}	OK
33	PUT : /api/v1/orgs/{id}	OK
34	GET : /api/v1/orgs/{id}/users	X
35	PUT : /api/v1/primary-account/deposit-	X
36	POST : /api/v1/primary-account/primary-	OK
37	GET : /api/v1/primary-account/primary-	X
38	PUT : /api/v1/primary-account/primary-	OK
39	DELETE : /api/v1/primary-account/primary-	OK
40	GET : /api/v1/primary-account/primary-	OK
41	PUT : /api/v1/primary-account/withdraw-	OK
42	PUT : /api/v1/primary-transaction	X
43	POST : /api/v1/primary-transaction	OK
44	GET : /api/v1/primary-transaction	X
45	GET : /api/v1/primary-transaction/{id}	X
46	DELETE : /api/v1/primary-transaction/{id}	X
47	GET : /api/v1/products	X
48	POST : /api/v1/products	OK
49	PUT : /api/v1/products	OK
50	DELETE : /api/v1/products/{id}	OK
51	GET : /api/v1/products/{id}	OK
52	GET : /api/v1/receipient	X
53	POST : /api/v1/receipient	OK
54	PUT : /api/v1/receipient	OK
55	DELETE : /api/v1/receipient/{id}	OK
56	GET : /api/v1/receipient/{id}	OK
57	PUT : /api/v1/savings-account/savings-	OK
58	GET : /api/v1/savings-account/savings-	X
59	POST : /api/v1/savings-account/savings-	OK
60	GET : /api/v1/savings-account/savings-	OK



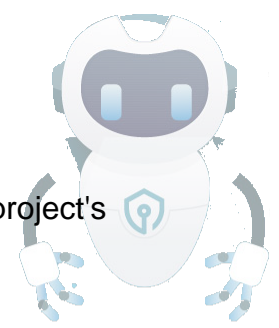
Tested/Discovered Endpoints

S/N	Endpoint	Tested
61	DELETE : /api/v1/savings-account/savings-	OK
62	GET : /api/v1/savings-transaction	X
63	POST : /api/v1/savings-transaction	OK
64	PUT : /api/v1/savings-transaction	OK
65	GET : /api/v1/savings-transaction/{id}	OK
66	DELETE : /api/v1/savings-transaction/{id}	OK
67	POST : /api/v1/transfers	OK
68	PUT : /api/v1/transfers	OK
69	GET : /api/v1/transfers	X
70	DELETE : /api/v1/transfers/{id}	OK
71	GET : /api/v1/transfers/{id}	OK
72	POST : /api/v1/users/enterprise-sign-up	OK
73	POST : /api/v1/users/personal-sign-up	OK
74	GET : /api/v1/users/status	OK
75	POST : /api/v1/users/team-sign-up	X



Tested Categories

OWASP Coverage	
#1 Broken Object Level Auth	✓
#2 Broken User Auth	✓
#3 Excessive Data Exposure	✓
#4 Rate Limiting	✓
#5 Broken Function Level Auth	N/A
#6 Mass Assignment	N/A
#7 Security Misconfiguration	✓
#8 Injection	✓
#9 Improper Assets Management	✓
#10 Insufficient Logging	✓



Remediations

Note: For more details on category-wise remedial suggestions, please visit the project's coverage page.

Unsecured

The following techniques may be utilized for having Secured Endpoints ⁽³⁾⁽⁵⁾⁽⁶⁾.

- Session Management and Authentication
- API Keys
- OpenID Connect, OAuth2, and SAML
- Access Controls
- Rate Limits
- Input Validation and HTTP Return Codes

ABAC_Level1

The following techniques may be checked for ensuring RBAC is in place ⁽²⁾⁽³⁾⁽⁴⁾.

- Implement a proper authorization mechanism that relies on the user policies and hierarchy.
- Prefer not to use an ID that has been sent from the client, but instead use an ID that is stored in the session object when accessing a database record by the record ID.
- Use an authorization mechanism to check if the logged-in user has access to perform the requested action on the record in every function that uses an client input to access a record in the database.
- Prefer to use random and unpredictable values as GUIDs for records' IDs.
- Write tests to evaluate the authorization mechanism. Do not deploy vulnerable changes that break the tests.

ADoS

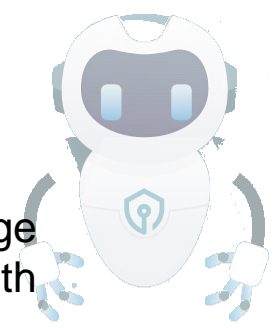
While it may not be completely possible to avoid an ADoS attack, but it is possible to identify and take remediation steps. Different protections that can be leveraged are

- Network Controls – Allowing for blacklisting of IP Addresses and CIDR Ranges
- Rate Controls – Different threshold criterion can be defined to avoid volumetric flooding
- Site Defenders – to identify Slow Posts that open HTTP connections and then slowing sending data very slowly.
- Web Application Firewalls – Many DoS tools have tell-tale fingerprints and can be easily identified and blocked.

Http_Authentication

The following may be employed to implement proper authentication scheme.

- HTTPS/TLS should be used with basic authentication.
- Classify data processed, stored or transmitted by an application. Identify which data is sensitive according to privacy laws, regulatory requirements, or business needs.
- Apply controls as per the classification.
- Encrypt all data in transit with secure protocols such as TLS with perfect forward secrecy (PFS) ciphers, cipher prioritization by the server, and secure parameters. Enforce encryption using directives like HTTP Strict Transport Security (HSTS).



About APIsec Inc.

APIsec is build to address fundamental security challenge
 - APIs are breached on a scale never seen before with web and mobile applications.

Attackers abuse business logic flaws and loopholes in APIs to expose and exploit the sensitive data of millions of people across the globe every year.

APIsec addresses the critical need to secure APIs before they reach production, providing the industry's only automated and continuous API security testing platform.

APIsec offers



Continuous Security

Continuous testing that keeps up with Development



Automated Testing

Automated test creation ensures APIs are fully examined



Complete Coverage

Tests every endpoint and method against OWASP risks



Speed

Executes complete API test suites in minutes

BUSINESS VALUE



Compliance
mandate



Secure Releases



Avoid Manual Security
Penetration Effort