Full Audit Report

Dogemiyagi Security Assessment

Real Cybersecurity
Protecting digital assets

Made in Thailand

SECURI LAB
(THAILAND) contact@securi-lab.com

Audit Report

FULL AUDIT REPORT

Table of Contents

FULL AUDIT REPORT

Report Information

About Report	**Dogemiyagi Security Assessment**
Version	**v1.1**
Client	**Dogemiyagi**
Language	**Solidity**
Confidentiality	**Public**
Contract Address	**0x6BE53e740DF3d2b93984Ec23b1B2AFf0Ed79a182**
Audit Method	**Whitebox**
Security Assessment Author	**Auditor** **Mark K. [Security Researcher | Redteam]** **Kevin N. [Security Researcher | Web3 Dev]** **Yusheng T. [Security Researcher | Incident Response]** **Approve Document** **Ronny C. CTO & Head of Security Researcher** **Chinnakit J. CEO & Founder**

*Audit Method

Whitebox: SECURI LAB Team receives all source code from the client to provide the assessment.
Blackbox: SECURI LAB Team receives only bytecode from the client to provide the assessment.

Digital Sign (Only Full Audit Report)

Digitally signed by Chinnakit J.
DN: cn=Chinnakit J., o=SECURI LAB (Thailand), ou=CEO & Founder, email=contact@securi-lab.com, c=TH
Date: 2023.04.15 20:03:15 +07'00'

Disclaimer

Regarding this security assessment, there are no guarantees about the security of the program instruction received from the client is hereinafter referred to as "**Source code**".
And **SECURI Lab** hereinafter referred to as "**Service Provider**", the **Service Provider** will not be held liable for any legal liability arising from errors in the security assessment. The responsibility will be the responsibility of the **Client**, hereinafter referred to as "**Service User**" and the
Service User agrees not to be held liable to the **service provider** in any case. By contract **Service Provider** to conduct security assessments with integrity with professional ethics, and transparency to deliver security assessments to users The **Service Provider** has the right to postpone the delivery of the security assessment. If the security assessment is delayed whether caused by any reason and is not responsible for any delayed security assessments.
If **the service provider** finds a vulnerability The **service provider** will notify the **service user** via the Preliminary Report, which will be kept confidential for security. The **service provider** disclaims responsibility in the event of any attacks occurring whether before conducting a security assessment. Or happened later All responsibility shall be sole with the **service user**.

Security Assessment Not Financial/Investment Advice Any loss arising from any investment in any project is the responsibility of the investor.

SECURI LAB disclaims any liability incurred. Whether it's Rugpull, Abandonment, Soft Rugpull

The SECURI LAB team has conducted a comprehensive security assessment of the vulnerabilities.
This assessment is tested with an expert assessment. Using the following test requirements

1. Smart Contract Testing with Expert Analysis By testing the most common and uncommon vulnerabilities.
2. Automated program testing It includes a sample vulnerability test and a sample of the potential vulnerabilities being used for the most frequent attacks.
3. Manual Testing with AST/WAS/ASE/SMT and reviewed code line by line
4. Visibility, Mutability, Modifier function testing, such as whether a function can be seen in general, or whether a function can be changed and if so, who can change it.
5. Function association test It will be displayed through the association graph.
6. This safety assessment is cross-checked prior to the delivery of the assessment results.

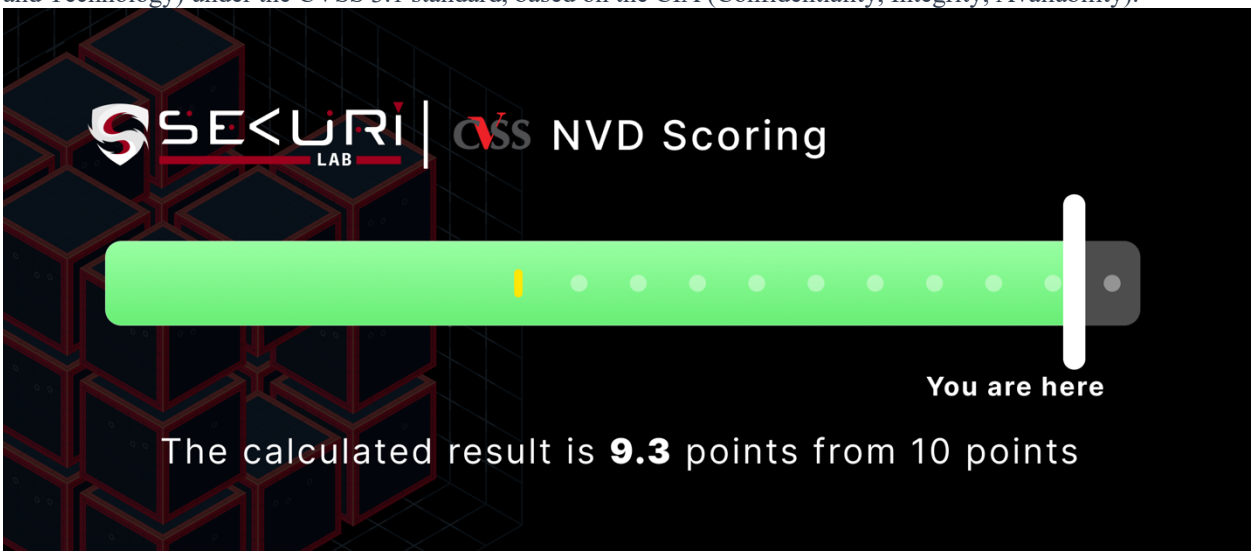
FULL AUDIT REPORT

Executive Summary

For this security assessment, SECURI LAB received a request from Dogemiyagi on Tuesday, April 11, 2023.

NVD CVSS Scoring

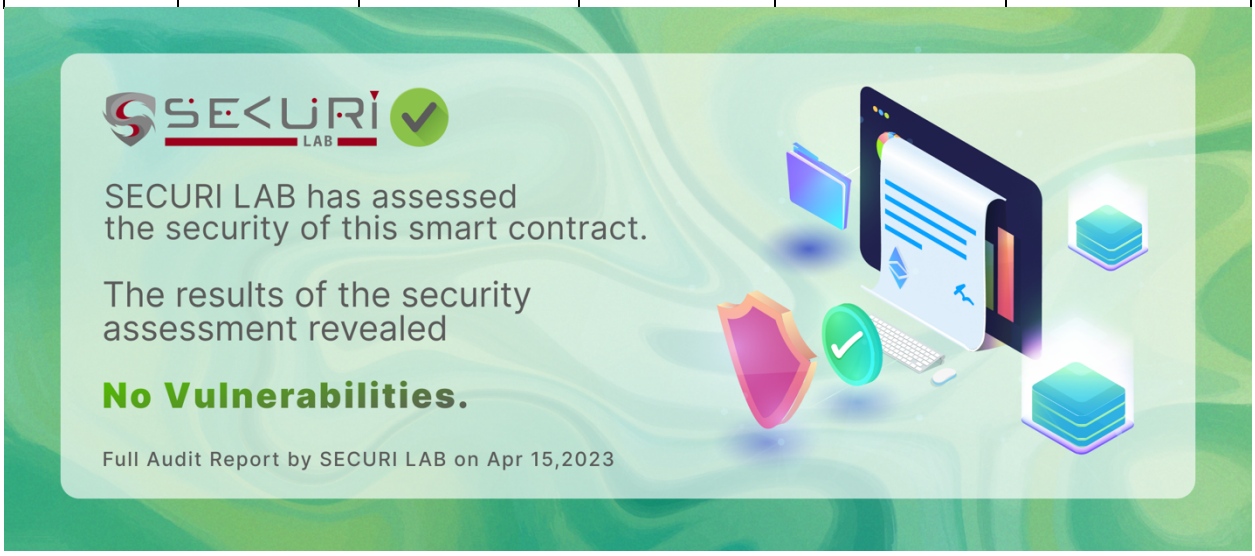
The score was calculated using the NVD (National Vulnerability Database) of NIST (National Institute of Standards and Technology) under the CVSS 3.1 standard, based on the CIA (Confidentiality, Integrity, Availability).



Audit Result

SECURI LAB evaluated the smart contract security of the project and found: [Total : 1]

Critical	High	Medium	Low	Very Low	Informational
0	0	0	0	0	1



FULL AUDIT REPORT

Project Introduction

Scope Information:

Project Name	**Dogemiyagi**
Website	**https://www.dogemiyagi.com/**
Chain	**Ethereum**
Language	**Solidity**

Audit Information:

Request Date	**Tuesday, April 11, 2023**
Audit Date	**Thursday, April 13, 2023**
Re-assessment Date	**Saturday, April 15, 2023**

Audit Version History:

Version	Date	Description
1.0	**Thursday, April 13, 2023**	**Preliminary Report**
1.1	**Saturday, April 15, 2023**	**Full Audit Report**

FULL AUDIT REPORT

Initial Audit Scope:

Smart Contract Address	0x6BE53e740DF3d2b93984Ec23b1B2AFf0Ed79a182
Compiler Version	v0.8.18

Type	File	Logic Contracts	Interfaces	Lines	nLines	nSLOC	Comment Lines	Complex. Score	Capabilities
📝	contracts/MIYAGI.sol	1		20	20	18	1	10	
📝	**Totals**	**1**		**20**	**20**	**18**	**1**	**10**	

- **Lines**: total lines of the source unit
- **nLines**: normalized lines of the source unit (e.g. normalizes functions spanning multiple lines)
- **nSLOC**: normalized source lines of code (only source-code lines; no comments, no blank lines)
- **Comment Lines**: lines containing single or block comments
- **Complexity Score**: a custom complexity score derived from code statements that are known to introduce code complexity (branches, loops, calls, external interfaces, ...)

FULL AUDIT REPORT

Initial Reassessment Audit Scope:

Smart Contract Address	0x6BE53e740DF3d2b93984Ec23b1B2AFf0Ed79a182
Compiler Version	v0.8.18

Type	File	Logic Contracts	Interfaces	Lines	nLines	nSLOC	Comment Lines	Complex. Score	Capabilities
	contracts/MIYAGI.sol	1		20	20	18	1	10	
	Totals	**1**		**20**	**20**	**18**	**1**	**10**	

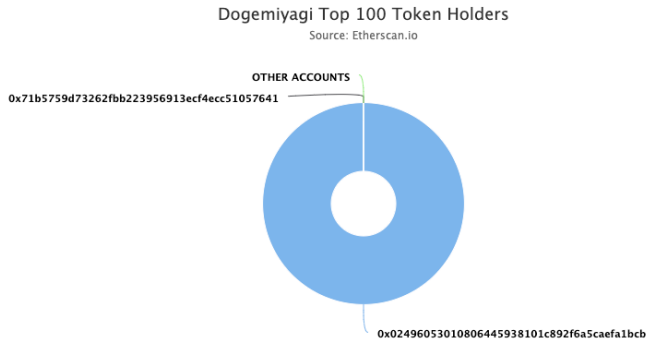
- **Lines**: total lines of the source unit
- **nLines**: normalized lines of the source unit (e.g. normalizes functions spanning multiple lines)
- **nSLOC**: normalized source lines of code (only source-code lines; no comments, no blank lines)
- **Comment Lines**: lines containing single or block comments
- **Complexity Score**: a custom complexity score derived from code statements that are known to introduce code complexity (branches, loops, calls, external interfaces, ...)

FULL AUDIT REPORT

TOKEN ANALYSIS

The top 100 holders collectively own 100.00% (96,192,515,112,011.00 Tokens) of Dogemiyagi

Token Total Supply: 96,192,515,112,011.00 Token | Total Token Holders: 2



(A total of 96,192,515,112,011.00 tokens held by the top 100 accounts from the total supply of 96,192,515,112,011.00 token)

Rank	Address	Quantity (Token)	Percentage
1	0x024960...AEfA1BCb	96,191,563,230,811	99.9990%
2	PinkLock02	951,881,200	0.0010%

Data on date Saturday, April 15, 2023 19:50PM UTC+7

Team Wallet Lock (2 Years)(Mon Apr 14 2025 07:00:00 UTC+0700)
Transaction Hash:0x0a70d5866ecf1662a661e321a4658200fa7adc8cc8eb1078345f77c6c514c5a2
Wallet : 0x9F9c3aD913fA6D8A8392c6B352d15eB3055afdc5
https://www.pinksale.finance/pinklock/record/1004455?chain=ETH

#	Name	Type	Data
0	owner	address	0x9F9c3aD913fA6D8A8392c6B352d15eB3055afdc5
1	token	address	0x6BE53e740DF3d2b93984Ec23b1B2AFf0Ed79a182
2	isLpToken	bool	false
3	amount	uint256	475940600000000000000000000
4	unlockDate	uint256	1744588800
5	description	string	{"l":"Dogemiyagi Team Wallet Lock"}

Marketing Wallet Lock (1 Year + Vesting %10 Every Month)(Tue May 14 2024 07:00:00 UTC+0700)
Transaction Hash: 0x1eb052cfdf2d6d3c6b4da9277ac394754239fcf5ba97121832bcb74d489f9040
Wallet : 0x69b7D438Eb83931F43f35A1C00AA77421C71D5b0
https://www.pinksale.finance/pinklock/record/1004456?chain=ETH

#	Name	Type	Data
0	owner	address	0x69b7D438Eb83931F43f35A1C00AA77421C71D5b0
1	token	address	0x6BE53e740DF3d2b93984Ec23b1B2AFf0Ed79a182
2	isLpToken	bool	false
3	amount	uint256	475940600000000000000000000
4	tgeDate	uint256	1715644800
5	tgeBps	uint256	1000
6	cycle	uint256	2592000
7	cycleBps	uint256	1000
8	description	string	{"l":"Dogemiyagi Marketing Wallet Lock"}

Security Assessment Procedure

Securi has the following procedures and regulations for conducting security assessments:

1.Request Audit Client submits a form request through the Securi channel. After receiving the request, Securi will discuss a security assessment. And drafting a contract and agreeing to sign a contract together with the Client

2.Auditing Securi performs security assessments of smart contracts obtained through automated analysis and expert manual audits.

3.Preliminary Report At this stage, Securi will deliver an initial security assessment. To report on vulnerabilities and errors found under Audit Scope will not publish preliminary reports for safety.

4.Reassessment After Securi has delivered the Preliminary Report to the Client, Securi will track the status of the vulnerability or error, which will be published to the Final Report at a later date with the following statuses:

- **a.Acknowledge** The client has been informed about errors or vulnerabilities from the security assessment.
- **b.Resolved** The client has resolved the error or vulnerability. Resolved is probably just a commit, and Securi is unable to verify that the resolved has been implemented or not.
- **c.Decline** Client has rejected the results of the security assessment on the issue.

5.Final Report Securi providing full security assessment report and public

Risk Rating

Risk rating using this commonly defined: $Risk\ rating\ =\ impact\ *\ confidence$

Impact The severity and potential impact of an attacker attack

Confidence Ensuring that attackers expose and use this vulnerability

Both have a total of 3 levels: **High**, **Medium**, **Low**. By *Informational* will not be classified as a level

Confidence / Impact [Likelihood]	Low	Medium	High
Low	**Very Low**	**Low**	**Medium**
Medium	**Low**	**Medium**	**High**
High	**Medium**	**High**	**Critical**

FULL AUDIT REPORT

Vulnerability Severity Summary

Severity is a risk assessment It is calculated from the Impact and Confidence values using the following calculation methods,

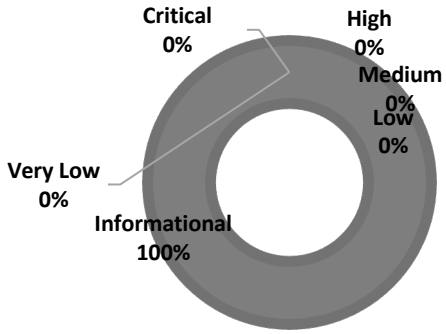
$Risk\ rating\ =\ impact\ *\ confidence$

It is categorized into

5 categories based on the **lowest severity**:

Very Low , Low , Medium , **High** , Critical .

For **Informational** & will **Non-class/Optimization/Best-practices will** not be counted as **severity**



Vulnerability Severity Level	Total
Critical	**0**
High	**0**
Medium	**0**
Low	**0**
Very Low	**0**
Informational	**1**
Non-class/Optimization/Best-practices	**0**

Category information:

Centralization

Centralization Risk is The risk incurred by a sole proprietor, such as the Owner being able to change something without permission

Economics Risk

Economics Risk is Risks that may affect the economic mechanism system, such as the ability to increase Mint token

Logical Issue

Logical Issue is that can cause errors to core processing, such as any prior operations that cause background processes to crash.

Authorization

Authorization is Possible pitfalls from weak coding allows unrelated people to take any action to modify the values.

Mathematical

Mathematical Any erroneous arithmetic operations affect the operation of the system or lead to erroneous values.

Naming Conventions

Naming Conventions naming variables that may affect code understanding or naming inconsistencies

Security Risk

Security Risk of loss or damage if it's no mitigate

Coding Style

Coding Style is Tips coding for efficiency performance

Best Practices

Best Practices is suggestions for improvement

Optimization

Optimization is performance improvement

Gas Optimization

Gas Optimization is increase performance to avoid expensive gas

Dead Code

Dead Code having unused code This may result in wasted resources and gas fees.

FULL AUDIT REPORT

Vulnerability Findings

ID	Vulnerability Detail	Severity	Category	Status
SEC-01	If different pragma directives are used (pragma)	Informational	Best Practices	**Acknowledge**

SEC-01: If different pragma directives are used (pragma)

Vulnerability Detail	Severity	Location	Category	Status
If different pragma directives are used (pragma)	Informational	Check on finding	Best Practices	**Acknowledge**

Finding:

```
❌ Different versions of Solidity are used:
    • Version used: ['0.8.18', '^0.8.0']
    • 0.8.18 (MIYAGI.sol:2)
    • ^0.8.0 (../../../node_modules/@openzeppelin/contracts/utils/Context.sol#4)
    • ^0.8.0
(../../../node_modules/@openzeppelin/contracts/token/ERC20/extensions/IERC20Metadata.s
ol#4)
    • ^0.8.0 (../../../node_modules/@openzeppelin/contracts/access/Ownable.sol#4)
    • ^0.8.0 (../../../node_modules/@openzeppelin/contracts/token/ERC20/ERC20.sol#4)
    • ^0.8.0 (../../../node_modules/@openzeppelin/contracts/token/ERC20/IERC20.sol#4)
```

Scenario:

-

Recommendation:

Recommendation: Use one Solidity version.

Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#different-pragma-directives-are-used

Alleviation:

Dogemiyagi Team has Acknowledge this issue.

SWC Findings

ID	Title	Scanning	Result
SWC-100	Function Default Visibility	Complete	No risk
SWC-101	Integer Overflow and Underflow	Complete	No risk
SWC-102	Outdated Compiler Version	Complete	No risk
SWC-103	Floating Pragma	Complete	No risk
SWC-104	Unchecked Call Return Value	Complete	No risk
SWC-105	Unprotected Ether Withdrawal	Complete	No risk
SWC-106	Unprotected SELFDESTRUCT Instruction	Complete	No risk
SWC-107	Reentrancy	Complete	No risk
SWC-108	State Variable Default Visibility	Complete	No risk
SWC-109	Uninitialized Storage Pointer	Complete	No risk
SWC-110	Assert Violation	Complete	No risk
SWC-111	Use of Deprecated Solidity Functions	Complete	No risk
SWC-112	Delegatecall to Untrusted Callee	Complete	No risk
SWC-113	DoS with Failed Call	Complete	No risk
SWC-114	Transaction Order Dependence	Complete	No risk
SWC-115	Authorization through tx.origin	Complete	No risk
SWC-116	Block values as a proxy for time	Complete	No risk

SWC-117	Signature Malleability	Complete	No risk
SWC-118	Incorrect Constructor Name	Complete	No risk
SWC-119	Shadowing State Variables	Complete	No risk
SWC-120	Weak Sources of Randomness from Chain Attributes	Complete	No risk
SWC-121	Missing Protection against Signature Replay Attacks	Complete	No risk
SWC-122	Lack of Proper Signature Verification	Complete	No risk
SWC-123	Requirement Violation	Complete	No risk
SWC-124	Write to Arbitrary Storage Location	Complete	No risk
SWC-125	Incorrect Inheritance Order	Complete	No risk
SWC-126	Insufficient Gas Griefing	Complete	No risk
SWC-127	Arbitrary Jump with Function Type Variable	Complete	No risk
SWC-128	DoS With Block Gas Limit	Complete	No risk
SWC-129	Typographical Error	Complete	No risk
SWC-130	Right-To-Left-Override control character (U+202E)	Complete	No risk
SWC-131	Presence of unused variables	Complete	No risk
SWC-132	Unexpected Ether balance	Complete	No risk
SWC-133	Hash Collisions With Multiple Variable Length Arguments	Complete	No risk

SWC-134	Message call with hardcoded gas amount	Complete	No risk
SWC-135	Code With No Effects	Complete	No risk
SWC-136	Unencrypted Private Data On-Chain	Complete	No risk

FULL AUDIT REPORT

Visibility, Mutability, Modifier function testing

Components

Contracts	Libraries	Interfaces	Abstract
1	0	0	0

Exposed Functions

This section lists functions that are explicitly declared public or payable. Please note that getter methods for public stateVars are not included.

Public	Payable
1	0

External	Internal	Private	Pure	View
0	1	0	0	1

StateVariables

Total	Public
1	0

Capabilities

Solidity Versions observed	Experimental Features	Can Receive Funds	Uses Assembly	Has Destroyable Contracts
0.8.18				

Transfers ETH	Low-Level Calls	DelegateCall	Uses Hash Functions	ECRecover	New/Create/Create2

TryCatch	Σ Unchecked

Dependencies / External Imports

Dependency / Import Path	Count
@openzeppelin/contracts/access/Ownable.sol	1
@openzeppelin/contracts/token/ERC20/ERC20.sol	1

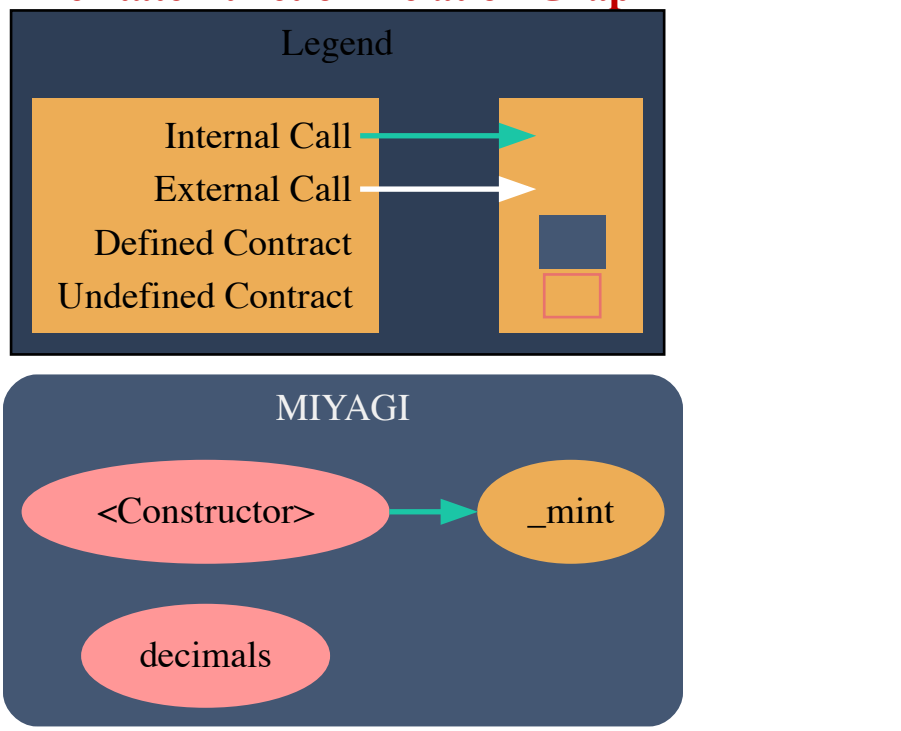
Contracts Description Table

Contract	Type	Bases		
L	**Function Name**	**Visibility**	**Mutability**	**Modifiers**
MIYAGI	Implementation	ERC20, Ownable		
L		Public !	🛑	ERC20
L	decimals	Public !		NO !

Legend

Symbol	Meaning
🛑	Function can modify state
💵	Function is payable

Inheritate Function Relation Graph



UML Class Diagram

MIYAGI contracts/MIYAGI.sol
Private: _decimals: uint8
Public: constructor(_name: string, _symbol: string, __decimals: uint8, _totalSupply: uint256) decimals(): uint8

About SECURI LAB

Enhance the security and legitimacy of your blockchain project with our professional Audit & KYC services. Our experienced team provides reliable, cost-effective, and secure verification processes.

Follow Us On:

Website	https://securi-lab.com/
Twitter	https://twitter.com/SECURI_LAB
Telegram	https://t.me/securi_lab
Medium	https://medium.com/@securi