


# LOCKER SECRETS MANAGER

LOCKER SECRETS MANAGER  
DESIGNED FOR THE NEEDS OF DEVELOPERS

**Locker API** 

Secrets: **16**

Environment: **99+**

Created at: 22/12/2022

- < Back
- ✓ Secrets
- 🌐 Environments
- 🛡️ Detection
- 🔍 Access Key
- 📄 Access Log
- ⚙️ Settings











Projects / Secrets

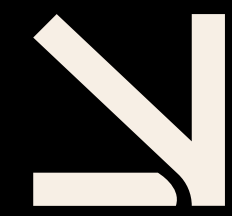
### Secrets

Secrets refer to sensitive information such as passwords, API keys, database credentials, and any other confidential data that is used to access and protect various resources in your application.

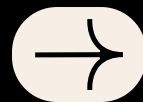
[How can I add secrets to my projects? →](#)

Key:  Environment:

Key	Value
 AWS_S3_REGION_NAME	.....
 AWS_S3_BUCKET	.....
 AWS_S3_ACCESS_KEY	.....
 AWS_S3_SECRET_KEY	.....
 MYSQL_USERNAME	.....
 MYSQL_PASSWORD	.....
 MYSQL_DATABASE	.....
 GITHUBTH_CLIENT_ID	.....
 GITHUB_AUTH_SECRET_KEY	.....
 GOOGLE_CREDENTIAL	.....



# Table of Content



- |    |   |    |                  |
|----|---|----|------------------|
| 01 | Common challenges in managing secrets   | 03 | Common use cases |
| 02 | Introduction of Locker Secrets Manager (Locker SM) <ul style="list-style-type: none"><li>2.1 How Locker SM works</li><li>2.2 Advantages of Locker SM</li><li>2.3 Functions of Locker SM</li><li>2.4 Locker SM core values are Security and Transparency</li><li>2.5 Availability of Locker SM</li></ul> | 04 | About CyStack    |
|    |   | 05 | Contact          |



# Common challenges

## WHAT IS SECRETS?

---

In programming, "secrets" refers to sensitive information such as passwords, API keys, access tokens, or any data that an application or system needs for authorization or user authentication. Safeguarding secrets is important to avoid unauthorized access or disclosure of sensitive information. Various tools and methods are often used to store, manage, and access secrets safely and securely.



LOCKER SECRETS MANAGER

LOCKER SECRETS MANAGER

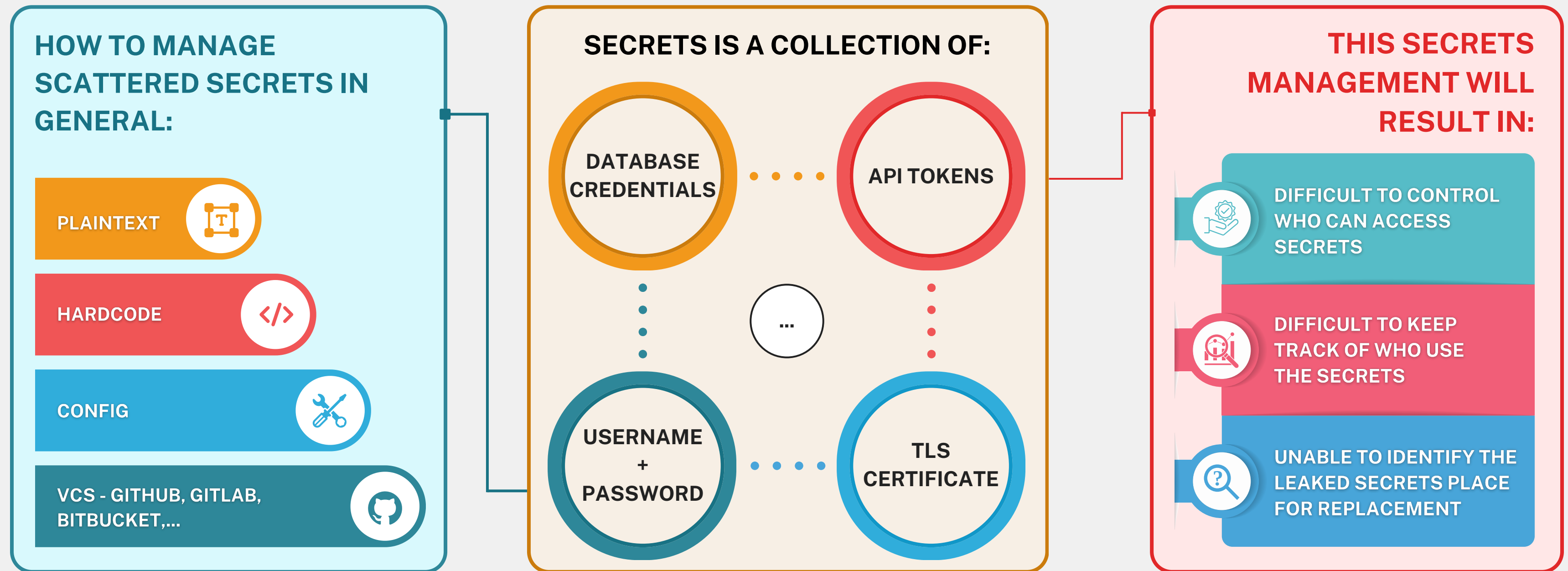
LOCKER SECRETS MANAGER

LOCKER SECRETS MANAGER

LOCKER SECRETS MANAGER



# Examples of secrets and challenges encountered in storing secrets:





# Summary of secrets management challenges

**01**

Keep the secrets safe

**02**

Store secrets securely

**03**

Must change key -  
lock regularly

**04**

Control secrets' viewers

**05**

Track secrets' users

**06**

Integrate secrets with  
existing systems  
automatically

**07**

Share secrets securely

**08**

Rotate secrets  
automatically

**09**

Raise awareness among  
developers about how  
to secure secrets

**10**

Don't rely too much  
on one solution



# Introduction of Locker Secrets Manager

Locker Secrets Manager streamlines the storage and management of SSH keys, API tokens, and other infrastructure secrets throughout the software development lifecycle. It is delivered as a command line tool and programming SDKs integrated into developer source code.








+ New Secret



Key	Environment
Please select	Please select

Key	Value	Environment
 0xdFD2vka9aDk93zl3S2hf618hk2362...	.....	<input checked="" type="radio"/> Staging <input type="radio"/> Production
 1HjTd195vfs5ha6JdS4hc73lbwt74FTs...	.....	<input checked="" type="radio"/> Staging <input type="radio"/> Production
 1dx5naJF9Bswq3xbGjrcD6gsFg2574v...	.....	<input checked="" type="radio"/> Staging
 0siwJeiv259vR4ka539bnSEb2bisvaw...	.....	<input checked="" type="radio"/> All
 v49fsjRSberiE3tGjfs483ivlapqVDbst3...	.....	<input type="radio"/> Production



LOCKER - A PRODUCT OF CYSTACK



# How Locker SM works



**01**

Locker stores secrets in a centralized data storage called secrets vault.



**02**

Data in Locker is kept completely secure using end-to-end encryption, zero-knowledge encryption, and a zero-trust model.

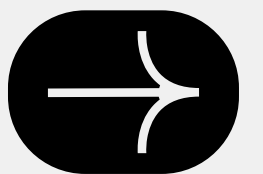
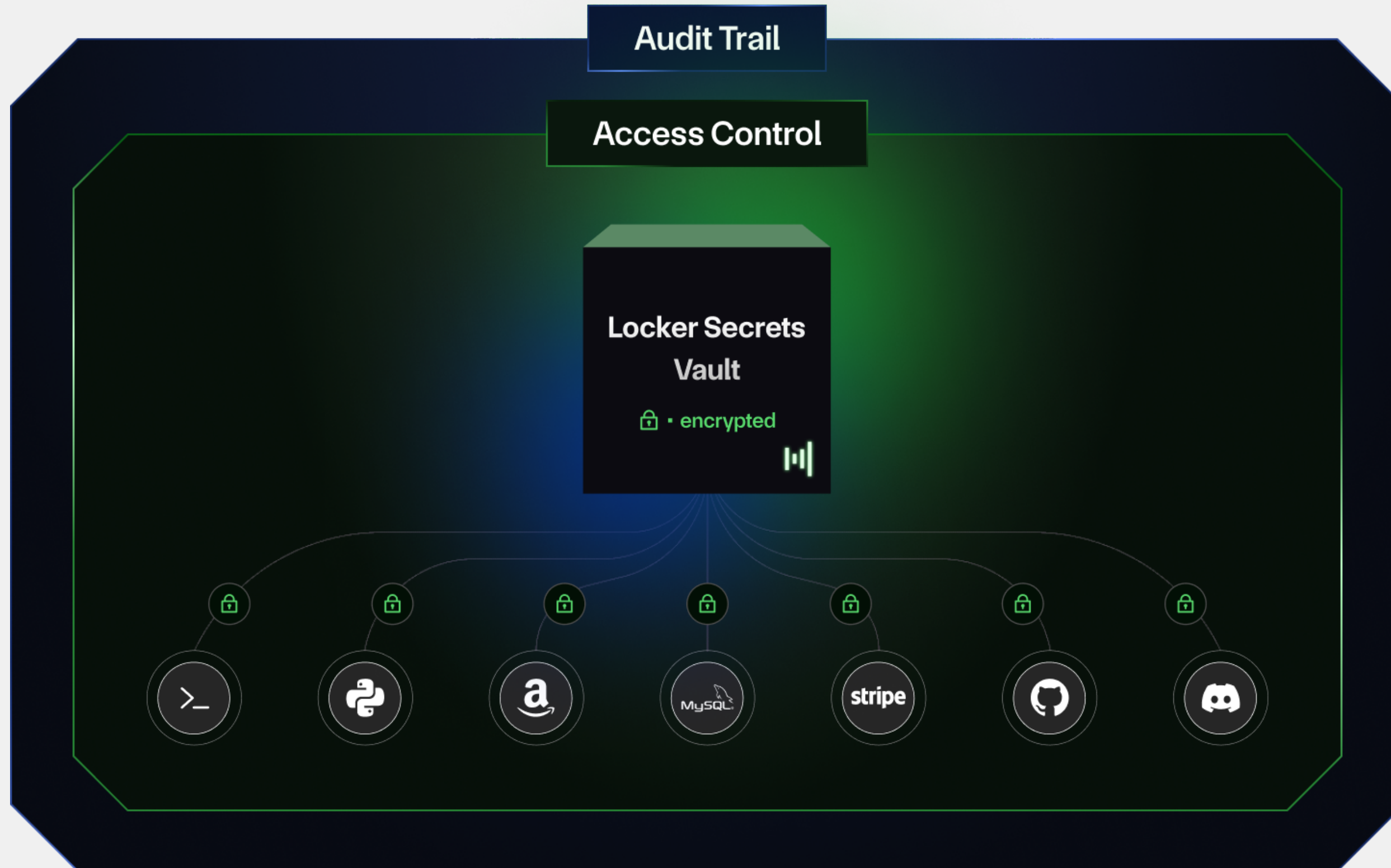


**03**

Third-party applications want to access secrets stored in Locker, or vice versa, Locker transmits secrets to third-party applications to grant permissions, the transmission is absolutely encrypted.

- *End-to-end encryption: ensures that data is securely encrypted and decrypted only on the end-user's device.*
- *Zero-knowledge encryption: ensures that only the user has the encryption key for their safe.*
- *Zero trust model: ensures every activity and user is thoroughly reviewed and authenticated.*

# Diagram of how Locker SM works







# Advantages of Locker SM



## 01 - ENHANCED SECURITY

By using Locker, sensitive information such as passwords, API keys, access tokens are stored and managed more securely. This minimizes the risk of unauthorized access and information disclosure, helping to protect your data.

## 04 - STRICT ACCESS CONTROL

Locker provides sophisticated access control mechanisms, allowing you to clearly specify who has access and use of secrets. This helps prevent unnecessary access and enhances system security.

## 02 - SIMPLIFIED SECRETS MANAGEMENT

Secrets are centralized within Locker, streamlining the management and tracking process conveniently. You can easily add, modify, or delete secrets without manual updates across multiple systems.

## 05 - ACTIVITY TRACKING AND AUDITING

Locker offers an audit trail function, allowing you to record and track activities related to secrets. You can easily check and verify the usage of these secrets.

## 03 - ROBUST ENCRYPTION

Data in Locker is strongly encrypted, protecting them from unauthorized access. Even in the event of an intrusion attack, sensitive information remains protected.

## 06 - FLEXIBLE INTEGRATION

Locker seamlessly integrates with popular tools and services in the software development process such as AWS, Azure, and other cloud services. This makes it easy to integrate and use secrets in your applications and systems without major friction.

# Functions of Locker SM

**01**

Store and manage encrypted secrets

**02**

Classify secrets according to development environment: dev, staging, production...

**03**

SDK support: with popular programming languages such as Python, Go, JS, C#

**04**

CLI support: convenient for CI/CD and DevSecOps processes

**05**

Control sharing and access to secrets for members at each level

**06**

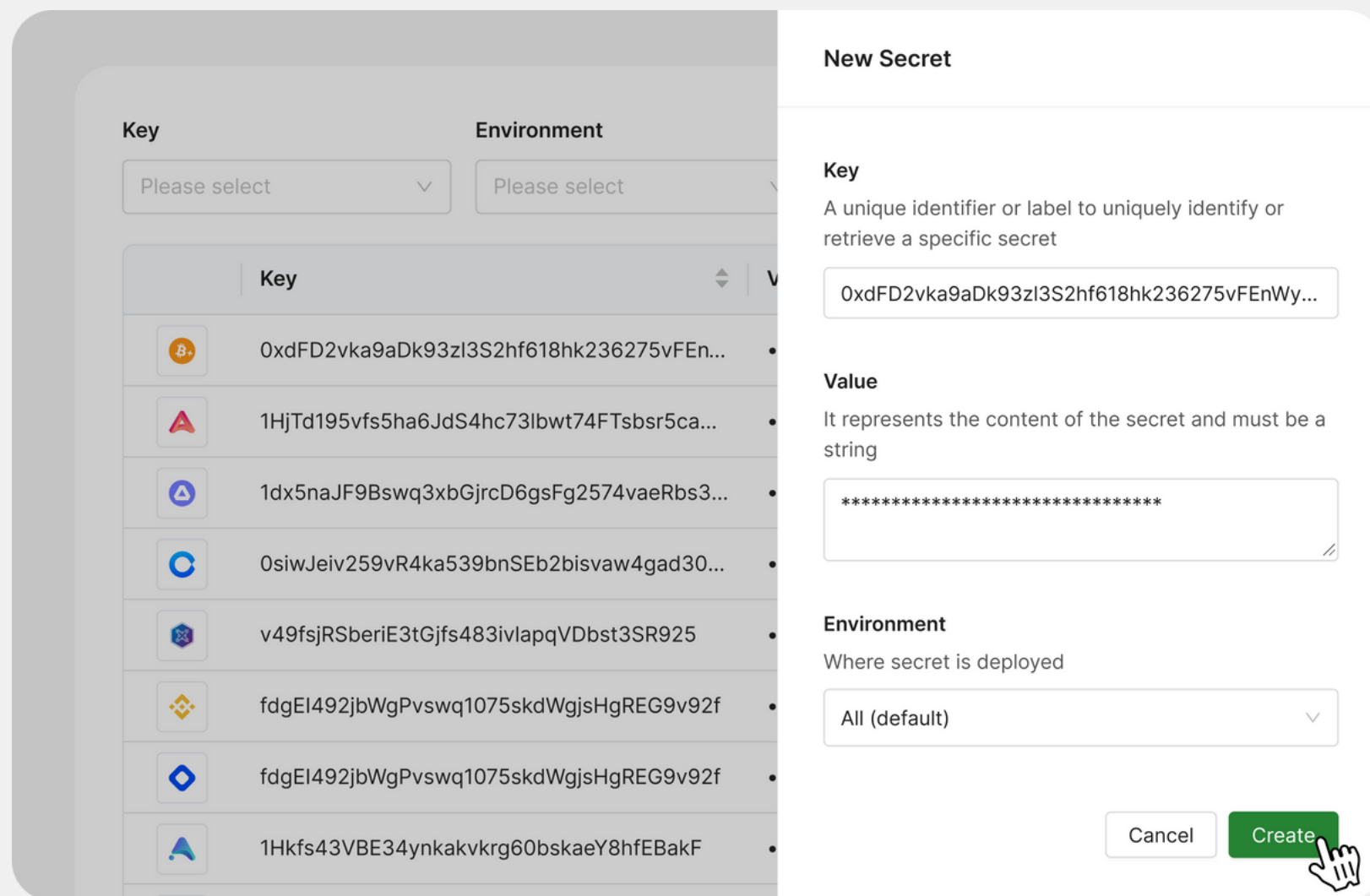
Check access logs by time, location, and user's IP address

**07**

Automatically detect secrets used insecurely in source code, automatically replace with Locker SDK, and delete Git history.

**08**

Automatically rotate the secret key periodically





SECURITY & TRANSPARENCY

# Locker SM core values are Security and Transparency



## ADVANCED ENCRYPTION

Locker employs robust **end-to-end** and **zero-knowledge encryption** to guarantee exclusive access to your vault. Your data is encrypted and decrypted on the fly on your device, ensuring that neither Locker nor anyone else can access your information.



## OPEN SOURCE CODE

Going open-source is our way of proving our commitment to delivering a transparent and reliable product. By publishing our source code, we ensure anyone can view, report bugs, and contribute changes to make Locker even better.



## INDEPENDENT TESTING

Locker undergoes regular independent audits conducted by the WhiteHub bug bounty platform and security experts from CyStack, enabling us to detect and address potential vulnerabilities while enhancing our security protocols and procedures.

# Availability of Locker SM

Flexible and adaptable through support for popular programming languages

---



<https://locker.io/>



SECURITY & TRANSPARENCY

SECURITY & TRANSPARENCY

SECURITY & TRANSPARENCY

SECURITY & TRANSPARENCY

SECURITY & TRANSPARENCY



# Common Use cases

## STORE SENSITIVE VARIABLES

### 01 - CHALLENGE

Eliminate hard coded credentials, keys, and tokens across disparate clouds and environments.

### 02 - PROBLEM

Hardcode, storing raw secrets in config files or environment variables makes it possible for anyone with access to the source code, files or environment variables to read and extract secrets. If the source code is leaked or compromised, secrets can easily be extracted and used for malicious purposes, causing damage to the system.



### 03 - SOLUTION

- Use Locker instead of environment variables to easily control secrets, even during program operations.
- Locker applies **end-to-end encryption** and **zero-knowledge technology** for storage, ensuring that secrets are absolutely confidential and that no one else can access the user's data warehouse except them.
- Allows users to organize groups of secrets into projects and environments for systematic management and access.
- Allows administrators to manage all secrets in the system. As the system increases in scale, centralized management of secrets helps this important data to be distributed, updated and synchronized safely and effectively.

# Common Use cases

## ➔ DATA ENCRYPTION

### 01 - CHALLENGE

Managing encryption keys at scale is difficult and time-consuming.

### 02 - PROBLEM

Many cloud providers offer key management services (KMS), where encryption keys can be issued and stored to maintain a source of truth. However, this often leads to manual lifecycle management when you want to use your own keys.



### 03 - SOLUTION

- Locker can be used as a key management service (KMS) to create and manage encryption keys such as AES, RSA, DES, XChaCha20... for data encryption.
- Centrally manage and automatically encrypt encryption keys across different environments.



# Common Use cases

## CRYPTO WALLET PRIVATE KEY STORAGE

### 01 - CHALLENGE

Ensure safety and high security but still be easily accessible to and convenient in use.

### 02 - PROBLEM

- Traditionally, storing private keys in crypto wallets often faces the risk of being attacked by hackers or malware, especially when stored on internet-connected devices.
- Managing and protecting private keys can become complex and require a lot of work.



### 03 - SOLUTION

- Locker is a secure place to store sensitive data such as the private key of a crypto wallet and can be retrieved through Locker's built-in tool.
- End-to-end encryption: Apply end-to-end encryption to protect data during transmission and storage, ensuring that data can only be decrypted by authorized personnel.

# Common Use cases

## ➔ LOCKER SM WORKS AS A KEY-VALUE DATABASE

### 01 - DEFINITION

A key-value database is a type of unstructured database in which data is stored and retrieved based on a corresponding key - value pair. Each key is unique and associated with a specific value.

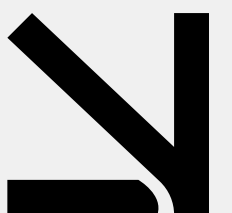
### 02 - WHY DO WE NEED TO USE A KEY-VALUE DATABASE INSTEAD OF USING A RELATIONAL DATABASE?

The primary advantage of using a key-value database is the rapid retrieval of information. Traditional databases often exhibit slower information retrieval speeds.



### 03 - USE LOCKER SM AS A KEY-VALUE DATABASE

For confidential data requiring swift access, such as system access keys, using Locker SM as a key-value database will ensure both security and speed.





# About CyStack

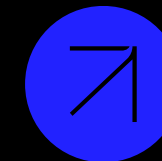
---

CyStack

CyStack is a leading cybersecurity company in Vietnam, known for its in-depth research capabilities and building pioneering security solutions.

Our products and services focus on vulnerability management, threat detection, community security, and data security. Locker is a product developed by CyStack.

TRUSTED BY  
LEADING BUSINESS




**CAKE**  
by VPBank

*Sendo*

**ACB**


**mo  
mo**

 **mitsubishi.com**

**vntrip.vn**

 **AGRIBANK**

 **OpenCommerce**

 **One Mount**



# Contact

LOCKER  
SECRETS  
MANAGER



## Locker Secrets Manager

Cloud version



## Locker Secrets Manager

Self-hosted version (self-deployed & managed)

### → PHONE

02471099656

### → ADDRESS

Tan Hong Ha Complex Building,  
317 Truong Chinh, Nga Tu So,  
Thanh Xuan, Hanoi

### → WEBSITE

<https://locker.io/>

### → EMAIL

[contact@locker.io](mailto:contact@locker.io)



# Thank you

LOCKER  
SECRETS  
MANAGER



2024

*LOCKER.IO*

WHERE SECRETS  
STAY SAFE!

