



GitHub

機密保護培訓





SESSION 學習目標

準備好將 GitHub 機密保護 付諸實踐

- ✓ 設置和管理 GitHub 高級安全配置
- ✓ 瞭解如何啟用和使用機密保護功能
- ✓ 在源頭阻止密鑰洩露，採用推送保護機制
- ✓ 具備自主修復能力

我們的議程



安全現狀

全球應用安全的現狀
如何



什麼是 GHAS?

GHAS 在其中扮演什麼角
色



使能

啟動秘密保護功能



秘密保護

深入解析所有 GitHub
Secret Protection 產品
和功能



第三方集成

數據匯出與同步



回顧

我們今天所學總結





GitHub 高級安全

安全現狀



DevSecOps 現狀



安全風險

應用程式是首要攻擊載體

80%的入侵事件是通過網頁應用漏洞攻擊實現的

停滯的進展

3個月後仍有65%的漏洞存在

只有33%的洩露是由組織的團隊或工具發現的

增幅器

組織認為人工智慧提供了更高的投資回報率

84%的高管計劃優先考慮生成式人工智慧網路安全解決方案，而非傳統網路安全解決方案

來源: [Verizon數據洩露調查報告2023](#)

來源: [Veracode 2023年安全現狀報告](#)

來源: [IBM CEO生成式人工智慧指南，2023年](#)



展望未來...

我們的攻擊面 正以前所未有的速度增長

我們如今生活在一個完全被軟體所吞噬的世界。每個組織都是軟體組織，必須學會如何在數位化領域蓬勃發展並實現創新。

700M

未來五年內將誕生更多應用程式
這比過去40年加起來還多



人工智慧驅動的 開發者平臺





GitHub 進階安全

GitHub 高級安全提供一套 AppSec 工具，說明您的組織免受安全風險。

供應鏈安全

保護您的應用程式免受第三方依賴帶來的風險，並實現您創建的軟體的可驗證來源。

代碼安全

識別第一方代碼中的易受攻擊的編碼模式，並自動修復生成式人工智慧的問題。

機密保護

檢測硬編碼的機密，防止開發者誤上傳憑證到倉庫。

License 功能清單



GitHub Enterprise

- Dependency Graph
- Dependency Insights
- Software Bill of Materials (SBOM) generation
- Dependabot Alerts
- Dependabot Security Updates
- Dependabot Version Updates
- Security Overview
- Secret risk assessment
- All Code Security features for *public repositories*
- All Secret Protection features for *public repositories*



代碼安全許可

- CodeQL
- Code scanning
- Copilot Autofix
- Security campaigns
- Dependency Review
- Dependabot auto triage rules



機密保護許可

- Secret scanning
- Push protection
- Custom patterns
- AI-pattern generation
- Copilot secret scanning



GitHub 高級安全

策略與配置





GHAS 政策

- GHAS 策略可以在企業層級進行管理
- 這些與 GHAS 安全配置不同，後者可在企業或組織層級進行管理
- 在策略推行之初就確立策略定義至關重要，以避免策略偏離



abhishek-ghec

Overview

Organizations

People

Policies

Repositories

Codespaces

Copilot

Actions

Hosted compute networking

Projects

Code security and analysis

Personal access tokens Beta

Sponsors

GitHub Connect

Code Security

Settings

Compliance

Code security and analysis

Policies Security features

Dependency Insights

Dependency Insights provides a place to view all the packages that repositories depend on, including aggregated information about security advisories and licenses.

All organizations: No policy

View your organizations' current configurations without the enterprise's policy.

Enable or disable Dependabot alerts by repository admins

If allowed, repository admins can enable or disable Dependabot alerts. If not allowed, repository admins cannot enable or disable Dependabot alerts.

All organizations: Allowed

GitHub Advanced Security policies

Repository Admins can Enable or Disable GitHub Advanced Security

By allowing this policy, repository admins can choose to enable or disable Github Advanced Security on organization-owned repositories

All repositories: Allowed

Repository Admins can Enable or Disable Secret Scanning

By allowing this policy, repository admins can choose to enable or disable secret scanning, push protection, and validity checks on organization-owned repositories

All repositories: Allowed



安全配置

- GHAS 設置也可以在 GitHub 的多個層級應用——企業級、組織級和倉庫層面
- 作為安全經理和 GitHub 企業管理員，瞭解你希望在哪個層級執行這些設置非常重要，以便正確管理下游使用者的期望
- 通過合理規劃，你可以優化機密保護許可證的利用率

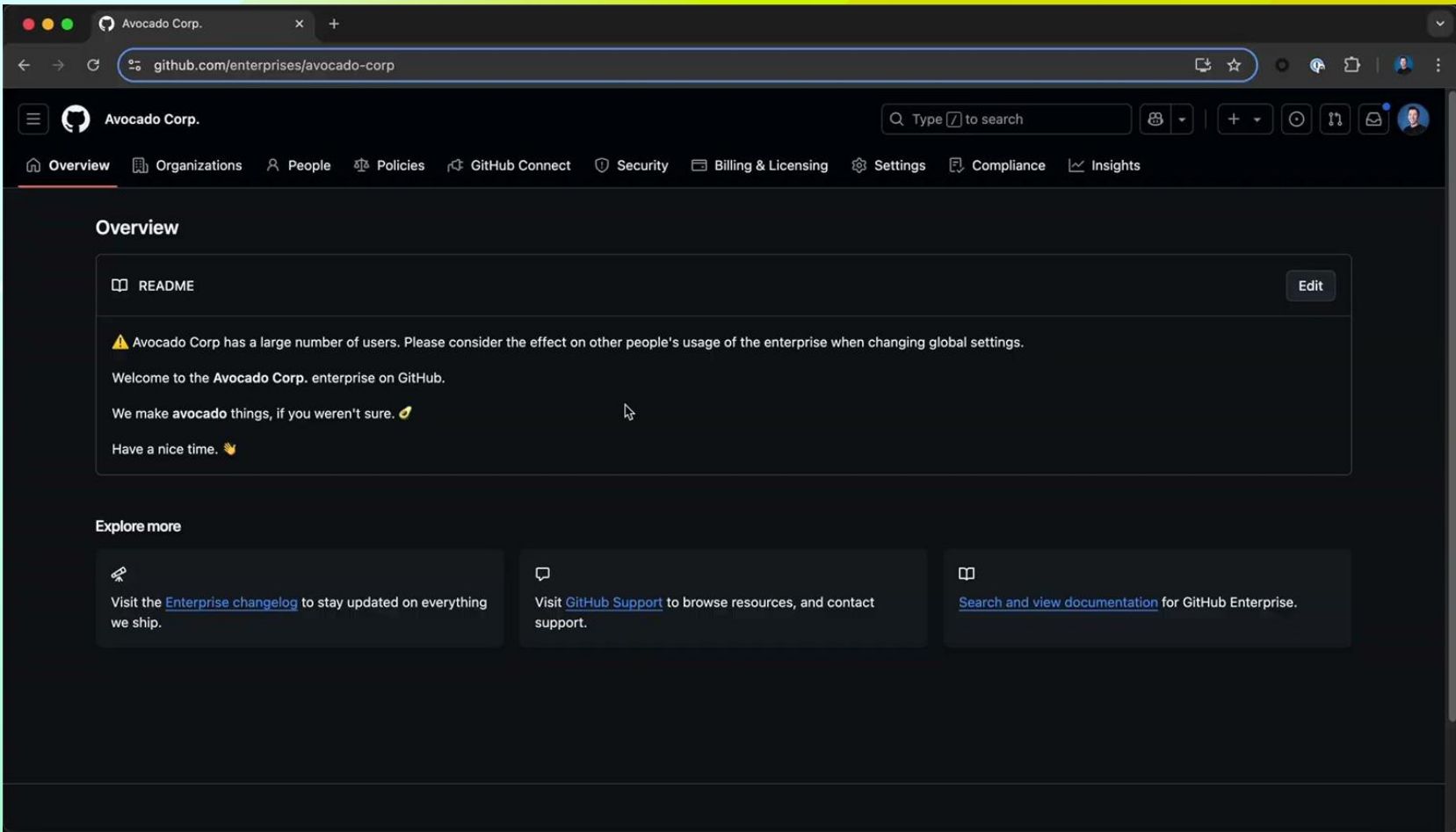


安全配置

- 通過定義可應用於多個倉庫組的安全設置集合，簡化 GitHub 安全產品的大規模推廣
- 應用“GitHub 推薦”的安全配置，或者創建自定義安全配置
- 根據不同風險配置檔或倉庫自定義屬性管理安全設置
- 查看應用配置所需的額外許可證數量，或通過禁用選定存儲庫中的功能來釋放許可證數量



Demo





Settings · Security configurations

github.com/organizations/callmegreg-sandbox/settings/security_products

GitHub Enterprise

Users managed by Volcano Coffee

callmegreg-sandbox

Type ↵ to search

+ ▾

🔍

🔗

📧

👤

Overview

Repositories

Projects

Packages


Teams

People

Security

Insights

Settings



callmegreg-sandbox

Organization, part of Volcano Coffee ↗ [Switch settings context ▾](#)

General

Policies ▾

Access

Billing and plans

Organization roles ▾

Repository roles

Member privileges

Import/Export

Moderation ▾

Code, planning, and automation

Repository ▾

Codespaces ▾

Planning ▾

Copilot ▾

Actions ▾

Webhooks

Security configurations

New configuration

Define and apply security configurations to make sure your repositories are protected.

Enterprise configurations

Managed by Volcano Coffee

GitHub recommended

GitHub Advanced Security

Enforced

Suggested settings for Dependabot, secret scanning, and code scanning. Default for all new repositories.

114 repositories

Apply to ▾

✎

Public Repository Default Settings

GitHub Advanced Security

This configuration includes your previous enterprise-level default settings for new public repositories as of December 2024. It will be applied if no organization-level defaults are set.

0 repositories

Apply to ▾

✎

Private/Internal Repository Default Settings

This configuration includes your previous enterprise-level default settings for new private/internal repositories as of December 2024. It will be applied if no organization-level defaults are set.

0 repositories

Apply to ▾

✎

Tip: As a Volcano Coffee admin, you can [manage callmegreg-sandbox configurations in enterprise settings.](#)

Apply configurations

4 GitHub Advanced Security licenses in use by Volcano Coffee.

Select repositories to apply configurations and view license consumption information.



Module 0: Lab exercises



Questions?



Break!



GitHub 高級安全

機密保護



機密掃描

- 掃描您 GitHub 倉庫中所有分支的完整 Git 歷史記錄以查找機密資訊，即使該倉庫已被歸檔。
- 同時掃描描述、問題中的評論、拉取請求以及 GitHub 討論。
- 啟用后，每次推送都會觸發掃描。GitHub 將定期執行完整 Git 歷史記錄掃描。



Secret scanning alerts

Q is:open		
<input type="checkbox"/>	6 Open	0 Closed
Bypassed		Validity
Secret type		
<input type="checkbox"/>	Amazon AWS Secret Access Key	wt61Vzva0QFx/U33PU8DrkMbn...
#6 opened 1 hour ago • Detected secret in storage/.../resources/.env:2		
<input type="checkbox"/>	Amazon AWS Access Key ID	AKIAZBVE345SKPTEAHQD
#5 opened 1 hour ago • Detected secret in storage/.../resources/.env:1		
<input type="checkbox"/>	Mailgun API Key	key-a67a11111111a11a1a1ba...
#4 opened 1 hour ago • Detected secret in storage/.../resources/.env:5		
<input type="checkbox"/>	GoCardless Live Access Token	live_A1N-kpH1H4wGhpLgwm5...
#3 opened 1 hour ago • Detected secret in storage/.../resources/.env:7		
<input type="checkbox"/>	Google API Key	AIzaSyDvc2t8M5wjfDonZ1e4x...
#2 opened 1 hour ago • Detected secret in storage/.../resources/.env:4		
<input type="checkbox"/>	Stripe API Key	sk_live_devboxbcct1Dfws2C...
#1 opened 1 hour ago • Detected secret in storage/.../resources/.env:6		


機密掃描





- 機密掃描的三種類型：
 - ✓ 供應商模式
 - ✓ 非供應商模式（通用型機密檢測規則）
 - ✓ 使用者自定義模式
- 可以為某些供應商進行有效性檢查


Secret scanning alerts


Q is:open


☐  6 Open ☒ 0 Closed Bypassed ▾ Validity ▾ Secret type ▾


☐  **Amazon AWS Secret Access Key** wt6lVzva0QFx/U33PU8DrkMbn...
#6 opened 1 hour ago • Detected secret in storage/.../resources/.env:2

☐  **Amazon AWS Access Key ID** AKIAZBVE345SKPTEAHQD
#5 opened 1 hour ago • Detected secret in storage/.../resources/.env:1

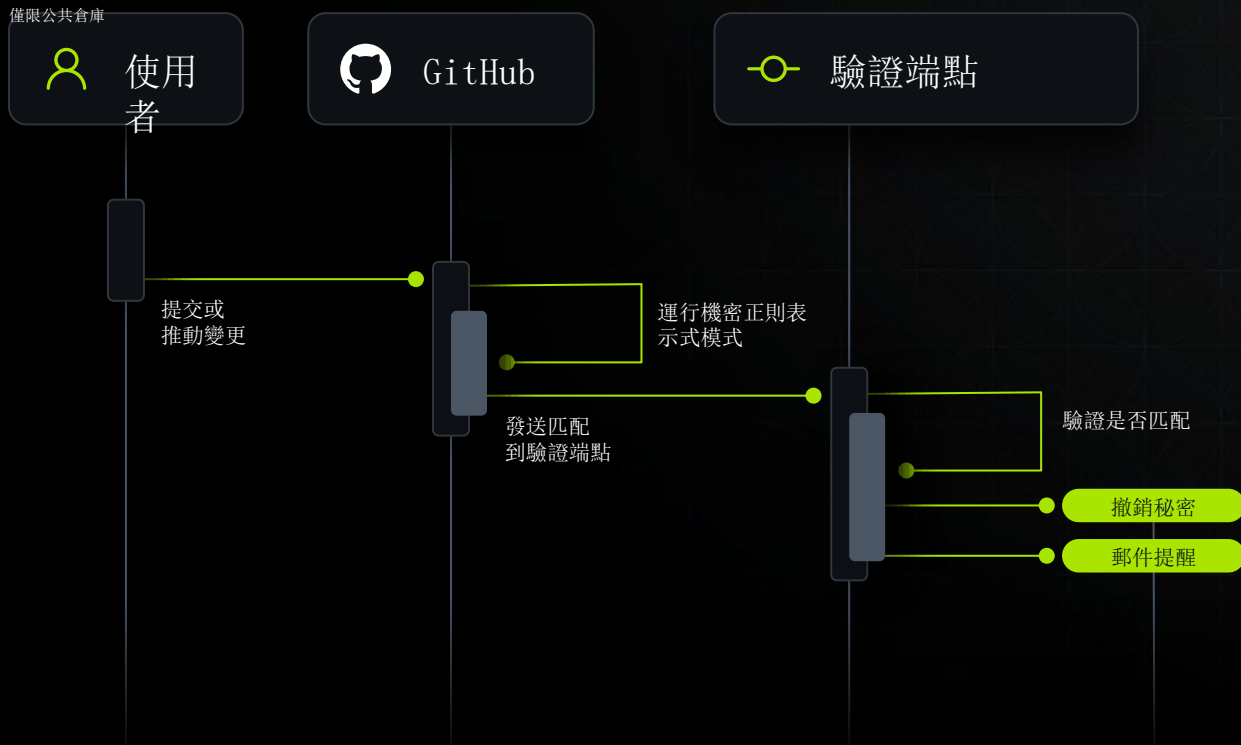
☐  **Mailgun API Key** key-a67a11111111a11a1a1ba...
#4 opened 1 hour ago • Detected secret in storage/.../resources/.env:5

☐  **GoCardless Live Access Token** live_A1N-kpH1H4wGhpLgwwm5...
#3 opened 1 hour ago • Detected secret in storage/.../resources/.env:7

☐  **Google API Key** AIzaSyDvc2t8M5wjfDonZ1e4x...
#2 opened 1 hour ago • Detected secret in storage/.../resources/.env:4

☐  **Stripe API Key** sk_live_devboxbcct1Dfws2C...
#1 opened 1 hour ago • Detected secret in storage/.../resources/.env:6

機密掃描合作夥伴計劃



- 150+ 服務提供者合作夥伴關係
- 主動通知：當供應商的憑證被洩露時發出警報
- 自動修復：合作夥伴可以選擇驗證和撤銷被洩露的機密
- 社區保護：保障所有開發者，而不僅僅是企業

自訂模式

- 使用正則表達式定義
作為密鑰掃描一部分檢測到的自定義模式
- 倉庫、組織與企業層級
- 密鑰掃描使用 Hyperscan 庫，該庫僅支援 Hyperscan 正則表達式構造。這是 PCRE 語法的子集
- 自訂模式指定符：
 - ✓ 金鑰格式
 - ✓ 金鑰前
 - ✓ 金鑰後
 - ✓ 附加匹配要求



Security & analysis / New custom pattern

Pattern name *

This cannot be edited after saving.

Secret format (specified as a regular expression) *

The pattern for the secret, specified as a regular expression. [Learn more about defining custom patterns.](#)

> More options

Test string *

Provide a sample test string to make sure your configuration matches the patterns you expect.

Save and dry run

自定義模式與人工智慧



- 正則表達式產生器允許使用者在無需掌握正則表達式知識的情況下生成自定義模式
- 輸入 – 待識別的模式的文字描述及可選示例字串
- 輸出 – 最多3個正則表示式

Generate regular expression

Beta

[Give feedback](#)



I want a regular expression that *

finds a valid URL that starts with http or https and contains the word CompanyXYZ

Examples of what I am looking for

`http://companyXYZ.com, https://companyxyz.com/`

This AI-powered feature may produce inaccurate results. Double-check the expressions generated and make any necessary adjustments.

Generate suggestions

推送保護

- 防止高可信度機密被推送到您的倉庫
- 通過安全配置在倉庫、組織或企業層面設置
- 自定義模式的推送保護按模式單獨管理
- 推送保護的委託繞過功能允許您管理貢獻者如何繞過推送保護規則，為未列入指定繞過清單的使用者添加審批流程。



❗ [Secret scanning](#) found a **Slack API Token** secret on [line 9](#).

Allowing this secret risks exposure. Instead, consider [removing the secret from your commit and commit history](#).

Exposing this secret can allow someone to:

- Verify the identity of this **Slack API Token** secret.
- Know which resources this secret can access
- Act on behalf of the secret's owner
- Push this secret to this repository without being blocked

☐ **It's used in tests**

The secret poses no risk. If anyone finds it, they cannot do any damage or gain access to sensitive information.

☐ **It's a false positive**

The detected string is not a secret.

☐ **I'll fix it later**

The secret is real, I understand the risk, and I will need to revoke it. This will open a security alert and notify admins of this repository.

Cancel

Allow Secret

通用機密與人工智慧



- 通用機密檢測是基於人工智慧的機密掃描擴展功能，可識別原始程式碼中的非結構化機密（密碼），並生成警報
- 存在誤報和漏報的可能性
- 檢測範圍僅限於代碼倉庫

The screenshot displays the GitHub Secret scanning alerts interface. On the left, a sidebar contains navigation links: Overview (selected), Reporting, Policy, Vulnerability alerts, Dependabot (121), Secret scanning (22), and Generic (1). The main content area is titled 'Secret scanning alerts' and features a search bar with the query 'is:open results:generic'. Below the search bar, there are filters for '1 Open' and '0 Closed' alerts. A table lists the detected secrets, with one entry visible: a 'Password' secret named 'secretsecret1234secretse...' detected in the file 'authn-service/authn-service.py:79' on August 30, 2024. The table has columns for 'Validity', 'Secret type', 'Provider', and 'Sort'.

Validity	Secret type	Provider	Sort
	Password		

修復 - 全倉庫掃描



已啟用
機密掃描



存儲庫被掃描
並打開警報



驗證機密



更換（輪換）機
密，並進一步調
查影響範圍



遵循貴公司在代
碼中引用機密
的最佳實踐



關閉警報

修復 - 推送保護



推送保護
已啟用



每次git推送都會被
掃描是否有機密



推送被拒絕



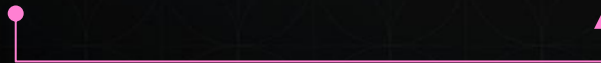
評估機密
是否真實



從所有提交中
移除該機密



再推
一次



繞行



委託繞行

- **受控繞過流程：** 只有指定的審核員才能批准繞過請求。
- **審批流程：** 沒有繞過許可權的貢獻者必須提交請求
- **詳細請求資訊：** 每個請求包含使用者資訊、倉庫、提交哈希、時間戳和文件路徑，方便決策。
- **細粒度許可權：** 賦予特定角色或團隊管理繞過請求的能力。
- **審計追蹤：** 所有作都會被記錄，提供透明度和問責制。



委託警報關閉

- **受限警報撤銷：** 只有組織擁有者和安全經理可以直接關閉機密掃描警報。
- **強制審核流程：** 其他用戶必須提交撤銷請求，需經指定審核員批准。
- **需要提供依據：** 每次撤銷請求都必須包含理由，以協助審計和合規工作。
- **審計日誌：** 所有撤銷行為均記錄在警報時間線和審計日誌中，確保可追溯性。

拉取請求中的機密掃描

- 機密掃描審查 是一種可選工作流程，可增強對拉取請求中新增機密警報的感知能力。
- 結合倉庫規則集使用，可在機密警報解決前阻止合併操作。
- 系統將生成拉取請求級別的摘要，包含新增機密警報的清單、位置、類型及狀態。



Workflow file for this run

.github/workflows/secret-scanning.yml at 1da9c50

```
1 name: 'Secret Scanning Review'
2 on: [pull_request]
3
4 jobs:
5   secret-scanning-review:
6     runs-on: ubuntu-latest
7     steps:
8       - name: 'Secret Scanning Review Action'
9         uses: advanced-security/secret-scanning-review-action@v1
10        with:
11          token: ${ secrets.SECRET_SCAN_REVIEW_GITHUB_TOKEN }}
12          fail-on-alert: true
13          fail-on-alert-exclude-closed: true
```



abhi-github-staff commented 4 hours ago

Author ...

🔒 PR#10 SECRET SCANNING REVIEW SUMMARY 🔒

Found [2] secret scanning alerts across [2] locations that originated from a PR#10 commit

Status	Secret Alert	Secret Type	State	Resolution	Push Bypass	Commit
	8	my-token	open		False	1da9c50
	7	test-pattern-1	open		False	1da9c50



Some checks were not successful

3 failing and 3 successful checks

Hide all checks

	Dependency Review / dependency-review (pull_request)	Failing after 5s	Details
	Secret Scanning Review / secret-scanning-review (pull_request)	Failing after 29s	Details
	CodeQL / Analyze (javascript-typescript, none) (pull_request)	Successful in 1m	Details
	CodeQL / Analyze (csharp, none) (pull_request)	Successful in 3m	Details
	CodeQL / compliance (pull_request)		Details

Google Chrome

Required

通知



歷史掃描

通知將發送至：

- 組織擁有者、企業擁有者和安全管理員 – 每次歷史掃描完成時，即使未發現任何機密資訊。
- 倉庫管理員、安全管理員以及具有讀寫許可權的自定義角色使用者 – 每次歷史掃描檢測到機密資訊時，並根據其通知偏好設置。
- 我們不會通知提交者。

增量掃描

通知將發送至：

- 倉庫管理員
- 安全管理員
- 具有讀寫許可權的自定義角色使用者
- 組織擁有者和企業擁有者（若其為洩露機密信息的倉庫管理員）
- 誤提交機密資訊的提交作者將收到通知，無論其通知偏好設置如何



Demo



Module 2: Lab exercises



Questions?



Break!



GitHub 高級安全

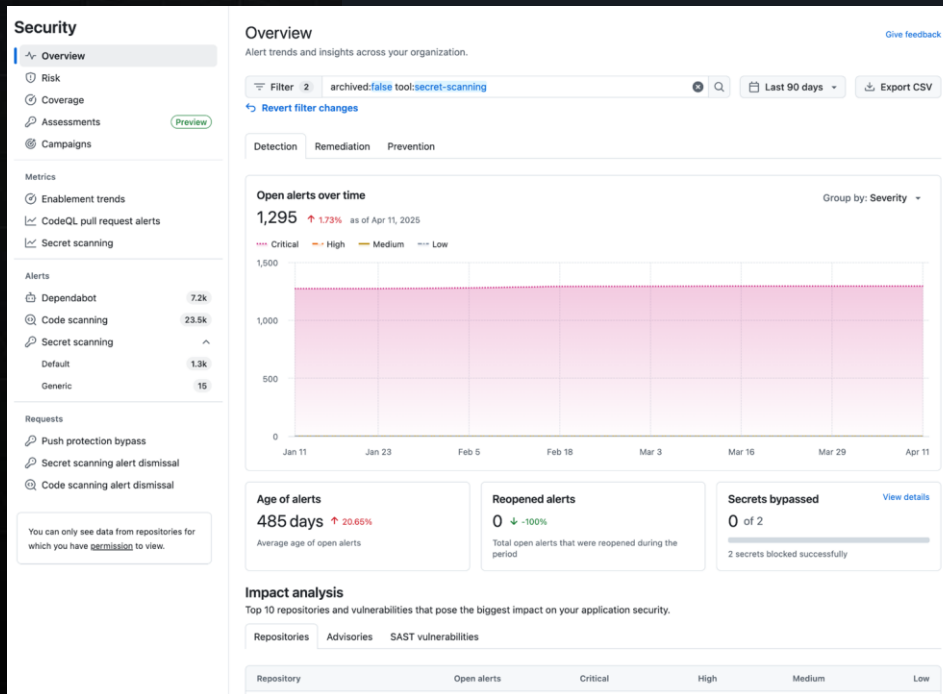
安全概述





安全概述

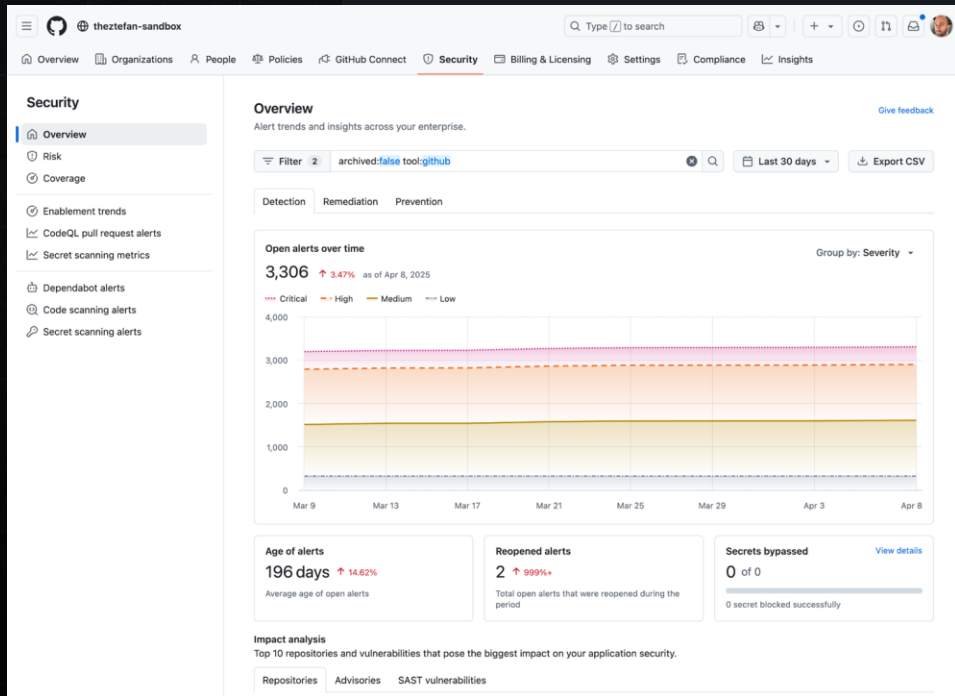
- 提供組織或企業安全態勢的高級概覽，便於快速識別需要干預的代碼庫
- 忽略目錄的 Secret 掃描警報將從這些檢視中省略





企業層級

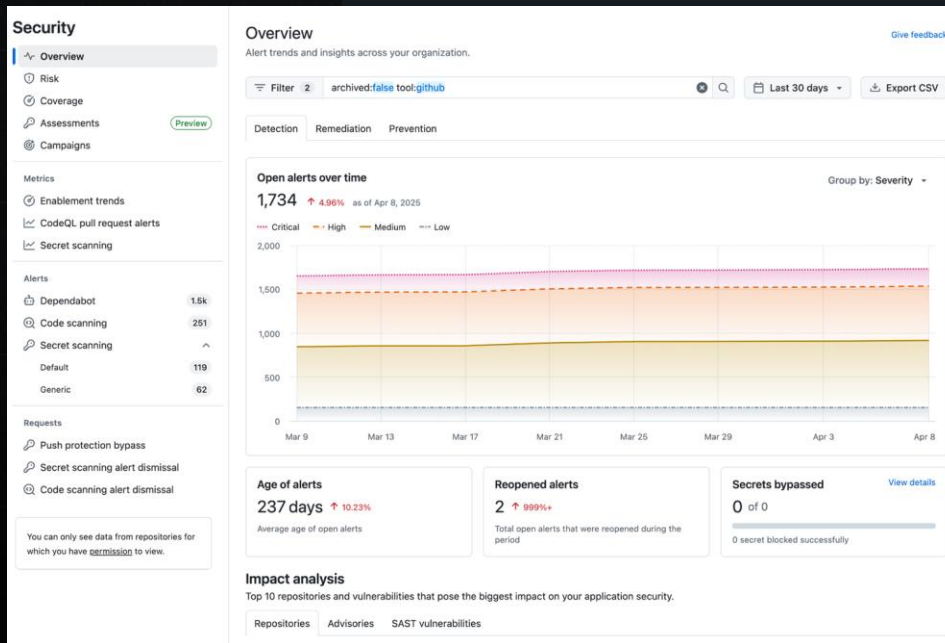
- 企業安全選項卡包含與高級安全相關的原生儀錶板
- 機密掃描指標選項卡包含與推送保護相關的攔截和繞過詳情
- 機密掃描警報選項卡顯示所有組織中所有警報的匯總資訊
- 您僅能查看已獲訪問許可權的警報數據





組織層級

- 在組織層面，查看整體風險狀況及各代碼庫的防護覆蓋範圍
- “預設”選項卡顯示來自供應商模式和自定義模式的警報。
- “通用”選項卡顯示來自非供應商模式及AI檢測到的機密警報





倉庫層級

- 在存儲庫層級，根據有效性檢查、金鑰類型或供應商對警報進行審查和篩選。
- 默認情況下，僅存儲庫管理員、組織擁有者和安全管理員可見密鑰掃描選項卡。

The screenshot displays the GitHub Security Center interface, specifically the 'Secret scanning alerts' section. The left sidebar shows navigation options: Overview, Reporting, Policy, Vulnerability alerts, Dependabot (66), Code scanning, Secret scanning (36), and Generic. The main content area is titled 'Secret scanning alerts' and includes a filter bar with 'Filter 2' and a search query 'is:open secret-type:aws_access_key_id,aws_secret_access_key,gocardless_live_access_token,google_api_key,mailg...'. Below the filter bar, there are two tabs: '8 Open' and '0 Closed'. The list of alerts shows various detected secrets, including 'mona_value', 'Amazon AWS Secret Access Key', 'GoCardless Live Access Token', 'Mailgun API Key', 'Stripe API Key', 'Amazon AWS Access Key ID', and 'Google API Key'. Each alert entry includes a status icon, a title, a truncated secret value, a description of where it was found, and a 'Public leak' label.

Validity	Secret type	Provider	Sort
<input type="checkbox"/>	mona_value	mona_value_abc124	#8 opened on 10 May 2024 • Detected custom pattern in storage/.../resources/application.properties:5
<input type="checkbox"/>	mona_value	mona_value_abc123	#7 opened on 10 May 2024 • Detected custom pattern in passwords.txt:3
<input type="checkbox"/>	Amazon AWS Secret Access Key	wt6lVza0QFx/U33PU8DrkMb...	#6 opened on 9 May 2024 • Detected secret in storage/.../resources/env:2 Public leak
<input type="checkbox"/>	GoCardless Live Access Token	live_ALN~kpH1H4wGhpLgwm...	#5 opened on 9 May 2024 • Detected secret in storage/.../resources/env:7 Public leak
<input type="checkbox"/>	Mailgun API Key	key~a67a1111111a1a1a1b...	#4 opened on 9 May 2024 • Detected secret in storage/.../resources/env:5 Public leak
<input type="checkbox"/>	Stripe API Key	sk_live_devboxbct1DfwS2...	#3 opened on 9 May 2024 • Detected secret in storage/.../resources/env:6 Public leak
<input type="checkbox"/>	Amazon AWS Access Key ID	AKIAZBVE345SKPTEAHQD	#2 opened on 9 May 2024 • Detected secret in storage/.../resources/env:1 Public leak
<input type="checkbox"/>	Google API Key	AIzaSyDvc2t8H5wjfDonZ1e4...	#1 opened on 9 May 2024 • Detected secret in storage/.../resources/env:4 Public leak



Demo



Module 2: Lab exercises



Questions?



GitHub 高級安全

第三方集成

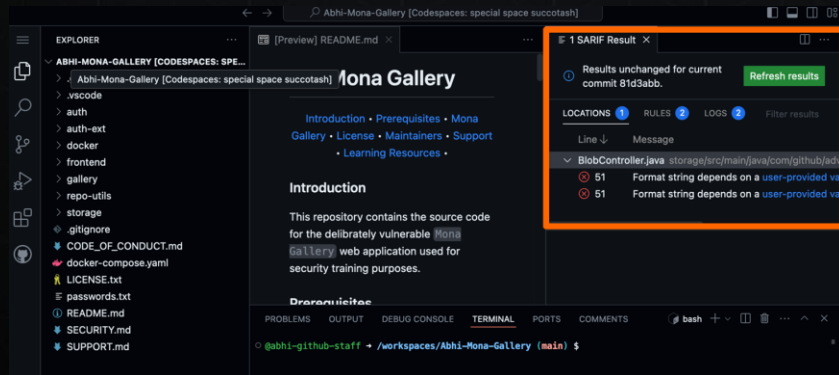




VS Code 集成

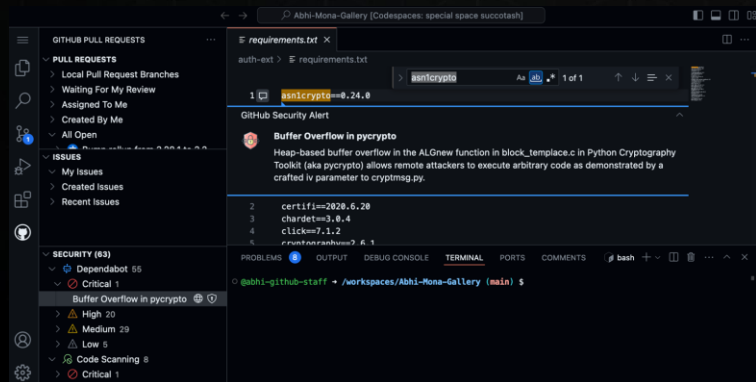
Sarif Viewer

開發者可以通過 SARIF Viewer 外掛程式在他們的 VS Code IDE 中查看 CodeQL 漏洞



GitHub Security Alerts

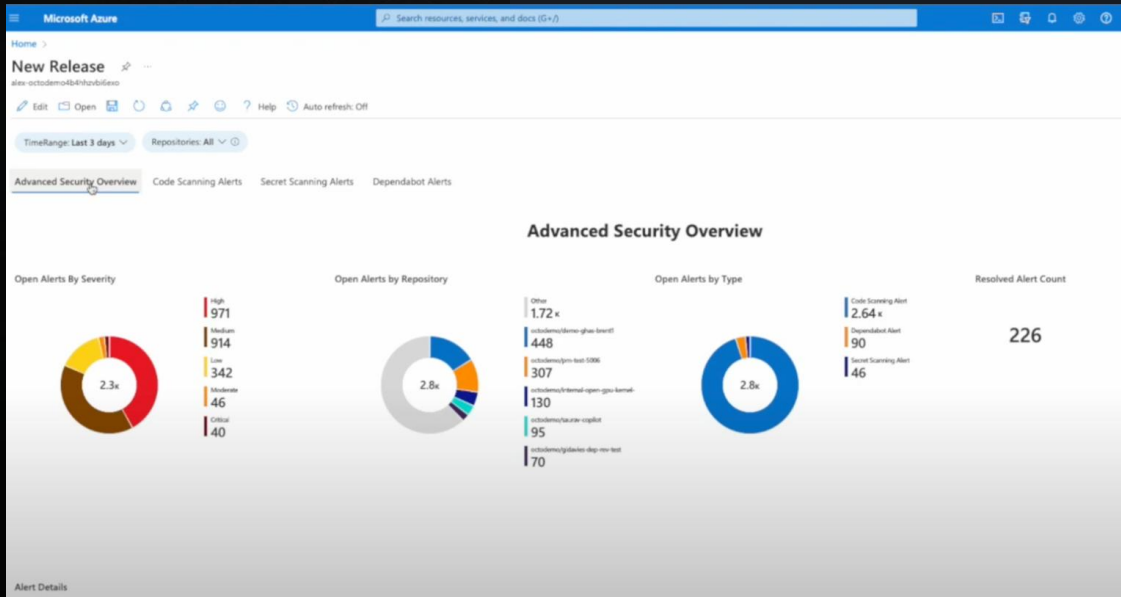
另一款第三方 VS Code 擴展是 GitHub Security Alerts 外掛程式，可將 GHAS 漏洞資訊導入 VS Code IDE 中





SIEM 集成

- Splunk
- Microsoft Sentinel
- DataDog
- Sumo Logic
- Elastic Security
- Panther





Jira 集成

Jira 現已原生支援將 GHAS 漏洞導入系統，並自動在 Jira 中創建相應問題。

Integrate GitHub Advanced Security with Jira

i These instructions are for connecting **GitHub Cloud** or **GitHub Enterprise Cloud** to Jira. [Show me how to connect GitHub Enterprise Server](#)

The security feature in Jira allows you to view, triage, and track security vulnerabilities from GitHub Advanced Security. To get this feature working, you'll need to:

1. Install the GitHub for Jira app.
2. Connect a GitHub organization.
3. Add GitHub Advanced Security to your Jira project.
4. Connect security containers to your project.

Before you begin

- i** To install and set up the GitHub for Jira app, you need:
- Site administrator permission for your Jira site.
 - Organization owner permission for your GitHub organization.

For some organizations, the task of integrating GitHub Advanced Security might involve multiple team members:

- A Jira site admin will install the GitHub for Jira app.
- A GitHub organization owner will connect a GitHub organization to your Jira site.
- A Jira project admin will add GitHub Advanced Security to a project and connect security containers.



第三方工具

- 代碼掃描允許您整合其他靜態分析安全工具的結果。
- 輸出數據格式應為 SARIF
- 如果在 GitHub Actions 中運行掃描（例如 IaC），請使用 *upload-sarif*
- 上傳也可以通過 REST API 完成
- 漏洞可在 GitHub 的安全概述中查看

```
Code Blame 43 lines (37 loc) · 1.09 KB
1 name: scan with KICS and upload SARIF
2
3 on:
4   push:
5     branches: [master]
6
7 jobs:
8   kics-job:
9     runs-on: ubuntu-latest
10    name: kics-action
11    strategy:
12      fail-fast: false
13    steps:
14      - name: Checkout repo
15        uses: actions/checkout@v3
16      - name: Mkdir results-dir
17        # make sure results dir is created
18        run: mkdir -p results-dir
19
20      - name: Run KICS Scan with SARIF result
21        uses: checkmarx/kics-github-action@v2.1.0
22        with:
23          path: 'terraform'
24          output_path: results-dir
25          platform_type: terraform
26          output_formats: 'json,sarif'
27          ignore_on_exit: results
28
29      - name: Show results
30        run: |
31          cat results-dir/results.sarif
32          cat results-dir/results.json
33
34      - name: Archive code coverage results
35        uses: actions/upload-artifact@v4
36        with:
37          name: result
38          path: results-dir/results.sarif
39
40      - name: Upload SARIF file
41        uses: github/codeql-action/upload-sarif@v1
42        with:
43          sarif_file: results-dir/results.sarif
```

審計日誌



dependabot_*

dependency_graph_*

repository_secret_scanning*

repository_vulnerability_*

secret_scanning_

Events Settings

Audit log



Filters

[Export Git Events -](#)


[Export -](#)

☐ Clear current search query

Events matching search query

-  **abhi-github-staff** - **secret_scanning_push_protection.bypass**
Bypassed the push protection for a secret as false positive `Reusable-Test/JuiceShop:alert#45`
Unknown location | Unknown IP address | 19 hours ago | [...](#)
-  **abhi-github-staff** - **secret_scanning_push_protection.bypass**
Bypassed the push protection for a secret as used in tests `Reusable-Test/JuiceShop:alert#44`
Unknown location | Unknown IP address | 19 hours ago | [...](#)

[Newer](#) [Older](#)

 **ProTip!** Exclude events created by you with [-actor:abhi-github-staff](#)



代碼安全審計日誌

`code_scanning.alert_*`

記錄代碼掃描警報的創建、關閉、撤銷、重新打開等操作。

`org|repo.code_scanning_*`

記錄代碼掃描相關配置的變更或代碼掃描分析的刪除。

`security_configuration*`

記錄組織或企業層級安全配置的創建、刪除或更新。

`org.codeql*`

記錄代碼掃描預設設置啟用或禁用時的日誌。

`business.code_scanning|code_security*`

記錄企業級 GHAS 策略的任何變更。

Webhook



`code_scanning_alert`

儲存庫、組織、應用程式

`dependabot_alert`

倉庫、組織、應用程式

`secret_scanning_alert`

儲存庫、組織、應用程式

`secret_scanning_alert_location`

儲存庫、組織、應用程式

`security_advisory`

應用程式

`security_and_analysis`

儲存庫、組織、應用程式

代碼掃描 Webhook



`code_scanning_event`

儲存庫中與代碼掃描警報相關的活動事件：創建、關閉、忽略、重新打開、出現在分支中

`dismissal_request_code_scanning`

使用者請求忽略警報時的事件

`security_and_analysis`

為存儲庫啟用或禁用代碼安全與分析功能時的事件

APIs



REST

`/code-scanning`

Enterprise, Organization & Repository

`/dependabot`

Enterprise, Organization & Repository

`/dependency-graph`

Repository

`/secret-scanning`

Enterprise, Organization & repository

GraphQL

Dependabot 更新

安全漏洞

安全公告

预览: DependencyGraph

機密掃描 APIs



REST

管理現有警報

列出警報位置

獲取機密掃描記錄

管理推送保護繞過請求

GRAPHQL

目前沒有秘密掃描支援



Questions?



Module 2: Lab exercises



GitHub
機密保護培訓

回顧



GitHub 機密保護



安全漏洞短期內不會消失



安全配置可讓您個人化部署方案



推送保護防止機密洩漏發生



自訂模式允許你擴展機密掃描



安全概述包括原生儀錶盤和洞察



API、webhook 和審計日誌可以匯出警報數據

接下來要做的三件事！

☑ 制定計劃

確定你何時以及如何為你的倉庫部署機密保護。

☑ 與開發者溝通

讓開發者知道什麼時候會發佈，以及對他們的期望。

☑ 需要時自定義

優先關注供應商模式和推送保護，隨後逐步推進通用密鑰與自定義模式





Thank you