Config Synk
วิธีการใช้ Cli

https://docs.snyk.io/snyk-cli

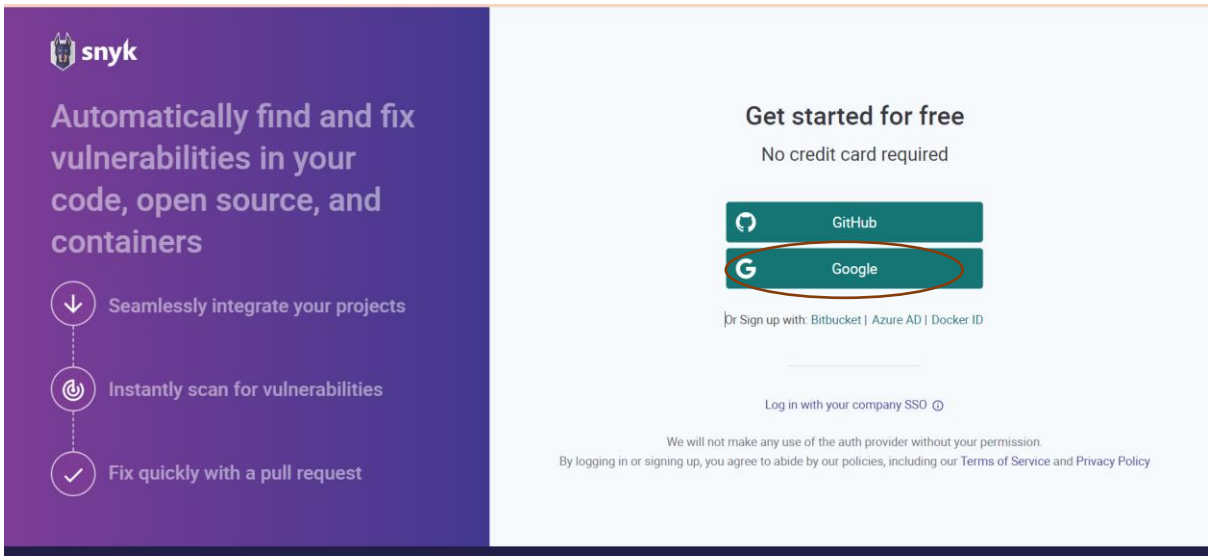
วิธีการลงทะเบียนใช้งาน

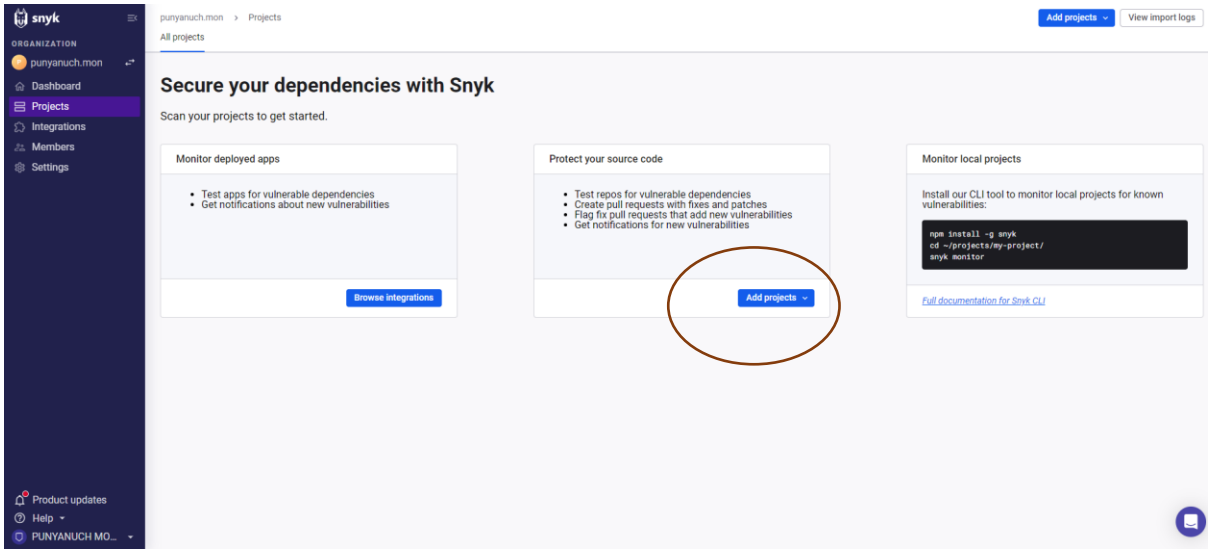
- **ลงทะเบียนใช้ Google ไปเลยมันจะได้ไม่ link กับ Github เรา**

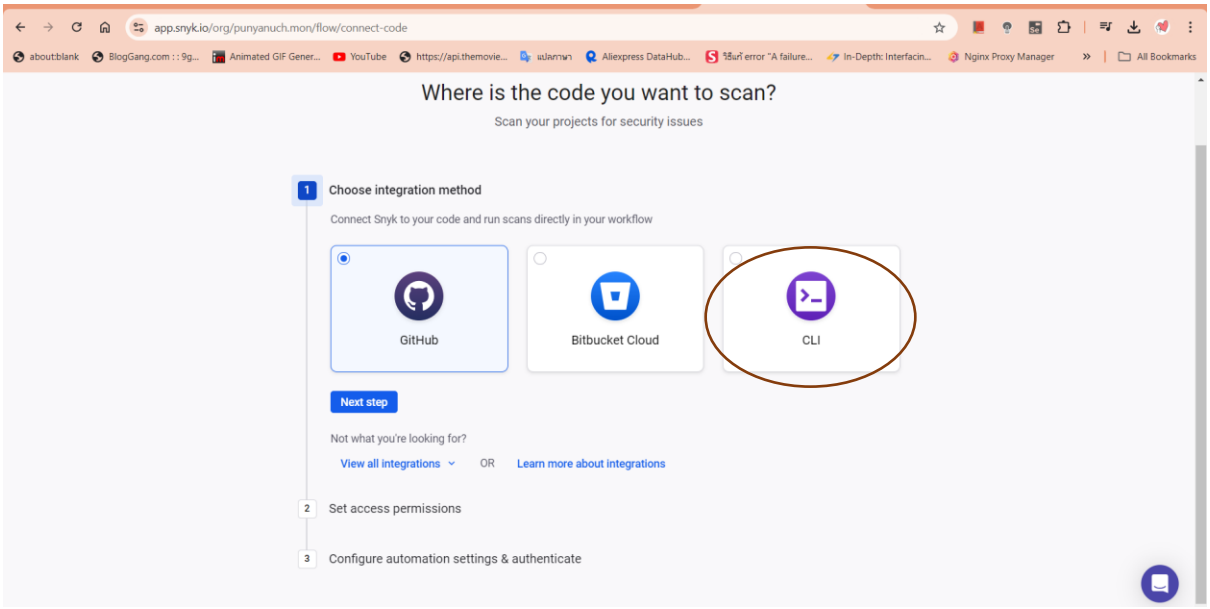
https://docs.snyk.io/getting-started/quickstart/create-or-log-in-to-a-snyk-account

- Snyk auth

https://app.snyk.io/login







- **curl** https://static.snyk.io/cli/latest/snyk-win.exe -o snyk.exe
- เอาเข้า **Project** เรา

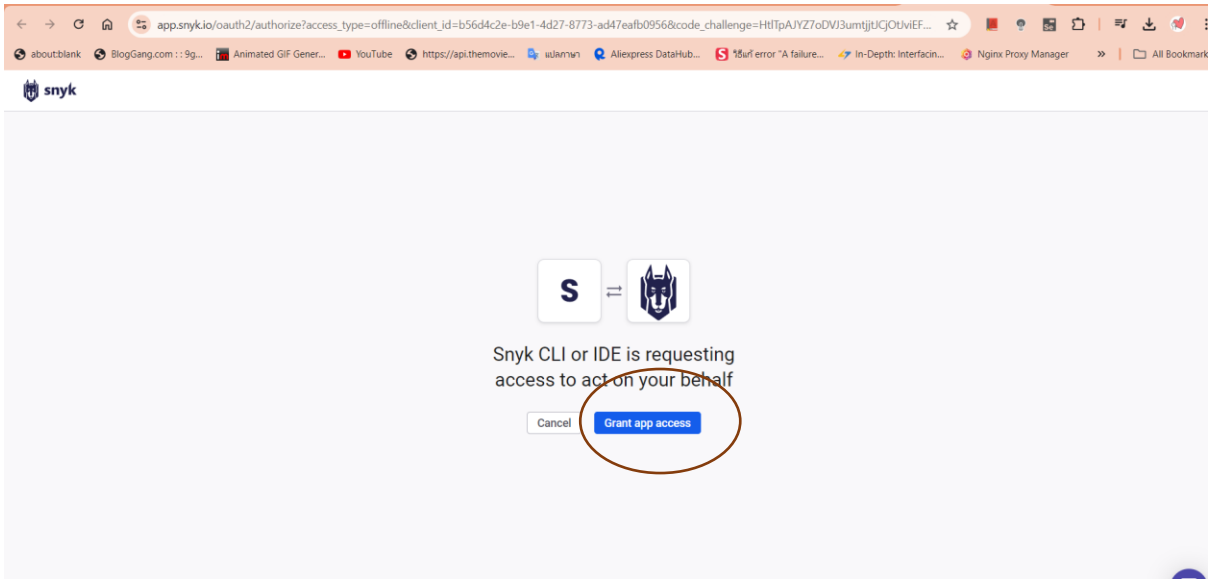
```
C:\Users\punyanuch\Desktop\Test Snyk>curl https://static.snyk.io/cli/latest/snyk-win.exe -o snyk.exe
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 96.0M  100 96.0M    0     0  30.2M      0  0:00:03  0:00:03 --:--:-- 30.3M
```

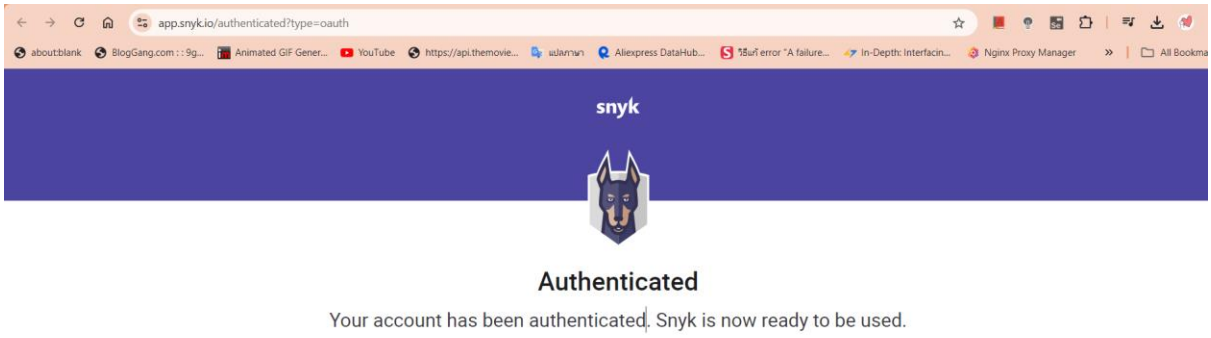
- **Synk**
- ตรวจสอบว่ามันเข้ามาแล้วยัง สามารถใช้ได้มั้ย

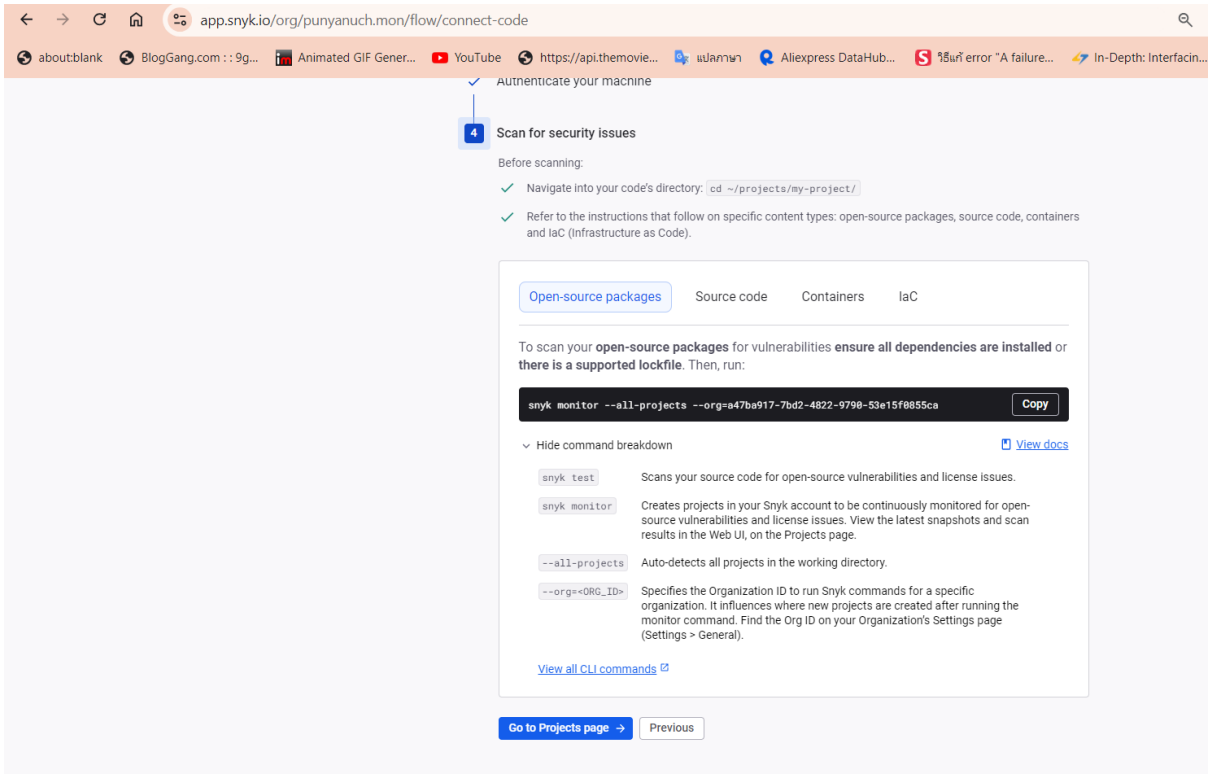
```
C:\Users\punyanuch\Desktop\Test Snyk>snyk
CLI help
  Snyk CLI scans and monitors your projects for security vulnerabilities and license issues.

  For more information visit the Snyk website https://snyk.io

  For details see the CLI documentation https://docs.snyk.io/features/snyk-cli

How to get started
  1. Authenticate by running snyk auth.
  2. Test your local project with snyk test.
  3. Get alerted for new vulnerabilities with snyk monitor.

Available commands
  To learn more about each Snyk CLI command, use the --help option, for example, snyk auth --help.

  Note: The help on the docs site is the same as the --help in the CLI.

  snyk auth
    Authenticate Snyk CLI with a Snyk account.
```

- **Synk auth**

```
C:\Users\punyanuch\Desktop\Test Snyk>snyk auth
```


app.snyk.io/oauth2/authorize?access_type=offline&client_id=b56d4c2e-b9e1-4d27-8773-ad47eafb0956&code_challenge=HtlTpAJYZ7oDVJ3umtjjtJCjOtJviEF...
snyk
Snyk CLI or IDE is requesting access to act on your behalf
Cancel
Grant app access


app.snyk.io/authenticated?type=oauth
snyk
Authenticated
Your account has been authenticated. Snyk is now ready to be used.



https://app.snyk.io/org/xxxxxxxxxxx/projects?groupBy=targets&before&after&searchQuery=&sortBy=highest+severity&filters[Show]=&filters[Integrations]=&filters[CollectionIds]=

ตัวที่ไป Clone อธิบาย

https://github.com/snyk-labs/nodejs-goof

*** ไม่จำเป็นต้องใส่ npm install นะ เพราะตอนที่พี่อัดวิดีโอ พี่เคยมีโปรเจคอยู่แล้ว พี่อธิบายไปใหม่ช่วงเวลาคลิป 17.00 เป็นต้นไป ***

nodejs-goof เป็น Project ที่ใช้ node js ซึ่งเจ้า Lab ตัวนี้ออกแบบมาเพื่อ ทดสอบและเรียนรู้เกี่ยวกับช่องโหว่ต่าง ๆ เช่น การโจมตีผ่านช่องโหว่ของ Dependency, การจัดการข้อมูลที่ไม่ปลอดภัย, และอื่น ๆ โดย Snyk

- yarn.lock
- package-lock.json
- package.json
- Gemfile.lock
- pom.xml
- build.gradle
- build.sbt
- Pipfile
- requirements.txt
- Gopkg.lock
- vendor/vendor.json
- obj/project.assets.json
- packages.config
- composer.lock
- build.gradle.kts
- go.mod

ซึ่งจากการแจ้งเตือน

https://docs.snyk.io/snyk-cli/scan-and-maintain-projects-using-the-cli/snyk-cli-for-snyk-code/view-snyk-code-cli-results --> คำอธิบายแต่ละ Result

สรุปปัญหาความปลอดภัยและการอัปเกรดแพ็กเกจ

- **adm-zip**:
    - ปัญหา: Directory Traversal (ความรุนแรงสูง), Arbitrary File Write via Archive Extraction (Zip Slip) (ความรุนแรงวิกฤต)

    - การอัปเกรด: จากเวอร์ชัน 0.4.7 เป็น 0.5.2

- **body-parser**:
    - ปัญหา: Prototype Poisoning (ความรุนแรงสูง), Prototype Override Protection Bypass (ความรุนแรงสูง)
    - การอัปเกรด: จากเวอร์ชัน 1.9.0 เป็น 1.19.2

- **cfenv**:
    - ปัญหา: Arbitrary Code Injection (ความรุนแรงปานกลาง)
    - การอัปเกรด: จากเวอร์ชัน 1.2.2 เป็น 1.2.4

- **dustjs-linkedin**:
    - ปัญหา: Prototype Pollution (ความรุนแรงสูง), Code Injection (ความรุนแรงสูง)
    - การอัปเกรด: จากเวอร์ชัน 2.5.0 เป็น 3.0.0

- **ejs**:
    - ปัญหา: Improper Control of Dynamically-Managed Code Resources, Arbitrary Code Injection, Cross-site Scripting (XSS), Denial of Service (DoS), Remote Code Execution (RCE)
    - การอัปเกรด: จากเวอร์ชัน 1.0.0 เป็น 3.1.10

- **errorhandler**:
    - ปัญหา: Regular Expression Denial of Service (ReDoS) (ความรุนแรงสูง)
    - การอัปเกรด: จากเวอร์ชัน 1.2.0 เป็น 1.4.3

- **express**:
    - ปัญหา: ReDoS, Open Redirect, Prototype Poisoning
    - การอัปเกรด: จากเวอร์ชัน 4.12.4 เป็น 4.19.2

- **express-fileupload**:
    - ปัญหา: Missing Release of Resource after Effective Lifetime, ReDoS, Denial of Service (DoS), Prototype Pollution
    - การอัปเกรด: จากเวอร์ชัน 0.0.5 เป็น 1.1.10

- **hbs**:

- ปัญหา: Prototype Pollution, Remote Code Execution (RCE), Denial of Service (DoS)
- การอัปเกรด: จากเวอร์ชัน 4.0.4 เป็น 4.1.2

- **humanize-ms**:
  - ปัญหา: ReDoS
  - การอัปเกรด: จากเวอร์ชัน 1.0.1 เป็น 1.2.1

อธิบายปัญหาความปลอดภัย

- **Directory Traversal (adm-zip)**:
  - **คำอธิบาย:** ช่องโหว่ที่อนุญาตให้โจมตีเข้าถึงไฟล์และไดเรกทอรีที่อยู่นอกพื้นที่ที่ควรจะเข้าถึงได้ ผ่านการจัดการพาธไฟล์ที่ไม่ถูกต้อง
- **Arbitrary File Write via Archive Extraction (adm-zip)**:
  - **คำอธิบาย:** ช่องโหว่ที่อนุญาตให้เขียนไฟล์ไปยังตำแหน่งที่ไม่ได้รับอนุญาต ผ่านการแยกไฟล์จาก ZIP ซึ่งอาจนำไปสู่การเขียนไฟล์ในตำแหน่งที่ไม่ปลอดภัย
- **Prototype Poisoning (body-parser, dustjs-linkedin)**:
  - **คำอธิบาย:** ช่องโหว่ที่ทำให้การโจมตีสามารถเปลี่ยนแปลงพฤติกรรมของโปรโตไทป์ของออบเจ็กต์ ซึ่งอาจนำไปสู่การทำงานที่ไม่คาดคิดหรือการโจมตี
- **Prototype Override Protection Bypass (body-parser)**:
  - **คำอธิบาย:** ช่องโหว่ที่ทำให้สามารถข้ามการป้องกันการเขียนทับโปรโตไทป์ ซึ่งอาจทำให้เกิดปัญหาความปลอดภัย
- **Arbitrary Code Injection (cfenv, dustjs-linkedin)**:
  - **คำอธิบาย:** ช่องโหว่ที่อนุญาตให้ผู้โจมตีสามารถฉีดโค้ดที่ไม่ต้องการเข้าไปในระบบ ซึ่งอาจนำไปสู่การควบคุมหรือการโจมตีอื่น ๆ
- **Improper Control of Dynamically-Managed Code Resources (ejs)**:
  - **คำอธิบาย:** ช่องโหว่ที่ทำให้ไม่สามารถควบคุมทรัพยากรโค้ดที่จัดการโดยไดนามิก ซึ่งอาจนำไปสู่การโจมตี
- **Cross-site Scripting (XSS) (ejs)**:
  - **คำอธิบาย:** ช่องโหว่ที่อนุญาตให้ฉีดสคริปต์ที่เป็นอันตรายไปยังหน้าเว็บที่ถูกแสดงให้กับผู้ใช้
- **Denial of Service (DoS) (ejs, express-fileupload, hbs)**:

- **คำอธิบาย:** ช่องโหว่ที่ทำให้ระบบไม่สามารถให้บริการได้เนื่องจากการโจมตีที่ส่งข้อมูลที่ทำให้ระบบช้าเกินไปหรือหยุดทำงาน

- **Remote Code Execution (RCE) (ejs, hbs)**:
  - **คำอธิบาย:** ช่องโหว่ที่อนุญาตให้ผู้โจมตีสามารถรันโค้ดจากระยะไกลบนเซิร์ฟเวอร์ ซึ่งอาจทำให้ควบคุมระบบทั้งหมดได้

- **Regular Expression Denial of Service (ReDoS) (errorhandler, humanize-ms, express-fileupload)**:
  - **คำอธิบาย:** ช่องโหว่ที่ทำให้การประมวลผลนิพจน์ปกติ (Regular Expression) ใช้เวลานานเกินไปหรือทำให้ระบบหยุดทำงานเมื่อเผชิญกับการโจมตี

- **Open Redirect (express)**:
  - **คำอธิบาย:** ช่องโหว่ที่ทำให้การเปลี่ยนเส้นทางของ URL ไปยังเว็บไซต์ที่ไม่ปลอดภัยหรือไม่พึงประสงค์ ซึ่งอาจนำไปสู่การโจมตีแบบฟิชชิ่ง

- **Missing Release of Resource after Effective Lifetime (express-fileupload)**:
  - **คำอธิบาย:** ช่องโหว่ที่เกิดจากการไม่ปล่อยทรัพยากรหลังจากที่หมดอายุการใช้งาน ซึ่งอาจทำให้เกิดปัญหาการใช้ทรัพยากรเกินพิกัด