OPENCHAIN KWG

최근 오픈소스 주요 동향

카카오 오픈소스기술파트 robin.hwang

- 미 정부의 'SBOM' 의무화
- Github Copilot 오픈소스 라이선스 논쟁

최근 대형 보안 사고

솔라윈즈(SolarWinds) 사태 (2020.12)

- 솔라윈즈 : 네트워크 모니터링 서비스 및 솔루션
- 솔라윈즈 소프트웨어 업데이트 파일을 통한 침투
- 2019년 9월부터 침투를 시도, 2020년 12월 공격 사실이 밝혀짐
- 국방부, 연방법원, 재무부 등 굵직한 미국 연방 정부기관과 유명 기업들 피해
- 공격자들은 솔라윈즈 이외에도 다양한 방법으로 피해를 입힘

공급망 공격
Supply Chain Attack

익스체인지(Exchange) 사태 (2021.03)

- MS 익스체인지 서버에서 발견된 제로데이 취약점으로 백도어 및 멀웨어 공격
- 취약점 패치가 이뤄졌으나 아직도 패치되지 않은 서버를 대상으로 공격이 시도됨

제로 데이 공격
Zero-Day Attack

콜로니얼 파이프라인(Colonial Pipelline) 사태 (2021.05)

- 미국의 대형 송유관 업체인 콜로니얼 파이프라인을 마비시킨 대형 랜섬웨어 사건
- 정상화 대가로 해킹 단체에 500만 달러 (약 56억원) 지불, 이후 추적을 통해 대부분 회수

OT 보안
Operational Technology Security

미 정부의 'SBOM' 의무화

지난 5월 행정 명령을 통해 소프트웨어 명세서(SBOM)의 위상을 격상

연방정부와 사업 계약을 맺은 기업이라면 SBOM 제출을 의무화

주요 내용

- 위협 정보 공유 장벽 제거
- 연방 정부의 사이버 보안 현대화
- 소프트웨어 공급망 보안 개선
- 국책 사이버안전심의위원회 설치
- 사이버 보안 취약성 및 사고에 대한 정부 대응 표준화
- 정부 네트워크 전반에 걸쳐 취약성 및 문제 감지 향상
- 조사 및 수정 기능 개선

1년 이내에 단계적으로 구체화 예정

미 정부의 'SBOM' 의무화

https://www.ntia.gov/blog/2021/ntia-releases-minimum-elements-software-bill-materials

2021년 7월 미국 정부, 소프트웨어 BOM 최소 필수 요소 발표

Supplier Name
주어진 소프트웨어 구성 요소를 제조하는 개인 또는 조직

Component Name
소프트웨어 단위에 부여되는 이름 (공급자가 결정)

Version of the Component
이전 버전에서 소프트웨어의 변경 사항을 지정하는 식별자 (공급자가 결정)

Other Unique Identifiers
SWID(Software Identification) 태그, PURL(Package Uniform Resource Locators), CPE(Common Platform Enumeration) 등과 같은 식별자로 SBOM 소비자가 주요 데이터베이스에서 구성 요소를 찾는 데 도움이 됨

Dependency Relationship
소프트웨어 구성 요소 의존 관계

Author of SBOM Data
저작권 정보 (소프트웨어 공급업체 혹은 개인/그룹)

Timestamp
소프트웨어 BOM이 어셈블된 날짜 및 시간

미 정부의 'SBOM' 의무화

관련 국내기사

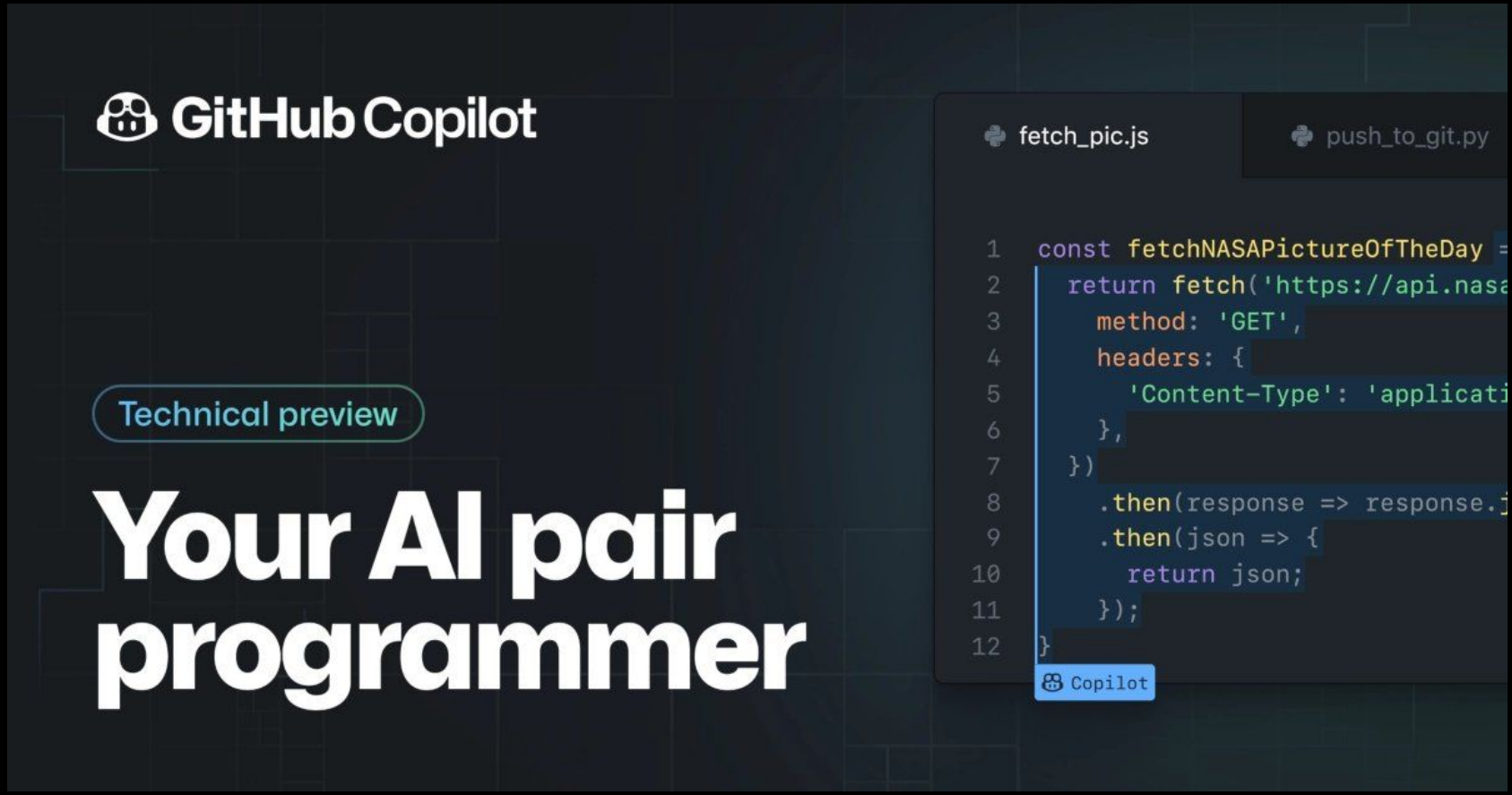
- 낱낱이 SW 요소 공개하라… 美 정부의 'SBOM 의무화'가 미칠 영향
- [박춘식 칼럼] 미국 사이버 보안 행정명령에 대한 단상
- 미국 NIST, 바이든 명령에 따라 특별 관리 소프트웨어의 정의 새롭게 내려
- [기고] 바이든 행정명령, 사이버 전쟁의 시대가 왔다
- '역대급' 해킹 사고로 점철된 바이든 행정부, 보안 강화 위한 행정명령 발표

관련 링크

- Cybersecurity Executive Order and Software Supply Chain Security
- The Minimum Required Elements of a Software Bill of Materials
- Open Source Report Example
- NTIA Releases Minimum Elements for a Software Bill of Materials

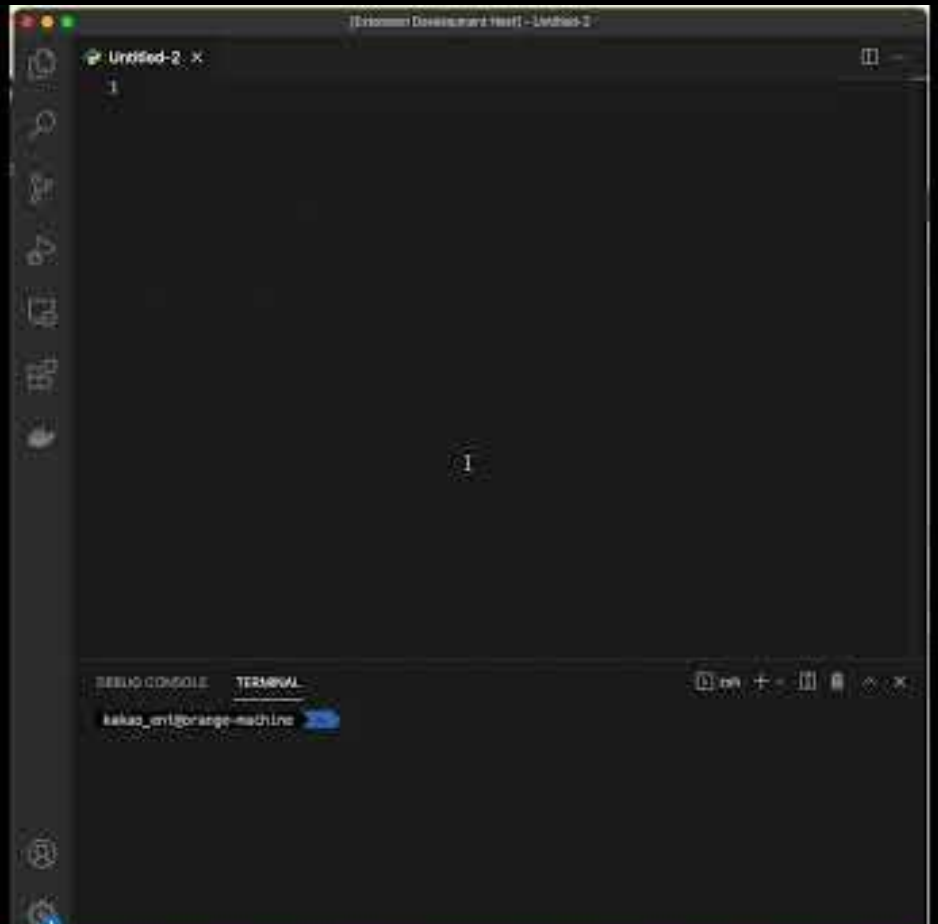
- 오픈소스 저장소 사용 시의 또 다른 위험, '의존성 혼동'이란 무엇인가
- 리눅스 커널 커뮤니티 논란

Github Copilot 오픈소스 라이선스 논쟁



Github Copilot 오픈소스 라이선스 논쟁

Github Copilot 오픈소스 라이선스 논쟁



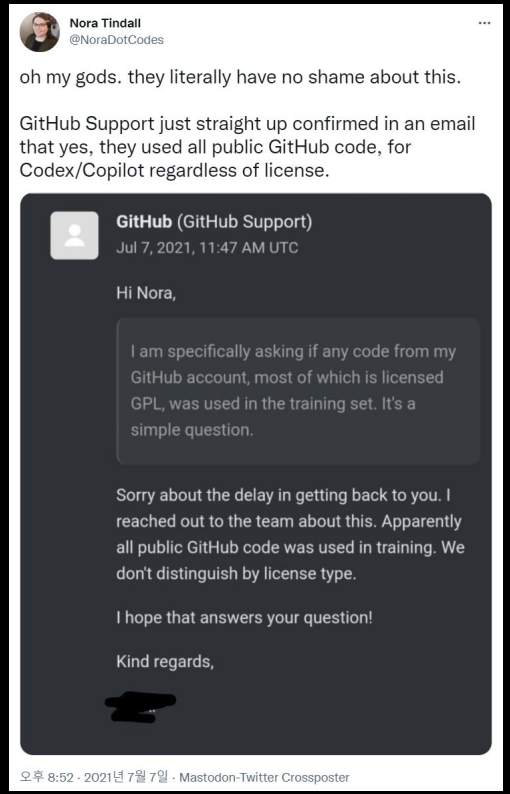
입코딩도 가능!!

GitHub Copilot
+ Kakao i Engine

https://youtu.be/OnxAKXqipsQ

Github Copilot 오픈소스 라이선스 논쟁

Github Copilot은 오픈소스 라이선스에 대한 논쟁을 촉발



Sentry 의 엔지니어링 이사이자 Flask의 제작자인 Leonora Tindall 은 Github 에 있는 그녀의 GPL 코드가 AI 학습에 포함이 되었음을 보여주는 영상을 GIF로 제작하여 트윗

GPL 코드 관련 문의 Github측 답변

- Github 에 공개된 코드는 트레이닝(학습)에 쓰여질 수 있다
- 라이선스 타입을 구분하지 않는다

Github Copilot 오픈소스 라이선스 논쟁

Github Copilot의 법적 의미 분석 - Fossa

Github Copilot 이 저작권 침해를 저지르고 있는가?
- Downing은 GitHub가 GitHub에서 호스팅되는 코드에서 Copilot을 학습함으로써 저작권 침해를 저지르고 있다고 생각 하지 않습니다
- GitHub에서 호스팅되지 않는 코드(따라서 GitHub의 서비스 약관이 적용되지 않음): Downing은 Copilot이 해당 코드를 변형적인 방식으로 사용하는 강력한 사례가 있다고 생각하며, 이는 저작권 침해가 없다는 공정 사용 주장을 뒷받침합니다. 그러나 궁극적으로 법원에서 문제가 해결될 때까지 어떤 식으로든 완전히 확신할 수는 없습니다.

GitHub Copilot 및 라이선스 규정 준수는 어떻습니까?
- Copilot은 완제품과는 거리가 멀고 코드 제안의 복잡성, 길이 및 완전성은 다양한 의견이 있는 것 같습니다.
- 이러한 이유로, 그리고 코드 제안이 저작권 표현의 표준을 충족할 만큼 충분히 독창적이어야 한다는 사실 때문에 Copilot을 사용하면 파생 작업이 생성되는지 여부를 자신 있게 평가하기 어렵습니다.

결론

"지금 Copilot을 사용하는 모든 사람에게 Copilot 제안의 특성에 세심한 주의를 기울이는 코드 작성을 돕도록 주의하겠습니다."라고 Downing은 말합니다. "예를 들어 여전히 주석이 첨부된 다른 소스에서 매우 명확하게 **제안된 코드는 참고해 보는 정도까지 사용하고 제안된 코드를 그대로 사용하지 마십시오.**"

https://fossa.com/blog/analyzing-legal-implications-github-copilot

Github Copilot 오픈소스 라이선스 논쟁

관련 링크

- 자유소프트웨어재단 "깃허브 코파일러, 자유SW 생태계 권리 침해"
- OSI, 오픈소스에서 Copilot은 무엇을 의미하나?
- GitHub Copilot은 저작권자들로부터 비판을 받는다
- Github Copilot은 오픈소스 커뮤니티에서 강한 비판을 받았습니다.
- Fossa, Analyzing the Legal Implications of GitHub Copilot
- Github 쓰면 고소각이라고?