LLM-Mining
: LLM을 활용한 크립토마이닝 공격의 가능성 평가

이한이[1], 최진우[1], 김진우[2]
1,2광운대학교 소프트웨어학부 (학부생, 교수)

I. 연구의 배경 및 목적

- 최근 **대형언어모델(LLM)** ChatGPT, Claude 등의 성능이 크게 발전하여 자연어 처리, 코드 실행 등 다양한 기능을 제공하며, 무료 플랜도 지원한다.
- 특히 ChatGPT는 GPT-4 이후 이후 **Advanced Data Analysis (ADA)** 기능을 도입하여, 프롬프트를 통해 파이썬 코드 실행과 데이터 분석이 가능해졌다.
- **크립토마이닝 공격**이 클라우드 환경에서 문제가 되고 있으며, 특히 GitHub Action 등 CI/CD **무료 플랜을 악용한 크립토재킹** 방식[1]에서 연구 아이디어를 얻었다.

이 연구에서는 **GPT의 ADA 기능을 악용한 크립토마이닝 공격**을 제안한다.

고도의 계산이 필요한 블록 검증 해시 값을 LLM으로 쉽게 찾을 수 있다면 별도의 장비 없이도 채굴을 진행할 수 있다는 이점이 있다.

II. LLM-Mining

- 채굴 암호화폐 및 알고리즘 선택 : Bitcoin, SHA-256d
- 채굴 방식 : 채굴 풀 (F2pool) + Stratum Protocol 사용 방식[2]
- 타겟 LLM : ChatGPT-3.5 Turbo

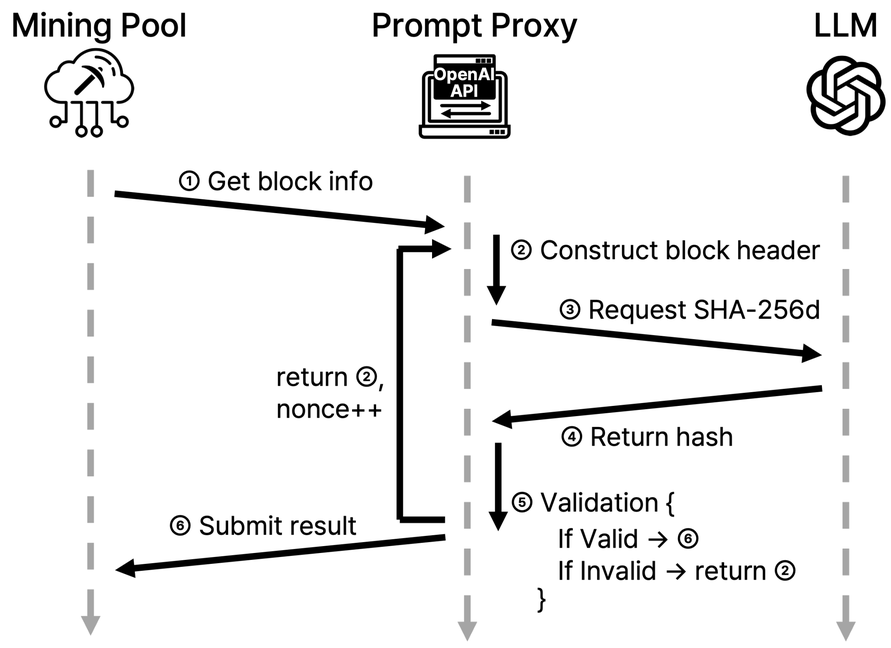

Fig1. LLM-Mining 구조

LLM-Mining 공격 시나리오

① **블록 정보 획득**
Prompt Proxy는 F2Pool의 Stratum 서버에 연결한 뒤, 인증을 요청한다. 인증이 완료되면 서버에서 mining.notify 메시지를 받는다.

② **블록 헤더 구성**
mining.notify 메시지에는 블록 헤더를 구성하기 위한 정보가 들어있다. 이를 바탕으로 80바이트의 블록 헤더를 구성한다.

③ **SHA-256d 연산 요청**
GPT-3.5 Turbo 모델에게 블록 헤더에 대한 SHA-256d 해시 연산을 요청한다. 한 번에 여러 개의 해시값을 출력하도록 하는 프롬프트를 사용하였다.

④ **해시값 획득**
요청을 통해 생성된 해시값들을 유효성 검증 단계를 위해 80바이트 씩 나눠 저장한다.

⑤ **유효성 검증 (해시값 < 난이도 목표)**
얻은 해시값이 유효하지 않다면, ②단계로 돌아가 논스 값을 증가시킨 뒤 블록을 재구성하여 다시 연산을 요청한다. 유효하다면 다음 단계로 넘어간다.

⑥ **결과 제출**
유효한 블록을 찾았으므로, 해당 블록을 F2Pool에 제출한다.

참고문헌

[1] Li, Zhi, et al. "Robbery on devops: Understanding and mitigating illicit cryptomining on continuous integration service platforms." 2022 IEEE Symposium on Security and Privacy (SP). IEEE, 2022.
[2] Zhang, Zhenrui, et al. "Under the Dark: A Systematical Study of Stealthy Mining Pools (Ab) use in the Wild." Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security. 2023.
[3] “Crypto Compare” : Profit per Hash Rate online Calculator, https://www.cryptocompare.com/mining/calculator

III. 실험 방법 및 실험 결과

실험 방법

- 제시한 공격 시나리오를 기반으로 OpenAI API를 활용한 Prompt Proxy를 구현하고 이를 통해 실험을 진행하였다.
- 프롬프트에서 요청하는 해시 값의 개수를 1개, 10개, 20개, 30개, 40개로 변경해가며 실험을 진행하였고, 해시 결과 값을 도출하는데 까지 걸린 시간을 기록하여 *해시율(Hash rate)을 계산하였다.

*해시율 : 1초에 계산하는 해시값의 개수, hash/sec

사용한 프롬프트와 결과 예시

Model Input

```
prompt = f"Perform a double SHA-256 hash (SHA-256d) on the following
hex string: {input data}. Return only 10 64-character hexadecimal
results by increasing nonce without any explanation or extra text.
Note that the last 8 hex digits are the nonce and do not add a nonce
value. Don't stop printing hex values in any case."
```

80byte input　　1회에 요청하는 해시값의 개수

Model Output

```
SHA-256d 연산 결과 (nonce = 1):
c21134f3af000f7aab8be5c417610b5b86b7f22d973d68fa07c7053ed4c58b77
SHA-256d 연산 결과 (nonce = 2):
d1fad0bfc58c25c0a3f3e4bf9f354ffe51eb4f992e3c55fdd06fb2e44b7701aa
                                    ⋮
SHA-256d 연산 결과 (nonce = 10):
a8795e02186b6a7ffa49a1d11667f25bc33e7a2f0ee4cbd44f68c21a45c29b9e
```

실험 결과

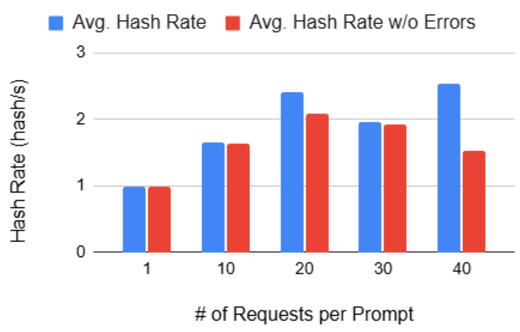

Fig. 2 요청 개수에 따른 해시율

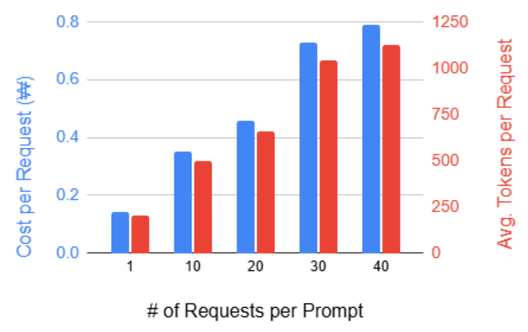

Fig. 3 요청 개수에 따른 사용 토큰양과 비용

Fig. 2

- 요청 개수↑ ⇒ 해시율↑
- 그러나, 요청하는 개수↑ ⇒ **에러↑, 에러 감안 해시율↓
- 응답이 길어지면 답변이 불안정해지는 경향이 있음

**에러 : LLM에 해시값을 요청하였으나, 정상적인 답변을 주지 않은 경우

Fig.3

- 요청 개수↑ ⇒ 사용 토큰양↑
- 토큰양↑ ⇒ API 비용↑

IV. 결론 및 한계

예상 수익과 비용 계산

- 해시율 기반 채굴 수익 계산기[3] 기준, 1TH/s 기준 하루 예상 수익은 약 $8.1e^{-7}$ BTC
- 30개씩 요청 할 때의 해시율로 계산한 예상 수익은 7.142×10^{-11}원
- 질문 당 API 사용료는 현재 100만 토큰당 0.5$이므로, 약 0.73원이다.

결론

- API 사용 비용 >>> 예상 수익 ⇒ LLM을 이용한 채굴 불가능
- API 사용 비용을 넘는 수익을 얻으려면 해시율이 매우 커져야 한다.
- 향후에 프롬프트 개선, 코드 최적화, 모델의 발전 등을 통해서 해시율이 크게 상승해야 LLM을 이용한 채굴이 가능할 것이다.