LLM-Mining: LLM을 활용한 크립토마이닝 공격의 가능성 평가*

이한이[1], 최진우[1], 김진우[2,†]

[1,2]광운대학교 (학부생, 교수)

LLM-Mining: Assessing the Feasibility of Cryptomining Attacks using LLMs*

Ha-Ni Lee[1], Ji-Nu Choi[1], Jin-Woo Kim[2,†]

[1,2]Kwangwoon University(Undergraduate Student, Professor)

요 약

클라우드 환경의 가상머신, 컨테이너 자원을 악용한 크립토마이닝 공격이 증가하고있다. 크립토마이닝은 랜섬웨어 등 기존 공격과 다르게 안정적인 수익을 보장한다는 것이 장점이다. 본 논문에서는 최근 널리 활용되는 거대 언어 모델(Large Language Model, LLM)을 이용한 크립토마이닝 공격 시나리오를 제시하고 가능성을 분석한다. 이를 위해 LLM의 코드 실행 특징을 활용하여 프롬프트를 입력, 블록 해시값을 구하는 시스템을 설계 및 구현하였다. 실험 결과 LLM을 통한 마이닝은 아직 개선의 여지가 많음을 보였다.

I. 서론

최근 ChatGPT, Gemini, Claude와 같은 대형 언어모델(Large Language Model, LLM)의 성능이 급격하게 발전하여 사용자에게 자연어 처리, 텍스트 생성, 질의 응답, 코드 실행 등 다양한 기능을 제공하고 있다. 또한 이를 일정 수준까지 무료로 사용할 수 있는 플랜을 제공하여 더 많은 사용자가 LLM으로부터 혜택을 보고있다.

특히, 가장 널리 사용되는 ChatGPT는 GPT-4 모델 이후 프롬프트를 파이썬과 같은 언어로 변환하여 실행되는 피쳐를 도입하였는데 이를 Advanced Data Analysis (ADA) 라고 한다. 프롬프트를 통해 간단한 연산 및 데이터 분석을 수행할 수 있어 널리 활용되고 있다.

한편으로 최근 크립토마이닝 공격이 클라우드에서 문제가 되고있다. AWS EC2 등에 접근하는 크리덴셜을 훔쳐 퍼블릭 클라우드 자원을 무단으로 사용하거나, GitHub Action 등 CI/CD 파이프라인의 무료 플랜을 악용하여 크립토재킹을 수행하는 공격이 대표적이다.

본 논문에서는 LLM의 코드 실행 특징을 악용하는 크립토마이닝 공격을 제안하고자한다. 크립토마이닝은 블록 검증을 위한 해시 값을 계산하는 작업으로, 주어진 난이도보다 작은 해시 값을 찾아내는 과정이다. 이 작업은 검증이 핵심 요소를 차지하며, LLM을 통해 쉽게 수행할 수 있다면 채굴 과정도 더욱 효율적으로 이루어질 수 있을 것이다.

II. 배경지식

2.1 크립토마이닝

크립토마이닝에서 해시 계산과 제출 과정은 채굴자가 네트워크의 새로운 트랜잭션을 수집하여 블록을 구성하는 것으로 시작되며, 이 블록에는 이전 블록의 해시값, 머클루트, 타임스탬프 등 새로운 블록을 생성하기 위해 필요한 정보들이 담겨있어야 한다. 채굴자는 이 블록 데이터에 논스(nonce)라는 임의의 숫자를 추가한 후, 전체 데이터에 채굴 알고리즘(예: SHA-256d)을 적용하여 해시값을 계산한다. 계산한 해시 값이 목표값(난이도)보다 작을 경우,

* 이 논문은 2024년도 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No. RS-2024-00457937)

† 교신저자(jinwookim@kw.ac.kr)

유효한 블록으로 인정되고 작업 증명(proof of work)이 완료된 것으로 간주된다. 목표값을 만족하지 못하면 논스를 변경하고 다시 해시를 계산하는 과정을 반복한다. 유효한 블록을 찾은 채굴자는 해당 블록을 네트워크에 제출하고, 다른 노드들과의 검증과 합의 과정을 거쳐 블록체인에 추가되면, 채굴자는 트랜잭션 수수료를 보상으로 받게 된다.

2.2 관련 연구

Li 등[1]은 Github Action 같은 CI/CD 플랫폼의 무료 리소스를 악용한 크립토재킹 공격을 탐지하고 완화하는 방법을 제안했다. 본 논문에서는 이를 참고해 LLM 자원 악용 아이디어를 제시한다. Zhang 등[2]은 채굴풀이 사용하는 프로토콜과 채굴 과정을 자세히 분석했다. 해당 연구에서 제시한 프로토콜과 채굴 방식을 참고하여, LLM 기반 채굴을 진행하고자 한다.

III. LLM-Mining

3.1 채굴 알고리즘 선택

초기에는 모네로(Monero)의 채굴 알고리즘인 RandomX를 검토하였으나, 이 알고리즘의 복잡성으로 인해 LLM이 계산을 수행하는데 어려움을 겪었다. 그래서 현재도 광범위하게 사용되고 있으며 상대적으로 구현이 간단한 채굴 알고리즘인 비트코인의 SHA-256d 알고리즘을 선택하게 되었다.

3.2 목표 LLM

여러 LLM에 SHA-256d 알고리즘으로 해시 값 계산을 요청했을 때, ChatGPT의 여러 모델과 Claude는 답변을 제공했으나, Gemini와 CLOVAX와 같은 다른 LLM들은 답변을 거부했다. 무료 제공 답변 한도와 대중적 사용성, API 비용 등을 고려하여 최종적으로 GPT-3.5 Turbo를 채택하였다.

3.3 채굴 방식

비트코인 채굴은 Bitcore를 통한 직접 연결 방식과 채굴풀 이용 방법이 있다. 본 연구에서는 2가지 이유로 채굴풀(F2Pool)을 선택하였다.

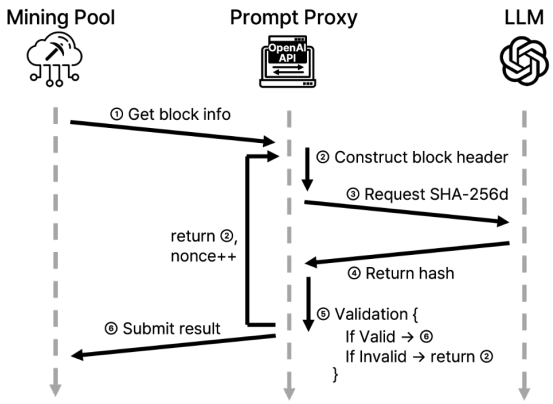

Fig. 1 LLM-Mining overview

첫째, Bitcore는 블록체인에서 직접 정보를 가져와 복잡한 계산으로 머클 루트를 계산해야 하지만, 채굴 풀에서는 간단한 계산으로 머클루트 값을 구할 수 있기 때문에 더 효율적이다.

둘째, 채굴 풀은 낮은 해시율(hash rate)로도 수익을 낼 수 있다. 비트코인 채굴은 경쟁이 치열해 개인 채굴자의 성공률이 매우 낮다. Bitcore로 개인이 채굴하려면 높은 해시율이 필요하지만, 채굴 풀은 많은 참여자가 있어 성공 가능성이 높아 수익성에 유리하다. 따라서 해시율이 낮은 채굴자는 채굴 풀에 참여하는 것이 이득이다.

3.3. 공격 시나리오

OpenAI API를 사용하여 SHA-256d 해시 연산을 수행하는 비트코인 채굴 자동화 시스템을 구현하였다. 주요 목표는 F2Pool로부터 비트코인 블록 헤더를 구성해 유효한 해시를 찾는 것이다. Fig. 1은 LLM을 악용한 마이닝의 구조에 대해 보여주고 있으며, Mining Pool, Prompt Proxy, LLM으로 이루어져 있다.

① 블록 정보 획득: Prompt Proxy는 블록 정보를 얻기 위해 F2Pool의 Stratum 서버에 연결해야 한다. F2Pool에서 제공한 서버 주소로 TCP 소켓을 생성하고, Stratum 프로토콜에 따라 mining.subscribe와 mining.authorize 요청을 보내 인증을 받는다. 인증이 완료되면 서버에서 mining.notify 메시지를 전송한다.

② 블록 헤더 구성: mining.notify 메시지에는 블록 헤더를 구성하기 위한 정보가 들어있다.

이를 바탕으로 버전, 이전 블록 해시, 타임스탬프, 난이도 목표, 논스, 머클루트를 생성하고 리틀엔디안 형식으로 변환해 80바이트의 블록 헤더를 구성한다.

③ SHA-256d 연산 요청: 블록 헤더가 구성됐다면 OpenAI의 GPT-3.5 Turbo 모델에게 블록 헤더에 대한 SHA-256d 해시 연산을 요청한다. 이 때 하나의 프롬프트에 여러 해시값을 요청할 수도 있다.

④ 해시값 획득: 요청을 통해 생성된 해시값(들)을 유효성 검증 단계를 위해서 80바이트씩 나눠 저장한다.

⑤ 유효성 검증: 얻은 해시값이 난이도 목표(difficulty target)보다 작은지 비교한다. 만약 모든 해시값이 난이도 목표에 도달하지 못하면, ②단계로 돌아가 nonce 값을 증가시킨 뒤 블록 헤더를 재구성하고 다시 연산을 요청한다. 만약 난이도 목표를 달성하면 다음 단계로 넘어간다.

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

IV. 검증 및 결론

제안한 시스템을 사용하여 계산한 해시율은 Fig. 2와 같다. 1개만 요청할 때는 약 0.98, 10개일 때는 1.65, 40개일 때는 2.53까지 오르면서 프롬프트 당 해시값의 개수를 늘릴수록 해시율이 커지는 것을 볼 수 있다. 그러나 에러도 같이 높아지기 때문에 이를 감안한 해시율은 더 낮아지게 된다. 에러는 LLM에 해시값을 요청했을 때, 제대로 답변을 주지 않았을 때로 계산하였다.

안정적으로 해시값이 높았던, 한 번에 물어보는 해시값의 개수가 30개일 때 수익성을 예상해보았다. Fig. 3은 한 번 요청했을 때 사용하는 OpenAI API 토큰의 개수와 요금을 나타내고 있다. 요청하는 개수가 많아질수록 사용하는 토큰량이 증가하여, 40개를 요청했을 때는 요청당 평균 0.78원까지 비용이 증가한 것을 볼 수 있다. 채굴 수익 계산기[3]를 사용해봤을 때, 1TH/s 기준 하루 예상 수익은 약 8.1e-7 BTC이다. 현재 비트코인 가격이

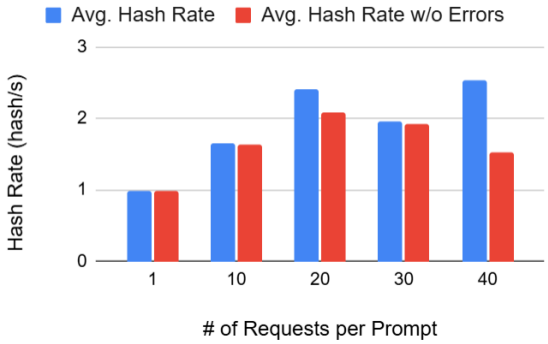

Fig. 2 Hash rates for # of requests per prompt

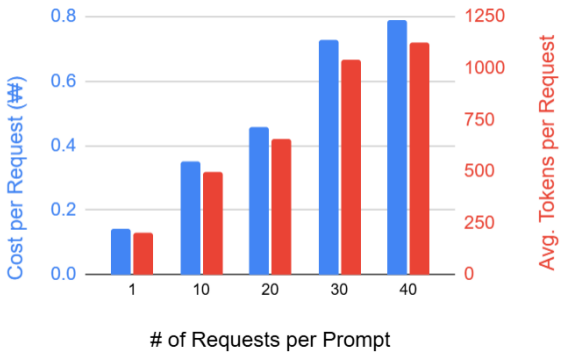

Fig. 3 Cost and avg. tokens per request

약 65,000$로, 한화로는 약 71.42원이다. 최종 예상 수익은 7.142 x 10^{-11}원으로 매우 작은 값이다. 질문 당 API 사용료는 현재 100만 토큰당 0.5$이므로, 약 0.73원이다.

결론적으로 API 사용 비용이 채굴 예상 수익보다 훨씬 크기 때문에, 해시율이 지금에 비해 현저히 커지지 않는 이상 LLM을 이용한 채굴은 어렵다는 것을 확인할 수 있었다. 현재는 어렵지만, 향후에 프롬프트 개선, 코드 최적화, 모델의 발전 등을 통해서 해시율이 크게 상승한다면 LLM을 이용한 채굴이 가능하다고 판단된다.

[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