Loki

개인 거래, 탈중앙식 통신

Kee Jefferys, Simon Harman, Johnathan Ross, Paul McLean

버전 3

2018 년 7 월 13 일

초록

작업 증명/서비스 증명 하이브리드 시스템은 풀 노드 (full node) 의 작업에 경제적으로 인센티브를 제공하는 독특한 방식을 제공한다. Loki 는 이렇게 인센티브가 제공되는 노드를 보강하여 부차적인 개인 라우팅 계층을 생성한다. 스웜 플래깅 (swarm flagging) 이라고 불리는 새로운 방식이 이 두 번째 계층의 최소 노드 기능을 수행하고 관리한다. Loki 는 Monero 소스코드를 수정한 버전으로 개인 정보를 최상으로 보호하면서 모든 거래를 처리할 수 있게 보장한다.

본 백서는 Loki 에 사용된 기술을 요약하여 설명한다. Loki 가 지속적으로 발전하는 한 이 기술도 끊임없이 변화할 수 있기를 바란다. 본 백서의 새로운 버전은 추후 다양한 변동과 업데이트 사항을 반영하여 재발표될 예정이다.

1 서론

디지털 통신과 거래에서 개인 정보 보호에 대한 요구는 날로 커지고 있다. 그러나 동시에 사용자의 데이터도 전무후무한 수준으로 수집, 처리, 거래되고 있다. 사용자의 브라우징 데이터, 이메일 내용부터 신용 등급, 소비 습관까지 그야말로 모든 정보가 세계 대기업과 정부 관계자들 사이에서 수집되고 팔리는 실정이다. Loki 는 이러한 검열에 저항하는 통합 도구 세트를 제공하여 사용자가 개인적으로 거래하고 소통할 수 있게 한다.

비트코인은 개인 정보 보호를 약속하며 등장했지만 지금은 정보의 추적 가능성이 그 어느 때보다 크다. Chainalysis 와 BlockSeer 같은 기업은 비트코인의 투명한 블록체인 구조를 활용하여 특정 거래를 추적하고 확인한다 [1]. Loki 가 제작 기반으로 삼은 플랫폼은 지금까지 사적이면서도 가장 안전한 거래 네트워크를 스스로 구축한 Monero 다 [2]. 그러나 Monero 에도 본질적인 문제는 있다. Monero 거래는 비트코인 거래 그 자체보다 중요도를 중시하여 순서를 정하며 대역폭, 프로세싱, 디스크 공간 등에 대한 엄청난 요건도 갖추고 있다. 그래서 네트워크가 성장하면서 Monero 노드 운영에 부담이 커졌고 네트워크 기여에 대한 인센티브나 보상도 제공하지 못하고 있다.

이는 노드의 운영 비용은 커지고 작업을 수행하고도 보상은 받지 못하는 현상을 낳았다. 그래서 서비스 노드 (Service Nodes) 라고 불리는 노드 보상 원리를 소개함으로써 노드 연산자에게 경제적인 인센티브를 제공하고 이러한 문제를 경감하고자 한다.

서비스 노드는 적절한 보상이 주어지면 개인 정보를 보호하는데 중점을 두고 다양한 기능도 제공할 수 있다. 서비스 노드 네트워크에서는 주로 사용자가 익명으로 데이터 패킷을 전송하고 수신할 수 있다. 이러한 개인 통신은 Sybil 에 대항하는 새로운 믹스넷 (mixnet) 에서 각 서비스 노드가 릴레이 역할을 함으로써 가능하며 Tor, I2P 와 비슷한 특성을 가진다 [3][4]. 더 나아가 이 새로운 통신 네트워크는 Loki Messenger 라 불리는 탈중앙식 말단간 암호화 메세징 서비스의 근간으로 사용될 것이다. 이 서비스를 이용하면 사용자는 타사에게 의존하지 않아도 되고, 양당사자는 동시에 온라인으로 접속하고 있지 않아도 직접 소통할 수 있다.

Loki 는 사립 거래소의 탄력적인 매개체이자 탈중앙식 익명 인터넷 서비스이기도 하다.

2 기본 한도

Loki 난이도 임계치 (블록시간)	120 초
난이도 알고리즘	Zawy LWMA [5]
해시 알고리즘	CryptoNight Heavy
타원 곡선	Curve25519 [6]

3 CryptoNote 요소

풀 노드 인센티브 시스템은 물론 그 어떤 암호화폐에서든 실행될 수 있지만, Loki 는 Monero 소스코드를 이용하고 있다. Monero 는 거래에 담고 있는 개인 정보 보호 수준이 높기 때문이다. 또한 Monero 는 CryptoNote 프로토콜의 진화형이며 링 서명 (ring signature), 스텔스 주소 (stealth address), 링 CT(RingCT) 를 이용하고 있다. 이로써 사용자가 거래에 서명하고 타당성 있는 명분을 유지하면서도 거래의 금액은 모호하게 만든다 [7].

Loki 생태계가 개인 정보를 보호하기 위해서는 내부 재정 시스템을 뒷받침하는 교환의 수단을 제공하는 것뿐 아니라 Loki 의 개별 단계를 통해 거래가 성사될 때 시간 분석의 리스크도 최소화해야한다. 예를 들어 거래 서비스 1 단계를 이용할 때 사용자는 2 단계에서도 개인 정보를 보호한다는 보장을 절대 놓쳐서는 안된다.

3.1 링 서명 (Ring Signatures)

링 서명에서는 이용 가능한 서명자들을 거래로 연결하지만 실제 서명을 전송하는 사람은 한 명뿐이다. 또한 Loki 는 링 서명으로 거래 출력값의 실제 거래 내역을 감춘다. 링 서명은 Loki 의 모든 거래에 필수적으로 적용된다. (블록 보상 거래 제외) 또한 Loki 블록체인에서는 링 크기를 10 으로 고정하여 실시한다. 즉, 각 입력값은 실제 출력값을 포함하여 1 에서 10 가지 출력값을 낼 수 있게 된다.

3.2 스텔스 주소 (Stealth Addresses)

Loki 는 스텔스 주소를 활용하여 실제 수신자의 공개 키와 거래가 절대 연결될 수 없게 한다. Loki 거래가 전송될 때마다 일회용 스텔스 주소가 생성되어 자금이 해당 주소로 전송된다. Diffie-Hellman 키 교환 방식을 활용하여 거래의 수신자는 이 스텔스 주소를 위해 개인 키를 계산할 수 있다. 이에 자금의 주인은 공개 키를 노출하지 않고 자금의 소유권을 증명할 수 있는 것이다 [8]. 스텔스 주소는 거래에서 수신자를 보호하며 Loki 의 개인 정보 보호 기능의 핵심이라고도 할 수 있다.

3.3 링 CT(RingCT)

링 CT 는 Monero Research Lab 에서 거래 금액을 숨기는 방법으로서 처음 제안되었다 [9]. 현재 링 CT 는 범위 증명에 사용되고 있다. 이는 Pedersen 서명 방식을 보완하여 거래 금액을 0 와 2^{64} 사이로 전송할 수 있다고 증명했다. 즉, 0 이상의 금액만이 전송되고 전송 시에는 실제 금액을 노출하지 않도록 보장한다. 최근 많은 암호화폐들이 BulletProof 프로토콜로 기존 링 CT 의 범위 증명 방식을 대체하고자 하는 움직임이 있다. 왜냐하면 이로써 거래 규모를 눈에 띄게 줄일 수 있기 때문이다 [10]. Loki 는 BulletProof 를 활용하여 노드가 보관, 전달하는 정보를 줄이고 확장성을 확보할 것이다.

4 서비스 노드 (Service Nodes)

Loki 는 CryptoNote 프로토콜 (7 참고) 에 새로운 변화를 적용하지만 Loki 의 네트워킹 기능과 확장성은 서비스 노드, 즉 인센티브를 받는 노드의 집단으로 실현된다. 서비스 노드가 작동하려면 노드 연산자는 Loki 의 엄청난 양에 시간 가치를 추가하고 최소 대역폭과 네트워크 저장소를 제공해야한다. 그 서비스에 대하여 Loki 서비스 노드 연산자는 각 블록으로부터 블록 보상의 일부를 수신한다.

결과적으로 네트워크에서는 Sybil 공격에 대항할 수 있는 시장 기반 저항성을 제공한다. 특히 현존하는 믹스넷과 개인 정보 보호 중심 서비스와 함께 다양한 문제를 제기한다. 이러한 저항성은 수요와 공급의 상호 작용에 기반을 두고 있기 때문에 한 명의 참여자가 지나치게 큰 지분을 가지고 Loki 의 개인 정보 보호 서비스 2 단계에 부정적인 영향을 주는 것을 방지한다. DASH 는 암호화 경제에서야말로 Sybil 공격에 대항하는 네트워크를 이끌어낼 수 있다는 이론을 처음 제시했다 [11]. 공격자가 Loki 에 축적되고 순환하던 공급이 줄어들면, 수요 측 압박을 받게 되고 Loki 의 가치가 올라간다는 것이다. 이러한 현상이 지속되면 Loki 를 추가적으로 구매하는 비용이 높아지고 공격을 시도하려 해도 엄두내지 못할 정도로 비싸진다.

Loki 는 이렇듯 경제적 보호라는 목적을 달성하기 위해 유통되는 공급량을 억제하도록 고무한다. 특히 방출 곡선과 담보물 요건은 유통되는 공급량을 충분히 고정시키고 합리적인 수익이 연산자에게 제공되어 Sybil 공격에 대항할 수 있게 설계해야한다.

4.1 블록 보상

Loki 에서 블록 보상은 작업 증명으로 이루어지며, 이 방식은 블록을 생성하고 거래를 주문한다는 목적에 부합하고 연구도 충분히 이루어진 시스템이다. 채굴자는 거래를 수집해서 작성하여 블록으로 담고 수수료를 받는다. Loki 의 합의 규정에 따라 각 블록은 보상 출력값이 여럿 있으며 그 중 하나만이 채굴자에게 돌아간다.

채굴 보상:
거래 수집 비용과 블록 보상의 45% 는 블록을 만드는 채굴자에게 지급된다.

서비스 노드 보상:
각 블록의 두 번째 출력값 (전체 보상의 50%) 은 하나의 서비스 노드로 간다. 또는 릴레이가 선택되면 두 개의 서비스 노드로 돌아간다. (6.3 참고) 서비스 노드는 시간에 근거하여 보상된다. 마지막으로 보상을 받았던 시점 (또는 등록한 시점) 기준이며 더 오래 기다린 노드를 먼저 선택한다. 서비스 노드가 네트워크에 등록될 때마다 대기열의 마지막 자리를 차지한다. 서비스 노드가 좋은 서비스를 유지하고 스웜 플래그가 대기열에서 제거하지 않는 이상 (7.3 참고) 상위 순서로 이동한다. 대기 순서의 앞쪽에 위치한 노드는 보상을 받을 자격이 되고, 보상을 받은 노드는 다시 대기열의 마지막 자리로 이동하고 다시 순위를 밟아 올라가야 한다.

관리 보상:
블록 보상의 마지막 5% 는 관리 작업 (9 참고) 에 분배된다. 그 중 3.75% 는 Loki 재단 주소로 전송되어 결론적으로 각 블록이 되고 나머지 1.25% 는 재정을 지원하는 블록의 출력값으로 보유한다. (9.2.3 참고)

4.2 증명 가능한 담보 보증

서비스 노드는 담보 금액 요건을 맞추고 있다고 네트워크에 증명해야한다. Loki 설계상 내재된 개인 정보 보호 기능으로 이를 증명하기는 어려운데, 특히 공개 주소의 잔액을 감시하거나 발신되는 거래를 보기 위한 키를 이용할 수 없다.

Loki 는 타임락 설정된 출력값을 새롭게 이용하여 Loki 코인에 블록체인이 지정된 블록의 높이까지 이를 때까지 시간 제한을 설정한다. 이 지정된 높이에 이를 때까지는 Loki 네트워크에서 타임락이 설정된 출력값 (time-locked output) 들을 무효 처리한다. Loki 는 이 절차를 활용하여 특정 서비스 노드가 해당 금액을 보유하고 있다고 증명하기 때문에 담보 금액을 속이는 행위를 방지한다.

연산자가 서비스 노드로 등록하려면 요구된 금액이 있고 타임락이 설정된 출력값을 생성한다. 최소 21,600 블록이 경과한 뒤 (약 30 일) 락을 해제한다. 서비스 노드는 거래의 공백 필드에 서비스 노드 보상을 수령해도 되는 Loki 주소를 포함한다. 이 주소는 스웜 투표 같은 서비스 노드 연산작업을 위한 공개 키로도 사용된다. 다만 월렛은 이 서비스 노드 등록 거래들을 혼합으로 사용하는 것을 피하기도 한다. 왜냐하면 실제 금액과 목적지가 노출되어 거래에 추가적인 익명성을 제공하는데 유용하지 않기 때문이다.

각 노드가 서비스 노드 네트워크에 진입하기 전에 다른 노드는 보증 요건이 줄어듦에 따라 개별적으로 명시된 노드의 담보 금액 결제가 요구된 금액과 맞는다는 점을 입증해야 한다. 담보 금액 거래가 30 일 이후 만료되더라도 월렛은 사전 동의가 된 자동 재보증 기능을 갖는다.

5 Lokinet

어니언 라우팅 (Onion Routing) 프로토콜은 분산된 네트워크로 사용자가 일종의 터널을 만들 수 있게 허용한다. 이 때 여러 노드를 홉 (hops) 으로 사용하기 때문에 데이터 패킷의 목적지와 출발지가 모호하다. Loki 네트워크의 서비스 노드는 대기 시간이 짧은 어니언 라우팅 프로토콜을 작동시키고 완전히 분산된 오버레이 네트워크 (overlay network), Lokinet 을 구축한다. 네트워크는 신뢰받는 당국에 의존하지 않고 온전히 블록체인에서 이끈다. 사용자는 개인 서비스 노드를 연결하고 패킷을 라우팅 하기 위해 양방향 경로를

만들 수도 있다. 네트워크는 내부적으로 호스트된 서비스, SNApps(6.2 참고) 에 접근하는데 이용될 수 있다. 사용자는 서비스 노드 출구 기능을 활용하여 IP 주소 노출 없이 외부 인터넷을 브라우징할 수 있다. (6.3 참고)

5.1 낮은 대기 시간의 익명 라우팅 프로토콜 (LLARP)

서비스 노드를 위한 모든 응용프로그램의 기저에는 익명 라우팅 프로토콜이 있다. 이것이 바로 각 서비스 노드간 통신하는 방식이다. 이에 Loki 는 새로운 라우팅 프로토콜 LLARP(Low Latency Anonymous Routing Protocol)[12] 를 소개한다. 이는 토르 (Tor) 와 I2P 의 하이브리드로 기존 라우팅 프로토콜과는 다르게 적절한 속성을 추가적으로 제공한다. LLARP 는 Loki 서비스 노드 네트워크 기반에서 운영되도록 특별히 설계되었으며 모든 LLARP 를 최적화할 때도 이 구조를 염두에 둔다. LLARP 의 목적을 이해하려면 기존 라우팅 프로토콜을 분석하여 LLARP 가 그 이상으로 어떤 점을 개선할 수 있는지 고려하는 것이 가장 효과적이다.

어니언 라우터 (Tor)

최근 몇 년간 토르는 가장 인기 있는 익명 믹스넷으로 이용되고 있다. 토르 네트워크는 검열에 높은 수준으로 저항하고 이를 유지하여 인터넷에서 개인 정보를 보호할 수 있는 훌륭한 도구임을 입증했다. 하지만 토르는 탈중앙식 네트워크가 아니며 오히려 계층적 네트워크에 가깝다. 토르는 토르 재단 (Tor Foundation) 과 친밀한 관계의 자원봉사자들이 이끄는 집단이 운영하는 중앙식 서버, 즉 디렉토리 권한 집단에 의존한다 [13]. 이 디렉토리 권한 집단은 두 가지 주요 기능을 맡고 있다. 첫째, 그들은 네트워크의 노드 상태에 대한 신뢰받는 보고자로 활동한다. 토르 이용자 (또는 릴레이) 가 처음으로 네트워크에 연결되면 이들은 디렉토리 권한 집단 5 개 중 하나로 연결될 수 있다. 이 권한 집단은 데이터를 쉽게 변경할 수 없게 코딩되어 있다. 그리고 사용자나 릴레이로부터 합의 (consensus) 라고 불리는 파일을 받는다. 이 파일은 토르 네트워크 내 현재 해당 작업에 있는 모든 릴레이 (relay), 가드 노드 (guard node), 출구 노드 (exit node) 목록 (연결 브릿지 제외) 을 제공한다. 둘째, 디렉토리 권한 집단은 각 릴레이가 네트워크에 제공할 수 있는 대역폭도 측정한다. 그들은 이 정보를 이용하여 릴레이들을 카테고리에 맞게 분류하고 노드가 릴레이, 가드 노드, 출구 노드 중 어떤 역할을 할 수 있는지 결정한다.

이러한 고도의 중앙화는 토르가 실패에 취약할 수밖에 없는 환경을 만든다. 2014 년 토르는 디렉토리 권한 집단의 서버를 급습한다 [14] 는 믿을 만한 소스로부터 위협을 받기도 했다. 미국과 독일 또는 네덜란드에 있는 디렉토리 권한 집단 사무소가 폐쇄된다면 10 개 중 5 개의 디렉토리 서버도 충분히 폐쇄될 수 있다. 이는 물론 토르 네트워크가 불안정해지는 결과를 가져올 것이며 새로운 릴레이도 네트워크와 상호작용할 수 있는 능력을 잃게 된다.

토르는 TCP 만 허용하기 때문에 그 통신 수단 역시 제한적이다. 토르 기반의 IP 는 가능하지만 UDP 기반 프로토콜 (VoIP 등) 을 위한 지원은 부족한 실정이다.

보이지 않는 인터넷 프로젝트 (I2P)

I2P 는 믹스넷 구조에 다양하게 접근하면서도 높은 수준의 신뢰 민첩성을 유지한다. 이는 신뢰받는 디렉토리 권한 당국이 아니라 분산 해시 테이블 (Distributed Hashing Table, DHT) 을 참고하여 네트워크 상태를 확인함으로써 가능하다 [15]. I2P 는 프로토콜의 상호작용을 방대하게 지원하면서 TCP 와 UDP 트래픽을 모두 허용한다. 그러나 I2P 는 꾸준한 발전 과정을 거치지 않았고 시간이 지남에 따라서 암호학적 용도 등 기술적 부채를 축적해왔다. I2P 는 타원 곡선 작업과는 다르게 2048 bit ElGamal 를 이용하기 때문에 암

호화와 해독의 속도가 느리다. ElGamal 에서 바꿔야하는 계획이 I2P 로드맵에 존재하는 경우 진행 속도가 느렸던 것이다.

I2P 는 출구 노드에 대한 실질적인 지원도 부족하여 네트워크의 트래픽 대부분이 Eepsites 라고 불리는 호스트된 웹사이트에서 내부적으로 접근하고 있다. 이는 I2P 네트워크가 사용자에게 접근할 수 있는 능력을 대폭 감소시켰다. 특히 이 사용자들은 더 광범위한 인터넷에 접근하기 위해 익명화된 네트워크를 이용한다.

또한 I2P 가 구축된 방식은 네트워크에 연결된 대다수의 사용자도 라우터가 된다는 것을 의미한다. 즉, 최종 네트워크가 신속한 경로를 만들기 위한 대역폭이 부족하게 되는 문제를 낳는다. 믹스넷의 네트워크 속도는 각 회로의 가능한 최소 노드로 막히게 되며, 활동이 적은 사용자들이 I2P 의 릴레이가 되어버리기 때문에 전체적인 처리 능력도 감소할 수 있다.

마지막으로 I2P 는 회로 교환 방식이 아니라 패킷 교환 방식의 네트워크를 제공한다는 점에서 토르와 다르다. 트래픽 전체가 통과하는 단일 장기 터미널을 구축하는 대신에 I2P 는 각 통신 패킷이 네트워크를 통해서 서로 다른 루트를 선택할 수 있는 다중 회로를 구축한다. 이로써 I2P 는 네크워크의 혼잡과 노드의 실패에도 투명하게 전송되는 능력을 갖는다.

I2P 와 토르 모두 Sybil 공격을 완전하게 차단할 수 있는 조치를 취하고 있지는 않다. 동기가 충분한 공격자에게 시간과 자본까지 제공되어 대량의 릴레이를 구매할 수 있다면 사용자 개인 정보 보호를 약화시키는 시간 분석도 진행할 수 있다. 이러한 분석의 효과로 공격자가 작업할 수 있는 출구 노드, 릴레이, 가드 노드가 더 많이 생긴다 [16]. 토르와 I2P 는 노드의 연산에 자신의 시간과 자금을 사용하고 있는 자원봉사자들에 의해서만 완전하게 운영되고 있다. 그래서 더 신뢰가 가는 서비스도 이타심보다는 경제적인 인센티브를 기반으로 한 네트워크에서 제공할 수 있고 공격에도 큰 회복력을 가질 수 있다고 예측한다.

LLARP

LLARP 는 운영상 디렉토리 권한 집단을 이용할 필요 자체가 없다. 대신 블록체인 채굴 거래에서 구축된 DHT 에 의존한다. 이 때 서비스 노드는 네트워크에서 라우터 같은 역할을 한다. DHT 에서 대역폭은 관리되거나 기록되지 않는다. 대역폭 측정 및 분류는 스웜에 근거하고 (6.1.1 참고) 각 노드를 평가하여 네트워크에 적절한 대역폭을 제공할 수 있는지 판단한다.

개방형 시스템 상호접속 (OSI) 모델에서 LLARP 는 익명 네트워크 계층을 제공하는 목적만 갖고 있다. 즉, 더 넓은 범위의 인터넷 프로토콜을 지원하고 출구 노드가 사용자 데이터그램 프로토콜 (UDP) 트래픽을 통해 전달한 파일 기술자 보관의 비용을 최소화한다 [17]. 추가적으로 LLARP 는 터널 기반의 라우팅보다 패킷이 전환된 라우팅을 선택한다. 이로써 더 나은 부하 균형과 네트워크를 통틀어 중복 역시 줄일 수 있기 때문이다.

Lokinet 의 최종 사용자는 패킷을 전송하도록 기대되거나 허용되지 않으며, 이는 Lokinet 이 서비스 노드 작업에 요구되는 경비가 커서 Sybil 공격에 훨씬 덜 취약하고 노출도 줄일 수 있다는 뜻이기도 하다.

6 Loki 서비스

서비스 노드의 각 연산자는 채굴자가 하드웨어에 투자하듯이 서비스 노드 운영 시작과 함께 Loki 코인을 동결한다. 이렇게 동결된 자본에는 두 가지 목적이 있다.

1. 모든 서비스 노드 연산자는 네트워크의 성공 여부에 큰 관련이 있다. 한 서비스 노드 연산자가 네트워크에 기여하는 바가 부정적이거나 부정직한 행동을 하는 경우 이는 네트워크 내부에서 스스로의 역할을 약화시키고 평가 절하되는 리스크를 지는 것이다.
2. 공격적인 방식을 더 강화할 수 있는 기회를 제공한다. 즉, 네트워크가 부정직한 노드가 보상을 받지 못하도록 효과적으로 제한할 수 있다면, 부정직한 노드는 보상 손실 및 담보물의 락 시간 (lockup time) 유지라는 기회 비용을 인정해야한다.

상기 지적한 사항들을 사실로 받아들인다면, 저조한 작업을 수행하는 노드 (7.3 참고) 를 공격적으로 처벌할 수 있다. 그리고 블록체인의 상태에 대한 합의에 이르거나 오프체인 노드 (off-chain node) 의 특정 행동 (6.1.1 스웜 참고) 을 강화하도록 의문을 제기할 수 있는 서비스 노드 집단도 생성할 수 있다. Loki 에서 이러한 행동은 네트워킹 및 저장소 활동과 연관성이 있다. 또한 오프체인 행동은 사용자 대면 응용프로그램의 백엔드가 된다. 그리고 Loki 서비스라고 불리는 바람직한 특징을 보강한다.

6.1 Loki 메신저

Loki 네트워크에서 처음으로 개발되어 사용될 Loki 서비스는 탈중앙식 종단간 암호화 개인 메시징 응용프로그램인 Loki 메신저다.

종단간 암호화 메시징 응용프로그램은 사용자가 메시지 내용을 공개하지 않고 상대방에게 전송할 수 있는 플랫폼이다. 하지만 그 서버는 중앙에서 통제하기 때문에 타겟이 되어 차단되거나 정지당할 수도 있다 [18][19]. 이러한 중앙집중식 서비스 모델은 소통 당사자간의 익명성에 큰 리스크를 부여해왔고 종종 신원 증명을 하는 전화번호나 기타 정보를 요구했다. 특히 사용자의 IP 주소를 통해 직접 연결하는 과정에서 해당 정보를 얻어냈다. 이 정보는 데이터 유출이나 법적 절차를 통해 서버에서 추출될 수 있으며 사용자에게 불리하게 이용될 수 있다. Loki 네트워크에서는 서비스 노드 구조를 보강함으로써 시중에서 인기 있는 중앙식 암호화 메시징 앱과 비슷한 서비스를 제공할 수 있다. 예를 들어 Signal 은 개인 정보를 높은 수준으로 보호하면서도 검열은 반대한다.

6.1.1 메시징 라우팅 (Messenger Routing)

Loki 네트워크의 메시징 라우팅은 사용자의 온라인, 오프라인 상태에 따라 달라진다. 두 사용자가 모두 온라인일 때는 메시지가 서비스 노드에 저장될 필요가 없기 때문에 상대적으로 더 높은 대역폭의 통신이 이루어질 수 있다.

Loki 에서 공개 키는 장기적인 암호화 키와 라우팅 주소, 두 가지 역할을 모두 한다. 대부분 이 키는 대역 외에서 교환해야하며 그래야 중간자 공격에 대항하여 보호할 수 있다. 이러한 교환은 직접 또는 기타 안전한 모드로 이루어진다. (6.1.2 참고)

온라인 메시징

Alice 가 Bob 의 공개 키를 알고 나면 그녀는 그가 온라인이라는 사실을 알고 그와 소통할 수 있는 경로 만들기를 시도한다. 그 어떤 서비스 노드든 Alice 는 이 과정을 DHT 를 문의함으로써 진행한다. 그리고 Bob 의 공개 키와 상응하는 소개자 세트 (introduction sets) 를 얻는다. LLARP 에서 소개자 세트는 각 사용자가 유지하고 있는 소개자 *(introducers)* 목록에 있다. 그리고 이 소개자를 통해서 경로가 만들어진다. Bob 의 소개자와 함께 Alice 는 이제 무작위 서비스 노드를 3 가지 선택하여 중개 홉스처럼 사용한다. 이는 그녀의 출발지와 목적지 (Bob 의 소개자) 사이를 잇는다. 이제 Alice 와 Bob 이 메시지를 전송할 수

있는 경로가 마련되었다. 그러면 OTR 을 사용하여 (6.1.2 참고) 인증 과정이 제대로 이루어지면 Alice 와 Bob 은 개인 정보를 최대한 보호하면서 소통할 수 있다.

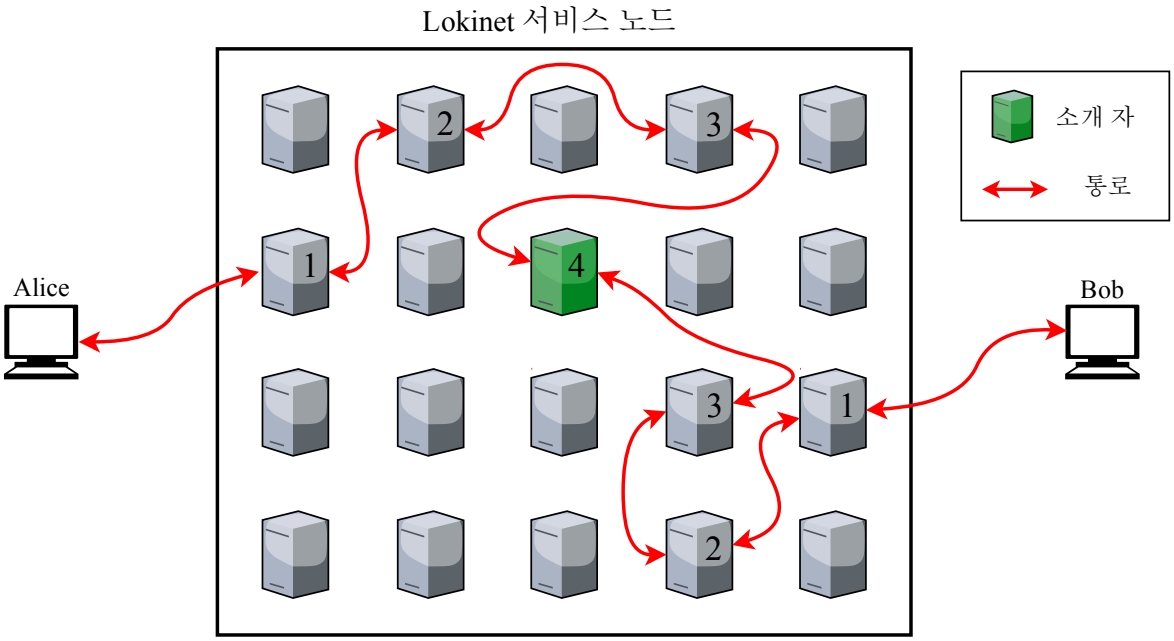

그림 1: *Alice* 와 *Bob* 이 무작위 서비스 노드를 이용하여 네트워크에 경로를 만들고 소통하는 온라인 라우팅의 간략 버전.

오프라인 메시징

Alice 가 Bob 으로부터 회신을 받지 못하면, 오프라인 메시징 절차를 진행할 수 있다. 오프라인 라우팅은 우편 서비스 스웜 (Postal Services over Swarm, PSS) 의 수정된 버전을 이용한다 [20]. 스웜은 서비스 노드의 논리 집단으로 공개 키와 블록의 해시를 모두 기반으로 한다. 이 때 블록은 채굴 거래가 처음 나타난 블록을 지칭한다. 각 스웜은 스웜 ID 가 있으며 9 개의 노드로 이루어져있다. Alice 는 Bob 에게 메세지를 보내기 위해서 그의 공개 키를 이용하여 Bob 의 스웜을 계산한다. 이 정보로 Alice 는 네트워크를 통해 익명으로 해당 스웜의 무작위 서비스 노드로 메시지를 전송할 수 있다. 서비스 노드가 그 스웜에 정해진 특정 메시지를 받으면 그 노드는 해당 메시지를 스웜 내 나머지 8 개 노드에 분배해야한다. 또한 모든 노드는 허용된 시간 기본값 (Time-to-live, TTL) 에 따라 메시지를 저장하도록 요구된다. (8.3 참고) Bob 이 온라인 상태가 되면 그는 그의 스웜 중 2 개의 노드에 의문을 제시하고 그가 해독할 수 있는 메시지를 확인한다. 오프라인 메시징은 각 메시지에 첨부된 간단한 작업 증명으로 스팸을 방지하고 보호된다. (7.2 참고)

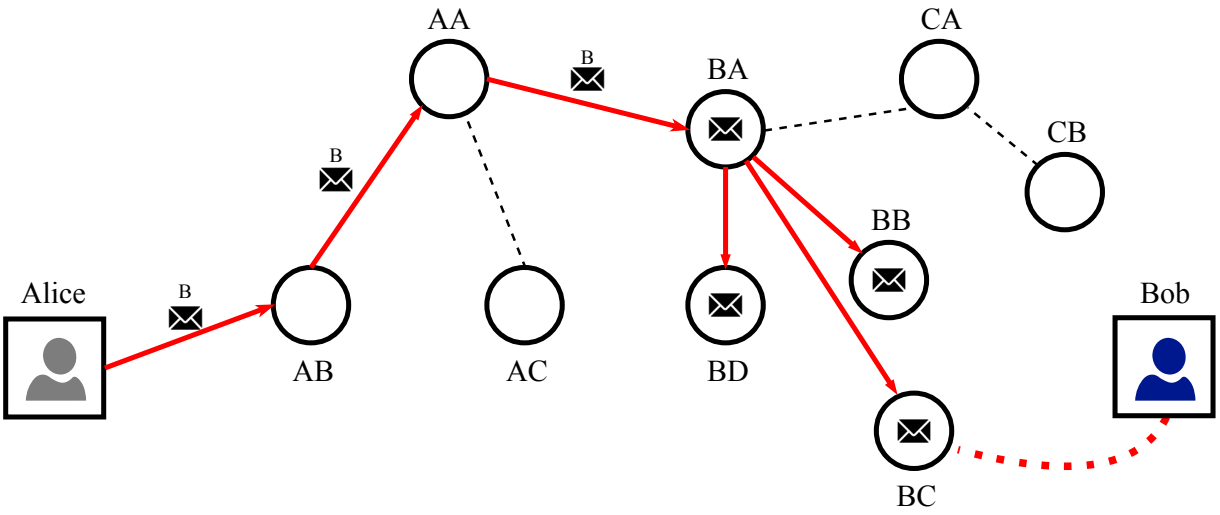

그림 2: *Alice* 는 *Bob* 에게 메시지를 보냈고 *Bob* 에게 할당된 스웜은 B 다. *Bob* 이 온라인 상태가 되면 그는 그의 스웜 내 무작위로 선택한 노드에게 의문을 제기하고 *Alice* 의 메시지를 받는다.

6.1.2 메신저 암호화와 인증

메시지 체인이 만들어지면 Loki 메신저는 완전 순방향 비밀성 (Perfect Forward Secrecy, PFS) 과 부인가능한 인증 (Deniable Authentication, DA) 을 강화한다. PFS 와 DA 는 비공개 (Off The Record, OTR) 메시징 프로토콜의 주요 개념이다 [21]. Signal 과 WhatsApp 같은 중앙 통제식 서비스는 암호화 기능을 이용하여 OTR 보호를 유지한다. Loki 는 기존 Tox 프로토콜에서 그 OTR 시행 모델을 참고했으며, 이는 분산식 P2P 인스턴트 메시징 프로토콜이자 고도로 검증된 NaCl 라이브러리를 이용한다 [22].

PFS 는 장기 키가 노출되는 공격에 대항할 수 있다. 새롭게 공유된 암호 키는 각 세션에 이용되며, 만약 단일 세션의 키가 공개되면 전체 메세지 체인이 제대로 작동한다. 타사가 메세지 체인의 암호를 해독하고자 하면 각 개인 세션에서 키를 얻어야한다. PFS 는 프리티 굿 프라이버시 (Pretty Good Privacy, PGP) 같은 현존하는 암호화 방식과 비교했을 때 Loki 메신저가 타협하기 매우 어렵도록 만든다. PGP 에서는 전체 메시징 체인을 타협하도록 장기 키 쌍 하나만 요구된다.

DA 는 두 당사자가 각자 새로운 메시지의 전송자라는 사실을 서로에게 증명하는 능력을 의미한다. 그러나 타사는 그 어떤 메시지도 누가 진짜 발신자인지 확인할 수 없다. DA 를 이용할 때는 메시지 인증 코드 (Message Authentication Codes, MAC) 가 각 세션 이후 발급되며, 타사는 그럴듯하게 메시지를 생성하여 발신자의 공개 주소에서 고안된 것처럼 보이게 할 수 있다. 이러한 절차가 올바르게 이행되면 그 어떤 타사도 특정 메시지의 발신가가 실제 발신자인지 증명하는 것이 불가능하다.

사용자 인증

사용자의 인증은 중간자 공격에 대항하는 보호 방식으로 매우 중요한 단계다. 예를 들어 Bob 이 Alice 로부터 메시지를 기다리는데 그녀의 공개 키를 아직 모른다면, 제 3 자 (Eve) 가 Bob 에게 Alice 인 체하고 메시지를 보낼 수 있기 때문이다. 그래서 사용자는 서로 개인 정보를 공유하기 전에 인증 절차를 거쳐야한다.

Loki 메신저는 Pidgin 이나 기타 OTR 메시징 서비스처럼 사전 공유키 (Pre-Shared Key, PSK) 인증을 이용한다. 사용자는 PSK 를 만들 때 여러 가지 옵션이 있다. 대역 외 키나 Loki 메신저의 PSK 에 동의할 수 있다. 이 때는 타사에서 정답을 알 수 없는 질문을 상대

방에게 던짐으로써 PSK 를 구축한다. Loki 는 Pidgin 암호화 인증 플러그인을 수정하여 PSK 인증을 실행할 것이다 [23].

6.2 SNApps (서비스 노드 응용프로그램)

SNApps(Service Node Applications) 의 기능은 많은 인기를 끌었던 소위 토르의 비공개 서비스와 비슷하다. 그들은 사용자들에게 믹스넷 환경에서 상호 작용할 수 있는 방식을 제공한다. 특히 외부에서 호스팅된 컨텐츠에 접속할 때보다 훨씬 더 높은 수준의 익명성을 제공한다. SNApps 로 사용자는 마켓플레이스, 포럼, 고발 웹사이트, 소셜 미디어, 대부분의 인터넷 응용프로그램을 자신만의 기계나 서버에 설치하고 호스팅할 수 있다. 동시에 서버 전체와 사용자 중심의 익명성도 유지할 수 있다. 이는 네트워크의 범위를 크게 넓히고 사용자가 Lokinet 안에서 의미 있는 커뮤니티를 조성할 수 있게 허용한다.

SNApps 운영자는 기존의 서버-클라이언트 모델을 이용하되, 서비스 노드가 Lokinet 을 통한 사용자의 연결에서 중개자가 된다는 주요 차이점이 있다. SNApps 를 네트워크에 등록하려면 기술자와 DHT 를 업데이트 해야한다. 이 기술자에는 다양한 소개자를 담고 있으며 이것이 바로 사용자가 SNApps 로의 경로를 만들 수 있는 특정 서비스 노드다. 이 경로가 준비되면, 사용자는 양 당사자가 네트워크에서 서로의 위치를 파악하지 않고도 SNApp 에 연결할 수 있다.

6.3 출구 노드

출구 노드는 사용자가 더 광범위한 인터넷을 요청하고 그러한 요청을 믹스넷을 통해 회신할 수 있도록 한다. 그래서 출구 노드를 정확하게 사용하면 사용자는 개인적으로 인터넷을 이용할 수 있고 사용자 IP 는 서버에 노출되지 않는다.

Loki 의 확장된 유틸리티에서 출구 노드는 필수적이며 출구 노드가 피해가 되는 상황에서도 모든 서비스 노드 연산자가 제대로 작동할 수 있게 만든다. 출구 노드로 작용한다는 것은 운영자가 법적인 리스크에 노출될 수 있다는 뜻이다. 왜냐하면 출구 노드의 사용자가 이를 프록시로 이용하면서 악의가 있는 활동을 할 수 있기 때문이다. 출구 노드는 인터넷에서 최종 사용자에게 트래픽을 전달하는 간단한 역할을 하지만, 보통 디지털 밀레니엄 저작권법 (Digital Millennium Copyright Act, DMCA) 으로부터 요청을 받거나 해킹 시도로 간주되기도 한다. 또한 대부분의 관할 지역에서 세이프 하버법 (safe harboring laws) 은 출구 노드 운영자를 보호하기도 한다. 하지만 서비스 노드 트래픽을 서버에 두고 있는 인터넷 서비스 제공업체는 법적인 리스크를 염려하여 출구 노드의 서비스를 중단하기도 한다.

서비스 노드는 시작부터 릴레이 플래그에 할당되고 Lokinet 의 라우팅 패킷으로는 제한된다. 하지만 절대 더 광범위한 인터넷으로의 요청은 하지 않는다. 운영자는 출구 노드가 되고 싶은 경우 사전에 동의해야하며, 그렇게 함으로써 그들은 추가 스웜 테스트 (8.3.1 참고) 에 제출될 때 리스크가 있다는 점을 이해하는 것으로 간주된다.

출구 노드로서 동의하면 운영자는 블록 보상에 선택되었을 때 일반 릴레이보다 보상을 두 배로 받는다. 이 인센티브로 출구 노드 운영자는 충분한 경제적 인센티브를 받고 출구 노드 네트워크를 장악하려는 목적을 가진 Sybil 공격에 대비하는데도 도움을 줄 수 있다. 이것이 바로 토르가 릴레이로의 출구 노드 부족으로 경험하는 취약성이다.

6.4 원격 노드

그 어떤 암호화폐 네트워크에서도 블록체인의 복사본 전체를 저장하는 것을 가능하지 않으며 사용자들에게도 현실적이지 않다. 비트코인과 이더리움에서 사용자들은 블록체인의 복사본을 갖고 있는 공개 풀 노드에 연결하도록 선택할 수 있다. 또한 네트워크에 의문점을 제시하거나 거래를 제출할 수도 있다. 이러한 활동이 비트코인과 이더리움에서 가능한 이유는 풀 노드가 효율적으로 거래를 위한 블록체인을 검색할 수 있기 때문이며 이로써 사용자의 공개 키는 타겟이 된다.

CryptoNote 통화의 구조 때문에 공개 풀 노드 (원격 노드라고 불림) 는 더 큰 압박을 받는다. 사용자는 원격 노드에 연결할 때 일시적으로 모든 블록 (월렛 생성 후 또는 마지막 블록 체크 후) 을 로컬 기계로 다운로드 해야한다. 공개 거래 키를 위한 각 거래, 즉 사용자가 개인적으로 볼 수 있는 키로부터 생성된 거래를 확인해야한다. 이 절차는 원격 노드 처리에 엄청난 영향을 준다. 이 서비스에 보상이 없다는 점을 감안하면 사용자가 라이트 클라이언트를 위한 동기화 서비스를 운영하지 않도록 단념시킬 수도 있다. CryptoNote 모바일 월렛은 보통 신뢰하기가 어렵고 가끔은 원격 노드 사이에서 여러 번 바뀌기도 한다. 이는 블록체인을 스캔하거나 거래를 제출하기 위해 안정적으로 연결되기 전에 일어난다.

또한 부정한 원격 노드 운영자는 여러 인기 있는 노드 중 하나에서만 운영되고 특정 거래가 알려질 때 사용자의 IP 주소를 기록할 수도 있다. 이러한 공격으로 실제 거래에 대한 그 어떤 정보도 노출되지는 않지만 거래와 특정 IP 주소가 연결될 수는 있다. 즉, 사용자의 실제 신원 정보와의 연결 고리로 이용되어 프라이버시가 확보되기 어려워진다.

Loki 는 각 서비스 노드가 일반 사용자가 사용할 수 있는 원격 노드로 작동하도록 요구하기 때문에 이런 문제를 피할 수 있다. 서비스 노드는 원래 그 자체로 블록체인의 완전한 복사본을 갖고 있고 고대역폭 노드의 널리 분산된 네트워크를 형성하기 때문에 이런 문제에 활용될 수 있다. 서비스 노드를 원격 노드로 사용함으로써 원격 노드 네트워크에 대한 경제적인 근본적인 한계는 있을 수 있다. 특히 특정 당사자가 소유할 수 있는 원격 노드 네트워크의 규모나 악의적인 노드 운영자가 수집할 수 있는 데이터의 양에 제한이 있는 것이다.

6.5 Blink

전형적인 블록체인 시스템에서 모든 거래의 확인 시간은 거래가 블록에 포함되는데 소요되는 시간을 의미한다. 채굴자들이 경쟁하는 과정에서 블록을 주지 않는 경우, 즉 Finney 공격으로 수신자들은 보통 거래를 갖고 있는 블록에 추가적으로 많은 블록 생성을 더 요구한다. 이는 원하던 블록이 승인될 때까지 지속된다 [24]. 이 절차는 각 블록체인의 다양한 요인에 따라 10-60 분이 소요되기 때문에 상인이나 소비자 같이 실제 상품 거래 서비스를 이용하는 사용자들이 대기 시간으로 불편함을 겪고 있다.

Loki 의 서비스 노드 구조에서는 실시간에 가까운 거래가 가능하다. Blink 는 Loki 의 메인 체인에서 블록에 포함되기 전에 승인되는 절차를 동일하게 진행하기 때문에 거래의 송수신자 모두 인증 과정을 확인하고 수신자는 이중 지불을 방지할 수 있다.

Blink 는 DASH 의 InstantSend 와 비슷한 방식으로 작동한다. 각 블록, 서비스 노드 스웜은 결론적으로 거래의 유효성을 인증하는 증인 세트로 선택되고 거래가 두 번 지불되는 상황을 방지한다. 특히 DASH 처럼 결제되지 않은 거래의 출력값을 잠금 설정하는 대신 키 이미지가 잠금된다. 키 이미지는 링 서명에서 결제되지 않은 출력값에 첨부된 독특한 키다. Blink 는 즉각적인 확인을 위해 선택된 스웜에게 네트워크에 신호를 보낼 수 있는 권한을 부여한다. 이 때 출력값과 관련된 키 이미지는 거래가 블록에 추가될 때까지 잠금

설정된다. 결제되지 않은 동일한 출력값을 중복적으로 시도하면 동일한 키 이미지가 생성되고 스웜을 거쳐 네트워크 전체에서 승인을 거절한다.

사용자는 Blink 거래를 이용하여 수수료를 더 많이 지불할 수 있으며 이 경우 몇 초 만에 더 빠르게 거래가 승인된다. 즉, Loki 를 위한 새로운 이용 사례를 확장하여 서로 대면하는 결제 거래의 실용성과 온라인 결제의 통합 가능성이 더욱 커질 수 있다. Loki 에 적용된 모든 개인 정보 보호 기능은 이 절차에서도 유지된다.

7 CryptoNote 의 변화

Loki 는 암호화폐로서 CryptoNote 코인과 비슷한 기능을 갖고 있다. 그러나 서비스 노드 추가 이외에도 함께 제공되는 관련 기능에 주요 차이점이 있다.

7.1 ASIC 저항성

응용 주문형 집적 회로 (Application-Specific Integrated Circuit, ASIC) 는 단일 기능을 위해 특별히 제작된 컴퓨터 칩이다. 채굴에서 ASIC 은 특정 해시 알고리즘을 위한 컴퓨팅에 이용된다. 이 칩들은 탈중앙화에 위험을 끼친다. 왜냐하면 모든 채굴 방식을 앞서면서도 특정 회사에서만 제작되기 때문이다. 또한 하드웨어의 특성상 유통 방식이 제한적인 경우 수익성 있게 제작되고 운용되려면 엄청난 자본이 필요하다. ASIC 칩에도 기대해볼 만한 잠재적인 혜택은 있는데, 바로 채굴자들이 특정 알고리즘 하드웨어에 투자하는 자본 비용 요건이다. 이 요건 덕분에 채굴자들은 자신의 투자에 해가 되는 부정행위를 할 가능성이 낮다. 그러나 ASIC 칩의 제조 및 유통은 아직도 고도로 발전된 해시 알고리즘과 함께 일부 대기업에 집중되어 있다. 이 기업들은 특정 지역으로의 배송을 거부하고 ASIC 칩의 최고 성능을 실현할 수 있는 지역과 소비자를 자의적으로 결정한다. 즉, 그들 스스로 운영 범위를 제한하고 가격을 조작할 수 있다는 뜻이다.

많은 암호화폐들은 ASIC 채굴자들이 네트워크 해시레이 (hashrate) 를 독점하지 못하도록 하기 위해서 Scrypt 나 Ethash 와 같은 ASIC 방지 해시 알고리즘을 개발했다 [25][26]. 최근 Monero 는 CryptoNight 해시 알고리즘을 사용하기도 했다. 이 알고리즘을 운영하려면 L3 캐시라는 엄청난 양이 요구된다. 이론상으로 봤을 때 엄청난 메모리 요구사항으로 ASIC 칩을 제작하는 것이 거의 불가능해야한다. 하지만 2018 년 Bitmain 은 ASIC 칩을 위한 CryptoNight 인 X3 를 출시하면서 GPU(그래픽 처리 장치) 속도를 10 배 높여 효과적으로 채굴할 수 있게 되었다 [27]. 다른 해시 알고리즘도 비슷한 문제를 겪고 있으며 Scrypt, Ethash, Equihash 모두 ASIC 칩으로 채굴되고 있다.

Monero 는 ASIC 칩의 사용에 대응하기 위하여 3-6 개월마다 CryptoNight 해시 알고리즘을 약간 바꿔 하드 포크 (hard fork) 를 하는 전략을 세웠다. (첫 포크는 CryptoNightV7 으로 이동 [28]) ASIC 칩을 만들기 위해 드는 자본과 시간은 어마어마한데다가 매우 정교한 하드웨어 설계도 필수기 때문에 해시 알고리즘에 변화를 약간만 준다는 것은 칩 디자인을 무효하게 만들고 ASIC 제조업체의 자본 및 시간 투자를 헛수고로 만들 수 있다. 그러나 이러한 접근법에는 역시 그 만의 문제가 있다. 알고리즘에 적용된 변동사항이 ASIC 칩의 재프로그래밍을 방지하기에 역부족일 경우 네트워크는 다음 하드 포크가 가능해질 때까지 해시레이트 중앙화로 취약해진다. 그래서 필드 프로그램이 가능한 게이트 어레이 (Field Programmable Gate Arrays, FPGA) 역시 ASIC 에 대항하는 전략으로 고려해보아야한다. 왜냐하면 해시 알고리즘을 이따금 변경하는 것이 FPGA 를 위해 쉽게 재프로그래밍될 수 있기 때문이다. 또 핵심 합의 메커니즘을 주기적으로 변경하면 예상치 못한 버그 발생율이 올라가고 동시에 개발진이 진행하는 이러한 변경 사항은 중앙으로 집중된다는 문제점이 있다.

다양한 대안적 작업증명 알고리즘은 주기적으로 하드 포크를 해야한다는 수요를 해결하기 위해 등장했다. 이러한 대안에는 Argon2, Balloon hash 와 같은 증명 가능한 메모리 하드 해시 알고리즘, ProgPoW 나 RandProg 와 같은 다형적 해시 알고리즘 (polymorphic hashing algorithm) 등이 있다 [29][30][31][32]. Loki 팀은 ASIC 저항성에 대한 장기적인 솔루션을 개발하기 위해 앞서 언급한 알고리즘에 대하여 추가적인 연구를 지속적으로 발표할 것이다.

또한 Loki 는 이러한 작업을 진행하면서 CryptoNight Heavy 라고 불리는 CryptoNight 의 새로운 버전을 만들 예정이다. 이 버전은 CryptoNight ASIC 채굴자들에게 대항하는 ASIC 저항성을 유지할 것으로 기대된다. CryptoNight Heavy 는 CryptoNight V7 과는 여러 방면에서 차이가 있다. 먼저 스크래치패드의 사이즈를 4mb 늘려 제공한다. 또한 내파 및 외부 폭발에 대응하는 방식도 변화가 있을 것이다 [33]. 이러한 변화는 ASIC 채굴자를 위한 최대 타겟인 Monero 의 CryptoNight V7 과는 상이하며 추후 영구적인 솔루션이 개발될 때까지 ASIC 개발에 저항하는 보호 장치로서 작용할 것이다.

7.2 동적인 블록 규모

Loki 는 다른 CryptoNote 코인과 마찬가지로 정해진 블록 규모가 없다. 블록 규모는 시간이 지나면서 변화하며 특히 네트워크의 거래 처리량이 더 커지면서 거래도 더 많이 포함할 수 있다. Loki 의 블록 규모 측정 체계는 최근 블록 100 개에서 규모의 중앙값을 확인하고 천천히 다음 새 블록의 최대 규모를 다시 변경한다.

다른 암호화폐들이 장기적으로 우려하는 사항은 큰 블록 규모가 노드에게 지나친 부담이 되어 거래를 저장하고 인증하는데 문제가 생길 수 있다는 점이다. 낮은 등급의 하드웨어를 이끄는 노드는 블록 규모가 커지면서 새로운 블록을 늘릴 수 없게 된다. 이는 결국 노드 네트워크가 중앙화되고 노드를 유지하는데 상업적인 이해관계에 의존하는 결과를 낳는다. 이렇게 되면 여러 노드에 걸친 블록체인을 분산하는 것이 서로 다른 다수의 당사자들 사이에서 체인의 상태를 확인하도록 허락하기 때문에 문제가 된다. 또한 유효성과 검열에 대한 저항성도 증가한다.

Loki 에서 블록 보상의 일부는 서비스 노드에게 전송되며 풀 노드로서 블록을 처리하고 전달한다. 대역폭과 성과가 충분하지 않은 서비스 노드는 서비스 노드 네트워크에서 제외되기 때문에 (7.3 참고) 보상 풀은 스스로 최소의 성과 요건을 제시한다. 이러한 인센티브 구조는 노드수도 높게 유지하면서 충분한 성과를 보여주는 노드가 네트워크에서 성공적으로 블록체인 데이터를 공유하도록 보장한다. 이는 블록체인이 성장하는 규모나 대역폭 요건에 상관 없이 확보된다. 물론 이러한 장점에도 불구하고 거래 규모를 최적화하려면 네트워크의 규모도 효율적으로 유지해야한다. 이는 서비스 노드의 운영 비용을 낮게 유지하여 장기적으로는 노드수를 높게 유지하기 위함이다.

7.3 링 서명 크기

링 서명은 거래에서 실제 출력값을 다른 값들 사이에 숨기기 위해 사용된다. 그래서 링 서명의 크기는 링을 구성하기 위해 혼합된 값의 수를 뜻하기도 한다. Monero 는 현재 최소 링 크기를 7 로 정하고 있고, 거래에서 실제 지불되지 않은 출력값과 혼합된 값 6 개를 포함한다.

하지만 paper 0001(Monero Research Lab 발표) 에서 언급한 바에 따르면, 링 크기가 클 때의 효과를 연구하는 경우는 매우 드물다. 오히려 규모가 큰 출력값을 블록체인에 보유하고 있는 공격자와 다양한 링 크기에 따른 효과를 대비하는 연구하는 경우가 더 많았다 [34]. 링 크기가 높을수록 지불되지 않은 출력값을 많이 보유하고 있는 악의적인 공격자

가 거래를 분석할 수 있는 타임프레임을 줄이는 것으로 밝혀졌다. 그래서 링 크기를 크게 하도록 지시하는 것 역시 EABE/Knacc 공격이라고 알려진 이론적인 공격으로부터 보호할 수 있다 [35]. 이 공격 때 제 3 당사자 (예: 거래소) 는 두 사용자 사이의 거래에 대해 제한되고 시간 분석을 진행할 수 있다.

또한 Monero 는 네트워크 합의 규정에 따라 최대 링 크기를 한정하지 않는다. Monero GUI 월렛과 비슷한 다양한 월렛이 링 크기를 26 으로 한정하고 있다. 사용자는 언제든지 7 이상의 원하는 링 크기로 거래를 생성할 수 있다. 하지만 대부분의 월렛이 기본 링 크기를 7 로 설정하고 있기 때문에 이는 오히려 문제가 된다. 거래의 링 크기를 7 이상으로 설정하면 눈에 띄기 때문이다 (그림 4). Monero 에서 개인의 거래가 항상 기본 외 링 크기를 사용한다면 (예를 들어 10) 외부 영향을 받은 타사가 블록체인을 분석하고 시간 분석 기술로 패턴을 추론할 수도 있다.

거래해시	링크기	tx규모[kB]
3feaff3f48de0bc4c92ec027236165337b64df404aca098e212c1215e9456697	7	13.47
39d484f7c0a2e8f3823a514056d7cb0bf269171cb4582e05955d4c5ee995cad0	7	13.47
e08f5a937e725011bedd44075334ae98dcca32749da231c56da1278d49c0a231	7	13.50
ab35e69d9cca39219c90df8b2b7aab4a54c82127fb1fbaae65d76357f8f76387	7	13.50
6d8ccd56dc2d3eb7de03ba767f0dbf4d5f42ae91e67f4c28f16d6f8b0229c272	10	13.87

그림 3: *xmrchain.net(Monero* 블록 탐색기*)* 에서 기본 외 링 크기가 눈에 띄는 예시

Loki 는 이런 문제들을 해결함과 동시에 링 크기를 고정하도록 강조하고 그 크기를 10 으로 설정한다. 링 크기를 고정하면 9 개 이상의 혼합으로 링을 구성하는 사용자들을 보호할 수 있다. 또한 최소 링 크기를 10 으로 설정함으로써 많은 출력값을 보유하고 있는 공격자가 링 서명 내에서 실제 결제 출력값을 알아내는 행위에 더 효과적으로 대비할 수 있다. 링 크기가 크면 기본적인 변동 유효성을 비선형으로 증가시키고 크기가 커지면서 그 효과도 더 커진다.

현재 거래 구조에서 링 크기를 10 으로 키우면 거래 규모가 2.6% 로 증가한다. 그러나 Bulletproof 가 시행되면 8-13% 증가된 거래 규모를 처리하게 될 것이다. 이는 Bulletproof 로 거래 규모가 전반적으로 감소하기 때문이다. 최소 링 크기는 네트워크에서 문제를 야기할 수도 있다. 특히 큰 규모의 거래를 지원하지 않는 구조의 단점이 드러난 네트워크는 부담해야할 비용도 높다. Loki 를 이용하면 운용에 인센티브를 지급받고 충분한 대역폭을 제공하는 서비스 노드가 이런 문제를 해결할 수 있다.

8 공격 방지책

8.1 IP 와 패킷 저지

서비스 노드 네트워크가 실패에 가장 중점적인 이유가 되지는 않지만 네트워크는 두 가지 중요한 검열 위협에 직면한다. 즉, 수집 공격과 심층 패킷 분석이 바로 그것이다 [36][37]. 수집 공격은 네트워크에서 작동하는 모든 서비스 노드의 IP 주소를 수집하며 ISP 수준의 방화벽을 사용하여 특정 주소들의 연결을 차단한다. 이런 검열은 중국의 토르 네트워크에서 정기적으로 진행된다 [38]. 심층 패킷 분석 (Deep packet inspection, DPI) 은 방화벽을 통과하는 개별 패킷의 구조를 조사하고 특별한 서비스와 관련된 패킷들을 선별적으로 차단한다. DPI 는 정부 단위의 활동가가 광범위하게 사용중이다 [39].

시중에는 DPI 를 피하기 위한 시스템을 설계하기 위해 수많은 노력이 있었다. 사용자는 접속 가능한 운송 종류를 보강할 수 있다. 이 때 운송 수단은 각 패킷이 차단되지 않은 트래픽으로 보이도록 서명을 바꾼다. IP 차단은 보통 도메인 프론팅 브리지 (domain fronting bridges) 를 운영하면서 방지할 수 있다. 이 브릿지는 트래픽을 Azure 나 Cloudflare 같은 미차단 서비스의 HTTPS 요청으로 암호화한다. 그들이 미차단 서비스에 접근하면 브릿지는 해당 요청을 원하는 위치로 전송한다. 도메인 프론팅의 경우 국가급의 관계자가 인기 브릿지로의 모든 트래픽 흐름을 방지하는 것이 어려워진다. 특히 인터넷의 일반적인 사용을 엄청나게 방해하기 때문이다.

Loki 에 구축된 관리 메커니즘 (9 참고) 은 도메인 프론팅 브릿지를 운영하는데 이용될 수 있다. 그래서 사용자들은 Loki 서비스를 대규모 인터넷 검열 정책이 적용되는 국가에서도 접근할 수 있는 것이다. 또한 OBFS4 접속 가능한 운송은 Loki 월렛의 서비스 노드 출시와 묶여서 지원되며 이는 DPI 에 대항하여 한 층 더 높은 보호 서비스를 제공하기 위함이다 [40].

8.2 서비스 거부 공격

탈중앙식 블록체인의 사용자들은 디지털 또는 물리적인 식별자를 제공하지 않아도 된다. 이는 신원 정보가 부족하거나 특정 박해를 받는 사람들에게 매우 유용하다. 하지만 신원 정보를 요구하지 않는 시스템은 Sybil 공격에 취약하게 되며 부정 행위자들이 수많은 위조 신원 정보를 만들 수 있다. (Loki 에서는 수많은 공개-개인 키 쌍을 의미함) 또한 이러한 신원 정보를 이용하여 네트워크에 요청을 무작위로 대량 전송하기도 한다.

많은 암호화폐들이 이 문제를 해결하기 위해 노력했고 서비스 요금 모델이나 작업 증명 모델을 시행하도록 강요 받았다. Siacoin 같은 서비스 요금 모델은 사용자가 이용 서비스에 대한 요금을 지불한다. Siacoin 의 경우 비용은 월간 저장 공간의 TB 당 결정한다 [41]. 서비스 요금 모델은 Sybil 공격을 줄이는데 효과적이지만 이용자 역시 감소한다는 단점이 있다. 시중에 무료로 서비스를 제공하는 시스템이 있으면 사용자는 그 쪽으로 이동하기 마련이다. (Google Drive 와 Onedrive 등) Hashcash 와 Nano 같은 작업 증명 시스템은 사용자가 메시지나 거래를 전송하기 전에 작업 증명을 계산하도록 요구한다 [42][43]. 이러한 작업 증명 시스템은 서비스 요금 모델보다 평등하다는 옹호의 의견도 있지만 컴퓨팅 파워가 강력한 공격자의 피해를 입을 수 있다.

Loki 는 작업 증명 방식을 수정하여 Sybil 공격의 표면 두 가지를 Loki 시스템에 제시한다. 바로 오프라인 메시지와 경로 생성이다. 오프라인 메시지는 각 메시지가 9 개 노드 스웜에 저장되어야하기 때문에 잠재적인 타겟을 보여준다. 남용이 심각해지면 부정행위를 하는 사용자가 엄청난 양의 메세지를 담은 특정 스웜을 저장하도록 과도하게 요구할 수 있다.

경로 생성 공격 시 공격자는 가능한 많은 노드 수로 경로를 생성하도록 시도한다. 그래서 대역폭 리소스를 차지하고 합당한 목적을 가지고 네트워크에 경로를 생성하는 사용자들에게도 서비스를 제공하지 못하게 한다.

이 두 가지 공격을 모두 예방하기 위해서 Loki 네트워크는 메시지와 경로가 생성될 때 짧은 작업 증명을 추가한다. 메시지에서는 이 작업 증명이 메시지의 Blake2b 해시로 계산된다. 경로 생성에서는 작업 증명이 경로 구축 과정에 포함된 노드 요청과 함께 전송된다. 모바일 사용자의 접근성 및 확장성을 보장하기 위해 작업 증명의 난이도 요건은 글로벌 네트워크 활동이 아니라 메시지나 경로의 존속 시간 (TTL) 에 근거하여 고정된다.

8.3 스웜 플래깅 (Swarm Flagging)

노드가 중요한 규칙을 강조하는 중앙의 리더 없이 무신뢰 환경에서 작동할 때, 네트워크에서 노드의 행위를 적절하게 유지하는 것은 매우 어렵다. Loki 의 서비스 노드가 담보 금액 요건을 정확하게 지켜야 하지만 트래픽을 전송하지 않거나 메모리 풀에 데이터를 저장하지 않는 경우도 발생할 수 있다. 그렇게 함으로써 경제적인 이득을 취할 수 있기 때문이다. (대역폭/CPU 주기/저장소를 적게 사용함) 그래서 분산된 플래깅은 실적을 내지 못하는 노드를 제거하도록 해야한다.

Loki 에서 그러한 분산된 플래깅은 수행시 관련 문제에 직면한다. 기본적으로 모든 서비스 노드는 다른 서비스 노드가 제대로 작동하지 않는다는 사실을 플래깅함으로써 경제적인 인센티브를 받는다. 이는 서비스 노드가 플래깅 되면 채굴 풀에서 제거되고 플래깅 한 대상이 보상을 받을 확률이 높아지기 때문이다. 분산된 플래깅의 가능한 한 방법은 플래깅할 때 증거를 제시하는 것이다. 하지만 이 해결책은 노드가 자신에게 유리하도록 증거를 조작하게 만들 수 있다. 반대로 제한 없이 플래깅을 허용하면 단일 노드든 협력하는 노드의 집단이든 의도적으로 블록 보상을 위해 정직하게 플래깅을 할 것이다. 이러한 문제를 피하기 위해 Loki 는 스웜 플래깅을 제안한다.

스웜 플래깅은 기존 스웜 (6.1.1 참고) 을 이용하여 각 테스팅 라운드에 참여하는 일원을 선택한다. 각 서비스 노드는 블록체인의 복사본을 보유하고 채굴자가 생성한 각 블록은 결론적으로 많은 테스트 스웜을 선택하게 된다. 각 블록, 즉 네트워크 스웜의 1% 는 테스팅 스웜에 참여하기 위해 선택된다. 참여하는 스웜을 계산하기 위해 이전 블록 5 개의 해시를 이용하여 Mersenne Twister 기능을 도입한다. 그리고 결정적인 리스트에서 포지션의 순서에 따라 스웜을 선택한다.

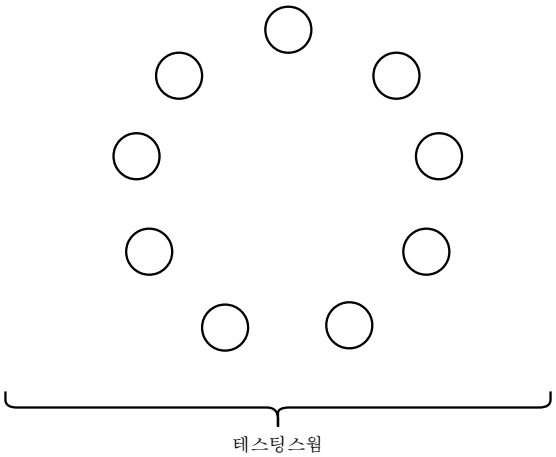

그림 4: 테스팅 스웜은 *9* 개의 노드로 선택된 스웜이다.

스웜이 참여 대상으로 선택되면 이 스웜의 노드는 스웜 내 다른 노드에 대해 모두 테스트를 진행해야한다. 이는 실제 테스트는 아니며, 각 노드의 정보, 특히 스웜 내 다른 노드와의 상호 작용 정보를 저장하는 일이다. 이 때 대역폭, 메시지 저장소, 블록체인 요청, 출구 노드 기능 등에 대한 정보가 수집되고 시간이 지나면서 유지된다. 이런 정보를 수집해야하는 스웜의 새로운 참여 개체는 가까운 스웜 외 서비스 노드에도 의문을 제기하고 그들이 테스트하는 각 서비스 노드에 대한 데이터도 수집한다.

각 서비스 노드는 다른 스웜 일원 각자를 어떻게 투표하는지 결정한다. 상기 언급된 테스트에 근거하여 결정을 내리면, 노드는 스웜에 대한 정보를 수집하고 전송한다. 스웜의 각 노드는 이제 모든 일원의 투표 내용을 확인할 수 있다. 스웜의 한 노드가 노드 투표의

50% 이상 반대표를 얻은 경우, 그 어떤 스웜 일원이든 제명 거래를 생성하기 위한 정보가 요구된다. 이 거래가 승인되고 블록에 추가되면, 모든 서비스 노드는 그들의 DHT 를 업데이트하고 반대 투표 후 선정된 노드를 제거한다.

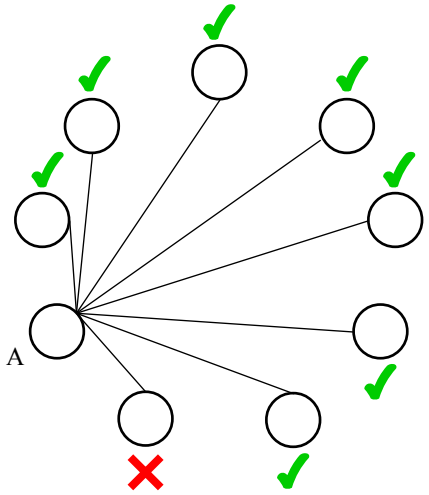

그림 5: 부정직한 노드는 노드 A 가 진행하는 테스트에서 실패한다. 노드 A 는 어떤 노드가 테스트에 실패하고 합격했는지 해당 스웜의 노드와 이해하게 된다.

8.3.1 테스팅 세트

네트워크가 이행 기준을 자발적으로 실시하기 위해서는 서비스 노드가 필수 툴을 갖춰서 다른 서비스 노드를 테스트해야한다. 이 테스트는 서비스 노드에게 부여되는 모든 기능을 다 확인할 수 있어야하며 그래야 불성실한 마스터노드 (masternode) 의 공격을 방지한다 [44]. 초기 테스팅 설계에서는 4 가지 테스트가 제시되었다. 추후 서비스 노드의 기능이 확장되면 테스트가 추가될 수도 있다.

연산자가 처음 서비스 노드 소프트웨어를 작동시키면 사전에 결정된 규모가 디스크에 할당되고 저장소가 필요한 작업을 위해 공간이 확보된다. 그 다음 Loki 재단이 운영하는 서비스 노드와 지리적으로 분산된 테스팅 서버간 간단한 대역폭 테스트가 이루어진다. 이러한 체크 사항은 선택할 수 있으며, 서비스 노드는 테스트를 건너뛰기, 무시, 실패하거나 신뢰할 수 없는 서비스 노드 풀에도 진입하도록 선택할 수 있다. 그러나 이 테스트를 진행하고 통과하면 긍정적인 지표가 제공되어서 추후 서비스 노드 연산자가 최소 요건을 갖추지 않은 노드에 담보물을 걸어둘 리스크에 대해서 결정할 때 참고하게 된다. 서비스 노드가 신뢰할 수 없는 서비스 노드 풀에 진입하면, 이들의 담보물은 락이 걸리고 다음에 선택된 스웜에서 테스트된다. 스웜 테스트는 합의를 통해 진행되며 새롭게 서비스 노드 네트워크에 진입하는 일원은 이 테스트를 피할 수 없다. 노드가 모든 스웜 테스트를 통화하면 그들은 신뢰할 수 있는 노드 플래그로 보상받고 패킷 라우팅을 시작할 수 있다. 테스트에 실패하면 그들은 네트워크에서 제거되고 담보물은 30 일간 잠금 상태로 유지된다.

대역폭 테스트

대역폭 테스트는 Loki 네트워크 테스트 세트의 근간을 형성한다. 노드가 이 테스트를 통과하면 최소 역치 이상에서 패킷을 라우팅할 수 있다고 간주된다.

노드는 다른 서비스 노드와 상호 작용할 때마다 새로 들어오는 대역폭의 기록을 만들고 보유한다. 시간이 지나면서 수천가지의 경로에 노드가 포함되고 수백만의 메시지를 처리

한다. 이러한 상호 작용으로 각 노드는 대역폭 표의 기본을 만든다. 이 표에서 노드는 스웜 내 서비스 노드에 관한 대역폭 테스트에 반응할 수 있다.

모든 노드가 다른 노드의 대역폭 표에 따른 질문에 반응하도록 예상된다. 즉 최근에 네트워크에 합류한 노드도 더 넓은 네트워크에 그들의 스웜에 있는 특정 노드에 대한 정보를 질문할 수 있다는 뜻이다.

메시지 저장 테스트

메시지 저장은 Loki 메신저의 사용자들을 위해 오프라인 메시지 기능에서 필수적이다. 서비스 노드는 이 능력을 확인하기 위해 메시지를 캐시에 저장하고 메시지의 존속 시간 (TTL) 에 따라 사용자에게 제공할 수 있어야 한다.

오프라인 메시지를 전송하는 사용자는 목적지 사용자 스웜에서 서비스를 무작위로 선택한다. 이 노드는 나머지 스웜 내 메시지의 복사본을 분배해야한다. 메시지 헤더에 부착된 작업 증명에 따라 복사본을 받는 서비스 노드는 TTL 동안 데이터를 저장한다. 원래 메시지의 TTL 이 다다르면 분배 노드는 스웜의 나머지 일원들에게 모두 논스를 전송한다. 스웜은 그 논스를 메시지에 추가하고 결과를 해시하여 분배 노드에 다시 보낸다. 이 테스트는 서비스 노드가 TTL 이 다 지날 때까지 메시지를 보유하고 있는지 확인하고 정확한 메시지 요약본을 생산하지 못하는 경우 해당 노드를 제거한다. 분배 노드의 표본이 일정하기 않기 때문에 시간이 지나면서 서비스 노드는 스웜 내 다른 일원의 성과 데이터로 수집할 수 있을 것이다.

블록체인 저장소 테스트

서비스 노드는 Loki 블록체인의 복사본 전체를 보유하도록 예상된다. 그렇기 때문에 서비스 노드는 네트워크의 사용자에게 필수적인 많은 일을 수행할 수 있다. 원격 노드로 활동하거나 거래를 인증하는 것, Blink 에서 거래에 락을 거는 것 역시 포함된다.

정직한 노드가 블록체인의 복사본을 보유하기 때문에 부정직한 노드는 테스트를 받을 때 정직한 노드에게 블록을 요청함으로써 전체 복사본을 저장하고 있지 않아도 된다. 이런 결과를 방지하기 위해 블록체인 저장소 테스트는 블록체인의 복사본을 저장하고 있는 정직한 노드만 통과할 수 있도록 설계되었다.

이를 달성하기 위해 테스팅 노드는 테스트 대상이 되는 각 노드에게 무작위로 K 개의 거래를 선택하여 연결시키고 해시한다. 이 무작위의 거래는 블록체인의 내역에서 선택되어온다. 이 해시는 테스트 대상이 되는 노드에게 다시 보내진다. 이 요청의 대기 시간을 측정함으로써 노드는 예측 회수 시간인 T 와 비교된다. T 의 정확한 값은 디스크에서의 로딩과 네트워크에서 블록을 다운로드 하는데 예측된 대기 시간 차이를 정확하게 계산해서 설정된다. 공격자는 K 개의 블록을 T 내로 다운로드하고 해시할 수 없기 때문에 piggybacking 공격도 어려워진다.

출구 노드 테스트

출구 노드로 활동하기로 선택한 서비스 노드는 추가 보상을 받고 기술 테스트를 진행해야한다. 이는 추가 보상이 남용되지 않기 위함이다.

기능 관련 출구 테스팅이 진행되면 서비스 노드는 사람이 진행하는 자연스러운 검색 활동을 모방할 수 있어야한다. 서비스 노드가 테스트된다는 사실을 감지하면 테스트 하거나 타당한 사용자 요청을 폐기하도록 선택하여 반응할 수 있다. 자연스럽게 페이지 요청을 모방하는 행동은 어렵지만 출구 테스트는 타당한 요청과 테스트를 구분하는 오버헤드

를 만들도록 설계할 수 있다. 그래서 타당한 노드와 부정한 노드의 대역폭 비용 차이는 경미하다.

서비스 노드는 로컬 영역에서 보유하고 있는 검색 엔진 목록과 사전을 사용하여 의사 임의적으로 자연스러운 검색어를 구성한다. 그러면 검색어는 검색 엔진에 들어가고 웹 페이지가 결과로서 무작위로 선택된다. 서비스 노드는 이제 무작위 노드와 경로를 구성하고 릴레이로 활동하며 테스트 중인 노드는 출구 노드로 작용한다. 이 출구에서 서비스 노드는 그 의사 임의적인 검색에서 웹페이지 결과를 요청한다. 출구 노드로 돌아온 결과가 서비스 노드에서 생성한 결과와 일치하면 출구 노드는 테스트를 통과한 것으로 간주된다.

9 관리, 자금 공급, 투표

관리는 암호화폐 설계에서 중요한 부분이며 프로토콜의 수준으로 지원되어야한다. 약식으로 정의된 관리의 위험성은 블록체인 기술 역사상 광범위하게 연구되어왔다. 비트코인과 이더리움은 논쟁의 여지가 있는 하드 포크를 경험했고 그들 각각의 커뮤니티에 대한 관심과 노력을 분산시켜버렸다. 하드 포크가 관리 전략으로 사용될 수는 있지만, 그들은 항상 모든 논쟁의 문제에 마지막 해결 방법으로 고려해야만한다. Loki 관리 시스템은 담론과 대표의 구조화된 환경을 제공함으로써 잠재적인 문제를 해결하도록 설계되었다. 또한 Loki 의 발전 자금도 외부 영향이나 의존 없이 마련하는 것도 중요한 목표 중 하나다.

관리 구조는 하드 포크를 방지하는 것뿐 아니라 내부적으로 새로운 프로젝트에 재정적 지원 수단을 마련하고 Loki 생태계를 발전시킨다는 의미에서도 중요하다. 내부적으로 재정 프로젝트는 사용자, 채굴자, 서비스 노드와 동기가 일치하지 않는 특별한 관심을 가진 집단의 형성을 막을 수도 있다. 우리는 비트코인과 많은 비트코인 포크에서 Blockstream, Bitcoin ABC, Bitcoin Unlimited 와 같은 영리 기관이 설립되는 것에서 이런 현상을 확인했다. 이런 기관들은 비트코인과 비트코인 캐시에 프로토콜 지향적 변화를 이끄는 개발자들만 고용한다고 비판을 받아왔으며 이는 사업적 목표를 달성하거나 그들의 특정 이데올로기를 목표로 하는 것이었다.

그래서 Loki 블록에는 모두 네트워크 관리 목적으로 보상의 5% 를 할당한다. 이는 Loki 가 커뮤니티 프로젝트, 소프트웨어 개발진, 통합 팀 등을 위해 분배할 수 있는 지속적인 흐름을 제공한다는 뜻이다. 이 5% 의 블록 보상에서 3.75% 는 Loki 재단에서 통제하고 1.25% 는 Loki 재단 시스템을 통해 서비스 노드가 관리한다. 이로써 서비스 노드의 공정성을 강조하고 커뮤니티를 위한 재정 관련 제안은 Loki 재단의 직접 통제가 아닌 외부에서 이루어질 수 있다.

9.1 Loki 재단

Loki 재단은 호주에 비영리 재단으로 등록되어있다. 중심 법인은 Loki 프로젝트가 탄탄한 법률 프레임워크에서 운영되도록 허용하고 있으며 프로젝트의 법적 보호와 의무를 다하고 있다. Loki 재단은 호주에서 2018 년에 설립되었고 호주 자선 및 비영리 위원회 (ACNC) 에서 제시한 예시와 같은 기관을 이용한다 [45]. 이 기관은 재단에게 다른 수많은 비영리 재단과 같은 법인 및 구조 관리를 제공하며 주주나 수혜자가 따로 없다. 관리 이사회 회원들은 각자 임기 기한이 있으며 다른 회원과 함께 투표를 통하여 다양한 결정을 내리고 수행한다. Loki 재단은 호주에 자선 단체로 등록되어있다.

이 기관은 구조상 수익 (블록 보상 관리 포함) 이 프로젝트를 발전시키거나 관련 계획을 주도하는 데만 사용되도록 제한되어있다. 또한 외부에서 감사 기관을 이용하는 조직으로서 Loki 재단이 수령하는 자선 단체라는 등록 형태를 유지하는데 투명성이 매우 중요하

다. 또한 Loki 재단은 정직하게 합리적인 범위 내에서 자금을 소비할 것을 일반 대중에게 보장한다. Loki 재단은 커뮤니티와 감사 기관에게 모두 책임을 다한다. 이 시스템이 궁극적으로 Loki 과 제반 프로젝트를 지원하는데 실패한다면 엄격한 보호도 뒤따른다. 네트워크의 합의가 충분히 이루어진 상태에서 하드 포크가 진행된다면, Loki 재단은 블록 보상의 수령 대상에서 제거되거나 대체될 수도 있다.

9.2 Loki 재정 시스템

Loki 재단이 Loki 프로젝트를 대변하는 여러 개인에 의해서 설립되었지만, 이 재단은 고유 관리 구조와 호주 법률의 지배를 받는다. 이는 재단이 결정할 수 있는 사항의 범위를 제한하는 요인으로 작용할 수도 있다. Loki 의 재정 시스템은 블록 보상의 일부가 서비스 노드에서의 투표로만 활용되도록 허용한다. 서비스 노드는 전세계 다양한 독립체를 대표하고 Loki 프로젝트 팀이나 재단의 영향력 하에 있지 않다. 그래서 그들이 내리는 결정에도 자율성이 새롭게 보장된다. 서비스 노드는 네트워크에서 채굴을 가장 많이 한 참여자로 Loki 의 가치를 높이는 결정을 내리면 경제적인 인센티브를 받는다.

9.2.1 제안

서비스 노드 앞에 놓이는 모든 제안은 Loki 블록체인에 올려진다. 특정한 참여자가 서비스 노드에 제안을 올리고 싶다면 제안 거래를 생성해야한다. 제안 거래의 컨텐츠는 출력값을 제거한 상태에서 읽기가 가능해야하기 때문에 Loki 의 전형적인 거래 관련 개인 정보 보호 기능을 포기해야한다.

자금을 제공하는 블록도 역시 43,000 개의 블록마다 생성된다. (약 60 일) 제안의 리더는 이 기간 동안 언제든지 제안을 제출할 수 있다. 하지만 각 제안 단계 초기에 가깝게 제출할수록 각 서비스 노드로부터의 투표를 얻는 시간이 더 오래 걸린다는 점을 고려해야할 것이다.

각 거래에는 각 서비스 노드가 제안에 투표하려면 이해해야하는 정보를 담고 있는 필드가 추가적으로 덧붙여진다. 해당 정보는 다음과 같다.

제안 제목, 제안에 대한 자세한 설명이 있는 URL 링크, 제안이 목표로 설정한 Loki 의 수, 결제 주소, 선택한 에스크로 대리인.

제안을 만든 사용자는 Loki 재단 합의에 대기하면서 Loki 재단이나 기타 타사를 위한 에스크로 대리인으로 선출될 수 있다. 이 경우 주요 단계에 이르렀을 때 자금을 해제할 수 있다. 또한 제안의 기준을 높이고 이런 거래로 일어날 수 있는 스팸을 방지하기 위해 각 제안 거래는 적지 않은 양의 Loki 를 제거해야한다.

9.2.2 투표

각 서비스 노드는 투표를 위한 특정 키를 갖고 있다. 이 키는 호스팅시 서버에 로그인하지 않고도 서비스 노드를 대신하여 투표에 활용되거나 내보낼 수 있다.

투표는 체인에서 진행되지 않고 각 서비스 노드가 블록체인 내 활성 제안에 대하여 찬성, 반대, 기권에 대한 신호를 보낸다. 서비스 노드는 제안이 블록체인에 회부된 순간부터 블록에 자금을 제공하는 다음 격월 주기까지 투표할 수 있다. 다음 자금 블록이 생성되기 바로 전에는 진행된 모든 투표 기록을 수집하는데 스웜이 선택된다. 이 총계는 노드의 멤풀에 제출되고 채굴자가 자금 블록에 도달할 때까지 머무른다. 이 정보는 받아들여진 제

안에 대한 보상이 할당되는 블록을 제작하는데 이용된다. 제안은 [찬성표 - 반대표] 값이 서비스 노드 네트워크의 노드 수에 15% 와 같은 경우에만 통과된다.

9.2.3 자금 분배

Loki 자금 시스템의 모든 수익금은 자금 블록을 통해서 지급된다. 자금 블록 보상은 기존의 블록 보상 구조와 비슷하게 작동한다. 즉, 완전히 자율적으로 Loki 를 분배하는 방식이다. 43,000 개 블록마다 (약 60 일) 채굴자가 자금 블록을 생성한다. 이 블록은 전체 자금 블록 기간 동안의 전체 블록 보상의 25% 를 차지한다.

유효한 자금 블록을 생성하려면 채굴자는 요구된 투표율을 달성한 제안을 평가할 수 있어야한다. 이는 서비스 노드가 블록체인에 제출한 정보를 이용함으로써 가능하다. 해당 정보로는 결제 주소와 모든 투표의 결과 등이 담겨 있다. 모든 서비스 노드는 채굴자가 블록에 자금을 지급했는지 인증하고 유효하지 않은 주소에 결제한 자금 블록은 모두 무시한다.

승인된 제안에서 Loki 에게 요구하는 총계는 종종 60 일의 기간 동안 달성한 총계보다 적거나 초과한다. 승인된 제안의 총계가 자금 블록에서 가능한 금액을 초과하면 채굴자는 자금 블록을 우선적으로 생성하여 이전에 블록체인에 제출된 제안을 일순위로 처리한다. 남아있는 승인된 제안들 역시 다음 자금 블록 때까지 블록체인에 남게 된다.

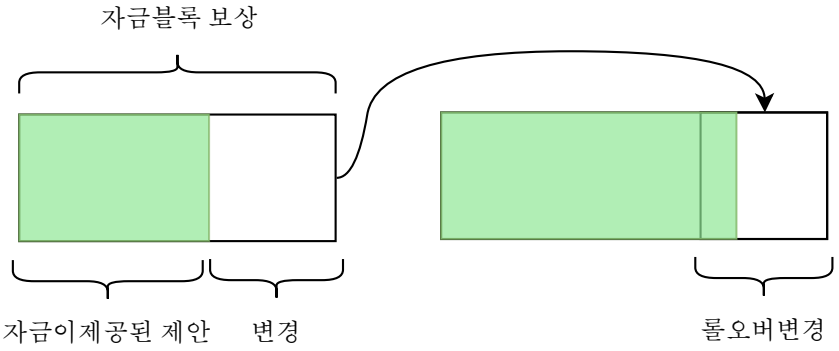

그림 6: 왼쪽에 사용되지 않은 자금은 다음 자금 블록의 보상으로 추가된다.

10 결론

Loki 는 익명 거래와 탈중앙식 통신의 모델을 소개한다. 이는 경제적으로 인센티브를 제공하는 노드 네트워크로 구축되었다. Loki 는 CryptoNote 프로토콜을 기반으로 하여 개인 정보의 보호를 보장하고 담보로 보증된 노드 시스템을 적용한다. 이로써 네트워크는 회복력과 기능을 향상시킬 수 있다.

또한 Loki 는 이전의 연구와 오픈소스 프로젝트에 개선점도 소개하며 새로운 익명 라우팅 프로토콜도 선사한다. 이 프로토콜로 기존의 프로토콜에는 중요한 이점을 제시한다. 고유 구조와 프로토콜 설계를 조합하여 만든 네트워크는 시장을 기반으로 한 Sybil 공격에 저항할 수 있는 능력을 기르며 시간 분석의 효과를 낮춰 사용자에게 디지털 프라이버시를 최대한 보장할 수 있다.

참고문헌

[1] Mike Orcutt, *Criminals Thought Bitcoin Was the Perfect Hiding Place, but They Thought Wrong* (September 11, 2017), https://www.technologyreview.com/s/608763/criminals-thought-bitcoin-was-the-perfect-hiding-place-they-thought-wrong.

[2] *Monero*, https://getmonero.org.

[3] *Tor Project*, https://www.torproject.org.

[4] *I2P Anonymous Network*, https://geti2p.net/en.

[5] *LWMA Difficulty Algorithm*, https://github.com/zawy12/difficulty-algorithms/issues/3.

[6] Daniel J. Bernstein, Peter Birkner, Marc Joye, Tanja Lange, and Christiane Peters, *Twisted Edwards Curves* (2008), https://eprint.iacr.org/2008/013.pdf.

[7] Nicolas van Saberhagen, *CryptoNote v 2.0* (2013), https://cryptonote.org/whitepaper.pdf.

[8] Whitfield Diffie and Martin E. Hellman, *New directions in cryptography*, IEEE Trans. Information Theory **IT-22** (1976), no. 6, 644–654. MR0437208

[9] Shen Noether, Adam Mackenzie, and Monero Core Team, *Ring Confidential Transactions* (2016), https://lab.getmonero.org/pubs/MRL-0005.pdf.

[10] Benedikt Bünz, Jonathan Bootle, Dan Boneh, Andrew Poelstra, Pieter Wuille, and Greg Maxwell, *Bulletproofs: Short Proofs for Confidential Transactions and More* (2017), https://eprint.iacr.org/2017/1066.pdf.

[11] Evan Duffield and Daniel Diaz, *Dash: A Privacy-Centric Crypto-Currency*, https://github.com/dashpay/dash/wiki/Whitepaper.

[12] *GitHub - loki-project/loki-network*, https://github.com/loki-project/loki-network.

[13] *Tor Project: Docs*, https://www.torproject.org/docs/faq#KeyManagement.

[14] *Possible upcoming attempts to disable the Tor network | Tor Blog.* (December 19, 2014), https://blog.torproject.org/possible-upcoming-attempts-disable-tor-network.

[15] Petar Maymounkov and David Mazières, *Kademlia: A Peer-to-peer Information System Based on the XOR Metric*, https://pdos.csail.mit.edu/~petar/papers/maymounkov-kademlia-lncs.pdf.

[16] Philipp Winter, Roya Ensafi, Karsten Loesing, and Nick Feamster, *Identifying and characterizing Sybils in the Tor network* (February 25, 2016), https://arxiv.org/abs/1602.07787.

[17] *OSI model - Wikipedia*, https://en.wikipedia.org/wiki/OSI_model.

[18] Farid Farid, *No Signal: Egypt blocks the encrypted messaging app as it continues its cyber crackdown* (December 26, 2016), https://techcrunch.com/2016/12/26/1431709.

[19] Matt Burgess, *Russia's Telegram block tests Putin's ability to control the web* (April 24, 2018), http://www.wired.co.uk/article/russia-google-telegram-ban-blocks-ip-address.

[20] *Go Ethereum - Postal Services over Swarm*, https://github.com/ethersphere/go-ethereum/blob/ddfc0a2a02ce574f4c252068ce81f0f5ada1c1ff/swarm/pss/README.md.

[21] Nikita Borisov, Ian Goldberg, and Eric Brewer, *Off-the-record Communication, or, Why Not to Use PGP*, Proceedings of the 2004 ACM Workshop on Privacy in the Electronic Society, 2004, pp. 77–84, DOI 10.1145/1029179.1029200.

[22] *NaCl: Networking and Cryptography library*, https://nacl.cr.yp.to.

[23] *Pidgin-Encryption - SourceForge*, http://pidgin-encrypt.sourceforge.net.

[24] *Irreversible Transactions - Bitcoin Wiki* (March 15, 2018), https://en.bitcoin.it/wiki/Irreversible_Transactions.

[25] *Scrypt - Litecoin Wiki - Litecoin.info* (February 12, 2018), https://litecoin.info/index.php/Scrypt.

[26] *Ethash · ethereum/wiki Wiki - GitHub*, https://github.com/ethereum/wiki/wiki/Ethash.

[27] *BITMAIN*, https://shop.bitmain.com/product/detail?pid=00020180314213415366s4au3Xw306A4.

[28] *Monero Cryptonight V7 - GitHub*, https://github.com/monero-project/monero/pull/3253/files/e136bc6b8a480426f7565b721ca2ccf75547af62.

[29] Alex Biryukov, Daniel Dinu, and Dmitry Khovratovich, *Argon2: the memory-hard function for password hashing and other applications* (December 26, 2015), https://password-hashing.net/argon2-specs.pdf.

[30] Dan Boneh, Henry Corrigan-Gibbs, and Stuart Schechter, *Balloon Hashing: A Memory-Hard Function Providing Provable Protection Against Sequential Attacks* (2017), https://eprint.iacr.org/2016/027.pdf.

[31] *GitHub - A Programmatic Proof-of-Work for Ethash*, https://github.com/ifdefelse/ProgPOW.

[32] *GitHub - hyc/randprog: Randomly generate a C (or javascript) program*, https://github.com/hyc/randprog.

[33] *GitHub - curie-kief/cryptonote-heavy-design: Cryptonote Heavy deign essay*, https://github.com/curie-kief/cryptonote-heavy-design.

[34] Surae Noether, Sarang Noether, and Adam Mackenzie, *A Note on Chain Reactions in Traceability in CryptoNote 2.0* (2014), https://lab.getmonero.org/pubs/MRL-0001.pdf.

[35] *Github Comment - EABE/Knacc Attack*, https://github.com/monero-project/monero/issues/1673#issuecomment-312968452.

[36] *I2P's Threat Model - I2P*, https://geti2p.net/en/docs/how/threat-model#harvesting.

[37] *Deep packet inspection - Tec Gov*, http://tec.gov.in/pdf/Studypaper/White%20paper%20on%20DPI.pdf.

[38] Philipp Winter and Stefan Lindskog, *How China Is Blocking Tor* (2012), https://arxiv.org/abs/1204.0447.

[39] *Egypt Quietly Blocks VOIP Services Skype, Whatsapp - TorGuard* (October 26, 2015), https://torguard.net/blog/egypt-quietly-blocks-voip-services-skype-whatsapp.

[40] *GitHub - Yawning/obfs4: The obfourscator (Development mirror)*, https://github.com/Yawning/obfs4.

[41] David Vorick and Luke Champine, *Sia: Simple Decentralized Storage* (2014), https://sia.tech/whitepaper.pdf.

[42] Adam Back, *Hashcash - A Denial of Service Counter-Measure* (2002), http://www.hashcash.org/papers/hashcash.pdf.

[43] Colin LeMahieu, *RaiBlocks: A Feeless Distributed Cryptocurrency Network*, https://raiblocks.net/media/RaiBlocks_Whitepaper__English.pdf.

[44] *Lazy Masternodes: do you actually have to do any work to get paid/vote?*, https://www.reddit.com/r/dashpay/comments/5t6kvc/lazy_masternodes_do_you_actually_have_to_do_any/.

[45] *ACNC template constitution for a charitable company*, https://acnc.gov.au/CMDownload.aspx?ContentKey=2efea0fa-af4f-4231-88af-5cffc11df8b7&ContentItemKey=6046cbc5-d7fd-4b6b-93ba-c8e3114b07ba.