Loki

秘匿化トランザクションと分散型コミュニケーション.

Kee Jefferys, Simon Harman, Johnathan Ross, Paul McLean

バージョン 3

2018 年 7 月 13 日

Abstract

プルーフ・オブ・ワーク (proof of work) とプルーフ・オブ・サービス (proof of service) を組み合わせることにより、フルノードのオペレーションに経済的インセンティブを付与するというユニークな手法を提案する。Loki はこのインセンティブ化されたノードを活用することで、秘匿性を有するセカンド・ルーティング・レイヤー (secondary private routing layer) を構築するものであり、ノードの最小要件については swarm flagging という新たな手法によって監視・強制がなされる。また、Loki はその基盤として Monero のソースコードの修正版を利用しており、全てのトランザクションにおいては高度なプライバシーが保証されるものである。

このホワイトペーパーでは Loki で使用されている技術について概説するが、当プロジェクトの発展とともにここで言及する技術は変わり得ることが見込まれる。将来、大幅な変更や更新がなされた際には、それらを反映した改定稿をリリースする。

1 Introduction

デジタルなコミュニケーションや取引におけるプライバシーの需要はますます高まっている。人々のデータはかつてない規模で収集・処理されており、それらが取引されている。ブラウジングデータや E メールの内容、クレジットスコアや消費習慣に至るまでの全てが世界的な大企業や国家組織によって収集され取引されているためだ。Loki の狙いは、ユーザーがプライバシーを維持した上で取引やコミュニケーションを行うことを可能にするための、検閲に強いツールを提供することである。Bitcoin はプライバシーの保証とともに現れたが、結果的にはこれまで以上にトレーサビリティが向上するものであった。Chainalysis や BlockSeer のような企業は、特定のトランザクションを追跡するために、Bitcoin のブロックチェーンの透明性を利用している [1]。

Loki は、現時点で最も秘匿性が高くプライベートなトランザクションネットワークを確立している暗号通貨の 1 つである Monero を基に構築している [2]。しかし、我々は

Monero に固有の欠点があることも認識している。Monero のトランザクションは Bitcoin のトランザクションよりも桁違いに大きいため、帯域幅・処理・ディスク容量を大きく必要とする。ネットワークが成長するにつれてこの点が Monero のノードオペレータへの大きな負担となるが、彼らのネットワークへの貢献に対しては何のインセンティブも報酬も発生していない。そのため、Monero においてノードを維持することは、ランニングコストを踏まえると報われない行為となっている。ノードへの報酬制度（サービスノード制度）を導入し、ノードオペレータに経済的インセンティブを付与することができれば、この負担を軽減することができる。

サービスノードはまた、適切にインセンティブを与えられることで、ノードを利用して他の広範なプライバシーを保持した機能を提供することも可能となる。主要な点として、サービスノードネットワークによって、ユーザーはデータパケットを匿名で送受信することが可能となる。このプライベートコミュニケーションは、Tor や I2P [3][4] と類似の特性を持ちながらも、新たにシビル攻撃耐性をも有しているミックスネット (mixnet) として、各サービスノードによって中継される。加えて、このコミュニケーションネットワークは、*Loki Messenger* と呼ばれる分散型のエンドツーエンドの暗号化されたメッセージングサービスのバックボーンとしても使用され、第三者を信用することなく、ユーザー間で直接コミュニケーションを取ることが可能となる（両当事者が同時にオンラインである必要もない)。

Loki は、プライベートなやり取りを可能にする単一障害点のない媒体というだけではなく、分散性と秘匿性を有するインターネットサービスのプラットフォームなのである。

2 Basic Parameters

Loki difficulty target (blocktime)	120 Seconds
Difficulty algorithm	Zawy LWMA [5]
Hashing algorithm	CryptoNight Heavy
Elliptic curve	Curve25519 [6]

3 CryptoNote Elements

フルノードのインセンティブスキームはどの暗号通貨上にも実装し得るが、トランザクションに対するプライバシー水準の高さを踏まえ、Loki は Monero のソースコードを用いている。Monero は CryptoNote プロトコルを発展させたものであり、リング署名 (Ring Signatures)、ステルスアドレス、RingCT を用いることで、ユーザーはもっともな否認性を保持しながら、トランザクションに署名し送金額を難読化させることができている [7]。

Loki のエコシステムがプライバシーを維持するためには、内部経済を支える媒体を提供するだけでなく、Loki の独立した各層（レイヤー 1 とレイヤー 2）の相互作用が発生した瞬間を解析されるリスクを最小限に抑えることも重要である。例えば、レイヤー 1 のトランザクション型サービスを利用する場合に、ユーザーはレイヤー 2 におけるプライバシーの保証も決して失うべきではなく、その逆もまた同様である。

3.1 Ring Signatures

リング署名は、署名者のうちの 1 人だけが実際の送信者であるトランザクションに、複数の署名可能者を含めたリングを構築することによって機能する。Loki はリング署名を使用して、トランザクションの真の出力履歴を難読化する。すべての Loki 取引（ブロック報酬取引を除く）にはリング署名が必須となり、Loki ブロックチェーンにおいては固定リングサイズ 10 が適用される。これは、各入力に対し、1 つの真の出力を含む 10 の出力が発生することを意味する（6.3 参照）。

3.2 Stealth Addresses

Loki はステルスアドレスを利用することで、受信者の真の公開鍵が決してその取引にリンクされないようにしている。Loki のトランザクションが送信されるたびに、ワンタイム・ステルスアドレスが作成され、このアドレスに資金が送信される。Diffie-Hellman 鍵交換を使用すると、トランザクションの受信者はこのステルスアドレスの秘密鍵を利用することができ、実際の公開アドレスを明らかにすることなく資金の所有権を獲得することができる [8]。ステルスアドレスは受信者にトランザクションの保護を提供する、Loki の中核となるプライバシー機能である。

3.3 RingCT

RingCT は取引金額を難読化する方法として Monero Research Lab によって提案された [9]。現時点の RingCT 技術では、Pedersen コミットメント (Pedersen commitments) を利用することで、送信されるトランザクションの金額が $0 \backsim 2^{64}$ であることを証明する範囲証明が使用される。この範囲では、負の値ではない量が送付されたことが判明するのみであり、実際に送付された金額が特定されることはない。近年では、トランザクションサイズの大幅な縮小のために、RingCT の従来の範囲証明の代わりに bulletproofs を実装することが提案されている [10]。Loki においても bulletproofs を利用することで、ノードが格納および中継するために必要な情報を減らし、スケーラビリティを向上させるものである。

4 Service Nodes

Loki は CryptoNote プロトコル（7 章参照）に斬新な変更を実装するものであり、Loki のネットワーキング機能とスケーラビリティの多くは、サービスノードと呼ばれる一連のインセンティブ化されたノードによって可能となる。サービスノードを運営するためには、相応の量の Loki をタイムロックし、最低要件を満たす水準の帯域幅とストレージをネットワークに提供する必要がある。そして、Loki サービスノードのオペレータは、そのサービス提供の代償として、各ブロックからブロック報酬の一部を受領する。

結果として構築されるネットワークは市場ベースのシビル攻撃耐性を有し、ミックスネットとプライバシーを中心とした既存のサービスに関するさまざまな問題を解決する。このシビル攻撃耐性は需要と供給の相互作用に基づいているため、特定の攻撃者が Loki のレイヤー 2（セカンドレイヤー）のプライバシーサービスに重大な悪影響を及ぼすほどの十分な Loki を保有することを防いでいる。シビル攻撃耐性ネットワークが暗号経済学から導かれることを初めて理論化したのは DASH である [11]。攻撃者が Loki を累積するにつれて、市場に循環している Loki 供給量が減少するため、需要側の圧力が増

加し、Loki の価格は上昇する。これが続くにつれて追加の Loki を購入するコストが増加していくため、攻撃には法外なコストが必要となる。

この市場経済的に基づくシビル攻撃耐性を達成するために、Loki は市場に循環する供給量の積極的な抑制を行う。特に、供給曲線と担保要件（ノードにおける Loki ロック量の設定）に関しては、市場供給に対して十分な量の Loki がロックされるようにした上で、ノードオペレータにはロックに見合った利益が提供されるようにし、シビル攻撃耐性が確実となるように適切に設計されなければならない。

4.1 Block Reward

Loki におけるブロック報酬の分配は、ブロック作成及び取引作成において堅牢で十分に研究されたシステムであるプルーフ・オブ・ワークを通じて行う。マイナーは、取引を集めてブロックに書き込み、その対価として手数料を徴収する。Loki のコンセンサスルールにおいて、各ブロックには複数の報酬出力が含まれ、そのうちの 1 つがマイナーに分配される。

Mining Reward:
取引手数料に加え、ブロック報酬の 45%が新たなブロックを構築するマイナーに与えられる。

Service Node Reward:
各ブロックの 2 番目の出力（総報酬の 50%）は、1 つまたは 2 つのサービスノードに送られる（6.3 参照）。サービスノードは、報酬を最後に受信してからの経過時間（または登録されてからの時間）に基づいて報酬を受け、長く待っていたノードが優先される。サービスノードがネットワークに登録される度に、待機列（待ち行列）の最後尾に配置される。サービスノードが良好なサービスを維持し、スウォーム・フラグ（7.3 参照）によって待機列から除外されることがなければ、サービスノードは待機列のより上位に緩やかに遷移していく。待機列の最前列近辺のノードは報酬の対象となり、報酬を付与されたノードは再び待機列の最後尾に戻り、徐々に上位に来るまでの作業を行う。

Governance Reward:
ブロック報酬の残りの 5%部分はガバナンス・オペレーションに分配される（9 参照）。そのうち 3.75%が Loki Foundations のアドレスに送信され、残りの 1.25%は資金調達ブロックへの分配分として確保される（9.2.3 参照）。

4.2 Verifiable Collateralisation

サービスノードは、必要な担保を保持していることをネットワークに証明する必要があるが、Loki の設計上、秘匿性がそれを困難にしている。具体的には、パブリックアドレスの残高を監査できないことや、トランザクションを表示するためのビューキー (viewkeys) を使用できないことである。

Loki で新たに導入するタイムロックの手法は、ブロックチェーンが一定のブロック数 (block-height) に達するまで Loki コインをタイムロックするということである。定められたブロック数に達するまで、Loki ネットワークはタイムロックされた Loki についての出力を無効とする。Loki はこのプロセスを利用して、特定のサービスノードによって残高が保持されていることを証明し、担保の流出を防止する。

サービスノードの登録時に、オペレーターは、最小 21,600 ブロック（約 30 日）が経過した後に必要量がロック解除されるタイムロック出力を作成する。また、トランザクショ

ンの別の領域に、サービスノードオペレータは、その報酬を受け取ることができる Loki アドレスを入力する。このアドレスは、swarm 投票などのサービスノードによるオペレーションのための公開鍵としても使用される。サービスノードに関するトランザクション（金額・宛先）が開示されることで、匿名性に余分な情報が付加されてしまうため、ウォレットはサービスノードとの混合使用を避けることもできる。

各ノードのサービスノードネットワークへの参加に際し、他のノードは新規ノードの担保が必要額に一致しているのかを、要求される担保額の経時的減少を踏まえ、個別に検証しなければならない。担保のためのロックは 30 日後に期限が切れるが、ウォレットに自動的に再担保が適用されるような設定にすることも可能である。

5 Lokinet

オニオン・ルーティング・プロトコルを用い、分散ネットワークによる複数のノードをホップとして利用することで、データパケットの宛先と発信元を難読化した通信経路を形成することが可能となる。Loki ネットワーク上のサービスノードでは、低レイテンシのオニオン・ルーティング・プロトコルを使用することで、Lokinet と呼ばれる完全に分散されたオーバーレイネットワークを形成する。ネットワークは特定の管理者の信頼性に依存することはなく、その状態は完全にブロックチェーンに依拠するものである。ユーザーは、個々のサービスノードに接続し、パケットをルーティングするための双方向のパスを作成することができる。このネットワークによって、内部でホストされた *SNApps* と呼ばれるサービスにアクセスすることができる（6.2 参照）。ユーザーは、サービスノード・イグジット機能を使用して、IP アドレスを公開することなく外部のインターネットをブラウズすることも可能である（6.3 参照）。

5.1 Low Latency Anonymous Routing Protocol (LLARP)

サービスノードにおけるすべてのアプリケーションは、秘匿化ルーティングプロトコルに基づいており、それによって各サービスノードが近隣ノードと通信する方法が定められている。Loki は、*LLARP* [12] と呼ばれる新しいルーティングプロトコルを提案する。これは、Tor と I2P の間のハイブリッドとして設計されており、既存のルーティングプロトコルを発展させた特性を提供する。LLARP は Loki サービスノードネットワーク上で動作するように特別に構築されており、このアーキテクチャを考慮して LLARP はあらゆる点で最適化されている。LLARP の目的を理解するためには、既存のルーティングプロトコルの分析を行い、LLARP がどのようにそれらを改善したのかを理解するのが最良である。

The Onion Router (Tor)

近年では Tor が最も知名度のある匿名のミックスネットである。Tor ネットワークは高水準の検閲抵抗を維持しており、インターネットのプライバシーを保護するための貴重なツールであることが証明されている。しかしながら、Tor は階層的なネットワークではあっても、分散型のネットワークではない。Tor は、ディレクトリ・オーソリティ (directory authority) のグループに依存しており、それは Tor Foundation に近いボランティアのグループによって運営されている集中管理されたサーバーに依拠したものである [13]。これらのディレクトリ・オーソリティは、2 つの主な機能を担っている。第 1 に、ネットワーク内のノードの状態に関する信用されたレポーターとしての役割である。Tor ユーザー（またはリレー）が最初にネットワークに接続すると、10 人のハードコードされたディレクトリ・オーソリティの 1 つに接続される。これらのディレクトリ・オ

ーソリティは、コンセンサスと呼ばれるファイルをユーザまたはリレーに提供する。このファイルには、Tor ネットワーク上の現在動作しているすべてのリレー、ガード・ノード、及びイグジット・ノードのリスト（ブリッジを除く）が表示される。第 2 に、ディレクトリ・オーソリティは、各リレーがネットワークに提供できる帯域幅を測定している。彼らはこの情報を使用してリレーをカテゴリに分類し、ノードがリレーやガード・ノード、またはイグジット・ノードとして動作できるかどうかを決定している。

この高度な集中化によって、Tor が脆弱となる障害点が生じている。Tor は 2014 年、ディレクトリ・オーソリティ・サーバー停止の脅威に関する信憑性の高い情報を提示されている [14]。米国、ドイツ及びオランダのディレクトリ・オーソリティを停止するのに必要なのは、10 個のディレクトリ・オーソリティ・サーバーのうち 5 つをシャットダウンするだけで十分であり、これがなされるだけで Tor ネットワークは非常に不安定になり、新しいリレーはネットワークとの相互通信能力を失うというものである。

また、Tor は TCP を介した通信のみを許可するため、Tor の通信方法も限られている。IP を介した Tor は可能なものの、UDP ベースのプロトコル（VoIP など）はサポートされていない。

Invisible Internet Project (I2P)

I2P では、ミックスネットのアーキテクチャに対して異なるアプローチを採用している。ディレクトリ・オーソリティへの信用に依存する代わりに、ネットワークハッシュテーブル（DHT）を参照することにより、より高いレベルの信頼性を維持している [15]。また、I2P は、TCP と UDP の両方のトラフィックを許可し、より広い範囲のプロトコルのやり取りをサポートしている。しかし、I2P は開発プロセスが安定しておらず、時間の経過とともに特に暗号化などに関する技術的な課題が山積している。I2P は 2048 ビットの ElGamal を使用しているため、楕円曲線操作とは対照的に暗号化と復号化が遅くなっている。ElGamal から移行する計画は I2P のロードマップに存在しているが、開発は進捗していない。

さらに、I2P ではイグジット・ノードが正式にサポートされていないため、ネットワーク上のトラフィックの大部分が *Eepsites* と呼ばれる内部でホストされた Web サイトへのアクセスとなっている。これにより、開かれたインターネットへのアクセスを目的として I2P の匿名化ネットワークを利用しているユーザーの利便性を大幅に損なっている。

加えて、I2P の構築手法ではネットワークに接続するユーザーの大半がルータとなっていることから、結果としてネットワークに高速パスを構築するための十分な帯域幅が不足していることが問題となっている。ミックスネットのネットワーク速度は、各経路の最も能力の低いノードがボトルネックになるため、低性能のユーザーが I2P においてリレーする結果、全体的なパフォーマンスを低下させている。

最後に、I2P はパケット交換（回線交換ではなく）ネットワークを提供している点で Tor と異なる。I2P では、すべてのトラフィックが通過する 1 つの長期間経路を確立する代わりに、通信中の各パケットがネットワークの異なるルートを使用できるような複数の経路を確立する。これにより、I2P はネットワークの輻輳やノードの障害を透過的にルーティングできる。

I2P と Tor はともにシビル攻撃を完全に緩和できてはいない。何らかの十分な動機を持ち、大量のリレーを取得するために時間と資本を投下できる攻撃者は、ネットワークのユーザーのプライバシーを侵害する解析を実行することができる。この攻撃の実効性は、攻撃者が操作するイグジット・ノード、リレー、及びガード・ノード数に比例する [16]。そして Tor と I2P は、時間と資金をノード運用に費やしているボランティアによって運営全般が成されているのである。我々は、利他主義ではなく経済的インセンティブで構

築されたネットワークの方が、より信頼性の高いサービスを提供し、攻撃に対するより大きな耐性を実現できるものと思慮する。

LLARP

LLARP では、ディレクトリ・オーソリティを必要とせずに動作し、代わりにブロックチェーン・ステイキング・トランザクションから構築された DHT に基づき、サービスノードをネットワーク内のルータとして動作させることができる。帯域幅は、DHT によって監視や記録されるのではなく、swarm (6.1.1 参照) によって各ノードが適切な帯域幅を有しているかを測定され、審査される。

Open Systems Interconnection モデル（OSI model）では、LLARP は秘匿化ネットワーク層の提供のみを試みる。これは、より広範なインターネットプロトコルをサポートし、イグジット・ノードによってユーザー・データグラム・プロトコル（UDP）トラフィック [17] を通すときにファイルディスクリプタを格納するオーバーヘッドを最小化することを意味する。さらに、LLARP は、トンネルベースのルーティングの代わりにパケット交換ベースのルーティングを採用しているため、ネットワーク全体の負荷分散と冗長性の向上が図られている。

Lokinet のエンドユーザーがパケットのルーティングをすることは要求されていない (許可もされていない）が、これは Lokinet においてサービスノード運用を行うためには十分な資本支出が要求されるためであり、このことは Lokinet がシビル攻撃に晒される可能性が低下することを意味している。

6 Loki Services

マイナーがハードウェアに投資を行うのと同様に、各サービスノードオペレータは、サービスノードの運用開始に伴い Loki コインが凍結される。このノードにおける資本凍結は、以下の２つの目的に寄与する。

1. 全てのサービスノードオペレータは、ネットワークの成功に大きく賭けることとなる。いずれかのサービスノードオペレータが、ネットワークに不十分なパフォーマンスを提供したり、不正を行ったりすることは、ネットワーク内の自身の投資を減価させるリスクを伴う。
2. さらにアグレッシブな措置として、ネットワークが不正なノードへの報酬を効果的に制限することができれば、不正なノードは報酬を失うだけではなく、担保がロックアップされている時間をも失うこととなる。

上記の点を踏まえ、悪質なノード（7.3 参照）に対して積極的な罰則を適用することができれば、サービスノードのグループを作成し、コンセンサスを得るためにブロックチェーンのステートを照会することや、特別なオフチェーンノードの動作を強制するといったこともできる（swarms 6.1.1 参照)。Loki において、この挙動はネットワーク処理とストレージの両方の動作に関連する。これらのオフチェーン動作は、Loki サービス (*Loki services*) として、これら望ましい特性を活用するユーザー向けアプリケーションのバックエンドとして統合される。

6.1 Loki Messenger

Loki ネットワーク上で開発され展開される最初の Loki サービスは、ロキ・メッセンジャー (*Loki Messenger*) と呼ばれる、分散化・暗号化されたエンドツーエンドのプライベ

ート・メッセージング・アプリケーションである。

暗号化された秘匿メッセージを送信するためのプラットフォームを提供するエンドツーエンドのメッセージング・アプリケーションはすでに存在しているが、それらは中央管理されたサーバーに依存しているために、標的にされたりブロックされたりシャットダウンすることが起こり得る [18][19]。また、これらの中央管理されたサービスモデルは、ユーザが電話番号または他の識別情報を登録し、ユーザの IP アドレスを介して直接接続する必要があるため、通信関係者を介して匿名性を脅かすリスクが非常に高い。これらの個人情報は、サーバーからデータ漏洩または合法的プロセスによって抽出され、ユーザーの意思に反して使用される可能性がある。Loki ネットワークにおけるサービスノード・アーキテクチャを活用することで、Signal のような中央管理に基づく一般的な暗号化メッセージングアプリと同様のサービスを、より高次のプライバシーと検閲耐性を持たせた上で実現することができる。

6.1.1 Messenger Routing

Loki ネットワーク上のメッセージルーティングは、受信ユーザーがオンラインかオフラインかによって異なる。両方のユーザーがオンラインの場合、メッセージをサービスノードに保存する必要がないため、より高い帯域幅の通信が可能となる。

Loki では、公開鍵 (public key) は長期暗号鍵 (long-term encryption key) とルーティングアドレスの両方として機能する。最も単純なケースでは、中間者攻撃 (man-in-the-middle attack) 対策を確実にするために、この鍵を帯域外で交換する必要があるが、そのような交換は、直接または別の安全な交換方式を介して行われるべきである（6.1.2 参照）。

Online Messaging

アリスはボブの公開鍵を知った後、まず彼がオンラインであると仮定し、彼へのパスの作成を試みる。アリスは、任意のサービスノードの DHT に照会することによってこれを行い、ボブの公開鍵に対応する誘導セットを取得する。LLARP における誘導セットでは、各ユーザー間においてイントロデューサー (*introducers*) のリストが共有され、これらのイントロデューサーを介して経路を確立することができる。(Figure 1 の例では) ボブのイントロデューサーを介した際、アリスと目的地（ボブのイントロデューサー）との間に、3 つのランダムなサービスノードが仲介ホップとして選出されている。このようにアリスとボブの間でメッセージを送信できるパスが確立され、さらに OTR（6.1.2 参照）による認証も成功すると、アリスとボブは高水準のプライバシーを維持した通信が可能となる。

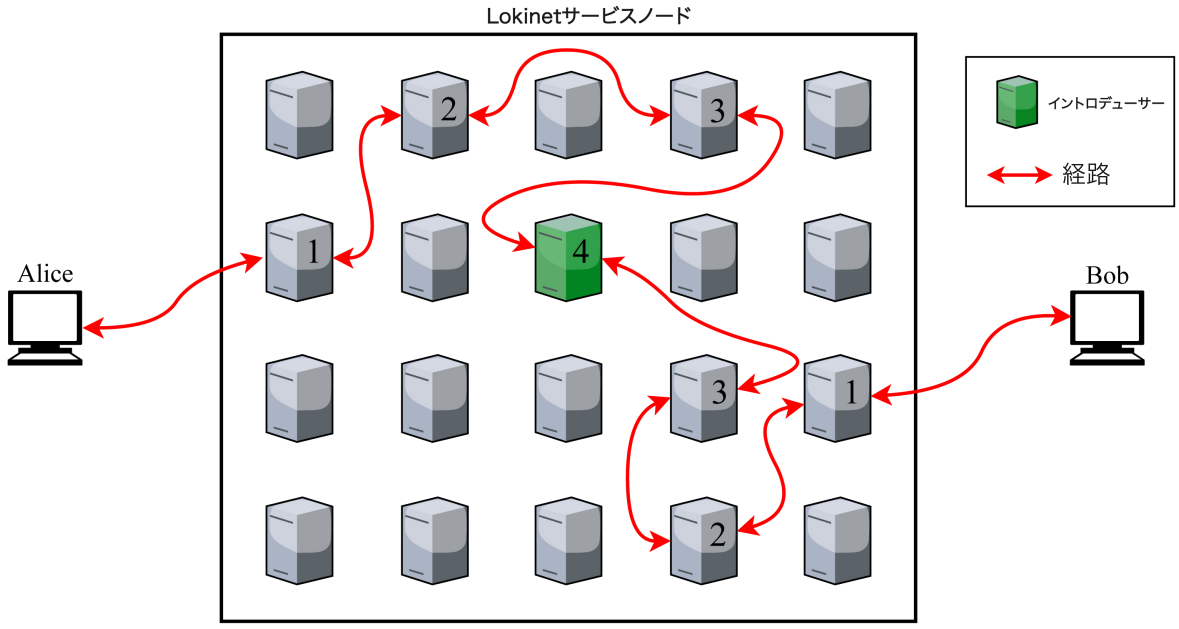

Figure 1: アリスとボブが、ランダムなサービスノードを用いてネットワークの通信パスを確立するオンラインルーティングの簡易図。

Offline Messaging

アリスがボブからの応答を受信できなかった場合、アリスはオフライン・メッセージング・プロセスに移行する。オフラインルーティングでは、Postal Services over Swarm (PSS) の修正バージョンを利用する [20]。Swarm（スウォーム）とはサービスノードのグループであり、グループを構成する各ノードの公開鍵とステークトランザクションが発生した際のブロックのハッシュの両方に基づいてロジカルに構成されるグループである。全てのスウォームは swarmID を持ち、9 つのノードで構成される。ボブにメッセージを送信するために、アリスはボブの公開鍵を使用してボブの属するスウォームを導出できる。この情報に基づき、アリスはボブの属するスウォーム内のサービスノードネットワークをランダムに経由させることによって匿名でメッセージを送信できる。サービスノードは、自身のスウォームに向けられたメッセージを受信すると、スウォーム内の他の 8 つのノードにそのメッセージを配布しなければならず、加えて、全てのノードは、割り当てられた Time-to-live (TTL) の期間においてメッセージを保持しなければならない（8.3 参照）。ボブがオンラインになると、彼は復号可能なメッセージの受信状況を、自身のスウォーム内の 2 つのノードに照会することができる。また、オフラインメッセージングは、各メッセージ送信に簡易な作業証明（proof-of-work）を要することで、スパム攻撃耐性を有している（7.2 参照）。

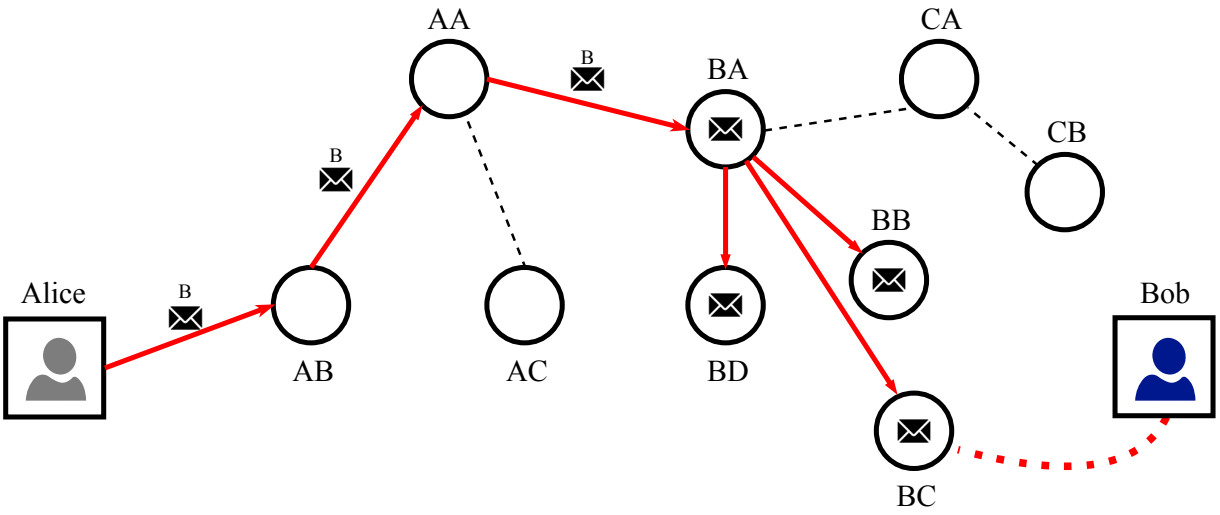

Figure 2: アリスがボブにメッセージを送信した際、ボブはスウォーム B に属している。ボブがオンラインになると、彼は属するスウォーム内のランダムなノードに照会し、アリスのメッセージを受信する。

6.1.2 Messenger Encryption and Authentication

メッセージチェーンが確立されると、Loki Messenger は Perfect Forward Secrecy (PFS) と Deniable Authentication (DA) を施す。PFS と DA は、Off The Record (OTR) メッセージングプロトコルの重要な概念である [21]。Signal や WhatsApp といった中央集権化されているサービスにおいても、暗号化機能を用いて OTR 保護を維持している。Loki における OTR の実装については、既存の分散化された P2P インスタントメッセージプロトコルである Tox プロトコルに基づいてモデル化されており、十分に監査された暗号技術ライブラリである NaCl ライブラリ [22] を用いたものである。

PFS によって、長期暗号鍵が公開されてしまった場合の攻撃耐性を持つことができる。各セッションでは毎回新しい共有暗号鍵が使用されるため、もしいずれかのセッションのキーが明らかになったとしても、メッセージチェーン全体が危険にさらされることはない。仮に第三者がメッセージチェーンを解読しようとしても、個々のセッションの全てのキーを入手する必要がある。PFS を用いることによって、Pretty Good Privacy (PGP) のようなメッセージチェーン全体の復号に対して一つの長期鍵しか必要としない暗号化の手法に対して、Loki Messenger は非常に解読が困難なものとなっている。

DA は、二者間においてやりとりされる各々の新しいメッセージについて、その真の送信者であることをお互いに証明する機能である。ただし、第三者にとってはメッセージの真の送信者を特定することはできない。これは、DA の適用において、各セッションの後に Message Authentication Codes（MACs）が公開されていることを利用し、第三者にとって見かけ上は正当な送信者のパブリックアドレスから発信されたかのようなメッセージを作成することが可能となるためである。これが適切に実装されていることで、いかなる第三者においても特定のメッセージの真の送信者が誰であるかを証明することは不可能となる。

User Authentication

中間者攻撃に対する保護を確実にするためには、ユーザーの認証が重要である。例えば、ボブがアリスからのメッセージを待ち受けていても、彼女の公開鍵が何であるのかをまだ分かっていない場合、第三者（イブ）はボブに対し、アリスだと見せ掛けたメッセー

ジを送ることができる。このため、ユーザーは個人情報を共有する前にお互いを認証する必要がある。

Pidgin や他の OTR メッセージングサービスと同様に、Loki Messenger は Pre-Shared Key （PSK）認証を使用する。ユーザーは、PSK の確立に複数の選択肢を有する。帯域外で確立することも可能であり、Loki Messenger の PSK（第三者が答えを知りえないような質問を利用）を用いてもよい。Loki は、Pidgin の暗号認証プラグインの修正版をベースとした PSK 認証を実装する [23]。

6.2 SNApps (Service Node Applications)

SNApps の機能は、Tor における隠されたサービス (hidden services) に類似のものである。それらはユーザーに、ミックスネット環境内で完全にやり取りを行う方法を提供し、外部にホストされたコンテンツにアクセスするときよりも匿名性をさらに高める。SNApps を使用すると、サーバーやユーザーサイドの匿名性を維持しながら、マーケットプレイス、フォーラム、内部告発サイト、ソーシャルメディア、その他のインターネットアプリケーションを自身のマシンやサーバーにセットアップしホストすることができる。これにより、ネットワークの範囲が大幅に広がり、ユーザーは Lokinet 内で有意義なコミュニティを構築することができる。

SNApp のオペレーターは伝統的なサーバークライアントモデルを使用しているものの、サービスノードが仲介者になるという点が Lokinet における重要な違いである。SNApp をネットワークに登録するときには、DHT の記述子を更新しなければならない。この記述子には、ユーザーが SNApp へのパスを形成するための接続ができるような特定のサービスノードに関する様々なイントロデューサーが含まれる。これらのパスが設定されると、ユーザーは互いにネットワーク内の所在を知らせることなく SNApp に接続することができる。

6.3 Exit Nodes

イグジット・ノードを使用すると、ユーザーはより広範なインターネットへのリクエストを行い、その結果をミックスネット経由で受けることができる。適切に使用すると、ユーザーは IP アドレスをサーバーに公開せずに、秘匿性を保ったままインターネットをブラウズすることが可能となる。

Loki の拡張ユーティリティとして、イグジット・ノードのオペレーションは不可欠であるが、全てのサービスノードオペレーターをイグジット・ノードとして動作させることは好ましくない。イグジット・ノードのユーザーが悪意のあるアクティビティを実行し、そのプロキシとしてノードを使用した際に、イグジット・ノードのオペレーターが法的リスクにさらされる可能性があるためである。イグジット・ノードは単にインターネットからエンドユーザーにトラフィックを中継しているだけであるが、Digital Millennium Copyright Act (DMCA) の適用や、ハッキングの試みの元にされてしまうことがしばしば起こりうる。ほとんどの管轄区域においては、宥恕規定 (safe harboring laws) によってイグジット・ノードのオペレーターは保護されるはずだが、インターネットのサービスプロバイダが法的リスクを避け、サーバー上でのサービスノードのトラフィックが遮断されてしまう可能性がある。

起動に際して、サービスノードにはリレーフラグが割り当てられ、パケットのルーティングが Lokinet 内に制限されるため、外部のインターネットへのリクエストが行われることはない。イグジット・ノードとしての運用をオペレーターが望む場合には、事前選

択 (opt-in) をしなければならない。それによって、オペレーターは追加のリスクに対する理解表明と追加のスウォームテストの提出がなされる（8.3.1 参照）。

イグジット・ノードを選択すると、ブロック報酬受領の際、オペレーターは通常のリレー報酬の 2 倍を得ることができる。このインセンティブは、イグジット・ノードの運用に際し、オペレーターへ十分な経済的インセンティブを提供するとともに、イグジット・ノード・ネットワークを標的としたシビル攻撃耐性に寄与するものである。この点は、イグジット・ノードの比率が少ないことに窮している Tor における脆弱性でもある。

6.4 Remote Nodes

いかなる暗号通貨のネットワークにおいても、ブロックチェーンの完全なコピーを保持することは、多くのユーザーにとって現実的ではなく実用的でもない。Bitcoin や Ethereum では、ブロックチェーンのコピーを保持しているパブリックのフルノードに接続し、トランザクションを照会してネットワークに送信することができる。これは、Bitcoin や Ethereum のフルノードが、ユーザーの公開鍵をターゲットとして持つトランザクションのブロックチェーンを効率的に検索できるためである。

CryptoNote 通貨の構造上、パブリックのフルノード（リモートノードと呼ばれる）には更に負荷がかかる。ユーザーがリモートノードに接続するときは、ローカルマシンに（ウォレット作成時または最後にチェックしたブロック以降についての）全てのブロックを一時的にダウンロードしなければならず、各トランザクションについても、プライベート・ビュー・キーから生成できる公開トランザクション・キーを用いての確認が必要である。このサービスへの対価がないことを考慮すると、ユーザーが同期サービスを用いる際に、低処理力のクライアントによって動作が妨げられてしまう可能性がある。CryptoNote のモバイルウォレットは信頼できないことが多く、ブロックチェーンをスキャンしたり、トランザクションを送信したりするための信頼性の高い接続を確立するにあたって、リモートノード間で複数回切り替えなければならないことがある。

さらに、一般的なノードの中に悪意のあるリモートノードオペレータが紛れていた場合、特定のトランザクションをブロードキャストするときにユーザーの IP アドレスを記録することが可能である。この攻撃によって、実際のトランザクション情報が明らかになることはないが、特定の IP アドレスをトランザクションとリンクさせることで実世界のアイデンティティとのリンクを確立し、プライバシーを侵害することができてしまう。

Loki では、各サービスノードを一般ユーザーが利用できるリモートノードとしても動作させることで、これらの問題を回避している。サービスノードは、ブロックチェーンの完全なコピーを保持しており、広帯域幅を有するノードによって広範なネットワークが形成されているため、難なく一連の処理を行うことができる。サービスノードをリモートノードとして使用することにより、各々が所有できるリモートノードネットワークの量は経済面から制限されることとなり、結果として悪意のあるノードオペレータが収集できるデータ量についても同様の限界が生じる。

6.5 Blink

一般的なブロックチェーンシステムにおいて、トランザクションの承認時間とは、トランザクションがブロックに含まれるのに掛かる時間のことである。ただし、競合するマイナー、保留ブロック (withheld blocks)、Finney 攻撃を踏まえると、受信者が特定のトランザクションが確定したと見做すためには、当該トランザクション後に多数の追加ブロックが作成されるのを待つのが通例となっている [24]。各ブロックチェーン固有の様

々な要素に依るものの、このプロセスには通常 10 ー 60 分掛かるため、商品販売やサービス提供の際にそのような承認を待たなければならないことは、売り手や消費者にとって不便である。

しかし、Loki のサービスノードアーキテクチャにおいては、瞬時のトランザクションが実現できる。“Blink”によって、ブロックに含まれる前に Loki メインチェーン上で行うのと同様のトランザクション確認を行い、送信者と受信者の双方にトランザクションの有効性を保証し、受信者を二重支払いから保護することが可能となるためである。

Blink は、DASH の InstantSend と類似の手法で動作する。各ブロックにおいて、サービスノード・スウォームは確定的に選択され、トランザクションの有効性の確認や、トランザクションが二重支払いに使われないようにロックする承認者として機能する。ただし、(DASH で行われているような) トランザクションで使用されている未使用出力がロックされるのではなく、その代わりにキーイメージがロックされる。キーイメージとは、リング署名の各未使用出力に添付されるユニークキーである。即時承認を提供するために、Blink は選択されたスウォームに権限を与えて、トランザクションがブロックに含まれるまで、出力に関連するキーイメージをロックする必要があることをネットワークに知らせる。仮に同一の未使用出力についての二重支払いが試みられた場合、同一のキーイメージが生成されることから、スウォーム及びネットワーク全体によって拒絶される。

ユーザーは、通常より高い手数料を支払うことで、数分ではなく数秒で承認される Blink トランザクションを送信することができる。これにより、対面支払いの実用性が増し、オンライン支払いとの統合も容易になるため、Loki の様々な新しいユースケースが開かれる。一方で、このプロセスを通し、Loki 固有のあらゆるプライバシー特性が犠牲になることはない。

7 CryptoNote Alterations

通貨としては、Loki は類似の CryptoNote コインと機能的に似ている。しかしながら、サービスノードの追加とそれがもたらす機能が重要な相違点である。

7.1 ASIC Resistance

特定用途向け集積回路（ASIC）は、単一の機能専用に設計されたコンピュータチップであり、マイニングにおいては、特定のハッシュアルゴリズムを計算するために ASIC が使用されている。他のあらゆるマイニング方法より高効率であること、特定の企業によって製造されていること、ハードウェアの特殊性のために流通経路が限定的であること、収益を伴う開発や運用に多額の資本コストが必要であることなどから、分散 (decentralisation) にとってのリスクを有している。ASIC には潜在的なメリットもある。例えば、マイナーがアルゴリズム固有のハードウェアに投資する際に要する資本コストを踏まえると、不正を行うことによって投資資本を損なう可能性は低下するはずである。しかしながら、成熟したハッシュアルゴリズムを用いた ASIC チップの流通および製造は、依然としていくつかの大企業に集中している。これらの企業は、特定の地域への出荷を拒否し、優れたパフォーマンスを発揮する ASIC をどの地域や顧客が入手するかを決定しているため、販路を制限して価格を操作することが可能である。

ASIC マイナーによるネットワークハッシュレートの独占を防ぐために、多くの暗号通貨において、Scrypt や Ethash [25][26] といった ASIC 耐性ハッシングアルゴリズムが開

発されてきた。Monero では直近まで CryptoNight ハッシュアルゴリズムを使用しており、これは大量の L3 キャッシュを要求するために、理論的には ASIC チップの製造が困難になるはずであった。しかし、2018 年に Bitmain は CryptoNight に特化した ASIC の X3 をリリースしており、それは GPU の 10 倍の速度でマイニングすることができるものだった [27]。他のハッシングアルゴリズムも同様の運命にあり、現在は Scrypt、Ethash、Equihash が ASIC によってマイニングされている状況である。

ASIC に対抗するために、Monero は CrytpoNight ハッシュアルゴリズムを 3 ー 6 か月毎に僅かに変更するようなハードフォークを行う戦略を提案した（最初のフォークとして CryptoNightV7 に移行している [28]）。ASIC の構築には資本と時間を要するものであり、ハードウェアに極めて特化した設計であることから、ハッシュアルゴリズムの微調整によってチップ設計が無効となると、ASIC メーカーの時間と資本投下も無駄になる。しかし、このアプローチも問題を内包している。アルゴリズムの変更が ASIC の再プログラミングに対して十分な場合、ネットワークが更なるハードフォークが可能になるまでハッシュレートの集中化に対して脆弱になる点である。また、FPGAs(Field Programmable Gate Arrays) についても ASIC 耐性ストラテジーとして考慮する必要がある。ハッシングアルゴリズムに対する僅かな変更であれば、FPGAs においては簡単に再プログラムすることができるためだ。それ以外にも、定期的なコアコンセンサスの変更は不慮のバグを誘発する可能性があり、それらの対応にコア開発者チームの対応が集中してしまうといった懸念もある。

定期的なハードフォークの必要性に対処するために、Argon2 や Balloon hash といったメモリーハード・ハッシングアルゴリズム [29][30] や、ProgPoW や RandProg といった多型 (polymorphic) ハッシングアルゴリズム [31][32] など、数多くの PoW アルゴリズムが提案されている。Loki チームは、ASIC 耐性に関する長期的な解決方法を開発するために、前述のアルゴリズムについての更なる分析を発表する予定である。

長期的な解決がなされるまでの間、Loki は、CryptoNight ASIC マイナーに対する ASIC 耐性を有する CryptoNight Heavy と呼ばれるバージョンの CryptoNight を組み込む。CryptoNight Heavy は、様々な面で CryptoNight V7 とは異なる。スクラッチパッドのサイズが 4MB に増やされ、implodes と explodes の手法が変更されている [33]。これらの変更によって、ASIC マイナーの最大のターゲットである Monero の CryptoNight V7 とは区別されており、より永続的な解決法が提案されるまで、ASIC 開発に対する堅固な保護が提供されるものである。

7.2 Dynamic Block Size

他の CryptoNote コイン同様、Loki は固定ブロックサイズを持たない。代わりに、ブロックサイズは時間とともに変化し、ネットワークのトランザクションスループット上昇とともに、より多くのトランザクションを含むようになる。Loki のブロックサイズは、直近 100 ブロックにおけるブロックサイズの中央値を基にスケールされ、それに応じて新しいブロックの最大サイズも徐々に再設定される。

他の暗号通貨における長期的な懸念事項として、ブロックサイズの拡大に伴い、トランザクションを格納し検証するノードの負荷も上昇していくことが挙げられる。ブロックサイズの拡大とともに、低性能のハードウェア上で動作するノードは新しいブロックを処理して伝播することができなくなるため、ノードネットワークはノードの維持に商業的関心を有するものに集中することになる。これは、多くのノードにブロックチェーンが分散されることによって、チェーンの状態が異なる様々な主体の間で確認・検証され検閲耐性を持つといったことへの懸念に繋がり得る。

Loki では、フルノードとしてブロックの処理や伝播を行っているサービスノードにブロック報酬の一部が与えられる。そして、帯域幅とパフォーマンスが不十分なサービスノードはサービスノードネットワークから除外され（7.3 参照）、パフォーマンスの最低要件は報酬プールによって自律的に強制される。このインセンティブ構造は、多くのノードが維持されることを保証するだけでなく、どんなにブロックチェーンが拡大し、いかに帯域幅が要求されようとも、ノードがネットワークを介してブロックチェーンデータを正常に共有するのに十分な性能レベルを有していることを保証するものである。その上で、ネットワークの効率的な拡張、サービスノードの運用コスト抑制、長期に亘るノード数の維持といったことを確実にするためには、トランザクションサイズの最適化が依然として必要である。

7.3 Ring Signature Size

リング署名は、他のトランザクションに紛れさせて真の出力を隠すために用いられる。リング署名のサイズは、リングを構成するためにどれだけのミックスインが含まれているのかを示している。Monero は現在、最小リングサイズが 7 となることを強制しており、6 つのミックスインがトランザクション内の偽の出力として用いられている。

大きなリングサイズを用いることの影響についての研究はまだ少ないが、Monero Research Lab が発行する“paper0001”において、ブロックチェーン上の多くの出力を把握している攻撃者に対しての、異なるリングサイズを用いた際の効果が分析されている [34]。その分析によると、大きなリングサイズは、多くの未使用出力を把握する悪意ある攻撃者がトランザクションの効果的な解析を行うためのタイムフレームを低減させる効果があることが分かった。より大きなリングサイズを義務付けることは、第三者（取引所など）によって限定された時間内における 2 者間の取引分析を行うことができるという EABE/Knacc 攻撃 [35] として知られる理論上の攻撃を防ぐことにも寄与する。

また、Monero にはネットワークコンセンサスルールによる最大リングサイズの制限がない。Monero GUI ウォレットのような多くのウォレットにおいては、設定できるリングサイズのキャップが 26 となっている。ただし、ユーザーは 7 以上の任意のリングサイズを自由に設定してトランザクションを作成することができる。この点において、ほとんどのウォレットのデフォルトリングサイズが 7 に設定されていることが問題となる。トランザクションのリングサイズを 7 より大きくすると、それが目立つためである (Figure 3)。更に、もし個人的な Monero の取引を常に非標準的なリングサイズ（例えば 10 に設定）を用いて行っている場合、第三者による時間分析 (temporal analysis) を用いたブロックチェーン解析によって、パターン推計が可能となる。

transaction hash	ring size	tx size [kB]
3feaff3f48de0bc4c92ec027236165337b64df404aca098e212c1215e9456697	7	13.47
39d484f7c0a2e8f3823a514056d7cb0bf269171cb4582e05955d4c5ee995cad0	7	13.47
e08f5a937e725011bedd44075334ae98dcca32749da231c56da1278d49c0a231	7	13.50
ab35e69d9cca39219c90df8b2b7aab4a54c82127fb1fbaae65d76357f8f76387	7	13.50
6d8ccd56dc2d3eb7de03ba767f0dbf4d5f42ae91e67f4c28f16d6f8b0229c272	10	13.87

Figure 3: xmrchain.net（Monero のブロックエクスプローラ）は、非標準的なリングサイズがいかに目立つかを示している。

Loki ではリングサイズを 10 に固定することを強制し、これらの問題を改善する。最大リングサイズを制限することで、10 以上のミックスインでリングを構築しようとするユ

ーザーを保護するとともに、最小値としても 10 であることは、多くの出力値を有する攻撃者が、リング署名によって使用された真の出力を判別しようとすることを効果的に防ぐことができる。また、リングサイズを（Monero より）大きくすることは、リング数の増加以上に非線形的に撹拌効果を増加させるため効果的である。

現在のトランザクションスキームでは、リングサイズを 10 に増やすと、トランザクションのサイズは 2.6%増加する。しかし、Bulletproofs が実装された際には、トランザクションの総サイズが約 8 ー 13%にまで減少することとなる。これは、Bulletproofs によるトランザクションサイズの全体的な縮小によるものである。最小リングサイズの増加は、オーバーヘッドが増大することによって、大きなサイズのトランザクションをサポートするための機構が欠如しているようなネットワークにおいては問題となる可能性がある。しかしながら Loki の場合は、サービスノードが十分な帯域幅を持ってオペレーションすることがインセンティブ化されているため、このような負荷は問題とならない。

8 Attack Prevention

8.1 IP and Packet Blocking

サービスノードネットワークに単一障害点はないが、ネットワークが直面する検閲上の 2 つの重要な懸念として、ハーベスト攻撃 (harvesting attacks) と、ディープ・パケット・インスペクション (deep packet inspection) が挙げられる [36][37]。ハーベスト攻撃は、ネットワーク上の全ての動作中のサービスノードの IP アドレスを収集し、ISP レベルのファイアウォールによってそれら特定のアドレスへの接続をブロックするものである。この種の検閲は、中国の Tor ネットワークにおいて定期的に行われている [38]。ディープ・パケット・インスペクション (DPI) は、ファイアウォールを通過する個々のパケット構造を調べ、特定のサービスに関連すると思われるパケットを選択的にドロップまたはブロックすることを目的としている。DPI もまた、国家レベルの組織によって広く行われている [39]。

一方で、DPI を回避するためのシステム設計としても多くの対策がなされている。ユーザーは、各パケットの署名を通常のブロックされていないトラフィックとして現れるように転送するような接続手法を活用することができる。IP のブロックに対しては、ドメイン・フロンティング・ブリッジ (domain fronting bridges) によって、Azure や Cloudflare といったブロック対象となっていないようなサービスへの HTTPS 要求を介した暗号化されたトラフィックを用いて、一般的には回避することができる。ブリッジは、ブロック対象となっていないサービス一旦経由させた上で、リクエストを目的の場所に転送する。ドメイン・フロンティングを用いた場合、国家レベルの組織にとっても、一般的なインターネットの使用に重大な影響を与えることなく、ブリッジを介した通信の全てを妨げることは困難である。

Loki に組み込まれたガバナンスメカニズムでは、ドメイン・フロンティング・ブリッジを用いたオペレーションによって、大規模なインターネット検閲政策が行われている国においてもユーザーが Loki サービスにアクセスすることが可能である（9 参照）。加えて、OBFS4 による接続や転送サポートが、DPI に対する保護をさらに強化するために Loki ウォレットのサービスノードリリースにバンドルされる [40]。

8.2 Denial of Service Attacks

ブロックチェーンのユーザーは非中央集権的であるため、電子的・物理的な身分証明を要求されることはない。これは、身分証を有していない者や、それによって迫害されているようなユーザーにとって有益である。しかし、認証を必要としないシステムでは、悪意を持った者が多数の偽のアイデンティティ（Loki のケースでは多数の公開鍵と秘密鍵のペア）を生成し、これらのアイデンティティを使用してネットワークへのスパム要求を行うといったようなシビル攻撃に対して脆弱である。

多くの暗号通貨はこの問題の解決に苦慮しており、有料サービスモデルか PoW モデルのいずれかを実施することを余儀なくされている。Siacoin などの有料サービスモデルでは、ユーザーが使用するサービスの料金を支払っており、Siacoin の場合の費用は月間の使用ストレージ TB によって決定されている [41]。有料サービスモデルはシビル攻撃の減少に効果的ではあるが、無料の類似サービスがある場合、多くのユーザーをシステムから遠ざけてしまう（Siacoin にとって、Google Drive や Onedrive 等）。Hashcash や Nano で使用されているような PoW システムでは、メッセージやトランザクションを送信する際に、簡易な PoW 演算をユーザーに要求している [42][43]。これらの簡易 PoW システムは有料サービスモデルよりも平等であることは間違いないが、大規模な計算能力を持つ攻撃者に狙われる可能性はある。

Loki では、オフラインメッセージ及びパス作成スキームにおいて Loki システムが直面する主要な 2 つのシビル攻撃を解決するために、修正版 PoW スキームを提案する。オフラインメッセージに関しては、9 つのノードによって構成されるスウォームによって全てのメッセージが保存される必要があるため標的になり得る。悪意のあるユーザーによって大量のメッセージを格納しなければならない場合、特定のスウォームが濫用される可能性が生じてしまう。パス作成攻撃は、攻撃者が可能な限り多くのノードでパスを作成し帯域幅リソースを専有することで、正当なユーザーによるネットワークでのパス作成プロセスを締め出してしまうものである。

これら両方の攻撃を防ぐために、Loki ネットワークでは、メッセージとパスが作成される際に簡易な PoW の添付を要求する。メッセージの場合、この PoW はメッセージの Blake2b ハッシュとして計算される。パス作成においては、ノードに経路構築プロセスを要求する際に PoW が併せて送付される。モバイルユーザーのスケーラビリティとアクセシビリティを確保するため、PoW の難易度要件はメッセージまたはパスの TTL（Time-To-Live）に基づいて設定され、ネットワーク全体のアクティビティには依拠しない。

8.3 Swarm Flagging

中央集権的な管理者による包括的なルール管理がない中、トラストレスな環境下でノードを動作させる場合、ネットワーク上のノードの動作を適切に維持することは困難である。Loki のサービスノードは適切な最低要件を満たさなければならないが、トラフィックをルーティングしないものや、データをメモリプールに格納しないような設定のノードがあるかもしれない。このような設定は（帯域幅/ CPU サイクル/ストレージ使用を削減するため）経済的に有益であるが、これらのパフォーマンスの悪いノードは、分散型フラグシステムによって排除されることが提案されなければならない。

Loki においても、そのような分散型フラグを実装するにあたっていくつかの重要な論点に直面した。基本的に、全てのサービスノードは、他の全てのサービスノードに対し不正であるとしてフラグを立てるような経済的インセンティブを有する。これは、他のサービスノードにフラグが立てられステイキングプールから除外されれば、それによって

フラグを付けた側は報酬を獲得する機会が増加するためである。分散型フラグの一つの想定される手法として、フラグイベントに証拠提出を要求することが挙げられるが、この手法ではノードが自己利益のために証拠を捏造することにも繋がり得る。逆に、制限なくフラグを立てることが出来てしまうと、単一ノードまたは共謀したノードグループが、正当なノードに対して意図的にフラグを立て、自らのブロック報酬獲得の可能性を高めることができてしまう。これらの問題を回避するために、Loki はスウォーム・フラグ *swarm flagging* を提案する。

スウォーム・フラグは、既存のスウォーム（6.1.1 参照）を利用して、各テストラウンドに参加するメンバーを選択することによって機能する。各サービスノードはブロックチェーンのコピーを保持しており、マイナーによって作成される各ブロックにおいて、いくつかのテストスウォームが確定的に選択される。全てのブロックにおいて、ネットワーク内のスウォームの 1%がテスト・スウォームとして選択される。参加スウォームの算出においては、直前 5 ブロックのハッシュがメルセンヌ・ツイスタ関数のシードとして用いられ、これによって決定されたリスト内での順番に基づきスウォームが選択される。

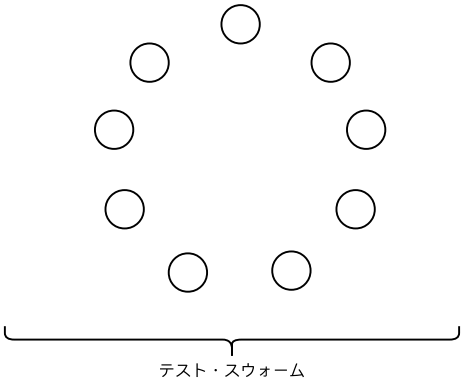

Figure 4: テスト・スウォームとして選択されたスウォームは、9 つのノードで構成されている。

参加スウォームが選択されると、そのスウォーム内の各ノードは、スウォーム内の他の全てのノードに対し多数のテストを実施することが期待される。これらはアクティブなテストではなく、各ノードは、そのスウォーム内の他のすべてのノードとの相互作用に関する履歴情報が記録される。帯域幅、メッセージストレージ、ブロックチェーン要求、及びイグジット・ノード機能に関する情報が経時的に収集され、保持される。こういった情報を収集していないスウォームへの新規参入者は、そのスウォーム外のサービスノードに照会することで、テスト対象の各サービスノードのデータを収集することができる。

各サービスノードは、他のスウォームメンバーそれぞれへの投票内容を決定する。前述のテストに基づいた決定が下されると、その投票結果は集められ、ブロードキャストされる。そして、スウォームの各ノードは全てのメンバーの投票を確認できるようになる。スウォーム内の 1 つのノードに対して 50%以上のノードが違反と見做す投票をした場合、スウォーム内のいずれのメンバーも登録抹消のトランザクションを構築するのに必要な情報を持つことができる。このトランザクションが検証され、ブロックに含まれると、全てのサービスノードが DHT を更新し、投票されたノードを除外する。

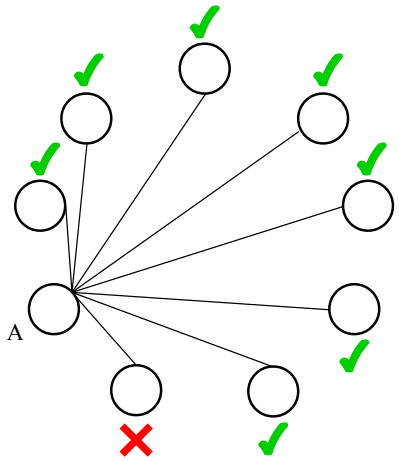

Figure 5: 不正なノードがノード A によってテストされ、不適と判定されている。ノード A は、所属内における各ノードについて、テストに適格であるか不適格であるかを識別している。

8.3.1 Testing Suite

ネットワークが自律的に性能水準を維持するためには、サービスノードは他のサービスノードをテストするために必要なツールを備えていなければならない。そしてそれらのテストは、レイジー・マスターノード攻撃 (lazy masternode attacks) [44] を防ぐため、サービスノードが提供する機能範囲を網羅している必要がある。初期設計では 4 つの基礎的なテストが企図されているが、サービスノードの機能拡張に伴い、テストセットも拡張されるものである。

オペレーターがサービスノードソフトウェアを初めて起動する際、所定のサイズの空ファイルがディスクに割り当てられ、作業を行うために必要なストレージ容量が確保される。次に、地理的に分散された Loki Foundation によって運営されるテストサーバーセットとサービスノードとの間で、簡易的な帯域幅テストが実行される。このチェックはオプションであり、これをスキップ、無視、または失敗したものについては、信頼できないサービスノードのプールに参加することができる。しかし、サービスノードを運営しようとするオペレーターにとっては、このテストの実行や合格は、彼らのノードが最小要件を満たさないために担保がロックされてしまうリスクを計れるため、有用な指標である。サービスノードが信頼できないサービスノードのプールに分類されてしまうと、それらの担保はロックされ、次のスウォームとして選択された際にテストされる。スウォームテストは総意のもと強制され、サービスノードネットワークへの新規参入者はこのテストを回避することができない。ノードがテストに合格すれば、信頼ノードフラグが与えられ、パケットのルーティングが開始される。これに失敗すると、ネットワークから除外され、担保は 30 日間ロックされることとなる。

Bandwidth Test

帯域幅テストは、Loki ネットワークテストセットの根幹を成すものである。ノードがこのテストに合格することで、最小閾値以上のパケットを正しくルーティングできるものと見做される。

ノードが別のサービスノードと交信するたびに、それによって齎される帯域幅について記録ならびに保持される。時間が経つにつれ、ノードは幾千もの経路を介し、幾百万ものメッセージを中継する。これらの相互作用によって各ノードに関する帯域幅テーブル

の基準が形成され、このテーブルに基づき、ノードはそのスウォーム内のサービスノードに関する帯域幅テストに応答することができる。

全てのノードは、他のノードが有する帯域幅テーブルの照会に応答することを求められる。これによって、ネットワークに新たに参加したばかりのノードであっても、そのスウォーム内の特定のノードに関する情報を広範なネットワークから参照することができる。

Message Storage Test

メッセージ・ストレージは、Loki Messenger ユーザーのオフラインメッセージ機能に不可欠である。サービスノードは、メッセージの有効期間（TTL ; Time-to-live）においてメッセージを保管し、ユーザーに提供する能力があるかどうかをテストされる必要がある。

オフラインメッセージを送信するユーザーは、宛先ユーザの属するスウォーム内のサービスノードをランダムに選択する。選択されたノード（配布ノード）は、スウォーム内の残りのノードにメッセージのコピーを配布しなければならない。メッセージヘッダーに PoW が添付されていれば、コピーを受け取ったサービスノードは TTL 期間中のデータを格納する。メッセージの TTL 期間が終了している場合、配布ノードはスウォーム内の他の全てのノードにナンス (nonce) を送信する。スウォームはナンスをメッセージに追加してから結果をハッシュし、最後にそれを配布ノードに送り返す。このテストによって、サービスノードが TTL 期間終了までメッセージを保持すること、及び正しくメッセージを処理できないものについては退出させられることが保証される。配布ノードのサンプリングはランダムであり、時間の経過とともに、各サービスノードはスウォームの他のノードのパフォーマンスデータを収集することができる。

Blockchain Storage Test

サービスノードは Loki ブロックチェーンの完全なコピーを保持していることが求められる。サービスノードは、ブロックチェーンの完全なコピーを保持することによって、リモートノードとしての動作、トランザクションの検証、Blink におけるトランザクションのロックなど、ネットワークのユーザーにとって不可欠な多くのタスクを実行することができる。

正当なノードがブロックチェーンのコピーを保持していることを利用し、不正なノードは、テスト時に単に正当なノードからブロックを要求するだけで、完全なコピーを保持することを避けることができてしまう。この結果を回避するため、ブロックチェーンのコピーを保持する正当なノードのみがこのテストに合格し、不正なノードが不合格となるように、ブロックチェーン・ストレージテストは設計されている。

これを達成するために、テストノードは、各被験ノードに対しブロックチェーンの履歴内から K 個のランダムトランザクションを選択して連結し、ハッシュすることを要求する。このハッシュはテストノードに返されるが、この要求へのレイテンシを測定することにより、テストノードはレイテンシと期待される復帰時間 T を比較することができる。T の値は、ディスクからのロードと、ネットワークからのブロックのダウンロードとの間における想定レイテンシを正確に区別するように厳密に設定される。いかなる攻撃者にとっても、T 時間内に K ブロックをダウンロードしてハッシュすることは実行不可能となるような設定がなされることで、結果として便乗攻撃 (piggybacking attacks) は困難になる。

Exit Node Test

イグジットノードとして機能することを選択したサービスノードは追加の報酬を受け取ることができるため、この追加報酬が悪用されないための効果的なテストが必要である。

効果的なイグジットテストが行われるには、サービスノードが人間の自然な検索動作をエミュレートできる必要がある。もしサービスノードがテストされていることを検出できる場合、サービスノードはテストにのみ応答し、ユーザーによる正当な要求を棄却するといったことができてしまう。自然なページ要求動作をエミュレートすることは困難であるが、正規の要求とテストの間のソートのオーバーヘッドを十分に難しくすることで、正規のノード運用と不正なノード運用における帯域幅コストの差が無視できる程度にイグジットテストを設計することはできる。

サービスノードは、擬似ランダム的に自然な検索語を構築するための辞書と、各地域の検索エンジンのリストを組み合わせて用いる。検索語は検索エンジンによって検索され、その結果からウェブページがランダムに選択される。サービスノードは、リレーとして動作するランダムノードによる経路と、イグジットノードとしてテストされているノードの経路をそれぞれ構築することができるため、これらの出口に対し、サービスノードは擬似ランダム探索から生成されたウェブページ結果を要求する。イグジットノードによって返された結果がサービスノードによって生成された結果と一致する場合、イグジットノードはテストに合格したものと見做される。

9 Governance, Funding, and Voting

ガバナンスは暗号通貨の設計における不可欠な要素であり、プロトコルレベルでサポートされる必要がある。不明瞭に定められている脆弱なガバナンスのリスクについては、ブロックチェーン技術の歴史を通して広く研究されてきた。Bitcoin と Ethereum は、それぞれのコミュニティの中心とその活動を分割するような激しいハードフォークを経験している。ハードフォークはガバナンス戦略として使用することはできるが、論争の度の解決手段としてではなく、常に最終手段として検討される必要がある。Loki のガバナンスシステムは、対話と意思表明のための体系化された環境を設け、Loki 発展のための資金源を外部勢力や利他的行為に依存せずに拠出することで、潜在的な問題を解決できるように設計されている。

ハードフォークを防止することだけではなく、ガバナンス構造によって、Loki のエコシステムを向上させる新規プロジェクトに資金を提供するような内部的な手段も構築されるべきである。内部資金プロジェクトは、ユーザー、マイナー、サービスノードと必ずしも同一の動機を持っていないような特別な利害グループの形成を防止することができる。ビットコインやビットコインの様々なフォークでは、Blockstream や Bitcoin ABC、Bitcoin Unlimited といった非営利団体が、独自のビジネス上の目的推進や独自のイデオロギーに従うようなプロトコル固有の変更をビットコインやビットコインキャッシュに行うために開発者を雇ったのではないかとしばしば批難されている。

このような点を踏まえ、Loki のブロック報酬の 5% はネットワークガバナンスに割り当てられている。これにより、コミュニティプロジェクト、ソフトウェア開発者、および統合チームに Loki の安定的な供給が行われる。この 5% のブロック報酬のうち、3.75% は Loki Foundation によって管理され、1.25% は Loki Funding System を通じてサービスノードによって管理される。サービスノードによる公平な意思表示を用いることで、Loki Foundation の管理外における資金提供の提案を可能とするものである。

9.1 The Loki Foundation

Loki Foundation は、オーストラリアに拠点を置く非営利団体として登録されている。このような法人を設置することで、Loki プロジェクトが法的な枠組みの中で活動して

いることを明確にし、プロジェクトに従事している者に法的な保護と義務を課している。Loki Foundation は 2018 年にオーストラリアに設立され、Australian Charities and Not-for-profits Commission (ACNC) の枠組みに準拠している [45]。この枠組みにより、Loki Foundation は、株主または受益者がいなく理事会メンバーが期限付きの議席を持つような他の多くの非営利組織と同様のコーポレート・ガバナンス体制を持ち、メンバーの提唱は決議に基づき実行することが規定される。また、Loki Foundation は、オーストラリアで慈善団体の地位を取得するように構成されている。

この組織は、プロジェクトの進展や関連した施策によるいかなる収入（ガバナンス・ブロック報酬を含む）についても法的な拘束を有する。Loki Foundation の慈善団体としての登録を維持するため、また、不正なく適切な範囲で支出がなされていることを公に保証するために、外部監査を受けた上での透明性は不可欠である。Loki Foundation は、コミュニティと監査の両方に責任を負うものであるが、このシステムが最終的に Loki やその関連プロジェクトに資するものでなくなった際の厳然たる保護手段も存在する。十分なネットワークコンセンサスを伴うハードフォークをもって、ガバナンス・ブロック報酬の受領者としての Loki Foundation を削除または置き換えることが可能である。

9.2 The Loki Funding System

Loki Foundation は、Loki プロジェクトを代表する多様な個々人によって構成されており、Loki Foundation としての自治体制とオーストラリアの法律の両方に従うものである。これは、Loki Foundation の行動決定範囲を制限する要因になり得るが、Loki Funding System によって、ブロック報酬の一部をサービスノードからの投票のみによって処理することを可能にしている。サービスノードは世界中の参加者を代表しており、Loki プロジェクトチームや Foundation からの指示を考慮する必要がないため、新たなレベルでの自律性を持って決定を下すことを可能としている。また、サービスノード群はネットワーク内で最も出資している参加集団であることから、Loki の価値を向上させるような決定を下す経済的インセンティブを有している。

9.2.1 Proposals

サービスノードによってなされる全てのプロポーザルは Loki ブロックチェーンに公開される。特定のパーティがサービスノードにプロポーザルを提示したい場合、パーティはプロポーザルトランザクションを構築する必要がある。プロポーザルトランザクションの内容は可読でなければならず、アウトプットは書き込まれなければならないため、通常の Loki トランザクションのプライバシー機能は適用されない。

資金調達ブロック (funding blocks) は 43,000 ブロック (約 60 日) ごとに作成される。プロポーザルの代表者は、この期間にいつでもプロポーザルを提出することができる。しかし、それぞれのプロポーザル・フェーズの始めに近いほど、各サービスノードからの投票を得る時間は増えることとなる。

各トランザクションには、プロポーザルの投票において各サービスノードが認識すべき情報が含まれた追加のフィールドが添付されている。この情報には以下の内容が含まれる；プロポーザルのタイトル、プロポーザルの詳細説明についての URL へのリンク、プロポーザルが求めている Loki の金額、支払い先アドレス、エスクローエージェント(任意選択)。

Loki Foundation からの合意が保留中の場合、提案者は、Loki Foundation または他の第三者をエスクローエージェントとして選択し、マイルストーンに達すると資金が拠出さ

れるようにすることもできる。加えて、高水準のプロポーザルを奨励し、スパムトランザクションを防止するため、各プロポーザルのトランザクションにおいては少なからぬ量の Loki を消費することを義務付ける。

9.2.2 Voting

各サービスノードは、投票のための特定のキーを用いる。このキーは、ホストされているサーバーにログインせずとも、エクスポートすることで、サービスノードに代わって投票を行うことができる。

投票はチェーン上で行われるわけではなく、各サービスノードは、ブロックチェーン上の各アクティブプロポーザルに対して、支持・不支持・白票を示す。各サービスノードは、次回の隔月の資金調達ブロックまでの間に、ブロックチェーンにコミットして直ちにプロポーザルに投票することができる。そして、次の資金調達ブロックが作成される直前に、それまでに投じられた全ての投票を集計するためにスウォームが選出される。この集計はノード・メモリプールに格納され、マイナーが資金調達ブロックに達するまでそこに保管される。そしてこの情報は、承認されたプロポーザルに報酬を配賦するブロックを作成するために使用される。プロポーザルは、支持投票から不支持投票を引いた結果がサービスノードネットワークの総ノード数の 15%以上であることをもって承認される。

9.2.3 Funds Distribution

Loki Funding System からの収入は全て、資金調達ブロックを通じて支払われる。資金調達ブロック報酬は、従来のブロック報酬と同様に、完全に非保管的 (non-custodial) に Loki を　配賦する。資金調達ブロックは 43,000 ブロック（約 60 日）ごとにマイナーによって構築される。資金調達ブロック全体を通して、総ブロック報酬の 1.25%が割り当てられることとなる。

有効な資金調達ブロックを構築するためには、要求される支持率にプロポーザルが達しているかをマイナーが評価できなければならない。これには、サービスノードがブロックチェーンにコミットする情報（支払先アドレスや全ての投票結果が含まれる）が用いられる。全てのサービスノードはマイナーの資金調達ブロックを検証し、無効なアドレスへ支払う資金調達ブロックについては破棄される。

多くの場合、承認されたプロポーザルで必要とされる Loki の合計額は、その 60 日間に積み上げられた合計額を超過もしくは下回る。承認されたプロポーザルの総額が資金調達ブロックで利用可能な額を超える場合、マイナーは先にブロックチェーンにコミットされたプロポーザルを優先させて資金調達ブロックを構築する。残りの承認済みのプロポーザルは、次回の資金調達ブロックまでブロックチェーンにコミットされたままとなる。

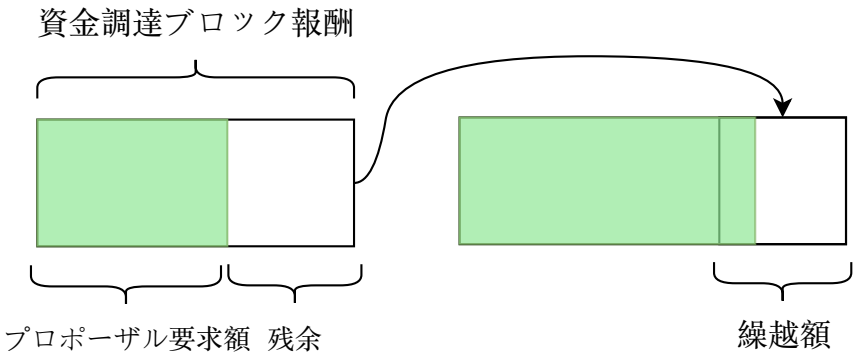

Figure 6: 未使用となった残額分、次の資金調達ブロックの報酬が増加する。

10 Conclusion

Loki は、経済的インセンティブを有するノードのネットワーク上に匿名取引と分散型コミュニケーションを構築するモデルを提案する。Loki は CryptoNote プロトコルの基盤を用いてプライバシーを確保しており、ネットワーク耐性と機能性を強化するために担保要件のあるノードシステムを実装している。

加えて、Loki は過去の研究やオープンソースプロジェクトを発展させることによって、既存のプロトコルに対し著しい利点を有する新規の匿名ルーティングプロトコルを提示する。このユニークなアーキテクチャーとプロトコル設計の組み合わせによって、市場のシビル攻撃耐性を持ったネットワークを構築し、時間分析の効率を低下させ、ユーザーに高度なデジタルプライバシーを提供する。

References

[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.