オーキッド:分散型ネットワーク ルーティング市場

ジェイク・S・カネル[1, 2]、ジャスティン・シーク[1, 2]、ジェイ・フリーマン[2]、グレッグ・ヘイゼル[2]、ジェニファー・ロドリゲス・ミュラー[2]、エリック・ホウ、ブライアン・J・フォックス、スティーブン・ウォーターハウス博士。

バージョン 2.0
2019 年 11 月 18 日

1:主な著者。2:技術設計を担当する協力者。
謝辞のセクションで説明した追加の貢献。

要約

オーキッドを紹介します。オーキッドは、匿名通信と仮想プライベート ネットワーキングの分散型市場です。既存のプライバシー ソリューションは、付随する集中化リスクを伴う不透明な商用サービスか、サービスの質に見合うインセンティブや大規模な経済的セキュリティを欠いた無料のピアツーピアネットワークのいずれかです。オーキッドは、ノード プロバイダーがトークンをステーキングしてサービスを宣伝する、イーサリアム ブロックチェーンを使用した帯域幅市場です。クライアントは、ステークに対してランダムに加重され、二次的な基準 (価格、場所など) でフィルタリングされたノードを選択することにより、シングルホップまたはマルチホップのオニオン ルーテッド回路を構築できます。ステーキングは、オペレーターの不正行為に対するインセンティブを調整し、特に線形のステークウェイティングは、シビル攻撃を中和します。オーキッドは、1 秒あたり数百万件のトランザクションに対応する確率的支払いシステムを使用しており、信頼できる中央パーティなしで流動性の高い帯域幅市場を実現させます。パケット規模の支払いにより、トランザクター間の暗黙的なフロート バランスを極小レベルまで低減することにより、信頼不要の高頻度の相互作用が可能になります。

1. イントロダクション

かつては自由で開かれたフロンティアであったインターネットは、現在ますます破壊され、監視され、検閲されるようになっています。政府や企業が接続の監視、検査、ブロックをこれまで以上に効果的に実施できるようになるにつれ、VPN (仮想プライベート ネットワーク) などのプライバシー保護や匿名ツールの需要が大きく高まりました。VPN はほとんどのユースケースで十分に機能しますが、中央集中型の信頼ベース モデル固有の弱点を抱えています。ユーザーは、政府の強制または追加収益の誘因によって VPN プロバイダーが密かにデータを記録および共有していないという確証を得ることはできません。VPN の定期的な支払いや価格設定モデルはロックイン効果を生み出すため、プロバイダーがブロックされたり遅い場合でも、ユーザーはプロバイダーを安価かつ迅速に切り替えることができません。Tor[1] や I2P[2] などの現在のピアツーピア システムは、マルチホップ回線を構築して、単一のパーティからルート情報を隠します。ただし、これらのシステムは無料であるため、パフォーマンスとセキュリティの両面で問題があります。インセンティブが不十分で、寄付された無料の帯域幅が非常に限られているため、パフォーマンスと品質が低下し、セキュリティ面でも、攻撃者が総ネットワーク帯域幅のかなりの部分を提供しようとする際の引き継ぎコストが低いという欠点があります。

必要なのは、適切な経済的インセンティブとナノペイメントの仕組みを備えたピアツーピア プライバシー ネットワークです。それによってクライアントは多くの異なるプロバイダーのノードの統一されたグローバル プールから、シングルホップまたはマルチホップのルートを構築できるようになります。オープン マーケット システムは、利益を追求する販売者が提供する帯域幅の供給が、ユーザーの需要増に合わせて弾力的に拡張できることを保証します。暗号通貨の契約メカニズムを使用すると、悪意のある動作に対抗するために必要なインセンティブを提供できます。

設計を推進する上で、いくつかの中核的な課題があります。それは、トラフィック分析、シビル攻撃、およびランダム選択の問題です。オーキッド自体を詳細に説明する前に、これらの問題のそれぞれについて簡単に説明します。

トラフィック分析

理論上および実践上、受信者以外の関係者に情報を漏らさずにメッセージを送信することは非常に困難です。*ミキシング ネットワーク*において、Chaum [3]が最初に提案した方法では、メッセージは多数のプロキシ ノードを経由してルーティングされ、各ステップでランダムに並べ替えられ、エンベロープがエンベロープに含まれるレイヤーのような形で暗号化されます。その後に開発され、Tor[1] に採用された*オニオン ルーティング*では、拡張性を高めるために、永続的接続のそれぞれに、単一の共有回路ではなく、固有のランダム プロキシ ノード パス (回路) と組み合わせた、同様の階層化された暗号化の概念を使用しています。*トラフィック分析*は依然として潜在的な問題 [4] ですが、帯域幅のバーニング (パディング) やランダムなメッセージ遅延により、実行コストを非常に大きくすることによって克服できます。*共謀*はもう一つの深刻な問題です。少なくとも回路内の他のすべてのノードが協調している場合、完全な回路を推測できるからです。

シビル攻撃

オープン ネットワークでは、多くの偽の ID を作成できるため、実際にはすべてが共謀しているノードを多数の独立したノードとして提示することが可能です。開放性という特性を保持しながら、単一の攻撃者がシステムを圧倒することを防ぐのは困難な場合があります。この問題の解決策の 1 つは*プルーフオブワーク*です。この方法は、HashCash [4] に由来し、後にビットコイン [5] に採用され、初期のオーキッド 0.9.2 [6] でも、シビル攻撃への防御策として提案されました。プルーフオブワークでは、各ノードが計算リソースを消費して、アイデンティティを証明する必要があります。したがって、多くの偽の ID を作成するには、それに比例して多額の費用が必要になります。*プルーフオブバーン*は、効果は似ていますが、暗号通貨の破棄の証明のみを必要とします。この方法は、破棄された通貨の価値が完全に無駄になるのではなく、その通貨のステークホルダーに再分配されるという利点があります。*プルーフオブステーク* ベースの暗号通貨では、ユーザーは、ブロック報酬を受け取ってネットワークに参加するために通貨をステーキングする必要があります。私たちは、シビル攻撃を打ち負かし、インセンティブを調整し、経済的セキュリティの重要な利点を提供するために、*ステークウェイティング* システムを採用しています。

ランダム選択

共謀の可能性が低い安全な回路を構築するには、シビル攻撃の影響を受けない方法でリレー ノードからランダムに選択する必要があります。私たちはこれを、線形ステーク加重されたランダム選択で実現しています。それが *シビル直交 (Sybil-Orthgonal)* です。この場合、攻撃者は自分のステークを複数の ID に分割しても、全く利益を得ることができません。また、この選択スキームは、簡単で効果的な負荷分散の手段となるため、最小の 1 ホップ回線の場合であっても、わずかですが追加の利点があります (結託の関連性が低い場合)。グローバル ランダム選択ポリシーを実装するには、クライアントがノード メタ データのグローバル リストを利用できる必要があります。以前のオーキッド 0.9.2 [6] は、この目的のためにカスタム コード [7] をベースとする DHT (分散型ハッシュ テーブル) を提案しました。簡単にするために、現在では私たちは、イーサリアム ブロックチェーン [8] (およびその基礎となる DHT) を直接使用して、グローバル ノード レジストリを提供しています。

概要

オーキッドは、クライアントがさまざまな潜在的用途を備えた高性能のオニオン ルーティング回路を構成できる分散型プラットフォームであり、そのような回路に資金を供給する新しい確率論的ナノペイメントシステムを搭載しています。オーキッド サーバー ソフトウェアを実行している帯域幅プロバイダーは、オーキッド トークン (「OXT」、ERC20 [1]イーサリアム ディレクトリ スマート コントラクト内の互換性のある暗号通貨) を使用して、ステーキングしたデポジットのサイズに応じてトラフィックと収益を受け取ります。一方でクライアントは、ステーク加重されたランダム選択を通じてノードを見つけます。これは、ツリー データ構造を使用して、スマート コントラクト機能として実装したものです。クライアントは、毎秒 1 回の頻度で送信される確率的ナノペイメントを使用してノードに支払いを行います。マルチホップ回線は、ホップごとのアカウントまたは間接的なオニオンの支払い転送を使用して、支払い自体からの情報漏洩を減らすことができます。回路は、技術的または経済的な理由 (つまり、クライアントのトラフィックの回路固有のコストが現在の予算を超える場合) で失敗する可能性があり、失敗時には単純に再サンプリングされます。私たちの設計の中核となるメカニズムは驚くほど単純ですが、当然のことながら、やっかいな問題は細部にあります。

2. 背景

オンラインへの情報の移行が進み、日々、より多くの脆弱性が明らかになる中、ネットワーキングにおけるプライバシーは長い間懸念されてきました。

私たちが使用している基本的なコンピューター ネットワーキング プロトコル [9] や実務の多くは[2] 1961 年から 1989 年の間[3] に高い信頼のある学問や趣味の文化で培われたもので、現在の携帯電話、ノートパソコン、デスクトップ パソコンでも引き続き使用されています。それらはすべて根本的に堅牢なものではなく、経済を考慮していませんでした。デフォルトの操作は、機械でタイピングされたハガキでいっぱいのメールシステムのようなもので、検証がなく、配信中に検出できない変更や交換を行うことが可能です[4]。

---

[1] https://theethereum.wiki/w/index.php/ERC20_Token_Standard
[2] https://www.people-press.org/2015/11/23/1-trust-in-government-1958-2015/
[3] http://www.catb.org/~esr/faqs/things-every-hacker-once-knew/#_key_dates
[4] https://en.wikipedia.org/wiki/Packet_injection

インターネット サービス プロバイダー (ISP) は電力会社であることが多いのですが、電力会社には、権威主義的な政権 [10] と協力し[5] (またはそのような政権によって運営され)、サービスを操作してユーザーの利益を損ないながら自分たちの利益 [11] を改善するという悪い評判があります。ISP はデータ送信サービスの価値を完全には破壊しないのが普通ですが (一部の例外を除きます[6])、一方で学者たちは、クライアント/サーバー間の自発的な双方向関係を ISP が破壊できる程度を最小化したプロトコルを設計することが自分たちの本来の仕事だとは全く考えていませんでした (そのような破壊は、プライベート データ伝送パイプを独占することで初めて可能になります)。

非権威国家においてさえ、ケーブル会社、電話会社、または専門会社が産業スパイを合法化するよう代表政府に働きかけ始めました[12]が、これは、すべてのパケット転送に関する元の規範に明らかに違反します [13]。Facebook の人気は 2014[14] 年以降劇的に低下しました (2019 年には、「良く知られた組織」の上位 100 社の 94 位にランクされました。これは、トランプ組織と米国政府自体よりわずかに上です)[7]。ただし、ユーザーは Facebook へのアクセスをやめることができ、そうし始めています [15]。対照的に、ISP は粘度の低い市場にサービスを提供しており、6,000 万人のアメリカ人が文字通りブロードバンドの独占に直面しています [16]。

一般的なプロトコルを強化する試みが行われましたが、完全に安全なファイアアンドフォーゲット プロトコルはほとんどありません (ほぼゼロです)。たとえば、SSH は比較的安全で広く使用されています [17] が、2003 年にトラフィック分析攻撃が確認され [18]、2019 年現在では、実際のコードでの問題にパッチが適用されるのはかなり偶然でしかありません[8]。

信頼できない ISP ルーターを介して送信される強化されていないプロトコルは、ほとんどのユーザーにとって差し迫った問題ではありませんが、多くの人々はコーヒーショップ、空港、またはホテルの WiFi を介してインターネットにアクセスします。これらの状況では ISP のようなインセンティブが若干再現されるため、スパイ、サービスの低下、価格の調整は珍しくありません。無料の WiFi 実装を試行する場合、技術的な予算の削減により、ユーザーが誤って互いをスパイするバグのある構成が発生する可能性もあります。世間の認識では、これらの課題はすべて、インターネット全般、特に Wi-Fi 経由でアクセスされるインターネットは、混乱を招く、潜在的に危険な方法でスパイされる可能性が高いと、漠然ととらえられています。

企業の世界では、仮想プライベート ネットワーク (VPN) テクノロジーは、従業員 (特に出張や在宅勤務の従業員) がより広い (デフォルトの信頼されていない) ネットワーク環境から暗号化された安全なトンネルを通って安全な作業場であるイントラネットに戻れるようにする方法として、当初、大量に採用されました。この設定は「VPN」と呼ばれました。これは、トンネリング ソフトウェアによって、「仮想的に」安全な「プライベート ネットワーク」の「内部」にいられるようにするものだからです。これによってプロトコル強化の問題が完全に解決されたわけではありません (トラフィックの形状とタイミングは保護されない傾向があります) が、このようなトンネルを介して強化プロトコルと非強化プロトコルを混合して送信すると、少なくともインジェクション攻撃や、ある種の推論攻撃からは保護することができます。

---

[5] https://en.wikipedia.org/wiki/BGP_hijacking#Public_incidents
[6] https://www.nicholasoverstreet.com/2010/03/new-wave-communications-the-worst-isp-in-america/
[7] https://theharrispoll.com/axios-harrispoll-100/
[8] https://zinglau.com/projects/ObfuscatedOpenSSHPatches.html

企業環境での VPN サービスの台頭により、基本的に同じテクノロジーを (同様のトンネリング コンセプトを使用して) 再利用し、消費者市場に提供することが可能になりました。この新しいエコシステムには、信頼できるローカル機関の役割を果たす雇用者がいないため、技術者、起業家、研究者などが、より信頼できる安全なネットワークを求めて、さまざまなソリューションを模索することになりました。コンシューマ VPN は、この可能なソリューションの範囲に沿ってニッチ分野を占有し、Tor は別のソリューションを占有しています。また、Tor を改善しようとする試みは、通常、インセンティブと支払い (またはその欠如) によってもたらされる課題に基づいています。

2.1 コンシューマー VPN

コンシューマー VPN 会社は ISP のユーザー向けの仕事に割り込んでいます。ISP には以前、次の 2 つの仕事がありました。(1) ワイヤをインストールし、(2) それらのワイヤの未強化データをスパイしないことです。しかし、現在の ISP には最初の仕事しかありません (ISP はユーザーの家に入るワイヤの独占権を持っているため、最初の仕事は保持されます)。現在、2 番目の仕事の一部は VPN のトンネリング ソフトウェア (データの暗号化) によって、一部は VPN 会社によって行われています。2 番目の仕事とは、強化を弱めたストリームにデータを復号化し、さまざまなサブストリームをより広いインターネットのさまざまな部分に転送することです。

これらのサービスは、コーヒーショップ、ホテル、空港など、信頼できない WiFi シナリオの危険の多くからユーザーのトラフィックを保護します。また、ISP の顧客がウェブサイトから IP アドレスを隠すことを望む場合、または自身の ISP からトラフィックを隠すことを望む場合など、さまざまなユースケースで人気を集めています。

VPN は、アクティブに活動すると、多くのプライバシー モデルと信頼モデルの観点から、事実上ユーザーの新しい ISP になります。ただしこれは、ISP が以前に実行できたすべての攻撃を、VPN プロバイダーも簡単に実行できることを意味します。他の集中型システムと同様、VPN は、VPN を支配している企業体と同程度に安全で信頼できるにすぎません。さらに、既存の支払いシステムとビジネス モデルでは、毎月またはそれ以上のサービス契約が必要となり、短期の使用には高いプレミアム価格が課されるため、ユーザーのロックインにつながります。

2.2 Tor、オニオン ルーター

プライベート インターネット接続を求めるユーザーには、(ほとんどが無料の) 分散型システムという形の代替手段があります。最も広く使用されているこのようなシステムは、Tor ネットワークです [1]。Tor の背後にあるコア コンセプトは、最終的な宛先に到達する前に、ランダムに選択された複数の統計的に無相関の中間ルーターを介してパケットを送信することにより、トラフィックを難読化することです。

残念ながら、Tor などの分散型システムには独自の問題が多数あります。主要な問題の 1 つは、可用性と帯域幅を増やしながら待ち時間を短縮するなど、ネットワークに適切な動作をさせるためのインセンティブです。これらの問題は、経済的インセンティブ メカニズムを通じて克服できます。

分散型システムのインセンティブ化は、単純な経済モデルをシステムに適用して適切な行動を促進することから始まりました。初期のアルゴリズムでは、分散リソースの割り当てに、帯域幅や待ち時間などのネットワーク プリミティブで報酬と罰をモデル化した、tit-for-tat[19] のような物々交換が頻繁に使用されていました。このアプローチは一般に安定した分散型システムを実現させましたが、多くの場合、フリーライダーの問題など、一見扱いにくい問題に依然として悩まされています [20]。分散型システムの開発が開始されると、ピアツーピア インセンティブに対する明示的な経済的報酬と罰のアプローチが現れ始めました。これらの方法は、インセンティブの経済的有用性の明示的な尺度を作り出すことで、良い行動を推進し、悪い行動を抑止するよう精巧に調整されたアプローチを可能にしました。

2.3 インセンティブ化された Tor

インセンティブ化されたピアツーピア プライバシー ネットワークの最初の例の 1 つは、インセンティブ化された Tor に登場しました [21]。Ngan らによるこの最初のプロポーザルでは、インセンティブ メカニズムとしてルーティング リソースを割り当てる *tit-for-tat* 戦略が提案されました。tit-for-that 戦略の中核は、あなたに向けて分配するのと同じ方法で、ピアに向けてもリソースを分配することです。ピアが非協力的に行動するなら、あなたも非協力的に行動し、ピアが協力的に行動するなら、あなたも協力的に行動します。このように、決定を反復するペイオフ マトリックスは、常にナッシュ均衡をもたらします。

最近では、Androulakil その他の人々[22] が、パケット転送をより直接的に促進するために、実際の支払いがどのように行われるかを実証しました。俯瞰的には、匿名の支払いスキーム (ルートの最初のノードの支払いに使用) と、回路の残りの部分の連鎖されたマイクロペイメントとのハイブリッドを中心に設計を考案します。この設計は、暗にパケット転送のための市場を意味しています。理想的には、Tor ユーザーは、最高のプライバシー、帯域幅、スループット、待ち時間を提供するピアを選択する傾向があり、サービスと引き換えに、デジタル通貨を支払います。現在では、パケットを送信するユーティリティは、定量化が困難な tit-for-that モデルのペイオフ マトリックスに対して保持するのではなく、金銭的インセンティブと直接一致させることができます。

経済的インセンティブの核となる考え方は、ピアツーピア システムで望ましい行動を推進するのに非常に強力ですが、いくつかの固有の問題があります。おそらく最大の問題は、トークンの鋳造に関して中央銀行に依存していることです。このペーパーで後述するように、この問題は支払いに分散型暗号通貨を使用することで解決できます。

上記のモデルの代替アプローチは、Ghosh その他によって提示されたプルーフオブバンドウィドゥス (帯域幅の証明) モデルによるインセンティブ化です[23]。このモデルでは、回路内の各ピアは、十分な帯域幅が送信された後にクライアントによって開始される新しいミンティング (鋳造) の証拠の生成を支援します。この情報はチェーンでブロードキャストされ、回路内のすべてのメンバーに、パケット転送に関する支払いが効果的に行われます。このプロトコルは理論的には有効に見えますが、ノードへの支払いがインフレーションに依存しており、市場主導の価格設定がないため、源泉徴収攻撃やその他の悪意ある行為にさらされる懸念があります。

最終的に、より多くの潜在的な攻撃にさらされることのない効率的なインセンティブ メカニズムを Tor に導入することは難しいようです。

2.4 支払いチャネルに基づくルーティング

支払いチャネルは、情報とお金の両方のルーティングに使用できます。この顕著な例は、Thomas と Schwartz [24] によって導入されたインターレジャー プロトコル (ILP) です。ILP のアトミック スワップ メソッドの背後にある中核的アイデアは、ハッシュ タイム ロック コントラクト (HTLC) を使用して、データ パケットの転送時にトークンを支払う暗号検証可能なマイクロペイメント チャネルをセットアップすることです。従来の支払いチャネルとは異なり、これらのマイクロペイメント チャネルはあまり頻繁にチェーンで決済されないため、取引手数料の償却と低遅延の両方が可能になります。ただし、このプロセスでは、ルートはネットワークから完全には隠されません。

Khosla [25] は、Tor のような機能を使用できるようにする、このような暗号検証可能なマイクロペイメントを伴った ILP 上に、オニオンルーティング ベースのプラグインを導入します。彼らのシステムは、マルチホップ データ回路のすべてのリンクに ILP 支払い回線を使用することで、レイテンシ、エラー確率、および複雑さの度合いを大幅に引き上げます。

支払いチャネルに裏付けされたルーティング方法は分散型支払いのための有望なレイヤー 2 スケーリング ソリューションとして大きな注目を集めていますが、再帰ルーティングを実行する必要があるため、その展開と効率に苦戦しています。エンド ユーザーは、1 つ以上の特定の支払いルーターに資金を預け入れ、信頼を要求し、カウンターパーティ リスクを導入する必要があります。支払いのルーティングには、O(log N) のステップと待ち時間が必要です。支払いは、そのサイズとキー エッジに沿って利用可能なデポジットに左右されるため、常にルーティング可能とは限りません。重要なエッジが配信されない場合、支払いルートが完全に失敗し、長い遅延が発生する可能性があります。これらの理由から、支払いチャネル ネットワークは一般的に広く採用されているマイクロペイメント ソリューションではなく、特にオニオン ルーティングには採用されていません。

3. 目標と制限

オーキッドの使命は、人々が、検閲、監視、仲介を恐れることなく、自分のコンピューターのネットワーク アクティビティについて理解し、それを管理できるようにすることです。その使命を果たすために私たちは、オープン ソース ソフトウェアを使用した幅広い対象者向けのソリューションを開発しており、イーサリアム ブロックチェーン上の確率論的ナノペイメントを活用した分散型 VPN 市場を構築しています。私たちの設計では、スケーラビリティ、分散化、使いやすさ、シンプルさ、拡張性を重視しています。オーキッドは、支払いの匿名性、スケーラビリティ、検閲抵抗という点で、イーサリアムから現在の制限を継承しています。さらに、当初は手頃な価格の高帯域幅と低レイテンシのルーティングを重視していたため、現在、最も洗練された理論的なトラフィック分析攻撃に対するオーキッドの防御能力は限定的です。ただしこのような制約は、想定される大衆消費者ユースケースのほとんどにとって障害ではありません (セクション 6)。

3.1 目標

スケーラビリティ

オーキッドのナノペイメントシ ステムは、現在のイーサリアム ブロックチェーン (セクション 5.9) で 1 秒に 1 回確率的なトランザクションを送信する数百万人のユーザーに拡張されます。また、イーサリアム 2.0 でのシャーディングを使用すれば、1 秒あたり数十億回のトランザクションに拡張できます。ノード選択プロセス (セクション 4.3) を使用すると、クライアントは信頼不要の方法でノード選択をサーバーノードにアウトソースすることができ、オーキッド クライアントの軽量な実装が可能になります。

分散化

ナノペイメントからノード ディレクトリおよびディスカバリーまで、設計のすべてのコンポーネントは分散化されています。イーサリアム ブロックチェーンは、機能する市場に必要な、最小限の契約上の決済を実施するために使用されます。OXT のステーキング比率が十分に分散していると仮定すると、オーキッドには特大の影響力や制御権を持つ、信頼できる特別な当事者は存在しません。

使いやすさ

使いやすさは広く採用されるための鍵であり、システムがユーザーごとに提供する匿名性は、ユーザー基盤の拡大とともに増加します。デフォルトのクライアント実装は、構成またはルート管理にユーザーの不要な決定を要求することなく、単に「機能する」だけです (ただし、詳細な構成オプションが必要な場合は利用できます)。クライアントは、予算編成やノード選択などの退屈な詳細の自動化も支援します。ほとんどのユーザーにとって、オーキッドを使用してネットワーク接続を保護することは、ボタンを押すのと同じくらい簡単です。

シンプルさ

このプロトコルはシンプルなもので、理解、実装、セキュリティ分析が容易です。私たちは、複雑なオークション メカニズムの代わりに、販売者が決定した帯域幅価格とクライアント価格フィルターを使用します。確率論的支払いプロトコルも比較的単純で、スマート コントラクトは、約 200 行の Solidity コードで構成されています。

拡張性

コア メカニズムは分離可能であり、将来容易に拡張や交換ができる範囲で直交しています。ナノペイメント プロトコルとスマート コントラクトは、他のシステムとは直接対話しません。同様に、ノード ディレクトリは、ノード メタデータ レジストリおよびその他のコンポーネントから隔離および分離されます。引き出し遅延などの主要なシステム設計のハイパーパラメーターは、適応を容易にするために、可能な限り契約上のパラメーターとしました。WebRTC ベースのトランスポート プロトコルも同様に直交しており、拡張可能です。このナノペイメント システムは、オーキッド帯域幅市場向けに構築されていますが、汎用的であり、幅広い用途に使用できる可能性があります。

3.2 制限

オーキッドは、スマート コントラクト機能、分散化、コミュニティの規模、エンゲージメントの面で世界をリードするブロックチェーンであるイーサリアムに基づいて構築されています。そのため、イーサリアムに固有のスケーリングとセキュリティのあらゆる問題を共有しますが、可能性のある危機に対処するために拡張イーサリアム コミュニティの力を借りることもできます。

ネットワーク依存

オーキッドの経済的セキュリティ (セクション 4.4) は、イーサリアム自体の経済的セキュリティによって上限を課せられています。イーサリアム ネットワークを不安定化または削除する機能を持つ攻撃者は、当然、オーキッドを削除することもできます。(さらに、イーサリアムに対するシャットダウン攻撃が成功すると、たとえ意図的ではなかったとしても、事実上オーキッドもシャットダウンされます)。強力な攻撃者は、たとえば、持続的な 51% 攻撃を開始することでこれを達成できます。攻撃はおそらく、DDOS や主要なイーサリアム ノードに対する他の攻撃によって増幅されるでしょう。

オーキッド サーバー ノードは、個々のレベルでイーサリアム ネットワークにも依存しています。これは、当選したナノペイメントの償還を処理するためにイーサリアム ノードへの信頼できる接続が必要だからです。したがって、オーキッドのノードは、イーサリアムへのエクリプス攻撃に対して個々に脆弱です。実際には、Alchemy や Infura などの商用イーサリアム ノード オペレーターは、これらのリスクを軽減するのに役立ちます。

ユーザーのスケーラビリティ

現在のオーキッドのナノペイメント システムには、効率/分散のトレードオフがあります。額面金額が大きなチケットは、分散を犠牲にしてチェーン上の支払いと取引手数料の頻度を減らします。分散に関するユーザーの許容範囲は限られていると予想されるため、これらの制約とイーサリアムの現在の最大トランザクション スループット (1 秒あたり約 12 トランザクション) を考えると、オーキッド ユーザーは数百万人に限定されます (セクション 5.9)。イーサリアム 2.0 のシャーディングを使用すれば、このユーザー制限を超えるスケーリングが可能です[9]。

支払いの匿名性

まれに発生する当選のナノペイメント チケットは、チェーン上のイーサリアム トランザクションを通じて引き換えられます。したがって、オーキッドのナノペイメントは疑似匿名に過ぎず、時折一部の情報が漏洩します (セクション 5.8)。より強力な匿名性を望むユーザーは、ナノペイメント アカウントにロードする前に OXT 通貨を匿名化する必要があります。

公開ノード ディレクトリ

オーキッドのノード ディレクトリは、イーサリアム ブロックチェーンで公開されているため、世界中に公開されています。したがって、検閲を行う攻撃者は簡単に、オーキッド ノードのリストされたすべての連絡先 IP アドレスを自動的にブロックできます。その意味するところと、オフチェーンで共有されているプライベート IP アドレスの使用など、考えられる回避策については、セクション 6.4 で説明します。

[9] https://github.com/ethereum/eth2.0-specs

トラフィック分析

私たちが最初に重視したのは、強力な匿名性を犠牲にして、高帯域幅、低遅延回路を実現することです。このトレードオフが基本です [26] が、私たちの設計では、ユーザーは、帯域幅のバーニング (帯域幅の効率は犠牲になる) によって匿名性を高めることができます。

トラフィックの難読化

オーキッドのネットワーク層は WebRTC 上に構築されています。 WebRTC は、難読化のための特定の初期容量を提供します。しかし、難読化と検出の間には進行中の研究競争があります。[27]洗練された攻撃者は既知の難読化技術のほとんどを打ち負かすことができるため、強力な難読化プラグインの開発が今後の課題です (セクション 7)。

4. 市場設計

オーキッド マーケットは分散型のピアツーピア (P2P) ネットワークです。オーキッド クライアントを実行しているユーザーは、オーキッド サーバーを実行している 1 人以上の販売者から帯域幅を購入して、インターネット上の特定のリソース (ウェブサイトなど) へのプロキシ回路を形成できます。

オーキッド マーケットの主な参加者の役割は次のとおりです。

- プロキシ回路接続を開始する、オーキッド **クライアント**を実行している**ユーザー**
- (オプション) 暗号化されたトラフィックを転送する 1 つ以上の**リレー ノード**
- 外部宛先 (ウェブサイトなど) への最終的な接続を提供する**出口ノード**
- トラフィック (リレーまたは出口) のためにナノペイメントを受け入れる**帯域幅販売者**

帯域幅販売者は、ノードをイーサリアム ブロックチェーンに登録し、ユーザー クライアントは、イーサリアム スマート コントラクトへの呼び出しを介してルートに適したノードを選択します。オーキッドは、ステークウェイティングを使用します。販売者は OXT トークンをロックして、ノードに関連付けられたステーク デポジットを形成し、相対的なステークに比例するトラフィックを受け取ります。

4.1 基本的な操作

大まかに言えば、オーキッド マーケットは次の主要な操作を提供します。

- 帯域幅販売者がステーキングを介してノードを登録する手段
- 帯域幅販売者がカスタム サービスとメタデータを登録する方法
- クライアントが、提供されるカスタム サービスとメタデータのノードを照会する手段
- ステークに比例する確率でランダム ノードを選択する方法。*シビル直交性*プロパティ保持など (ノード X、ステークサイズ S、および乗数定数α):

$$P(\textit{選択}\,(X) \mid \textit{ステーク}\,(X) = \alpha S) = \alpha\, P(\textit{選択}\,(X) \mid \textit{ステーク}(X) = S)$$

シビル直交性には、リソースを複数のサブアカウントに分割する攻撃者が、単位時間あたりの選択確率と結果として予想される接続要求で有利にならないことを保証する線形選択プロパティが必要です。そのため、シビル攻撃には利点がなくなります。この線形に加重された選択プロパティが与えられている状況で、システムの合計ステーク S のうち集約ステーク A を持つ攻撃者が存在する場合 (何人でも)、ランダムに選択されたノードは以下の確率を備えた攻撃者では*ありません*。

$$P(\textit{選択}\,(\neg \textit{攻撃者})) = 1 - \frac{A}{S}$$

ステークウェイティングの使用により、オーキッド ネットワークの経済的セキュリティは、デポジットされた合計ステークのサイズに比例して線形にスケーリングすることができます。これは、OXT 時価総額のかなりの割合になると予想できます (ステークの経済性については、以下のセクション 4.5 で詳しく分析します)。ステーク加重選択プロセス自体は、以下のセクション 4.3 で説明するオンチェーン ツリー データ構造を使用して実装されます。これによってクライアントは、スケーラブルかつ信頼不要の方法でノードの選択を他のノードにアウトソースできるようになり、ウェイトの軽いクライアントが完全なノード ディレクトリをダウンロード、保存、または処理する必要がなくなります。

4.2 ノード ディレクトリ

オーキッドのノード ディレクトリは、イーサリアム ブロックチェーンに保存されるデータ構造のセットとして設計されており、クライアントが帯域幅販売者のノードを効率的に選択できるようにします。オーキッド ネットワークは実質的に、イーサリアム ネットワーク上でオーキッド固有のシンプルなオーバーレイを形成します。ノード ディレクトリ コントラクトは、いくつかの主要な機能を提供します。

- **プッシュ:** 様々な*量*の OXT トークンを特定の*ステーキー*にステークする方法。既存のエントリに追加するか、(ステーカー、*ステーキー*) を鍵とする新たなステーク デポジットのエントリーを作成する。プッシュ関数には、その後の引き出しロックアップ期間を決定する*遅延*パラメーターも必要です。
- **プル:** (ステーカー、*ステーキー*) に固定された既存のデポジット エントリーから保留中の様々な*量*の OXT トークンの引き出しを開始する方法。
- **テイク:** 遅延期間後に保留中の引き出しを完了し、引き出した資金を通常の流動性の高い OXT ERC20 残高に振り替える方法
- **スキャン:** ランダム シード パラメーターが与えられ、相対ステークで加重されたランダム ノードを選択する方法

ノード メタデータ レジストリ

ノード メタデータ レジストリにより、誰でもノードにメタデータを「タグ付け」することができます。帯域幅販売者は、これを使用して、ブロックチェーン上のノードに関連付けられたカスタム メタデータを保存し、サービスを宣伝できます。制約となるのはイーサリアム トランザクション料金のガス コストのみです。メタデータ レジストリは汎用であるため、将来のカスタム拡張の簡単な手段となります。これにより、ノード オペレーターは新しいサービスを宣伝することができ、クライアントはそれを、コード更新なしで選択できるようになります。

ノードディレクトリ ツリー

スキャン関数を効率的に実装するために、オンチェーンのバイナリ加重ツリー データ構造を使用します。ツリー内の各ノードは、ステーク エントリです。ステーク エントリには、ツリー ポインタと左右のサブツリーのステーク小計に加えて、ステーキー、金額、遅延が保存されます。この構造は実質的に、すべてのステーク デポジット上にプレフィックス サムのツリーを形成し、各ノードでの単純な降下決定で特定のランダム ポイントを含むサブツリー (またはノード間) を見つけられるようにします。指定されたランダム ポイントを含む正確なノード間を見つけるために必要なのはステップの対数のみです。

引き出しの遅延

引き出しの遅延は、重要なセキュリティ制限です。それは、オーキッド クライアントの接続リクエストの大部分を取得しようとする攻撃者にとって障害となります。私たちは特に、攻撃者がデポジット ステークの大部分を獲得し、クライアントを悪意のあるサーバーに誘導する *システミック テークオーバー攻撃*の防止について懸念しています。悪意のあるサーバーは、意図的に貧弱な接続を提供し、トラフィックをログに記録して報告するか、アクティブな接続攻撃 (SSL ダウングレードなど) を試みます。

プルーフオブステーク (POS) 暗号通貨と同様、システミック テイクオーバー攻撃に対する私たちの主な防御策は、OXT ステーク全体の大きな部分を取得しロックアップするコストを非常に高くすることです。引き出しの遅延がなければ、この障壁は単に十分な流動性にアクセスできないという問題になり、実際の攻撃の純コストはわずかです。引き出しの遅延は、ステーク ポジションの最低利息または機会費用を発生させます。攻撃が成功すると、ネットワークが混乱し、おそらく OXT トークンの価値が低下します。そのため、十分に長い引き出し遅延は、攻撃者が最終的に攻撃を終了し、大規模な OXT ポジションを売却するまでに、追加の損失が発生する可能性を高めます。

根底のメカニズムはかなり異なりますが、引き出し遅延の期間が短いオーキッドにおける伴うシステミック攻撃は、*プルーフオブワーク* (PoW) ブロックチェーン システムの *レンタル攻撃*に似ています。Nicehash などのハッシュパワー レンタル サービスの台頭[10]によって、必要なハードウェアを購入する代わりに、使用可能なハッシュパワーの流動性の大きなプールが提供され、PoW システムに対する 51% 攻撃のコストが劇的に低下しました。攻撃者は、レンタル ハッシュパワーを使用して、多くの小さなコインに対して二重支出攻撃を実行しました。上位 20 コインであるイーサリアム クラシックに対してさえ、2019 年初頭に、攻撃が成功しました。[11]

理想的な引き出し遅延は、市場がシステミック テイクオーバー攻撃を検出して対応する必要があると予想される時間よりも長くする必要があります。しかし、引き出しの遅延が長くなると、ステーク デポジットのポジションの削減や、引き出しを望む正直な帯域幅販売者に機会費用がかかります。この 2 つの制約間の理想的なトレードオフを先験的に推定するのは難しいため、私たちは引き出し遅延を柔軟なパラメーターにすることを選択しました。その後、クライアント ソフトウェアは、引き出しの遅延に基づいて *フィルタリング*し、遅延がクライアントのしきい値未満のステーク デポジットを無視します。私たちの最初のクライアント ソフトウェアは、3 か月以上の引き出し遅延を受け入れますが、これを柔軟なパラメーターにすることにより、将来のクライアントの更新で、ハードフォークや関連する困難な調整をすることなく、このパラメーターを変更できます。

---

[10] https://www.nicehash.com/
[11] https://cointelegraph.com/news/ethereum-classic-51-attack-the-reality-of-proof-of-work

4.3 ノードの選択

クライアントは、ランダムな相対ステーク加重線形選択と、それに続く 2 次制約フィルタリングによる 2 段階プロセスを使用して、プロキシ回路のノードを選択します。第 1 段階の線形選択は、ノード ディレクトリ ツリーのスキャン関数によって実行されます。クライアントはローカルにランダムなポイントを生成し、それをスキャンの単一引数として渡します。この引数は、ノード ディレクトリ ツリーを下っていきます。この検索は、ステーク セグメントが選択されたランダム ポイントと交差する単一かつ一意のリーフ (またはノード間) で終了します。

スマート コントラクトを使用してメイン ノードのスキャン関数を実装することにより、この選択プロセスは容易にノードにアウトソースできます。クライアントは、1 つ以上のスキャン呼び出しを要求し、リモート ノードに各スキャンをローカルで実行させ、イーサリアム JSON RPC API の eth_getProof 関数と eth_getStorageAt 関数を使用して、正確性の簡単な証明を送り返すことができます[12]。このメカニズムにより、クライアントは、ノードが悪意を持って自分自身またはエイリアスを選択しなかったことを確実に信頼でき、イーサリアム ブロックチェーンの完全なコピーでローカルに機能を実行した場合と同じ結果を返しました。スキャン機能のアウトソーシングにより、オーキッド クライアントの軽量な実装が可能になります。

線形の相対的なステーク ウェイティングに基づいて 1 つ以上のノードを選択した後、クライアントは任意で、出口のジオロケーション、レイテンシ/ping、ノード ホワイトリスト、またはカスタム メタデータ タグなどの追加基準によるフィルタリングを実行できます。

ジオロケーション

今日の VPN の人気の高い使用例は、ジオロケーションに基づいたコンテンツ フィルタリングを迂回することです。Netflix などのストリーミング サービスには国固有のコンテンツ ライセンスがあり、このライセンスはユーザーの IP アドレスを検出することで実行されます。したがって、適切な場所にある VPN サーバーや出口サーバーは、通常はブロックされているコンテンツにアクセスすることが可能になります。

特定の IP アドレスが実際に特定の場所にあることを証明するのは困難です。さらに、クライアントが接続する最終的なサーバーは、正当な理由により、ディレクトリ コントラクトにリストされているものとは異なる IP アドレスを持つ場合があります。大規模な帯域幅プロバイダーは、負荷分散のために、着信クライアント接続を多くのプロキシ サーバーの 1 つにリダイレクトする可能性があります。これらの考慮事項により、特定の出口ジオロケーションに関心のあるオーキッド クライアントは、公開されたノード メタデータを使用して、要求されたジオロケーションをフィルター処理できますが、最終的に、最後の出口接続が実際に要求されたロケーションにあるかどうかを確認する必要があります。このチェックは、位置情報データベースへのパブリック IP アドレスの使用により、ある程度自動化できます。

[12] https://github.com/ethereum/wiki/wiki/JSON-RPC#eth_getproof

レイテンシ

場合によっては、ユーザーはランダムに選択されたノードよりもレイテンシの短い接続を望むと予想しています。クライアントは、ジオロケーションに使用されるものと同様のレイテンシの推測およびチェック戦略を採用できます。主張されている IP アドレスを確認するには、その IP アドレスをロケーションにマッピングして既知のデータベースと照合すれば、離れたサーバーを除外できます。最終的に、ルートが構築されたら実際のレイテンシを測定する必要があります。レイテンシがターゲットのしきい値よりも高い場合、新しい異なるルートをサンプリングする必要があります。オーキッド ルートとナノペイメントの軽量な性質により、高速ルート設定とパラレル ルート テストが可能になります。

価格

販売者が独自の帯域幅の価格を設定するとき、手ごわい料金を避けるために、クライアントは合理的な価格レベルを決定できなければなりません。オーキッド クライアントは、ユーザーの残高と、予算がどのくらいの期間維持されるべきかを表す目標に基づいて、現在の支出上限を決定するカスタマイズ可能な予算アルゴリズムを使用します。たとえば、ユーザーは 50 ドル相当の OXT を自分のナノペイメント ウォレットにロードし、その資金を 1 年間の帯域幅購入の予算とするようにクライアントに指示できます。クライアント ソフトウェアは、この予算を使用して、その期間に支払う金額の制限を決定します。クライアントが使用している帯域幅に対して、サーバーが課金している金額よりも少ない金額をクライアントが支払うと、サーバーは接続を制限します。調整されたスループットが許容できないほど低い場合、クライアントは新しいプロバイダーを選択します。したがって、価格は暗黙的なフィルターを形成し、クライアントの現在の使用率および予算支出率と互換性のない帯域幅の価格を持つノードを除外します。

ホワイトリスト

オーキッド クライアントは、実行可能なノードをカスタム サブセットにフィルタリングするオンチェーン キュレーション リストを使用できます。公式のオーキッド クライアントの初期リリースでは、この機能によって信頼できる VPN パートナーで構成されるデフォルトの出口ノード ホワイトリストを使用し、悪意のある出口ノードからの特定の種類の攻撃 (SSL ダウングレード攻撃など) を防ぐことを想定しています。カスタマイズされたオーキッド クライアントは独自のホワイトリストを使用できますが、最終的には、有名なサードパーティがホワイトリストのキュレーターとして登場することが期待されています。ホワイトリストは、ステーキングによって提供される経済的インセンティブ ベースの信頼を、外部の評判による信頼で補完するための簡単な手段です。

カスタム メタデータ タグ

帯域幅販売者は、ノード メタデータ レジストリを使用して、ブロックチェーン上のノードに関連付けられた任意のメタデータ タグを保存できます。将来、販売者はこれを使用して、共有されていない IP アドレスなど、新しいカスタム サービスを宣伝できます。ユーザーは、関連するタグでクライアント フィルターを使用して、そのサービスを提供していると主張するノードを見つけることができます。虚偽の広告 (実際に提供していないサービスを主張している) を行う販売者は、人気のあるホワイトリストから除外されるリスクを冒しています。

4.4 ステーク ウェイティングの選択

オーキッド 0.9.2[6] はプルーフオブワークのメダリオンに基づいた設計をシビル攻撃に対抗する主要なメカニズムとして提示し、プルーフオブステークに対して明示的に議論しました。このセクションでは、ステークウェイティングと他の選択肢を分析し、プルーフオブステークに類似したステークウェイティング アプローチに移行した理由を説明します。

事前準備: 攻撃コスト

ビットコイン、イーサリアム、および他のほとんどの分散型システムと同様に、オーキッドはオープン ソース ソフトウェアから構築されたオープン ネットワークであり、誰でもオーキッド ノード ソフトウェアをダウンロードして、リソースが許す限り多くのノードを実行できます。オープンな分散型システムがシステミック攻撃に対して実行できる防御は、最終的には*経済性*です。つまり、システムは、攻撃コストがその攻撃者にとっての利益を上回っている限り、またはいずれにしろコストがかかりすぎて実行できない場合には安全です。

経済的セキュリティは、絶対的制約と相対的制約に分けることができます。相対的な経済的セキュリティは、必要なリソースに関わりなく、攻撃が*不採算*であるという状況です。絶対的な経済的セキュリティは、高コストの障壁自体がセキュリティとなり、リソースが不十分な攻撃者を排除します。ビットコインには現在、数百億ドルの絶対的な経済的セキュリティがあります。新しい暗号通貨の場合、絶対的セキュリティははるかに低いかもしれませんが、それでも、現実的な攻撃者を阻止するのに十分な相対的セキュリティに依拠することは可能です。

プルーフオブワーク

プルーフオブワーク システムは、システム内で有効な ID を証明するために無駄にしなければならない計算能力からセキュリティを引き出します。オーキッド 0.9.2 の設計 [6] はメダリオンを使用していました。メダリオンは、新しいイーサリアム ブロックごとにシードされる計算パズルを解くことで現在のアクティブ ステータスを維持するため、*継続的*なプルーフオブワークを必要とします。したがって、そのメカニズムは、ビットコインなどのプルーフオブワーク ブロックチェーン システムに非常に似ています。

プルーフオブワークの設計に ASIC 耐性が*ない*ために、特殊なチップは一般的なチップよりも劇的に効率的で、*かつ*、そのチップに大きなレンタル市場が存在しないと仮定した場合、プルーフオブワーク システムの経済的セキュリティの制約はほぼ、以下のようになります [28]。

$$N\,C > V_{sabotage} \qquad (3)$$

ここで、N は誠実な (攻撃者ではない) ハッシュパワーの合計、C は単位ハッシュパワーあたりの総資本コスト、$V_{sabotage}$ は攻撃者がシステム妨害から導き出す価値を表しています。式 3 の左辺は攻撃コストであり、絶対的なセキュリティ バリアでもあります。

2019 年半ばのビットコインの場合、NC の価値は数百億ドルです。ビットコインのプルーフオブワーク仕様は ASIC 耐性がないため、ASIC チップは、再利用可能な汎用チップよりも桁違いに高い効率によって圧倒的に多く使用されています。一方、イーサリアムは、意図的に ASIC 耐性のプルーフオブワーク仕様を設計しました。その結果、ASICS は、イーサリアムのマイニングを支配していた汎用グラフィックス処理ユニット (GPU) に比べて最小限の利点しかありません。GPU の目的は一般的なものであるため、GPU には流動性のあるレンタル マーケットが存在します。そのため攻撃者が支払う必要があるのは攻撃中のハッシュパワーのレンタル コストのみです。攻撃中に攻撃者が獲得するブロック報酬を無視すると、レンタル攻撃に対する経済的セキュリティによる制約は、レンタル コストの時間単位 *t*、単位ハッシュパワーあたりの単位時間レンタル コスト *c* の場合、およそ以下のようになります。

$$t\,\mathrm{N}\,c > \mathrm{V}_{\mathrm{sabotage}} \qquad (4)$$

攻撃に必要な時間である *t* は一般にハードウェアの減価償却期間よりも桁違いに短いため、レンタル シナリオは経済的セキュリティを劇的に低下させます。オーキッド 0.9.2 [6] のプルーフオブワークであるメダリオンの設計は、ASIC 耐性スキームである equihash[] に意図的に依存していました。メダリオンはエンド ユーザーが生成する必要があるという要件を考えると、エンド ユーザーの多くは携帯電話レベルのハードウェアしか持っていないと考えられるため、これは必要なことと言えました。その結果、ASIC に対応したプルーフオブワーク アルゴリズムでは、携帯電話の CPU を使用するエンド ユーザーに対して、ASICS を使用する攻撃者は非常に有利になります。残念ながら、ASIC 耐性アルゴリズムの使用は、流動的なレンタル市場が存在することを意味するため、上記の式 #4 のセキュリティは低くなります。

プルーフオブワークのパズルに費やされた計算は無駄になります。それは、システムが提供する帯域幅の正味の価値に対する税金のようなものです。単位時間あたりの収益 P は、帯域幅のコスト B とメダリオンを維持するために必要な計算の暗黙的なコストの和に等しくなります。

$$\mathrm{P} = \mathrm{B} + \mathrm{N}\,c \qquad (5)$$

経済的考慮によって、N*c* と B はほぼ同等となります。そうでない場合、オーキッドはコンシューマーにとって、代替品と比較して価格が高すぎることになります。式 5 を式 4 に代入すると、セキュリティ条件が算出されます。

$$t\,(\mathrm{P} - \mathrm{B}) > \mathrm{V}_{\mathrm{sabotage}} \qquad (6)$$

具体的な例として、オーキッドのユーザーが 100 万人存在し、その一人一人が合計で年間約 63 ドル (卸売帯域幅コストに暗黙のプルーフオブワークの計算コストを加算した金額) を支払い、プルーフオブワークのオーバーヘッドはコストの約 50% であると仮定します。この場合、P - B は 1 秒あたり約 1 ドルにすぎません。これらのパラメーターを使用すると、攻撃者は、約 3,600 ドル相当のレンタル コンピューティングだけで 1 時間のすべてのオーキッド トラフィックの約半分、約 86,400 ドル相当のコンピューティングで 1 日のすべてのオーキッド トラフィックの約半分をキャプチャできることになります。

ステークウェイティング

現在のステークウェイティング アプローチでは、帯域幅販売者が OXT 通貨をタイムロックされたデポジットにステーキングして身元を証明し、相対的なステーク デポジット サイズに比例してトラフィックを受け取ります。最初に、攻撃に役立つのに十分な流動性を備えた OXT を借りられる市場は存在しないと仮定します。オーキッド トラフィックの 50% を制御するには、攻撃者は非攻撃者の合計ステークに等しい量の OXT を獲得してステーキングする必要があります。攻撃が成功すると、OXT の交換価値が低下します。攻撃の主なコストは、ステーク ポジションの損失です。S が正直な (非攻撃者) ステーク デポジットの合計で、x_d は攻撃後の OXT の交換価値の (予想される負の) 変化率とすると、相対的なセキュリティ条件は次のようになります。

$$-x_dS > V_{sabotage} \qquad (7)$$

攻撃者は攻撃を実行するためにサイズ S の資本を使う必要があるため、攻撃コストと絶対的なセキュリティ バリアは S (ステーク デポジットのサイズ) だけです。

私たちは、帯域幅販売者が市場の状況に応じてステーク デポジットを増減させることにより、全体の収益性を最適化すると予想しています。帯域幅販売者がトラフィックを受けとるために OXT 通貨をロックする必要があるという要件は、帯域幅販売者の資本に暗黙の機会費用が発生することを意味します。競争均衡では、帯域幅販売者に流入する総収益 R は、帯域幅のコスト B と単位時間 I_r あたりの機会コストまたは金利の和を必要とされるステーク資本で乗じたものとほぼ等しくなります。

$$R = B + I_rS \qquad (8)$$

そこで、合計ステーク S は、帯域幅のコスト、収益フロー、および金利に関して次のように書き換えることができます。

$$S = (R - B) / I_r \qquad (9)$$

ステーク デポジット資本の機会費用は、プルーフオブワークの例における破棄される計算費用と同様の役割を持つオーバーヘッドの一形態です。50% のオーバーヘッドという同じ仮定を立てると、機会コストは帯域幅のコストに等しくなります。前の例と同じパラメーターを使用して、年間 100 万人のユーザーが 63 ドル相当の帯域幅を購入し、その 50% がサプライヤーの帯域幅コストに使用され、年間 10% の金利または機会コストを想定すると、式 9 によってステークの合計額 S は 3 億 1,500 万ドルとなります。これは、式 7 から算出される絶対攻撃コストの制約でもあります。この額は、継続的なプルーフオブワークであるメダリオンを使用した攻撃コストよりも 3 桁以上大きくなります。

今度は、OXT ステークの流動的なレンタル市場がある場合のシナリオを考えてみましょう。まず、借り手が別の通貨で担保を設定する金融市場を想定してみます。これは、ショート ポジションに似ていますが、資金の使用に制約はありません。この種のレンタル市場がある場合でも、攻撃コストと S の絶対的セキュリティ制約は変化しませんが、攻撃者が OXT の価値の低下による損失を回避できるようになるため、相対的なセキュリティ制約のダイナミクスは変化します。

攻撃者にとってより有用なのは、担保なしでステーク デポジットを直接貸し出す市場です。デポジットは非流動的であるため、賃借人はそれらを使用することはできませんが、オーキッド ノードのトラフィックに関しては、ステーク デポジットのすべてのメリットを利用できます。このシナリオでは、攻撃コスト、相対的および絶対的なセキュリティ制約は、利息コストのみを含む流動方程式に変更されます。

$$t\, I_r S > V_{sabotage} \qquad (10)$$

上記の式 10 で、攻撃コストは、攻撃期間 *t* 中、攻撃前の合計ステーク (「正直な」ステーク S の残り) の 50% を借りる利息だけになりました。式 9 の右辺を式 10 の S に代入すると、プルーフオブワーク セクションに記述されている式 6 と同じになります。

$$S = (R - B) / I_r \qquad (9)$$
$$t\,(P - B) > V_{sabotage} \qquad (6)$$

そのため、ステークが完全にレンタル可能なステークウェイティングの最悪のケースは、ハッシュパワーが完全にレンタル可能なプルーフオブワークと同様の脆弱なセキュリティ条件となります。

ただし、引き出し遅延パラメーターは、非常に重要な攻撃時間パラメーター *t* に下限を設定します。以前と同じパラメーターを使用して、100 万人のユーザーが年間 63 ドルと 50% のオーバーヘッドを支払う場合、3 か月の引き出し遅延により、約 790 万ドルの攻撃コストが発生します。これは依然、プルーフオブワークのメダリオンに対する攻撃コストよりも数桁大きな金額です。

ステークの引き出し遅延をさらに長くすることも提案できますが、引き出し遅延によって経済的セキュリティが単調に強化されることはまずありません。引き出し遅延は、ステーク ポジションを解消する誠実な参加者にも追加の機会費用を発生させるため、その費用が高すぎると、競争力のある帯域幅販売者を圧迫する可能性があります。その場合、実効金利 I_c が上昇し、または帯域幅の基本コスト B_c が引き上げられ、実質的にシステム効率が低下します。引き出し遅延は、市場が最終的に決定するカスタマイズ可能なパラメーターです。

OXT は特殊な資産であり、その主要な保有者は、審査を受けていない未知の事業体に巨大なステーク ポジションを貸し出すことを奨励されていません。その意味で、OXT のレンタル ダイナミクスは、レンタルに使用できるハッシュパワーが全体のごく一部にすぎないビットコイン ASIC のレンタル ダイナミクスに似ていると言えます。ホワイトリスト メカニズム (セクション 4.3) は、ステークホルダーが、自分自身がホワイトリストから削除されるリスクを冒してまで、同じホワイトリストに載っていない事業体にレンタルすることをさらに強く阻止することで、ステーク レンタル市場の安全に役立つと考えています。本質的に、これにより、リストからの削除 (引き出しの遅延により発生する) のペナルティが、オペレーターである賃貸人からステークホルダーである賃借人に強制的に移行します。

バーンウェイティング

また、破壊されたことが証明可能なステーク デポジットが OXT 通貨に置き換えられるバーンウェイティング モデルも検討しました。バーンウェイティングは、実際には、無限の引き出し遅延を伴うステークウェイティング モデルと同等です。引き出し遅延が無限の場合、ステーク デポジットは実質的に燃やされた (破棄された) ことになるからです。式 7 のパーセンテージの減少 x_d はわずか-1 となります (常に

完全なポジションが失われるため)。したがって、この式は、攻撃コストの条件を、単なる (バーンされた) ステーク デポジットの合計に単純化します。

そのため、引き出し遅延の長期化に伴う経済的セキュリティの非単調性に関する議論は、バーンウェイティング (無限の引き出し遅延) にも当てはまります。遅延が長期化すると、ステークホルダーは資本のデポジットの選択肢を失うため、失われた選択肢を補うためにより高い実効金利を要求する傾向があります。

バーンウェイティングはすでに現在のステークウェイティング設計のパラメーター モードであるため、将来的には引き出し遅延を徐々に長期化することにより、バーンウェイティング モデルに移行することができます。もちろん、長期化を拒否するクライアントには分岐または市場セグメンテーションのリスクがありますが、理論的にはそのような変更は十分に可能であり、引き出し遅延をパラメーター化することで、より容易になります。

利息ウェイティング

検討した最後の代替案は、直接のステークウェイティングを、ウェイティング期間としての引き出し遅延に対するステーク デポジットの実効金利または機会費用に置き換えることです。この設計の背後にある動機は、ステーカーの時間依存のロックアップ コストをより直接的に補償することにより、多様な引き出し遅延を奨励することです。利息ウェイティングのウェイティング期間は $(1 - e^{(-w_t I_r)}) S$ となります。ここで w_t は変数演算子によって決定された引き出し遅延、I_r はグローバルな「金利」パラメーター、S はステーク デポジットのサイズです。

この利息ウェイティング設計では、主要な設計パラメーター I_r は、おそらく実際の市場金利または OXT ステーク デポジットの機会費用に近い値に設定する必要があります。金利条件 I_r が市場金利よりもはるかに低い場合、参加者は w_t が無限大 (またはその最大値) の引き出し遅延を選択するように奨励され、システムは崩壊してプルーフオブバーンの形態になります。I_r が市場金利よりもはるかに高い場合には、参加者は非常に短い引き出し遅延を選択するように奨励され、このシステムは短い引き出し遅延を伴うステークウェイティングに似たものとなります。

システミック攻撃が成功すると OXT の価値、ひいてはステーク ポジションの価値が大幅に低下するため、真剣な攻撃者は事実上 OXT に対して非常に高い金利または機会費用を抱えていることになります。彼らは OXT の価値が崩壊すると信じているからです。したがって、金利条件 I_r が市場金利に近いと仮定すると、真剣な攻撃者は当然、非常に長い引き出し遅延を選択し、利息ウェイティングとステークウェイティングの攻撃コストを効果的に引き下げようとするでしょう。それは、このような条件下では、ほとんどの市場参加者が合理的な引き出し遅延を選択するからです。その結果、ウェイティング期間の値は 1 よりもはるかに小さくなり、ステークウェイティングに対するステーク デポジットの合計は小さくなります。一方で、攻撃者はウェイティング期間 1 に対して無限の遅延を選択します。

これらのセキュリティ上の懸念、市場の均衡に向かうグローバルな金利パラメーター I_r を調整するための未知の動的メカニズムによる複雑さの増大、最後に累乗と乗算を含む複雑なウェイティング関数によるイーサリアム実装の懸念などにより、私たちは利息ウェイティングには反対することにしました。

要約

私たちがステークウェイティングに移行したのは、以前のプルーフオブワークのメダリオン設計と比較して、次の重要な利点があるからです。

1. プルーフオブワークを使用すると、エンドユーザーに追加の計算負荷が発生する
2. プルーフオブワークに対する攻撃コストは、レンタル マーケットを想定しても、遅延を伴うステークウェイティングよりはるかに低い
3. 一般的なコンピューティング レンタル市場はすでに存在しており、その流動性は、将来の OXT ステーク デポジット レンタル市場で予想される流動性をはるかに上回っている
4. ステークウェイティングは、帯域幅販売者の将来の割引利益を捕捉するため、ベースラインとなるトークンの時価総額がより大きな規模となる。このトピックについては、次のセクションで説明します。

4.5 トークノミクス

ステークウェイティングは、大きな価値を獲得できるという点で、ユーティリティ トークン システムの競合メカニズムに対して明確な利点を持っています。このセクションでは、関連する経済的仮定の一部を簡単に説明、分析して、ユーザーのナノペイメント デポジットとノード ステーク デポジットに焦点を当てたシンプルなモデルに落とし込みます。これらのカテゴリ以外の付加価値要素 (ERC20 トークンの短期高速回転それ自体など) の貢献は比較的小さいと想定しています。

市場規模

まず、オーキッドが月平均 5 ドルを支払う 200 万人のカスタマーを擁している、すなわち年間 1 億 2,000 万ドルの総システム収益を得ているというシナリオから始めます。参考までに、世界の VPN 市場の規模は 2020 年に 270 億ドルに達すると予想されています[13]。

ユーザーのデポジット

私たちは、ほとんどのユーザーが少なくとも 3 か月分の帯域幅、この例では 15 ドル相当の OXT を支払うのに十分な OXT をナノペイメント アカウントに事前に提供すると予想しています。数か月または数年分のサービスの前払いは VPN 市場の標準的な支払いモデルになっているため、VPN ユーザーは既にこの支払い方法に慣れています。

したがって、この例でユーザーのデポジットの合計額は 3,000 万ドルとなります。

[13] https://www.statista.com/statistics/542817/worldwide-virtual-private-network-market/

ノード ステーク デポジット

オーキッドは競争力のある帯域幅市場であるため、私たちは最終的には、総収益が基礎となるコストに近づき、ほぼ均衡に達する状態までシステムが進化することを期待しています。この基礎となるコストには、サプライヤーにとっての帯域幅の原価と、ステーク デポジット資本の利息費用または機会費用の両方が含まれます。*セクション 4.4* の式 8 と式 9 を思い出してください。

$$R = B + I_r S \qquad (8)$$

$$S = (R - B) / I_r \qquad (9)$$

ここで、R は総収入フロー、B は販売者の帯域幅原価、I_r は実効金利 (機会コスト)、S は総ステーク デポジットです。

現在、保有者がノードを実行することで、自身が保有するステークに関する利息を得る仕組みのプルーフオブステークの暗号通貨システムはたくさんあります。各コインのステーキングに対する利率は、システムの詳細、認識される交換リスクなどに基づいてかなり異なります。OXT ステークの有効な APR (年率) は 20% であり、これはステーキングの収益率として典型的な範囲内です[14]。

IP のトランジット価格は場所によって異なりますが、妥当な中央値の見積もりは 1 Mbps あたり 1 か月 1 ドルです[15]。300 GB を超えるデータについては 1 か月 1 ドルまたは GB あたり約 0.003 ドルになります。私たちは、卸売帯域幅価格である 0.01 ドル/GB を使用します。米国のブロードバンド世帯の月間平均データ使用量は、月あたり 268 GB です[16]。したがって、1 か月あたりのカスタマーごとの VPN データの推定として 100 GB /月を使用します。これは、1 ユーザーあたりの帯域幅の原価として 1 ヵ月 1 ドル、1 年で 12 ドルを意味します。これにより、B の条件での 1 年間の総帯域幅コストは 2,400 万ドルになります。

したがって、この例のノード ステーク デポジットの合計額は、式 9 により、約 4,800 万ドルとなります。

ビットコインなどの暗号通貨は、主に*価値の保管場所*として使用されます。オーキッドを動かす OXT 暗号通貨はイーサリアム ブロックチェーン上に実装される *ユーティリティ トークン* です。ユーティリティ トークンが価値の保管場所として使用される可能性や、オーキッドのナノペイメント システムがオーキッド外部で使用される可能性もありますが、ステーキング メカニズムが価値の大部分を獲得すると予想されています。現在、ステーキング報酬と、様々な APR 利回りを提供する多くの暗号通貨システムがあります。これらのステーキング コインのステーキング率 (時価総額を超える合計ステーク デポジットの価値) も非常に多様で、Decred のステーキング率は 50%[17]、NXT のステーキング率は 15% です18。

[14] https://stakingrewards.com/
[15] https://blog.telegeography.com/yup.-price-erosion-is-still-a-thing
[16] https://www.telecompetitor.com/report-u-s-household-broadband-data-consumption-hit-268-7-gigabytes-in-2018/
[17] https://stakingrewards.com/asset/dcr
[18] https://stakingrewards.com/asset/nxt

5. ナノペイメント

5.1 イントロダクション

現在、ほとんどのレイヤー 1 オンチェーン支払いオプションは、主に、長い確認時間、低いスループット、高いトランザクション料金に関連する使いやすさの欠如に悩まされています。例として、イーサリアムとビットコインの確認時間はそれぞれ 15 秒と 10 分、取引手数料は約 0.10 ドルです。[29] [30] オーキッド ネットワークでは、パケット送信 (および拡張、帯域幅) に価値を関連付けます。したがって、パケットを送信するためのトランザクション料金と確認時間が、現在のレイヤー 1 ソリューションが提供するのと同じほど高い場合には、オーキッドのネットワーク経済は完全に崩壊します。簡単に言えば、パケットの送信に関連するトランザクション料金と確認時間がパケット自体の値と伝播時間より桁違いに大きくなるのは望ましくありません。

支払いのスケーラビリティ要件は必然的に、ネットワークの支払いバックボーンとしてレイヤー 2 マイクロペイメント ソリューションの使用を示唆しています。ただし、データ送信は支払い情報と密接に関連しているため、帯域幅とパケットに関するオーキッドの保証を支払いにも適用する必要があります。特に、インターネットの監視と検閲を減らすというオーキッドの目標は、データ送信プロトコルと支払いプロトコルの両方に検閲耐性があり、匿名で、かつ、分散型または信頼不要のものであることを意味します。以下では、これらのユースケース要件を技術的評価ポイントに分解して、既存の作業と提案されたプロトコルの両方が、支払いに関するオーキッドの主要な課題をどれだけうまく解決できるかを評価します。

スケーラブル: このシステムは、頻繁に小さなトランザクションを行う (1 秒に 1 回程度) 数百万人のユーザーをサポートする必要があります。これは、1 支払いあたりのトランザクション料金が非常に少額であることを意味します。

信頼不要: このシステムは、機能が特定のパフォーマンスや善意に依存するなどのように、参加者に特定のエンティティを信頼するよう要求すべきではありません。

匿名性: 支払いで漏洩する実世界の身元に関する追加情報は最小限とすべきです。さらに、資金の送信、受信、または伝播を疑われるシステム内のすべての関係者に対する否認権が必要です [31]。

検閲不能: 攻撃者によるトランザクションの検閲が法外に高価である必要があります。これは大まかに言うと、情報を破損したり、そのアクセスや公開を防止することが経済的または暗号的に実行不可能であることを意味します [31]。言い換えれば、支払いやパケットを検閲しようとする悪意のある行為者によってネットワークの大部分が制御されない限り、任意のエンドポイントに破損なしに資金を送受信する何らかの方法を見つけることが可能なはずです。

以下のセクションでは、既存の支払いソリューションについて、それらが上記の評価フレームワークにどのように適合するかについて説明し、私たちの特定のユースケースに対しては、オーキッドの支払いフレームワークの方が既存のソリューションよりも優れた保証を提供できることを示します。

5.2 既存の作業との比較

上記で提案したように、潜在的にはパケット レベルまで、任意の量の帯域幅に関連付けられた価値を転送するための前提条件は、堅牢なマイクロペイメント インフラストラクチャが存在することであり、その中ではレイヤー 2 ソリューションが最も人気があります。レイヤー 2 ソリューションは、チェーン上の支払いのセキュリティを、すべてのトランザクションでメイン ブロックチェーンに直接関与しないプロトコルと結び付けます。理論的には、これにより、取引手数料の削減、確認時間の短縮など、いくつかの大きなメリットが得られます。残念ながら、現在、エコシステムで利用できる生産準備の整ったマイクロペイメント ソリューションは存在しません。*セクション 5.1* で説明した主要な評価ポイント内で、既存のスキームの失敗を調査し、確率的価値交換のための新しいナノペイメント プロトコルの提案に進みます。

5.2.1 集中化された支払い

従来の金融支払いは、銀行や決済サービス プロバイダーなどの当事者間交渉を通じて決済される取引です。これらの決済はしばしば、支払いカードの場合は ISO/IEC 7816 [32]、給与計算やクレジット振替の場合は ACH[19]、ATM トランザクションの場合は NYCE [36] および SWIFT [34] などの集中型プロトコルによって行われます。これらのネットワークの参加者は、電子支払い領収書と手動調整を組み合わせて、ローカルの元帳を中央ネットワークと同期します [37]。

残念ながら、集中型の支払いシステムは、*セクション 5.1* に列挙されている要件のほとんどをサポートしていません。集中型の金融エコシステムにおける詐欺の蔓延 [38]、ならびに詐欺に対するソリューション、つまり逆取引 [39] は、いずれも*信頼不要*操作の原則に反しています。集中型システムでは応答性は非常に高いものの、ビザンチン型のフォールト トレランスとサブシステム間の相互運用性の欠如により、グローバル システムが一部しか利用できないと同時に、一貫性の問題に悩まされることになります。最後に、支払いインフラストラクチャに参加し、管理している信頼できる当事者は、通常、各トランザクションに関する詳細なメタデータ (送信者、受信者、金額、時間) を持っており、検閲と匿名化に関与し、それを遵守するために必要なすべての要素を備えています [40]。

オーキッド 0.9.2 [6] で述べたように、集中型支払いの取引手数料は、支払いカード取引 [41] のわずか数セントから、国際電信送金 [42] の 75 ドルまで大きな幅があります。多くのシステムは、それらの料金の代わりに、またはそれに加えて、支払いカード [43] の場合の 3.5% から、銀行振り込み [44] の場合の 13% までの料金を請求します。通常、固定料金はマイクロペイメントには不適切なサイズですが、パーセンテージによる手数料のシステムはマイクロペイメントに合理的なバックボーンを提供できます。特に、アジアでの WeChat Pay と Alipay の採用は、通常 0.0%ー0.1% の非常に低いパーセンテージベースの手数料が、商業的に実行可能であることを示しています [45]。しかし残念ながら、これらのシステムは、前述の集中型のすべての欠点を抱えています。

19 https://en.wikipedia.org/wiki/Automated_clearing_house

F =全機能、P =一部機能、N =機能なし

スケーラブル	信頼不要	検閲不能	匿名性
[N、P、F]	N	N	N

5.2.2 支払いチャネル

支払いチャネルは、従来のレイヤー 1 ブロックチェーン システムのセキュリティと保証を拡大するための、新しいレイヤー 2 ソリューションです。ビットコインの Lightning ネットワーク [46] は、このタイプのソリューションを検討した最初のプロトコルの 1 つでした。抽象的なレベルでは、ほとんどの支払いチャネルには 3 つのステップがあります。すなわち、エスクロー内の資金のロック、それらの資金をオフチェーンで使用した取引、支払いチャネル閉鎖時におけるエスクローへの最終状態のブロードキャストと 2 人のチャネル参加者への支払いです。

ただし、既存の支払いチャネル インフラストラクチャには多くの問題があるため、オーキッド ネットワークでは使用できません。第一に、支払いチャネルを介した資金のルーティングの複雑さは平均で $O(log(n))$ ホップです。ここでn は、ネットワーク内のノードの数を表します。エンドツーエンドの支払いルートの各ホップは、ネットワークのコストが非常に低く、主にルーティング/計算コストに集中していますが、ルート全体では、支払いチャネルのペアごとのセットアップとティアダウンのコストもかかります。これに関連する問題は、ネットワーク内の 1 つのホップが支払いに失敗すると、ルート全体を停止させるタイムアウトが発動される可能性があることです。これは、$O(c * n)$ のセットアップと分解の複雑さが、支払いチャネルの平均寿命にわたって償却されることを意味します。ここで c は、各ノードが維持する支払いチャネルの数を表します。さらに、資金のロックアップ コストがあります。資金が支払いチャネルにロックアップされている場合、他の場所で使用することはできません。これは、多くのノードとピアリングする場合に問題になります。ノードはすべてのトークンをピアへのマイクロペイメントに使用できますが、ロックされた各トークンは単一のピアとしか対話できません。

支払いチャネルは通常、ハッシュ タイム ロック コントラクト (HTLC) によって、ルート チェーンに関して暗号的に実行されることに注意してください。支払いチャネルの取引手数料も通常は安価です。ただし、支払いチャネルの検閲可能性はもう少し微妙な問題です。ビットコイン ネットワークの場合、Heilman のエクリプス攻撃 [47] の分析によると、ビットコインのノードへの接続を妨害するには、わずか 400 の IP アドレスのボットネットを使用するだけで、$> 50\%$の確率で実現可能だとしています。この攻撃を支払いチャネルに適用するには、ノードがより大きな L1 ネットワークと通信できない必要があります。これは、ピアリングが実際に処理される方法によって大きく異なるため、エクリプス攻撃の複雑さは L1 プラットフォームによって異なります。匿名性とプライバシーに関しては、残念ながら、現在の支払いチャネルテクノロジーでは非常に限定されています。

F =全機能、P =一部機能、N =機能なし

スケーラブル	信頼不要	検閲不能	匿名性
F	P	P	N

5.2.3 確率的マイクロペイメント

確率的マイクロペイメントの概念は、Wheeler [48] と Rivest [49] によって 1990 年代後半に、従来のマイクロペイメントに対する取引手数料の影響を軽減する方法として導入されました。Pass と Shelat [50] は、MICROPAY1 のこのアイデアをブロックチェーン ベースの支払いシステムに拡張し、分散型システム上で同じメリットを提供しています。このクラスのマイクロペイメントの中心的な考え方は、支払いチャネルの考え方と似ています。すなわち、多数の取引にわたる取引手数料の費用を償却するというものです。ただし、ブロックチェーンを利用した確率的マイクロペイメントのコア メカニズムは HTLC ではなく、ロータリー (抽選) ベースの支払いです。そのようなシステムでは、$\$X$ の支払いは、$C * \$X$ の価値があり、当選確率は $\frac{1}{C}$ の「抽選チケット」として実際に送信されます。そのため、このチケットの期待価値は $C * \$X * \frac{1}{C} = \$X$ です。

このスキームは、一般的に次のように説明できます。

A は B に支払いをしたい
A は、ビットコインの、新しく生成された鍵のエスクロー アドレス h_E に通貨を預け入れる
B は乱数 R_B を生成し、隠された署名付きコミットを A に送信する
B は受信者のアドレス h_B も A に送信する
A は乱数 R_A を生成し、支払い情報とともに、それに平文で署名し、B に送信する
$R_A \oplus R_B$ が 00 で終わり、R_B が隠された署名付きコミットと一致した場合、そのチケットは当選チケットであるため、エスクローが B に支払いを行う

このスキームは、設計上、理論的にはごくわずかなトランザクション料金でどのような規模にも対応できます (ほぼ完全にチェーン外であるため)。残念ながら、実際には、ほとんどの既存のスキームはプロトコルのどこかで集中型の仲介者に依存しているため、信頼不要ではありません。さらに、検閲抵抗については、支払いチャネルのサブセクションでのエクリプス攻撃と同じ問題が発生します。確率的マイクロペイメントと支払いチャネルの最大の違いは、確率的マイクロペイメントの O(1) 支払いルーティングの複雑さです。

F =全機能、P =一部機能、N =機能なし

スケーラブル	信頼不要	検閲不能	匿名性
F	P*	P*	N

**既存の実装の制限のため*

5.3 オーキッドのナノペイメント スキーム

オーキッドのナノペイメント スキームは、Pass と Shelat [50] が、*セクション 5.2.3* で簡単に説明している MICROPAY1 スキームの概念によって強く動機付けられています。私たちの支払いシステムの根本原理は、MICROPAY1 スキームから合理的な反復を試み、特に、無視できるほどのセキュリティ コストでシステムを経済的に拡張できるようにします。そのために、スケーラブルで信頼不要の、無検閲の匿名支払いシステムの要件を満たすことを目的としたプロトコルを作成しました。

これらの特性を念頭に置いて、オーキッドのナノペイメント スキームについて説明します。そのために、以下の定義を行います。

アクター

送信者: ナノペイメントの送信者。送信者は、イーサリアム アカウントを持ち、いくつかのイーサリアム ノードに接続して、ナノペイメント アカウントをセットアップし、自身のアカウントに資金を供給することが期待されています。送信者は、受信者のハッシュ コミットメントと宛先アカウントを含むメッセージを受信した後、チケット (以下で定義) を受信者に送信することにより、支払いを送信します。
受信者: ナノペイメントの受信者。受信者にはイーサリアム アカウントとイーサリアム ノードへのアクセスが必要です。受信者はハッシュ コミットメントを生成して、それを自身の宛先アカウント ID とともに送信者に送信し、送信者から 1 つ以上のチケットを受信します。受信者は、送信者が受け取った支払いパラメーターが正しいこと、および必要な資金が利用可能であることを確認する責任があります。
支払い/メンバーシップ スマート コントラクト: 当選チケットの支払いプロセスの決済に責任を負うスマート コントラクトは、送信者側のフロントランニング、グリーフィング、二重支出、その他の悪い行動に対する暗号経済的インセンティブも実行します。

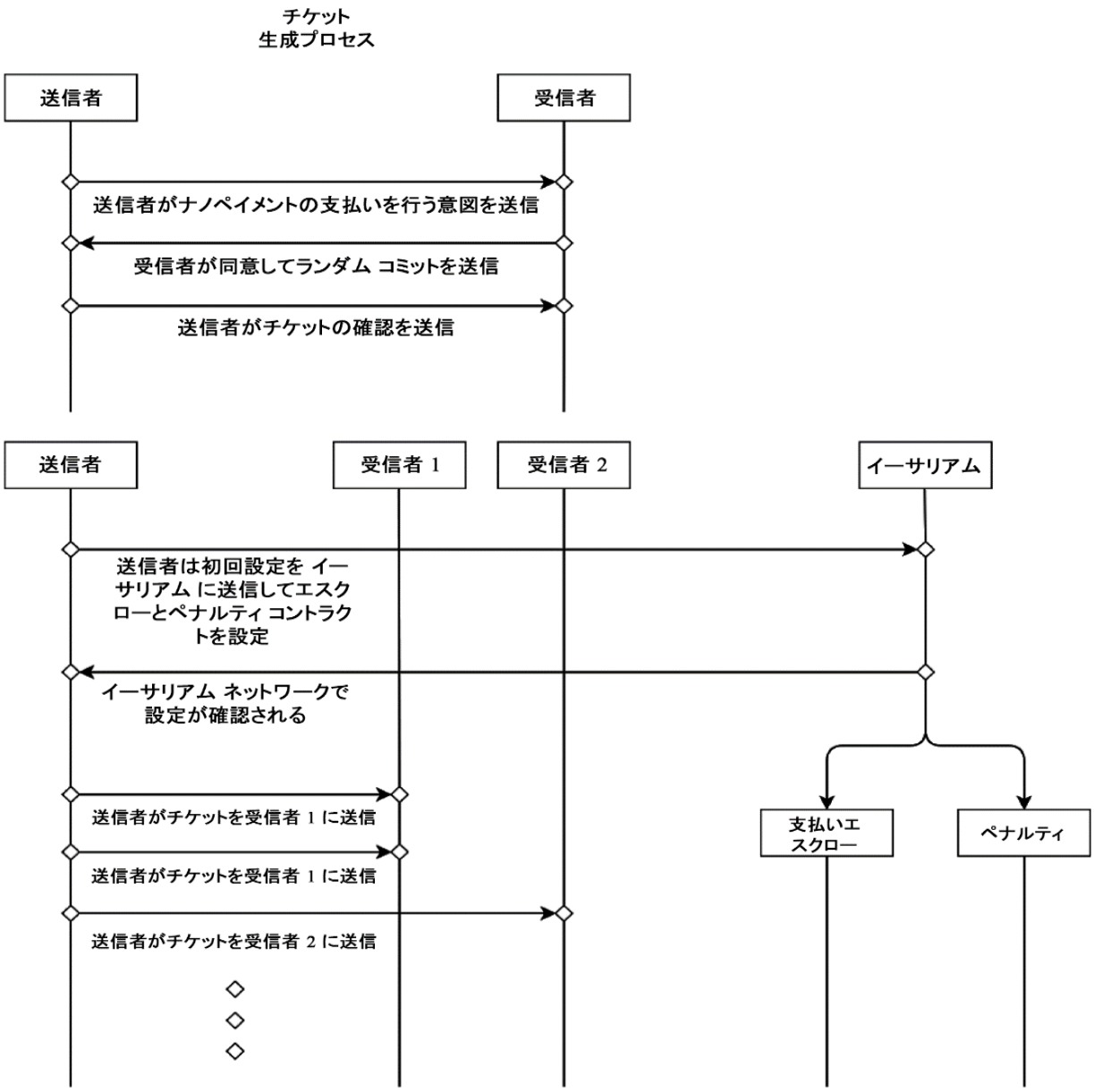
メッセージ:

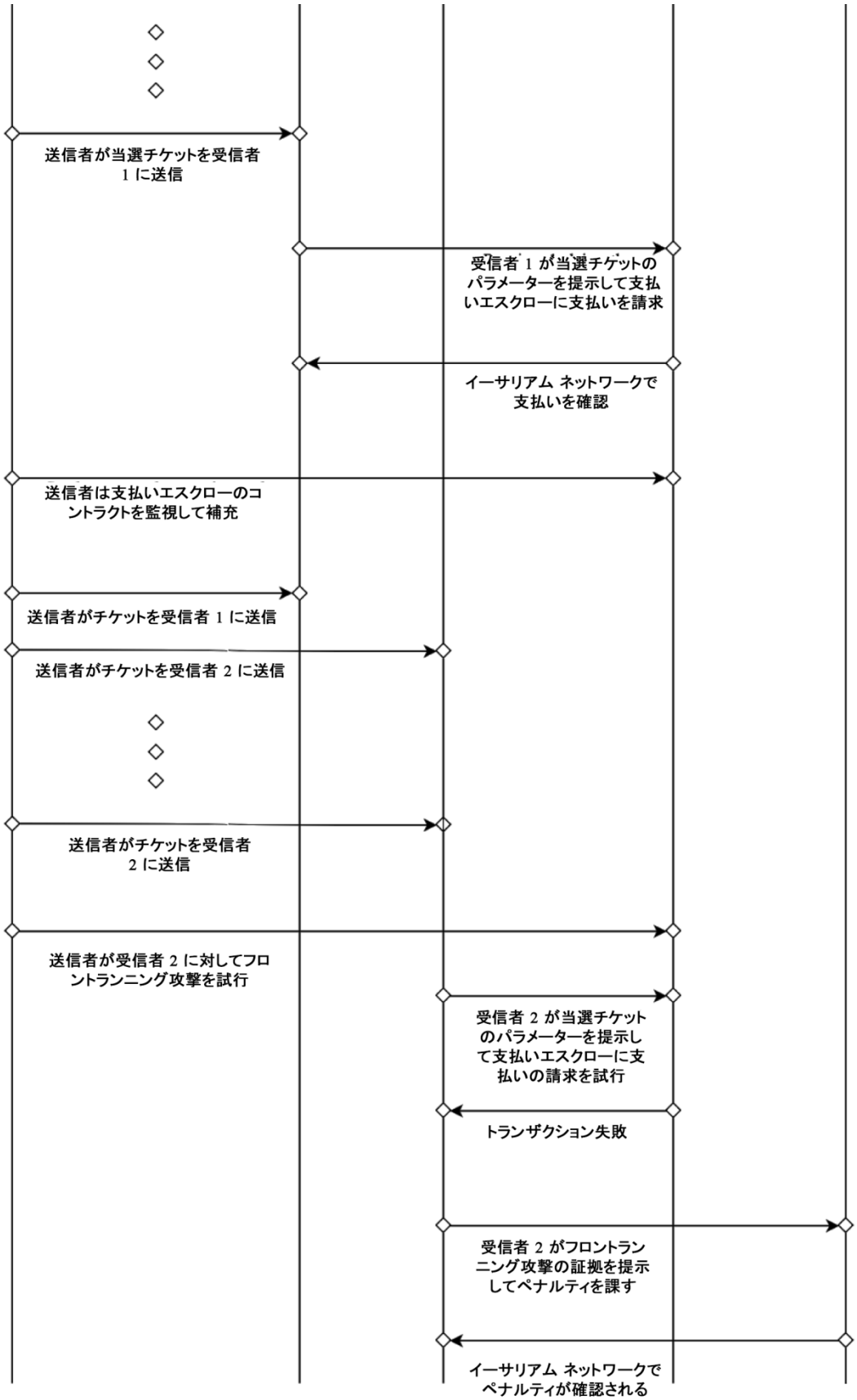
ランダム コミット: ランダムに生成された番号にコミットするために、チケットの受信者が最初に送信者に送信するコミット メッセージ。このコミットは、ハッシュ関数を介して乱数自体を隠します。
チケット: インタラクティブなチケット生成プロセスを完了するために送信者が受信者に送り返すメッセージ。これには、送信者の乱数と、完了したナノペイメントの主要なフィールドを確認する署名が含まれます。チケットの有効値は期待値であることに注意してください。真の償還値は、チケットが当選チケットである場合は送信者と受信者が合意した額面、または 0 です。チケットは、チケット生成プロセスが決済条件を満たす乱数を作成する場合にのみ当選チケットです。
当選チケット: 特定の額面で決済するための条件を満たした完了したナノペイメントで、特に当選条件を満たす乱数を含むもの。これは、決済を要求するために送信者の支払いエスクローからイーサリアム ネットワークにブロードキャストされるメッセージ、またはグリーフィングを証明するために使用されるメッセージです。

プロセス:

チケットの生成/送信: インタラクティブなプロセスを通じてチケットを「送信」 するプロセス。より正確には、チケットを生成するプロセスです。受信者は、まずランダム コミットを送信者に送信して、乱数生成プロセスを開始します。その後、送信者は受信者の残りの情報を含むチケットを送り返して、送信者の乱数を含むチケットを生成します。
決済/償還: 当選チケットの引き換え、決済、換金のプロセスです。当選チケットは、まず受信者が受信した情報に署名することで生成され、次にイーサリアム ネットワークにブロードキャストされます。その後、支払い契約により、支払い残高から受信者のアドレスに資金が支払われます。

以下のプログラム フローで、オーキッドのナノペイメント スキームを使用して、支払者と受信者の間で支払いがどのように行われるかを説明します。


チケット
生成プロセス
送信者
受信者
送信者がナノペイメントの支払いを行う意図を送信
受信者が同意してランダム コミットを送信
送信者がチケットの確認を送信
送信者
受信者 1
受信者 2
イーサリアム
送信者は初回設定を イーサリアム に送信してエスクローとペナルティ コントラクトを設定
イーサリアム ネットワークで設定が確認される
送信者がチケットを受信者 1 に送信
送信者がチケットを受信者 1 に送信
送信者がチケットを受信者 2 に送信
支払いエスクロー
ペナルティ


送信者が当選チケットを受信者
1 に送信
受信者 1 が当選チケットの
パラメーターを提示して支払
いエスクローに支払いを請求
イーサリアム ネットワークで
支払いを確認
送信者は支払いエスクローのコ
ントラクトを監視して補充
送信者がチケットを受信者 1 に送信
送信者がチケットを受信者 2 に送信
送信者がチケットを受信者
2 に送信
送信者が受信者 2 に対してフロ
ントランニング攻撃を試行
受信者 2 が当選チケット
のパラメーターを提示し
て支払いエスクローに支
払いの請求を試行
トランザクション失敗
受信者 2 がフロントラン
ニング攻撃の証拠を提示
してペナルティを課す
イーサリアム ネットワークで
ペナルティが確認される

このプログラム フローには、3 つの重要なポイントがあります。第一に、支払者にはワンタイム設定しかないため、他の既存のソリューションに比べて設定にかかるコストを非常に低く抑えることができます。これにより、二重支出やフロントランニングといった潜在的な問題が引き起こされますが、このような問題が起こる可能性は数学的および実験的に極めて低いことをペーパーの後半で説明します。第二に、各受信者は同じ支払いエスクローやメンバーシップ コントラクトとやり取りすることにより、個々の送信者と受信者のペアの設定コストを簡単に抑えることができます。さらに、さまざまな受信者の支払いに使用される資金をロックしたり、受信者間で分割したりする必要がなくなります。これにより、支払いチャネルを担保するために流動性のないエスクローにロックアップされる資金を減らすことができます。これは、ネットワークで頻繁に見られる統計的多重化現象が原因で発生します。最後に、すべてのナノペイメントはオフチェーンで発生するため、効率保証を維持しながら、信頼をオンチェーンでデリゲートして決済に対処できます。これにより、以前の確率的マイクロペイメントの方法で弱点だったサードパーティへの依存を最終的に排除することができます。

以下の表では、オーキッド ナノペイメント スキームと既存のマイクロペイメント スキームの機能が比較されています。これらの主張が正当であることを、次のセクションと付録でさらに説明します。

F =全機能、P =一部機能、N =機能なし

支払いソリューション	スケーラブル	信頼不要	検閲不能	匿名性
中央集中型	[N、P、F]	N	N	N
支払いチャネル	F	P	P	N**
確率的マイクロペイメント	F	P*	P*	N**
オーキッド ナノペイメント	F	F	F	N**

**既存の実装の制限のため*

*** ミキシング、ワンタイム アドレスなどで対処できます。詳細については、匿名性に関するセクション 5.8 で説明します*

$n =$ *L2 ネットワークのノード*

$C =$ *ノードあたりの平均接続数*

支払いソリューション	ルーティングの複雑さ	ネットワーク設定の複雑さ	資金配分係数*
中央集中型	該当しない	該当しない	該当しない
支払いチャネル	$log_C(n)$	C	$\frac{1}{C}$
確率的マイクロペイメント	1	C	$\frac{1}{C}$
オーキッド ナノペイメント	1	1	1

** 各ピアが取引できる総資金の割合を示します。一般的に、割合が低いと、全体的にネットワーク スループットが低下します。*

5.3.1. マイクロペイとの違い

オーキッドのナノペイメント プロトコルの一般的なスキームはマイクロペイ [40] に似ていますが、オーキッド スキームでは、効率性に関して確実な利益を得るために、基礎となる仮定にいくつかの変更が加えられています。さらに、これらの仮定により、元のスキームの理念の背後にある理論的なスケーラビリティと検閲耐性を維持する実装を導入できます。

オーキッド ナノペイメント スキームでは、次の仮定が変更されます。

1. 変更前: 各支払いエスクローは、二重支出を避けるために 1 人の受信者のみが使用できます
    a. 変更後: 各支払いエスクローは、当選チケットを引き換えるために複数の受取人が使用できます
2. 追加: 2 人の異なる受信者による資金の枯渇を軽減する方法が必要です
3. 変更前: ビットコイン スクリプトを使用します
    a. 変更後: イーサリアム スマート コントラクト、およびそれらをサポートする、基礎となる暗号関数を使用します
4. 変更前: 支払いエスクローに対処するために、相互に信頼できるサードパーティを使用します
    a. 変更後: 支払いエスクローに対処するために、イーサリアムベースのスマート コントラクトを使用します

セクション 5.10 で、これらの変更がセキュリティ、二重支出、フロントランニングなどにどのように影響するかについて説明します。

5.4 オーキッド トークン (OXT)

オーキッド トークン (OXT) は、ERC20 に準拠した新しいトークンで、10 億単位の固定供給と、ETH のように小数点以下 18 桁までの標準の準分割可能性を備えています。インフレーションはありません。二重支払いを防止するためにナノペイメント アカウントで使用されるような、通貨を「燃やす」契約上のペナルティ メカニズムが導入される可能性 (セクション 5.10) によって、若干の追加的デフレ圧力が生じます。

オーキッド マーケットの通貨として新しいカスタム トークンを使用すると、ETH などの一般的な通貨を使用した場合には得ることのできない経済的インセンティブのメリットが得られます。より具体的には、大規模なプロバイダーが市場に特化した大量のカスタム ユーティリティ通貨をステーキングすることが要求されると、プロバイダーの行動がカスタム マーケット トークンの価格、ひいてはそのステーク ポジションの価値に大きく影響するため、一般的な通貨を使用するよりも強力なインセンティブ調整の効果を生み出します。代わりに ETH などの一般的な通貨を使用すると、オーキッド マーケットの健全性によって ETH の価格に与える影響が予想よりもはるかに少なくなるため、この相互関係は非常に弱くなります。

5.5. オーキッドのガス コスト

キー チケットの引き換え機能に関する現在のオープン ソースの堅牢な実装では、当選チケットで呼び出されるときに約 10 万ガスが使用されます。これには、基礎となる ERC20 転送のコストが含まれます (機能は当選チケットでのみ呼び出されます)。

5.6 検閲耐性

オーキッドの支払いプロトコルでは、イーサリアムの検閲耐性が継承されています。これは、他のブロックチェーン暗号通貨プロトコルと同様です。ナノペイメント プロトコルには、当選していないチケットに関する通常運用中の送信者と受信者間の直接通信のみが含まれます。受信者がイーサリアム ブロックチェーンにトランザクションを送信する必要があるのは当選チケットの場合のみであるため、オーキッド ナノペイメントには通常のイーサリアム トランザクションと同じ検閲耐性があります。

オーキッド固有のすべてのイーサリアム トランザクション (または、特定の受信者に関するすべてのオーキッド償還トランザクション) を検閲するには、大多数のマイナーがこれらのオーキッド トランザクションを含むすべての当選ブロックを無視することに同意する必要があります。このシナリオが実現する可能性は極めて低いと考えています。それは、利益喪失のリスクやコストが高く、イーサリアムのマイニング コミュニティが分散化された性質を持つためです。イーサリアム ノードの一部が当選ブロックにオーキッド トランザクションを含めることを拒否した場合、限定的かつ部分的な検閲形式を達成できますが、1 / (1-X) に比例して取引手数料が増加するだけです (X は検閲グループの相対的なハッシュパワーを示します)。

セクション5.2.3 の支払いチャネルのように、特に支払者や受信者がフル ノードを実行しており、ネットワークにトランザクションを送信するためにそのフル ノードへの信頼に依存している場合、エクリプス攻撃により害が生じる可能性があることに注意してください。ただし、オーキッドのナノペイメントの場合、支払者と受信者はオーキッドのナノペイメント ネットワークに参加するためにフル ノードを実行する必要はありません。さらに、ノードを実行しているすべてのパーティは、信頼しているピアや有名なパブリック ピアにトランザクションを送信して、トランザクションが検閲されないようにできます。これはオーキッドのスキームの重要な利点の 1 つであり、既存の L2 支払いチャネル スキーム全体に関連のある実装です。

5.8 匿名性

オーキッドのナノペイメントは疑似匿名のみです。当選チケットの引き換え中に、受信者は通常プライベートなオフラインのクライアントサーバー支払い情報をチェーンに投稿することで、永続的なパブリック レコードを作成できます。紛失したチケットは投稿されないため、支払い情報は受信者にのみ公開されます。これにより、支払い情報の漏えいリスクは低減しますが、数週間から数か月の使用後も当選チケットは以前として蓄積されます。また、ユーザーの公開アカウント鍵と、支払い先であるオーキッド プロバイダーの情報の一部をリンクする公開情報トレイルも残ります。支払いチケットからクライアントが接続された特定のサーバーがわかるわけではなく、プロバイダーの公開鍵が明らかになるだけですが、巧妙な攻撃者はユーザーになりすまして、サーバーの公開鍵と物理アドレスのモデルを構築する可能性があります。

ほとんどのユーザーにとってこのわずかな情報漏えいは深刻な問題になりませんが、支払いのプライバシーを強化したいユーザーは、ナノペイメント アカウントに資金を提供する前に、イーサリアム アカウントと現実世界の ID のリンクを解除するための適切な措置を講じることができます (ミキシング サービスの使用、匿名暗号通貨への変換など)。マルチホップ ルートの場合、オーキッドのクライアントは、回路内のノードごとに個別のナノペイメント アカウントと公開鍵を使用して、オンチェーンの支払い履歴からのルート推論を防ぎます (複数の資金調達アカウントの事前の適切な解約を想定しています)。

5.9 スケーラビリティの分析

オーキッドのナノペイメント システムは、既存のレイヤー 1 ブロックチェーン支払いシステムよりも桁違いに高いトランザクション スループットを提供できるレイヤー 2 スケーリング ソリューションですが、最終的に実行可能な最大トランザクション スループットは、基礎となるレイヤー 1 のスループットの乗数になります。ナノペイメント システムには、オンチェーン トランザクションの 3 つの主要なソースがあります。

1. ユーザーがナノペイメント アカウントに入金する、またはナノペイメント アカウントから出金する
2. 販売者がステーク レジストリ アカウントに入金するまたはレジストリ アカウントから出金する
3. 販売者による当選チケットの引き換え

最初に取引手数料の観点からスケーラビリティを評価し、次にイーサリアムの基本的なトランザクション スループット制限の観点からスケーラビリティを評価します。

ガス コストが 2 万ガスまでの標準トランザクションの場合、一般的に、イーサリアム トランザクションの平均手数料は約 0.05 ドルです [51]。一般的な VPN ユーザーは、6 か月から 1 年以上前払いします。そのため、ほとんどのオーキッド ユーザーは、通常、ナノペイメント アカウントに 10 ドルから 50 ドル程度を入金することで「前払い」し、帯域幅購入のために数か月分の資金を提供すると想定しています。したがって、ユーザーの入出金の取引手数料は、高いガス コストを想定しても、わずかな間接費にすぎません。帯域幅の売り手のステークの入出金手数料はさらに少なくなります。一般的な売り手が少なくとも数千のクライアントを抱え、月間収益が 1000 ドルを超え、月に一度だけステークを追加または削除する場合、取引手数料の間接費は 0.1% 未満になります。

チケット償還トランザクションの間接費は変動します。ナノペイメントの期待値は当選確率に額面金額を掛けたもので、分散と取引手数料のトレードオフを柔軟に行うことができます。低い当選確率と高い額面金額に設定すると、分散の増加という犠牲が伴いますが、単位時間あたりの当選チケットの予想数が減少するため、取引手数料のコストが低減します。逆に、額面金額の低いチケットの当選確率を高く設定すると、当選者の増加、償還、取引手数料を犠牲にして分散が低減します。

現在のオーキッド スマート コントラクトの支払い償還機能では、最大で約 10 万ガスが使用されます。現在の価格を反映すると、約 0.02 ～ 0.20 ドルの取引手数料に換算されます。5% の取引手数料の間接費が合理的であると仮定すると、チケットの額面金額は 4 ドルになります。4 か月分の帯域幅使用量としてナノペイメント アカウントに 40 ドルを入金するユーザーには、4 か月間で平均 10 枚の当選チケットが発行されます。

二項分布を使用して、その残高の枯渇をモデル化することができます。チケットは、毎秒約 1 の償却レートで発行されるか (この使用量パターンは分析の重要な機能ではありませんが、説明のために使用されます)、10^{-6} の当選確率で 4 か月あたり約 1,000 万発行されると仮定します。10 名の当選者がいるプールでは、アカウントが 2 か月以内、つまり予想の 2 倍以上の速さで枯渇する可能性が最大で 1.8% あります。逆に、アカウントが 8 か月以上維持される可能性は最大で 0.6% しかありません。

この例の取引手数料を最小化するには、当選確率を 10 分の 1 に引き下げ、額面金額が 40 ドルのチケットを使用します。そうすると、4 か月あたり 1 枚の当選チケットのみが発行されると予想できます。これにより、取引手数料の間接費は 0.4% に低減します。ただし、これらの設定を使用すると、枯渇のリスクが非常に大きくなります。2 か月以内にアカウントが枯渇する可能性は、最大で 30% になります。

イーサリアム ブロックチェーンのトランザクション スループットは、トランザクション ガス コスト (トランザクションのコンパイル済み EVM コードの固定プロパティ)、およびブロック ガス制限とブロック生成率に依存します。どちらも時間の経過とともに変化します。チケット請求機能では約 10 万ガスが使用されます。現在、イーサリアムではブロックあたり 1,000 万ガスがサポートされており [52]、13 秒ごとに 1 ブロックの割合で生成されます [53]。そのため、10 万ガスのトランザクションのスループットは約 7 tps (または月間 1,800 万のトランザクション) になります。イーサリアムがオーキッド トランザクションのみに使用されたと仮定して、先ほどの例のように 1 ユーザーごとに 1 か月あたり約 2.5 枚の当選チケットが発行されるとすると、最大ユーザー数は約 700 万人に制限されます。

オーキッドのナノペイメント システムを数千万人以上のユーザーが使用するには、シャーディングを導入したイーサリアム 2.0 や、より高いベース スループットを備えた新しいレイヤー 1 ソリューションへの移行など、基礎となるレイヤー 1 ブロックチェーンにおけるスケーリング改善の展開と利用が必要になります。

5.10 グリーフィングを避けるための暗号経済学的手法

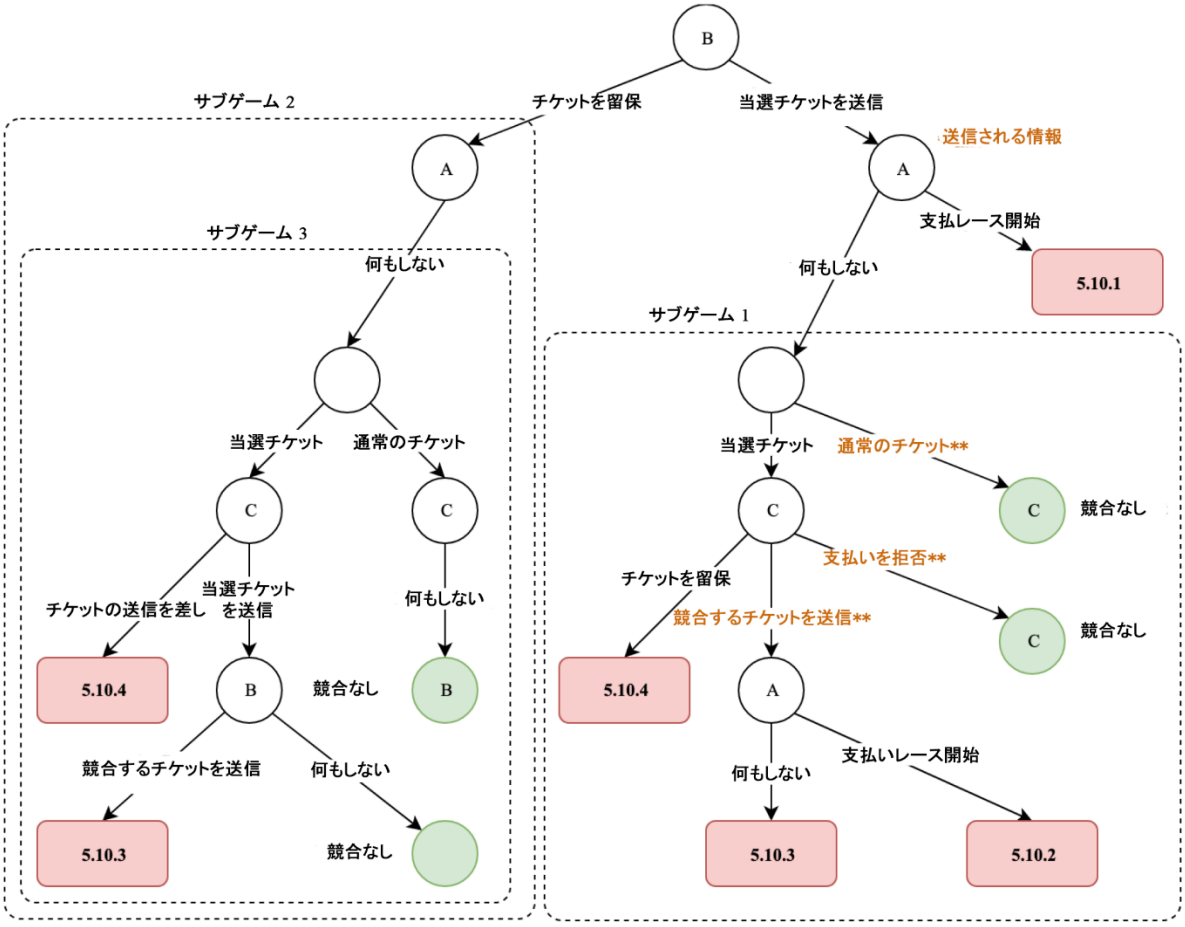
セクション 5.3.1 で述べたように、既存の確率的マイクロペイメント スキームとオーキッド ナノペイメントの主な違いの 1 つは、悪意のある攻撃を防ぐための暗号経済学的インセンティブを導入する必要があるという点です。これは、特定の支払者 *A* からナノペイメントを受信しているすべての受信者が、同じペイメント スマート コントラクト (*A* に属する) から当選チケットを引き換えているためです。効率に与える影響については、*セクション5.3* で説明します。

この設計がもたらす最大の問題は、1 つのアイデアを中心にして生じています。受信者は支払者から当選チケットを受信しますが、支払いエスクローの資金が不足しているため決済できません。これは、送信されるすべての当選チケットをカバーするための十分な残高が支払いエスクローにない場合に明示的に発生します。以下の展開型ゲーム ツリーで、さまざまな攻撃事例がどのように発生する可能性があるかを概説します。次の仮定を設定します。

1. 展開型ゲーム ツリーを使用する目的は、最適な戦略を見つけることではなく、悪い戦略を避けることです
2. ネットワーク ガバナンスの観点からすると、「悪い」戦略とは、正当な当選チケットが全額支払われなくなる戦略のことです

3. したがって、正当な支払いにつながるすべての戦略を概説する代わりに、悪い戦略につながる可能性のある行動に焦点を当てます
4. 行為者が悪意を持っているか、無害であるか、プロトコルに準拠しているか、プロトコルから逸脱しているかについての仮定は提供しません
5. 攻撃にかかるコストが予想される純利益を上回る場合、合理的な行為者は攻撃を選択しないという最小限の仮定を設定します

目標は、これらの仮定 (およびその欠如) を適切に考慮して、潜在的な悪い戦略を見つけ、それらの悪い戦略を回避するためのインセンティブ モデルを導入することです。行為者は当選チケットを支払うのに十分な残高が支払いエスクローにある場合にのみ行動すると仮定します。これが真実でなければ、支払者と受信者両方のインセンティブとディスインセンティブは崩壊し、支払いシステムは機能しなくなります。支払いシステムが機能しなければ、誰にとってもシステムを使用する理由がなくなるからです。展開型ゲーム ツリーの簡易バージョンの概要を以下に示します。ツリーは単純化されていますが、このツリーの他のブランチについては、*セクション 5.10.5* の図に記載されている 4 つの障害事例に集約して示しています。



A = 支払者　　*B、C = 受信者*

ご覧のとおり、オーキッドのナノペイメント スキームの参加者には、悪い戦略への関与につながる可能性のあるステップがたくさんあります。以下でそれぞれのステップを分析し、各サブケースのローカルインセンティブによって、このような悪い戦略の顕在化を防ぐ方法を示します。

B が当選チケットを送信した後の 「送信される情報」 とは、B が当選チケットを獲得したことをネットワークに伝播させるものであることに注意してください。この知識の受信は、それ自体で何かが決定されるわけではありませんが、多くの決定がこの知識の存在によって左右されます。特に、*サブゲーム 1* の上部にあるランダム ノードは、C に送信される支払いを生成しています。B の当選チケットについて知っている誠実なノード C は、A からのすべての支払いをすぐに拒否します。オーキッドが提供している実装では、A の支払いに対して請求があることを知っている場合、その後のすべてのパケットは拒否される必要があります。これは私たちが提供する実装ですが、無害な行為者 (または悪意のある行為者) がこれに従わない場合、または B の当選チケットに関する知識をまだ受信していない場合でも、以下の脆弱性分析が、悪い戦略を取らないよう動機づける暗号経済学的インセンティブになることがわかっています。

5.10.1 支払者が単一エンティティの場合のフロントランニング攻撃

この攻撃は一般的にフロントランニングとして知られています。これは、支払者が決済を回避しようとして B が決済する前に支払いエスクローに当選チケットを送信しようと試み、B への支払いを回避する場合に発生します。この攻撃を抑止するカギは、フロントランニングを試みることで受けるペナルティが、フロントランニングから受ける利益を上回ることです。以下に定義をリストします。

B_{Escrow} = *支払いエスクロー残高*
$B_{Membership}$ = *メンバーシップ残高*
V_{Ticket} = *チケットの額面金額*
r_{win} = *チケットの当選確率*
V_{txn} = *トランザクションのコスト*
$V_{Ticket} - V_{txn}$ = *チケットの決済額*

フロントランニング攻撃の場合、私たちの目標は、支払者がこの攻撃を合理的に選択しないよう、攻撃しようとする意欲を十分にくじくことで、悪い戦略の存在を減少させることです。具体的には、支払者がこの攻撃を行うのにかかるコストが、単にチケット代を支払うのにかかるコスト*よりも高く*なる必要があります。言い換えると、支払者がこの攻撃を実行することで得られる有用性が、単にチケット代を支払うのにかかるコスト*よりも低く*なる必要があります。

$$V_{Ticket} - V_{txn} < B_{Membership} - V_{txn}$$

オンチェーンの特定と検証を容易にに行える上記の不等式が成り立つ限り、メンバーシップ デポジットを削減することで、悪い戦略を実行する方が、単に受信者に支払うよりも高くつくようにすれば、合理的な支払者がこの 2 番目のケースを選択しないよう動機づけることができます。

5.10.2 支払者が複数エンティティの場合のフロントランニング攻撃

複数の受信者が立て続けに当選チケットを受信した場合、支払いエスクローを要求している者が複数存在することを誰も知らないうちに、一部の受信者が決済を開始する可能性があります。この場合、支払者は上記の不等式を回避できるフロントランニング攻撃を実行する可能性があります。立て続けに送信される n 当選チケットがある場合、複数エンティティのフロントランニングを防ぐための不等式は次のようになります。

$$Payout = n * (V_{Ticket} - V_{txn}) < B_{Membership} - V_{txn}$$

残念ながら、この不等式を維持するには 2 つの問題があります。第一に、n が無制限である場合、ペナルティ残高としてロックする必要があるトークン数は、受信者の数と支払い額に比例して増加するため、オーキッド ナノペイメント スキームは他の支払い方法と比べて効率的な資金配分ができなくなります。第二に、*セクション 5.10.3* で説明するように、完全に無害な当選チケットのコリジョンによって引き起こされる潜在的損害が増加します。したがって、システム設計の仮定に違反するか、良い行動を阻害するインセンティブを導入しない限り、論理的な方法での前進はないように思われます。幸いなことに、このジレンマは、わずかに強力な仮定を導入すれば解決できます。

私たちは、悪い戦略を実行することで得られるユーティリティの期待値が他の戦略によって大きく影響される場合、合理的な行為者は悪い戦略を選択しないと仮定します。この仮定により、システムが負う可能性のあるリスクを制限できます。また、フロントランニング攻撃の予想コストを低く抑えてロックされる資金を最小限にすることができます。次に、この限度を利用することで、$B_{Membership}$ により適切な限度を導入できます。そのために、追加の定義と次の仮定を導入します。

Δ = A が当選チケットの決済を送信してから、B がそれに気付くまでにかかる平均時間
r_{OXT} *= 支払者が受信者に送信する 1 秒あたりの OXT の償却レート*
VΔ = 支払者と受信者の間で時間Δをかけて転送された OXT の値
NΔ = 支払者と受信者の間で時間Δをかけて送信されたチケットの数
$r_{チケット}$ *= E (1 秒あたりの当選チケットの数)*

定義から以下の式を導きます。

$$V_{\Delta} = r_{OXT} * \Delta$$

$$N_{\Delta} = \frac{V_{\Delta}}{チケットの期待価格} = \frac{V_{\Delta}}{V_{Ticket} * r_{win}}$$

$$r_{Ticket} = \frac{r_{OXT}}{V_{Ticket}}$$

支払いピアの合計数が n の場合、当選チケットがコリジョンする確率は、
$W =$ *枚の当選チケットを発見*、次の式のようになります。

$$P(c\ コリジョン|W) = C_c^{n-1} P(\ 特定の受信者のコリジョン|W)^c P(特定の受信者のコリジョンなし|W)^{n-c-1}$$

$$P(特定の受信者のコリジョン|W) = P(特定の受信者のコリジョンなし)^{N_{\Delta}} = (1 - r_{win})^{N_{\Delta}}$$

つまり、当選確率 r_{win} が低下するにつれて、コリジョンの確率も低下します。したがって、コリジョンを防ぐために支払いハイパーパラメーターを選択する直感的なアプローチは、単に当選確率を下げるこ

とです。同様に、このアプローチを使用してメンバーシップの残高をどのように制限できるかも見てみましょう。

$$P(\textit{特定の受信者のコリジョン}|W) = 1 - P(\textit{特定の受信者のコリジョンなし}|W) \approx 1 - e^{(\frac{-r_{OXT} * \Delta}{V_{Ticket}})} \; if \; r_{win} \ll 1$$

$$P(c \textit{ コリジョン}|W) \approx C_c^{n-1}(1 - e^{(\frac{-r_{OXT} * \Delta}{V_{Ticket}})})^c (e^{(\frac{-r_{OXT} * \Delta}{V_{Ticket}})})^{n-c-1}$$

今回はコリジョンの確率がわかっているため、フロントランニング攻撃により予想される損失を制限できます。

$$E(\textit{支払い}) \approx V_{Ticket} + \sum_{i=1}^{n-1} \; (P(i \textit{ コリジョン}|W) * i * V_{Ticket})$$

$$E(\textit{支払い}) \approx V_{Ticket} + \sum_{i=1}^{n-1} \; (C_i^{n-1}(1 - e^{(\frac{-r_{OXT} * \Delta}{V_{Ticket}})})^i (e^{(\frac{-r_{OXT} * \Delta}{V_{Ticket}})})^{n-i-1} * i * V_{Ticket})$$

$E(\textit{支払い}) < B_{Membership}$ が成り立つ限り、フロントランニング攻撃を試みることは概して得策とはなりません。$E(payout)$ を最小化し、それによって、悪い戦略に対する暗号経済学的ディスインセンティブを維持しながら、ロックしておく必要のある資金の量を減らすために、上記のハイパーパラメーター戦略をここでも適用します。私たちがすべきことは単純に大きな V_{Ticket} と、それに対応して低い r_{win} を選択することです。したがって、上記の支払いハイパーパラメーターの選択に対する直感的なアプローチでは、単にチケットの当選確率を下げることにより、実質的に、メンバーシップ残高にロックされる資金 (受信者の数に対して実質的に一定となるように下げることが可能) に証明可能な制限を設定することができます。

ただし、このモデルは、受信者の全員とは言わないまでも、その多くを支払者が効果的に監視できるよう考慮しているわけではないことに注意してください。それが可能であれば、支払者は利益が生じない状況に対してフロントランニング攻撃の試行を回避することができます。これに対する防御策として、上記のハイパーパラメーター戦略は、これらのケースが起こる確率をきわめて低くすることができます。コリジョンの確率が非常に低くなると、コリジョンの予想コストもほぼ無くなるため無視できるほどになります。

以下に、悪い支払いと良い支払いのハイパーパラメーターに関するいくつかの経験的選択と、結果として生じるコリジョンの確率を示します。単一のコリジョンの存在についても、コリジョンの確率を測定していることに注意してください。このセクションの分析の主な目的は、フロントランニング攻撃から受信者を保護することです。つまり、このセクションのコンテキスト内では、安全なパラメーターに適合する支払いのみを受け入れるかどうかは受信者の手に委ねられています。

パラメーター	Δ	r_{OXT}	r_{win}	V_{Ticket}	コリジョンの確率 $n = 2$	コリジョンの確率 $n = 10$	コリジョンの確率 $n = 100$
悪い戦略	300 秒	$3 * 10^{-6} \frac{OXT}{s}$	10^{-2}	$0.12 \; OXT$	$\sim 0.747\%$	$\sim 6.527\%$	$\sim 52.41\%$

許容範囲内の戦略	30 秒	$3 * 10^{-6} \frac{OXT}{s}$	10^{-3}	1.2 OXT	~ 0.007	~ 0.067	~ 0.740%
良い戦略	3 秒	$3 * 10^{-6} \frac{OXT}{s}$	10^{-4}	12 OXT	~ 0.000	~ 0.000	~ 0.00742

5.10.3 複数エンティティの支払いレース

複数エンティティの支払いレースとは、支払者に悪意のない当選チケットのコリジョンのことです。これらの支払いレースは必然的に発生します。これらの支払いレースが到達する可能性のあるケースは 2 つあります。そのうちの 1 つは*サブゲーム 1*、もう 1 つは*サブゲーム 2* で概説されています。以下にケースの概要を示し、それらを防ぐ方法について説明します。

サブゲーム 1: 意図しない支払いレース

意図しない支払いレースが発生した場合には、*セクション 5.10.2* のコリジョン分析を使用して、意図しない支払いレースを最小限に抑えるハイパーパラメーターを選択できます。非同期設定では支払いレースを完全に防ぐことはできませんが、それらに関連するリスクについては次のとおりです。

$$P(\textit{秒あたりのコリジョン}|W) = r_{Ticket} * P(\textit{コリジョン}|\, W)$$

$$P(\textit{秒あたりのコリジョン}|W) = \frac{r_{OXT}}{V_{Ticket}}(1 - P(0\ \textit{コリジョン}|\, W))\ .$$

$$P(\textit{秒あたりのコリジョン}|W) = \frac{r_{OXT}}{V_{Ticket}}(1 - C_0^{n-1}(1 - e^{(\frac{-r_{OXT} * \Delta}{V_{Ticket}})})^0(e^{(\frac{-r_{OXT} * \Delta}{V_{XTicket}})})^{n-1})。$$

$$P(\textit{秒あたりのコリジョン}|W) = \frac{r_{OXT}}{V_{Ticket}}(1 - (e^{(\frac{-r_{OXT} * \Delta}{V_{Ticket}})})^{n-1})\ .$$

$$E(\textit{ペナルティ}|W)\ \textit{秒あたり} = P(\textit{秒あたりのコリジョン}|W) * B_{Membership}$$

$$E(\textit{ペナルティ}|W)\ \textit{秒あたり} = \frac{r_{OXT}}{V_{Ticket}}(1 - (e^{(\frac{-r_{OXT} * \Delta}{V_{Ticket}})})^{n-1}) * B_{Membership}$$

支払者が意図しない支払レースの損失で負うリスクを、いくつかの例を用いて以下に示します。*セクション5.10.2* の限度を使用して $B_{Membership}$ を計算します。

パラメーター	Δ	r_{OXT}	r_{win}	V_{Ticket}	1 秒あたりの支払いレース ペナルティ $n = 2$	1 秒あたりの支払いレース ペナルティ $n = 10$	1 秒あたりの支払いレース ペナルティ $n = 100$
悪い戦略	300 秒	$3 * 10^{-6} \frac{OXT}{s}$	10^{-2}	0.12 OXT	$2.258 * 10^{-8} \frac{OXT}{s}$	$2.089 * 10^{-7} \frac{OXT}{s}$	$2.735 * 10^{-6} \frac{OXT}{s}$
許容範囲内の戦略	30 秒	$3 * 10^{-6} \frac{OXT}{s}$	10^{-3}	1.2 OXT	$2.251 * 10^{-10} \frac{OXT}{s}$	$2.026 * 10^{-9} \frac{OXT}{s}$	$2.236 * 10^{-8} \frac{OXT}{s}$

良い戦略	3 秒	$3 * 10^{-6} \frac{OXT}{s}$	10^{-4}	12 OXT	$2.250 * 10^{-12} \frac{OXT}{s}$	$2.025 * 10^{-11} \frac{OXT}{s}$	$2.228 * 10^{-10} \frac{OXT}{s}$

前のセクションのコリジョン分析は、主に受信者をフロントランニングから保護するためのものでした。そのため、適切なハイパーパラメーター戦略の選択を促す、受信者側のインセンティブが生じる結果となりました。意図しない支払いレースでは、支払者は自分が制御できなかったレースのために誤って罰せられることに注意してください。上記の最悪のケースでは、ハイパーパラメーターの選択が不十分なために、最大で約 1% の料金が取られる可能性があります。良い戦略では、この料金は無視してもよいほどになります。したがって、意図しない支払いレースの存在により、優れたハイパーパラメーター戦略の選択を促す、支払者側のインセンティブも生まれることになります。

サブゲーム 2: 受信者による差し控え攻撃

ある受信者が当選チケットの送信を差し控え、別の受信者が当選チケットを送信した直後にブロードキャストした場合、受信者は支払者に対して強引に攻撃することができます。これは悪い戦略のきっかけとなり、支払者に損害が生じます。したがって、私たちの目標は、受信者がこのような攻撃を行わないように、差し控えを十分に抑止することです。*セクション 5.10.1* を思い出してください。単一エンティティのフロントランニング攻撃を阻止するためには、不変式、つまり $V_{Ticket} < B_{Membership}$ が成り立つ必要があります。したがって、受信者がグリーフィング (嫌がらせ) をしようと考える場合、最初の検査で発生する損害の量は損害を与えるコストよりも高くなります。1 つの解決策は、経過時間が長くなるにつれて燃焼量 (破棄される量) を減らすことです。ただし、これは支払者主導の攻撃ベクトルを防ぐ上述の暗号経済学的な不変条件に干渉し始めます。

したがって、次善の措置は、差し控えによって生じうる損害を減らすよう試みることです。上記の分析により、当選チケットがすでに見つかっていると仮定して、コリジョンの確率を計算できることに注意してください。言い換えると、上記の Δ の期間中のみ当選チケットを有効と見なす場合、時間の経過に伴う支払いレースのペナルティは *サブゲーム 1* のペナルティと同様になります (実証分析については上記を参照してください)。差し控え攻撃で予想される損害率が非常に小さくなるだけでなく、損害自体は Δ の期間中のみ有効となります。この有効期間は可変であり、実際には損害発生の可能性をさらに低くするための請求にかかると予想される時間よりも短くなる可能性があります。
したがって、適切なパラメーターが選択されると、差し控え攻撃による予想損失は無視できるほど小さくなります。このように予想損失が無視できるほど小さくなると、攻撃者の予想コストは被害者の予想損失と比較して非常に高くなります。その結果、合理的な攻撃者モデルにより、差し控え攻撃による悪い戦略は防止されます。*サブゲーム 1* で述べたように、これにより、適切なハイパーパラメーター戦略を選択する支払者側のインセンティブがさらに生み出されます。

5.10.4 差し控え

合理的に言って、これは、無害または誠実な受信者にとって有効な戦略ではないことに注意してください。受信者は、差し控えによってごくわずかの具体的な利益を受け取ると期待できますが、最悪の場合、当選チケットとそれに関連する支払いを放棄することになります。実際、差し控えについて気にかけるのは次の 2 つの場合、再帰的なサブゲームへの差し控えの降下、および差し控え攻撃の場合のみです。他のすべてのケースはネットワークに害を及ぼすことばなく、悪い戦略のきっかけとなるわけでもありません。実際、差し控えはそれを行う者に直接的な経済的損害を与えるだけです。したがって、5.10.3 のサブゲーム

2 で述べたように、差し控えで予想される損害を常に低減し、受信者ができるだけ早く当選チケットを決済するように動機づけるため、すべての当選チケットには有効期限を定める必要があります。

5.10.5 再帰的なサブゲーム

このセクションの展開型ゲームには、行為者の人数や発生する可能性のあるアクションの数に制限がありません。ただし、上記の各障害事例では、展開型ゲームの基本的な仮定 (単一の当選チケットをカバーするのに十分な資金が支払いエスクローにあること) を無効にすることで、ゲームのフレームワークが終了するか、再帰的にサブゲーム 1 または 2 につながります。このマッピングを以下に列挙します。

- 5.10.2 は、基本仮定の無効化という結果になります。ネットワークの残りの部分がまだその結論に到達していない場合、ノード 5.10.2 は、任意の受信者 B と C を持つサブゲーム 1 とサブゲーム 3 両方のエントリ ノードにつながります。
- 5.10.3 も基本仮定の無効化という結果になります。ネットワークの残りの部分がまだその結論に到達していない場合、ノード 5.10.3 は、任意の受信者 B と C を持つサブゲーム 1 とサブゲーム 3 両方でエントリ ノードにつながります。
- 5.10.4 はサブゲーム 2 のエントリ ノードにつながります。

サブゲームの障害事例はそれぞれ、既存のゲーム ツリーの事例に再帰する可能性があることに注意してください。無害なパス (緑のノード) は、(任意の行為者が新しいチケットを受信するだけで) サブゲームのいずれかに再帰的に降下することにより、さらなる潜在的な攻撃のきっかけとなる可能性があります。ただし、これらのサブゲームは結局のところ常に (現在は実行不可能な) 悪い戦略につながるか、または無害なパスで終了します。

5.10.6 要約

結論として、現在では、上記の暗号経済学的モデルに基づいて悪い戦略が実行されるのを包括的に防ぐ、一連の条件とローカル戦略があります。特に、攻撃者の餌食となる上記のすべてのケースには、合理的な攻撃者を想定した場合、前払いに合意して悪い戦略の実行可能性を防ぐことができる一連のハイパーパラメーターが存在します。実際、適切なハイパーパラメーターを選択すると、そのパラメーター セット内のすべての潜在的な攻撃は無視できるほどわずかな損害しかもたらさなくなるため、不合理な攻撃者でさえネットワークに対して合理的な攻撃を行うことはできません。利益主導または無害な行為者の場合はすべて、このようなインセンティブにより、行為者は、乱数性の悪影響を最小限に抑えるハイパーパラメーターに同意するようになります。無害な行為者の仮定であるか攻撃者の仮定であるかを問わず、各プレイヤーのローカル インセンティブは、悪い戦略が実現可能になることを必然的に回避します。

6. 攻撃と防御

このセクションでは、特定のユースケースを評価し、関連する攻撃者が採用する可能性のある主な攻撃を要約して、それらに対する防御の設計能力を分析します。

6.1 脅威モデル

攻撃者の主な目標は、いくつかの (包括的な) カテゴリに分けることができます。

- *トラフィックの確認*: 攻撃者は、ユーザー A が宛先 B と通信しているかどうかを確認しようとします。ここで、A は既知のユーザーで、B は既知の宛先エンティティ (ウェブサイトなど) です。

- *トラフィックの分析*: 攻撃者は、関連するメタデータとともに、宛先 B* と通信している一連のユーザー A* の全員または一部を知ろうとします。
- *トラフィックの遮断*: 攻撃者は、一連のユーザー A* と一連の宛先 B* の間の接続を遮断しようとします。
- *コンテンツの変更*: 攻撃者は、一連のユーザー A* と宛先 B* の間の通信ストリームのコンテンツを、あからさまに、またはひそかに変更しようとします。

下記のような権限の組み合わせにより、ローカルのアクティブな攻撃者は限定されると想定しています。
- *監視*: ネットワーク トラフィックの一部を受動的に監視します
- *潜入*: オーキッドやイーサリアム ノード、または外部サーバーの一部を制御します
- *操作*: ネットワーク トラフィックの一部を積極的に変更します
- *推論*: 収集されたデータに計算を適用して、*監視されていない*興味を引く情報を推測します

オーキッドは、すべてのトラフィックやノードを監視または変更できる、強力でグローバルな攻撃者から保護することはできません。私たちは、攻撃者の権限が事実上コストによって制限される経済的モデルを想定しており、そのほとんどはユーザーごとにスケーリングします。

トラフィック分析 (推論) 攻撃

匿名システム (特に Tor) に対する推論攻撃については広範な研究があり、いくつかの主要なカテゴリに分類できます。

*パッシブ フロー相関*では、攻撃者はネットワーク上の 1 つ以上のポイント (通常は出入りする場所) でトラフィックを監視します。その後、統計的推論を使用して、マルチホップ回線でストリームの相関分析を行います [54、55][56、57]。最近のディープ ラーニングの進歩により、これらの攻撃の費用対効果は向上しています [54]。

*アクティブ フロー相関*を使用すると、攻撃者はトラフィックを操作 (タイミング遅延の挿入など) して透かしパターンを作成し、精度と再現性を大幅に高めることもできます [58–60]。これらの攻撃では、トラフィックの透かしを挿入するために、ストリームの入口でハードウェアを制御する必要があります。

*サイド チャネル相関分析攻撃*は、低遅延リレーでも可能です。1 つのストリームのタイミング測定では、同じリレーを通過する監視されていないストリームの相関分析を行うのに十分な情報を明らかにすることができます [61][62]。これらの攻撃によって回路のノードが明らかになる可能性がありますが、一般的に完全な回路をユーザーの IP までたどるには不十分です。
*ウェブサイトのフィンガープリント攻撃*により、攻撃者は接続の出口ポイントだけを監視して、トラフィック パターンをウェブサイト固有のフィンガープリントの既知のライブラリと一致させることにより、回路を通るストリームの相関分析を行うことができます [63]、[64]。このようなフィンガープリントは、ディープ ラーニング技術によって自動的に生成できます [65–67]。ウェブサイトのフィンガープリント攻撃に、攻撃者が実際に使用するのに十分なほどの精度や再現性があるかどうにかは議論の余地があります [68]。

対象範囲

攻撃者の目標、能力、予算の可能性が広範囲に及ぶことを考慮すると、すべての、または広範囲に及ぶ攻撃者に対する一般的な防御は、オーキッドのような低遅延、高帯域幅のオーバーレイ ネットワークの範囲を超えています [26]。代わりに、最も一般的ないくつかの*経済的に関連のある*ユース ケースと、それらの暗黙の攻撃者のモデルに焦点を当てます。

6.2 地理的コンテンツ制限の回避

ウェブ コンテンツの地理的制限の回避は、今日の VPN の最も一般的な使用例の 1 つです[20]。Netflix などのストリーミング サービスは、IP アドレスからユーザーの場所を推測することで地理的なライセンス制限を実施した後、その特定の場所に合わせてカスタマイズされたライブラリへのコンテンツ アクセスを制限します。

この場合の攻撃者は、コンテンツの変更を目標としており、目的のウェブサイト自体を制御します。そのため、いくつかの興味深い課題があります。攻撃者にとって、最も一般的な VPN やプロキシ サービスを IP アドレスによって検出し、ウェブサイトへのアクセスを完全にブロックすることはかなり簡単です[21]。攻撃者は、基本的な形式のターゲット トラフィック分析により、IP 登録データベースを使用して既知の VPN 会社に関連付けられた IP アドレスの範囲を見つけることができます。または、同じ IP アドレスを共有する多数の異なるアカウントを探して、特定のアドレスがプロキシや VPN サーバーのアドレスである可能性が高いことを判断できます。

地理的コンテンツのロック回避に適切な、難読化された IP アドレスをクライアントに提供するために、現在の VPN が使用できる戦略は、いくつかあります。最も簡単ではあるものの最も高価な方法は、アドオン サービスとして個々のクライアントに固有の IP アドレスを提供することです。別の方法としては、IP アドレスを (サブリースなどで) 迅速に切り替えて、常にブロックされていない新しいアドレスをクライアントに提供することができます。

原理的には、オーキッド のメタデータ レジストリ (セクション 4.2) では、帯域幅販売者はカスタム タグ (「unique_ip」など) を使用して一意の IP アドレスを宣伝できます。クライアントは、このタグと位置情報を指定してフィルターをかけることにより、特定の場所で一意の IP アドレスを使用する出口ノードを見つけることができます。ここで障害となるのは、オーキッド マーケットでは高速かつステートレスで半匿名のトランザクションが前提となっているのに対して、一意の IP アドレスを使用するには設定にかなりのコストがかかることです。ユーザーが、新しい一意の IP アドレスを提供するノードに接続し、数秒後に接続を切断した場合、サービスに対して数マイクロドルを支払うことになり、提供コストがおよそ数百万倍になります。これに対して オーキッド の販売者は一意の IP アドレスを提供するサービスに、より高額のマクロペイメントを請求できます。その場合には、クライアント UI で請求額が大きくなることに対する明示的な承認をユーザーから得る必要があります。これは、キュレーションされた信頼性の高い販売者のみ実現可能だと予想しています。

別の方法として、販売者は特定のストリーミング サービスのブロックを解除できることを明示的に宣伝することもできます。このように主張した機能の実装は、販売者に委ねられます。新しい IP アドレスの

[20] https://www.geosurf.com/blog/vpn-usage-statistics/
[21] https://help.netflix.com/en/node/277

ローテーションとユーザー/IP アドレスの比率を低く保てば、ブロック解除を実現できます。成功した場合、販売者は、事前のマクロペイメントを必要とせず、このサービスの帯域について通常よりも高い金額を請求できます。

長期的に見ると、このようなユースケースを実現できることで大きなメリットがあります。ユーザーはさまざまなプロバイダーからサーバーにアクセスでき、現在の VPN モデルに伴うロックインのリスクを回避できるからです。単一の VPN サブスクリプション契約では、特定のプロバイダーのサーバーが突然ブロックされるとユーザーはリソースを使用できなくなります。オーキッド では、ユーザーは、いつでも簡単に、ほぼ即座にプロバイダーを切り替えることができます。

6.3 ピアツーピアの共有システム

ピアツーピア ネットワークは、中央集中型のコンテンツ ソースを使用せずに、ユーザー間で直接コンテンツを共有するのによく使用されている手段です。ISP (インターネット サービス プロバイダー) はさまざまな理由から、ピアツーピアの共有ネットワークに制限をかけたいと考えます。ピアツーピアの共有ネットワークが、ケーブル テレビやストリーミングの収益に対する脅威になる、帯域幅が大量に消費される可能性がある、保護されているコンテンツの共有が可能になる、などと考えられるためです。攻撃者の目標は、主に何らかの妨害ですが、これはトラフィック解析から始まります。攻撃者は特定の P2P ネットワークを使用している、または特定のコンテンツを共有しているユーザーを特定しようとします。

このユースケースでは、攻撃者にできることはかなり限られます。主な攻撃の戦略は、P2P のパケットを検出してシェーピングやフィルタリングを行うか、または、自らもノードを運用してピアツーピア ネットワークに入り込み、対象ユーザーの IP アドレス、アクション、メタデータのログを収集するなどです。Bittorrent のような、現在一般的なピアツーピア ネットワークの経済的セキュリティは低いため、ネットワークへの潜入にそれほどコストがかかりません。多くの司法管轄区では、VPN は、トラフィックの暗号化とユーザーの IP アドレスの秘匿によって、このようなユースケースに対する適切な保護を行っています。これは、法律上または金融上の義務からログを保管する必要がない限り有効な方法であり、攻撃者がログを取得することは困難になります。

オーキッドでは、ステークに応じた加重選択のしくみとホワイトリストを組み合わせることにより、このユースケースに対して、VPN と同様の有効な防御策を実現しています。ログが収集されないことが明らかな信頼できるプロバイダーのみが登録されたホワイトリストを使用するオーキッド クライアントは、ログが収集されないことが明らかな VPN のリストからランダムに VPN を選択した場合と同等かそれ以上の確率で、ノードと攻撃者の結託を回避できます。

攻撃者がこの攻撃に成功するのは、ユーザーの選択したオーキッド ノードと P2P ファイル共有ネットワーク (Bittorrent など) のノードの*両方*が攻撃者のコントロール下にある場合です。この状態が発生する確率は、次のようになります。

$$p(compromise(x,y): x \in A_o, y \in A_B) = p(x \in A_o)\, p(y \in A_B) \quad (20)$$

$$p(y \in A_O) = \frac{S_{A \cap W}}{S_W} \quad (21)$$

$$p(y \in A_B) = \frac{B_A}{B_T} \quad (22)$$

x、y: それぞれ、選択された オーキッド ノードとファイル共有ノード
A_o、A_B: それぞれ、攻撃者のコントロール下にある一連の オーキッド ノードおよびファイル共有ノード
W: クライアントのホワイトリスト、一連のオーキッド ノード
S, S_W:それぞれ、OXT ステークの合計と W のノードの OXT ステーク
$S_{A \cap W}$:$A \cap W$ のノードの OXT ステークの合計、W に存在する攻撃者のノード
B_A、B_T: それぞれ、ファイル共有ネットワークの攻撃者の帯域幅と、合計の帯域幅

ホワイトリスト W がない場合、S_W はシステムのステークの合計 S_T と等しくなり、攻撃者の オーキッド ノードを選択する確率は $\frac{S_A}{S_T}$ となります。これは、攻撃者のコントロール下にある OXT ステークの合計に対する割合です。オーキッド のユーザー数が数百万人で、オーキッド ステークの総額が 10 億ドル程度の場合を仮定すると (セクション 4.4)、オーキッド ノードの予算として 1 千万ドルを持つ攻撃者の成功率は、シングルホップの オーキッド ユーザーでは、ユーザーが保護されていない場合と比較して 3 桁低くなります。攻撃者がコントロールするファイル共有ノードに接続している オーキッド ユーザーが、攻撃者の オーキッド ノードの 1 つに同時に接続する確率はわずか 0.1% です。

この場合、$\frac{S_{A \cap W}}{S_W} = \frac{S_A}{S}$ となるため、ランダムなホワイトリストは効果がありません。慎重に選定を行ったホワイトリストでは、$S_{A \cap W}$ が S_W よりもはるかに減少し、侵害が生じる可能性を大幅に低下させることができます。

攻撃者に有効なトラフィック タイミング解析攻撃を実行する能力がないと仮定すると、マルチホップの回路では、選択の確率を大幅に低下させることができます。

$$p(compromise(X_k)) = \left(\frac{S_{A \cap W}}{S_W}\right)^{\lceil k/2 \rceil} \frac{B_A}{B_T} \qquad (23)$$

ここで X_k は、ホップ数が k の回路を表します。攻撃者が経路全体を推測するには、攻撃者がコントロールできるノードがこの回路に 1 つおきに配置されている必要があります。典型的な 3 ホップの回路の場合、攻撃者はそのうち、最初と最後の 2 つのノードをコントロールできる必要があります。上記と同じパラメーターでホワイトリストを使用しない場合、攻撃者のファイル共有ノードに接続しているユーザーが、侵害されている 3 ホップの回路を同時に使用する確率は、わずか 10^{-6} となります。

高度な攻撃者は、アクティブ フローの相関分析を行って、マルチホップ回路の有効性を低下させる可能性があります。一時的なフィンガープリント パターンをトラフィック ストリームに挿入し、エンドポイントでそれを検出することにより、理論上、攻撃者は最初の オーキッド の入口ノードとエンドポイント (この場合はファイル共有ノード) をコントロールするだけで、ホップ数の多い回路であっても相関分析を行い、侵害することが可能です [23-25]。オーキッド クライアントでは、帯域消費 (bandwidth burning) の選択的な使用を、このような攻撃への防御に役立てることができます。具体的には、パケット ストリームをダミーのデータ パケットで埋めることにより、継続的な変動の小さいフローを疑似的に再現して、一時的なシグナルを検出できないようにします。

ただし、このユースケースでは、このような高度なトラフィック解析攻撃が行われる可能性は低いと考えられます。この種の攻撃者は、ユーザーあたりの予算が非常に限られているためです。トラフィック解析の手法では、統計的な相関関係の痕跡が得られ、監視に役立ちますが、一般に誤検出の割合が非常に高くなります。

6.4 ISP による検閲の回避

現在、多くの国で政治的に好ましくないインターネット コンテンツの検閲が行われており [69]、この検閲は通常、当該地域の ISP (インターネット サービス プロバイダー) によって実施されます。検閲の範囲は国によって大きく異なりますが、大まかには、2 つの主要カテゴリーに分類することができます。1 つは、検閲は行うものの VPN の使用は容認する国々 (インドネシア、パキスタン、タイなど)、もう 1 つは、大規模な検閲を行い、VPN の使用を禁止または制限している規制の厳しい国々 (中国、ロシアなど) です。

軽度の検閲

VPN/プロキシ サービスが容認されている国では、インターネット検閲を回避するためにオーキッドを使用するのは簡単です。クライアントでシンプルな地域のフィルターを使用すれば、被制限国以外のノードを選択できます。ただし、実際にはこれも不要と考えられます。被制限国の出口ノードが大量のトラフィックを受信する可能性は低いため、出口ノードは検閲がほとんどない地域に固まっている傾向があるからです。検閲回避の防止に大きなリソースを投じないという意味で、このような国の攻撃者は「軽度」です。

重度の検閲

VPN/プロキシ サービスが積極的に制限されている国では、問題が大きくなります。特に中国では大規模なテクノロジー ソリューションを導入して、包括的なインターネットの監視/検閲を実施しており、これはグレート ファイアウォール (GFW) と呼ばれています。中国ではさらに、VPN の使用が見つかった個人に対して罰金を科し始めました [70]。それでもなお、中国では外部 VPN の人気は衰えず [71]、プロバイダーとのいたちごっことなっています。この攻撃者はさまざまな能力を持っていますが、中でも特に検閲回避に関連するものが 3 つあります。

- GFW では、*ディープ パケット インスペクション*を使用して、VPN/プロキシ サーバーの可能性のあるものを一斉検出します。
- GFW では*アクティブ プローブ*を使用して、疑わしいサーバーを検査します [72]
- GFW では、自動および手動のプロセスを使用して、VPN/プロキシ サービスと関連のある IP アドレスを禁止します

オーキッド クライアントでは、WebRTC を使用してトンネルを構築し、通信を行います。それにより、難読化のレイヤーが追加され、一般的な VPN/プロキシの検出用に調整されたディープ パケット インスペクション ツールによる検出を回避します。しかし、オーキッドが中国で広く使用されるようになれば、オーキッドの WebRTC トラフィックを認識できるように GFW のパケット インスペクション システムが調整される可能性が高くなります。その場合は、さらに高度な難読化のプラグインを開発する必要があります。

さらに問題なのは、オーキッドのメインの探索プロセスはイーサリアム ブロックチェーンで公開されているパブリック ノード ディレクトリに依存していることです (セクション 4.2)。オーキッドが中国で広く使用され、注意が向けられるようになれば、GFW が自動的にイーサリアム ブロックチェーンを監視し、パブリック ディレクトリに登録されているすべてのオーキッド ノードの IP アドレスを禁止する可能性は非常に高いでしょう。

このような障害があるとしても、中国国外の友人や愛好家が (おそらくは無償の) 入口ノードを運営し、個人的にアドレスを共有することにより、草の根レベルで限定的ではあるものの、中国の人々は、そのままオーキッドを使用することができます。支援者や慈善家は、秘密のオーキッド ノード アドレスと同じように、プライベートのソーシャル チャネルで暗号通貨 OXT を流通させることにより、この活動を支援できます。GFW の回避を強化し、中国における OXT の流通を促進するためのコア部分の設計の改善は、今後の研究が期待される分野です (セクション 7)。

6.5 監視の回避

インターネットの監視は、インターネットの検閲よりもさらに広く行われています。ほとんどの司法管轄区において、ISP は、司法当局から正当な監視の要請があれば応じなければならないという法的義務を負っています。また、欧米の主要な情報機関で超法規的な監視行動が広く行われていることは、いまや公然の秘密となっています。ここでは、攻撃者の能力の組み合わせをいくつか想定して、このように幅の広いシナリオをいくつかのモデルに分解してみます。

受動的な ISP の監視

世界のほとんどの地域において、インターネット サービス プロバイダーは、自社の顧客のインターネット トラフィックを監視し、ログの収集を行うことが機能的にもその立場的にも可能です。司法管轄区によっては、司法当局の調査に協力するためのログの収集が法律で義務付けられています。また、ISP では、戦略的な理由から特定のアプリケーションを優先するためのトラフィック シェーピングを目的として、パケットの解析を行う場合があります。ユーザーの閲覧履歴を収集して広告主に販売する場合もあります。

このようなシナリオについては、攻撃者には、オーキッド ネットワークや宛先のエンドポイントに侵入する動機や能力はないと考えられます。このケースでは、接続先のエンドポイントも同じ ISP の管理下にあるのでない限り、一般的な非標的型のトラフィック解析による監視を回避するには、シングル ホップの回路で十分です。粗雑なパケット解析ツールであれば、WebRTC のエンコーディングによってオーキッド トラフィックを通常のウェブ リクエストと見せかけることもできますが、攻撃者がオーキッドを知っていて、高度なディープ パケット インスペクションの手法を使用している場合は、見抜かれます。

攻撃者がウェブサイトのフィンガープリントの手法を使用している場合、シングルホップの回路では攻撃者に対する防御が弱くなります [65–67]。マルチホップの回路では、このような攻撃の精度や再現性は低下しますが、攻撃を無効にするには十分ではありません。このような相関手法は、大規模に実施するにはコストが高すぎるものの、標的型攻撃ではユーザーの脅威となる可能性があると考えられます。

受動的な ISP およびエンドポイントの監視

次のシナリオでは、攻撃者はエンドポイントのトラフィックを監視することはできるものの、ISP を通過するユーザーの入力トラフィックの積極的なシェーピングやコントロールはできない場合を考えます。このシナリオは、特定のエンドポイント (ウェブサイトなど) を積極的に監視し、トラフィック解析を使用して対象のエンドポイントを使用するユーザーの情報を収集する機関に相当します。攻撃者は、対象のユーザーの IP アドレスを発見すると、次にそれを使用して、ISP からそのユーザーのトラフィックのログや個人情報を取得します。

これにより攻撃者は、さらにパッシブ フローの相関分析を行えるようになります [20-22]。ただし、この手法も、ISP を通過するすべてのトラフィックを対象として大規模に実行するにはコストが高すぎると考えられます。攻撃者が分析にかけられる予算はある程度限られており、可能性の高いユーザーの IP アドレスを相関分析の対象とする必要があります。

このケースでは、エンドポイントの接続も HTTPS/SSL を使用して暗号化されていて、ユーザーがまだ標的になっていなければ、シングルホップの回路で十分に監視を回避できます。攻撃者が確認できるのは、オーキッド ノードからエンドポイントまでの接続だけであり、ユーザーの IP アドレスを容易に特定することはできません。

セクション 5.8 で説明したように、攻撃者がエンドポイントのトラフィックをすべて監視している場合、オーキッド ノードとエンドポイントの間のトラフィックのタイミングと、そのノードによる当選チケットの引き換えの相関性を特定できる可能性があります。チケットによって支払者の オーキッド ナノペイメント アドレスが明らかになると、攻撃者はそれをたどってユーザーを特定できます。ユーザーは、適切な手順を実行して、暗号通貨 OXT を匿名化することにより、これを回避できます。

エンドポイントおよびオーキッドへの潜入

次に、攻撃者にはユーザーの ISP におけるデータを監視する能力がない一方、エンドポイントまたはオーキッド ネットワークへの潜入は可能である場合を考えます。これは、ユーザーの ISP が大規模なトラフィックのログ収集を行っていない場合や、重要なトラフィック データを攻撃者と共有していない場合に現実的なモデルです。この場合、ユーザーから最初の オーキッド ノードまでの回線上のトラフィックを監視できなければ、フローの相関分析攻撃は非常に困難になります。

攻撃者は、オーキッド ネットワークに潜入することにより、フローの相関分析攻撃を実行できます。潜入の有効性は、オーキッド ノードに対してかけられる攻撃者の予算に左右されます。ステーキングのしくみにより、キャプチャにかかるユーザーあたりのコストは相対的に高くなります。さらに、オーキッドのユーザー数が増加すると、それに比例して、一定のパーセンテージのオーキッド接続をキャプチャするコストが高くなります。これについては、*セクション 4.4* で説明しました。攻撃者は、トラフィックログを保管しているオーキッド ノードのオペレーターと結託して、オペレーターにログを要求するか、オーキッド ノードを直接コントロールすることにより、回路を侵害することができます。単一のノードが侵害される確率は、次のとおりです。

$$p(compromise(x)) = p(x \subseteq \alpha) + (1 - p(x \subseteq \alpha))p(x \subseteq A) \quad (24)$$

$$p(x \subseteq \alpha) = \frac{S_{\alpha \cap W}}{S_W} \quad (25)$$

$$p(x \subseteq A) = \frac{S_{A \cap W}}{S_W} \quad (26)$$

x: ランダムに選択されたオーキッド ノード
α:攻撃者のためにデータのログ収集を行う、攻撃者と結託している一連のオーキッド ノード
A :攻撃者が直接コントロールしている一連のオーキッド ノード
W: 一連の Orchid ノードが登録された、クライアントのホワイトリスト
S_W:W のノードの OXT ステークの合計
$S_{\alpha \cap W}$:$\alpha \cap W$ のノードの OXT ステークの合計、W にも存在する結託している一連のノード
$S_{A \cap W}$:$A \cap W$ のノードの OXT ステークの合計、W にも存在する攻撃者のノード

攻撃者がリンクを確認するために直接的な IP アドレスのメタデータが必要な場合、マルチホップの回路では、すべてのエッジ、すなわち 1 つおきのノードを侵害する必要があります。そこで、マルチホップの回路で侵害が生じる確率は、シングルホップの場合の確率のべき関数となります。

$$p(compromise(X_k)) = (\frac{S_\alpha \cap W}{S_W} + (1 - \frac{S_\alpha \cap W}{S_W})\frac{S_A \cap W}{S_W})^{[k/2]} \qquad (27)$$

攻撃者にトラフィック解析を行う余裕があり、不完全な統計精度/再現性を許容できる場合を除き、マルチホップの回路ではセキュリティを大幅に向上させることができます。*セクション 6.1* で説明したように、攻撃者がフローの相関分析を使用する場合、マルチホップ回路の侵害の確率は、シングルホップ回路の場合に近付きます (式 24)。

強力な攻撃者

強力な攻撃者は、ISP または自律システム (AS) レベルでパケットをコントロールできる場合があります。ユーザーの ISP のトラフィックを監視する能力しかない攻撃者であっても、ウェブサイトのフィンガープリント攻撃を使用することにより、マルチホップ回路でユーザーとウェブサイトの相関分析を行える可能性があります。その際、主な障害となるのはコストです。クライアントでは、*帯域消費*を使用することにより、このような攻撃に対してある程度の保護を実現できます。暗号化されたトラフィック ストリームにパディングを行い、本来のデータ ストリームに関係なく、一定のサイズのパケットを短い間隔で定期的に送信することにより、大半のトラフィック解析手法で使用されている時間相関を検出できないようにします。攻撃者がユーザーあたりの解析の予算を潤沢に確保でき、検出および推定の能力に優れている場合は、このように付加的な保護手段がなければ、マルチホップの回路でも侵害される可能性があります。この可能性については、次のセクション「今後の課題」で議論します。

7. 今後の課題

オーキッドは、スケーラブルなオフチェーンのナノペイメントにより、分散型プロキシ サービスの帯域幅マーケットプレイスを実現します。これを土台として、匿名性、使いやすさ、検閲耐性、経済的セキュリティを改善するさまざまな方法を確認してきました。

トラフィック解析への耐性

オーキッドの現在のルーティング設計では、レイテンシが最小限に抑えられ、帯域幅を最大限に活用できますが、トラフィック解析攻撃があった場合に匿名性が損なわれるリスクがあります。このようなレイテンシ、帯域幅、匿名性のトレードオフは根本的なことと言えるでしょう [26]。ユーザーが強力な匿名性を求める場合は、帯域消費 (一定レート送信ストリーム) を使用することができます。これにより、時間で変動する特徴の多くを消すことができ、トラフィック解析に対抗するのに役立ちます。さまざまな推論攻撃に対抗するには、帯域消費を超えるさらなる機能強化が求められるでしょう [73]。本格的な分析については、今後の課題です。レイテンシを考慮した経路構築を改善するだけで、同じレイテンシ

で長い回路を実現でき、少ないアクティブ エッジでより多くのストリームをミキシングして、密度の低いコネクション グラフを使用することにより、ミキシングを改善できます。

支払いの匿名性

オーキッドのナノペイメント システムはイーサリアムをベースとしているため、半匿名に過ぎません。そこで、完全な支払いの匿名性を求めるユーザーは、ナノペイメント アカウントに資金を供給する前に、暗号通貨 OXT を外部で匿名化する必要があり、これが使い勝手に関する障害となります。

別の方法として、オーキッドのナノペイメントおよび回路自体で高速ミキシングを可能にすることも考えられます。ディレクトリ サービスを転用して、ミキシングを行うノードやミキシング ピアを登録するノードを宣伝できます。このユースケースでは、二重支払いやグリーフィングの対策機構に負担がかかる可能性があり (5.10)、二重支払いの検出と防止のための改善が必要になる可能性があります。

低分散型のナノペイメント

現在のオーキッドのナノペイメントのしくみには、根本的に分散とオーバーヘッドのトレードオフがあります。分散の主な発生源は、チケットの統計的独立です。相互排他的なチケット スキームを使用することにより、分散を排除することが可能です。もっともシンプルな形では、ペイメント アカウントごとに当選チケットが 1 枚となります。一連のチケット全体で当選者は 1 名となるため、分散は排除されます。トレードオフの 1 つは、相互排他的なチケットでは、シンプルな 2 者間のエントロピー プロトコルではなく、マルチパーティのエントロピーの発生源を使用して、チケット当選者の決定を将来に先延ばしする必要が生じることです。イーサリアムのブロックチェーン自体をシンプルなエントロピーの発生源として使用でき、ナノペイメント決済で必要な、小規模な取引額を扱う上でおそらく十分に安全でしょう。ただし、当選者の決定を先延ばしにすると、未決済の支払いが大幅に増えることになり、ナノペイメントあたりのストレージ コストが増加することになります。

トラフィックの難読化

トラフィックの難読化と検出の研究分野では、激しい競争が続いています。トラフィックの難読化には、ランダム化 [74、75]、変形/擬態 [76]、トンネリング [76、77]、生成モデリング [78] などの手法が用いられます。残念ながらこれらの手法はいずれも、実際のトラフィックと難読化されたトラフィックのサンプルでトレーニングを行った機械学習ベースの検出 [27] システムに対しては強くありません。一般に、強力な難読化ツールほど、必要なバイトあたりの計算負荷が高くなります。難読化の問題は、ある種の GAN [79] として考えてみることができます。生成ネットワークで、可逆性や復元特性を維持しながら検出を回避するためのトラフィック ストリームの変形を学習し、識別ネットワークで、実際のストリームと変形されたストリームの識別を学習します。これにより、ディープ ラーニング ベースの難読化 (および検出ツール) の道が開けます。

検閲耐性の強化

オーキッドの国家レベルの検閲を回避する能力は、主にイーサリアムブロックチェーンのノードのパブリックな宣伝により制限されます。強力な検閲耐性を実現するには、何らかのプライベートな宣伝が必要になります。これは、帯域幅販売者が、正規の顧客にはブロックされていない IP アドレスを宣伝し、攻撃者に対しては秘匿しようとするゲームとしてモデル化することができます。販売者は、IP アドレスを知った正規の顧客ごとに将来予定される収益を得ますが、その IP アドレスを攻撃者が発見してブロッ

クすると、その後の収益は失われます。販売者が実行できる戦略は、アフィリエイトのスキームを使用することです。宣伝してくれるピアに対して将来の収益の一部を報酬として渡します。これにより、結託する攻撃者を避けながら、ノードを見つけて正規のユーザーに宣伝することに長けたアフィリエイトのニッチ市場が生まれます。

ホワイトリストの保証証券

特定のホワイトリストにノードが含まれることに対して OXT をステーキングできるようにすると、ステーキングとステークウェイティングのプラスのインセンティブの調整効果を最大限に高めることができます。(ステークが引き出される前に) ノードがこのリストから削除された場合、ステーク デポジットは没収され、消失します。このステークは、ノード プロバイダーの信頼性を証明する保証証書のような役割を果たします。ノード プロバイダーに悪質な行為があればそのプロバイダーのお金が失われるリスクがあるためです。考え方はシンプルですが、インセンティブを注意深く設計し、検証する必要があります。

革新的なインセンティブの構造を持つ独自のキュレーション リストをぜひ作成してください。

8. 謝辞

オーキッド は共同チーム プロジェクトであり、特に Gustav Simonsson 氏および David Salamon 氏には、(ホワイトペーパーのバージョン 0.9.2 の執筆も含め) その知性により多大な貢献をいただいたことに謝意を表します。

参考文献

1. Dingledine R、Mathewson N、Syverson P、「Tor: The Second-Generation Onion Router」(インターネット)、2004 年、入手元: http://dx.doi.org/10.21236/ada465464

2. Shahbar K、Nur Zincir-Heywood A、「Effects of Shared Bandwidth on Anonymity of the I2P Network Users」(インターネット)、2017 IEEE Security and Privacy Workshops (SPW)、2017 年、入手元: http://dx.doi.org/10.1109/spw.2017.19

3. Chaum D、「Untraceable Electronic Mail, Return Addresses and Digital Pseudonyms」(インターネット)、「Advances in Information Security」シリーズ、2003 年、p. 211 〜 219、入手元: http://dx.doi.org/10.1007/978-1-4615-0239-5_14

4. 「HashCash」(インターネット)、2002 年 (2019 年 9 月 10 日引用)、入手元: http://www.hashcash.org/hashcash.pdf

5. 「Bitcoin: A Peer-to-Peer Electronic Cash System」(インターネット)、(2019 年 9 月 10 日引用)、入手元: https://bitcoin.org/bitcoin.pdf

6. 「オーキッド (0.9.2)」(インターネット)、2019 年 (2019 年 9 月 10 日引用)、入手元: https://www.orchid.com/assets/whitepaper/whitepaper.pdf

7. Stoica I、Morris R、Liben-Nowell D、Karger DR、Kaashoek MF、Dabek F ほか、「Chord: a scalable peer-to-peer lookup protocol for internet applications」(インターネット)、Vol. 11、IEEE/ACM Transactions on Networking、2003 年、p. 17 〜 32、入手元: http://dx.doi.org/10.1109/tnet.2002.808407

8. Wood DD、「ETHEREUM:A SECURE DECENTRALISED GENERALISED TRANSACTION LEDGER」、2014 年 (2019 年 9 月 11 日引用)、入手元: https://pdfs.semanticscholar.org/ee5f/d86e5210b2b59f932a131fda164f030f915e.pdf

9. 「A Protocol for Packet Network Intercommunication」(インターネット)、『The Best of the Best』、2009 年、入手元: http://dx.doi.org/10.1109/9780470546543.ch54

10. Fadilpašić S、「China 'hijacked traffic' to spy on the West」(インターネット)、ITProPortal、2018 年 (2019 年 11 月 17 日引用)、入手元: https://www.itproportal.com/news/china-eavesdropping-on-western-communication-for-years-research-claims/

11. 「YouTube, Netflix Videos Found to Be Slowed by Wireless Carriers」(インターネット)、(2019 年 11 月 17 日引用)、入手元: https://www.bloomberg.com/news/articles/2018-09-04/youtube-and-netflix-throttled-by-carriers-research-finds

12. Morran BC、「House Votes To Allow Internet Service Providers To Sell, Share Your Personal Information」(インターネット)、Consumer Reports、(2019 年 11 月 17 日引用)、入手元: https://www.consumerreports.org/consumerist/house-votes-to-allow-internet-service-providers-to-sell-share-your-personal-information/

13. 「Net Neutrality:Caught in a web of lobbying and regulatory uncertainty」(インターネット)、Sustainalytics、2018 年 (2019 年 11 月 17 日引用)、入手元: https://www.sustainalytics.com/esg-blog/net-neutrality-caught-in-a-web-of-lobbying-and-regulatory-uncertainty/

14. Rosenberg S、「Facebook's reputation is sinking fast」(インターネット)、Axios、2019 年 (2019 年 11 月 17 日引用)、入手元: https://www.axios.com/facebook-reputation-drops-axios-harris-poll-0d6c406a-4c2e-463a-af98-1748d3e0ab9a.html

15. Marks G、「Facebook Usage Drops 26 Percent…And Other Small Business Tech News This Week」(インターネット)、Forbes、2019 年 (2019 年 11 月 17 日)、入手元: https://www.forbes.com/sites/quickerbettertech/2019/10/27/facebook-usage-drops-26-percentand-other-small-business-tech-news-this-week/

16. Brodkin J、「50 million US homes have only one 25Mbps Internet provider or none at all」(インターネット)、Ars Technica、2017 年 (2019 年 11 月 17 日引用)、入手元: https://arstechnica.com/information-technology/2017/06/50-million-us-homes-have-only-one-25mbps-internet-provider-or-none-at-all/

17. 「SSH COMMUNICATIONS SECURITY CELEBRATES 20 YEARS AS INDUSTRY STANDARD」(インターネット)、(2019 年 11 月 17 日引用)、入手元: https://www.ssh.com/press-releases/111-ssh-communications-security-celebrates-20-years-as-industry-standard

18. Fu X、Graham B、Bettati R、Zhao W、「Active traffic analysis attacks and countermeasures」(インターネット)、『2003 International Conference on Computer Networks and Mobile Computing』、2003 年、ICCNMC 2003、入手元: http://dx.doi.org/10.1109/iccnmc.2003.1243024

19. Dixon C、Bragin T、Krishnamurthy A、Anderson T、「Tit-for-Tat Distributed Resource Allocation」、(2019 年 9 月 23 日引用)、入手元: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.182.1544

20. Karakaya M、Korpeoglu I、Ulusoy Ö、「Free Riding in Peer-to-Peer Networks」(インターネット)、『Vol. 13, IEEE Internet Computing』、2009 年、p. 92 〜 98、入手元: http://dx.doi.org/10.1109/mic.2009.33

21. Ngan T-W “johnny”、Dingledine R、Wallach DS、「Building Incentives into Tor」(インターネット)、「Financial Cryptography and Data Security」シリーズ、2010 年、p. 238 〜 256、入手元: http://dx.doi.org/10.1007/978-3-642-14577-3_19

22. Androulaki E、Raykova M、Srivatsan S、Stavrou A、Bellovin SM、「PAR: Payment for Anonymous Routing」(インターネット)、『Privacy Enhancing Technologies』、p.219 〜 236、入手元: http://dx.doi.org/10.1007/978-3-540-70630-4_14

23. Ghosh M、Richardson M、Ford B、Jansen R、「A TorPath to TorCoin: Proof-of-Bandwidth Altcoins for Compensating Relays」、2014 年 7 月 18 日 (2019 年 9 月 23 日引用)、入手元: https://apps.dtic.mil/dtic/tr/fulltext/u2/a621867.pdf

24. 「A Protocol for Interledger Payments」(インターネット)、(2019 年 9 月 23 日引用)、入手元: https://pdfs.semanticscholar.org/ab98/c62a7efdc5362c7f36589680597a93f3111f.pdf

25. Khosla A、Saran V、Zoghb N、「Techniques for Privacy Over the Interledger」、2018 年 (2019 年 9 月 23 日引用)、入手元: https://pdfs.semanticscholar.org/02f3/aae499723063cf9c3cc42508cae13d16aa7d.pdf

26. Das D、Meiser S、Mohammadi E、Kate A、「Anonymity Trilemma: Strong Anonymity, Low Bandwidth Overhead, Low Latency - Choose Two」(インターネット)、『2018 IEEE Symposium on Security and Privacy (SP)』、2018 年、入手元: http://dx.doi.org/10.1109/sp.2018.00011

27. Wang L、Dyer KP、Akella A、Ristenpart T、Shrimpton T、「Seeing through Network-Protocol Obfuscation」(インターネット)、「CCS '15- Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security」、2015 年、入手元: http://dx.doi.org/10.1145/2810103.2813715

28. Budish E、「The Economic Limits of Bitcoin and the Blockchain」(インターネット)、2018 年、入手元: http://dx.doi.org/10.3386/w24717

29. ウェブサイト (インターネット)、(2019 年 10 月 2 日引用)、入手元: https://bitinfocharts.com/comparison/ethereum-transactionfees.html

30. 「Bitcoin Avg. Transaction Fee chart」(インターネット)、BitInfoCharts、(2019 年 10 月 2 日引用)、入手元: https://bitinfocharts.com/

31. Khattak S、Elahi T、Simon L、Swanson CM、Murdoch SJ、Goldberg I、「SoK: Making Sense of Censorship Resistance Systems」(インターネット)、『Proceedings on Privacy Enhancing Technologies』 Vol. 2016、2016 年、p. 37 〜 61、入手元: http://dx.doi.org/10.1515/popets-2016-0028

32. Wikimedia プロジェクト寄稿者、ISO/IEC 7816 - Wikipedia (インターネット)、Wikimedia Foundation, Inc、2002 年 (2019 年 10 月 2 日引用)、入手元: https://en.wikipedia.org/wiki/ISO/IEC_7816

33. EBICS.ORG ホームページ(インターネット)、(2019 年 10 月 2 日引用)、入手元: http://www.ebics.org/home-page

34. ウェブサイト (インターネット)、(2019 年 10 月 2 日引用)、入手元: https://www.swift.com/

35. ウェブサイト (インターネット)、(2019 年 10 月 2 日引用)、入手元: https://www.swift.com/

36. ウェブサイト (インターネット)、(2019 年 10 月 2 日引用)、入手元: http://www.nyce.net/about

37. ウェブサイト (インターネット)、(2019 年 10 月 2 日引用)、入手元: http://www.investopedia.com/terms/r/reconciliation.asp

38. [タイトルなし] (インターネット)、(2019 年 10 月 2 日引用)、入手元: https://www.aba.com/-/media/archives/endorsed/rippleshot-state-of-card-fraud.pdf

39. Mian A、Hameed A、Khayyam M、Ahmed F、Beraldi R、「Enhancing communication adaptability between payment card processing networks」(インターネット)、『IEEE Communications Magazine』 Vol. 53、2015 年、p. 58 〜 64、入手元: http://dx.doi.org/10.1109/mcom.2015.7060519

40. 「Banks and WikiLeaks」、NY Times (インターネット)、2010 年 12 月 25 日 (2019 年 10 月 2 日引用)、入手元: https://www.nytimes.com/2010/12/26/opinion/26sun3.html

41. 「What are common credit card processing fees?」(インターネット)、Quora、(2019 年 10 月 2 日引用)、入手元: https://www.quora.com/What-are-common-credit-card-processing-fees

42. ウェブサイト (インターネット)、(2019 年 10 月 2 日引用)、入手元: https://www.nerdwallet.com/blog/banking/wire-transfers-what-banks-charge

43. ウェブサイト (インターネット)、(2019 年 10 月 2 日引用)、入手元: https://www.valuepenguin.com/what-credit-card-processing-fees-costs

44. ウェブサイト (インターネット)、(2019 年 10 月 2 日引用)、入手元: https://www.economist.com/blogs/dailychart/2010/12/remittances

45. ウェブサイト (インターネット)、(2019 年 10 月 2 日引用)、入手元: https://financefeeds.com/alipay-vs-wechat-pay-vs-unionpay-important-research/

46. Joseph Poon TD、「The Bitcoin Lightning Network: Scalable Off-Chain Instant Payments」(インターネット)、入手元: https://lightning.network/lightning-network-paper.pdf

47. [タイトルなし] (インターネット)、(2019 年 10 月 2 日引用)、入手元: https://www.usenix.org/system/files/conference/usenixsecurity15/sec15-paper-heilman.pdf

48. Wheeler D、「Transactions using bets」(インターネット)、『Security Protocols』、1997 年、p. 89 〜 92、入手元: http://dx.doi.org/10.1007/3-540-62494-5_7

49. Rivest RL、「Peppercoin Micropayments」(インターネット)、『Financial Cryptography』、2004 年、p. 2 〜 8、入手元: http://dx.doi.org/10.1007/978-3-540-27809-2_2

50. Pass R、Shelat A、「Micropayments for Decentralized Currencies」(インターネット)、「CCS '15- Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security」、2015 年、入手元: http://dx.doi.org/10.1145/2810103.2813713

51. 「Ethereum Avg. Transaction Fee chart」(インターネット)、BitInfoCharts、(2019 年 10 月 2 日引用)、入手元: https://bitinfocharts.com/

52. ウェブサイト (インターネット)、(2019 年 10 月 2 日引用)、入手元: https://etherscan.io/chart/gaslimit

53. ウェブサイト (インターネット)、(2019 年 10 月 2 日引用)、入手元: https://etherscan.io/chart/blocktime

54. Nasr M、Bahramali A、Houmansadr A、「DeepCorr: Strong Flow Correlation Attacks on Tor Using Deep Learning」、『Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security』、ACM、2018 年、p. 1962 〜 1976

55. Borisov N、Danezis G、Mittal P、Tabriz P、「Denial of service or denial of security?」、『Proceedings of the 14th ACM conference on Computer and communications security』、ACM、2007 年、p. 92 〜 102

56. Sun Y、Edmundson A、Vanbever L、Li O、Rexford J、Chiang M ほか、「RAPTOR: Routing Attacks on Privacy in Tor」、2015 年 (2019 年 9 月 16 日)、入手元: https://pdfs.semanticscholar.org/76c7/73bb98b0a266970a589f2cabbd24565b6e19.pdf

57. Johnson A、Wacek C、Jansen R、Sherr M、Syverson P、「Users get routed」(インターネット)、『CCS '13- Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security』、2013 年、入手元: http://dx.doi.org/10.1145/2508859.2516651

58. Houmansadr A、Kiyavash N、Borisov N、「Multi-flow attack resistant watermarks for network flows」(インターネット)、『2009 IEEE International Conference on Acoustics, Speech and Signal Processing』、2009 年、入手元: http://dx.doi.org/10.1109/icassp.2009.4959879

59. Zhang L、Wang Z、Xu J、Wang Q、「Multi-flow Attack Resistant Interval-Based Watermarks for Tracing Multiple Network Flows」(インターネット)、『Computing and Intelligent Systems』、2011 年、p. 166 〜 173、入手元: http://dx.doi.org/10.1007/978-3-642-24010-2_23

60. Yu W、Fu X、Graham S、Xuan D、Zhao W、「DSSS-Based Flow Marking Technique for Invisible Traceback」(インターネット)、『2007 IEEE Symposium on Security and Privacy (SP '07)』、2007 年、入手元: http://dx.doi.org/10.1109/sp.2007.14

61. Murdoch SJ、Danezis G、「Low-Cost Traffic Analysis of Tor」(インターネット)、『2005 IEEE Symposium on Security and Privacy (S&P'05)』、入手元: http://dx.doi.org/10.1109/sp.2005.12

62. Chakravarty S、Stavrou A、Keromytis AD、「Traffic Analysis against Low-Latency Anonymity Networks Using Available Bandwidth Estimation」、『Computer Security – ESORICS 2010』、Springer (ベルリン、ハイデルベルク)、2010 年、p. 249 〜 267

63. Panchenko A、Niessen L、Zinnen A、Engel T、「Website fingerprinting in onion routing based anonymization networks」(インターネット)、『WPES '11- Proceedings of the 10th annual ACM workshop on Privacy in the electronic society』、2011 年、入手元: http://dx.doi.org/10.1145/2046556.2046570

64. Cai X、Zhang XC、Joshi B、Johnson R、「Touching from a distance」(インターネット)、『CCS '12- Proceedings of the 2012 ACM conference on Computer and communications security』、2012 年、入手元: http://dx.doi.org/10.1145/2382196.2382260

65. Rimmer V、Preuveneers D、Juarez M、Van Goethem T、Joosen W、「Automated Website Fingerprinting through Deep Learning」(インターネット)、『Proceedings 2018 Network and Distributed System Security Symposium』、2018 年、入手元: http://dx.doi.org/10.14722/ndss.2018.23105

66. Bhat S、Lu D、Kwon A、Devadas S、「Var-CNN: A Data-Efficient Website Fingerprinting Attack Based on Deep Learning」(インターネット)、『Proceedings on Privacy Enhancing Technologies』 Vol. 2019、2019 年、p. 292 〜 310、入手元: http://dx.doi.org/10.2478/popets-2019-0070

67. Sirinam P、Imani M、Juarez M、Wright M、「Deep Fingerprinting: Undermining Website Fingerprinting Defenses with Deep Learning」、『Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security』、ACM、2018 年、p. 1928 〜 1943

68. 「Tor Blog: A Critique of Website Traffic Fingerprinting Attacks」(インターネット)、(2019 年 9 月 17 日引用)、入手元: https://blog.torproject.org/critique-website-traffic-fingerprinting-attacks

69. Pearce P、Ensafi R、Li F、Feamster N、Paxson V、「Augur: Internet-Wide Detection of Connectivity Disruptions」(インターネット)、『2017 IEEE Symposium on Security and Privacy (SP)』、2017 年、入手元: http://dx.doi.org/10.1109/sp.2017.55

70. Humphries M、「China Starts Issuing $145 Fines for Using a VPN」(インターネット)、PCMAG、2019 年 (2019 年 9 月 15 日引用)、入手元: https://www.pcmag.com/news/365860/china-starts-issuing-145-fines-for-using-a-vpn

71. 「Global Trends in the VPN Industry: VPN Usage Statistics」(インターネット)、GeoSurf、2019 年 (2019 年 9 月 15 日引用)、入手元: https://www.geosurf.com/blog/vpn-usage-statistics/

72. Ensafi R、Fifield D、Winter P、Feamster N、Weaver N、Paxson V、「Examining How the Great Firewall Discovers Hidden Circumvention Servers」(インターネット)、『IMC '15- Proceedings of the 2015 ACM Conference on Internet Measurement Conference』、2015 年、入手元: http://dx.doi.org/10.1145/2815675.2815690

73. Chen、Chen C、Asoni DE、Perrig A、Barrera D、Danezis G ほか、「TARANET: Traffic-Analysis Resistant Anonymity at the Network Layer」(インターネット)、『2018 IEEE European Symposium on Security and Privacy (EuroS&P)』、2018 年、入手元: http://dx.doi.org/10.1109/eurosp.2018.00018

74. Meiklejohn S、Mercer R、「Möbius: Trustless Tumbling for Transaction Privacy」(インターネット)、『Proceedings on Privacy Enhancing Technologies』 Vol. 2018、2018 年、p. 105 〜 121、入手元: http://dx.doi.org/10.1515/popets-2018-0015

75. Winter P、Pulls T、Fuss J、「ScrambleSuit: A Polymorph Network Protocol to Circumvent Censorship」(インターネット)、2013 年 (2019 年 9 月 18 日引用)、入手元: http://arxiv.org/abs/1305.3199

76. Moghaddam HM、「Skypemorph: Protocol Obfuscation for Censorship Resistance」、2013 年、p. 54

77. Brubaker C、Houmansadr A、Shmatikov V、「CloudTransport: Using Cloud Storage for Censorship-Resistant Networking」(インターネット)、『Privacy Enhancing Technologies』、2014 年、p. 1 〜 20、入手元: http://dx.doi.org/10.1007/978-3-319-08506-7_1

78. Dyer KP、Coull SE、Shrimpton T、「Marionette: A Programmable Network Traffic Obfuscation System」、『Proceedings of the 24th USENIX Security Symposium』(USENIX Security '15)、2015 年、p. 367 〜 382、

79. Goodfellow I、Pouget-Abadie J、Mirza M、Xu B、Warde-Farley D、Ozair S ほか、「Generative Adversarial Nets」、『Advances in Neural Information Processing Systems』、2014 年、p. 2672 〜 2680、