Computer Security Symposium 2019
21 - 24 October 2019

MWmessage: 追跡困難メッセージングを実現するための Mimblewimbleの拡張

宮前 剛[1,2,a)]　松浦 幹太[2]

概要：暗号資産の代表である Bitcoin は、トランザクション分析技術の急速な進歩に伴い、追跡困難性が低下している。そのような状況の中、追跡困難性とスケーラビリティを両立する匿名暗号資産プロトコル Mimblewimble が注目されている。しかし、現時点で Mimblewimble はメッセージングに関して多くの課題を抱えている。そこで、我々は、Mimblewimble のメッセージングの課題を解決するために、Mimblewimble 自身を拡張してブロックチェーン形式の追跡困難メッセージングを実現する MWmessage を提案する。
本稿では、まず Mimblewimble の仕組みと特徴およびそのメッセージングに関する課題を整理した上で、課題を解決するための MWmessage の仕組みを説明する。特に、メッセージの改竄を困難にするためのメッセージハッシュの仕組みと、Mimblewimble のスケーラビリティを犠牲にすることなく追跡困難メッセージングを実現するためのメッセージへの有効期限付与の仕組みに焦点を当てる。最後に、MWmessage を一般の匿名メッセージングシステムとして机上評価する。

キーワード：ブロックチェーン，Mimblewimble，匿名メッセージング，追跡困難性、スケーラビリティ

MWmessage: An Extension of Mimblewimble for Unlinkable Messaging

Takeshi Miyamae[1,2,a)]　Kanta Matsuura[2]

Abstract: In Bitcoin, which has been the representative cryptocurrency until today, unlinkability is being reduced because of the various current technologies for transaction analysis. In this situation, Mimblewimble, an anonymous cryptocurrency protocol which is designed to achieve unlinkability and scalability at the same time, is attracting public attention. However, Mimblewimble has several problems in its messaging at this moment. Therefore, we propose MWmessage, a novel blockchain-based unlinkable messaging system to solve Mimblewimble's messaging problems by extending Mimblewimble itself.
In this paper, we first summarize Mimblewimble's architecture, features, and problems in messaging. We next show MWmessage's architecture, which solves the problems. Especially, we focus on the mechanism of message hash to keep the message payload tamper-resistant and the mechanism introducing message expiration time to realize unlinkable messaging without sacrificing Mimblewimble's scalability. Finally, we evaluate MWmessage logically as a general private messaging system.

Keywords: blockchain, Mimblewimble, private messaging, unlinkability, scalability

1. 序論

1.1 背景

ブロックチェーンを応用した暗号資産のうち今日まで最もよく使われてきた Bitcoin は、トランザクション分析技術の急速な進歩に伴い、追跡困難性が低下している。例え

[1] （株）富士通研究所セキュリティ研究所
Security Laboratory, Fujitsu Laboratories Ltd.
[2] 東京大学生産技術研究所
Institute of Industrial Science, The University of Tokyo
[a)] miyamae.takeshi@fujitsu.com

ば、Reid ら [1] は、Bitcoin のトランザクション情報を元にアドレス間の関係をグラフ構造で表現し分析する手法を紹介した。また、Ron ら [2] は、Bitcoin のトランザクション情報を分析することにより、一見無関係に思わせる異なるアドレスが実は同一ユーザのものであると推測できることを示した。そのような背景から、暗号資産の研究領域では、近年、取引情報のプライバシーを維持するための匿名化技術に注目が集っている。なお、暗号資産の追跡困難性（unlinkability）の定義は文献によって異なるため引用を避けるが、本稿では「異なるトランザクションまたは立場（送金者と受領者）で使用される同一アドレスが、当事者以外からは同一であると推定することが困難であること」と定義する。

1.2 Mimblewimble の仕組みと特徴

1.2.1 概要

Mimblewimble は、Tom Elvis Jedusor を名乗る匿名の人物が Internet Relay Chat(IRC) に投稿したアイデアを、Poelstra [3] が定式化した匿名暗号資産のプロトコルである。Mimblewimble は、追跡困難性とスケーラビリティを両立できる性質のために、数ある匿名暗号資産の技術の中でも特に注目されている。

現在、Mimblewimble の実装のうち、Beam[4] と Grin[5] の 2 つが広く知られている。

1.2.2 Confidential Transactions

Mimblewimble は、Confidential Transactions(CT)[6] と呼ばれるトランザクション方式を用いている（図 1）。CT は、通貨量 v とブラインディングファクター r を秘密情報とする Pedersen Commitment(PC)[7] : $(rG+vH)$ の集合として定義されるが、

(1) 全ての PC を足し合わせると 0 になること。
(2) 全ての PC の通貨量 v の部分だけを足し合わせても 0 になること。

の 2 つの制約を満たすようにするために、全体を足すと 0 になるように計算された $v_e = 0$ の特別な PC である剰余値（excess value）を追加する。

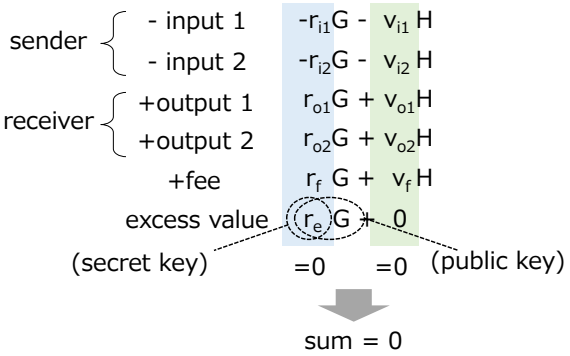

図 1　Confidential Transactions (CT)

Mimblewimble は、CT を非中央集権的な暗号資産システムのトランザクションとして動作させるため、まずマイナーがブロック生成と同時に既定の通貨量 v_m を持つ PC を生成することを許可し、これをマイニングの報酬とする。以降、送金者が未使用の PC を符号反転させた数値の集合で inputs を構成し、受領者が新しい PC を生成して outputs を構成し、最後に受領者が剰余値を追加して上記の条件を満たすように構成した CT をマイナーが承認することにより、送金者と受領者の間で通貨の所有権の移転が行われる。なお、受領者がこのような剰余値を計算できるようにするためには、受領者は予め送金者から inputs の PC の秘密情報を全て開示されている必要がある。

このとき、受領者は剰余値の通貨量が $v_e = 0$ であること（PC の集合が CT の (2) の条件を満たすことと等価）を証明するため、秘密鍵に相当する秘密情報 r_e を使って剰余値 r_eG に対して生成した電子署名が公開鍵に相当する剰余値 r_eG で検証できることを示す。

Mimblewimble では、CT を利用することにより送金者も受領者もアドレスを使う必要が無くなるため、送金トランザクションは常に追跡困難性を満たす。

1.2.3 Cut-Through 手法

CT では、トランザクションに含まれる全ての PC の和が 0 になるため、ブロックの全てのトランザクションに含まれる PC を全て足し合わせても和が 0 になる。このとき、図 2 に示すように、あるトランザクションの output である PC が既に異なるトランザクションの input として使われている場合、両者の和が 0 のため、両方の PC を相殺することができる。その結果、ブロックサイズを削減することができる。

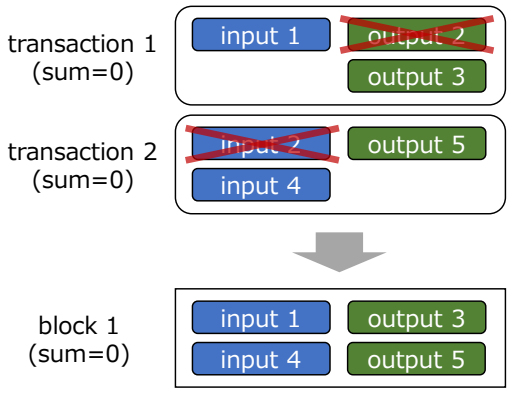

図 2　ブロックレベルの Cut-Through 手法

また、CT では、ブロックに含まれる全ての PC の和が 0 になるため、ブロックチェーン全体で全てのブロックに含まれる PC を全て足し合わせても和が 0 になる。ブロックチェーン上の全ての PC を集約したデータを「チェーンステート」と呼ぶが、チェーンステート内の全ての PC の和が

0 になるということは、図 3 に示すように input と output の PC を相殺できるだけでなく、チェーンステート内の PC を改竄できないことを示す。従って、Mimblewimble では、ブロックチェーンの整合性を検証するためにブロックチェーン全体を保存しておく必要はなく、チェーンステートだけを保持しておけば良い。

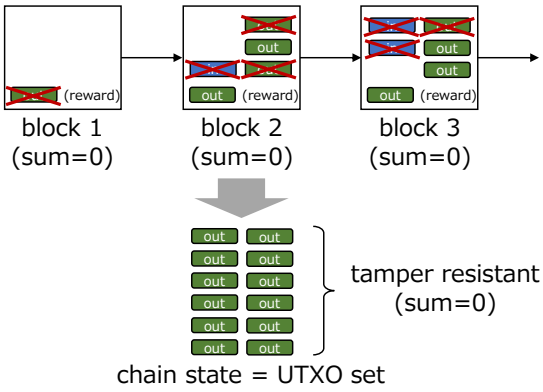

図 3　チェーンステートレベルの Cut-Through 手法

Bitcoin のスケーラビリティの制約要因の一つは二次記憶の容量と言われているが [8]、Mimblewimble の場合はチェーンステートのサイズを Cut-Through 手法により大幅に削減できる上に、チェーンステートさえ保持できれば台帳の整合性をチェックできるため、スケーラビリティを大幅に改善できる余地が残されている。

1.2.4　他の暗号資産との比較

Mimblewimble の主要な特性を他のいくつかの暗号資産と比較すると、表 1 のようになる。

まず、CryptoNote や Zerocash といった他の匿名暗号資産と比較して、Mimblewimble のスケーラビリティの改善余地が圧倒的に大きいことが分かる（ブロックチェーンサイズが小さいほどスケーラビリティの改善余地は大きい）。ちなみに、Grin のホワイトペーパー [5] によると、チェーンステートサイズで比較した場合、最大で Bitcoin の 1/300 程度まで削減できる可能性がある。

また、Bitcoin と比較した場合、Mimblewimble の方が追跡困難性が高く、かつ、スケーラビリティの改善余地も大きくなっており、両指標はトレードオフの関係になっていない。

1.3　Mimblewimble の課題

本節では、Mimblewimble のメッセージングに関する課題を整理する。

1.3.1　受領者側の処理

1.2.2 節で見たように、CT では有効なトランザクションを構築するために送金者が受領者に秘密情報を開示する必要がある。

	追跡困難性	ブロックチェーンサイズ [9]
Bitcoin[10]	-	1
CryptoNote[11][12] (Monero[13])	✔	24.10
Zerocash[14] (Zcash[15])	✔	8.64
Mimblewimble (Beam)	✔	0.36

表 1　暗号資産の特性比較

Bitcoin の場合、送金者がトランザクションに署名してネットワークにトランザクションを送信すればトランザクションが成立するため、受領者側がアクティブでなくても良いのに対し（図 4）、Mimblewimble の場合、受領者は秘密情報を受信し新しい outputs を生成してトランザクションに署名をする必要があるため、受領者側がアクティブでなければならない（図 5）。従って、送金者は送金の度に受領者との間で何かしらのセキュアなメッセージング・チャネルを開設する必要がある。

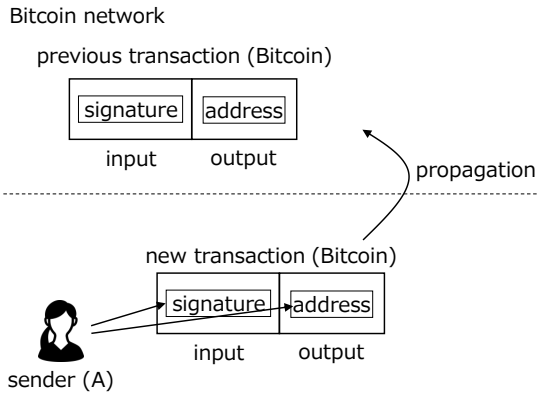

図 4　Bitcoin のトランザクション生成

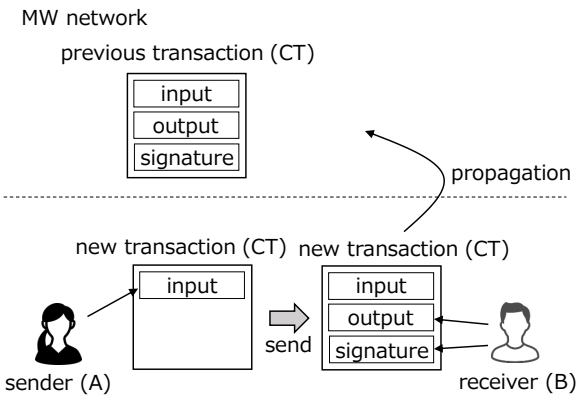

図 5　Mimblewimble のトランザクション生成

1.3.2 Beam の解決方法と課題

Beam のコミュニティでは、Mimblewimble のメッセージング・チャネルの課題を解決するために、Secure Bulletin Board System(SBBS)[16] と呼ばれるセキュアな掲示板システムを提案している（図 6）。

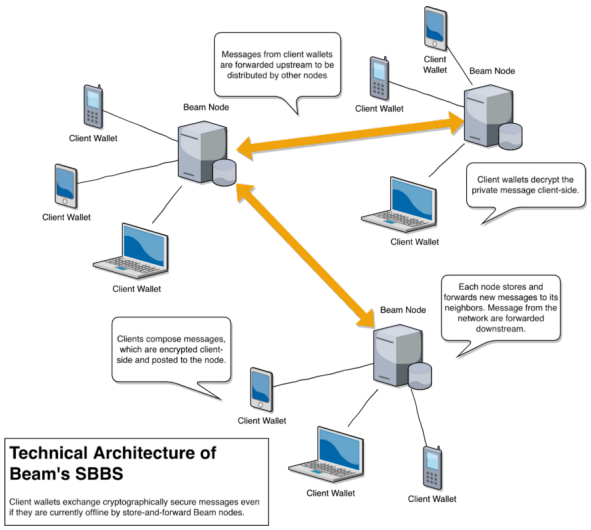

図 6　Beam のセキュア掲示板システム (SBBS) [16]

しかし、SBBS は次に挙げるように、多くの課題を持つ。

- **追跡困難性**
  ホワイトペーパーには、「クライアント（ウォレット）はノードが新しいメッセージを受け取った場合に通知が受け取れる」と書かれている。これは、取りも直さずメッセージングシステム側がメッセージの受信者を識別していることを意味しており、追跡困難性を満たしているとは言えない。仮に SBBS が追跡困難性を満たせないとすると、Mimblewimble 自身が持つ追跡困難性の意味が無くなる。
- **メッセージ転送インセンティブ**
  ホワイトペーパーには、「メッセージの転送は Beam のノードが行う」と書かれているが、いくら Beam のノードを使用すると言っても、手数料を付与する仕組みが無いのであれば、メッセージ転送コストを Beam ノードがボランティア的に負担せざるを得ず、システムが将来に亘って安定的に運用される保証は無い。
- **スケーラビリティ**
  Beam ノードにメッセージの保管を義務付けるのであれば、ブロックチェーンサイズが小さいという Mimblewimble の折角のメリットを打ち消すことになりかねない。これは、Mimblewimble のスケーラビリティに影響を及ぼすことを意味する。
- **開発コスト**
  SBBS が自力で追跡困難性を確保するためには、ゼロ知識証明のような暗号化技術や Tor[17] のような匿名化技術を用いる必要がある。これは、場合によっては非常に大規模な開発コストを要する。

2. MWmessage の提案

本章では、1.3 節で指摘した Mimblewimble のメッセージングの課題を解決するために、Mimblewimble 自身を拡張して追跡困難なブロックチェーン形式のメッセージング機構を実現する MWmessage を提案する。

2.1 MWmessage トランザクション

MWmessage のトランザクションは、従来の Mimblewimble のトランザクションとメッセージ部（サイズ、有効期限、ペイロード）から構成される。メッセージ部が改竄されないことを保証するために、メッセージハッシュ（r_h）を計算した上で、剰余値を計算する前に新しい PC（r_hG）を Mimblewimble トランザクションに追加しておく（従来の CT の仕様を修正せずに実現することが可能）。ただし、メッセージ部が存在しない場合は $r_hG = 0$ を追加する仕様とする（省略は不可）。

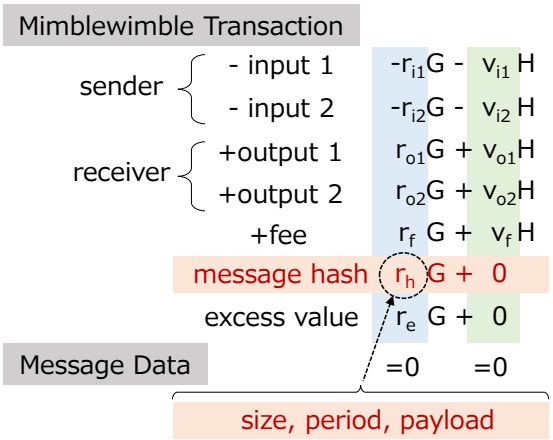

図 7　MWmessage トランザクション

メッセージを送信する場合、メッセージの内容を保証するためには、送金時と異なり、メッセージの送信者側が剰余値を計算する必要がある。また、MWmessage は Mimblewimble の送金を補助するためのプロトコルであるから、MWmessage トランザクション自体は送金を行えないと考えるのが自然である。従って、MWmessage ではメッセージ送信者が単独でトランザクションの構築を行って Mimblewimble のネットワークにトランザクションを送信する仕様とする。このとき、メッセージ送信者は Mimblewimble トランザクションの outputs に手数料のみを設定する。なお、従来の Mimblewimble の送金トランザクションでは、従来通り受領者側が剰余値を計算する仕様のため、今回の拡張で受領者側がメッセージを乗せることが可能になる。その際のメッセージの内容は、送金トランザクションとは関係ないもの（例えば、次回の別の送金のための piggyback メッセージ）でも問題ない。

2.2 MWmessage ブロック

MWmessage ブロックは、従来の Mimblewimble ブロックに 2.1 節で新規に追加されたメッセージ部を集約したメッセージブロックを新規に追加し拡張されたブロック全体を指す（図 8 の白い部分は従来の Mimblewimble ブロック、白い部分と赤い部分を合わせた領域全体が MWmessage ブロック）。

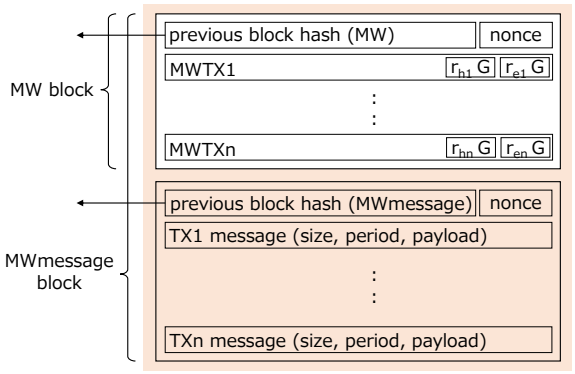

図 8　MWmessage ブロック

2.3 並列ハッシュチェーン

MWmessage は、図 9 のように、Mimblewimble ブロックのハッシュ値をつなぐ従来のハッシュチェーンに加えて、MWmessage ブロックのハッシュ値をつなぐ新しいハッシュチェーンを持つ（並列ハッシュチェーン）。マイナーは、Mimblewimble ブロック内の従来の nonce の値をマイニングした後、MWmessage ブロック内の新しい nonce の値をマイニングして初めて、Mimblewimble のネットワークに MWmessage ブロックを送信することができる。

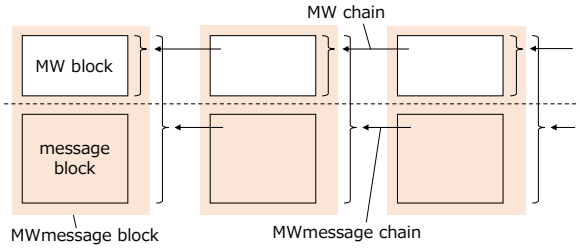

図 9　並列ハッシュチェーン

2.4 メッセージの有効期限

MWmessage は、Mimblewimble のスケーラビリティを犠牲にすることなく追跡困難なメッセージングを実現するために、メッセージに有効期限を付与する仕組みを提供する。有効期限はメッセージ送信者側が決定し、MWmessage トランザクションのメッセージ部の中に含める（2.1 節）。全てのメッセージに有効期限を付与することにより、どのメッセージブロックも将来のいずれかの時点で破棄することが可能になり（図 10）、マイナーが長期的に保存すべきメッセージのデータ量を大幅に削減できる。

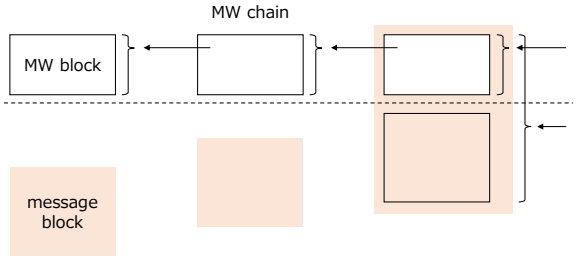

図 10　メッセージブロックの破棄

2.5 MWmessage ブロックの整合性検証

メッセージブロックを破棄することによって、MWmessage ブロックのハッシュチェーンをジェネシスブロックから辿って検証することが不可能になる。そのため、MWmessage ではマイナーが一定のブロック高毎に MWmessage ブロックのハッシュ値を Bitcoin 上に記録する。破棄されていない直近の MWmessage ブロックの整合性は、Bitcoin 上に記録された MWmessage ブロックのハッシュ値を起点にして検証を行う。

2.6 MWmessage の手数料

MWmessage トランザクションのサイズには、Mimblewimble トランザクションのサイズに加えて、メッセージ部のサイズを含む。MWmessage のマイナーは、MWmessage ブロックに含まれる MWmessage トランザクションのサイズの合計が、既定のサイズ（MWmessage ブロックサイズ）に収まるようにマイニングすることを要求される。マイナーはこの規約の元で自身が得られる手数料を最大化しようとするため、メッセージ送信者には、メッセージサイズに比例した適切な手数料を設定することが要求される。

2.7 抽象ブロックサイズ

MWmessage の利用者としては、メッセージの有効期限が長いほど利便性が高まる。一方、MWmessage のマイナーは、メッセージの有効期限が短いほどメッセージブロックを早期に破棄できるため、メッセージブロックを保存するコストを削減できる。

そこで、メッセージの有効期限をデータサイズのオーダに換算し、MWmessage トランザクションのサイズに含めてブロックサイズを計算することにより（抽象ブロックサイズ）、メッセージの有効期限が必要に応じた適切な長さに抑制されるようになる。

2.8 ペイロードの暗号化

MWmessage トランザクションのペイロードに含めるデータの内容は基本的にはメッセージ送信者が自由に決めることができるが、信頼できる公開鍵認証基盤 (PKI) を利用して送信者が送信メッセージに対して行った署名を追加した上で、受信者の公開鍵で暗号化したものをペイロードとすることによって、追跡困難性を満たすセキュアな匿名メッセージングを実現することができる（図 11）。ただし、メッセージ受信者は、MWmessage 上の全てのメッセージを復号化し、自身に対するメッセージが含まれているか否かを確認する必要がある。

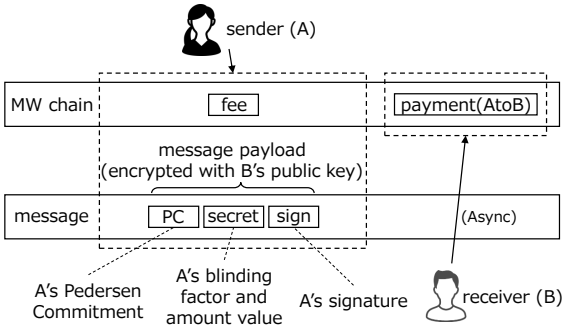

図 11 ペイロードの暗号化

3. MWmessage の机上評価

MWmessage は、Mimblewimble プロトコルに必要な追跡困難メッセージングを、Mimblewimble 自身のネットワークを利用して実現する技術である。この発想により、1.3.2 節で挙げた従来技術の課題のうち、追跡困難性、インセンティブ、開発コストの課題を同時に解決することができる。

本章では、更に、MWmessage を一般の匿名メッセージングシステムと見なした場合のセキュリティ、プライバシー、スケーラビリティの机上評価を行う。

Unger ら [18] は、これまでに世の中で提案されてきた匿名メッセージングシステムのセキュリティ/プライバシーを Trust Establishment、Conversation Security、Transport Privacy の 3 つの技術領域に分類し体系的に比較評価した。我々は、3.1 節、3.2 節において、MWmessage を一般の匿名メッセージングシステムと見なした場合のセキュリティ/プライバシーを、[18] の Conversation Security、Transport Privacy の評価基準に基づいて評価する。3.3 節では、MWmessage のユーザビリティとアドプションについて評価する。また、3.4 節では、MWmessage のスケーラビリティについて評価する。

3.1 セキュリティ

MWmessage は、ペイロードの暗号化（2.8 節）により、機密性（Confidentiality）、Message Unlinkability を実現している。また、ブロックチェーンで実現しているため、完全性（Integrity）と可用性（Availability）、Speaker Consistency、Causality Preserving、Global Transcript を満たしている。また、公開鍵認証基盤 (PKI) を利用することにより、認証情報の共有（Authentication）、Participant Consistency、Destination Validation、Message/Participation Repudiation も満たすことができる。

評価結果を一般の匿名メッセージングシステムと比較すると、図 12 のようになる。

3.2 プライバシー

MWmessage はブロックチェーンのため、受領者の匿名性（Recipient Anonymity）、Participation Anonymity、追跡困難性（Unlinkability）を満たしている。また、ペイロードの暗号化（2.8 節）により、Anonymity Preserving を実現している。なお、[18] における匿名メッセージングシステムの追跡困難性（unlinkability）の定義は、1.1 節で定義した暗号資産の場合の定義と異なるが、論文の評価結果との比較のために、本節では [18] の定義（「2 つの異なるメッセージが、当事者以外からは同一の参加者によって交わされたものであるか否かを推定することが困難であること」）に従った。

評価結果を一般の匿名メッセージングシステムと比較すると、図 13 のようになる。MWmessage はブロックチェーンのため、Message Broadcast のカテゴリに分類される。

3.3 ユーザビリティ・アドプション

[18] では、セキュリティ/プライバシーのトレードオフの指標の一つとしてユーザビリティ（使い易さ）・アドプション（導入の容易さ）も評価している。

MWmessage のメッセージには有効期限が付与されているため、非同期受信（Asynchronicity）を満たすことができない。メッセージを受信し逃した場合、システム側では自動的に再送を行わないため、消失メッセージ再送（No Message Drops）も満たさない。また、ブロックチェーンの場合は先に送信されたメッセージを全て受け取らないとメッセージの整合性を確認できないため、メッセージ欠落耐性（Dropped Message Resilient）は無いと判断した。

また、MWmessage は手数料を前提にしているため、手数料不要（No Fees Required）も満たさない。ただし、ブロックチェーン特有の手数料に関しては、メッセージ中継者へのインセンティブ付与に基づく通信品質の向上、スパム対策などのメリットも存在する。

3.4 スケーラビリティ

まず、ブロックチェーンの場合、メッセージを全ての参加者が共有するため、一般的な匿名メッセージングシステムと比較するとスケーラビリティが劣る。しかし、MWmessage

Scheme	Example	Security and Privacy															Adoption					Group Chat				
		Confidentiality	Integrity	Authentication	Participant Consistency	Destination Validation	Forward Secrecy	Backward Secrecy	Anonymity Preserving	Speaker Consistency	Causality Preserving	Global Transcript	Message Unlinkability	Message Repudiation	Particip. Repudiation	Out-of-Order Resilient	Dropped Message Resilient	Asynchronicity	Multi-Device Support	No Additional Service	Computational Equality	Trust Equality	Subgroup Messaging	Contractable	Expandable
TLS+Trusted Server†*	Skype	-	-	-	-	-	-	-	-	-	-	●	●	●		●	●	●	●	-	●	●	●	◐	◐
Static Asymmetric Crypto†*	OpenPGP, S/MIME	●	●	●	-	-	-	-	●	-	-	-	-	-	-	●	●	●	●	●					
+IBE†	Wang et al.	-	●	●	-	-	-	-	●	-	-	-	-	-	-	●	●	●	●	-					
+Short Lifetime Keys	OpenPGP Draft	●	●	●	-	-	◐	◐	●	-	-	-	-	-	-	●	●	●	●	-					
+Non-Interactive IBE†	Canetti et al.	●	●	●	-	-	●	-	●	-	-	-	-	-	-	◐	●	●	●	●					
+Puncturable Encryption†	Green and Miers	●	●	●	-	-	●	-	●	-	-	-	-	-	-	●	●	●	●	●					
Key Directory+Short Lifetime Keys†	IMKE	●	●	◐	-	●	◐	◐	-	-	-	-	●	●	●	●	●	-	-	-					
+Long-Term Keys†	SIMPP	●	●	◐	-	●	◐	◐	-	-	-	-	●	●	-	●	●	-	-	-					
Authenticated DH†*	TLS-EDH-MA	●	●	●	●	●	◐	◐	●	-	-	●	●	●	◐	●	●	-	-	●					
+Naïve KDF Ratchet*	SCIMP	●	●	●	●	●	●	◐	●	◐	-	●	●	●	◐	◐	◐	-	-	●					
+DH Ratchet†*	OTR	●	●	●	●	●	◐	●	●	◐	◐	-	●	●	◐	◐	◐	-	-	●					
+Double Ratchet†*	Axolotl	●	●	●	●	●	●	●	●	◐	◐	-	●	●	◐	●	◐	-	-	●					
+Double Ratchet+3DH AKE†*	-	●	●	●	●	●	●	●	◐	◐	◐	-	●	●	●	●	◐	-	-	●					
+Double Ratchet+3DH AKE+Prekeys†*	TextSecure	●	●	●	●	●	●	●	-	◐	●	-	●	●	●	◐	◐	●	-	-					
Key Directory+Static DH+Key Transport†	Kikuchi et al.	●	●	-	-	●	◐	◐	-	-	-	-	●	●	-	●	●	●	-	-	-	-	-	●	●
+Authenticated EDH+Group MAC†	GROK	●	●	◐	-	●	◐	◐	●	-	-	●	●	●	-	●	●	●	-	-	-	-	-	●	●
GKA+Signed Messages+Parent IDs†	OldBlue	●	●	●	●	●	◐	◐	●	●	-	-	-	-		●	●	◐	-	●	●	●	-	-	-
Authenticated MP DH+Causal Blocks†*	KleeQ	●	●	◐	◐	◐	●	●	●	◐	◐	●	-	●	●	●	●	◐	-	●	●	●	-	-	●
OTR Network+Star Topology†	GOTR (2007)	●	●	-	-	-	◐	●	-	-	-	-	●	●	●	◐	●	◐	-	●	-	-	-	●	●
+Pairwise Topology†		●	●	●	●	●	◐	●	●	-	-	-	●	●	●	◐	●	◐	-	●	●	●	●	●	●
+Pairwise Axolotl+Multicast Encryption*	TextSecure	●	●	●	-	●	●	●	-	●	●	-	●	●	●	●	●	●	-	-	●	●	●	●	●
DGKE+Shutdown Consistency Check†	mpOTR	●	●	●	●	●	◐	◐	●	◐	-	-	-	●	●	●	●	-	-	●	●	●	-	-	-
Circle Keys+Message Consistency Check†	GOTR (2013)	●	●	●	●	●	◐	◐	●	●	●	●	●	●	●	●	-	-	-	◐	●	●	-	●	●
Blockchain	MWmessage	●	●	●	●	●	-	-	●	●	●	●	●	●	●	●	-	-	●	●					

図 12　匿名メッセージングシステムの Conversation Security（[18] の TABLE II を修正）

Scheme	Example	Privacy					Usability						Adoption								
		Sender Anonymity	Recipient Anonymity	Particip. Anonymity	Unlinkability	Global Adv. Resistant	Contact Discovery	No Message Delays	No Message Drops	Easy Initialization	No Fees Required		Topology Independent	No Additional Service	Spam/Flood Resistant	Low Storage	Low Bandwidth	Low Computation	Asynchronous	Scalable	
Store-and-Forward†*	Email/XMPP	-	-	-	-	-	●	◐	●	●	●		●	-	-	●	●	●	●	●	
+DHT Lookup†*	Kademlia	◐	◐	-	-	-	●	◐	●	●	●		●	●	◐	●	◐	●	●	●	
Onion Routing+Message Padding†*	Tor	●	-	●	●	-	-	◐	●	●	●		●	◐	-	●	●	●	-	●	
+Hidden Services*	Ricochet	●	●	●	◐	-	-	◐	●	●	●		●	◐	-	●	●	●	-	●	
+Inbox Servers†	-	●	-	●	●	-	-	◐	●	●	●		●	-	-	●	●	●	●	●	
+Random Delays†*	Mixminion	●	-	●	●	◐	-	-	●	●	●		●	-	-	◐	●	●	●	●	
+Hidden Services+Delays+Inboxes+ZKGP*	Pond	●	-	●	●	◐	-	-	●	●	●		●	-	●	◐	●	●	●	●	
DC-Nets†*	-	●	●	-	-	●	-	-	●	●	●		-	●	-	●	●	●	-	-	
+Silent Rounds†	Anonycaster	●	●	-	-	●	-	-	●	●	●		-	●	◐	●	●	●	-	-	
+Shuffle-Based DC-Net+Leader†	Dissent	●	●	-	-	●	-	-	●	●	●		-	●	●	●	●	●	-	-	
+Shuffle-Based DC-Net+Anytrust Servers†	Verdict	●	●	-	-	●	-	-	●	●	●		-	-	●	●	●	●	-	◐	
Message Broadcast†	-	-	●	●	●	●	●	●	●	●	●		●	●	-	-	-	◐	-	-	
+Blockchain	-	◐	●	●	●	●	●	-	-	●	-		●	◐	●	-	-	-	●	-	
	MWmessage	●	●	●	●	●	-	-	-	●	-		●	-	●	●	-	-	-	-	
PIR*	Pynchon Gate	-	●	●	●	●	●	-	●	◐	●		●	-	-	-	◐	◐	●	◐	

図 13　匿名メッセージングシステムの Transport Privacy（[18] の TABLE III を修正）

の場合は、メッセージへの有効期限の付与により、メッセージブロックを破棄できるため、図 13 の Low Storage の項目が改善される。

次に、MWmessage は、他のブロックチェーンとの比較において、Mimblewimble が持つスケーラビリティの優位性を維持できることを示す。なお、ここでは送金機能を含めた比較となるため、Mimblewimble の元々のブロックチェーンを含めてブロックチェーンサイズを計算する。

まず、メッセージブロックは長期的な観点では直近に生成されたものしか残らないため、ブロックチェーンサイズの見積もりには含めない。すると、MWmessage のブロックチェーンサイズに含まれるものは、Mimblewimble のチェーンステートのみとなる。

MWmessage の拡張に伴い、Mimblewimble トランザクション一つにつきメッセージハッシュに相当する PC が一つ増加する。この PC の個数は剰余値の個数と同じだから、仮に元々の Mimblewimble のチェーンステートにおける剰余値の割合が多めに見積もって 50%だとすると、Mimblewimble のチェーンステートのサイズは従来の 1.5 倍に増加する。また、Mimblewimble トランザクション数自体も倍増するため、Mimblewimble のチェーンステートのサイズは単純計算で従来の 3 倍に増加することになる。

しかし、Mimblewimble のチェーンステートのサイズは最大で Bitcoin の 1/300 程度まで削減できる可能性があることを考慮すると、仮に MWmessage のオーバヘッドが 3 倍だとしても、依然として Bitcoin と比較して 100 倍近くスケーラビリティを改善できる余地が残る。

3.5 総評

MWmessage は、セキュリティ/プライバシーに関しては概ね一般の匿名メッセージングシステムと同等の特性を備えていると言える。一方で、メッセージに有効期限を付与することにより、ユーザビリティ・アドプションの一部を犠牲にしている点は否めないが、スケーラビリティに対するトレードオフであるため、やむを得ない仕様と言える。

4. 結論

本稿では、Mimblewimble の特徴と課題を整理した上で、Mimblewimble 自身を拡張して追跡困難なブロックチェーン形式のメッセージング機構を実現する MWmessage を提案した。MWmessage は、Mimblewimble のメッセージングにおける追跡困難性、インセンティブ、開発コストの課題を同時に解決できる。また、MWmessage は、メッセージに有効期限を付与することで、Mimblewimble のスケーラビリティを犠牲にすることなく追跡困難なメッセージングを実現できる。従って、将来的に MWmessage は Mimblewimble プロトコルの重要な拡張機能として利用されることが期待される。

参考文献

[1] Reid, F. and Harrigan, M.: An Analysis of Anonymity in the Bitcoin System, *in Security and Privacy in Social Networks*, pp. 197–223 (online), DOI: 10.1007/978-1-4614-4139-7 (2013).

[2] Ron, D. and Shamir, A.: Quantitative Analysis of the Full Bitcoin Transaction Graph, *in Proc. 17th International Conference, Financial Cryptography and Data Security(FC'13)*, pp. 6–24 (2013).

[3] Poelstra, A.: Mimblewimble, `https://download.wpsoftware.net/bitcoin/wizardry/mimblewimble.pdf` (2016).

[4] Beam community: BEAM: THE SCALABLE CONFIDENTIAL CRYPTOCURRENCY, `https://docs.beam.mw/BEAM_Position_Paper_v0.2.3.pdf` (2018).

[5] Grin community: Introduction to MimbleWimble and Grin, `https://github.com/mimblewimble/grin/blob/master/doc/intro.md` (2019).

[6] Maxwell, G.: Confidential Transactions, `https://people.xiph.org/~greg/confidential_values.txt` (2016).

[7] Pedersen, T. P.: Non-Interactive and Information-Theoretic Secure Verifiable Secret Sharing, *in Proc. 11th Annual International Cryptology Conference(CRYPTO'91)*, pp. 129–140 (1992).

[8] Poon, J. and Dryja, T.: The Bitcoin Lightning Network: Scalable Off-Chain Instant Payments, `https://lightning.network/lightning-network-paper.pdf` (2016).

[9] Beam Privacy: What's the difference between Monero, Zcash, and BEAM?, `https://medium.com/beam-mw/whats-the-difference-between-monero-zcash-and-beam-953eafd89354` (2018).

[10] Nakamoto, S.: Bitcoin: A Peer-to-Peer Electronic Cash System, `https://bitcoin.org/bitcoin.pdf` (2008).

[11] Saberhagen, N. V.: CryptoNote, `https://cryptonote.org/whitepaper.pdf` (2013).

[12] Noether, S., Mackenzie, A. and the Monero Research Lab: Ring Confidential Transactions, *Ledger*, Vol. 1, pp. 1–18 (online), DOI: 10.5195/ledger.2016.34 (2016).

[13] The Monero Project: Monero github, `https://github.com/monero-project/monero` (2019).

[14] Ben-sasson, E., Chiesa, A., Garman, C., Green, M., Miers, I., Tromer, E. and Virza, M.: Zerocash : Decentralized Anonymous Payments from Bitcoin, *in Proc. 35th IEEE Symposium on Security and Privacy(S&P'14)*, pp. 459–474 (online), DOI: 10.1109/SP.2014.36 (2014).

[15] Zcash Community: Zcash github, `https://github.com/zcash/zcash` (2019).

[16] Lahat, R.: The Secure Bulletin Board System (SBBS) implementation in Beam, `https://medium.com/beam-mw/the-secure-bulletin-board-system-sbbs-implementation-in-beam-a01b91c0e919` (2018).

[17] Dingledine, R., Mathewson, N. and Syverson, P.: Tor : The Second-Generation Onion Router, *in Proc. 13th USENIX Security Symposium(USENIX Security '04)* (2004).

[18] Unger, N., Dechand, S., Bonneau, J., Fahl, S., Perl, H., Goldberg, I. and Smith, M.: SoK: Secure Messaging, *in Proc. 36th IEEE Symposium on Security and Privacy(S&P'15)*, Vol. 2015-July, pp. 232–249 (online), DOI: 10.1109/SP.2015.22 (2015).