軽量匿名通信プロトコルにおけるパス完全性と検証法に関する一考察

河内山 深央[1,a)]　吉仲 佑太郎[1,b)]　武政 淳二[1,c)]　小泉 佑揮[1,d)]　長谷川 亨[2,e)]

概要：インターネットにおける国家や企業による監視が一般的になるのに伴い，ユーザのプライバシーを保護する重要性が増している．これに対して，ネットワーク層で匿名性を提供する軽量匿名プロトコル PHI は，高性能でインターネット全体に展開可能な有望なプロトコルである．一方，パケットが指定した経路を通過したことを保証できない，すなわち，経路完全性を提供できないため，経路上の悪意あるルータがパケットの経路を変更する攻撃に脆弱である．本稿では，PHI に経路検証を追加することで，経路完全性を提供する経路匿名プロトコルを設計する．設計では，経路検証の際に，経路上のルータに経路長などの匿名性の消失に繋がる情報を漏洩させないことが鍵である．

キーワード：軽量匿名プロトコル，経路完全性，経路検証

A study on path integrity and verification for a light weight anonymity protocol

Mio Kochiyama[1,a)]　Yutaro Yoshinaka[1,b)]　Junji Takemasa[1,c)]　Yuki Koizumi[1,d)]　Toru Hasegawa[2,e)]

Abstract: Protecting anonymity of Internet users is becoming increasingly important as pervasive monitoring is becoming popular on the Internet. A lightweight anonymity protocol like Path-Hidden Lightweight Anonymity Protocol (PHI) is promising to universally provide relationship anonymity. PHI provides well-balanced anonymity and performance; however, it does not provide path integrity, where packets were actually forwarded on an intended path. Due to its weak integrity mechanism, PHI is not resilient against path modifying attacks, which leads to anonymity violation. This paper integrates path validation to PHI so as to be resilient against path modifying attacks. An important research challenge is how anonymity is protected against nodes on the path during the path validation.

Keywords: Lightweight Anonymity Protocol, Path Integrity, Path Validation

1. はじめに

今日のインターネットでは，国家や企業などが様々な目的で監視を行っており [5]，ユーザのプライバシーを守るためには，匿名通信が有用である．軽量匿名通信プロトコル PHI/dPHI [1,4] は，単一の自律システム (AS) あるいはルータのみが攻撃者となる，弱いが現実的な攻撃者を仮定し，パケットのヘッダのみを暗号化することで，軽量さと高速性を実現している．具体的には，送信者が，経路上の各ルータの前ホップと次ホップのアドレス対（以降，**転送状態**と呼ぶ）のリストをパケットヘッダに持たせ，各ルータが転送状態を復号して次ホップを決定することで，宛先

[1] 大阪大学 大学院情報科学研究科
〒 565-0871 大阪府吹田市山田丘 1-5
Graduate School of Information Science and Technology, Osaka University
1-5 Yamadaoka, Suita, Osaka 565-0871 Japan
[2] 島根大学 材料エネルギー学部
〒 690-8504 島根県松江市西川津町 1060
Faculty of Materials for Energy, Shimane University
1060 Nishikawatsu-cho, Matsue, Shimane 690-8504 Japan
[a)] m-kochiyama@ist.osaka-u.ac.jp
[b)] y-yoshinaka@ist.osaka-u.ac.jp
[c)] j-takemasa@ist.osaka-u.ac.jp
[d)] ykoizumi@ist.osaka-u.ac.jp
[e)] t_hasegawa@mat.shimane-u.ac.jp

を知られることなくパケットを転送する．

PHI/dPHI では，転送状態リストに従って正しく転送されるように，転送状態に対してメッセージ認証コード (Message Authetication Code タグ：MAC タグ）を付加して，上流ルータが転送状態を不正に書き換えたことを検出できる．全ルータが PHI/dPHI を実装することにより，リンクには必ず前ホップあるいは次ホップのルータが接続されていることを仮定する．この結果，各転送状態が書き換えられていないことを検証することで，転送状態リストに従って転送されたことを保証している．以下では，指定した経路に従って転送されることを**経路完全性**と呼ぶ．

一方，全ルータが PHI/dPHI を実装する仮定は現実的でなく，その普及には，IP ネットワークにオーバレイさせることが現実的である．しかしながら，IP にオーバレイさせると，転送状態の前ホップと次ホップは IP アドレスで指定することになる．PHI/dPHI の仮定が成立せず，受信したパケットは転送状態に記載された前ホップから転送されたことが保証されないため，経路完全性が保証されなくなる．

本研究では，軽量匿名通信プロトコルとして dPHI を採用し，経路完全性を提供できるように拡張する．具体的には，既存の経路検証プロトコル [2,3,6–8] の内，E-OPT [6] を組み込むことで，関係匿名性と経路完全性を両立する．

本研究における挑戦は，経路上のルータに対して，経路長などの送信者の匿名性に繋がる経路情報を漏洩させない経路検証プロトコルを設計することであり，本研究の貢献は以下の通りである．

- 経路検証において，ルータの身元，具体的には証明書を用いない MAC 鍵交換を実現した．
- 経路検証の基礎となる，MAC 連鎖の計算，検証において，検証数，すなわち経路長が漏洩しない検証を実現した．
- dPHI に対応して 2 つのサブ経路に分割した経路を検証できるように E-OPT を拡張した．

本論文の構成は，以下の通りである．2 章で dPHI 及び E-OPT を概観し，3 章でシステムモデルとゴールを説明する．4 章で設計根拠を示し，5 章でプロトコルを設計する．6 章で性能を解析し，7 章で関連研究を概観する．8 章で本論文を纏める．

2. dPHI と E-OPT

本章では dPHI と E-OPT について説明する．表 1，表 2，表 3 はそれぞれ，システム，dPHI，E-OPT で使用する記号を定義している．本稿では，セッション鍵や MAC 鍵の交換に ECDH 鍵交換を使用するため，ECDH 鍵交換及び ECDH 公開鍵を，単に鍵交換，公開鍵と呼ぶこととする．

2.1 軽量匿名通信プロトコル dPHI

dPHI [6] は，ソースルーティングベースで，盗聴者にソース S とデスティネーション D を紐づけさせない関係匿名性を提供する．経路設定フェーズとデータ転送フェーズからなり，経路設定フェーズでは，S から D に経路を設定する．経路は，ルータ R_i の秘密鍵 k_{R_i} で暗号化された転送状態のリストで表現され，V^1 と V^2 の 2 つのサブ経路から構成される．V^1 は S とミッドウェイルータ W 間の経路で，V^2 は W と D 間の経路であり，W は中継地点である．V^1 と V^2 は，ヘッダ上では長さ l の配列のフィールドとして保存するため，以降，V^1 と V^2 は，$V^1[l]$ と $V^2[l]$ を意味する．

表 1　本稿で使用する記号

記号	説明
S	送信者 (ソース)
D	受信者 (デスティネーション)
R_i	経路上の i 番目のルータ
IP_X	X の IP アドレス

表 2　dPHI で使用する記号

M	ヘルパー
W	ミッドウェイルータ
V^1	S から W までの経路の転送状態のリスト（配列)
V^2	W から D までの経路の転送状態のリスト（配列)
$H.midway$	V^1 を検証するためのハッシュ値
k_{R_i}	R_i が持つ秘密鍵で転送状態を暗号化
K_D^+	D の公開鍵
sid	セッション識別子
$k1_{S-D}$	S と D のセッション鍵でペイロードを暗号化
pk_S と sk_S	S がセッションを通して使用する鍵ペア

2.1.1 経路設定フェーズ

経路設定フェーズは，2 つのラウンドから構成される．

2.1.1.1 第 1 ラウンド

V^1 作成のために，S はヘルパーと呼ばれるサーバ M 宛にミッドウェイ要求パケットを送信する．以降は，経路設定に用いるパケットを，**経路設定パケット**あるいは単にパケットと呼ぶ．パケットには，S によって乱数で初期化した V^1 と，S と M が共有するセッション鍵で暗号化した受信者の IP アドレス IP_D を持つ．V^1 上のルータに D を知られないように，IP_D を暗号化してしている．M に向けて経路設定パケットは IP ルーティングに従って転送され，途中の各ルータは，自身がデータ転送フェーズで使用するための転送状態を作成し，V^1 に追加する．その後，M に向けてパケットを転送する．パケットが到達すると，M は復号した IP_D をパケットに書き込み，S にパケットを返送する．この時，作成した V^1 に従ってパケットを転送する．

パケットが W に到達すると，W は自身の秘密鍵 k_W で IP_D を暗号化し，IP_D と V^1 を使用して生成したハッシュ値 $H.midway$ をパケットに書き込む．これは，S から W の間で V^1 が改竄されていないことを，S が検証するためである．ここで,$H.x$ は，ヘッダのフィールド x を表す．最後に，パケットを受け取った S は $H.midway$ の値を検証し，

V^1 を保存する．

S と D はペイロード暗号化用のセッション鍵 $k1_{S-D}$ を ECDH により交換する．D の公開鍵 K_D^+ には，公開鍵証明書を持つ D の秘密鍵で証明書を持たせ，D の真正性を保証する．一方，S の公開鍵 pk_S はセッション識別子 sid のハッシュ値とすることで，偽造を困難としている．sid は，認証付き暗号を用いて暗号化した認証データとしている．

V^2 作成のため，S は V^1 と k_W で暗号化した IP_D をヘッダに書き込んで，W にパケットを送信する．V^1 に従って転送され，受信した W は V^2 を乱数で初期化し，k_W で復号した IP_D 宛にパケットを送信する．このパケットは IP ルーティングに従って転送され，W から D までの各ルータは，V^1 の場合と同様に V^2 を作成する．パケットを受信した D は，ヘッダに V^1 と V^2 を書き込み，これらを S と D のセッション鍵 $k1_{S-D}$ で暗号化してペイロードに書き込んでから，S に返送する．受信した S は，ペイロードに書き込まれた V^1 と V^2 を復号し，これが保存した V^1 とパケットヘッダ上の V^1 と V^2 と一致しているかを検証する．

2.2 データ転送フェーズ

S と D は，V^1, V^2 をヘッダに書き込んで，パケットを D と S に送信する．各ルータは V^1, V^2 から自身に対応する転送状態を読み出し，自身の鍵 k_{R_i} で復号することで，次ホップのアドレスを得る．転送状態は，前ホップと次ホップのアドレスの対と k_{R_i} で生成した MAC タグから構成されており，各ルータは自身の転送状態が改竄されていないことを検証する．

2.3 経路検証 Extended-OPT

表 3 経路検証で使用する記号

記号	説明
PVF_S	経路検証で使用する MAC 連鎖
OPV_X	S が計算する経路検証用の値
OPV'_X	X が計算する経路検証用の値
PVF_D	経路検証で使用する MAC 連鎖

Extended-OPT（E-OPT） [6] は，送信者が設定した経路に対して，パケットが実際に指定されたルータの順に転送された**経路完全性**を検証する (**経路検証**と呼ぶ)．送信者と各ルータ，受信者と各ルータがそれぞれ交換した MAC 鍵を用いたメッセージ認証コード（Message Authentication Code: MAC）の連鎖よる検証をベースとしている．パケットには，送信者と受信者の MAC 連鎖のフィールド PVF_S と PVF_D を持たせ，図 1 に示すように更新する．

2 つのフェーズから構成され，鍵交換フェーズで，S と D は，鍵交換プロトコルにより，経路上のルータと MAC 鍵を交換するとともに，S と D も MAC 鍵を交換する．以降，X と Y が交換した MAC 鍵を k_{X-Y} と表記する．

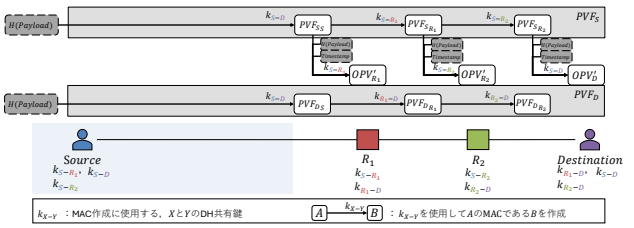

図 1 E-OPT

データ転送フェーズにおいて，S は PVF_S と PVF_D の初期値 PVF_{S_0}，PVF_{D_0} として，$MAC_{k_{S-D}}(H(P)||TS)$ をヘッダに書き込む．ここで，$H(P)$ はペイロードのハッシュ値，TS はパケットのタイムスタンプである．さらに，S は経路上の各ルータ $R_i (i = 1,,,n)$ が，パケットが S により作成されたことと，R_i まで経路上のルータ列を転送されてきたことを検証するための値 $OPV_i (i = 1,,,n)$ を全ルータ分，および D 用の OPV_D を，下記の式に従って，$i = 1,,,n$ まで繰り返し計算して，ヘッダに書き込む．ここで，n はルータ数である．

$$PVF_S \leftarrow PVF_{S_i} = MAC_{k_{S_i}}(H(P)||PVF_{S_{i-1}})$$
$$PVF_D \leftarrow PVF_{D_i} = MAC_{k_{E_i}}(H(P)||PVF_{D_{i-1}})$$
$$OPV_i \leftarrow MAC_{k_{S-R_i}}(PVF_{S_{i-1}}||H(P)||R_{i-1}||TS).$$

パケットを受信したルータ R_i は，MAC 連鎖 PVF_S と PVF_D を MAC 鍵 k_{S-R_i} と k_{D-R_i} で更新するとともに，S が書き込んだ OPV_i に相当する $OPV'_i \leftarrow MAC_{k_{S-i}}(PVF_S||H(P)||R_{i-1}||TS)$ を計算する．パケットが，ルータ R_{i-1} までの全ルータに到着し，正しく PVF_S を更新した場合，$OPV_i = OPV'_i$ が成立する．すなわち，$OPV_i = OPV'_i$ を検証することで，ここまでの経路検証を行い，$OPV_i \neq OPV'_i$ の場合，経路検証が失敗し，パケットを破棄する．

D はパケットを受信すると，PVF_S をセッション鍵 k_{S-D} で暗号化して S に返送する．S は受信した PVF_S と、自身で計算した PVF_{S_n} を比較し，一致しない場合，経路検証に失敗する．一方，D は受信した PVF_D と自身で計算した MAC 連鎖 PVF_{D_n} を比較する．さらに，受信した PVF_S を用いて計算した $OPV_D \leftarrow MAC_{k_{S-D}}(PVF_S||H(P)||R_{n-1}||TS)$ と自身の持つ MAC 鍵で作成した OPV'_D を比較する．どちらか一方が一致しない場合，経路検証が失敗する．以上のように，S と D は独立に経路検証を実施する．

3. 問題設定

本章では，システムならびに脅威モデルを述べてから，dPHI の脆弱性に対する pPHI の目標を述べる．

3.1 システムモデル

システムは, 送信者 S, 受信者 D, pPHI を実装する Anonymous Protocol（AP）ルータ，ミッドウェイルータ W，IP ルータ，ヘルパー M，PKI サービスから構成される．pPHI は複数の AS の協力で実行され，各 AS は少なくとも 1 つの AP ルータを持ち，AP ルータ同士は IP ルータで接続される．ヘルパーと PKI サービスは第三者機関によって運営される．表 4 及び表 5 に pPHI の定義にも用いる，記号及び暗号プリミティブを示す．

表 4 本プロトコルで使用する記号

記号	説明
V^{key1}	S から W までの経路の鍵交換用のフィールド
V^{key2}	W から D までの経路の鍵交換用のフィールド
pk_X	X がセッション単位で使用する ECDH 公開鍵
sk_X	X がセッション単位で使用する ECDH 秘密鍵
G	ECDH のベースポイント
x_{Si}	S が S から見て i 番目の AP ルータと共有する ECDH 秘密鍵
x_{Di}	D が D から見て i 番目の AP ルータと共有する ECDH 秘密鍵
$x_{R_{iS}}$	R_i が S と共有する ECDH 秘密鍵
$x_{R_{iD}}$	R_i が D と共有する ECDH 秘密鍵
k_{X-Y}	X と Y が共有する MAC 作成用の ECDH 鍵
$x2_{Si}$	S が S から見て i 番目の AP ルータと共有する ECDH 秘密鍵
$x2_{R_iS}$	R_i が S と共有する ECDH 秘密鍵
$k2_{S-X}$	S と X が共有する暗号化用の ECDH 鍵
PVF_{D1}	パス検証で使用する MAC チェーン
PVF_{D2}	パス検証で使用する MAC チェーン

表 5 本プロトコルで使用する暗号プリミティブ

記号	説明
$EncAE_{key}(text)$	認証付き共有鍵暗号の暗号化アルゴリズム
$DecAE_{key}(text)$	認証付き共有鍵暗号の復号アルゴリズム
$MAC_{key}(text)$	text に対する key を使用した MAC 作成アルゴリズム
$Enc_{key}(text)$	認証無し共有鍵暗号の暗号化アルゴリズム
$Dec_{key}(text)$	認証無し共有鍵暗号の復号アルゴリズム

3.2 脅威モデル

攻撃者として，アクティヴでローカルな攻撃者を想定する [1,4,11]. 攻撃者は、高々 1 つの AS のみ乗っ取ることが (ローカル) 可能であり，乗っ取った AS の AP ルータは，盗聴に加えて，パケットの変更，リプレイ，破棄など任意の攻撃が可能である．一方，送信者 S はオネストである．また，経路上のパーティ，すなわち AP ルータ，ヘルパー，受信者は結託しない．

3.3 dPHI の脆弱性：ルータスキップ攻撃

dPHI が AP ルータ同士が隣接するのに対して，pPHI は IP にオーバレイするため，AP ルータは送信元 MAC アドレスや IP アドレスを偽造可能である．攻撃者は，この脆弱性を用いて，経路上の AP ルータに気づかれることなく，下流の AP ルータをスキップできる (**ルータスキップ攻撃**).

攻撃者が乗っとった AP ルータは，S から D 宛のパケットを受信すると，次ホップの AP ルータをスキップし，その次ホップと推測する AP ルータにパケットを転送する．転送先の AP ルータは，インターネットのトポロジー情報などの補助情報を利用して，候補を推測する．推測が誤った場合は転送先の AP ルータで破棄されるが，偶然，正しい AP ルータに転送出来た場合，受信した AP ルータ以降は正しくパケットが転送される．すなわち，次ホップのルータのスキップに成功する．

転送したパケットが TCP のような双方向通信で使用されている場合，攻撃者の AP ルータは次ホップのスキップに成功したことを知る．以降，推測に成功した AP ルータの次ホップ以降のルータに対して，同様にルータスキップ攻撃を繰り返すことで，最終的に D に接続される AP ルータを知る．このように，D が存在可能な IP アドレス空間を狭めることが可能で，受信者匿名性が損なわれる．同様に，D から S へのルータスキップ攻撃が可能である．

3.4 達成目標

pPHI では，関係匿名性を維持しながら，ルータスキップ攻撃などの経路変更への耐性を持たせるため，以下を達成する．

- 関係匿名性：送信者を除く全てのパーティで送信者匿名性又は受信者匿名性の少なくとも一方を達成する．
- 経路完全性：パケットは，転送状態のリストに含まれる全てのオネストな AP ルータによって，正しい順で転送される．
- 経路長の秘匿性：関係匿名性に関わるような，パス長に関するいかなる情報も漏洩しない．

4. 設計根拠

匿名性を保ちながら E-OPT を dPHI に組み込むには，3 つの課題が存在する．第一に，受信者の経路検証において，受信者は経路上の AP ルータと MAC 鍵を交換することで，受信者に経路長が漏洩する．第二に，dPHI では，経路は 2 つのサブ経路 V^1、V^2 に分かれており，単純に E-OPT を適用できない．第三に，MAC 鍵交換において，中間者攻撃を防ぐための，AP ルータの公開鍵証明書を使用できない．これらの問題に対して，以下に示す通り，解決する．

4.1 モックルータによる経路長隠ぺい

受信者 D は，PVF_D の計算に全 AP ルータと MAC 鍵を

交換することで，D に AP ルータ数が漏洩する．これに対して，MAC 鍵数を常に一定にするため，S はダミーとして機能するモックルータを挿入する．一方，PVF_S は送信者 S が使用するため，MAC 鍵数を S が知っても問題ない．図 2 に，経路長 4 の場合の PVF_D の更新を示す．

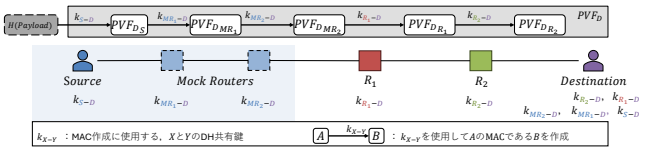

図 2　Mock Router

S は，$4-2=2$ 個のモックルータ分の D と共有する MAC 鍵による連鎖の計算を行ってから，最初のルータ R_1 に PVF_D を渡す．また，D との MAC 鍵交換においても，S が D とモックルータ数分の MAC 鍵を交換しておく．

4.2　転送状態のリスト毎の経路検証

経路は 2 つの転送状態リスト V^1 と V^2 であるため，MAC 連鎖 PFV_D は，V^1 と V^2 用に PFV_{D1} 及び PFV_{D2} を準備する．V^1 と V^2 の経路長を固定にするため，V^1 に対しては，図 2 に示すように，S から 1 ホップ目の AP ルータの前にモックルータを置く．一方，V^2 に対しても，V^2 の先頭であるミッドウェイルータ W の前にモックルータを置く．

分割した PVF_D で検証可能なのは，パケットが実際に通った経路に含まれるオネストな AP ルータ列の整列集合 T から任意の要素を抜き出して 2 つに分割したときに，それぞれが $L_1=\{R_1,\ldots,R_{p-1}\}(1\le p\le n)$，$L_2=\{R_{q+1},\ldots,R_n\}(1\le q\le n)$ と一致することのみである．集合 $\{R_p,\ldots,R_q\}$ は攻撃者に乗っ取られた AP ルータを表す．よってミッドウェイルータ W が含まれる AS が乗っ取られているとき，例えば $T=\{R_1,R_{q+1},R_2\ldots,R_{p-1},R_{q+2},\ldots,R_n\}$ の場合でも検証に成功する．これは，R_{p-1} が V^1 に，R_{q+1} が V^2 に属しているために R_{p-1} と R_{q+1} の順序関係が示せず，整列集合 T において $R_{p-1}\le R_{q+1}$ であることが検証できないためである．

これに対して，$R_{p-1}\le R_{q+1}$ であることを示すため，ミッドウェイルータ $R_w=W$ にパケットが到達した時点で，V^1 に含まれる R_w を除くオネストな AP ルータを全て通過したことを検証する方法を追加する．このため，S と V^1 上のルータは，MAC 鍵とは別に PVF_{D2} の初期値を暗号化するための鍵 $k2_{S-R_i}$ を共有する．PVF_{D2} の初期値は，V^1 を通過後，R_w に最初の MAC 演算が行われる．S は，$k2_{S-R_i}$ を用いて複数回 PVF_{D2} の初期値を暗号化し，V^1 上のルータはオニオンルーティングのように共有した鍵 $k2_{S-R_i}$ で復号する．正しく V^1 の AP ルータを通過した場合にのみ，R_w は正しい PVF_{D2} の初期値を受信する．この結果，V^1 に含まれるオネストな AP ルータを全て通過したことを検証できる．

この暗号化で満たすべき条件は 2 つである．第一は，送信者 S とルータ R_i のみが行える可逆反応であることである．他のルータが PVF_{D2} を復号できてしまうと通過していない経路を通過したと偽証できる．このため，送信者と $R_1,\ldots,R_{w-1}$ の共通鍵を使用する．第二は，攻撃者による有限回の試行によって情報が漏洩してはいけないことである．経路検証では，暗号化された PVF_{D2} の復号結果がパケットに格納されて転送されるため，攻撃者が指定した暗号文に対する平文が得られないように，選択暗号文攻撃への耐性が必要である．これに対して，PVF_{D2} の暗号文が改竄されていないこと検証するため，V^1 に含まれる AP ルータについて，OPV_X を利用したメッセージ認証を行う．すなわち，OPV_X, OPV'_X の計算に PVF_{D2} の値を使用するよう変更することにより，PVF_{D2} が改竄された場合は OPV_X と OPV'_X の値が一致せずにパケットを破棄する．この結果，PVF_{D2} の改竄を検出し，選択暗号文攻撃を防ぐ．

4.3　鍵交換

4.3.1　V^{key1}, V^{key2} を用いた鍵交換

pPHI では，パス設定フェーズでの V^1，V^2 の作成と同時に，鍵交換のため，V^1 と V^2 の鍵交換用の配列 V^{key1} と V^{key2} をヘッダに持たせる．図 3 では，1 つの転送状態リスト用いた場合の，鍵交換を説明している．まず，送信者は AP ルータに送信する公開鍵 $x_{Si}G$ を，配列 V^{key} に書き込む．パケットを受信すると AP ルータ R_i は，S との MAC 鍵 k_{S-R_i} を生成してから，D との ECDH 公開鍵 $x_{iD}G$ を同じフィールドに書き込む．受信者 D に到達すると，ヘッダには全ての AP ルータの $x_{iD}G$ が書き込まれているので，D は MAC 鍵 k_{D-R_i} を生成する．次に，D 用の EDCH 公開鍵 $x_{Di}G$ を書き込んで，S 宛にパケットを返送する．経路上の AP ルータは，同様に配列を S への EDCH 公開鍵に書き換えながら転送するので，最終的に，S と D と AP ルータは，MAC 鍵を生成する．

4.3.2　課題

dPHI では鍵交換において，AP ルータの S と D へのなりすまし，すなわち中間者攻撃を防ぐため，AP ルータの公開鍵証明書を用いる．一方，pPHI では，関係匿名性を満たすため，自身のアイデンティを漏洩させる公開鍵証明書を使用できない．以下では，AP ルータの公開鍵証明書を用いずに，中間者攻撃を防ぐ解法を提案する．

4.3.3　解法

4.3.3.1　真正性を保証した MAC 鍵 k_{S-D} の生成

S と D の MAC 鍵 k_{S-D} の鍵交換において，D の公開鍵の真正性を保証する．dPHI で生成したセッション鍵 $k1_{S-D}$ を使用することも考えられるが，前方秘匿性が無いため，別途 MAC 鍵 k_{S-D} を以下の通り，生成する．(1) D は，k_{S-D} 用の公開鍵/秘密鍵の対 pk_D/sk_D と pk_D への署名 σ

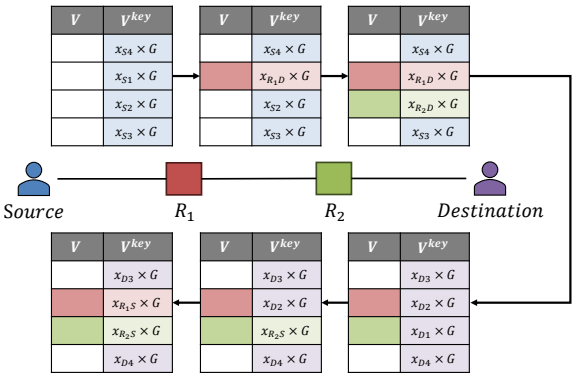

図 3　鍵交換の例

を生成する．(2) D は sk_D, pk_S から共有鍵 k_{S-D} を導出する．(3) D は $V^1||V^2||\sigma$ をセッション鍵 $k1_{S-D}$ で暗号化して，pk_D とともにペイロードに書き込んで S に送信する．(4) S は $k1_{S-D}$ で復号して署名 σ の検証を行い，sk_S, pk_D から共有鍵 k_{S-D} を導出する．

ここで，S の鍵ペア pk_S/sk_S は dPHI で使用している鍵ペアを再利用する．そのため，セッション ID が pk_S のハッシュ値と一致することを以って pk_S の真正性を保証できる．

4.3.3.2　経路検証を用いた中間者攻撃の防御

攻撃者が乗っ取った AP ルータは，S と D になりすまして鍵交換を行える．図 3 から分かるように，攻撃者の AP ルータは，S の経路検証に対して，自身と D の間の AP ルータに送信される，S の公開鍵を書き換えられる．一方，D の経路検証に対しては，自身と D の間の AP ルータに送信される，D の公開鍵を書き換えられる．すなわち，攻撃者の AP ルータは，D の経路検証に対して，S になりすます可能性があり，一方，S の経路検証に対して，S になりすます可能性がある．図 4 に攻撃者のルータ R_2 が S と D になりすました，ECDH 公開鍵の偽造の例を示す．

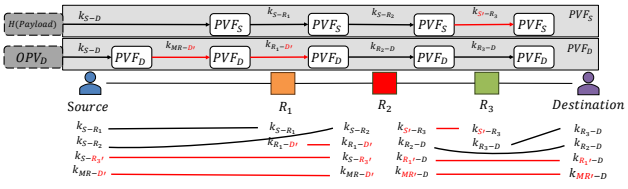

図 4　中間者攻撃による鍵の偽造

これに対して，鍵交換の後に転送されるパケットに対して，S と D が経路検証を行うことで，S と D になりすます中間者攻撃を防ぐ．まず，D へのなりすましについて，攻撃者の AP ルータ R_i は，自身より下流の AP ルータ $R_j(1 \leq i-1)$ の D の経路検証用の PDF_D の MAC 鍵は偽造できるが，S と D の MAC 鍵 k_{S-D} を知らないため，S が計算 s る PDF_D の初期値を知らない．このため，D に成りすませない。一方，S へのなりすましについて，攻撃者の AP ルータ R_i は，S の経路検証のための，自身より上流の AP ルータ $R_j(i+1 \leq n)$ の OPV_j は計算できるが，S と D の MAC 鍵 k_{S-D} を知らない．この結果，ヘッダ上の OPV_D を計算（偽造) できないため，S になりすませない．

5.　プロトコル

pPHI は，dPHI をベースに，パス設定フェーズで鍵交換を行い，データ転送フェーズで MAC 連鎖を計算する．本節では，E-OPT の dPHI への組み込にあたって，dPHI ならびに E-OPT に修正を加えた点を中心に説明する．

5.1　経路設定フェーズ

経路設定フェーズは，dPHI と同様に 2 ラウンドから構成され，転送状態リストの作成に加えて，経路検証用の鍵交換を平行して行う．第 1 ラウンドでは送信者-ミッドウェイルータ間で $k2$ の交換を，第 2 ラウンドでは，S と D と AP ルータの MAC 鍵の交換を行う。このため，V^1 と V^2 の AP ルータとの鍵交換において，公開鍵を交換するための配列 V^{key1} と V^{key2} を追加する．ここで，$k2_{S-i}$ の交換には，V^{key1} を使用する．

V^1，V^2 を含めて全ての配列は，AP ルータに経路長などの情報が漏洩しないように，循環リストとして使用する．このため，V^1 に対しては，S が第一ホップの AP ルータの配列上の位置 $sourcePos$ を決定し，ヘッダのフィールド $H.pos$ に記録する．以降は，転送するごとに，剰余 l で加算することで，次ホップの AP ルータに配列上の位置を知らせる．ミッドウェイルータ W は，自身の V^1 上の位置を $midwayPos1$ として記録し，V^2 に対して，W 第一ホップの AP ルータの配列上の位置 $midwayPos2$ を決定する．

5.1.1　第 1 ラウンド

S は，鍵交換の公開パラメータ G を公開し，配列 V^{key1} に $k2_{S-R_i}$ のための公開鍵を書き込む．一方，V^1 上の AP ルータは自身の公開鍵に書き換えることで，公開鍵を交換する．配列上の位置は，$H.pos$ で決まる．この時，データ転送フェーズで送信者がモックルータの数を計算できるように，ミッドウェイルータは，S に返送するパケットに，$midwayPos1$ 及び自身が決めた $midwayPos2$ を，W と S のセッション鍵 $k2_{S-W}$ で暗号化して返送する．

5.1.2　第 2 ラウンド

往路では，転送状態の作成と，$x_{S_j}G, x_{R_iD}G$ の共有を行う．初めに，Source が k の交換のため $2l$ 個の鍵ペアを生成し，$x_{S_j}G$ を V^{key1} に，$k_{S(j+l)}G$ を V^{key2} に書き込む．パケットを受け取った S と W 間の AP ルータは，$V^{key1}[H.pos]$ 上の $x_{S_j}G$ を保存し，代わりに $x_{R_{iD}}G$ を書き込む．W は同様

の手順で鍵交換の操作を行ったのちに，$V^{key1}[H.pos]$ が，V^{key1} の末尾の要素になるように V^{key1} 全体を回転させる．その後，W と D 間の AP ルータは，転送状態を作成した後に，$V^{key2}[H.pos]$ 上の $x_{S_j}G$ を保存し，代わりに $x_{R_{iD}}G$ を書き込む．最後に，D は V^{key1}, V^{key2} 上の $x_{R_{jD}}G$ を保存する．

復路では，$x_{R_{jS}}G, x_{S_j}G$ の共有を行う．初めに，D が $l*2$ 個の ECDH 鍵ペアを生成し，$x_{D(j+l)}G$ を V^{key1} に，$x_{D_j}G$ を V^{key2} に書き込み，パケットを送信する．パケットを受け取った W-D 間の AP ルータは，$V^{key2}[H.pos]$ 上の $x_{D_j}G$ を保存し，代わりに $x_{R_{iS}}G$ を書き込む．次に，W はリスト V^{key1} の末尾が $V^{key1}[H.pos]$ になるように V^{key1} 全体を回転させ，その後 W-D 間の AP ルータと同様の手順で鍵交換の操作を行う．その後，S-W 間の AP ルータは $V^{key1}[H.pos]$ 上の $x_{D_j}G$ を保存し，代わりに $x_{R_{iS}}G$ を書き込む．最後に，S は V^{key1}, V^{key2} 上の $x_{R_{jS}}G$ を保存する．

5.2 データ転送フェーズ

データ転送フェーズは，S から D と，D から S に転送されるパケットの経路検証を行う．図 5 には，S から D への経路検証を例示している．

図に示すように，OPV'_X の計算には，W までの AP ルータは PVF_S，ペイロードのハッシュ値，タイムスタンプ，PVF_{D2} を，それ以外の AP ルータは PVF_S，ペイロードのハッシュ値，タイムスタンプを用いる．これは，4.2 節で述べたように，PVF_{D2} の改竄を OPV_X と OPV'_X の不一致によって検出可能にするためである．また，受信者は OPV_D を用いた検証以外に PVF_{D1}, PVF_{D2} を用いた検証を任意で行うことができる．

一方，D から S 方向では，受信者が PVF_D，ペイロードのハッシュ値，タイムスタンプを用いて OPV_X を計算し，経路上の AP ルータと送信者が OPV'_X を再計算して経路検証を行う．この場合も，同様に，送信者は PVF_S を用いた検証を行うことが可能である．

6. 性能評価

本章では，ヘッダ長ならびに暗号処理と MAC 演算の回数を評価する．

6.1 ヘッダサイズ

パケットヘッダ上のリスト長に拠らないフィールドは 131 バイトで，内訳は，セッション ID，ペイロードのハッシュ値 $H(P)$，$OPV_D, PVF_S, PVF_{D1}, PVF_{D2}$，タイムスタンプ TS が各 16 バイトと，dPHI の状態を表す変数，H.pos が各 1 バイト，$H.midway$ が 17 バイトである．また，転送状態 1 つが 39 バイト，OPV_{R_i} が 16 バイト，PVF_{D2} の暗号化に使用する Initizlized Vectot (IV) は暗号化 1 回につき 16 バイトである．転送状態のリスト V^1, V^2 の要素数を l としたとき，転送状態と OPV_{R_i} は $2l$ 個，IV は l 個存在する．よってヘッダ長は，$131 + ((39 + 16) \times 2 + 16)l$ バイトとなり，例えば $l = 8$ のとき 1013 バイト，$l = 12$ のとき 1643 バイトとなる．

dPHI のヘッダ長は $50 + (39 \times 2)l$ バイトであり，$l = 8$ のとき 674 バイト，$l = 12$ のとき 986 バイトである．即ち，pPHI のヘッダ長は，dPHI と比べて，$l = 8$ のとき 1.69 倍，$l = 12$ のとき 1.67 倍となっていることが分かる．

ここで，E-OPT は中間ルータが n 個の場合の追加のヘッダサイズが $84 + 16n$ バイトとなり，dPHI に重畳した場合の単純な和は $134 + (55 \times 2)$ バイトで pPHI の場合より小さい．この差異は，PVF_D を二つに分けたことと PVF_{D2} 暗号化に用いる初期化ベクトル IV をパケットヘッダに記載していることによる．しかしながら，PVF_{D2} の認証を認証付き暗号で行うのではなく OPV に重畳することで，パケットヘッダ上に認証タグを保存する必要がなくなるため，$16l$ バイトの圧縮している．

6.2 演算回数

V^1, V^2 それぞれの長さを l，経路上の AP ルータの数を n，V^1 に含まれるモックルータ数を m 個とする（ただし，$n + m \leq 2l$）．このとき，パス設定フェーズ第 1 ラウンド・第 2 ラウンドの演算回数はと，データ転送フェーズ第 1 ラウンドの演算回数は表 6 のようになる．評価から分かるように，AP ルータでの演算回数は常に $O(1)$ であり，また Source，Destination の演算回数も $O(l)$ である．

7. 関連研究

匿名性と経路完全性の両立を目指した研究は開始されたばかりである．匿名性を考慮した経路検証プロトコルとして，PrivNPV，VALNET [9, 10] が提案されたが，双方ともに，経路検証に主体を置いて，匿名性と経路検証を組み合わせたプロトコルである．ソースルーティングを行い，送信者が中間ルータに対してその前後のノードのみを共有することで関係匿名性を達成している．プロトコルは 2 つのフェーズに分かれ，パス設定フェーズでパスの共有と鍵交換を，データ転送フェーズで経路検証を行う．しかし，受信者に対する送信者匿名性は達成しておらず，ユーザのプライバシーが十分に守られていない．

8. おわりに

本研究では，dPHI における経路完全性の脆弱性を示し，それを改善させる形で軽量匿名通信プロトコルに経路検証を導入したプロトコル pPHI を設計した．

謝辞 本研究は、JSPS 科研費 23K28073 によるものである．

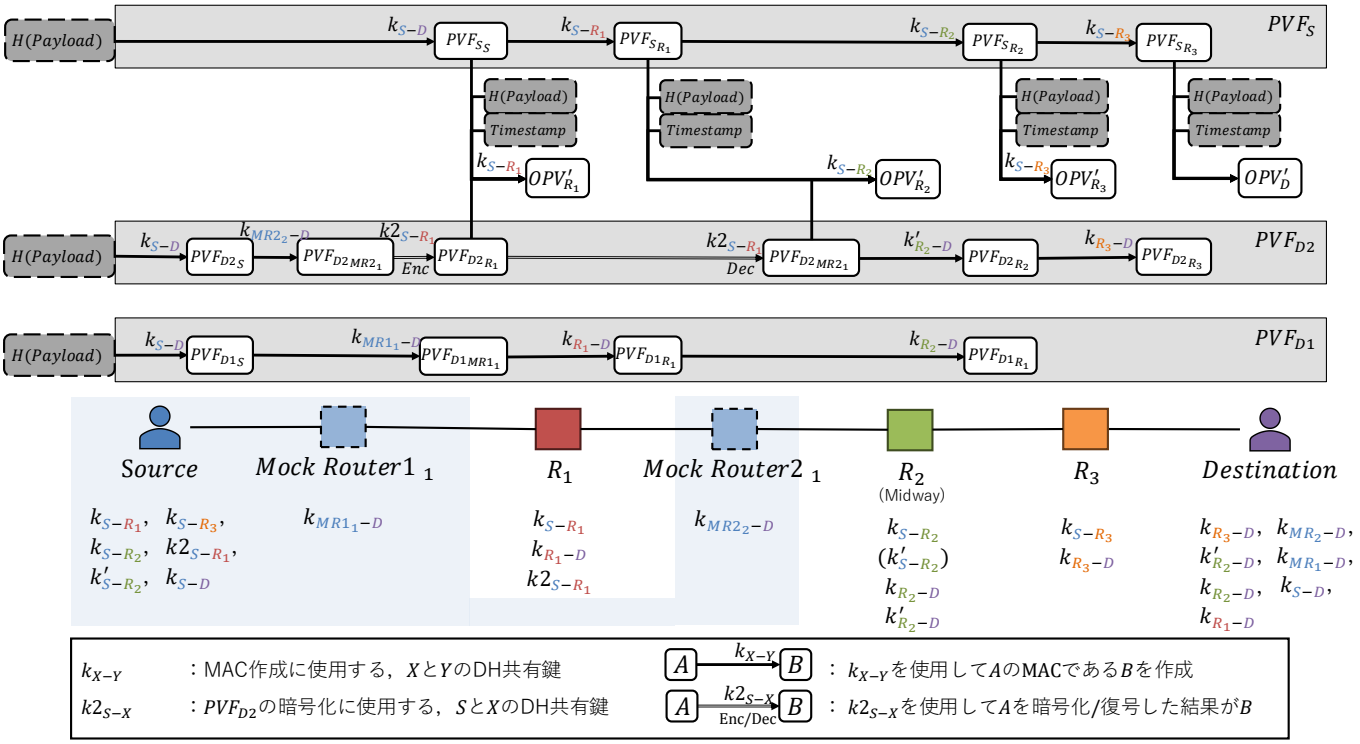

図 5　データ転送フェーズ

		S	$R \in V^1$	W	M	$R \in V^2$	D
path setup	make tmp key pair	$3l+1$	3	3	1	2	$2l+1$
	exchange ECDH key	$3l+4$	3	3	2	2	$2l+2$
	Encrypt routing infomation	0	1	1	0	1	0
data transmission	Decrypt routing infomation	0	1	1	-	1	0
	Enc/Dec PVF_{D2}	$l-m-1$	1	0	-	0	0
	MAC for PVF_S	1	1	1	-	1	0
	MAC to make OPV/OPV'	$n+1$	1	1	-	1	1
	MAC for PVF_{D1}/PVF_{D2}	$2+l-n-1$	1	2	-	1	$2l$

表 6　各フェーズの演算回数

参考文献

[1] Bajic, A. and Becker, G. T.: dPHI: An improved high-speed network-layer anonymity protocol, *Proceedings on Privacy Enhancing Technologies* (2020).

[2] Bu, K., Laird, A., Yang, Y., Cheng, L., Luo, J., Li, Y. and Ren, K.: Unveiling the Mystery of Internet Packet Forwarding: A Survey of Network Path Validation, *ACM Computing Surveys*, pp. 292–296 (2020).

[3] Cai, H. and Wolf, T.: Source Authentication and Path Validation in Networks Using Orthogonal Sequences, *Proceedings of of IEEE ICCCN*, pp. 1–10 (2016).

[4] Chen, C. and Perrig, A.: Phi: Path-hidden lightweight anonymity protocol at network layer, *Proceedings on Privacy Enhancing Technologies*, Vol. 2017, No. 1, pp. 100–117 (2017).

[5] Farrell, S. and Tschofenig, H.: Pervasive monitoring is an attack, Technical report (2014).

[6] Kim, T. H.-J., Basescu, C., Jia, L., Lee, S. B., Hu, Y.-C. and Perrig, A.: Lightweight source authentication and path validation, *Proceedings of the 2014 ACM Conference on SIGCOMM*, pp. 271–282 (2014).

[7] Levin, D., Lee, Y., Valenta, L., Li, Z., Lai, V., Lumezanu, C., Spring, N. and Bhattacharjee, B.: Alibi routing, *Proceedings of of ACM SIGCOMM*, pp. 611–624 (2015).

[8] Naous, J., Walfish, M., Nicolosi, A., Mazieres, D., Miller, M. and Seehra, A.: Verifying and enforcing network paths with ICING, *Proceedings of the Seventh Conference on Emerging Networking Experiments and Technologies*, pp. 1–12 (2011).

[9] Sengupta, B.: Valnet: Privacy-preserving multi-path validation, *Computer Networks*, Vol. 204, p. 108695 (2022).

[10] Sengupta, B., Li, Y., Bu, K. and Deng, R. H.: Privacy-preserving network path validation, *ACM Transactions on Internet Technology (TOIT)*, Vol. 20, No. 1, pp. 1–27 (2020).

[11] Yoshinaka, Y., Takemasa, J., Kuizumi, Y. and Hasegawa, T.: gPHI: ALightweight Anonymity Protocol for Anonymity at Host and AS Levels, *Proceedings of IFIP Networking Conference*, pp. 1–9 (2022).