

Hole Punching を用いた NAT 越え Mobile PPC の設計

鈴木 秀和[†] 渡邊 晃^{†,††}

[†] 名城大学大学院理工学研究科 〒468-8502 愛知県名古屋市天白区塩釜口 1-501

E-mail: [†]h.suzuki@wata-lab.meijo-u.ac.jp, ^{††}wtnbakr@ccmfs.meijo-u.ac.jp

あらまし IPv4 ネットワークにおいて移動透過性を実現する技術として、End-to-End でそれを可能とする Mobile PPC (Mobile Peer-to-Peer Communication) がある。IPv4 ネットワークでは、移動ノードが通信中にグローバルネットワークとプライベートネットワークを跨って移動することが想定される。しかし、従来の Mobile PPC では通信経路上に NAT が存在すると、IP アドレスが途中で変換される影響により通信を継続できないという課題があった。本稿では NAT 越え手法として知られている Hole punching の原理を Mobile PPC に適用することにより、上記課題を解決する仕組みを示す。

キーワード 移動透過性, Mobile PPC, IPv4, NAT 越え, Hole punching

Design of NAT Traversal for Mobile PPC Applying Hole Punching

Hidekazu SUZUKI[†] and Akira WATANABE^{†,††}

[†] Graduate School of Science and Technology, Meijo University

1-501 Shiogamaguchi, Tempaku-ku, Nagoya-shi, 468-8502 Japan

E-mail: [†]h.suzuki@wata-lab.meijo-u.ac.jp, ^{††}wtnbakr@ccmfs.meijo-u.ac.jp

Abstract Mobile Peer-to-Peer Communication (Mobile PPC) is an End-to-End technology that realizes mobility in IPv4 network. It is thought that a mobile node sometimes moves between a global network and a private network in IPv4 network during communication. In case of the existing Mobile PPC, the communication breaks because of the IP address is translated by NAT when NAT exists on the communication path. In this paper, we present a NAT traversal scheme for Mobile PPC to solve the above-mentioned problem by applying a principle of hole punching technique.

Key words Mobility, Mobile PPC, IPv4, NAT traversal, Hole punching

1. はじめに

携帯端末や無線ネットワークの普及により、ユーザはいつでも、どこからでもネットワークに接続できるようになった。現在は携帯電話網の利用が圧倒的に多いが、近年では、無線 LAN を搭載した携帯端末も登場しており、今後ますます公衆無線 LAN サービスを利用するユーザが増加すると考えられる。ユーザはこのような無線 LAN 携帯端末を利用することにより、いつでもインターネットに接続できる。しかし、IP ネットワークでは通信中にユーザが移動すると IP アドレスが変化するため、通信が切断されてしまう。この問題を解決するために、移動透過性を実現する様々な方式 [1] が提案されている。

移動透過性を実現する技術は、IPv6 の利用を前提としている場合が多い。しかし、現状では IPv6 ネットワークがまだ広く浸透しておらず、今後も IPv4 が重要な役割を果たすと考えられる。従って、IPv4 ネットワークにおける移動透過性技術の実現は大きな意義がある。IPv4 ネットワークではアドレス枯渇

問題のため、全ての移動ノード（以下 MN; Mobile Node）にグローバルアドレスを割り当てることは困難であり、プライベートアドレスを積極的に利用する必要がある。そのため、グローバルアドレス空間のネットワーク（以下グローバルネットワーク）とプライベートアドレス空間のネットワーク（以下プライベートネットワーク）を跨った移動があり得る。このような移動では移動前と移動後の通信経路のどちらかに NAT (Network Address Translator) が介在することになる。その結果、移動通知情報に含まれる IP アドレスと実際の通信で用いられる IP アドレスが一致せず、移動前後の IP アドレスの関係を正しく管理できないという課題がある。Mobile IP [2] においては、この問題を解決するため、移動通知を UDP によりカプセル化処理を行ったり、NAT に独自の機能を追加するなどの対策を検討している [3]~[6]。しかし、この方法ではカプセル化処理に起因するヘッダオーバーヘッドのため伝送効率が低下したり、特殊な NAT が必要になるなどの課題が残る。

筆者らは IPv4 に対応し、かつエンドエンドで移動透過性を実

現する Mobile PPC (Mobile Peer-to-Peer Communication) [7] を提案している。Mobile PPC では、DDNS (Dynamic DNS) により通信相手ノード (以下 CN; Correspondent Node) の IP アドレスを取得してから通信を開始する。MN が通信中に移動して IP アドレスが変化した場合、CN に対して移動前後の IP アドレスの関係を直接通知し、その対応関係を IP 層に記憶する。その後、IP 層で全ての通信パケットにアドレス変換処理を行うことにより、上位層から IP アドレスの変化を隠蔽することができる。Mobile PPC は経路の冗長が発生せず、パケット長が変化しないため高スループットが実現できる特徴があり、今後のユビキタスネットワークに適した方式である。筆者らは既に文献[8]において、Mobile PPC の拡張により MN がグローバルネットワークとプライベートネットワークを跨った移動透過通信を実現できる方法を示している。しかし、MN の移動先ネットワークが特殊な NAT の配下でなければならないという課題があった。

そこで、本稿では既存の NAT をそのまま利用でき、かつ MN がアドレス空間の違いに関わらず自由に移動できる方式を検討した。提案方式では NAT のアドレス変換に対応するために、NAT 越え技術の代表的な手法として知られている Hole punching [9] の原理を導入する。本稿では CN がグローバルネットワーク上に位置し、MN がグローバルネットワークとプライベートネットワークを相互に移動する状況を想定する。MN は CN に対して Hole punching に当たる Binding 処理を行うことにより、NAT にマッピング情報を生成し、さらに NAT の外側に割り当てられた IP アドレスとポート番号を取得する。その後、MN は取得した情報を CN 当てに送信する移動通知情報に含めることにより、CN 側で適切なアドレス変換処理を実行させることができる。提案方式では既存の NAT をそのまま利用することができる。また、Binding 処理は直接エンドノード間で実行するため、Mobile PPC の特徴をそのまま維持することができる。

以降、2. で従来の Mobile PPC と Hole punching の原理を述べ、3. で提案方式の仕組みを示す。4. で考慮すべき事項やセキュリティに関する考察を行い、5. でまとめる。

2. 既存技術

2.1 Mobile PPC

本稿で用いる記号を以下のように定義する。

- G_i ; グローバル IP アドレス
- P_i ; プライベート IP アドレス
- $A : p$; IP アドレス A , ポート番号 p
- $S \rightarrow D, D \leftarrow S$; S から D への通信
- $S \leftrightarrow D$; S と D 間の通信
- $S \rightleftharpoons D$; S から D , または D から S へのアドレス変換

図 1 に Mobile PPC における通信開始から、移動後の通信継続までの一連の流れを示す。MN は通信開始に先立ち、CN と Diffie-Hellman (以下 DH) 鍵交換を用いて認証鍵を共有する [10]。認証鍵は以後の移動情報通知処理において用いる。その後、MN は送信する TCP/UDP パケットの接続識別子

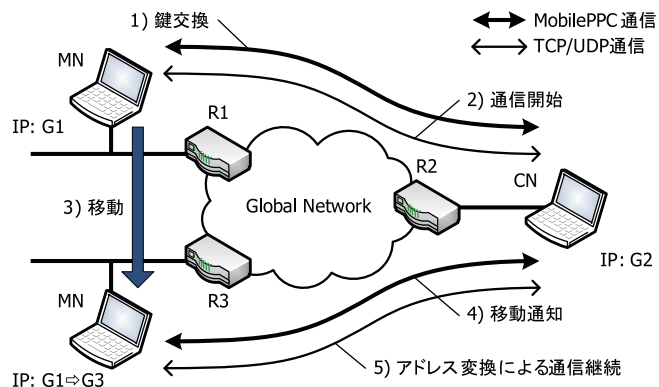


図 1 Mobile PPC の通信手順

Fig. 1 Communication procedure of Mobile PPC.

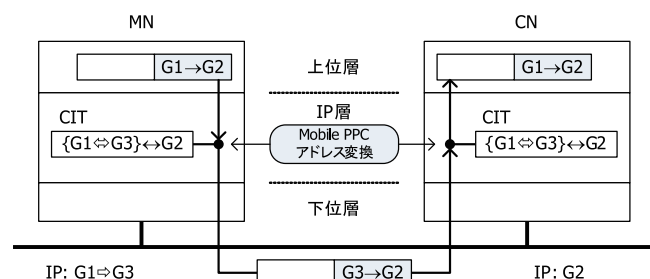


図 2 CIT に基づくアドレス変換

Fig. 2 Address translation based on CIT.

CID (Connection ID) ^(注1) を用いて、式 (1) に示す CIT (Connection ID Table) ^(注2) を生成し、通信を開始する。CN は TCP/UDP パケット受信時に MN と同様の CIT を生成する。

$$G1 : s \leftrightarrow G2 : d \quad (1)$$

MN が通信中に別のネットワークに移動して、新しい IP アドレス “G3” を取得すると、CN との間で移動通知処理を実行する。CN に送信する CU (CIT Update) Request には、移動前の CID と移動後の IP アドレスが記載され、通信開始時に共有しておいた認証鍵による署名が付加される。CU Request を受信した CN は認証処理を終えた後、MN の IP アドレスが “G3” となるように自身の CIT を式 (2) に示すように更新し、MN に CU Response を応答する。

$$\{G1 : s \leftrightarrow G3 : s\} \leftrightarrow G2 : d \quad (2)$$

MN も CN と同様に CIT を更新し、移動通知処理を完了する。以後、IP 層において更新した CIT に基づいて全通信パケットに対してアドレス変換が行われる。

図 2 に CIT に基づくアドレス変換と、移動後の通信の仕組みを示す。MN は上位層から渡されたパケットの送信元 IP アドレスを、移動前の “G1” から移動後の “G3” へ変換して CN へ

(注1): TCP コネクション、または UDP ストリームを識別するための情報であり、送信元/宛先 IP アドレス、ポート番号とプロトコルタイプの 5 つの値の組からなる。

(注2): 実際には TCP/UDP プロトコル別に生成されるが、本稿ではプロトコルに関する記載を省略する。

送信する。CN では受信したパケットの送信元 IP アドレスを、MN の移動後の“G3”から移動前の“G1”へ変換して上位層へ渡す。以上の処理により、上位層から IP アドレスの変化を隠蔽し、かつ正しいルーティング処理が行われ、通信を継続することができる。

ここで、本稿で想定するように MN と CN の通信経路上に NAT が介在すると、今のままでは以下のような課題が生じる。

課題 1 MN がグローバルネットワークからプライベートネットワークへ移動した場合、CN が MN から受け取る移動後 IP アドレスはプライベート IP アドレスのため、グローバルネットワーク上を正しくルーティングできない。

課題 2 仮に MN の移動後 IP アドレスを NAT の外部 IP アドレスに変換できたとしても NAT までは正しくルーティングされるが、NAT マッピング情報が存在しないため、MN に到達できない。

課題 3 MN がプライベートネットワークからグローバルネットワークへ移動した場合、CN が受け取る移動前 IP アドレスはプライベート IP アドレスであるが、CN は MN の移動前 IP アドレスを NAT の外部 IP アドレスとして認識している。従って、CN は正しく CIT を更新することができない。

2.2 Hole punching

Hole punching は NAT 越え技術として知られており、STUN (Simple Traversal of UDP Through NATs) [11] などの技術にも応用されている。図 3 に Hole punching による NAT 越え通信の概要を、STUN を例に挙げて示す。プライベートネットワークに存在する STUN クライアントは、グローバルネットワークに設置された STUN サーバへ Binding Request を送信する。STUN サーバは受信パケットの送信元 IP アドレスとポート番号の組、すなわち NAT の外側に割り当てられた IP アドレスとポート番号を MAPPED-ADDRESS として、Binding Response のデータ部に記載し応答する。これにより、STUN クライアントは MAPPED-ADDRESS を取得することができる。STUN クライアントは実際の通信相手 CN に対して、取得した MAPPED-ADDRESS をクライアント自身のトランスポートアドレス^(注3)として通知する。このアドレス通知処理は一般に SIP (Session Initiation Protocol) [12] が用いられることが多い。CN は通知されたトランスポートアドレスに向けて通信を開始すると、NAT では Binding 処理により生成されていた NAT マッピング情報に基づいて、パケットを STUN クライアントへ転送する。このように、CN は予め生成された NAT マッピング情報の MAPPED-ADDRESS がわかれば、NAT 越え通信を実現することができる。

Hole punching は既存の NAT を変更することなく NAT 越え問題を解決できるという利点がある。上記手法を Mobile PPC に適用することにより、CN に対して MAPPED-ADDRESS を通知し、課題 1 と課題 3 を解決できる。また、Binding 処理により NAT マッピング情報が生成されるため、課題 2 も解決できる。

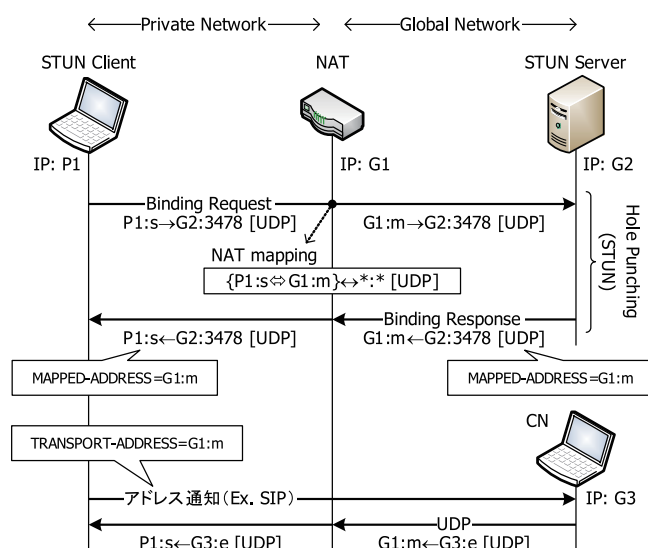


図 3 Hole punching による NAT 越え (例: STUN)

Fig. 3 NAT traversal by Hole punching (Ex. STUN).

ただし、Hole punching は Symmetric NAT^(注4)と呼ばれるタイプの NAT には適用できないうえ、TCP に対応できないという課題がある。またグローバルネットワーク上にサーバを設置する必要がある、さらには STUN クライアントのアプリケーションにこの機能を実装しなければならない。これらの課題は Mobile PPC の利点、すなわち上位プロトコルに依存しない、特有のサーバを必要としないという特徴を損ねてしまう。そこで、Hole punching の手法をそのまま適用するのではなく、上記利点を保持できるような工夫を追加する。

3. NAT 越え対応 Mobile PPC の検討

上記のような考察から、NAT 越えに対応する Mobile PPC の設計方針として、以下の 4 点を満たすことを目的とした。

- (1) 特有のサーバを設置することなく、NAT 越えを実現できる。
- (2) TCP/UDP の両プロトコルに対応できる。
- (3) Symmetric NAT を含むあらゆる NAT に対応できる。
- (4) アプリケーションには機能を一切追加しなくてよい。

これらの目的を実現するために、Mobile PPC に関わるネゴシエーションにおいて MN と CN の通信経路上に NAT の存在を確認した場合、両ノード間で直接 Hole punching を実行する。このために、Mobile PPC に新たに Binding Request/Response メッセージを定義する。以降、MN がグローバルネットワークからプライベートネットワークへ移動する場合と、その逆方向の移動について述べる。

3.1 アドレスに関する用語の定義

本方式で用いるアドレスを以下のように定義する。

- PREV-ADDRESS; CN から見た MN の移動前 IP アドレス・ポート番号の組
- MOVED-ADDRESS; CN から見た MN の移動後 IP アド

(注3) : 送受信に使用する IP アドレスとポート番号の組。

(注4) : 宛先が変化すると NAT の外側に割り当てられるポート番号も必ず変化する方式。

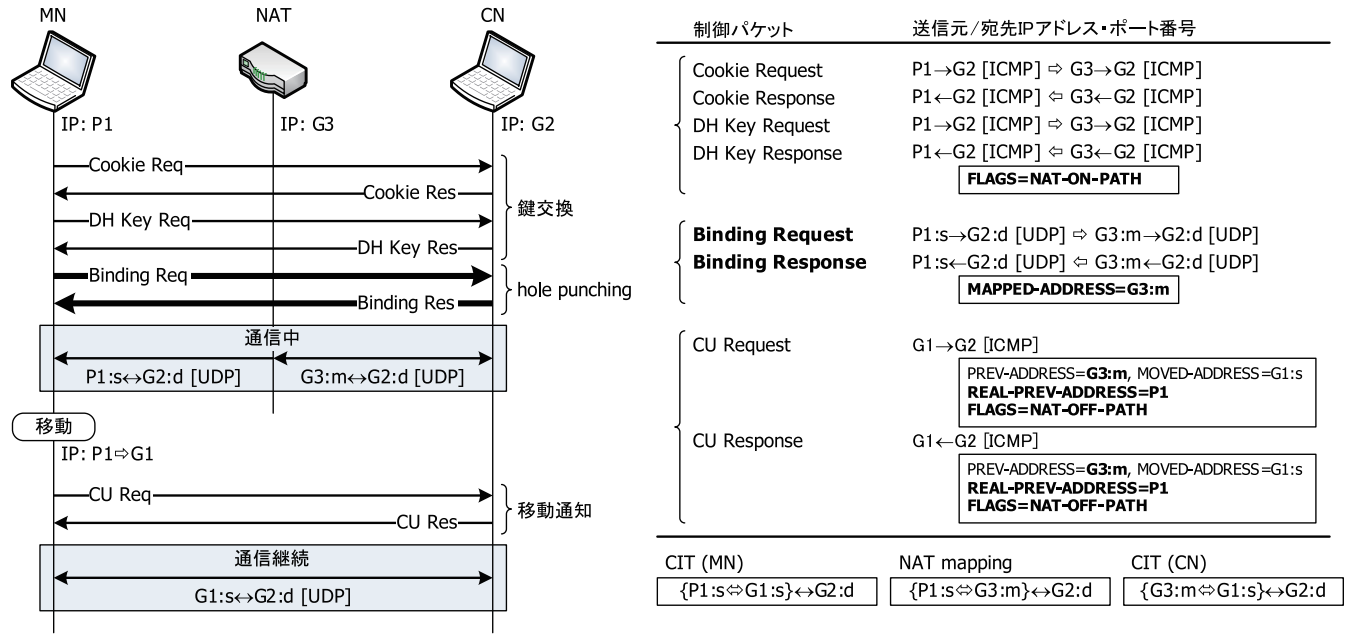


図6 プライベートネットワークからグローバルネットワークへ移動する場合の通信シーケンス

Fig. 6 Communication sequence when MN moves from private to global network.

らに新たに定義された Code と Flags の値（図中の太字部分）をチェックすることにより，Binding Request かどうか判別する．Binding Request であつたら，送信元 IP アドレス “G3” とポート番号 “m” を MAPPED-ADDRESS として，Binding Response のデータ部に記載して MN に応答する．

MN は上記 Binding Response を受信すると，取得した MAPPED-ADDRESS “G3 : m” を MOVED-ADDRESS に変更し，MN のプライベート IP アドレス “P1” を REAL-MOVED-ADDRESS として CU Request の情報に追加する．その後，フラグに NAT-ON-PATH を設定して CN へ再度 CU Request を送信する．上記 CU Request を受信した CN は認証処理を終えた後，従来の Mobile PPC と同様に自身の CIT を式 (6) のよう更新し，NAT-ON-PATH フラグをセットしたまま MN に CU Response を応答する．

$$CN : \{G1 : s \leftrightarrow G3 : m\} \leftrightarrow G2 : d \quad (6)$$

MN は上記 CU Response を受信したら，MOVED-ADDRESS は参照せず REAL-MOVED-ADDRESS “P1” を用いて CIT を式 (7) のように更新する．

$$MN : \{G1 : s \leftrightarrow P1 : s\} \leftrightarrow G2 : d \quad (7)$$

以後，IP 層において更新した CIT に基づいてアドレス変換が行われる．MN は上位層から渡されたパケットの送信元 IP アドレスを，移動前の “G1” から移動後の “P1” へ変換して CN へ送信する．NAT はマッピング情報に基づいて，送信元 IP アドレスとポート番号を “P1 : s” から “G3 : m” へ変換して CN へ転送する．CN は受信したパケットの送信元を移動後の “G3 : m” から移動前の “G1 : s” へ変換して上位層へ渡す．以上の動作により，グローバルネットワークからプライベートネットワークへ移動しても通信を継続することができる．

3.3 グローバルネットワークへの移動

次に，MN がグローバルネットワークからプライベートネットワークへ移動する場合の通信シーケンスを図 6 に示す．基本的な仕組みは 3.2 と同様である．通信開始時に MN と CN は DH 鍵交換により認証鍵を共有する．上記シーケンスには MN のプライベート IP アドレスが含まれているため，CN はパケットの送信元 IP アドレスと比較することにより，NAT を経由していることがわかる．そこで CN は鍵交換シーケンスの最後の応答に NAT-ON-PATH フラグを設定する．MN は認証鍵を生成した後，CN と Binding 処理を実行して MAPPED-ADDRESS “G3 : m” を取得する．

MN が通信中にグローバルネットワークへ移動して新しい IP アドレス “G1” を取得すると，CN に対して移動通知を行う．MN は通信開始時に NAT-ON-PATH フラグを取得しているため，CU Request には MAPPED-ADDRESS “G3 : m” を PREV-ADDRESS に，MN の移動前プライベート IP アドレス “P1” を REAL-PREV-ADDRESS として記載する．その後，NAT-OFF-PATH フラグを設定して CN へ送信する．CU Request を受信した CN は認証処理を終えた後，通常の Mobile PPC と同様に自身の CIT を式 (8) のよう更新し，MN に CU Response を応答する．

$$CN : \{G3 : m \leftrightarrow G1 : s\} \leftrightarrow G2 : d \quad (8)$$

MN は NAT-OFF-PATH フラグが設定された CU Response を受信するので，送信元 IP アドレスが MOVED-ADDRESS “G1” となるように CIT を式 (9) のように更新する．

$$MN : \{P1 : s \leftrightarrow G1 : s\} \leftrightarrow G2 : d \quad (9)$$

以後，MN は上位層から渡されたパケットの送信元 IP アドレスを，移動前の “P1” から移動後の “G1” へ変換してから CN へ送信する．CN では受信したパケットの送信元を移動後の

“G1 : s” から、移動前の “G3 : m” へ変換して上位層へ渡す。以上の動作により、プライベートネットワークからグローバルネットワークへ移動しても通信を継続することができる。

4. 考 察

提案方式により、CN がグローバルネットワーク上に存在する場合、MN がグローバルネットワークとプライベートネットワークを相互に移動しても、通信を継続できることを示した。これ以外の移動パターンとして、MN が異なるプライベートネットワーク間を移動するケースが考えられる。この場合は、3.3 の通信開始時のシーケンスと、3.2 の移動後のシーケンスを組み合わせることにより、実現可能である。

提案方式では Hole punching に相当する Binding 処理を実際の通信相手と直接実行するため、STUN サーバのような装置は不要である。さらに実際の通信パケットと同じ CID を用いて Binding メッセージを生成しているため、TCP/UDP の両プロトコルと Symmetric NAT に対応できる。Binding 処理をカーネルレベルで実行しているため、アプリケーションを一切変更する必要はない。また、Hole punching の原理を適用しているため、プライベートネットワークが階層的に構成されていても動作可能である。

既存の NAT を利用して NAT 越え通信を実現するには、NAT のマッピング情報を維持するために Keepalive が必要となる。特に UDP 通信の場合、Binding 処理により生成されたマッピング情報は一定期間の無通信状態が発生すると消去されてしまい、通信の継続に影響を及ぼす。そのため、MN は CN に対して Keepalive パケットを定期的に送信する必要がある。Keepalive の送信間隔をマッピング情報の有効時間以下にすればよいが、各セッション単位で Keepalive を実行する必要があるため、CN に対する負荷が増加する可能性がある。マッピング情報保持時間は実装の種類や設定により異なるため、最適な値を検討する必要がある。

次に、本方式で導入した Binding 処理の安全性に関して考察する。攻撃者は Binding Response に記載されている MAPPED-ADDRESS を改ざんすることにより、セッションハイジャックを試みることが考えられる。Mobile PPC では MN と CN は通信開始時に認証鍵を共有しているため、Binding Request/Response に署名を付加することによりメッセージ完全性を保証できる。従って、Binding 処理の改ざんを検出することが可能である。

攻撃者が偽の Binding メッセージを送信することにより、セッションの妨害を試みることが考えられる。MN が CN へ Binding Request を送信後、ただちに攻撃者が偽の Binding Response を MN へ送信する。この攻撃も MN と CN 間で共有している認証鍵を用いることにより、メッセージ完全性の検証過程で検出することができる。NAT によるアドレス変換を考慮すると、メッセージの完全性保証範囲は TCP/UDP ペイロード以降となる。そこで、攻撃者が正規の Binding Request を盗聴し、送信元を詐称した Binding Request を CN へ送信することが考えられる。この場合、CN は Binding Request を正規のメッセージと判断してしまい、詐称されたアドレスを MAPPED-ADDRESS として

Binding Response を生成し、詐称されたアドレス宛（一般には攻撃者）へ応答する。攻撃者は受信した Binding Response の送信元を CN に詐称して、NAT を経由して MN へ送信する。MN も Binding Response を正規のメッセージと判断してしまうため、以後の移動通知処理が完了すると攻撃者にセッションをハイジャックされてしまう。このような攻撃には、Binding Request のメッセージ部にシーケンス番号を設定して攻撃者による再送を防ぐ方法や、Ingress Filtering [13] を設定することで対策を講ずることが可能である。

5. ま と め

本稿では CN がグローバルネットワーク上に存在する場合、MN がグローバルネットワークとプライベートネットワークを相互に移動できる方式を提案した。Mobile PPC に Hole punching の原理を適用することにより、既存の NAT をそのまま利用することを可能とした。今後は提案方式の実装と性能測定を行い、有効性を確認する。また、CN がプライベートネットワーク上に存在する場合について検討を行う予定である。

謝辞 本研究の一部は、日本学術振興会科学研究費補助金（特別研究員奨励費）の助成を受けたものである。

文 献

- [1] 寺岡文男, “インターネットにおけるノード移動透過性プロトコル,” 信学論 (D-I), vol. J87-D1, no. 3, pp. 308–328, March 2004.
- [2] C. Perkins, “Ip mobility support for ipv4,” RFC 3220, IETF, Jan. 2002.
- [3] H. Levkowitz, and S. Vaarala, “Mobile ip traversal of network address translation (nat) devices,” RFC 3519, IETF, Apr. 2003.
- [4] G. Montenegro, “Reverse tunneling for mobile ip, revised,” RFC 3024, IETF, Jan. 2001.
- [5] 井戸上彰, 久保健, 横田英俊, “プライベートアドレスを使用するモバイルネットワーク間のローミング手順とその実装,” 情処学論, vol. 44, no. 12, pp. 2958–2967, Dec. 2003.
- [6] 野地川栄光, 加藤聰彦, 伊藤秀一, “プライベートネットワークに配置されたホームエージェントを用いた mobile ipv4 手順の設計,” 情処学 MBL 研報, vol. 2007, no. 98, pp. 17–24, Sept. 2007.
- [7] 竹内元規, 鈴木秀和, 渡邊晃, “エンドエンドで移動透過性を実現する mobile ppc の提案と実装,” 情処学論, vol. 47, no. 12, pp. 3244–3257, Dec. 2006.
- [8] 榎本万人, 鈴木秀和, 坂本順一, 渡邊晃, “プライベートアドレス空間とグローバルアドレス空間を跨る移動透過性の検討,” DICO2006, vol. 2006, no. 6, pp. 813–816, July 2006.
- [9] B. Ford, P. Srisuresh, and D. Kegel, “Peer-to-peer communication across network address translators,” Proc. USENIX Annual Technical Conference, pp. 179–192, Anaheim, CA, Apr. 2005.
- [10] 瀬下正樹, 渡邊晃, “Mobile ppc における認証方式の実装,” DI-COMO2006, vol. 2006, no. 6, pp. 809–812, July 2006.
- [11] J. Rosenberg, J. Weinberger, C. Huitema, and R. Mahy, “Stun - simple traversal of user datagram protocol (udp) through network address translators (nats),” RFC 3489, IETF, March 2003.
- [12] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler, “Sip: Session initiation protocol,” RFC 3261, IETF, June 2002.
- [13] P. Ferguson, and D. Senie, “Network ingress filtering: Defeating denial of service attacks which employ ip source address spoofing,” RFC 2827, IETF, May 2000.

Hole Punchingを用いた NAT越えMobile PPCの設計

名城大学大学院理工学研究科

鈴木 秀和 渡邊 晃



IPv4ネットワークにおけるモビリティの実現

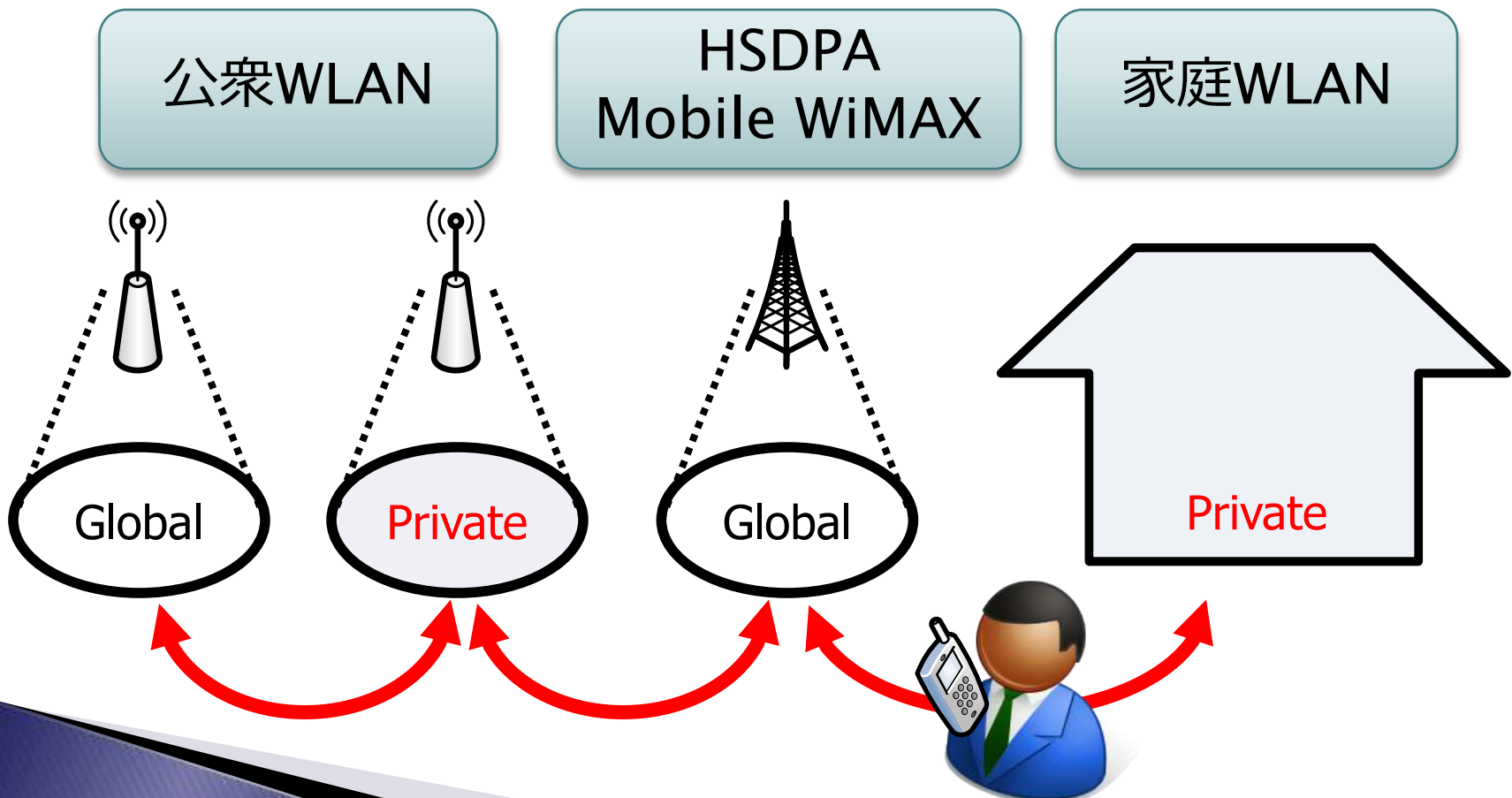
- ▶ 移動透過性の技術はIPv6が主流
 - 現状：IPv6ネットワークは広く浸透していない
 - 近未来：IPv4/IPv6ネットワークが混在



**IPv4ネットワークにおいても移動透過性を実現することは
移動体通信の利便性向上の観点から大きな意義がある**

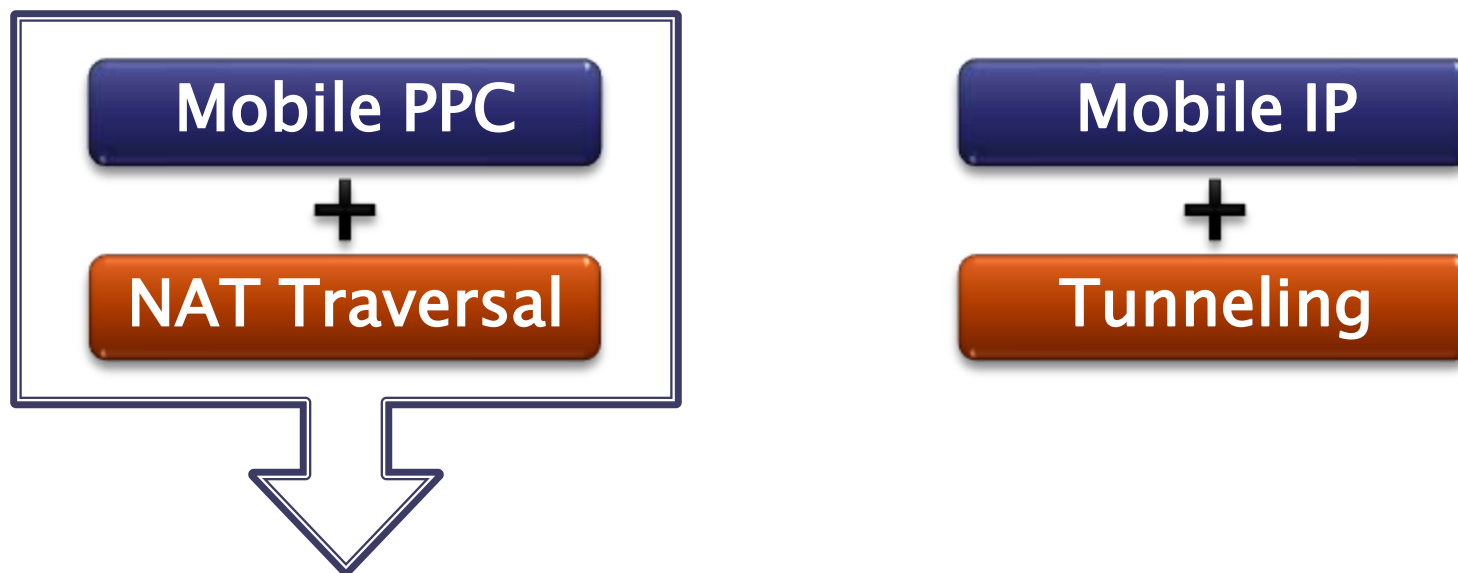
IPv4における移動ケース

- ▶ Global NWとPrivate NWをまたがった移動
 - NATの影響を考慮する必要がある



本発表の要旨

- ▶ **Mobile PPC** (Mobile Peer-to-Peer Communication)
 - End-to-Endで移動透過性を実現

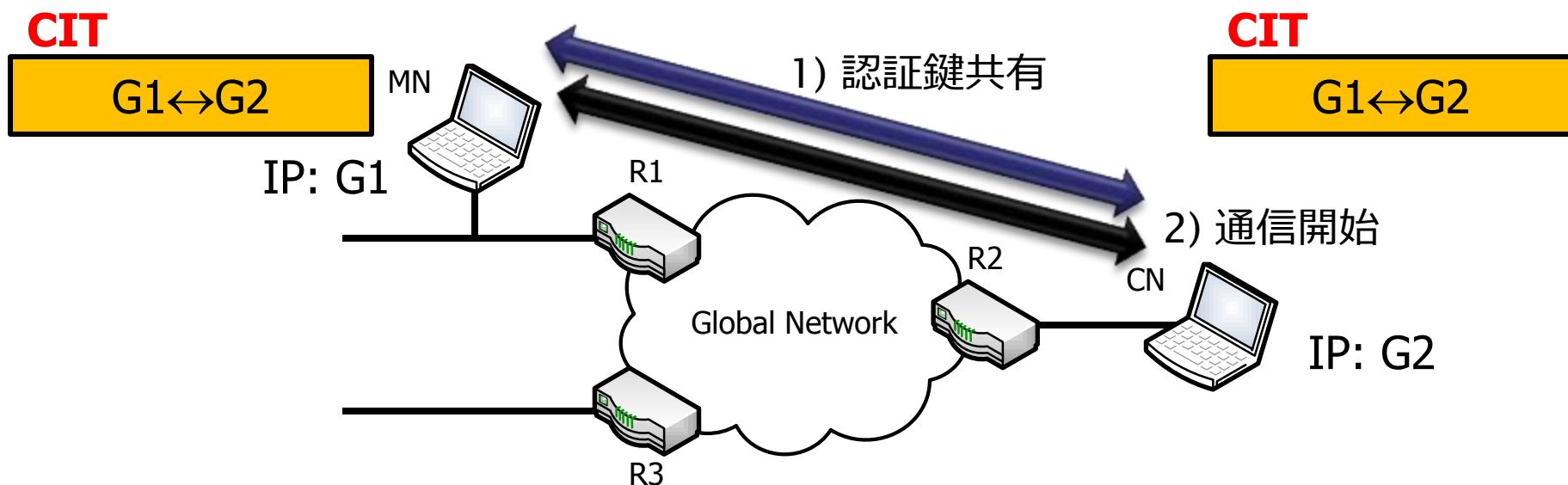


グローバルネットワークとプライベートネットワークを
相互に移動可能な方式を提案

Mobile PPCの概要

▶ 通信開始時

- CNと認証鍵を共有
- IP層に**CIT** (Connection ID Table) を生成

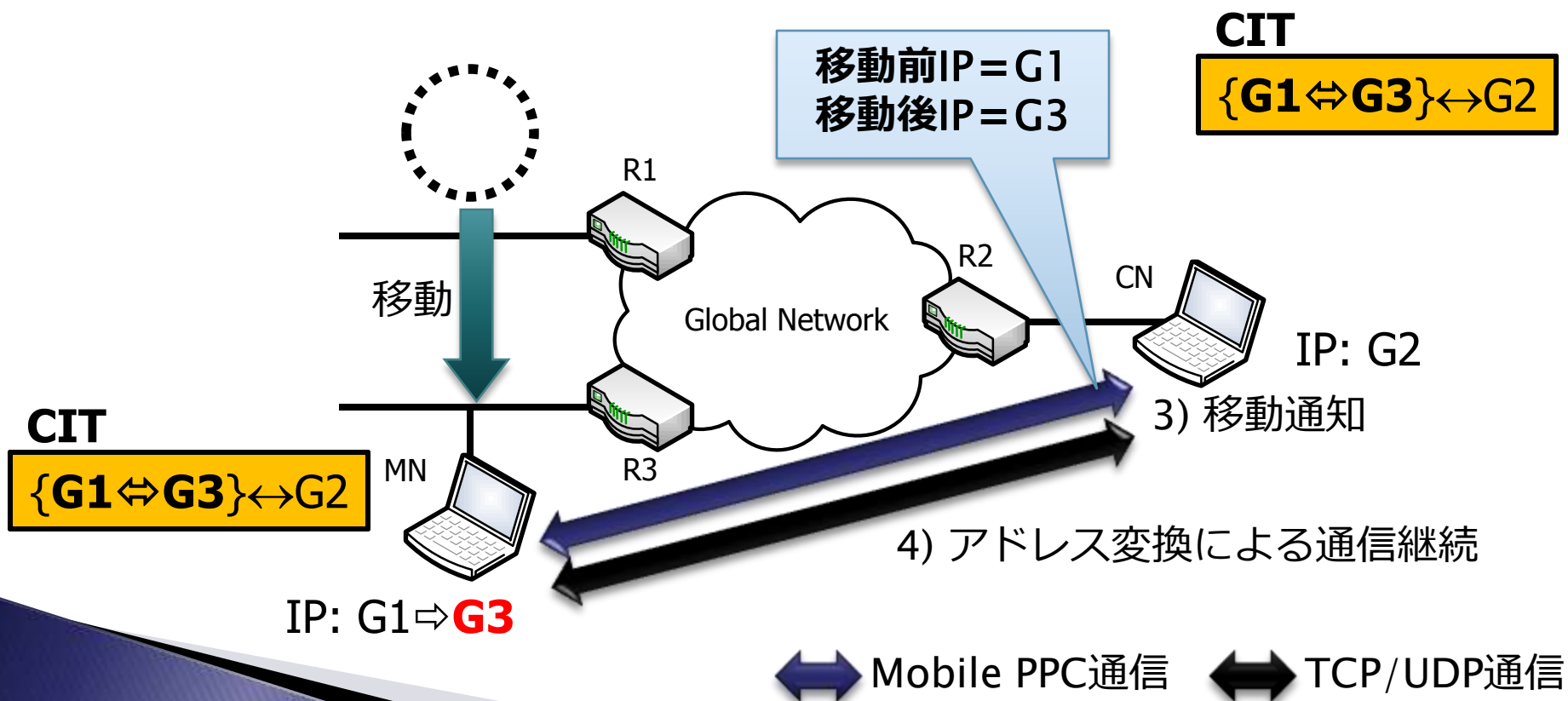


↔ Mobile PPC通信 ↔ TCP/UDP通信

Mobile PPCの概要

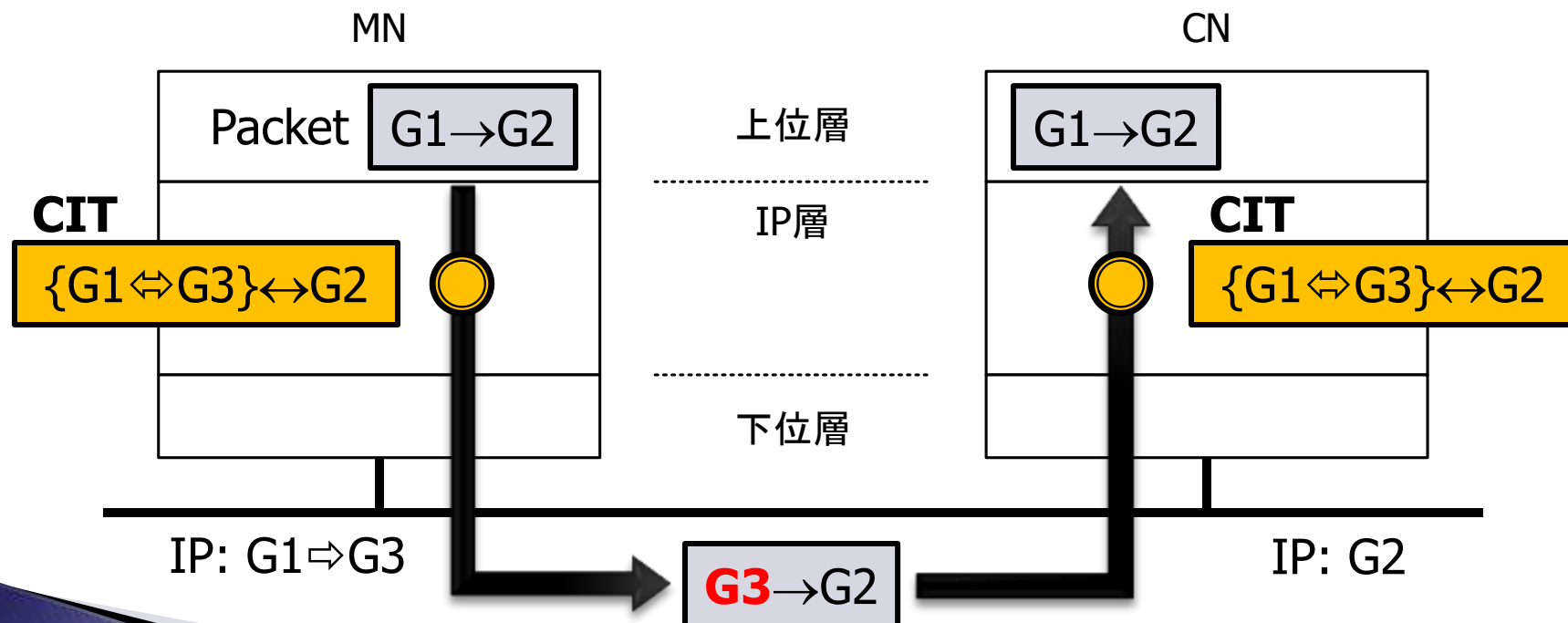
▶ MN移動時

- CNに**移動通知**（認証鍵による署名付）→**CIT更新**
- **アドレス変換**により通信を継続



Mobile PPCのアドレス変換

- ▶ アドレス変換により通信を継続
 - 上位層：移動前のIPアドレスで認識
 - 下位層：移動後のIPアドレスで正しくルーティング



Mobile PPCの特徴

- ▶ End-to-End方式
 - ➔ Mobile IPのHA, FAのような**特有のサーバが不要**
 - ➔ 冗長な経路がなく**低遅延**
- ▶ カプセル化不要
 - ➔ ヘッダオーバーヘッドやフラグメントが発生しないため**高スループット**
- ▶ IP層における技術
 - ➔ **上位プロトコルに依存しない**

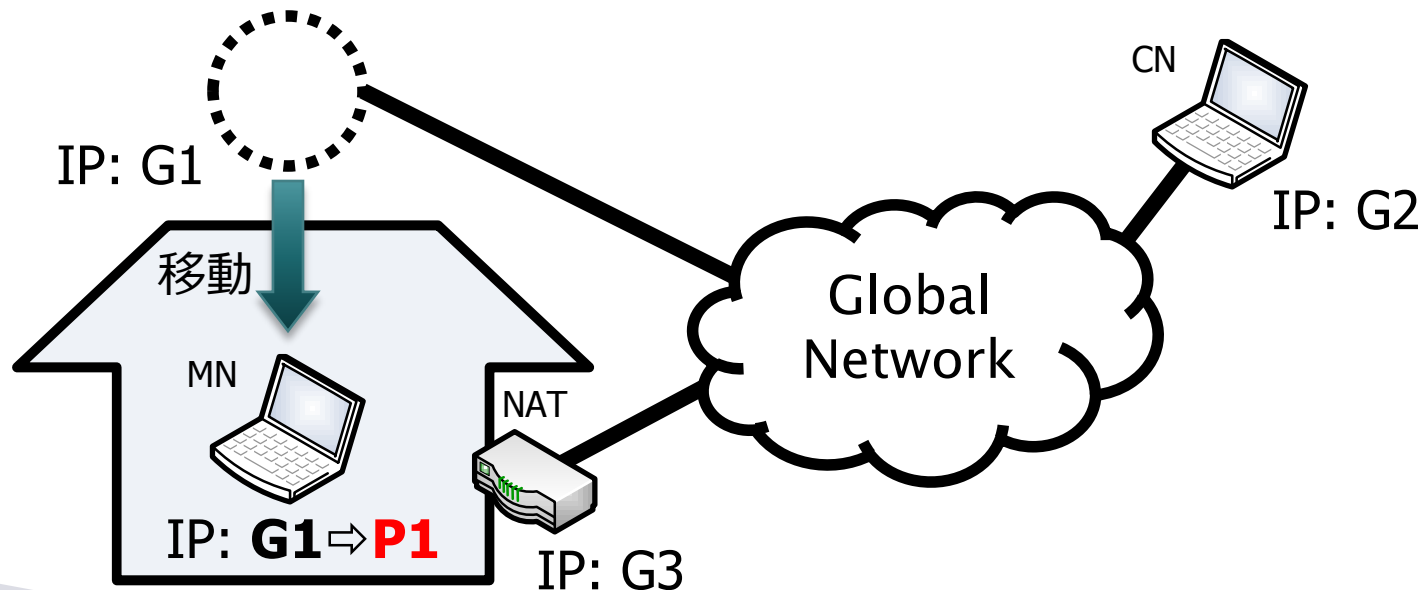
想定する移動ケース

▶ Global NWからPrivate NWへ

▶ Private NWからGlobal NWへ

▶ Private NWからPrivate NWへ

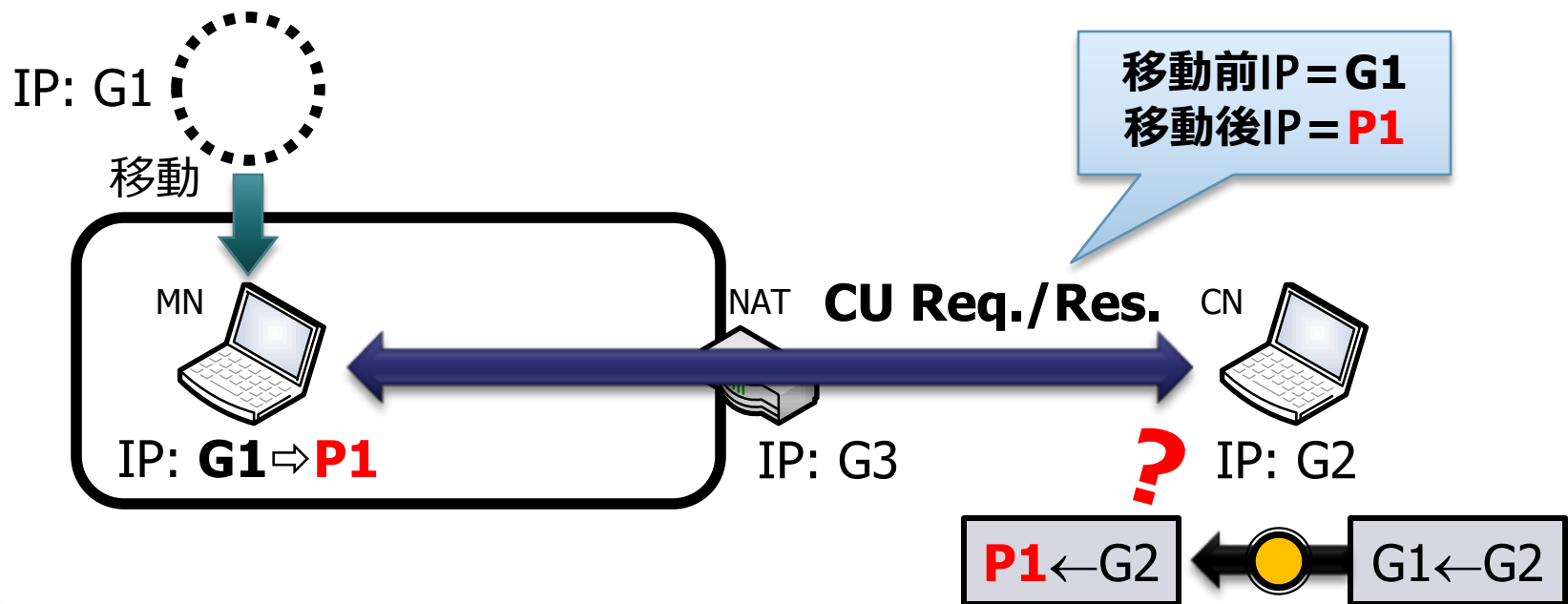
今回発表する内容



NATが存在する場合の問題

課題1 :

CNがMNから受ける移動後IPアドレス=プライベートIP
→ **正しく送信できない**

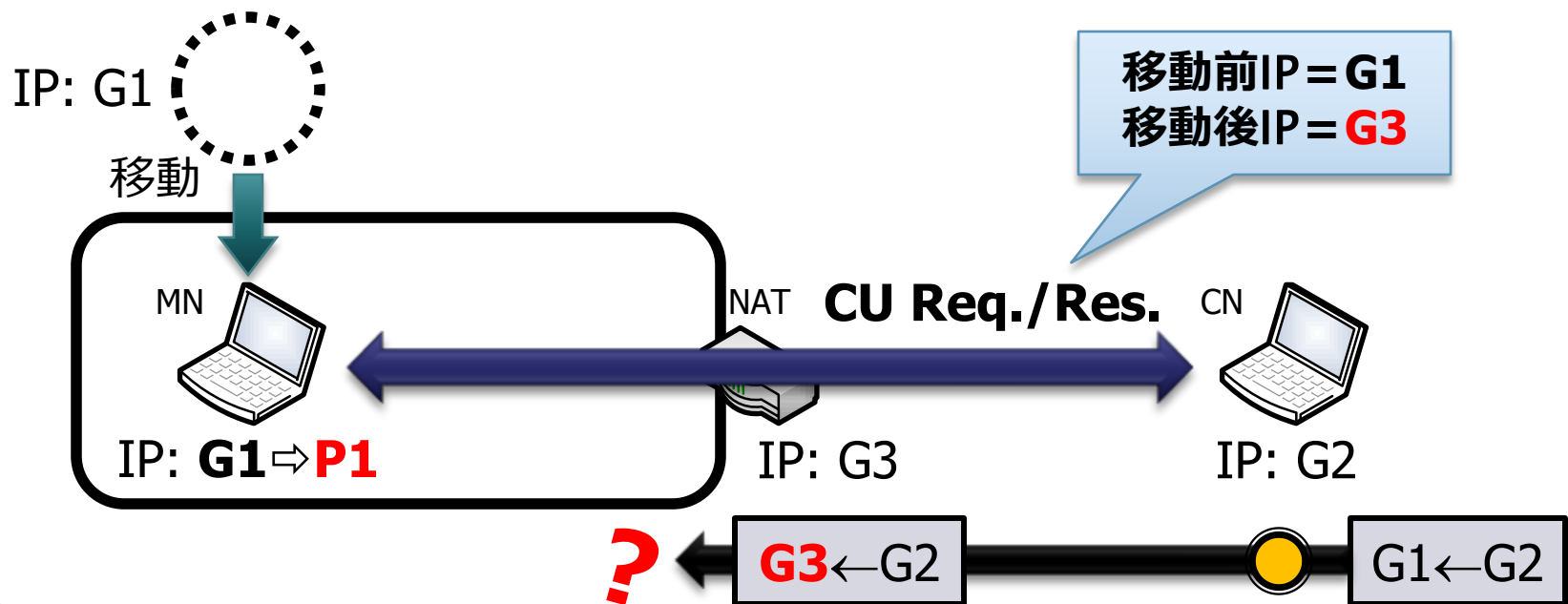


NATが存在する場合の問題

課題2：

移動後IPアドレス=NATのグローバルIPアドレス

→NATマッピング情報がない ∴MNまで転送できない



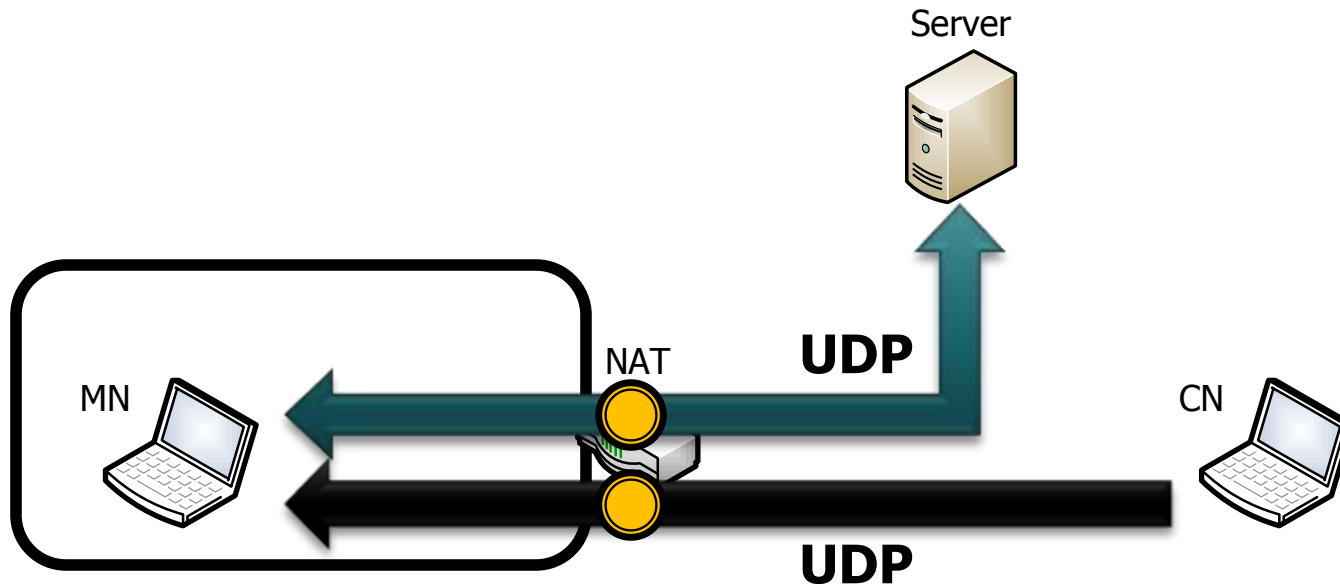
1. 通知する移動後IPがプライベートIP
→ NATのグローバルアドレスを通知すればよい
2. NATマッピング情報がないため転送できない
→ 事前にマッピング情報を生成しておけばよい



NAT越え技術である**Hole Punching**を使えば課題を解決できる!!

Hole Punching

- ▶ 事前にNATに「**穴を開けて**」からEnd-to-End通信
= マッピング生成
 1. ServerにUDP通信を行い, **NATマッピングを生成**
 2. Serverから**マッピングされたアドレスを取得**
 3. CNはマッピングされたアドレスに向けて通信



Mobile PPCに適用すると...

▶ 利点

- 既存のCone型NATで動作可能
- カプセル化不要 & End-to-End通信を実現
→ **Mobile PPCのコンセプトに一致**

▶ 欠点

- Symmetric NAT, TCPに対応できない
- グローバルネットワークにサーバが必要
- アプリケーションに対して機能追加が必要

Mobile PPCの利点を損ねてしまう
(上位プロトコルに依存しない, 特有のサーバが不要)

▶ Hole Punchingの原理を適用

1. 特有のサーバを設置することなく実現
2. TCP/UDPの両プロトコルに対応
3. Symmetric NATを含む既存のNATに対応
4. アプリケーションの機能追加は不要

▶ 本方式のキーポイント

- 通信経路上にNATの存在を確認
→ **IPレベルでHole PunchingをEnd-to-Endで実行**
- 新たに**Binding Request/Response**メッセージを定義

アドレスに関する用語の定義

▶ PREV-ADDRESS

- CNから見たMNの移動前IPアドレス・ポート番号

▶ MOVED-ADDRESS

- CNから見たMNの移動後IPアドレス・ポート番号

▶ MAPPED-ADDRESS

- NATにマッピングされたIPアドレス・ポート番号

▶ REAL-PREV-ADDRESS

- MNの移動前実IPアドレス

▶ REAL-MOVED-ADDRESS

- MNの移動後実IPアドレス

従来方式と提案方式の相違点

従来方式

移動前 (NATなし)

- 認証鍵の共有
- 通信開始

移動後 (NATなし)

- 移動通知
- 通信継続

提案方式

移動前 (NATなし)

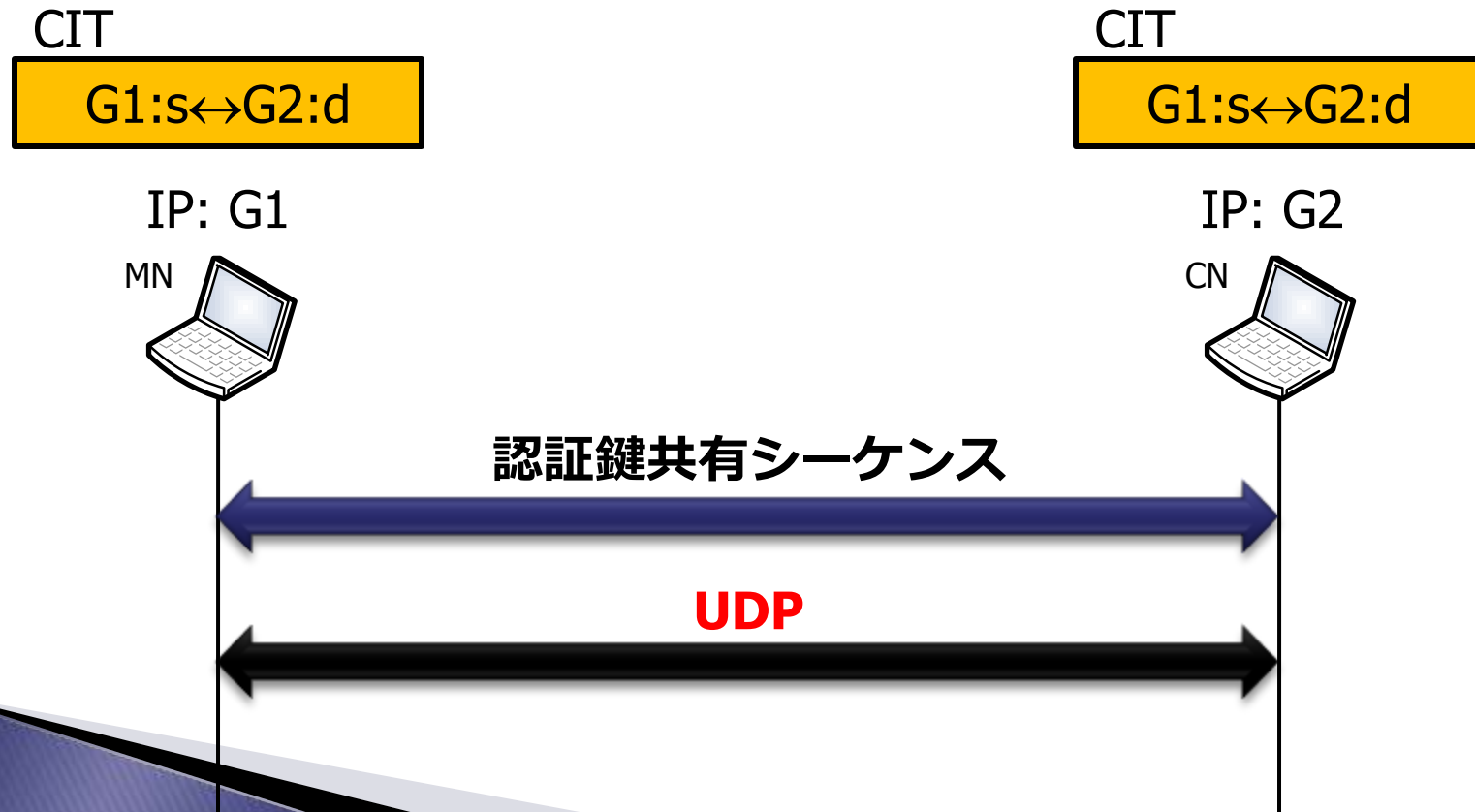
- 認証鍵の共有
- 通信開始

移動後 (NATあり)

- 移動通知 (**中断**)
- **Binding**処理
- **移動通知** (2回目)
- 通信継続

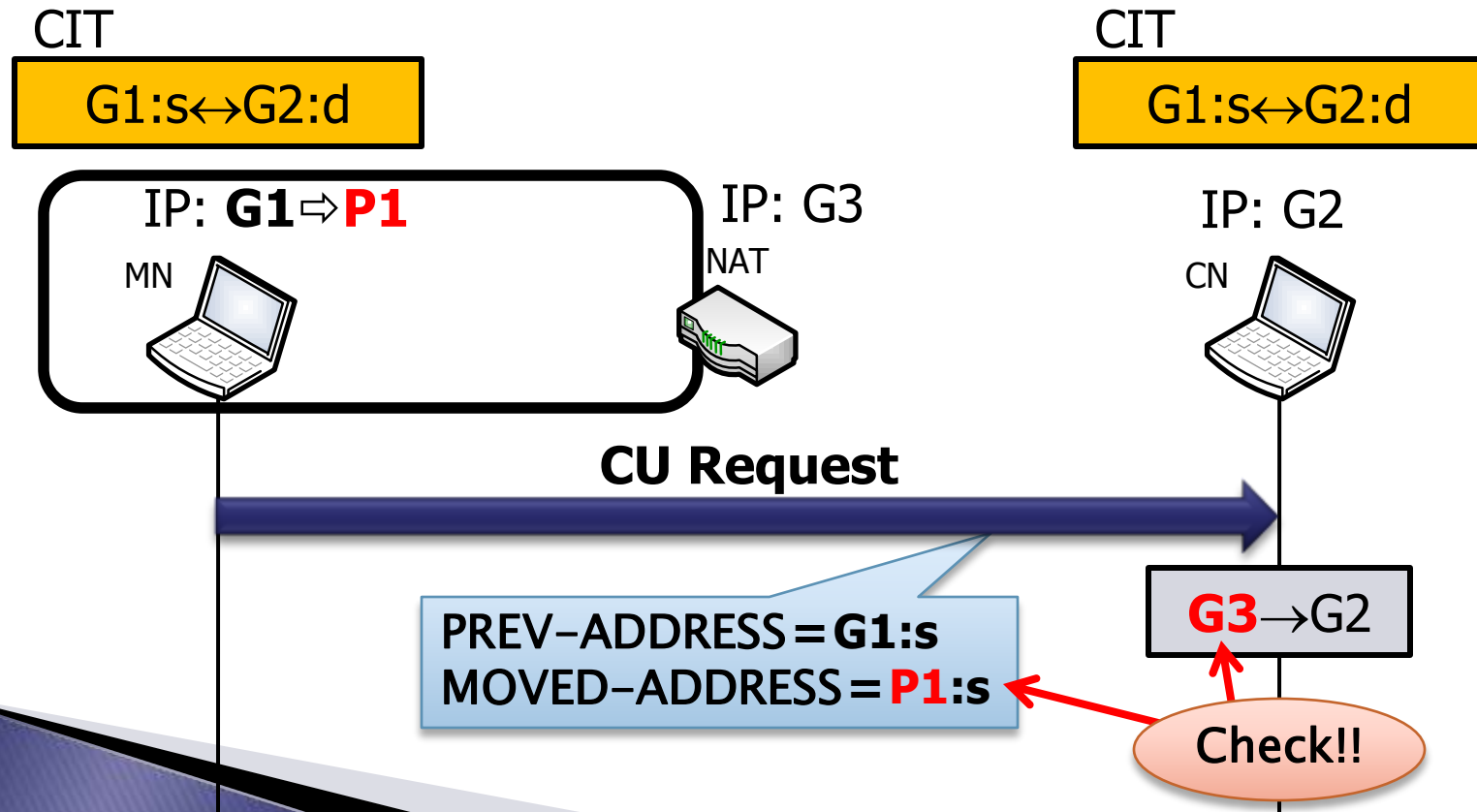
通信開始時

- ▶ 認証鍵を共有
 - 従来方式と同様
 - 通信を開始（例：UDP）



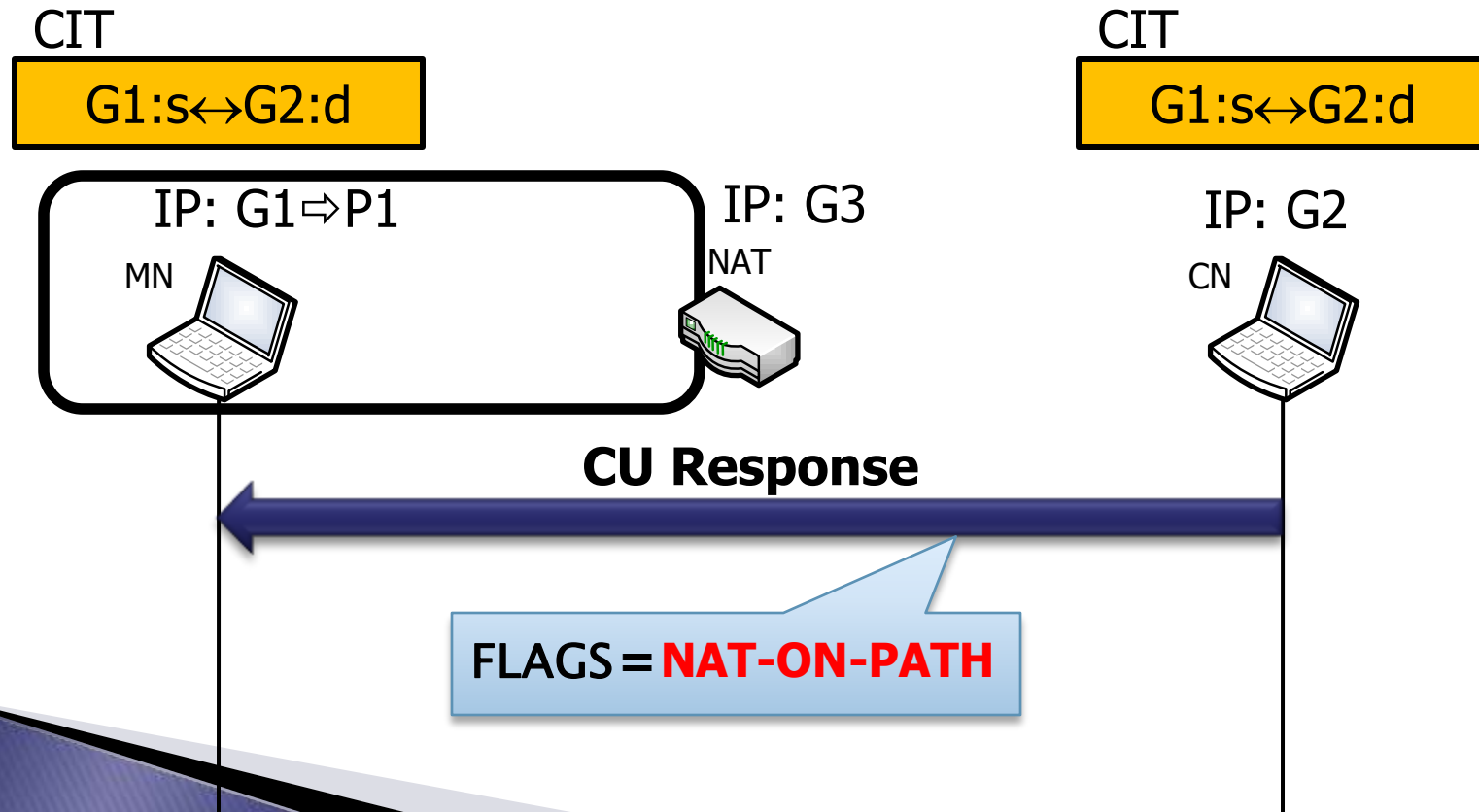
移動通知処理

- ▶ プライベートIPアドレスに変化
 - 従来方式と同様, CU Requestを送信
 - CNは**MOVED-ADDRESS**と**送信元IPアドレス**をチェック



Private NWへの移動

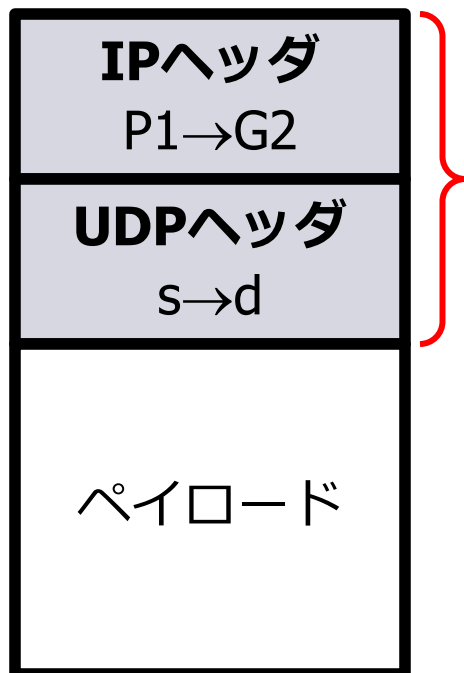
- ▶ アドレス不一致 → **NATが存在**
 - CNはCITは更新せず, CU Responseを返信
 - **NAT-ON-PATH** : NATが存在することを示す



Binding Requestの生成

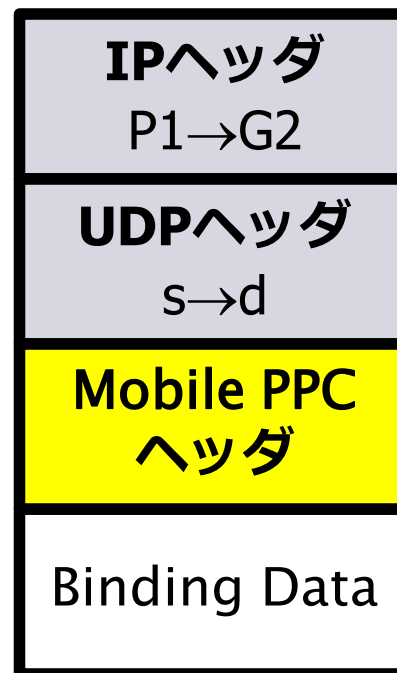
- ▶ 実際の通信パケットと同じL3/L4ヘッダ
 - L4ヘッダの次にMobile PPCヘッダを配置

移動後にMNが送信する
通信パケット



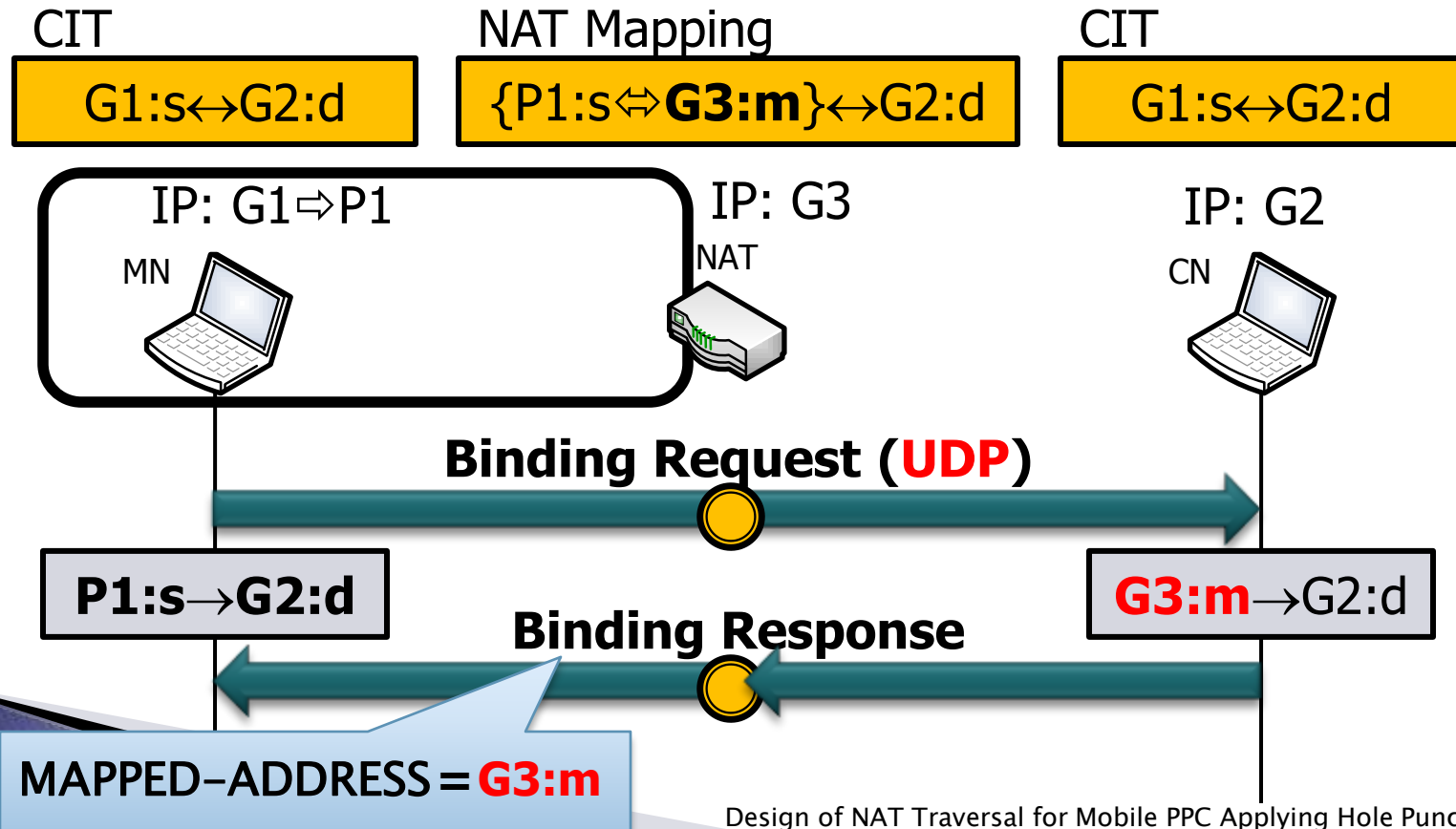
=

Binding Request



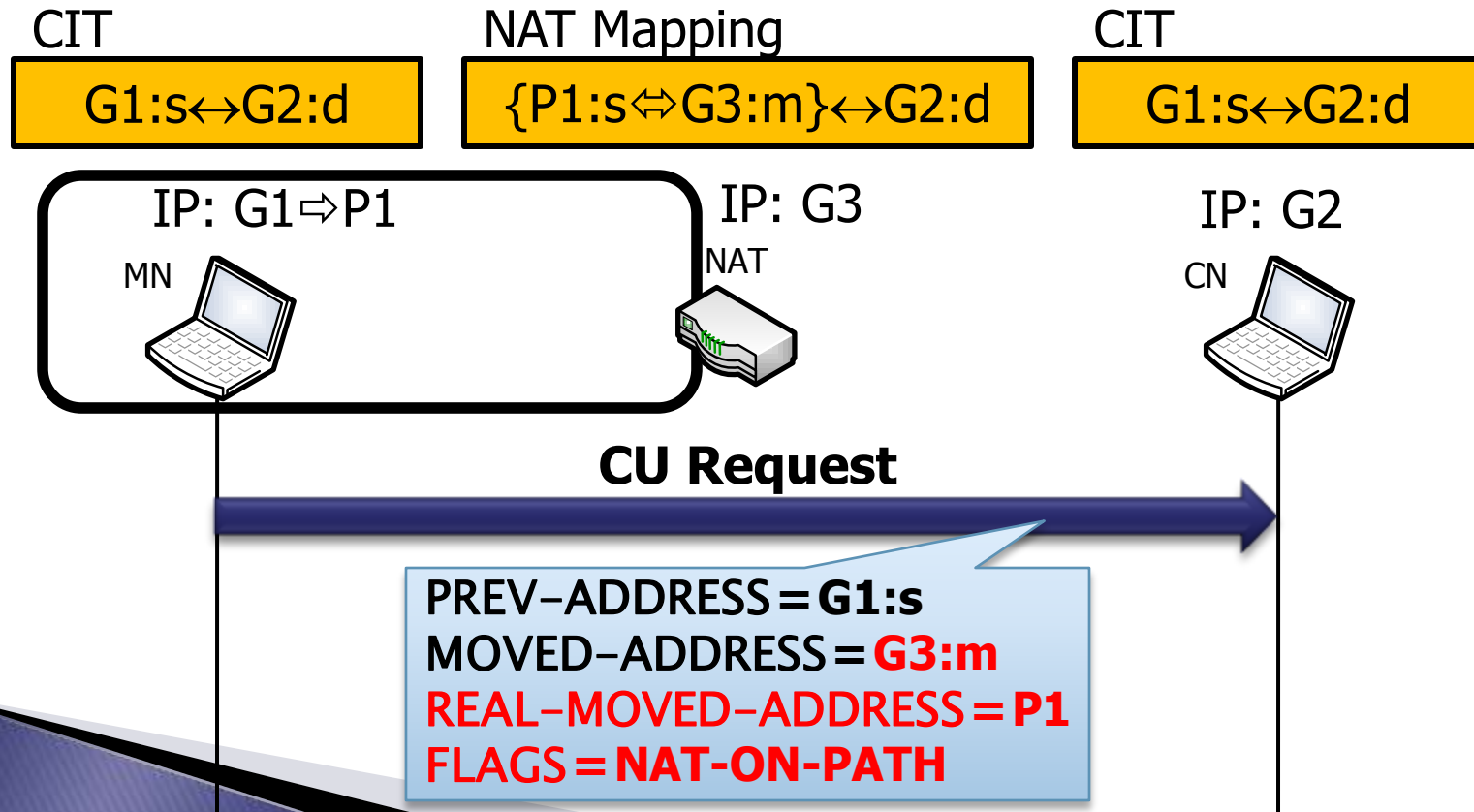
Binding処理

- ▶ Binding RequestをCNへ送信→マッピング生成
- ▶ CNはBinding Responseを返答
→ **MAPPED-ADDRESS**を通知



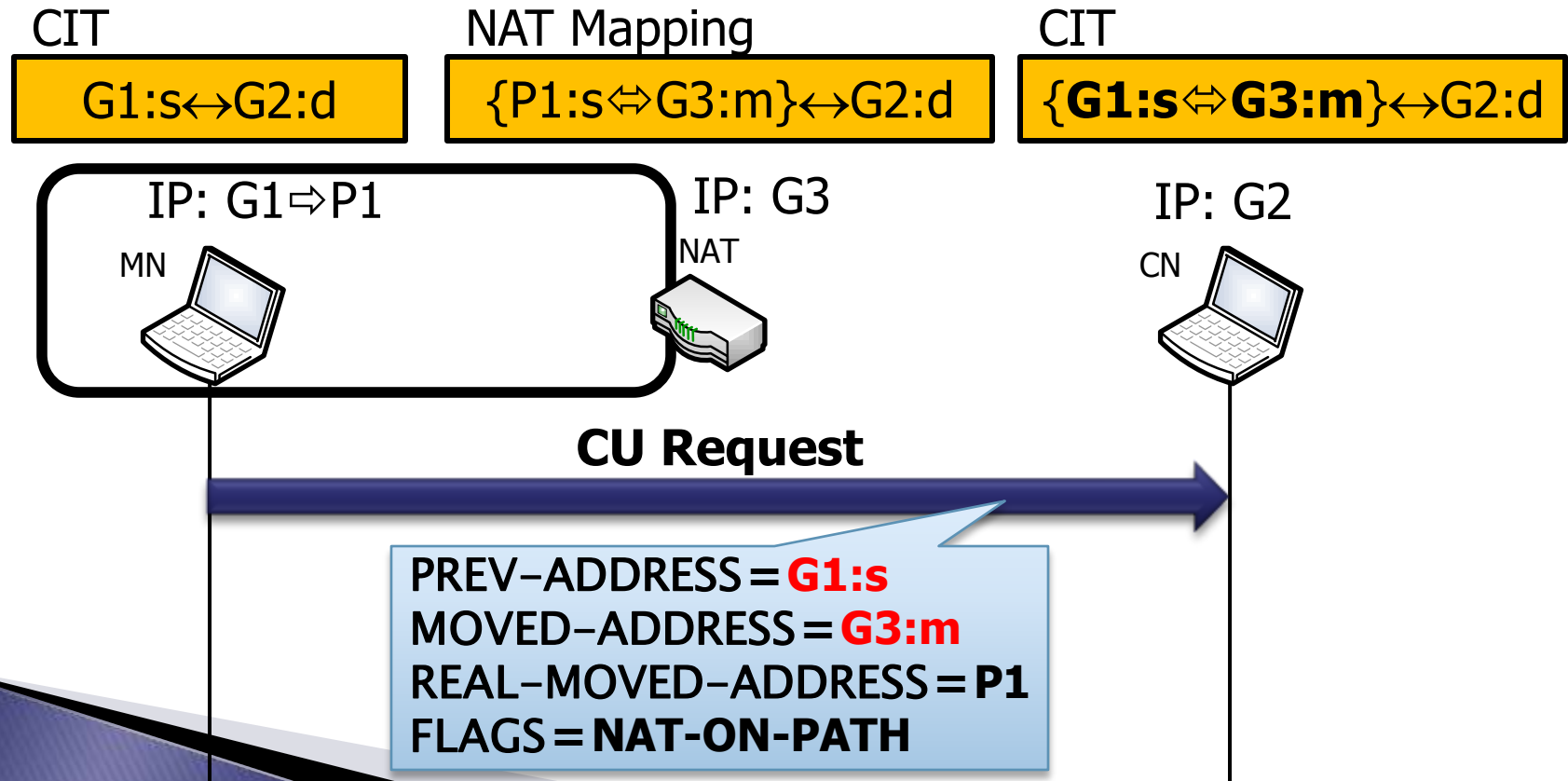
移動通知処理 (2回目)

- ▶ CU RequestをCNへ送信 (**NAT-ON-PATH**付)
 - 取得したMAPPED-ADDRESS → **MOVED-ADDRESS**
 - MNのIPアドレス → **REAL-MOVED-ADDRESS**



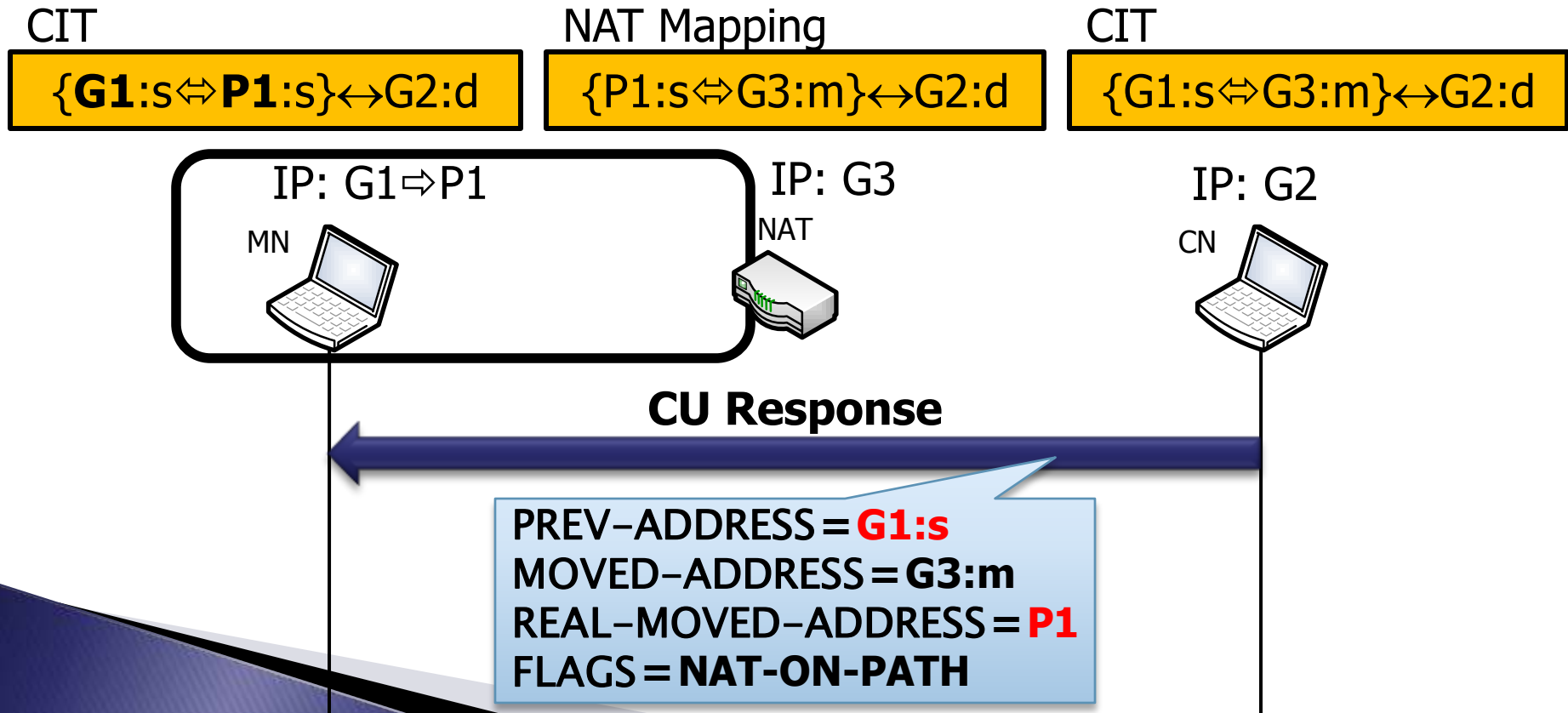
移動通知処理 (2回目)

- ▶ CNは従来と同様にCITを更新
 - **PREV-ADDRESS**と**MOVED-ADDRESS**を選択
- ▶ CU Responseを応答



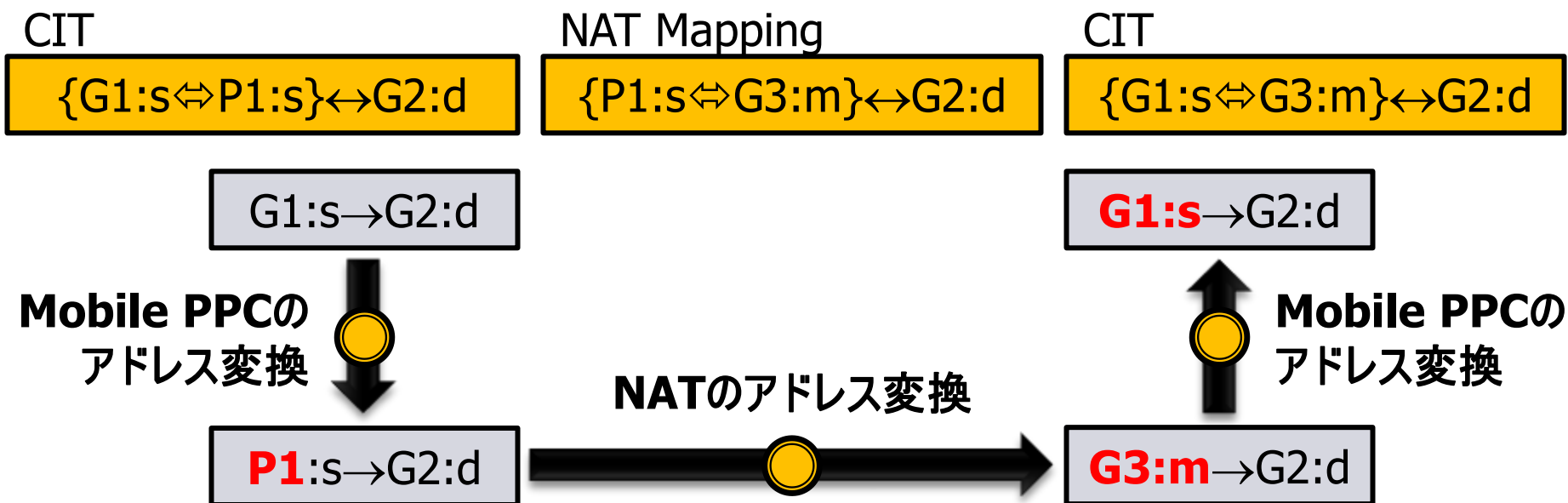
移動通知処理 (2回目)

- ▶ MNは以下の情報でCITを更新
 - **PREV-ADDRESS**と**REAL-MOVED-ADDRESS**を選択



アドレス変換処理

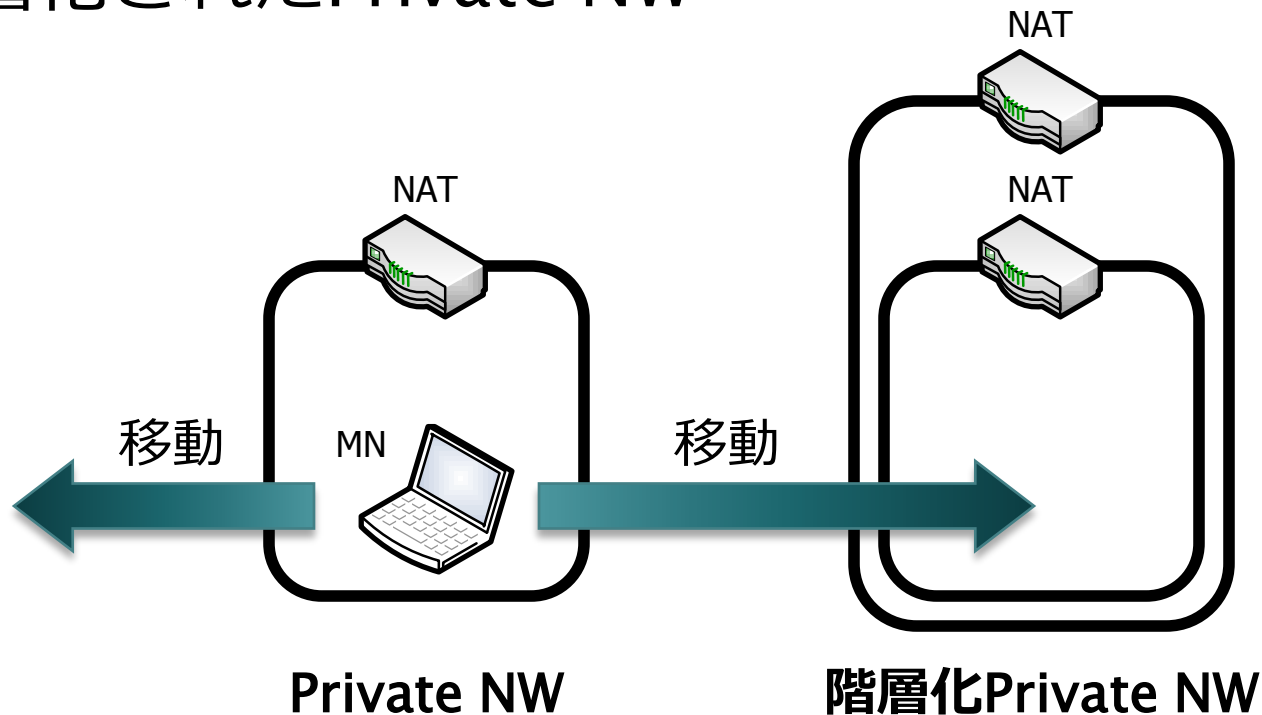
- ▶ CITによるアドレス変換は従来方式と同じ



グローバルネットワークから
プライベートネットワークに移動可能!!

提案方式が対応する移動ケース

- ▶ Private NWからGlobal NWへ
- ▶ Private NWからPrivate NWへ
- ▶ 階層化されたPrivate NW



- ▶ Coreな部分の実装は完了
 - Binding処理

市販ブロードバンドルータを用いて動作検証を行った結果、通信を継続できることを確認

- ▶ 未完成部分
 - セキュリティ対策
 - Binding Request/Responseメッセージの署名
(原稿には記載しているが発表できなかった部分)

- ▶ Mobile PPCにHole Punchingの原理を適用
 - IPレベルでBinding処理をEnd-to-Endで実行
 - ➔ 移動後にマッピングされたアドレスを通知

既存のNATをそのまま利用してグローバルネットワークとプライベートネットワークを相互に移動可能

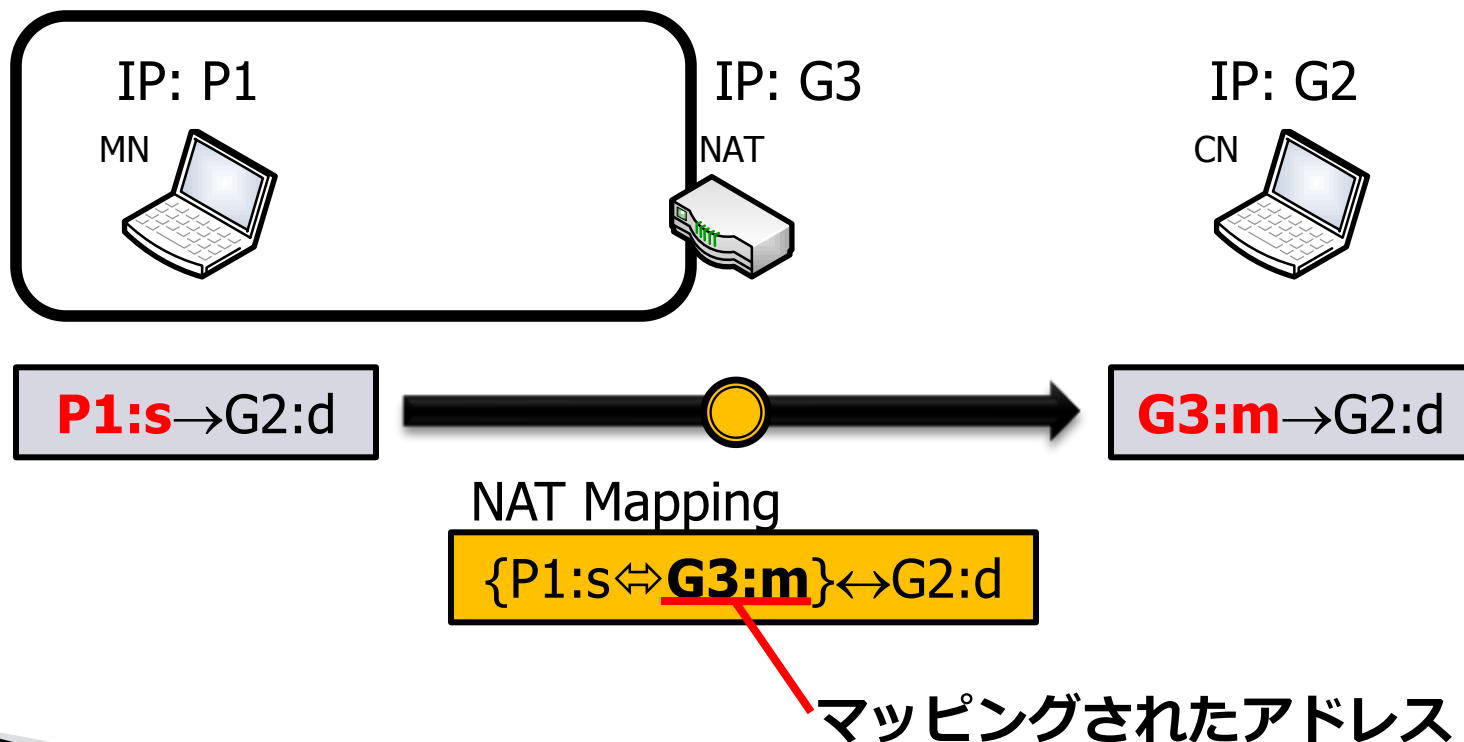
- ▶ 今後の課題
 - セキュリティ部分の実装
 - CNがプライベートネットワーク内に存在する場合の対応

付録

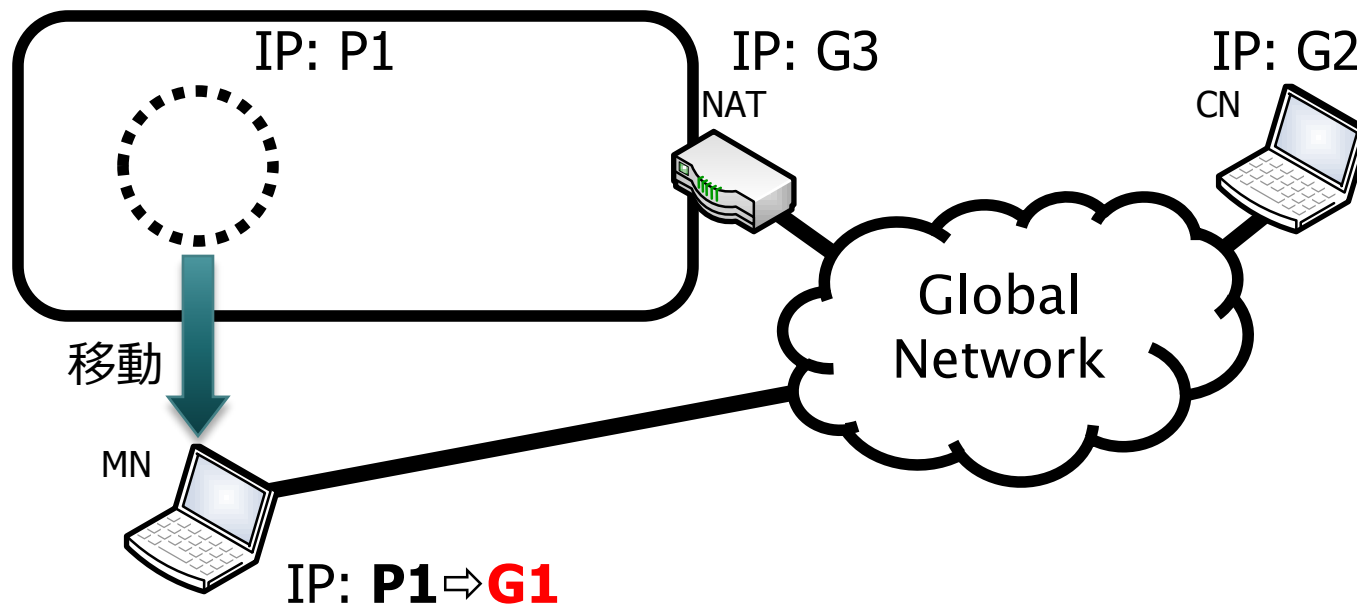
通信経路上にNATが存在すると...

▶ NATによるアドレス変換

- 外部宛パケットの転送時にNATマッピング情報を生成
- パケットのIPアドレス・ポート番号を変換



Global NWへの移動

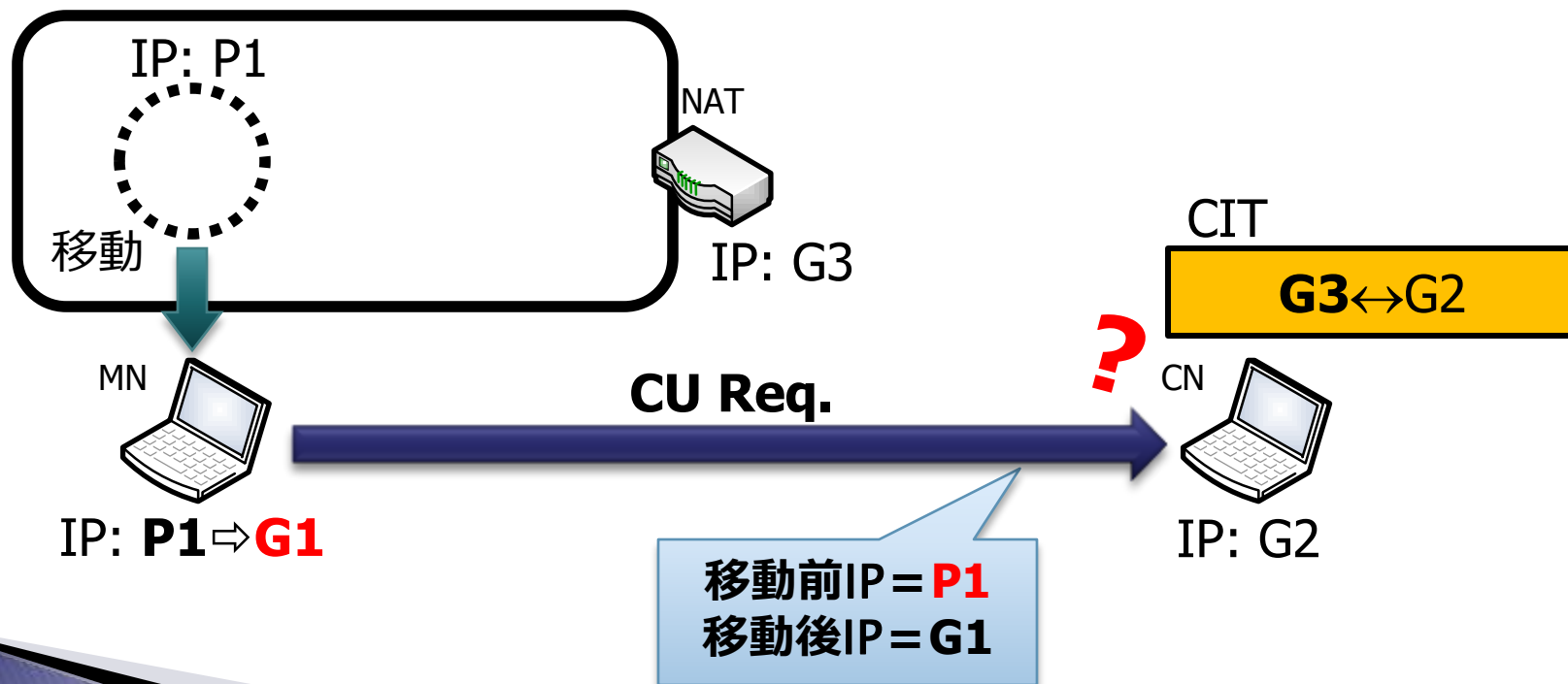


考え方はPrivate NWへ移動する場合と同じ

通信経路上にNATが存在すると...

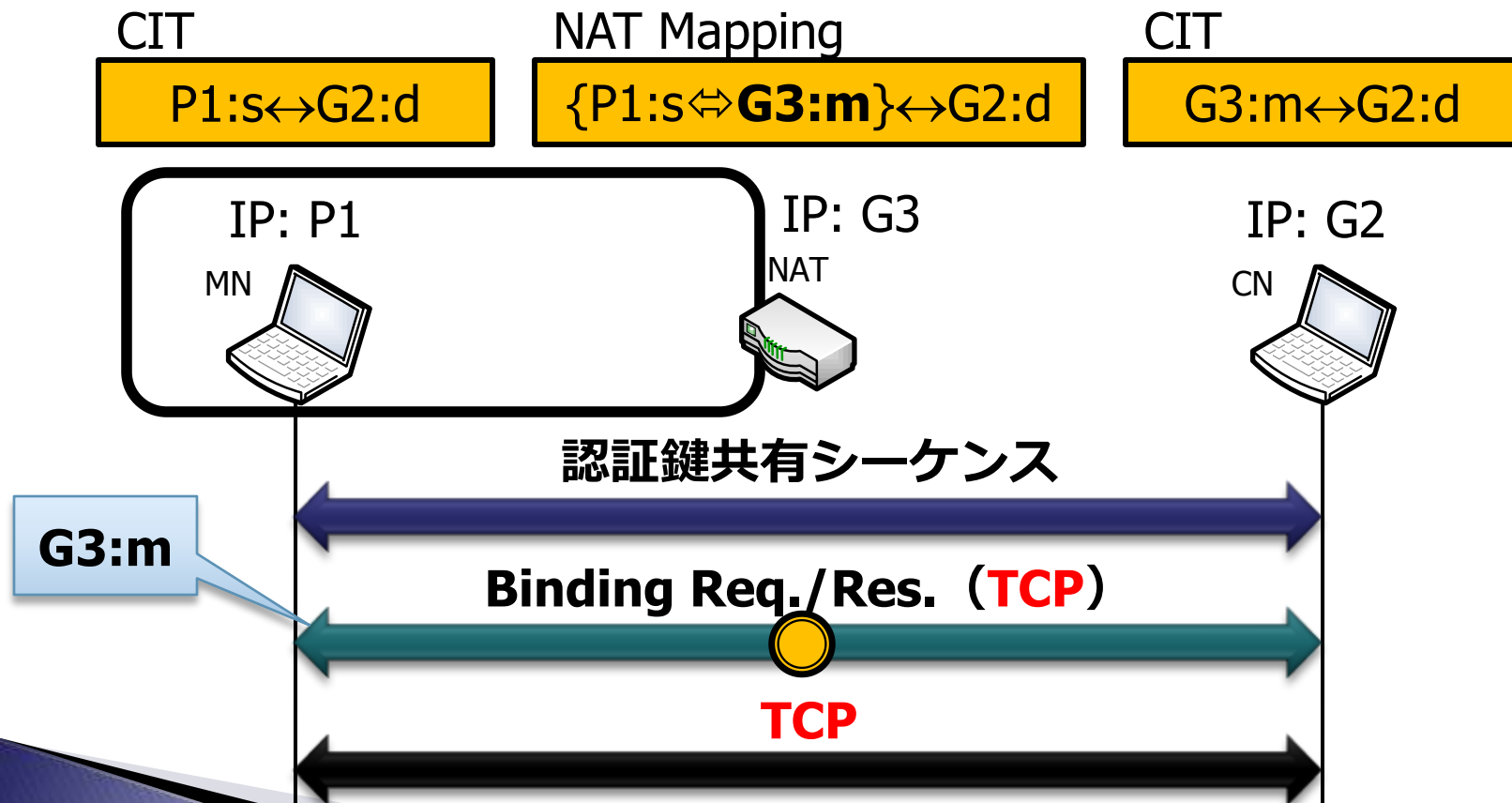
課題3 :

CNが認識してる移動前IPアドレス = NATのグローバルIP
→ **正しくCITを更新できない**



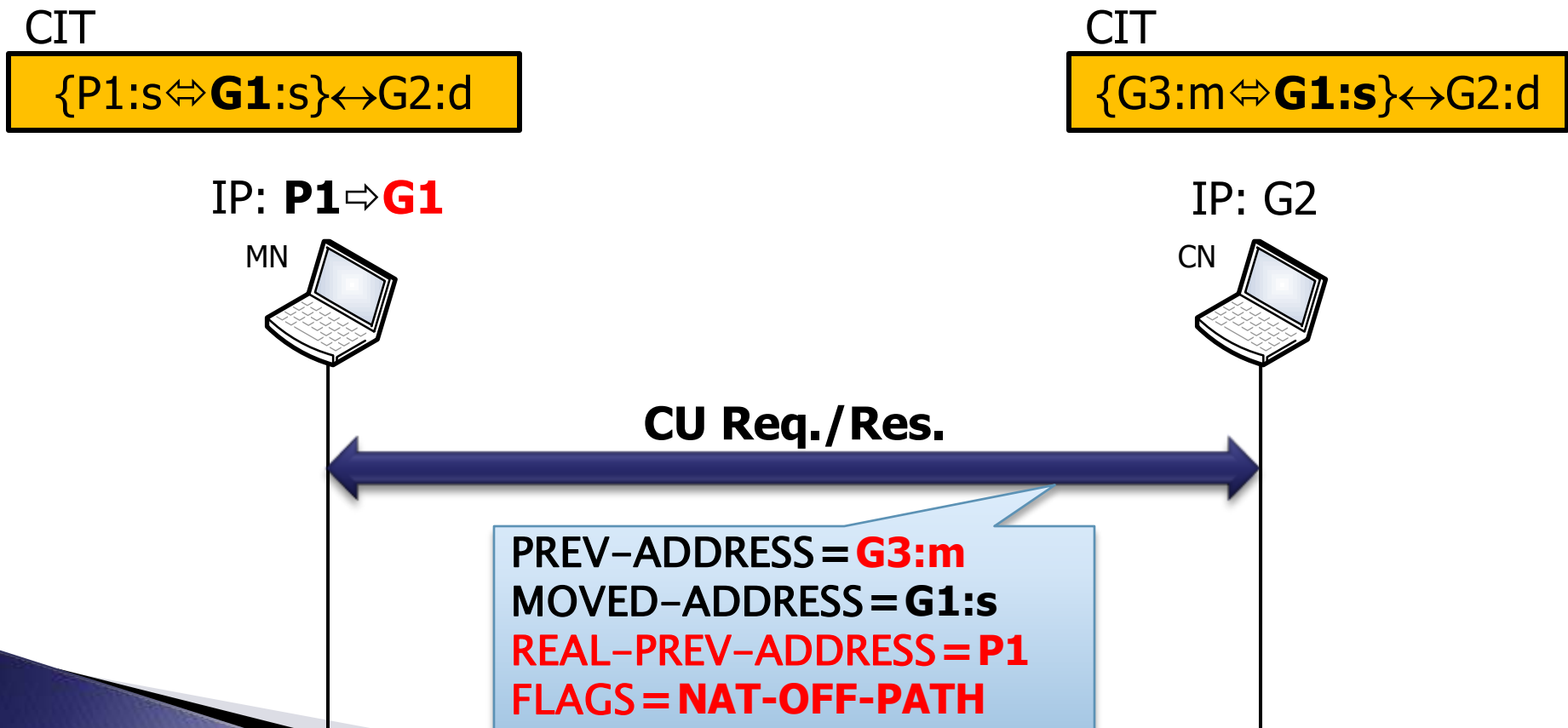
Global NWへの移動

- ▶ 通信開始時（NATあり，例：TCP）
 - 認証鍵共有シーケンス後にBinding処理



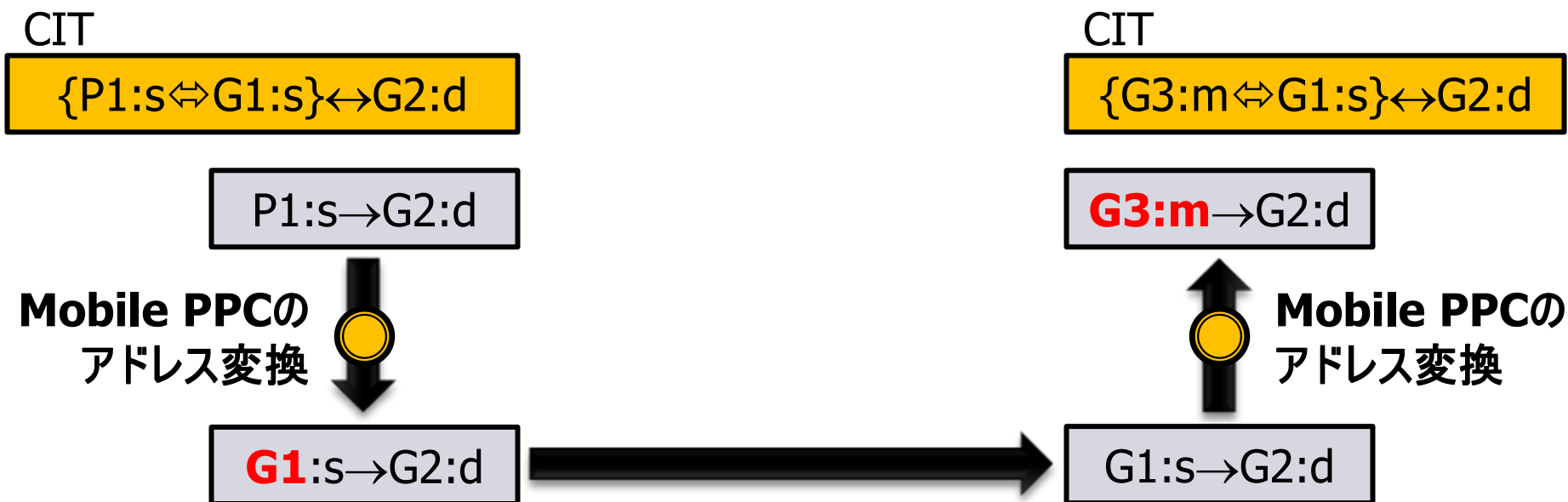
Global NWへの移動

- ▶ グローバルネットワークへ移動
 - 移動通知処理 (**NAT-OFF-PATH**フラグ付)



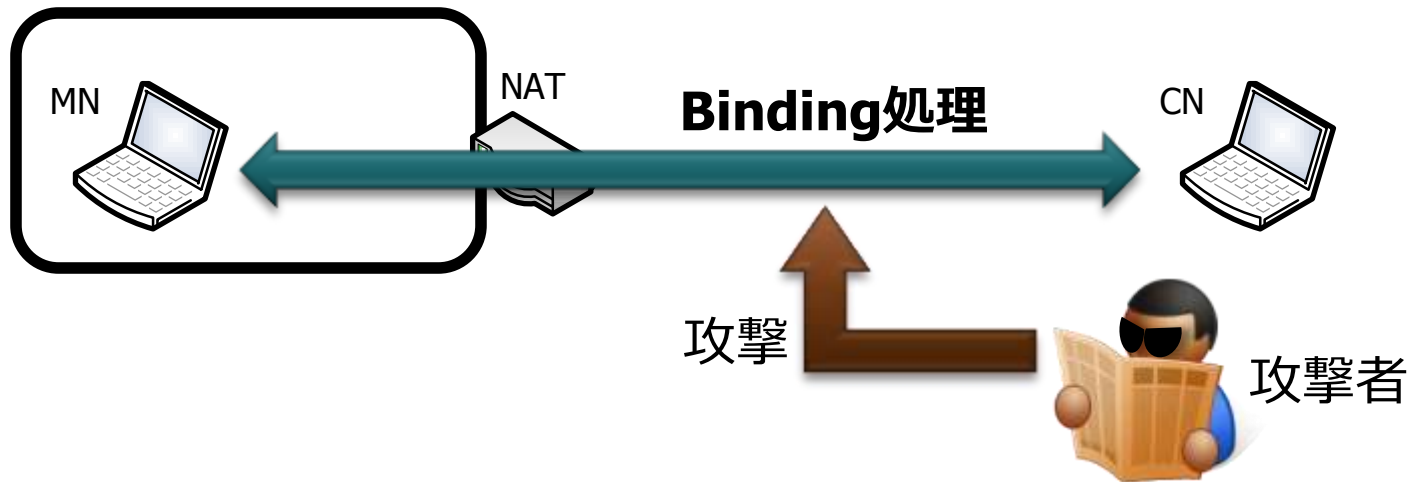
Global NWへの移動

- ▶ CITによるアドレス変換は従来と同じ



プライベートネットワークから
グローバルネットワークに移動可能!!

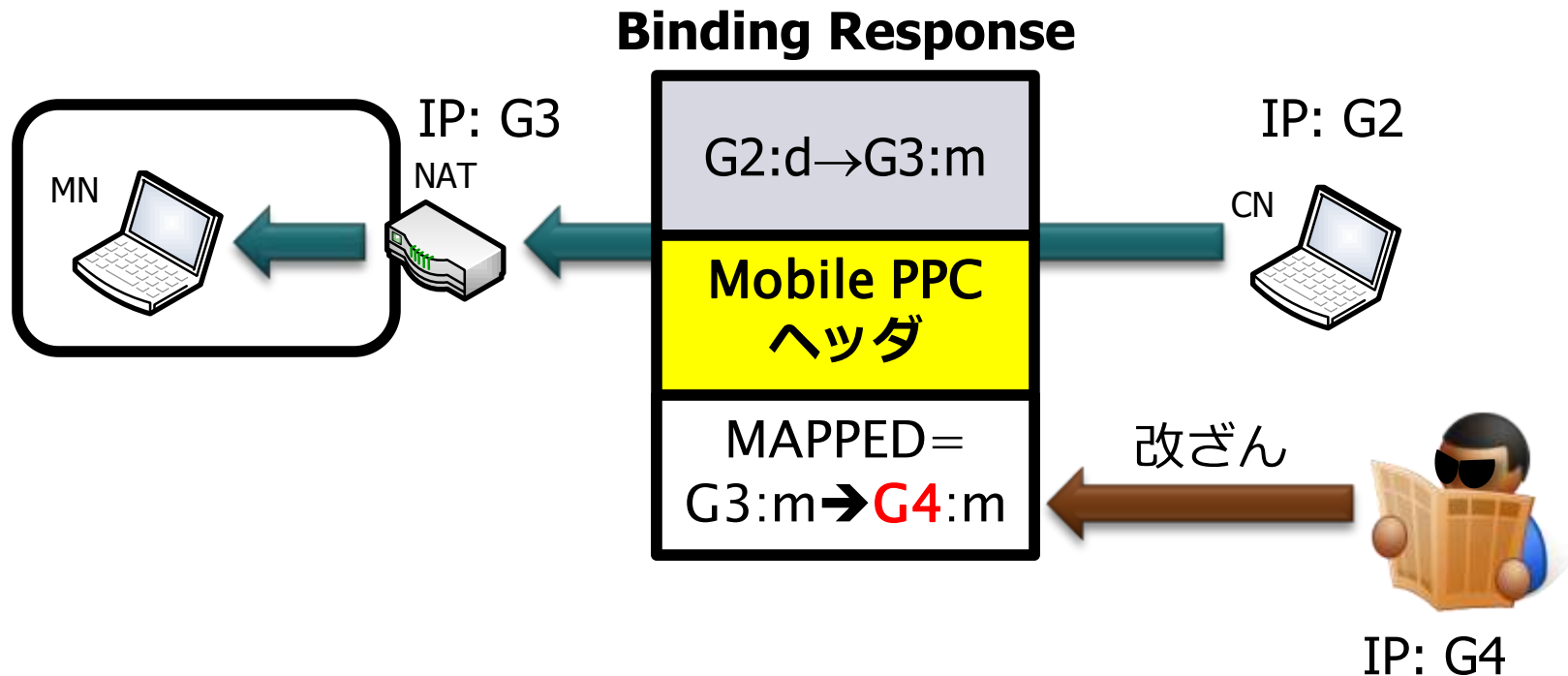
- ▶ Binding処理に対する攻撃に着目
 - セッションハイジャックに対する安全性・対策



認証鍵の共有が安全に行われていることを前提

セキュリティ(1)

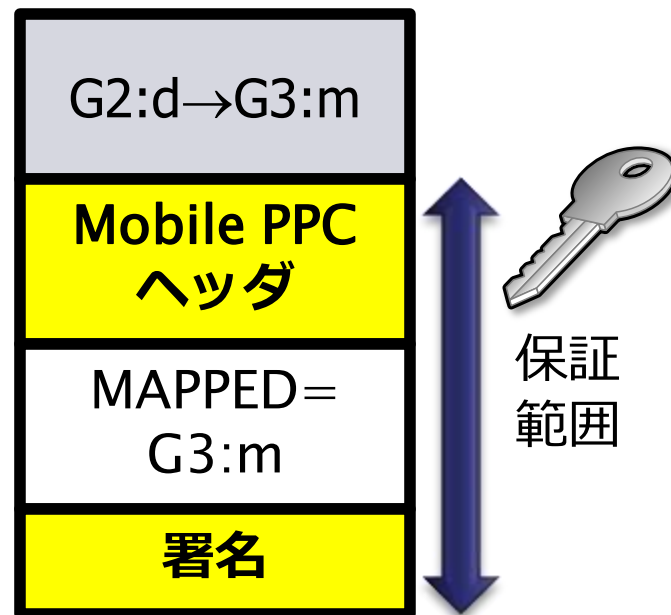
- ▶ **攻撃(1) : MAPPED-ADDRESSの改ざん**
 - MNは移動後, 攻撃者のIPアドレスをCNに通知
→ CIT更新後にセッションハイジャックされてしまう



セキュリティ(1)

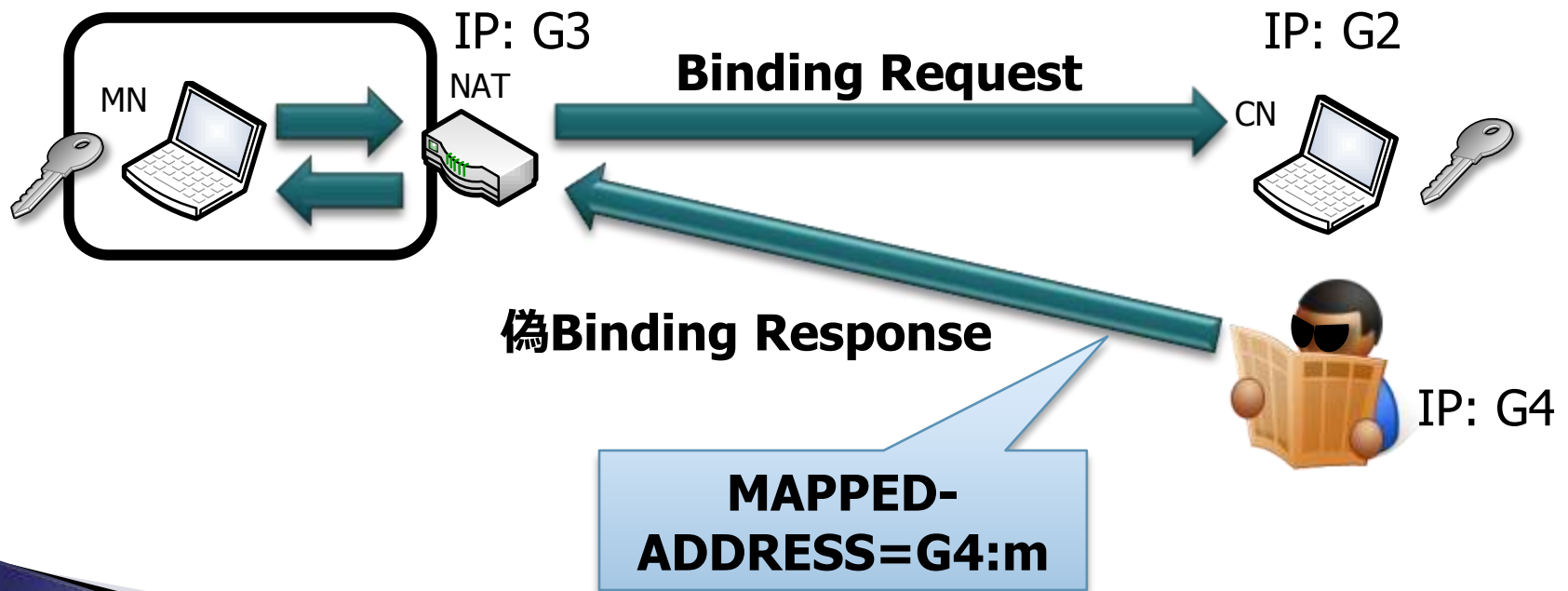
- ▶ 認証鍵による署名を付加
 - メッセージ完全性を保証
- ➔ Bindingメッセージペイロードの改ざんを検出可能

Binding Response



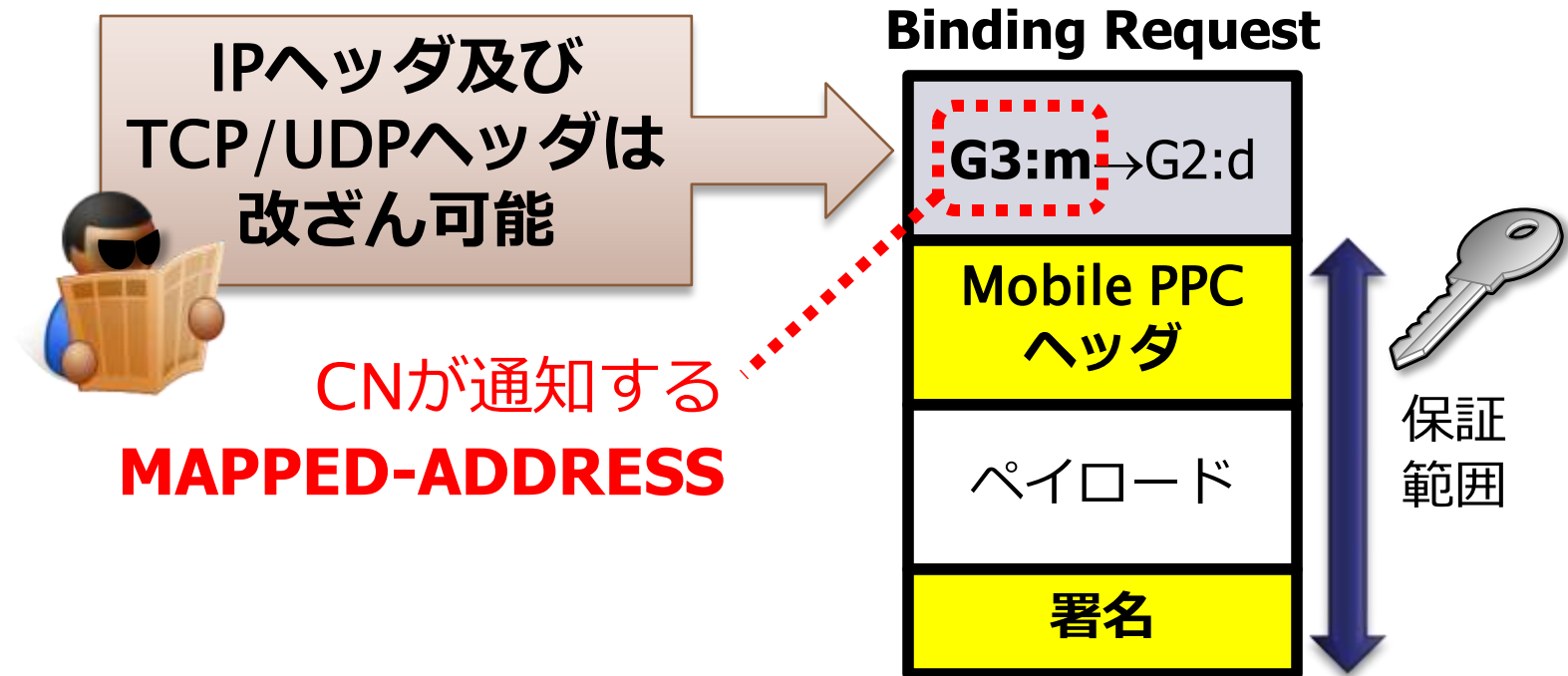
セキュリティ(2)

- ▶ **攻撃(2) : 偽のBinding Responseを送信**
 - MNのBinding Request送信後, 攻撃者が直ちに送信
 - ➔ **MNにおけるメッセージ完全性の検証過程で検出可能**
(攻撃者は正しい署名を付加できないため)



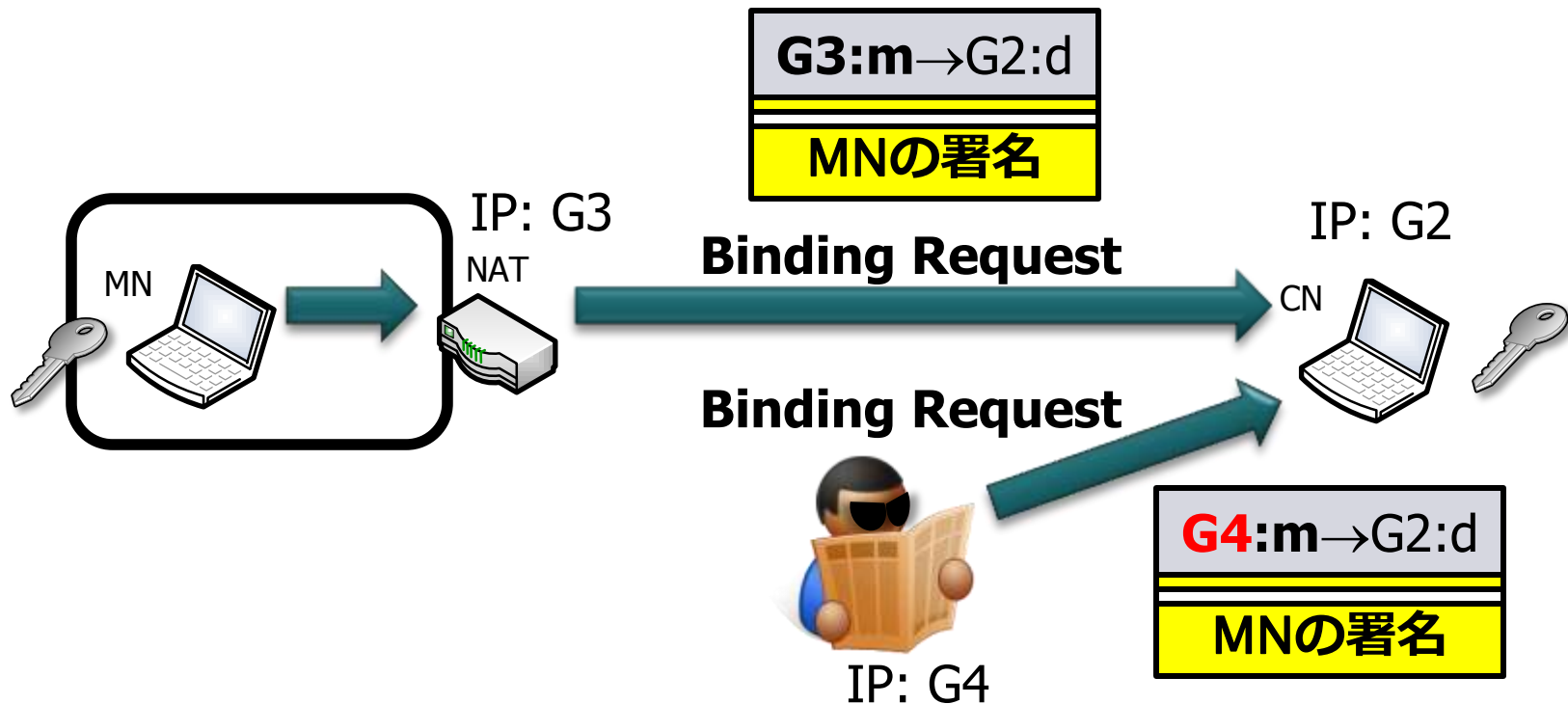
セキュリティ(3)

- ▶ BindingメッセージのL3/L4ヘッダは保証範囲外
 - NAT通過時にアドレス変換されるため



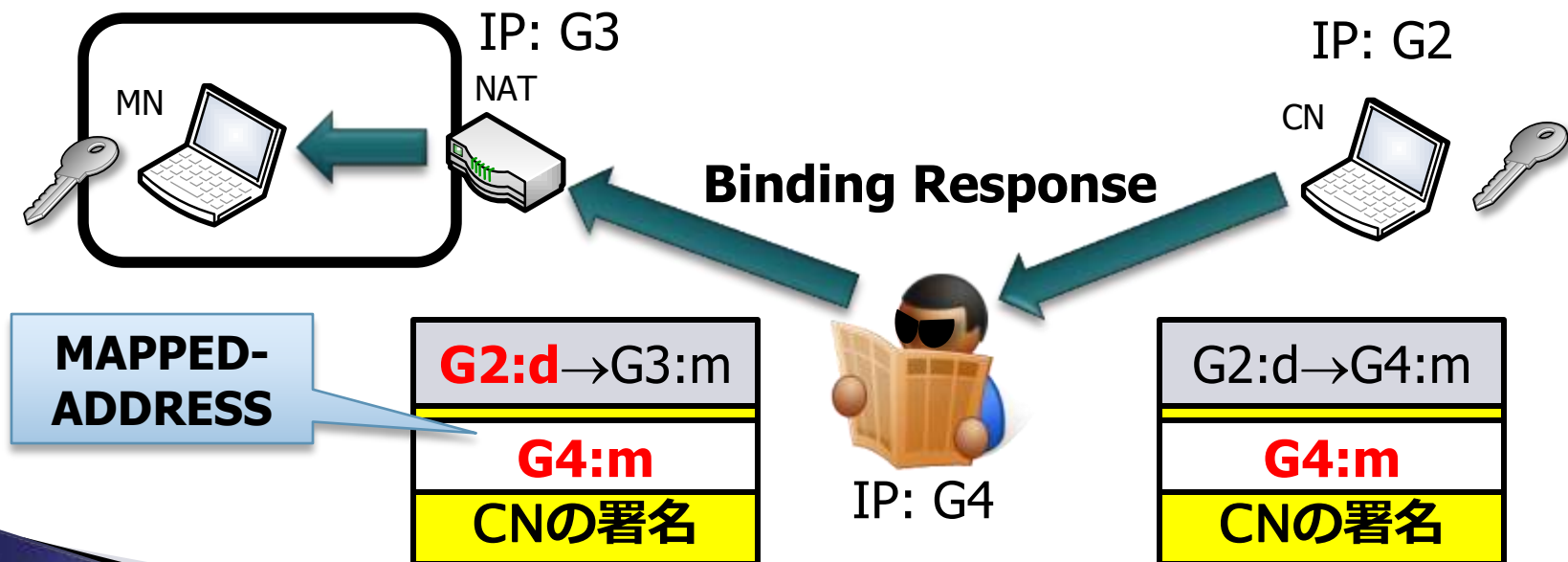
セキュリティ(3)

- ▶ **攻撃(3) : 改ざんされたBinding Requestを送信**
 - 攻撃者は正規のBinding Req.を盗聴し, 送信元IPを詐称
→ CNは正規のメッセージと判断



セキュリティ(3)

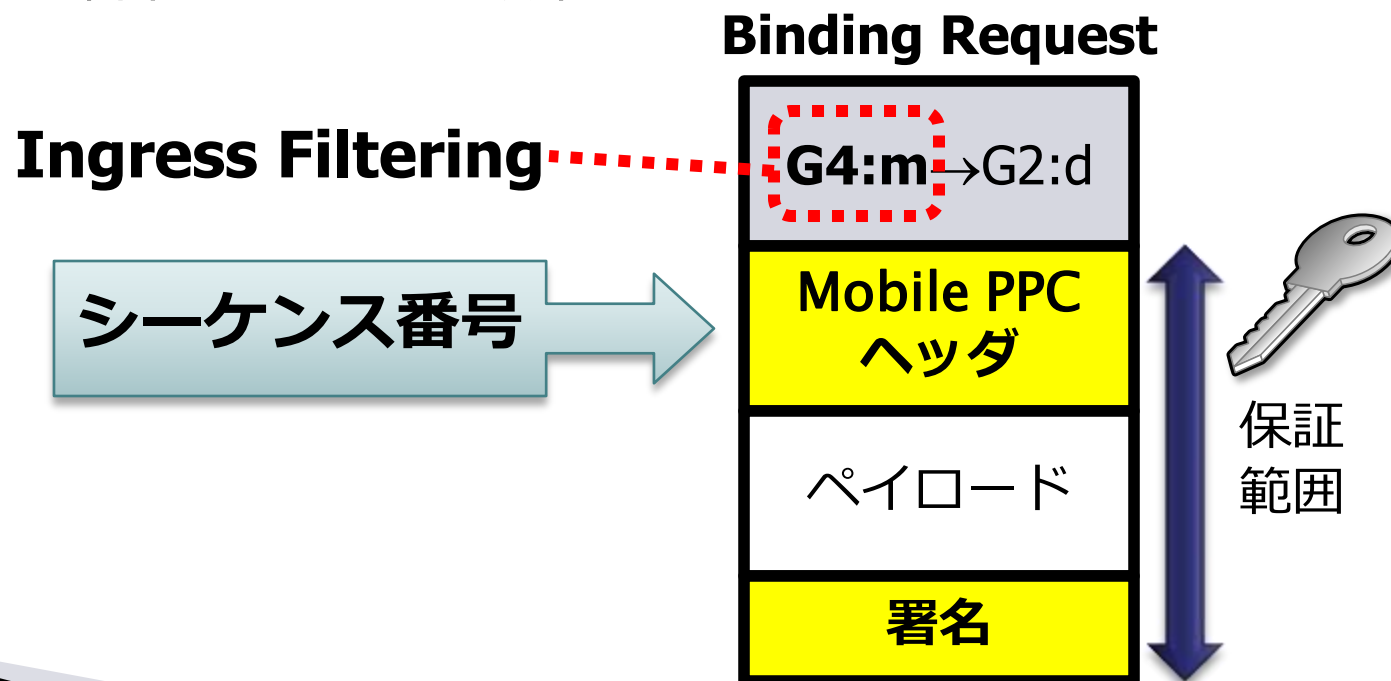
- ▶ CNは攻撃者にBinding Responseを返信
 - 攻撃者は送信元をCNに詐称して転送
 - MNは正規のメッセージと判断し，移動通知処理を実行
- ➔ CIT更新後にセッションのハイジャックが可能



セキュリティ(3)

▶ 対策

- 再送攻撃の防止
 - Bindingメッセージにシーケンス番号を設定
- Ingress Filtering
 - 送信元詐称パケットの破棄



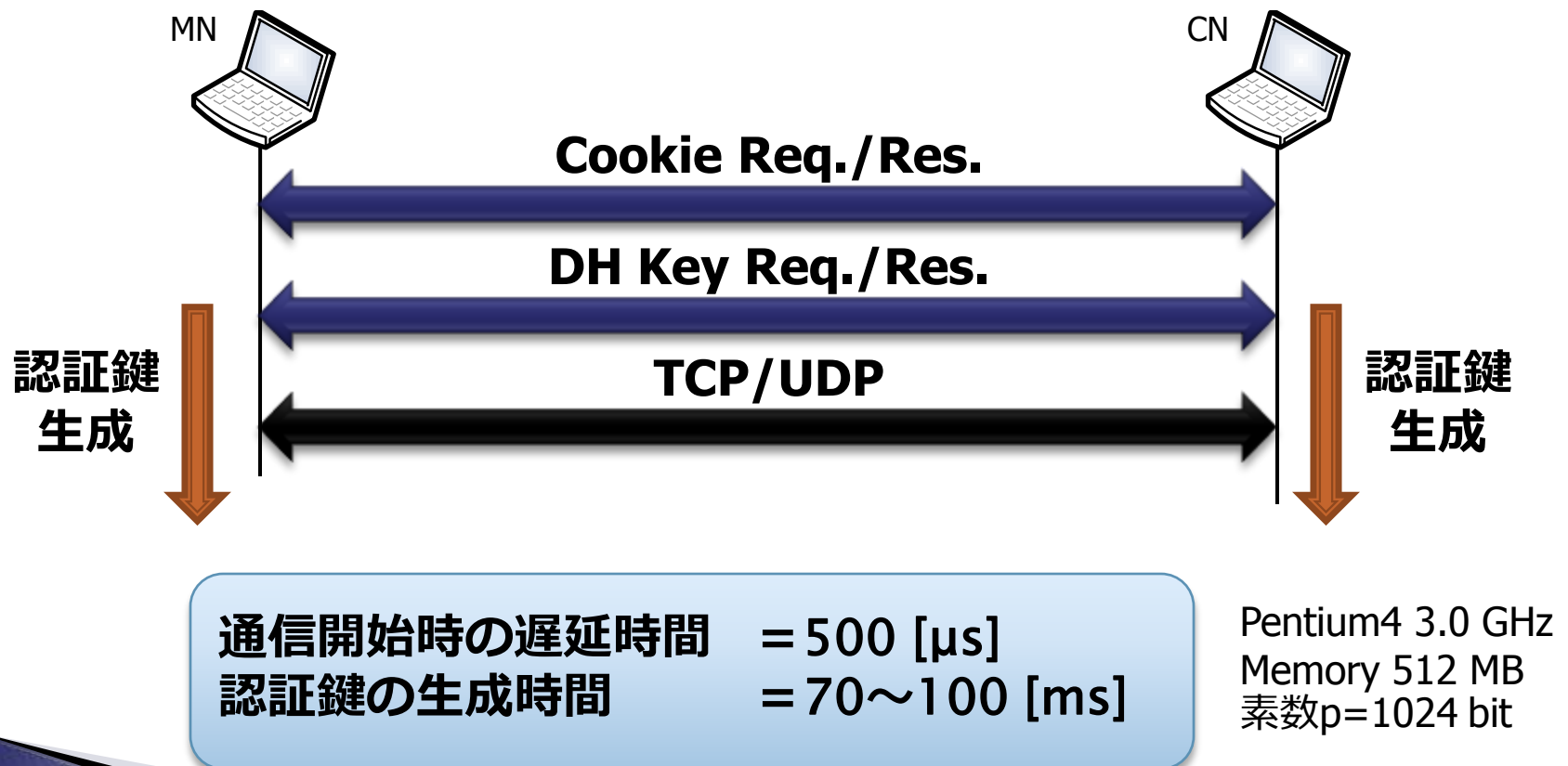
認証鍵共有シーケンス

- ▶ DH鍵交換をベース（別研究にて実施）
 - Cookie Request/Response
 - DH Key Request/Response



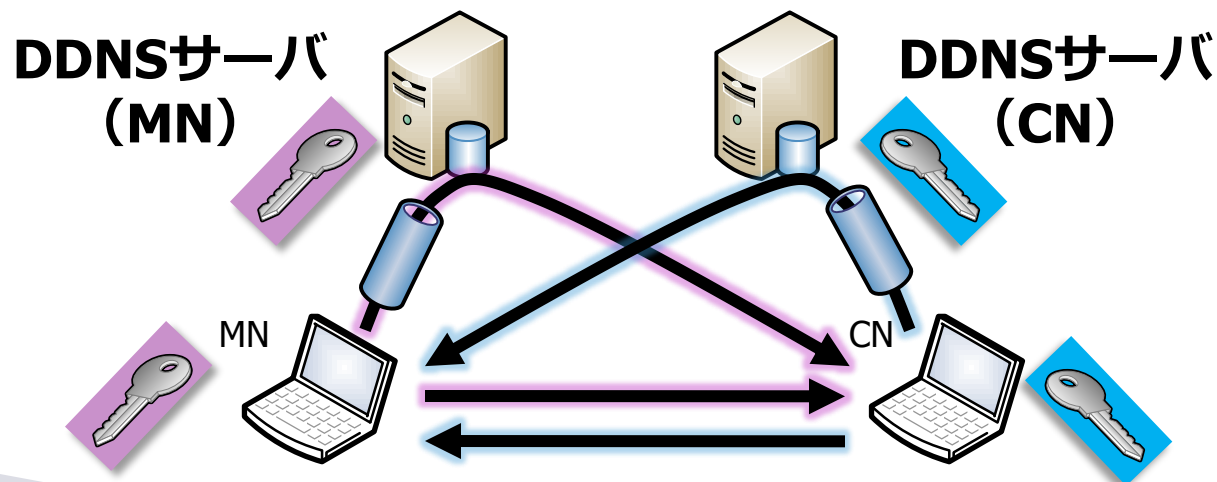
認証鍵の生成

- ▶ 通信の開始と平行して認証鍵を生成
→ 認証鍵生成による遅延を抑えることに成功



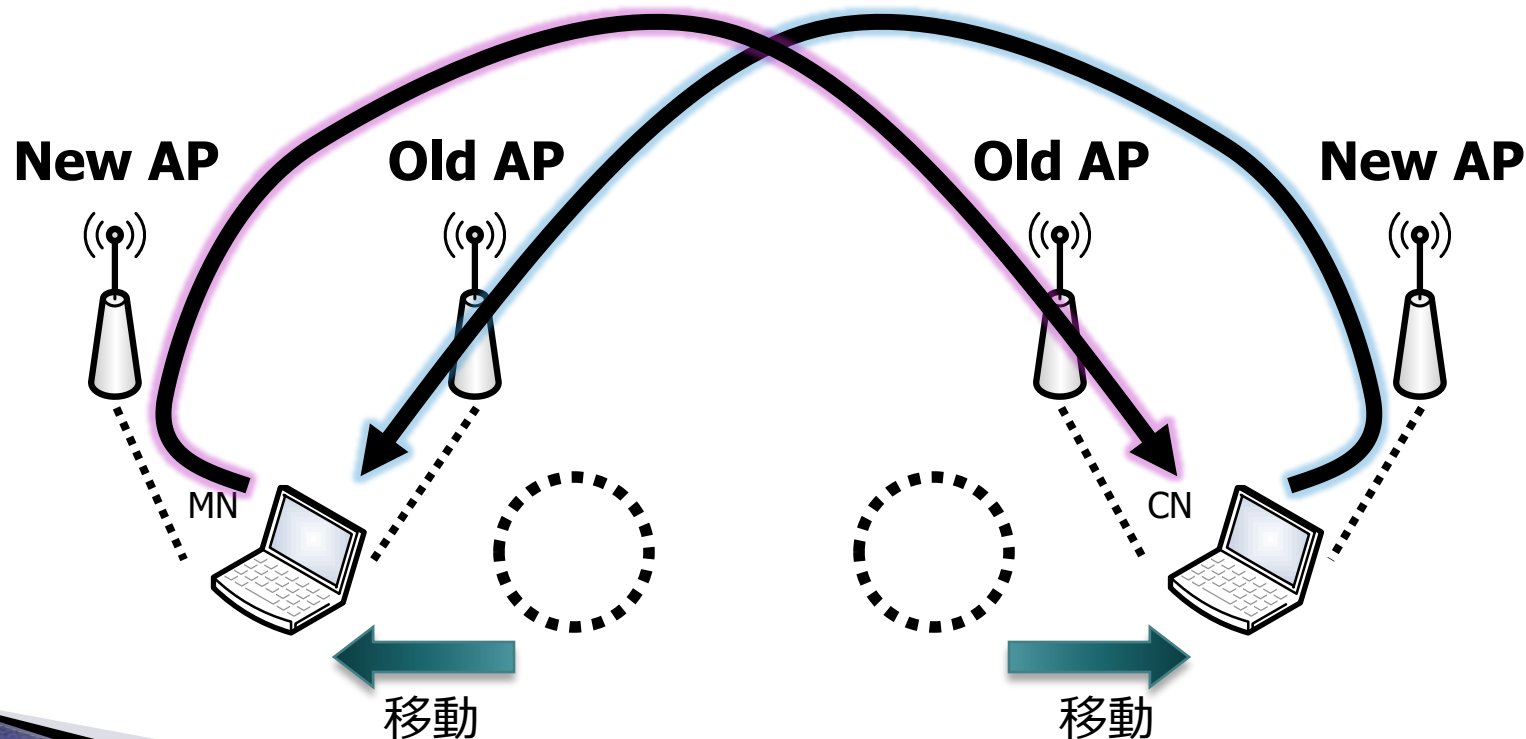
認証鍵共有時の安全性

- ▶ End-to-Endによる鍵交換
 - **中間者攻撃**に対して脆弱性あり
- ▶ 対策
 - **Dynamic DNSサーバを中継する鍵交換**
 - ・ エンドノードとセキュアな通信を形成するのは比較的容易
 - ・ DH鍵を分割して「**中継**」と「**直接**」の2経路を利用



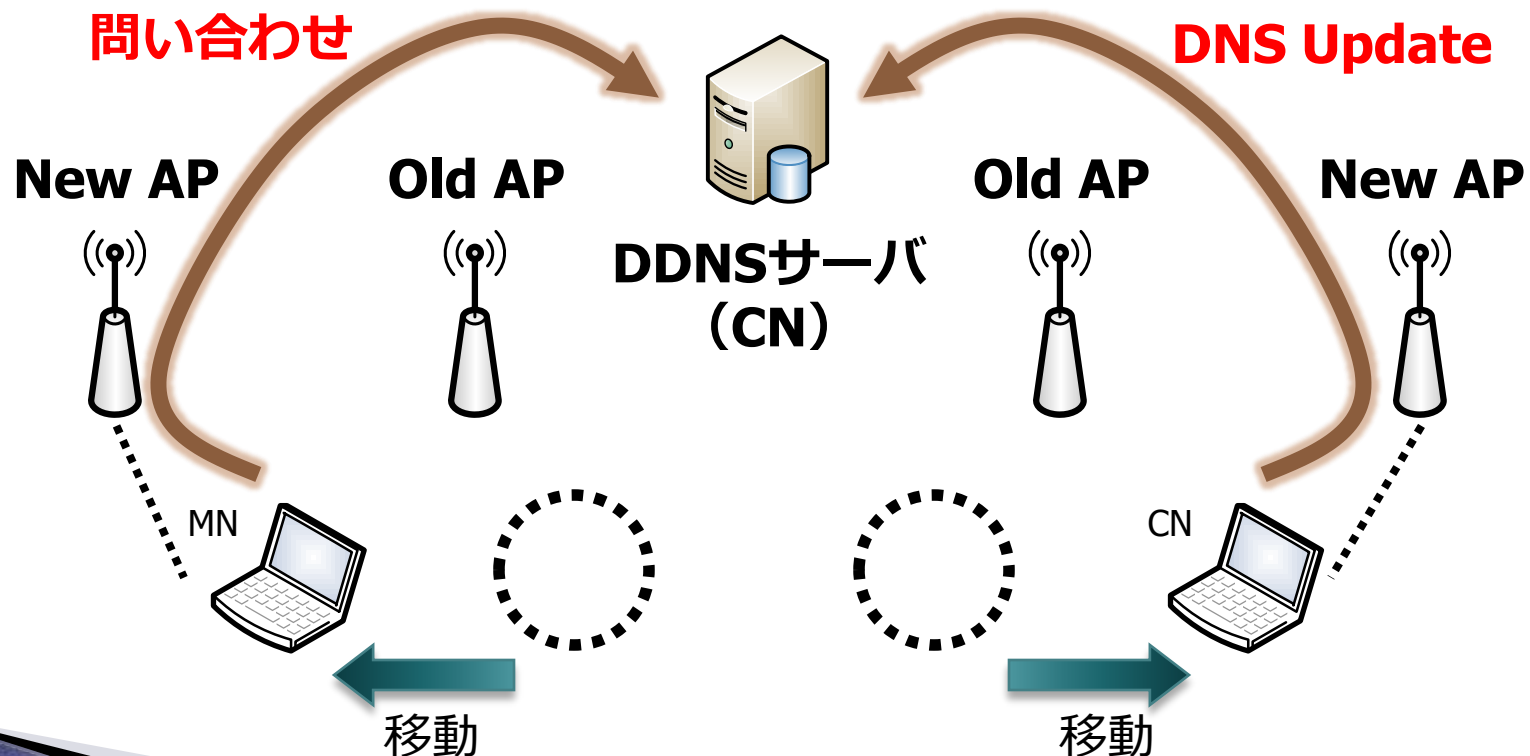
Double Jump

- ▶ 複数インタフェースにより解決（別研究にて実施）
 - MN及びCNをマルチホーム化
 - ➔ 新しいリンクを確立後，一定時間は古いリンクからの受信も可能



Double Jump

- ▶ 移動後にDynamic DNSサーバへUpdate
 - 通信開始時のIPアドレス解決のため
 - ➔ 相手からの応答がない場合はDDNSサーバに問い合わせ



Double Jump

- ▶ 新しいIPアドレスに向けてネゴシエーション
 - 遅延は発生するがネゴシエーションは可能

