UDPとICMPの深いぃ話

TCP/IP再認識～忘れちゃいけないUDP、ICMP
Internet Week 2014
2014/11/18
松平直樹
富士通株式会社

自己紹介

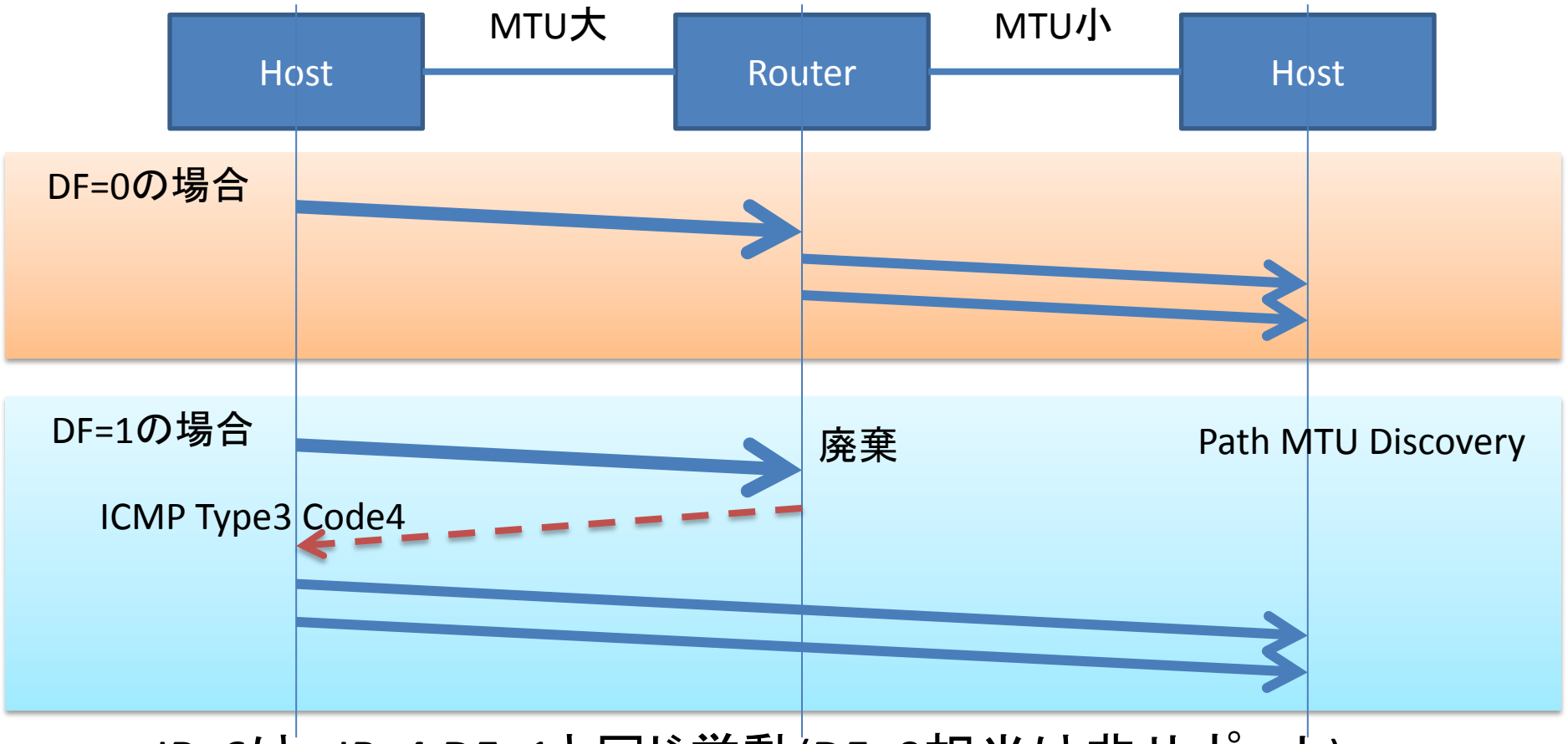
- 1986年(株)富士通研究所入社、1996年富士通(株)に転社
- 1993年頃からTCP/IPを業務に
  - 以後、TCP/IP関連業務(主としてルータ)を担当
  - それ以前は民需伝送系製品や情報系通信機器等の研究開発を担当
  - 一貫してデータネットワークに携わる
- IETF会議に、第34回会議(1995年12月)より出席
  - アジア初の開催となった2002年54th IETF横浜会議に深く関与
- 次世代インターネットプロトコル対応
  - KAMEプロジェクト、「IPv6ネットワーク実践構築技法」(オーム社)
- この数年：SA46T技術ファミリーのIETF提案等
- 「フラグメンテーションの今後を考えよう」@JANOG33.5
  - http://www.janog.gr.jp/meeting/janog33.5/program/index.html

目次

1. フラグメンテーションのおさらい
2. IPv4 - IPv6移行技術とフラグメンテーション
   – カプセル化(IPv4 over IPv6, IPv6 over IPv4)
   – IPv4-IPv6変換
3. クラウドとフラグメンテーション
   – GRE, VxLAN, NVGRE
4. IETFでの最近の議論と対処法

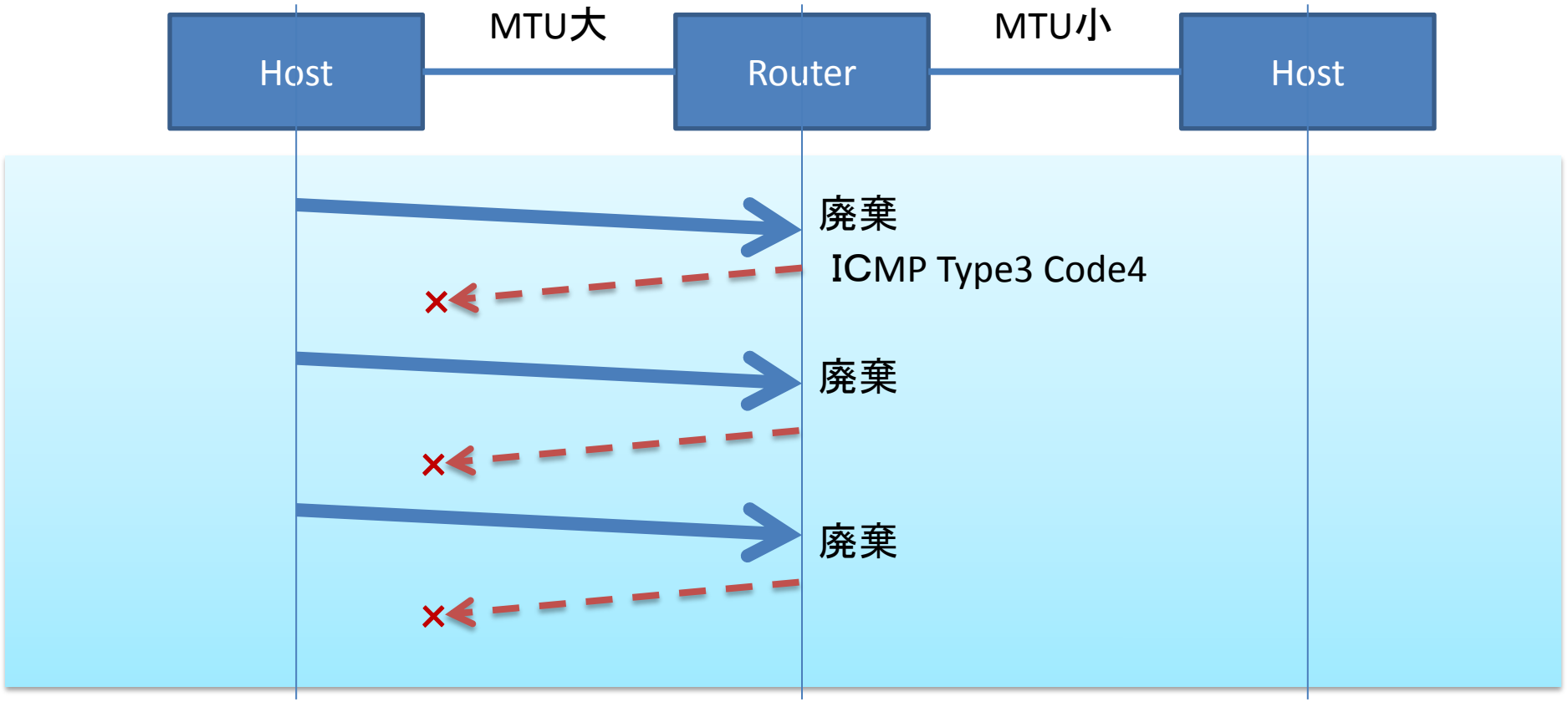
1. フラグメンテーションのおさらい

フラグメンテーション(IPv4)



- IPv6は、IPv4 DF=1と同じ挙動(DF=0相当は非サポート)
  - IPv6には、そもそもDFビット相当は存在しない

PMTU ブラックホール



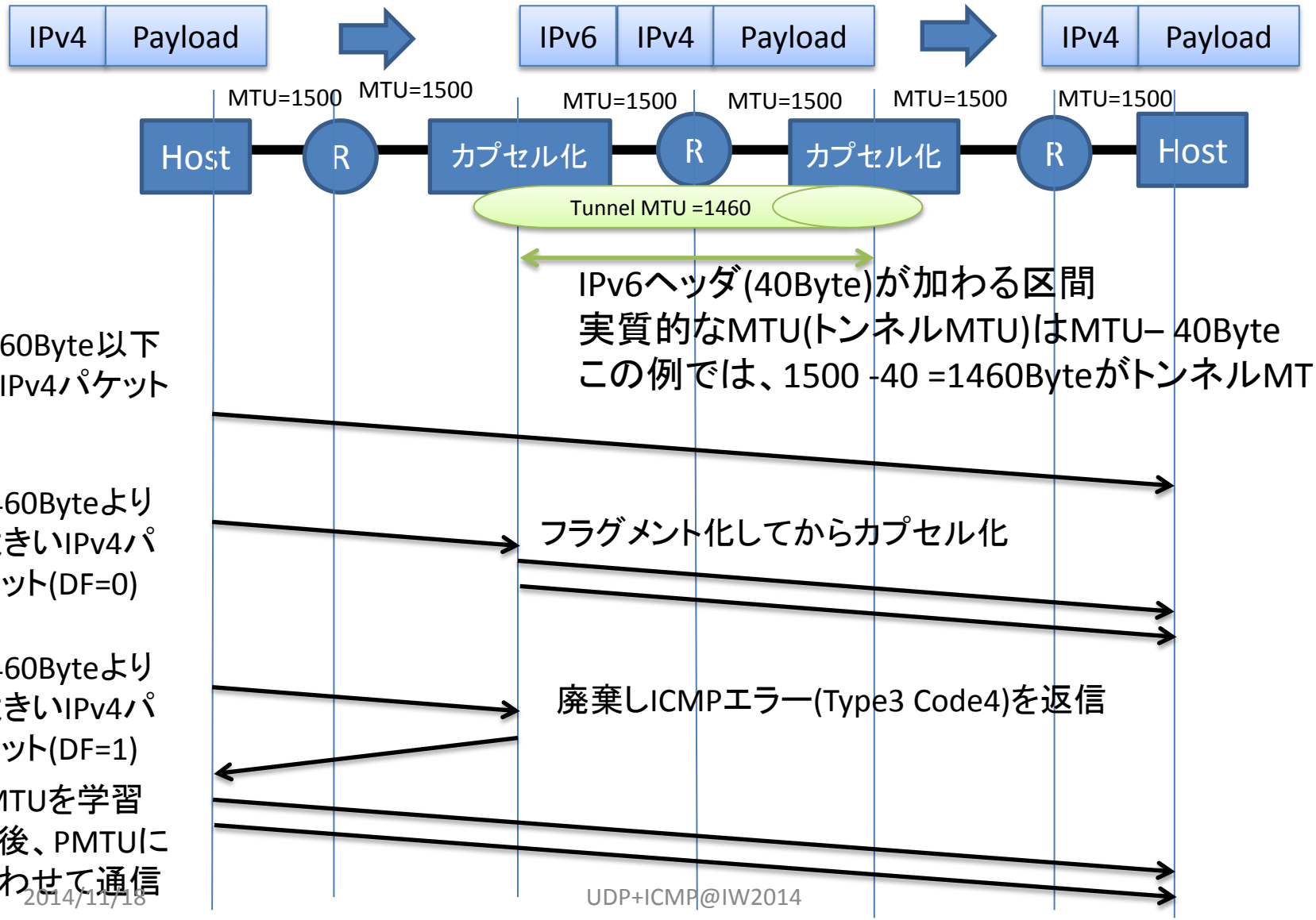
- ICMPエラーメッセージがフィルタ(廃棄)される
- Path MTU長が発信ホストに伝わらないので廃棄されるサイズで送信を繰り返す
- 永遠に通信できない

フラグメンテーション関連で過去に起きたこと

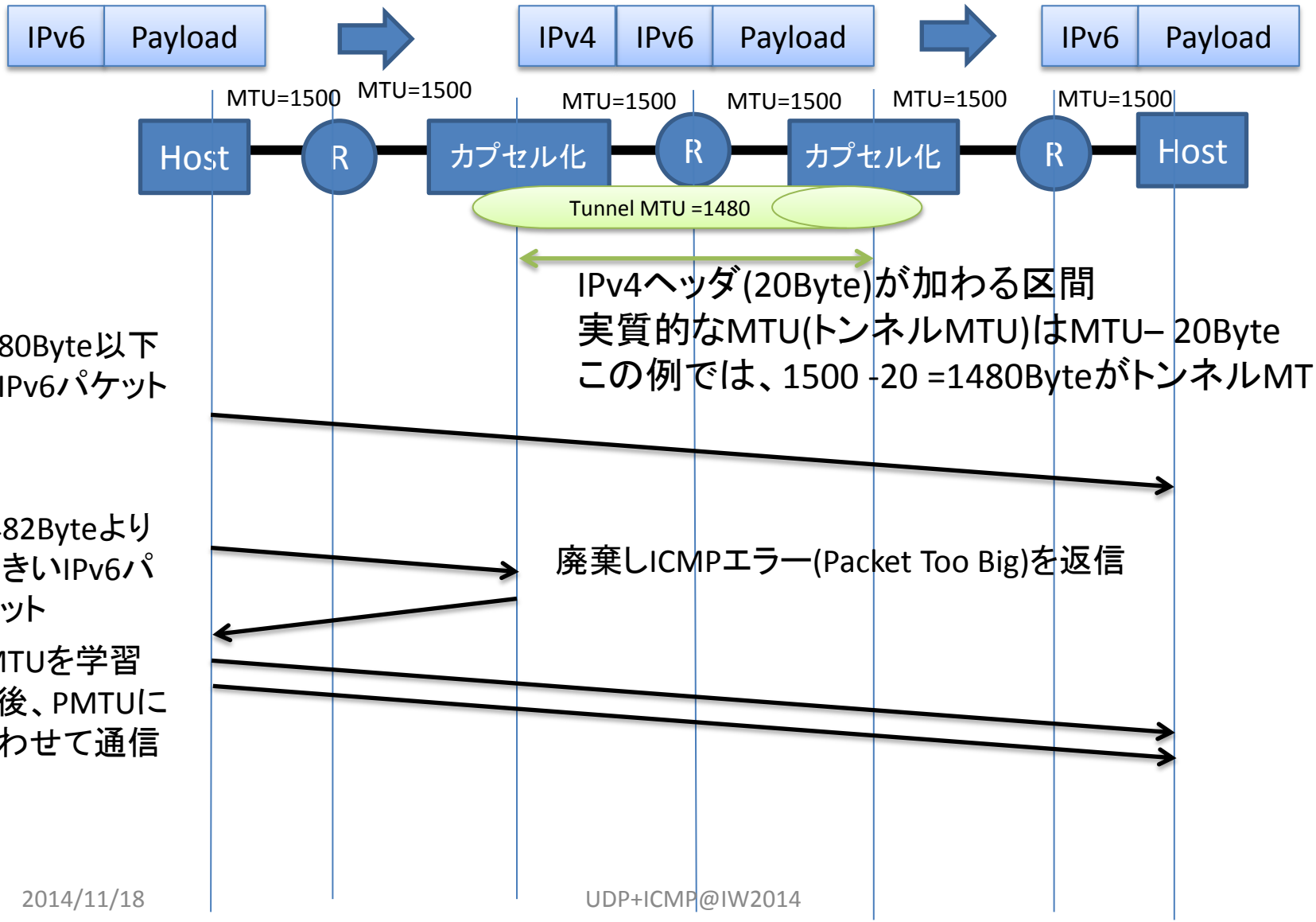
- 何故、PMTUDが追加されたか
  - FDDI
- TCPの挙動とUDPの挙動
  - NFS
- 過去、大きく2度話題になった
  - FDDI導入の際＋ADSL(PPPoE)導入の際
  - 何が起き、どう対処したか／しなかったか:TCP MSS
  - 問題にならなかった時代:PPP、すなわちdialup ip
- 2度あることは3度ある?
  - DNS EDNS0, DNSSEC:「ランチのおともにDNS」で扱われる?
  - IPv4-IPv6移行技術(カプセル化、IPv4-IPv6変換)
  - NVO3 (GRE, VxLAN, NVGRE等)

2. IPv4 - IPv6移行技術とフラグメンテーション

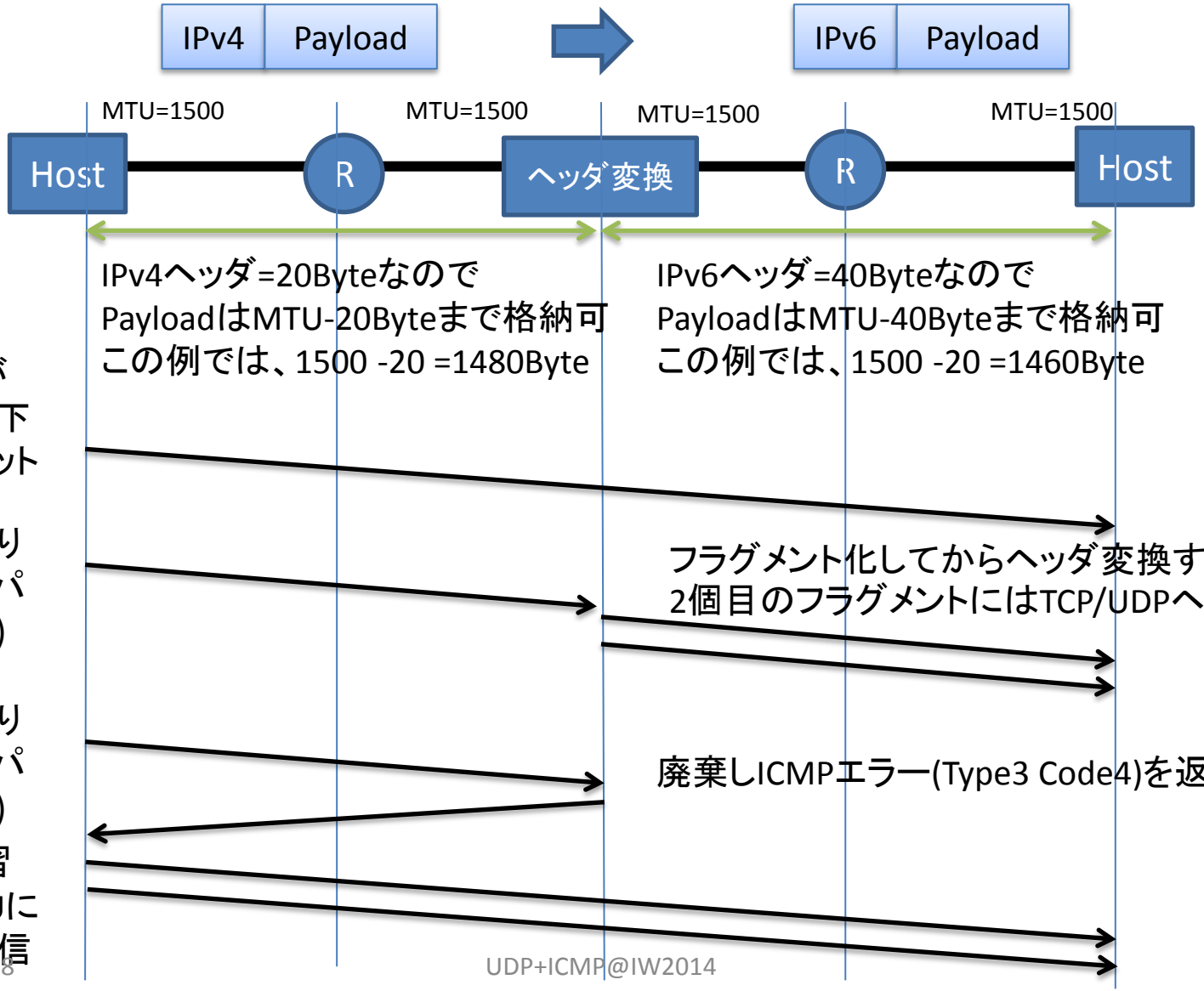
IPv4 over IPv6



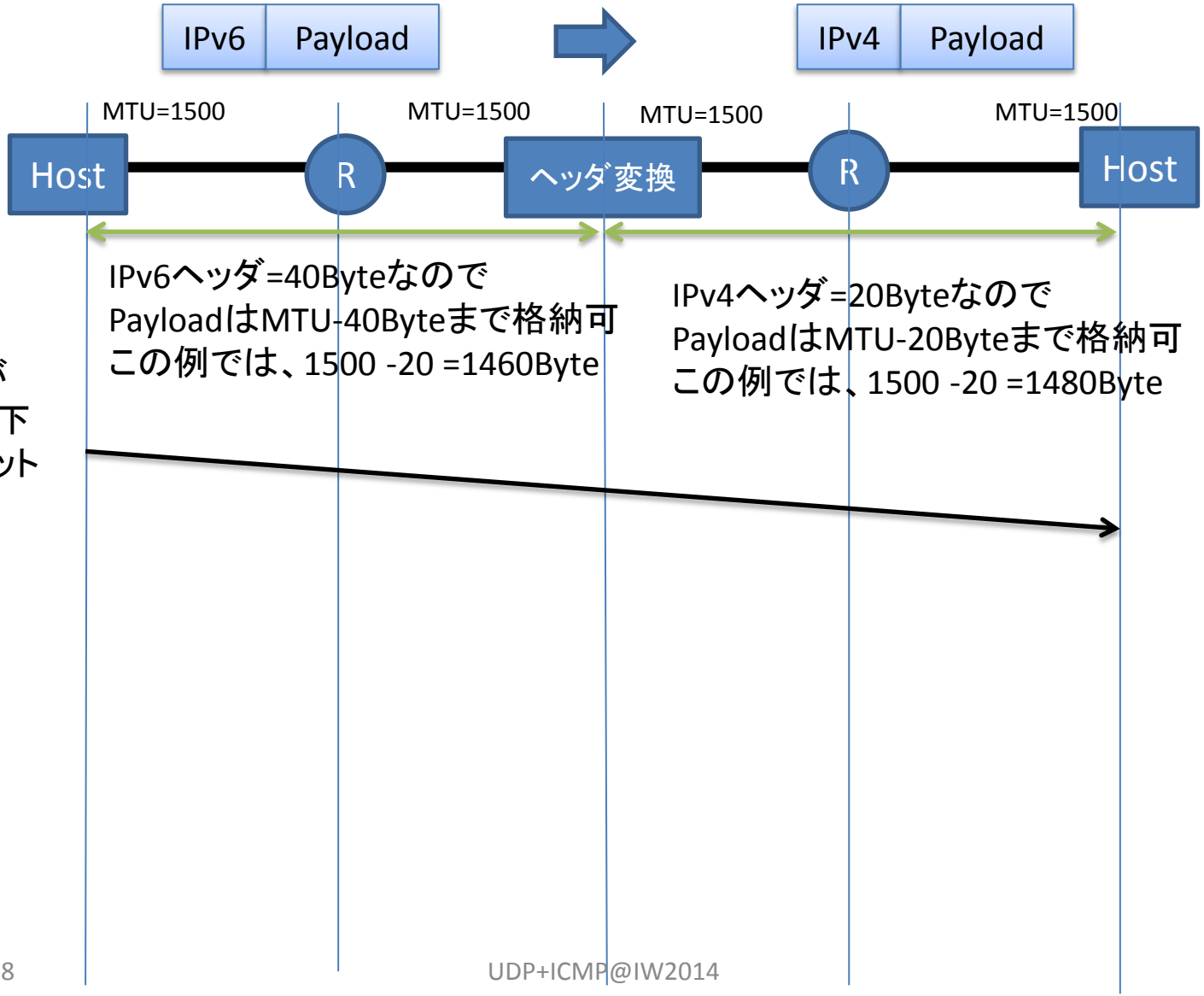
IPv6 over IPv4



IPv4-IPv6変換

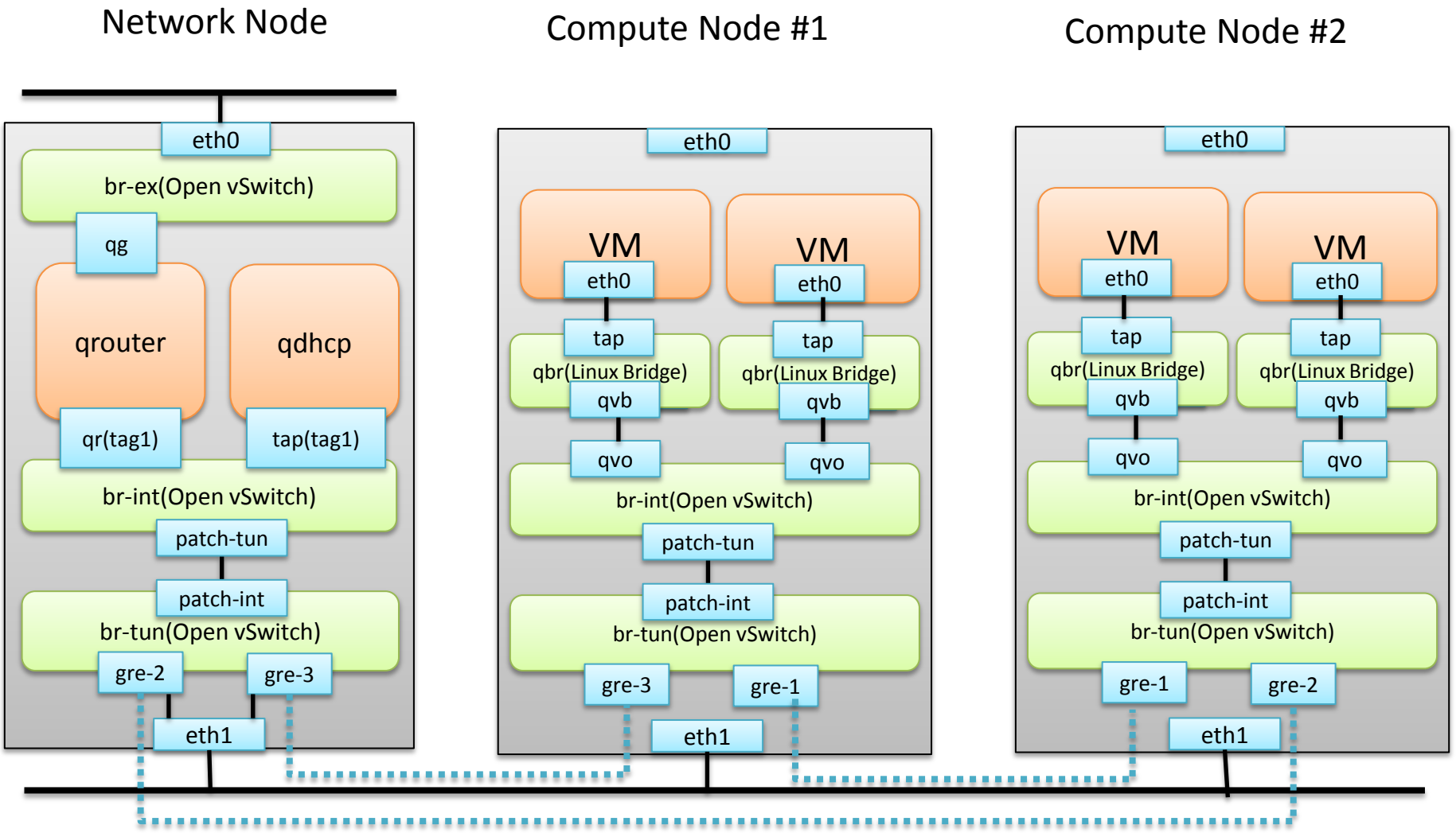

IPv4
Payload
IPv6
Payload
MTU=1500
MTU=1500
MTU=1500
MTU=1500
Host
R
ヘッダ変換
R
Host
IPv4ヘッダ=20Byteなので
PayloadはMTU-20Byteまで格納可
この例では、1500 -20 =1480Byte
IPv6ヘッダ=40Byteなので
PayloadはMTU-40Byteまで格納可
この例では、1500 -20 =1460Byte
Payload長が
1480Byte以下
のIPv4パケット
1480Byteより
大きいIPv4パ
ケット(DF=0)
フラグメント化してからヘッダ変換すれば良いが、
2個目のフラグメントにはTCP/UDPヘッダ無し
1480Byteより
大きいIPv4パ
ケット(DF=1)
廃棄しICMPエラー(Type3 Code4)を返信
PMTUを学習
以後、PMTUに
合わせて通信

IPv6-IPv4変換


IPv6
Payload
IPv4
Payload
MTU=1500
MTU=1500
MTU=1500
MTU=1500
Host
R
ヘッダ変換
R
Host
IPv6ヘッダ=40Byteなので
PayloadはMTU-40Byteまで格納可
この例では、1500 -20 =1460Byte
IPv4ヘッダ=20Byteなので
PayloadはMTU-20Byteまで格納可
この例では、1500 -20 =1480Byte
Payload長が
1460Byte以下
のIPv6パケット

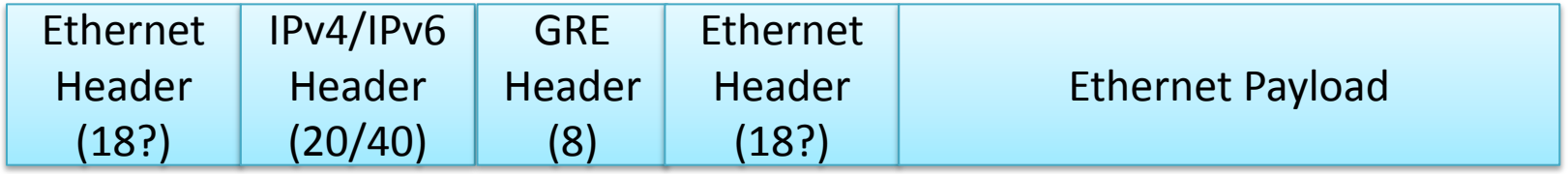
3. クラウドとフラグメンテーション

OpenStackの内部構造 (Neutron + Nova)

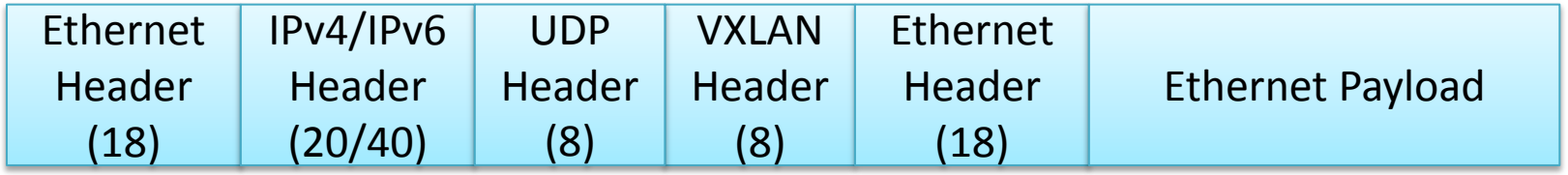


L2 over L3(IP)技術

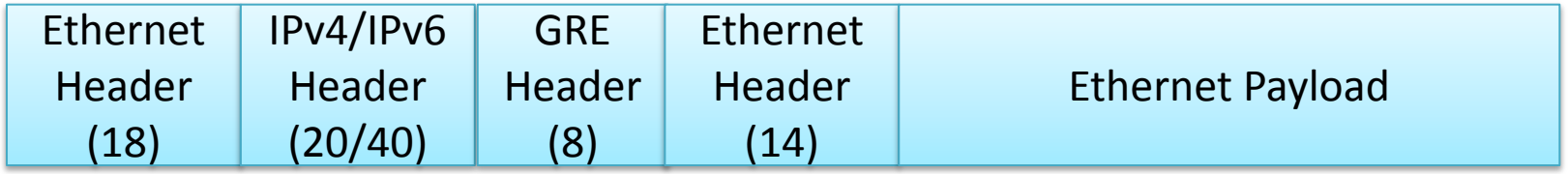
GRE(RFC2784): Protocol TypeがTransparent Ethernet Bridgeの場合



VxLAN(draft-mahalingam-dutt-dcops-vxlan-09.txt)

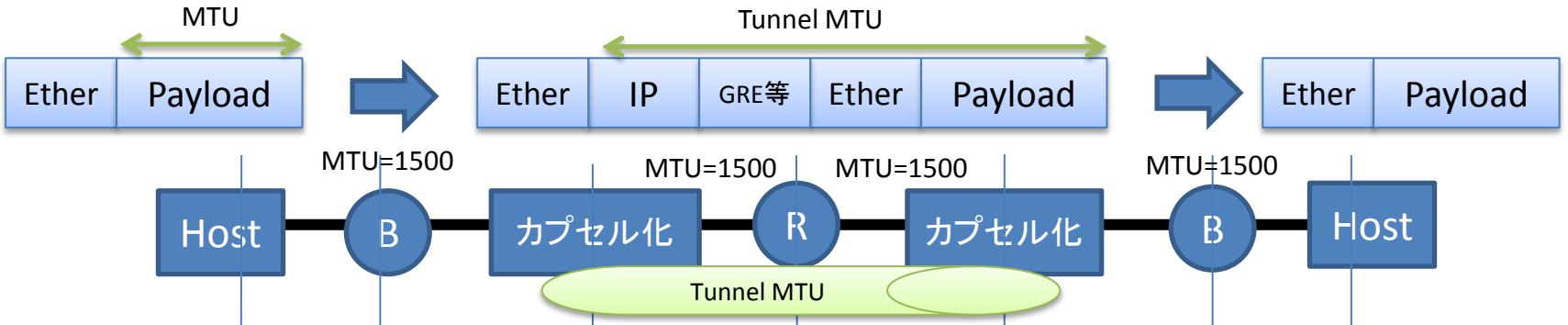


NVGRE(draft-sridharan-virtualization-nvgre-06.txt)



- いずれもEthernetフレームをIPでカプセル化

L2 over IP



IPヘッダ+GREヘッダまたは(VxLANヘッダ+UDPヘッダ)+Ethernetヘッダが加わる区間

カプセル化後IPパケット長がTunnel MTU以下の場合

カプセル化後IPパケット長がTunnel MTUより大きい場合

カプセル化した後フラグメント化

フラグメントを組みなおす必要あり

PayloadがIPパケットなら、IPパケットのフラグメント化で調整可能

廃棄しICMPエラー(Type3 Code4)を返信

draft-mahalingam-dutt-dcops-vxlan-09.txtの記載

VTEPs MUST NOT fragment VXLAN packets. Intermediate routers may fragment encapsulated VXLAN packets due to the larger frame size. The destination VTEP MAY silently discard such VXLAN fragments. To ensure end to end traffic delivery without fragmentation, it is RECOMMENDED that the MTUs (Maximum Transmission Units) across the physical network infrastructure be set to a value that accommodates the larger frame size due to the encapsulation. Other techniques like Path MTU discovery (see [RFC1191] and [RFC1981]) MAY be used to address this requirement as well.

- トンネルの始点でのVXLANパケットのフラグメント禁止、トンネル終点でフラグメント化されたVXLANパケットを受信したら廃棄
- フラグメント化されないようにMTUの値をセットすることを推奨
- Path MTU Discoveryなど使ってもよい

draft-sridharan-virtualization-nvgre-06.txtの記載

4.4. IP Fragmentation

RFC 2003 [11] Section 5.1 specifies mechanisms for handling fragmentation when encapsulating IP within IP. The subset of mechanisms NVGRE selects are intended to ensure that NVGRE encapsulated frames are not fragmented after encapsulation en-route to the destination NVGRE endpoint, and that traffic sources can leverage Path MTU discovery. A future version of this draft will clarify the details around setting the DF bit on the outer IP header as well as maintaining per destination NVGRE endpoint MTU soft state so that ICMP Datagram Too Big messages can be exploited. Fragmentation behavior when tunneling non-IP Ethernet frames in GRE will also be specified in a future version.

- フラグメントの扱いはRFC2003(IP Encapsulation within IP)に準拠
  - 注：RFC2003はRFC2002, mobileip(IP in IP)とセットのRFC
- 詳細は将来明らかにされる
- IPパケット以外のイーサネットフレームについては将来規定される

L2 over IPとマルチキャスト

- L2マルチキャスト、L2ブロードキャスト
- IPマルチキャスト
- IPマルチキャストとフラグメント化
  - MTU目いっぱいのEthernetフレームがマルチキャスト、ブロードキャストされることはあるのか？
  - IPマルチキャストでPath MTU Discoveryは動くのか？
    - マルチキャストグループ内で一番小さいMTUに合わせてパケット化する必要がある

4. IETFでの最近の議論

IETF会議に於ける議論の状況

- 85th IETF (Atlanta): 2012/11
  - v6ops: Why Operators Filter Fragments
- 86th IETF (Orlando): 2013/3
- 87th IETF (Berlin): 2013/8
  - 6man: IPv6 Fragment Header Deprecated
  - intarea: GRE MTU
- 88th IETF (Vancouver): 2013/11
  - IEPG88: Fragmentation and Extension Header Support in the IPv6 Internet
- 89th IETF (London): 2014/3
  - 6ops: Why Operators Filter Fragments
  - intarea: GRE MTU
- 90th, 91stは出席できなかったので、フォローしていません

関連Internet Draft

- Why Operators Filter Fragments and What It Implies
  - draft-taylor-v6ops-fragdrop-02
- IPv6 Fragment Header Deprecated
  - draft-bonica-6man-frag-deprecate-02
- A Fragmentation Strategy for Generic Routing Encapsulation (GRE)
  - draft-bonica-intarea-gre-mtu-04

IPv6フラグメントヘッダ廃止の議論

- 87th IETF Berlinの6man WGで提案
- 背景
  - ICMPv6 Packet Too Bigのフィルタリング
  - 拡張ヘッダのついたIPv6パケットのフィルタリング
- 上位レイヤでの対応を期待
  - PLPMTDU(RFC4821)
- 建設的な提案なのか？
  - 悲鳴なのでは？
- 現在、I-DはExpire

PLPMTUD

- Packetization Layer Path MTU Discovery
- 56th IETF San Francisco (2003/3)でBOF開催
- 2007/3にRFC4821として発行
- Packetization Layerで、Path MTUをprobeする
  - Loss reporting mechanisms
  - congestion control algorithms, rate limited
  - diagnostic tools
- 小さいMTUから少しずつ大きくしていく
- PMTUDが動く場合は容易に学習可能PMTUに合わせてパケット化
  - PMTUDが動くにこしたことはない

GRE MTU

- GRE(RFC2784)に於いてフラグメントに関する記載が不足していたために、実装がベンダ依存(ベンダにより動作が異なる可能性がある)になっている
- 現状の調査と整理
- RFC2784の改版を目的としない？
- 取り得る処理
  - ペイロードを廃棄(PMTUD/PLMTUD)
  - ペイロードをフラグメントしてカプセル化
  - カプセル化後にフラグメント

Why Operators Filter Fragments

- 仮説
  - 実装の問題と運用の問題の双方が原因
- 具体的な記載
  - Stateful inspection
    - 組み立てなおすことによる性能劣化
  - Stateless ACLs
    - 2個目以後のフラグメント
  - Performance
    - Forwarding planeでなくControl planeで処理
  - Other
    - バグなど
- 現在、活動が見えなくなっている模様

可能性のある対処法(現実的かどうかはともかく)

- アプリケーション層
  - NFSのような方法(確か：記憶が定かなら)
  - UDPを使用しているアプリをTCPを使うように変更する(DNS等)
- トランスポート層
  - TCP/MSS(解決策というより緊急避難?, IPsec時破綻)
  - UDPやGREの解は無い
  - (ICMPv4/ICMPv6は大丈夫か??：エラーパケットを詰める)
  - RFC4821: “Packetization Layer Path MTU Discovery”
- ネットワーク層
  - DFを0に書き換える(規約違反：解決策というより緊急避難?)
- 物理層／データリンク層
  - Ethernetのジャンボフレームを使う
  - FDDI, ATMを使う
  - 最低1500Byteのパケットが届くように網設計する

解決に向けて

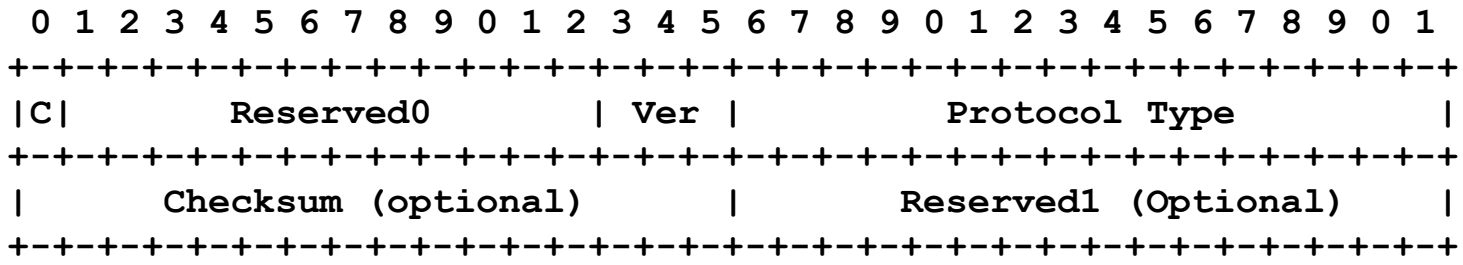
- 誰が捨てているのか、というか、本当に捨てているの？？
  - ネットワーク、データセンター、キャッシュ、サーバそのもの
  - どのくらいの影響があるのか?
  - TCP/MSSで救われているのはどれくらいか?
- なぜ捨てているのか
  - やむにやまれない理由があるのか
  - 実は、たいした理由は無いとか
- 将来どのような問題が予見されるか(未然に防げれば良い)
  - DNS AAAA(EDNS0): PMTUD動作はIPv6対応に含む
  - DNSSEC：PMTUD動作はセキュアであると言えるな条件
  - 移行技術(カプセル化、IPv4-IPv6変換)
  - クラウド／NVO3
- インターネット全体の問題であり、業界を挙げた問題解決が必要なのではないか？
  - PMTUDがきちんと動くことが目指すべき姿だと思います
  - 一方で、フラグメントの貧弱性を利用した攻撃も可能と指摘あり

まとめ

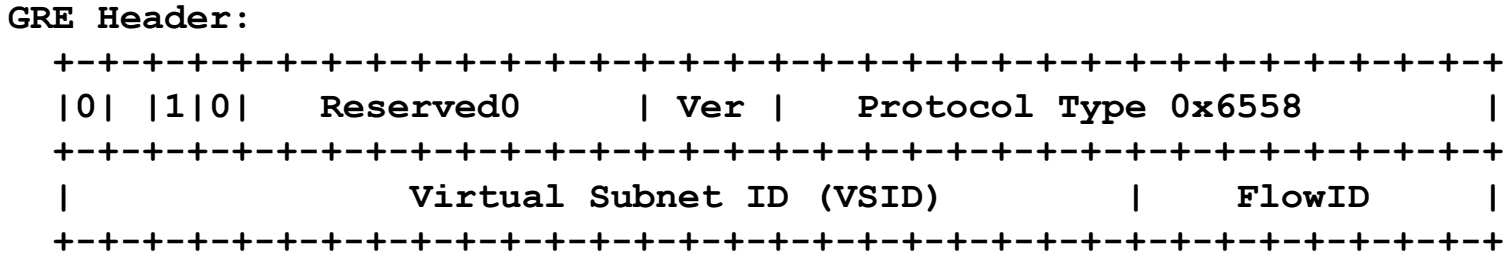
- TCP以外にも、以下のプロトコルがある
  - UDP
  - ICMP
- TCP, UDP, ICMP以外にも以下のプロトコルがある
  - L3 over L3 (IPv4 over IPv6, IPv6 over IP)
  - IPv4-IPv6変換(SIIT, NAT64)
  - L2 over L3 (GRE, VxLAN, NVGRE)
  - IPマルチキャスト
  - ARP (IPv4のみ)
- IPv6時代(含：移行)、クラウド時代でますます重要に

backup

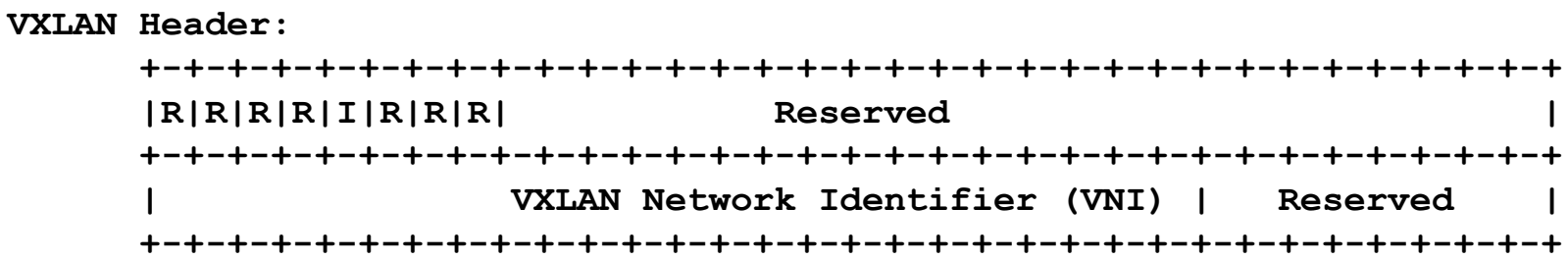
GRE Header(RFC2784)



NVGRE


GRE Header:
|0| |1|0| Reserved0 | Ver | Protocol Type 0x6558
Virtual Subnet ID (VSID) | FlowID

VxLAN


VXLAN Header:
|R|R|R|R|I|R|R|R|
Reserved
VXLAN Network Identifier (VNI)
Reserved