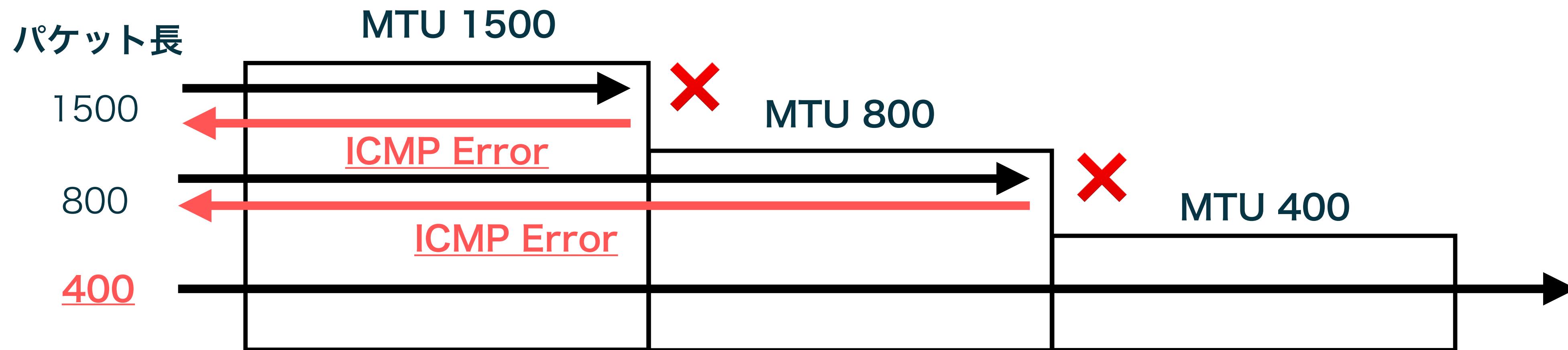


search mtuの調査 2025-12-18

2455026 平野碧生

経路内で断片化を必要としない最大のMTUの探索

DFを設定したパケットの送信時に断片化が必要となった際の応答パケットを利用



✗ ICMP Error

タイプ	コード	チェックサム
00		Next-Hop MTU
EchoRequestのコピー		

Type3 Code4のICMPパケット

- 未使用領域 → Next-Hop MTU
- 自作したtracerouteを流用

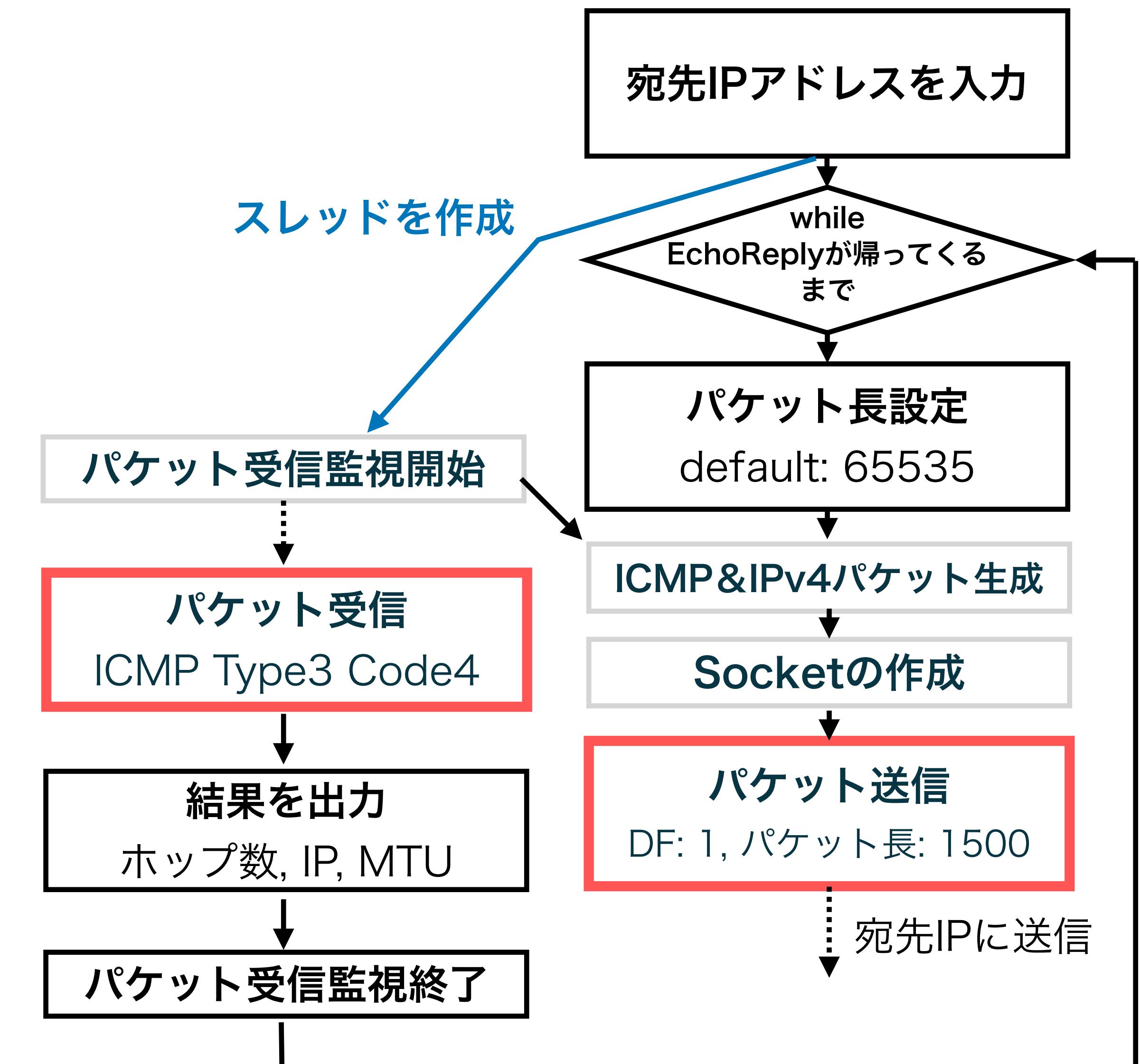
search mtuの実装

推定値	MTU	利用ネットワーク	参考
65535	65535	MTUの最大値	RFC791
	65535	ハイバーチャネル	RFC1044
32000	—	—	万一に備えて
17914	17914	16Mビット/秒 IBM トーカン・リング	—
8166	8166	IEEE802.4	RFC1042
	4464	IEEE802.5 (最高4Mビット/秒)	RFC1042
4352	4352	FDDI (Revised)	RFC1188
	2048	Wideband Network (広帯域ネット)	RFC907
2002	2002	IEEE802.5 (4Mビット/秒推奨)	RFC1042
	1536	Exp. Ethernet Nets	RFC895
1492	1500	Ethernet Networks (イーサネット)	RFC894
	1500	ポイント・ツー・ポイント (標準)	RFC1134
	1492	IEEE802.3	RFC1042
	1006	SLIP (シリアル回線インターネット・プロトコル)	RFC1055
1006	ARPANET		BBN1822
	576	X.25 Networks	RFC877
508	544	DEC IP Portal	—
	512	NetBIOS	RFC1088
	508	IEEE802/Source-Rt Bridge	RFC1042
	508	ARCNET	RFC1051
	296	ポイント・ツー・ポイント (低遅延時)	RFC1144
68	—	MTUの最小値	RFC791

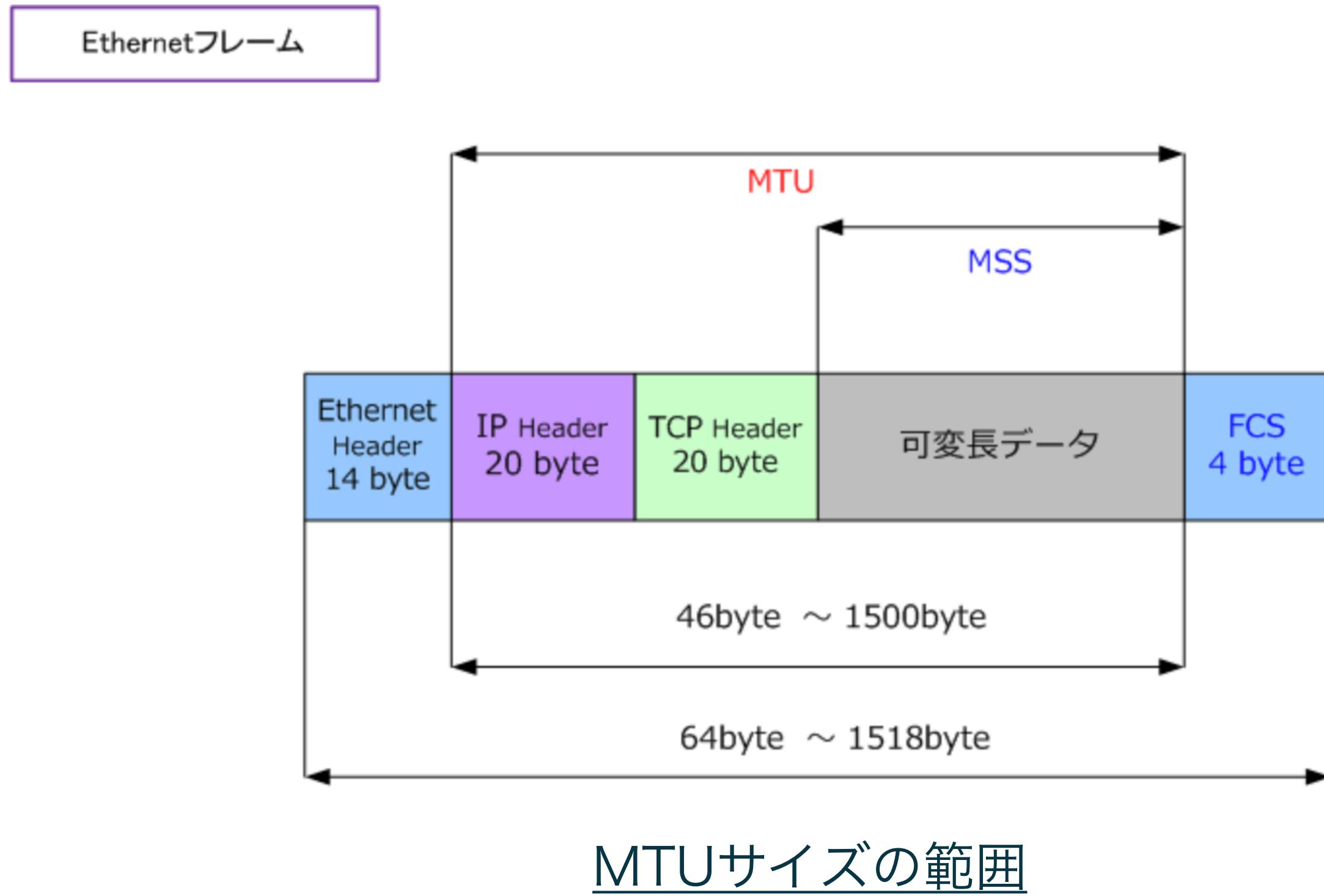
MTU一覧表

最初に送信するパケット

- DF: 1
- パケット長: 1500



指定mtuを元にパケットを生成



MTU = 1500の場合、

$$1500 - 20(\text{IPヘッダ}) - 20(\text{TCPヘッダ}) = 1460$$

中身を「a」で埋める

MTUと同じサイズのパケットの生成のため、61(a)でペイロードを埋め尽くす

実行結果

```
search mtu to 8.8.8.8, 64 hops max, first mtu is 1500
fragment needed! 7.515 ms 192.168.3.1
| max mtu updated! :1500 -> 1422
Echo Reply!! 48.156 ms 8.8.8.8
final mtu: 1500 -> 1422
```

実行結果

ひとまずうまく実装できた

一度経路上でフラグメントが必要な箇所があったことがわかる

1500から1422までMTUが減っていた

参考文献

- 日系XTECH 経路MTU探索:
<https://xtech.nikkei.com/it/article/COLUMN/20051202/225628>
- Wikipedia Maximum Transmission Unit:
https://ja.wikipedia.org/wiki/Maximum_Transmission_Unit
- RFC 792:
<https://datatracker.ietf.org/doc/html/rfc792>
- RFC 1191:
<https://datatracker.ietf.org/doc/html/rfc1191>
- ネットワークエンジニアとして Path MTU Discovery
<https://www.infraexpert.com/info/5.2adsl.htm>