

# 受信可能な最大パケットサイズ

2015/8/3

藤田将輝

## 1 はじめに

本資料では、本デバッグ支援環境において処理できる最大のパケットサイズの調査について記述している。受信ディスクリプタにパケットのサイズを表す部分があり、これの最大値が 16383 である。このため、最大で 16383 バイトを処理できると考え、実験を行った。実験を行った結果、FCS の 4 バイトを除いた、16379 バイトを処理できることがわかった。

## 2 パケットサイズの算出

NIC ドライバでパケットを受信した際、パケット受信処理の関数内で、パケットのサイズを算出する。パケットサイズは、受信ディスクリプタを参照することで算出する。受信ディスクリプタ内にはパケットのサイズを保持する部分があり、これを参照することで、パケットを算出している。受信ディスクリプタのパケットサイズを表す部分は、14 ビット割り当てられており、これによってパケットのサイズを表している。したがって、受信ディスクリプタが表せるパケットサイズの範囲は 0 ~ 16383 バイトであると考えられる。このため、16383 バイトのパケットを作成し、このパケットが正常に処理できるかどうかの実験を行った。また、NIC ドライバのソースコード内のパケットサイズを求める部分で、受信ディスクリプタが表すサイズから 4 を引いた値をパケットのサイズとして算出している。これは、EthernetFrame の FCS を処理の対象外としているためだと考えられる。したがって、実際に表せる最大のパケットサイズは 16379 バイトである。FCS についての詳細は調査できていない。

## 3 実験

### 3.1 実験環境

本実験における実験環境を表 1 に示す。

表 1 実験環境

項目名	環境
OS	Fedora14 x86_64(Mint 3.0.8)
CPU	Intel(R) Core(TM) Core i7-870 @ 2.93GHz
NIC ドライバ	RTL8169
ソースコード	r8169.c

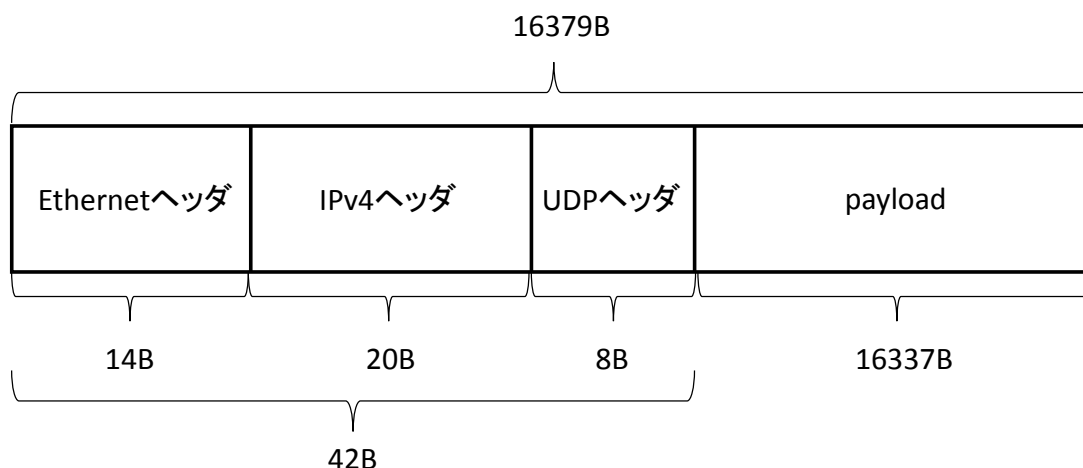


図1 パケットの構成

### 3.2 実験手順

本実験では、表現可能な最大のサイズのパケットを作成し、これを NIC ドライバで処理させ、正常に処理できるかを検証する。本実験の実験手順を以下で説明する。

- (1) パケットジェネレータにより、パケットを作成する。受信ディスクリプタで表現可能な最大のサイズのパケットを作成する。
- (2) パケットを共有メモリに配置し、IPI を送信する。
- (3) NIC ドライバがパケット受信処理を行う。デバッグ対象 OS 上で動作する、受信した UDP の payload を標準出力に出力するプログラムによって payload が表示されることを確認する。

### 3.3 作成したパケット

作成したパケットを図1に示し、説明する。作成したパケットは EthernetFrame であるため、Ethernet ヘッダ、IPv4 ヘッダ、および UDP ヘッダを含んでいる。それぞれのサイズは、12 バイト、20 バイト、および 8 バイトである。これらのヘッダのサイズは不変であるため、payload を任意のサイズに設定することで、パケットのサイズを変更する。今回は、受信ディスクリプタで表現できる最大のサイズのパケットを作成するため、payload のサイズを表現できる最大のサイズである 16383 バイトからヘッダの 42 バイトを引いた 16337 バイトとしている。また、payload に 16337 個の「0」を配置することで実現している。

### 3.4 実験結果

実験を行った結果、デバッグ対象 OS 上で動作するプログラムが標準出力に 16337 個の「0」を出力していたため、正常に処理できていると判断した。また、最大よりも大きなパケットを処理させた際は、

計算機の電源が落ちたため、正常に処理できていないと判断した。これらから、本デバッグ支援環境では1 パケットで最大 16379 バイトのパケットを処理できることを確認した。

## 4 おわりに

本資料では、本デバッグ環境で処理できるパケットの最大のサイズを検証した。この結果、最大で 16379 バイトのパケットを処理できることを確認した。今後は、連続でパケットを送信した際に、どの程度の速度やサイズであれば正常に処理できるかを測定する。