ダウンロード・データあります

パケットづくりではじめる ネットワーク入門

第16回 インターフェースの種別を知る

坂井 弘亮

本連載はネットワーク上を流れるパケットを直接扱うようなツールを自作しつつ，ネットワークの仕組みを勉強していきます．テーマは「自作」，「現物ベース」，「動く感動」の三つです．ネットワークにはイーサネットとIPを想定しています．

● 前回までの自作ルータ・プログラムで目をつぶっていたこと…IPヘッダ取得時のムダ処理

前回までは自作の簡易ルータにNAT機能を追加していきました．具体的にはNAPTや複数プロトコル(TCP/UDP/ICMP)の対応，ポート・フォワーディングなどの機能を追加しました．これにより，一般的なブロードバンド・ルータに実装されている機能がどのような動作をするのか，実装を通して知ることができました．

しかし前回までのプログラムには，実は問題点があります．ヘッダのアラインメントを考慮しなくて済むようにヘッダ情報を毎回コピーして処理しているため，無駄が多いという点です．これを防ぐためには，L2ネットワークの種別に応じたオフセット位置からパケットを格納し，IPヘッダの先頭がアラインメントにそろうようにする必要があります．

● 今回やること

今回はルータから離れて，L2ネットワークの種別を知る方法と，それを利用して受信バッファの先頭を調整する方法について説明します．

プログラミングの課題…IPヘッダを効率良く取得するのは難しい

● ホントは避けたい…memcpyによる非効率なIPヘッダ操作

リスト1は，連載の第3回で作成した簡易アナライザ(ネットワーク・パケットのアナライザ)のソースコード(ip-analyzer.c)中にあるTCPヘッダの解析部分です．今回問題にするのは，関数内の3行目にあるmemcpy()の呼び出しです．

リスト1では関数内でローカルに定義した構造体にパケットのヘッダ部分を一時的にコピーした上で，コピー先からヘッダ情報を取得しています．

これはヘッダのアラインメント問題を回避するための対策です．しかしmemcpy()によるメモリ・コピーが行われてしまうため無駄があり，処理速度の観点からも非常に不利だと言えます．

そして前回までに作成してきたツール類のソースコードには，実はこのようなコピー処理が多くあります．つまり全体的に，無駄な処理が多く残っているわけです．

● CPU/OSに依存するメモリ・アクセスの問題…4バイト・アラインメントずれ

そもそもアラインメント問題とは，どのようなものでしょうか？

通常，**リスト2**のように整数型の変数を定義した場合，その変数が配置されているメモリ上のアドレス(&val)は，変数のサイズ(sizeof(val))の倍数の値になっています．つまり**リスト2**の例では，if文の条件は真になります．

リスト1　TCPヘッダの解析処理(ip-analyzer.c)

```
static void proc_tcp(pktbuf_t pktbuf)
{
  struct tcphdr tcphdr;
  char *p = pktbuf_get_header(pktbuf);
                    /* オプション領域からヘッダ位置を取得 */
  memcpy(&tcphdr, p, sizeof(tcphdr));
  printf("TCP¥tsrc port: %d¥n",
                          ntohs(tcphdr.th_sport));
  printf("¥tdst port: %d¥n",
                          ntohs(tcphdr.th_dport));
  printf("¥tseq number: %u¥n",
                           ntohl(tcphdr.th_seq));
  printf("¥tack number: %u¥n",
                           ntohl(tcphdr.th_ack));
  printf("¥tflags: 0x%02x¥n", tcphdr.th_flags);
}
```

リスト2　整数型の変数を定義した場合

```
int val;
if ((((int)&val) % sizeof(val)) == 0)
  printf("alignment OK.¥n");
```

本連載のプログラムのソースコードは以下の筆者のホームページからダウンロードできます．ライセンスはKL-01というもので，組み込み機器などでも自由に利用できます．
http://kozos.jp/books/interface/ethernet2/