



chapter 1 インターネットの仕組み

- 1-1 インターネットとは** 2
 - ネットワークとインターネット 2
 - 初期のインターネット設計思想 3
 - パケットという考え方 4
- 1-2 OSI 7 層モデル** 6
 - 物理層とリンク層 6
 - ネットワーク層 7
 - トランスポート層 8
 - セッション層とプレゼンテーション層、アプリケーション層 8
- 1-3 TCP と UDP の概要** 9
 - コネクション型プロトコル～ TCP 10
 - コネクションレス型プロトコル～ UDP 11
- 1-4 IP アドレス** 11
 - ネットワーク部とホスト部 11
 - IP アドレスの種類 12
 - IP アドレスの一意性 14
 - プライベートアドレス 15
- 1-5 ドメイン名とホスト名** 16
 - ドメインネームシステム 16
- 1-6 Web とメールの仕組み** 18
 - Hyper Text Transfer Protocol 18
 - Web ブラウザが行う URL の解釈処理 19
 - メールの仕組み 20

chapter 2 TCP 通信の基礎

- 1-7 IPv4 と IPv6** 22
 - IPv6 への移行 23
- 1-8 Chapter 1 のまとめ** 24
- 2-1 TCP によるプログラミングの流れ** 26
 - ソケットとプログラミング 26
 - Linux におけるソケットの作成 27
- 2-2 TCP サーバ/クライアントの実装** 32
 - 単純な TCP サーバの実装 34
 - 単純な TCP クライアントの実装 36
- 2-3 ソケットプログラミングのエラー処理** 39
 - errno と perror 39
 - ソケット作成の失敗と perror 利用例 40
 - perror とスレッド処理 41
 - printf () と perror () の実行順 42
 - bind () の意味 43
 - listen () の意味 46
 - 無効になったソケットに対するデータ送信 47
 - 文字列でのエラー内容取得 49
- 2-4 名前解決の実装** 50
 - 名前解決のサンプルプログラム 51
 - エラー解析関数 52
 - IPv6 と IPv4 両方に対応する 53
 - getaddrinfo () を bind () で使う 54
- 2-5 単純なファイル転送プログラム** 56
 - ファイル送信側サンプルプログラム 57
 - ファイル受信側サンプルプログラム 59

2-6 単純な HTTP クライアント / サーバ 61
 HTTP クライアントの実装 62
 HTTP サーバの実装 65

2-7 Chapter2 のまとめ 67

chapter 3 UDP 通信の基礎

3-1 UDP の特徴とプログラミング 70

3-2 UDP のプログラミング 71
 単純な UDP 受信プログラム 72
 単純な UDP 送信プログラム 75
 getaddrinfo () を利用した UDP プログラム 78
 IPv6 が未設定の環境を想定する 81

3-3 ブロードキャストプログラミング 83
 ブロードキャスト送信プログラム 83

3-4 マルチキャストプログラミング 86
 マルチキャスト送信プログラム 86
 マルチキャスト受信プログラム 87

3-5 UDP ソケットの「名前」と bind () 90
 UDP での「返信」 94

3-6 ソケットバッファ 97

3-7 Chapter3 のまとめ 100

chapter 4 ネットワーク情報の取得と設定

4-1 ネットワークインターフェース 102
 インターフェース名とインターフェース番号 102

インターフェース情報一覧の取得 104
 インターフェース一覧のみの取得 108
 インターフェース名の変更 111
 ハードウェアアドレスの取得 112
 IP アドレスの設定 113
 ネットマスクの設定 118
 up/down 状態 119
 ハードウェアアドレスの設定 123
 MTU の取得 / 設定 124

4-2 /proc インターフェース 127
 経路テーブル 127
 ネットワークデバイス統計情報 128
 IPv6 アドレス 128
 ARP 128
 開いているソケット情報 129

4-3 Chapter 4 のまとめ 130

chapter 5 Netlink による情報取得と設定

5-1 Netlink の概要 132
 Netlink を使ったプログラムの流れ 132
 Netlink ソケットの作成 133

5-2 Netlink メッセージの送受信 134
 Netlink ユニキャストと Netlink マルチキャスト 134
 sockaddr_nl 構造体 136
 Netlink メッセージフォーマット 137
 NETLINK_ROUTE プロトコル 139

5-3 インターフェース情報の取得 142
 インターフェースを down 状態にする 148
 MTU の設定 151

- 5-4 IP アドレスの取得 153
- 5-5 IP アドレス変更通知の受け取り 157
 - マルチキャストに必要な特権 160
- 5-6 IP アドレスの設定 161
 - IP アドレスの削除 163
- 5-7 Chapter 5 のまとめ 163

chapter 6 基本的な通信関連関数

- 6-1 ノンブロッキングモード 166
 - ノンブロッキングでの UDP 受信 166
- 6-2 エンディアンとネットワークバイトオーダー 168
 - バイトオーダー 170
 - エンディアン同士の相互変換 172
 - 数値や構造体を送信するときの注意点 173
 - 相互接続の注意点 176
- 6-3 ソケットの設定を取得する 176
 - ソケットの名前を取得する 179
 - 接続相手の情報を取得する 180
- 6-4 シグナルハンドリング 181
- 6-5 知っておくと便利なマクロ 183
 - IPv4 アドレス用マクロ 183
 - IPv6 アドレス用マクロ 184
 - IP アドレス用文字列のサイズ 185
- 6-6 Chapter 6 のまとめ 186

chapter 7 プロセス間通信

- 7-1 プロセス 188
 - プロセスを生成する 189
- 7-2 スレッド 192
- 7-3 プロセス間通信とは 193
- 7-4 シグナル 194
 - kill () システムコール 195
 - シグナルハンドラ 196
- 7-5 UNIX ソケット 197
 - SOCK_DGRAM で UNIX ソケットを使う 200
- 7-6 ソケットペア 202
- 7-7 パイプ 204
 - コマンドラインの パイプ 206
- 7-8 System V IPC 208
 - IPC 資源と IPC キー 208
 - 共有メモリ 210
 - セマフォ 215
 - メッセージキュー 222
 - POSIX IPC 関数 227
 - pthread mutex をプロセス間で利用する 227
- 7-9 Chapter 7 のまとめ 230

chapter 8 複数のソケットを扱う

- 8-1 epoll とは 232

epoll_create () システムコール 232
 epoll_ctl () システムコール 233
 epoll_wait システムコール 235

8-2 単純な epoll プログラム 236
 データの受信プログラム 236
 データの送信プログラム 239

8-3 epoll_wait () の活用 240
 送信可能かどうかのチェック 241
 タイムアウトのチェック 245
 connect () 処理の終了 248

8-4 epoll_wait() の解除 250
 シグナルを送信する 251
 ソケットペアを使う 252

8-5 select () と poll () 255
 select () システムコール 255
 poll () システムコール 260
 送信可能かどうかのチェック 263
 タイムアウトのチェック 264

8-6 Chapter 8 のまとめ 267

chapter 9 通信プログラムのデバッグ

9-1 パケットキャプチャによるデバッグ 270
 機器内で直接パケットを見る方法 271
 通信路でパケットを見る方法 271
 パケットキャプチャと暗号化 272

9-2 インターネットとカプセル化 272
 プロミスキャスモード 274

9-3 Wireshark 275
 Wireshark の基本動作 275
 キャプチャフィルタ 278
 Follow TCP Stream 279
 フィルタ機能 280

9-4 各種ヘッダ 282
 IPv4 ヘッダ 282
 IPv6 ヘッダ 283
 TCP ヘッダ 284
 UDP ヘッダ 286
 ICMP ヘッダ 287

9-5 tcpdump 287
 主なオプション 288
 キャプチャパケットをファイルに保存 288
 条件式を指定してフィルタリング 289

9-6 デバッグ TIPS 290
 なぜ接続できないのか? 290
 TCP 接続を切断したのはどちら? 292
 TCP でデータがこない? 294
 パケットが出てる? 来てる? 296

9-7 Chapter 8 のまとめ 297

chapter 10 TCP サーバプログラミング

10-1 複数セッションの並列処理 300
 fork () を利用する 300
 スレッドを利用する 302
 epoll を利用する 306

10-2 TCP と TIME_WAIT 311
 TIME_WAIT 状態 311

ポート番号をすぐに再利用する 313

10-3 shutdown () の活用 314

10-4 ルートディレクトリを仮想的に変更 318
 chroot jail を抜ける 319

10-5 デーモン化 321

10-6 syslog への対応 322
 syslog () の利用プログラム 325

10-7 Chapter 10 のまとめ 326

chapter 11 デュアルスタックプログラミング

11-1 sockaddr_in6 構造体 328

11-2 sockaddr_storage 構造体 330

11-3 IPv4 射影アドレス 332

11-4 IPv4 アドレス枯渇と IPv6 335
 インターネットアーキテクチャの変化 335
 Large Scale NAT 337
 日本国内における影響 338

11-5 Chapter11 のまとめ 338

chapter 12 生パケットプログラミング

12-1 RAW ソケット 340

12-2 AF_PACKET 343

SOCK_DGRAM と SOCK_RAW の違い 344

sockaddr_ll 345

AF_PACKET を利用した ICMP ECHO の送受信 345

プロミスキャスモード 349

bind () 351

12-3 pcap を使ったキャプチャ 352
 インストールとコンパイル 353
 パケットキャプチャの流れ 353
 pcap を使ったキャプチャ 355
 フィルタを利用する 357
 ファイルへの書き出し 359
 ファイルからの読み込み 360

12-4 Chapter 12 のまとめ 362

chapter 13 マルチキャストプログラミング

13-1 マルチキャストアドレス 364
 動的なマルチキャストアドレス割り当て 365
 TTL とスコープ 365
 IPv6 でのマルチキャストアドレス 367
 マルチキャスト TTL の設定 (IPv4) 369

13-2 Source Specific Multicast 371
 SSM を利用する 372
 SSM への送信 373

13-3 出カインターフェースの指定 378
 出カインターフェースを指定する (IPv4) 378
 出カインターフェースを指定する (IPv6) 379

13-4 ASM におけるマルチキャストソースフィルタ 380

13-5 マルチキャストフィルタ情報の取得 383

| | | |
|-------------|------------------------------------|------------|
| 13-6 | IP_MULTICAST_LOOP | 385 |
| | JOIN できる最大数 | 386 |
| 13-7 | マルチキャスト関連の旧 API | 386 |
| | IP_ADD_MEMBERSHIP の利用 (IPv4) | 387 |
| 13-8 | Chapter 13 のまとめ | 389 |

Appendix

| | |
|----------------------------------|------------|
| IPv4 しか扱えない関数 | 392 |
| gethostbyname () | 392 |
| gethostbyaddr () | 394 |
| herror (), hstrerror () | 395 |
| inet_ntoa (), inet_addr () | 396 |
| IPv4 専用関数の落とし穴 | 397 |

コラム

| | |
|-------------------------------|-----|
| システムコールとは | 32 |
| サンプルプログラムの実行について | 38 |
| man と章番号 | 50 |
| 非同期シグナルセーフな関数 | 68 |
| サブネットブロードキャストを利用した DoS | 85 |
| 困ったときの net-tools ソースコード | 130 |
| ビッグエンディアンとリトルエンディアンの語源 | 171 |
| SIGHUP | 197 |
| IPC_PRIVATE ? | 230 |
| select () と MSG_OOB | 267 |
| あやしい UNIX 用語 | 274 |
| Wireshark と ethereal | 275 |
| Bugzilla | 298 |
| プリプロセッサで処理される組み込みマクロ | 326 |