

# TCPとUDPの違いについて

03-190512 宮田寛生

---

## はじめに

今回のI2実験では、I1とは違い、TCP/IPの知識を前提とした、ソケットプログラミングを行った。僕はこれまでTCPしか知らず、UDPというプロトコルを知らなかった。そこで、UDPという今回始めて知った知識について深堀していこうと思う。

## そもそもTCP・UDPとは？

TCPとUDPはOSIのトランスポートプロトコルであり、ポート番号によってアプリケーションを識別したり、チェックサムという仕組みでデータが壊れていないか確認したり、シーケンス番号によってパケットの順序を揃えたり（TCPのみ）する役割を持つ。

そもそもこれはなぜ必要かという、IPの送信はBestEffortで順序がないので、バラバラに送信したデータを正しく復元する必要があるからある。

## TCPとは？

TCPは、高信頼性(3 way handshake、ackによる受信確認や再送処理、等)や通信効率の最適化機能(Windowによるフロー制御や輻輳制御)を提供している。これは、TCPがデータを正しく届けるために作られたからである。

TCPについての詳しいことについては割愛するが、一般的に、欠けて困るデータはTCPで送る。

TCPを使うアプリケーション層のプロトコルとして有名なのは、

- POP3(Well known port: 110)
- SMTP(Well known port: 25)
- HTTP(Well known port: 80)

が挙げられる

## UDPとは？

UDPのポート番号の考え方についてはTCPと同様である。しかし、TCPのように3way handshake、確認応答・順序制御・再送制御・ウィンドウ制御・フロー制御などの機能はなく、ほとんど何もしないプロトコルである。UDPは、TCPと比べて信頼性が高くはないが、速さやリアルタイム性を求める通信に使用されるプロトコルである。

UDPは、コネクションレス型のプロトコルであることから、TCPに比べると信頼性がないものの高速に転送を行うことができる。また、UDPヘッダサイズは8byteと少ないので、その分アプリケーションのデータを多く送受信することができる。ただし、パケットが到達する保証がないことで、パケットロスなどの例外処理に関してはアプリケーション側での実装が必要となる技術的難易度の高さは挙げられる。

なので、UDPの使用用途は

- 音声や映像などのリアルタイム性のあるデータを転送する場合 リアルタイム性が求められるデータ通信では信頼性よりも高速なデータ転送の方が求められるので、音声通信ではUDPの方が品質が高まる。
- 複数の相手に同じデータを同時に転送する場合 TCP通信では、通信相手と1対1でコネクションを確立する必要があるため、1つのパケットを複数の宛先に送るようなマルチキャスト通信やブロードキャスト通信では、TCPを使用することができません。このように1対Nの通信を行うためにはコネクション確立の必要がないUDPが適している。
- 信頼性がなく、少量のデータ転送をしたい場合 お互いに一度したデータを送信しないような通信の場合、3way handshakeで3回のやりとりをしていては効率が悪いので、そのような少量のデータ転送で完了する通信にはUDPが適している。

などが挙げられる。

TCPを使うアプリケーション層のプロトコルとして有名なのは、

- DNS(Well known port: 53)

が挙げられる。

考察を終えて

I2実験ではTCPで実装したが、I3ではUDPに変更しようと思う。