ネットワーキング コンセプト

Developer

目次

図、表

概要

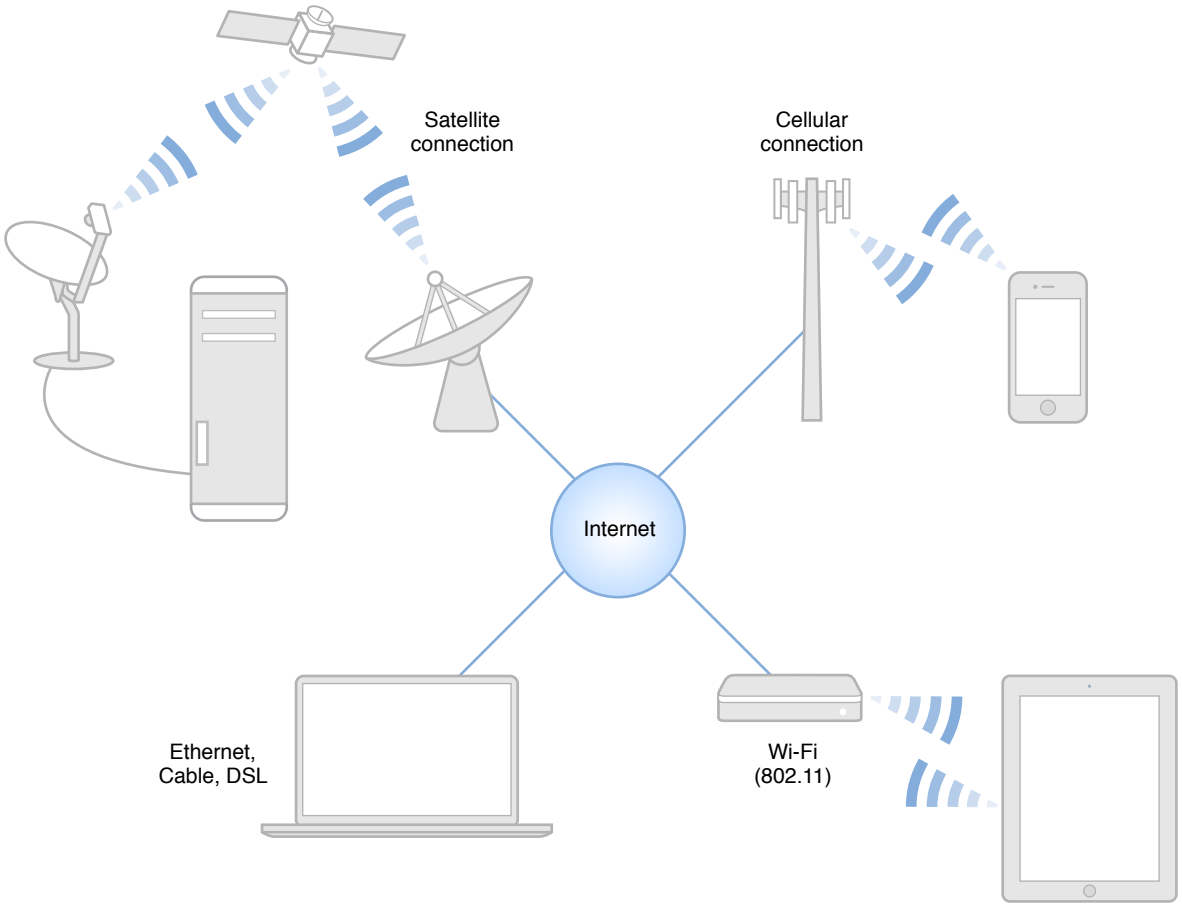
Important: この章には未確定の事項が一部記載されています。技術的に間違いないことは確認しておりますが、最終的なものではありません。この段階で公開したのは、ここで説明する技術やプログラミングインターフェイスを採用する一助にしていただくためです。内容は変わる可能性があるので、ソフトウェアを実装する際には、正式な資料が出た段階で確認してください。資料の更新については、AppleデベロッパWebサイトでご案内します。該当する参照ライブラリのページで、「Documents」欄に資料名を入力してください。

OS XやiOSのネットワーク機能の大部分はTCP/IPという通信プロトコルを基盤にしているので、実際にコードを記述するに当たっては、TCP/IPのネットワークモデルの基礎事項を理解しておく必要があります。プロトコルについて知らなくても高レベルのネットワーク処理は可能かもしれませんが、低レベルの動作の仕組みを知っていれば、問題が起こったときの原因や対処法をよく理解できるでしょう。

初めに

インターネットは、相互に接続されたコンピュータその他の機器から成る巨大なネットワークで、これを使って世界中と通信できます。ウェブサイトに接続したとき、そのコンテンツを供給するサーバは、同じ室内にあるかもしれませんし、海を越えた遠隔地にあるかもしれません。コンテンツ要求は、Wi-Fi（Wireless Fidelity）ルーター、ケーブルモデム、トランクライン、衛星接続、あるいは伝書

鳩（RFC 1149）を経由してやり取りされます。大づかみに言えば（性能特性を考えなければ）、ネットワーク接続媒体はいずれも同等です。しかし細かく見ていけば、機器相互の通信方法は、利用する物理ネットワークの種類に応じてさまざまです。



低レベルでは、ウェブサイトにアクセスしようとすると、コンピュータはその要求を「パケット」という断片に分割し、「ルーター」という特殊な機器に各パケットを送信します。ルーターはこれを別のルーターに転送し、これを繰り返して、最終的に要求先のサーバに到達するのです。応答も同様にして返されてきます。この経路の各段階で、関与するハードウェアによっては、パケットがさらに分割されたり、他の種類のパケットに包まれたりします。

この資料では、目に見えないところでインターネットがどのように動いているか、プログラム開発者向けに、やや大づかみに解説します。読み終えた後、さまざまなインターネット技術についてさらに詳しく知りたくなった方に向けて、さまざまな書籍が出版されています。さらに、低レベルのインターネット技術の多くは、プロトコルを詳細に述べたRFC（Request for Comment）に説明されています。

関連項目

良質のネットワーク処理コードを記述する方法を説明した、『*Networking Overview*』もお読みください。

ネットワーク技術

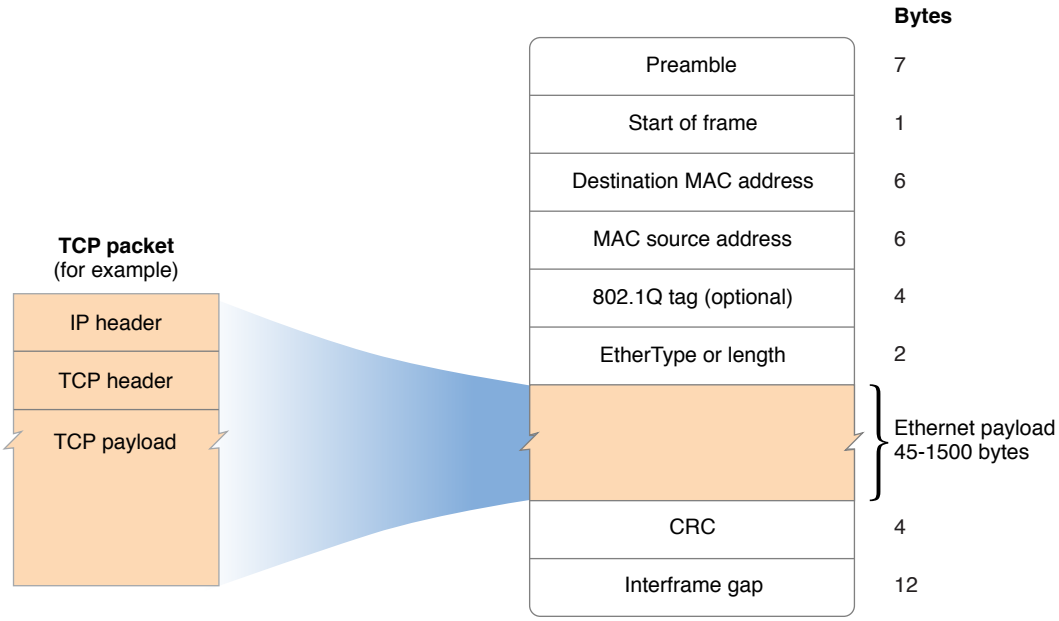
ネットワーク技術において**ホスト**とは、ネットワークに接続され、ネットワーク通信の終点としての機能を担う機器のことです。デスクトップコンピュータ、サーバ、iOSデバイス、サーバ上の仮想機械、さらにはVoIP電話もホストになりえます。この呼び名は、当該機器上で動作するアプリケーションやデーモンの「進行役を果たす」ことに由来します。

基盤機器も同様に、ネットワーク機能を提供する機器を指します。両者の違いは、基盤機器がホストから渡された情報を受け渡しするのに対し、ホストは主に自分自身が主体となって情報を送受信することにあります。TCP/IPネットワークの視点からは完全に透過的な、Ethernetハブやスイッチ（後述）などの基盤機器もあります。

ホストがネットワーク経由でデータを送信する際には、データを**パケット**という断片に分割します。その長さは可変ですが、最大長は物理ネットワーク接続（後述）によって決まります。

パケットは一般に、3つの部分から成ります。**ヘッダ**にはパケットの送り先その他の情報、**ペイロード**には実際に送信するべきデータ、**トレーラ**にはパケットを正しく受け取れたかどうか確認するためのチェックサムが入っています。パケットの種類によっては、チェックサム情報がヘッダに入っている代わりに、トレイラがないこともあります。例としてEthernetパケットの構成を示します。

図 1-1　Ethernetパケットの構成

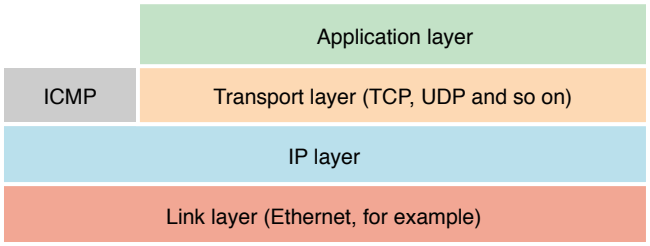


パケットを受け取ったホストはこれを組み立て直して、プロトコルの種類に応じ、バイトストリームまたはメッセージ列としてプログラム本体に渡します。受け取ったプログラムは、データを送って応答することができます。この場合もデータをパケットに分割して送信することになります。

パケットに別のパケット（一般に異なる型）を収容して送る場合、これを**カプセル化**と言います。たとえば図 1-1のEthernetパケットには、ペイロードしてTCP/IPパケットが含まれています。この場合、TCP/IPパケットはEthernetパケットにカプセル化されている、と言います。

ネットワークの階層

TCP/IPネットワークモデルは、基本的な4つの階層から成ります。リンク層、IP層（「Internet Protocol」の頭字語）、トランスポート層、アプリケーション層の4つです。以下、それぞれの階層について説明します。



リンク層

最下層に当たるのが**リンク層**（または**物理層**）です。ネットワークスタックのこの階層には、通信に用いる実際のハードウェアと、物理的に接続された隣接ホストが関与します。同じ物理ネットワークに属する、あるホストから別のホストに向けて、生パケットを伝送する働きがあります。

リンク層の通信路としてエンドユーザがよく見かけるのは、Wi-Fi、移動体通信網、Ethernet、Bluetooth、FireWireの5つでしょう。しかし伝送経路上では、さまざまな種類のリンクをたどって目的地に到達することになります。たとえば、DOCSIS（Dta Over Cable Service Interface Specifications）に基づくケーブルモデムや、VHDSL（超高速デジタル加入者線）をベースとするDSLモデムを使って、データを送信したとします。データはその後、（トランクラインの）SONET/SDH（Synchronous Optical NETwork / Synchronous Digital Hierarchy）リンク、IPoS（IP over Satellite）やIPoDVB（IP over Digital Video Broadcast）による衛星接続などを経由して、目的地に到達します。

ネットワークインターフェイスとは、リンク層の相互接続機能を提供するハードウェアのことです。ホストにはいくつでもネットワークインターフェイスを組み込むことができます。たとえば、複数の（有線）Ethernet、Wi-Fi、Bluetoothの各インターフェイスから、移動体通信モデムまで備えているホストがあります（MacにUSB接続、iPhoneやiPadに組み込みなど）。さらに、ソフトウェアで実装された（仮想）ネットワークインターフェイス、すなわちVPNソフトウェアや仮想化環境（OS X 10.4以前のClassic環境など）もあります。

各ネットワークインターフェイスには、一般に、結線を施して接続します。この接続を**リンク**と言います。結線と言っても物理的に接続されているとは限りませんが、オペレーティングシステムやソフトウェアからは、リンクは物理的な結線と同等に見えます。たとえば、Ethernetハブやスイッチで接続された2台のコンピュータ間には、（ハブもスイッチも透過、すなわち両端の機器がその存在を認識する必要がないので）リンクがあります。一方、ネットワークルーターで分割された2台のコンピュータに間はありません。相手に直接ではなく、ルーターに向けてデータを送信しなければならないことを、オペレーティングシステムが認識している必要があるからです。

ソフトウェアを記述する際には、リンク層と直接やり取りする（生パケットを送受信する、ネットワークインターフェイスを設定する、ネットワークインターフェイスのハードウェアIDを読み取るなど）ソフトウェアを除き、オペレーティングシステムがリンク層間の違いを隠してくれます。もっとも、帯域幅、レイテンシ、信頼性などの違いを、完全に吸収できるわけではありません。

IP層

リンク層の上位に位置するのが**IP層**です。IP層には、あるホストから別のホストへ、いくつもの物理ネットワークを経由してパケットを伝送する機能があります。

この層には経路制御（ルーティング）という機能があり、パケットを遠隔地の送信先に送るために重要な役割を果たします。ホストから出たパケットは、同じ物理リンクを通って近隣ルーターに到達し、そこから別のルーターに送られ、これを繰り返して最終的に目的地に到達します。応答パケットも同じように経路をたどって戻って来ます。パケットが通る経路を**ルート**とも言い、その経路上、あるルーターから次のルーターまでのリンクを**ホップ**と言います。

IP層にはリンク層の違いを吸収し、抽象化する働きがあります。特に、リンク層によってパケットの最大長（**最大伝送単位（MTU、Maximum Transmission Unit）**）は異なります。この違いを隠蔽するため、IP層ではパケットを複数の部分に分割（**断片化**）し、終点で再度組み立てるようになっています（断片化とはパケットを分割することだけを言います。TCP/IPの場合、送信元ホストはデータを細分してTCP/IPパケットに仕立てますが、パケットを細分するわけではないので断片化とは言いません）。

多くのネットワークで、パケットひとつの伝送に要する総時間は、パケット長がMTU以下であれば具体的な長さに依存しません。たとえばEthernetネットワークの場合、100バイトのパケットでも1500バイトのパケットでも、伝送に要する時間は同じなのです。そのため、断片化を繰り返し行うと、オーバーヘッドが無視できないようになります。

たとえば1500バイトのパケットを送信するとしましょう。通信経路の大部分はMTUが1400バイトですが、途中に1300バイトの箇所があるとします。すると最初に、1400バイトと100バイトのパケットに分割されます。その後、1400バイトのパケットは1300バイトと100バイトに分かれるので、最終的に3つのパケット（1300 + 100 + 100）になります。一方、最初から1300バイトと200バイトに分けて送信すれば、最後のネットワークで占める帯域幅は3分の2で済みます。

断片化にはほかにもコストがかかります。断片が1つでもなくなれば、パケット全体がなくなったのと同じことになります。すなわち、パケットの紛失率が高いネットワークでは、断片化の結果、再送を要する確率も上がってしまうのです（その結果、紛失率はさらに悪化します）。

これを回避するため、TCP通信を行う最近のオペレーティングシステムの多くが、パスMTU検出（PMTUD、Path MTU Discovery）という技術により、断片化を起こさずに通信できる最大パケット長を求め、帯域幅の使用効率を改善しています。生のパケットを送受信するコードを記述する場合、同様の処理を実装しなければなりません。しかしTCP/IPを採用すれば不要なので、できるだけそのようにしてください。

トランスポート層

IP層の上に**トランスポート層**がいくつかあります。ネットワーク処理コードを記述するほとんどの場合、いずれかのトランスポート層、またはその上に実装されたより高レベルの層とやり取りすることになります。

この階層で最もよく使われるのが、**TCP（Transmission Control Protocol、伝送制御プロトコル）**と**UDP（User Datagram Protocol、ユーザデータグラムプロトコル）**です。

いずれもIPと同じように、あるホストから別のホストにデータを伝送する仕組みですが、**ポート番号**という概念を取り入れています。これは、同じホストが複数のサービスを提供する場合に、メッセージをどのサービスに対して送るのか区別するための手段です。

UDPには、これより下の階層と同様、データが必ず宛先に届くという保証がありません。したがって、再送が必要な場合、アプリケーション側でその制御をする必要があります。UDPは、実時間ゲームなど、状態更新データが数ミリ秒以内に届かなければ無意味になってしまうため、遅延をできるだけ小さくしたい状況に向いています。しかし一般に、これを前提とする既存のプロトコルに対応しなければならない場合を除き、使うのは避けてください。

> **注意:** UDPパケットは、あるパケットが相手先に届いたことを確認してから次を送信する、ということがないので、適切に使わないとひどい輻輳が生じる虞があります。UDPを使う必要がある場合、通信が一定時間途絶えたら、パケットの送信を停止しなければなりません。広帯域幅のデータ（実時間の音声/動画ストリームなど）を伝送する場合、各エンドポイントが受信に失敗したパケット数を数えられるように、プロトコルを設計してください。また、輻輳が発生したらそのエンドポイント間の伝送速度を自動的に落とすこと、可能であればパスMTU検出の機能を実装することも必要です。

TCPには、UDPにない次のような機能があります。

- 配送の保証。TCPを使ったデータ伝送では、（一時的に伝送に失敗したとしても）送信した順序通りに全体が届くことが保証されます。所定の時間にわたって送信できなかった場合は切断されます。しかし、その時点までに受信したデータはすべて順序通りであり、途中で断片が欠落していることもありません。
- 輻輳制御。リンクにデータを大量に送った結果、経路上でデータが脱落するようであれば、送信速度を落とし（て再送し）ます。
- フロー制御。受信側ホストがビジー状態のとき、送信側に、準備が整うまで待つよう通知します。
- ストリームベースのデータフロー。アプリケーション側では、個々のレコード（UDPの用語ではメッセージ）の列ではなく、バイト列の形でデータを受け取れます。なお、TCPでばらばらのレコード列を送信する場合、その境界は（たとえばMIMEマルチパートメッセージエンコーディングにより）自分でエンコードしなければなりません。
- パスMTU検出（PMTUD、Path MTU Discovery）。断片化が起こらない最大のパケット長を調べます。

一方、UDPにできてTCPにはできないことが3つあります。

- IPv4の**ブロードキャスト**メッセージ。ブロードキャスト（同報）アドレス宛のパケットは、そのブロードバンドドメイン内（通常はそのサブネット）のホストすべてが受け取ります。
- **マルチキャスト**メッセージ。マルチキャスト（限定同報）アドレス宛のパケットは、これを購読するホストすべてが受け取ります。このホストは、LAN上であってもルーターを越えた先でも構いませんが、通常はLAN上に限るのが一般的です。
- レコード（パケット）境界の保存。UDPの場合、連続したバイトストリームではなく、個々のメッセージの形で受信することになります。

TCPもUDPも、**ICMP（Internet Control Message Protocol、インターネット制御通知プロトコル）**という、やはりIP層上のプロトコルに依存します。ICMPパケットは、接続に失敗した旨（不達など）を、接続元や送信元のホストに報告するために使います。TCPやUDPのソケットはICMPがなくても接続できますが、接続失敗を検知する能力は著しく損なわれます。

TCPやUDPの補助機能として使われるほか、ICMPのエコーパケットは、ネットワークの問題診断によく使われる、`ping`ツールの基盤でもあります。相手ホスト（またはルーター）はICMPエコーパケットを受け取るとそのまま送信元に送り返すようになっているので、ホスト間のパケット紛失率を調べることができます。

注意: IP層の上にはほかにもRSVP、IGMP、IPSECなど多くのプロトコルがありますが、いずれもある目的に特化したものなので、この資料では扱いません。大部分はファイアウォール（トラフィックの種類によって通過を制限するルーター）で遮断されます。

以上のトランスポート層プロトコルの上に、TCPであればTLS（Transport Layer Security）、UDPであればDTLS（Datagram Transport Layer Security）のような暗号化層を設けることも可能です。

同様に、トランスポート層プロトコルの上に、さまざまなカプセル化の仕組みを構築することもできます。たとえば多くのVPN（仮想専用網）は、L2TP（Layer 2 Tunneling Protocol）やPPTP（Point-to-Point Tunneling Protocol）を使います。このようなカプセル化層は仮想リンク層を提供するため、プログラム側にとっては、独立したネットワークインターフェイスのように見えます。しかし実際には、このインターフェイス上のトラフィックも、特定のホスト（VPNサーバ）宛ての他のTCP/IPストリームと変わるところはありません。

アプリケーション層

アプリケーション層はプロトコルスタックの最上位に位置します。この階層には、HTTP（HyperText Transfer Protocol）、FTP（File Transfer Protocol）などのプロトコルがあります。

アプリケーション層という名前は、この階層の詳細が、特定の（あるいは一群の）アプリケーションごとに決まっていることに由来します。すなわち、この階層はプログラムが直接制御できるのです。アプリケーション層プロトコルを独自に実装しても、単にオペレーティングシステムに任せても、これは変わりません。

レイテンシについて

ここで言うレイテンシ(遅延時間)とは、要求が往復する時間のことです。すなわち、送信した最初のパケットが送信先に到達し、送信先が応答し、それが要求元に到達するまでの時間です。レイテンシはあらゆるネットワークにあります。同じ遠隔ホストに接続しても、ネットワーク状況によってレイテンシは増減します。

ネットワーク接続が過負荷でないと仮定すると、レイテンシが生じる原因の大部分は物理法則によるものです。地球上の2点間を信号が伝わる時間の最小値は、その距離を、光や電子が媒体を伝わる速度（真空中を進む光の速度よりは大幅に遅いのが普通）で割って求めることができます。

たとえば、ニューヨークとサンフランシスコ（約2900マイル、4670km）を往復するパケットを考えてみましょう。

- 銅線の場合、データは0.66c～1c程度で移動します（cは光速、銅線の種類に依存）。したがって、少なくとも片道15～24ミリ秒、往復30～48ミリ秒かかります。これはほとんど気づかない程度のレイテンシです。
- 光ファイバの場合、データの移動速度は0.65c程度です。したがって、少なくとも片道24ミリ秒、往復48ミリ秒かかります。これもほとんど気づかない程度のレイテンシです。
- 衛星接続の場合、パケットは静止軌道の高さ（35,786km）まで行って帰ってこなければなりません。往復にはその2倍かかります。したがって、ほぼcで移動するとしても、往復のレイテンシは最小でも477ミリ秒で、ほぼ半秒に当たります。これは*明らかに*遅延があると気づく程度です。

この計算は、それぞれの媒体におけるレイテンシの*下限*を求めたものです。他に次のような要因によって、さらに遅延が加わります。

- 経路制御による遅延。ネットワークパケットは、経路上のルーターがバッファリングを行うために、さらに遅延が生じます。あるネットワークホップの容量を超えるデータが入ってくると、それが送出されてしまうまで待たなければなりません。過度に輻輳したネットワークホップを通過する際にも、バッファリングによる大きなレイテンシが生じえます。

  たとえば、あるネットワークホップが毎秒100パケットに制限されている場合、ルーターは10ミリ秒ごとにしかパケットを送出することができません。したがって、これ以上のパケットを受信した場合、空きが生じるまでバッファに蓄積しておく必要があります。送出を待っているパケットがなくなれば、10ミリ秒ごと、あるいはそれより頻繁に（平均5ミリ秒ごと）送出できます。一方、待っているパケットが3つあれば、ここで30ミリ秒の待ちが生じます。時間スロットの制約によるレイテンシは、EDGE（Enhanced Data Rate for Global Evolution）などの移動体通信網では特に顕著です。

- 再送による急激な速度下落。ホストはTCPパケットの送信後、送信先から受信確認が届くまで待ちます。一定時間内に確認できなければパケットを再送します。送信失敗の回数が増えると、送信元は再送までのレイテンシを急激に（指数関数的に）増やしていきます。パケットが紛失するのは送信先に到る経路上のリンクが飽和しているため、と想定して対処するのです。したがって、パケット紛失率が高い（1〜2%以上）と、性能が著しく低下します。
- パケットを受信、送信、転送、再生中継するハードウェア内の信号伝播遅延。たとえばEthernetスイッチは、パケット全体を受信してからでないと、送出を始めることができません。1台のスイッチやリピータの遅延はわずかでも、長距離通信ではそれが積み重なって相当の遅延になることがあります。たとえば初期の光ファイバ網では、少なくとも10kmごとにリピータを設置する必要がありました。したがって、ニューヨークからサンフランシスコまでを結ぶとすれば、約500台のリピータが必要です。

アドレシング方式とドメイン名

ネットワーク処理のどのレベルでも、各ホストには、あるネットワーク内で一意な（全世界で一意とは限らない）数字識別子をいくつか割り当てています。この識別子の具体的な形は、高レベルのネットワーク処理（IPなど）か、低レベル（生のEthernetパケットなど）かによって異なります。

リンク層のアドレシング

リンク層（物理層）では通常、各ネットワークインターフェイスを、世界的に一意性が確保されているハードウェアIDで識別します。

- Ethernet：MACアドレス（「Media Access Control」の頭字語）
- Wi-Fi：MACアドレス（「Media Access Control」の頭字語）
- Bluetooth：Bluetoothハードウェア ID（MACアドレスに似ているが同じ名前空間を共有しない）
- GSM携帯電話：IMEI（International Mobile Equipment Identity）
- CDMA携帯電話：ESN（Electronic Serial Number）またはMEID（Mobile Equipment Identifier）

この識別子で個々のネットワークインターフェイスを識別できるので、同じ物理ネットワーク上に、同じ物理アドレスの機器が複数台見つかることはありません。より重要なのは、単一のホストに同じ種類のネットワークインターフェイスが複数あっても、ハードウェアIDはやはり*異なる*ことです。

ハードウェアIDは、各機器がパケットを監視し、自分宛のものとして取り込むか、それとも無視するかを判断するために使います。物理ネットワークによっては、送信先ハードウェアIDに該当するインターフェイスの有無に応じて、各通信線に送り出すパケットを制限することもあります。このようなネットワークを**交換網**と言います。これに対し、どのホストからもすべてのパケットが見えるネットワークを**共用網**と言います。

> **注意:** プログラムがリンク層アドレスを直接使うことはほとんどありません。

IP層のアドレシング

IP層以上では、ホストを**IPアドレス**（あるいはIP番号）で識別します。これにはIPv4とIPv6という2つの形式があります。

IPv4アドレスは4バイトから成り、通常は4つの数字を小数点で区切った形で表します。たとえばループバックアドレス（データを送信元にそのまま返すアドレス）は`127.0.0.1`です。

IPv4はアドレス空間に限界がある（しかも急速に枯渇しようとしている）ため、IPv6という規格が考案されました。割り当て可能なアドレスを大幅に拡張した規格です。

IPv6アドレスは128ビット値で、通常、16ビットずつの8つの組に分け、それぞれを16進表記し、コロンで区切って表します。各組の先頭の0は省略できます（各組に1桁は必要）。たとえばIPv6のループバックアドレスである`0000:0000:0000:0000:0000:0000:0000:0001`は、`0:0:0:0:0:0:0:1`と省略して表示できます。

さらに、0の組がいくつか連続する場合、その部分をダブルコロンに置き換えて表示できます。ただしこれは1箇所だけです（2箇所以上あるとアドレスが不確定になるため）。たとえばループバックアドレス`0:0:0:0:0:0:0:1`をさらに省略すると、`::1`となります（先頭から7組の0をダブルコロンに置き換え）。

Domain Name System（DNS）

IPv4アドレスは覚えにくく、IPv6アドレスは（その4倍もあるので）輪をかけて難しくなっています。ホストを容易に言い表せるよう、DNS（Domain Name System、ドメイン名システム）が考案されました。覚えやすいこと以外にも、次のようにさまざまな利点があります。

- IPアドレスを変更してもサービスの提供を継続できます。同じホストのIPアドレスが、時間経過とともに変わることもあります。携帯電話がWi-Fiの圏外に出た、自宅のケーブルモデムを再起動した、サーバ管理者がISP（Internet Service Provider）を変更した、などの場合です。ドメイン名レコードがサーバの新しいアドレスを指すよう更新すれば、ユーザは同じ名前でそのままサーバにアクセスできます。
- 同じホストに複数のアドレスでアクセスできます。サーバの頑健性や処理性能を向上するため、複数系統の物理ネットワーク接続を用意し、複数のIPv4/IPv6アドレスを割り当てることがあります。これに同じドメイン名を対応させておけば、クライアントはどのアドレスでも接続できることになります。アドレスを追加した場合でも、クライアント側は自動的にそれを認識します。
- 複数台の物理ホストを1台のように見せることができます。たとえばルートドメイン名サーバは、実際には世界中に分散した12台のサーバで提供して、処理性能を改善しています。
- 基盤技術の変化に対応できます。たとえば、名前によってホストに接続するアプリケーションは、それがIPv4とIPv6のどちらで接続するかを気にする必要がありません。ドメイン名リゾルバにアドレスを問い合わせれば、適切な接続情報（データ構造）が得られます。

以下、ドメイン名の構成要素とDNS検索処理について、大づかみに解説します。

（ドメイン）名の中身はどうなっているか

ドメイン名とは、ある特定のホストを表す、人が読める形式の名前のことです。ピリオドで区切られた、いくつかの部分から成ります。これは例を示して説明すると分かりやすいでしょう。たとえば「mail.example.com」というホスト名があるとします。

- com — この部分は、自分では管理できません。これが複数の部分から成ることもありますが、少なくともTLD（Top-Level Domain）は、自分では管理できない部分になります。TLDは通常、以下のように分類します。
  - 一般TLD。.com、.org、.netなど。
  - 制限付きTLD。.biz、.edu、.govなど。
  - スポンサ付きTLD。.mobi、.museum、.travelなど。
  - 2文字の国別コード。.usなど。

  TLDに国別コードが含まれる場合、自分では管理できない部分が複数に分かれることになります。たとえばイギリスの会社は、.co.ukという名前空間のアドレスを使っています。

  一方、このTLDは、ドット（.）ルートドメインに属しています。通常の使い方では、ルートドメインは、ドメイン名そのものの特徴付けには寄与しません。しかし、普段意識することはありませんが、完全修飾ドメイン名はすべてピリオドで終わることになっています。たとえば「apple.com.」という名前を解決すると、厳密に1つのドメインが得られます。「apple.com」という名前も一般には「apple.com.」と解決されますが、検索に失敗した場合、最終的に「apple.com.example.com」と解決されることもありえます（「example.com」をデフォルトの検索ドメインとして設定していた場合）。
- example — ドメイン部。個々の会社や組織が管理する部分で、当該会社や組織全体に適用されます。
- mail — ホスト部。同一ドメイン内にあるサーバを一意に識別するために使います。さらに、（ホスト部を除く）ドメイン名自体も有効なホスト名です（たとえば「apple.com」）。

ドメインはサブドメインをいくつでも含むことができます。たとえば「www.david.example.com」は、「example.com」ドメインに属する「david」サブドメインの、「www」というホストを表します。あえて言うならば、「example.com」というドメインは「com」のサブドメインであり、さらにこれもルートドメインのサブドメインと言うことができます。（ルート以外の）ドメインはすべて、ある別のドメインのサブドメインになっているのです。

エンドユーザにとって「ドメイン」という用語は、ドメイン階層のうち、料金を支払って取得し、自身のサービスを提供するために使うことができる、最大の部分を指すことが多いようです。したがって、たとえばcom階層では、ドメインは「apple.com」のように2つの部分から成ります。同様にau階層では、「apple.com.au」のように3つの部分から成ります。このような使い方に限定するのは、厳密には正しくないのですが、一般にこの意味で使われています。

一般にドメイン名はASCII文字列で、英数字およびハイフンのみから成ります（国際化ドメイン名の規格を考慮しなければ）。また、大文字と小文字の区別はありません。国際化ドメイン名については、http://www.icann.org/en/resources/idnを参照してください。

名前の検索

コンピュータその他の機器が、ドメイン名によってホストに接続するためには、当該ドメイン名に対応するIPアドレスを取得しなければなりません。そのためには、DNSサーバを利用する必要があります。該当するIPアドレス（複数の場合もある）を取得すれば、そのいずれかを使って遠隔ホストに接続できることになります。

各国のさまざまな組織が連携してドメイン名システムを管理しているので、アドレス検索処理もやや複雑です。一般に、ドメインの検索処理は次のようになります。

1. コンピュータその他の機器が、ローカルDNSサーバに問い合わせを送る。
2. ローカルDNSサーバは、いずれかの中央サーバ（これを「ルートサーバ」と呼ぶ）に問い合わせを送り、当該ドメインの権威サーバを訊ねる。
3. ルートサーバは一般に、検索処理を他のサーバに委任している。たとえば「`www.david.example.com`」のIPアドレスを訊ねられたとき、それを解決して返すのではなく、別のサーバに問い合わせるよう応答し、そのIPアドレスを答える。
4. ローカルDNSサーバは、こうして知ったTLDサーバに問い合わせを送り、当該ドメインの権威サーバを訊ねる。
5. TLDサーバはデータベースを検索して、当該ドメインに関する問い合わせに応答できるサーバを調べる。
6. TLDサーバは、該当するドメインの権威サーバに検索処理を委任しており、そのIPアドレスを応答する。
7. ローカルDNSサーバは、TLDサーバから返されたサーバに対し、問い合わせを送る。これは該当するホストのIPアドレスを訊ねるものである。
8. サーバはホストのIPアドレスを返すか、さらに別のサーバに処理を委任し、そのIPアドレスを返す。

たとえば`com`というTLDに属するホストを検索する場合、IPv4アドレスの問い合わせと、IPv6アドレスの問い合わせの、2回にわたって要求を出すことになります。ルートサーバはその処理をgTLD（一般TLD、generic Top-Level Domain）サーバに委任し、gTLDサーバはさらに、当該ドメインの権威サーバに処理を委任します。権威サーバは、問い合わせの答えを返すか、またはサブドメインを管理する別のサーバに委任します。

他のサーバに対し、上記のようにして情報を得るよう要求する処理を、**再帰**的問い合わせといいます。一般に、ユーザが使うことを意図したローカルDNSサーバ（キャッシュサーバ）は再帰的問い合わせに応じることができるのに対し、あるドメインに権威を持つDNSサーバは、そうではありません。そこで、ローカルDNSサーバはいくつものDNSサーバとやり取りし、最終的にあるドメインやサブドメインに関する回答ができるDNSサーバを見つけることになります。

`whois`というツールを使って、多くのレジストリが管理するドメイン情報（あるドメインの権威サーバはどれか、など）を得ることができます。詳しくは、`whois`のmanページを参照してください。また、`nslookup`や`dig`は従来型のユニキャストDNSサーバ検索、`dns-sd`はBonjourサービスの閲覧、解決、広告に使います。

DNSの他の用途

DNSの検索機能により、IPアドレス以外の情報も取得できます。ホスト名のDNSレコードには各種のレコード型があり、それぞれに応じた情報を提供しています。よく使われるレコード型として、次のようなものがあります。

- `A` — IPv4アドレス。
- `AAAA` — IPv6アドレス。
- `CNAME` — 正式名（あるホスト名を別のホスト名の別名として扱う）。
- `DNSKEY` — DNSSEC（ドメイン名システムに暗号化を施してセキュリティを確保する仕組み）で用いる暗号化キー。DNSによる応答の正当性を検証するために使います。
- `MX`（Mail eXchanger）— 指定されたドメインに代わってメールを受け付けるサーバ。
- `NS` — あるレコードに関するDNSの委任（当該レコードに関する要求は他のサーバが応答する旨）。
- `PTR` — アドレスから正式名へのポインタ。`CNAME`レコードに似ていますが、このレコードが見つかった時点でアドレス解決の処理は終わりです。必要ならば別途、`CNAME`レコードを検索して正式名を取得しなければなりません。これは主としてDNSの逆引きに使います（IPアドレスを得るのではなく、IPアドレスから名前を取得することが目的）。また、DNS Service Discoveryが、人が読める形式のサービス名を格納するためにも使います。
- `SOA`（Start Of Authority）— 主として、クライアントが検索結果をキャッシュ保存しておける期間、このドメインに関する権威サーバなどを示すために使います。
- `SRV` — あるサービスを提供するホスト名とポート番号を記述します。DNS Service Discoveryが使います。
- `TXT` — DNS Service Discoveryが使う情報属性を記述します。

パケットの経路制御と配送

ネットワーク接続には、2ホスト間の対向（Point-to-Point）リンクもありますが、多くは3台以上のホストが関与し、ハブ、スイッチ、ルーターなどのハードウェアを経由して接続することも少なくありません。したがって各ホストは、ネットワーク上にある他のホスト宛のパケットを受け取ることもあります。そのため、自分宛のパケットかどうか認識し、そうでないパケットに、誤って応答しないようにしなければなりません。

そこで各パケットには、宛先を表すリンク層アドレスがついています。他のホスト宛のパケットは受け取っても無視します。もっとも、多くの場合これはハードウェアで実装されているので、オペレーティングシステムが他のホスト宛のトラフィックを意識する必要はありません。

Important: リンク層には正しい宛先かどうかを識別する機能がありますが、これはセキュリティ保全機能では*ありません*。オペレーティングシステムが他のホスト宛のトラフィックを、TCP/UDPソケットを介してプログラムに渡すことはありませんが、他ホスト宛のパケットをすべて廃棄するよう義務づけられているわけではないのです。`tcpdump`のようなパケット監視ツールを使えば、ネットワークインターフェイスをプロミスキャス（不特定多数）モードに切り替え、ネットワーク上にある他ホスト宛のパケットを捕捉することも容易です。

このような設計のため、2台のホストがローカル物理ネットワークを介して通信する場合、送信元ホストが送信先ホストのリンク層アドレスを知ってからでないと、パケットを送信することができません。そこで、オペレーティングシステムのハードウェアに近い層には、論理アドレス（IPアドレスなど）を物理アドレス（EthernetのMACアドレスなど）に変換する、さまざまな手段が実装されています。その仕組みはネットワークの種類によって異なります。

- Ethernetやそれに類するネットワークの場合、オペレーティングシステムはIPv4アドレスから物理アドレスを得るために、**ARP（Address Resolution Protocol、アドレス解決プロトコル）**を使います。ARP要求は、あるIPアドレスのホストに応答を求めるブロードキャストメッセージ（ネットワーク上のホストすべてに宛てたメッセージ）です。該当するホストは自分自身のリンク層アドレスを返します。

  Ethernet上のIPv6通信の場合は、ICMPを基盤とする、これに似た**NDP（Neighbor Discovery Protocol、近隣探索プロトコル）**を使います。

- Ethernetとは異種のネットワークの場合、さまざまなプロトコルが使われていますが、大まかな考え方は一般によく似ています。ネットワークドライバ（その他、代理として振る舞う他のソフトウェアやハードウェア）が、IPアドレスを、パケットの宛先を一意に識別できる、ハードウェア固有の値に変換する手段を提供します。

- 対向型のネットワーク（VPNトンネルなど）では、この種の対応付けは必要ありません。リンクの他端にあるホストに、パケットをすべて送信すればよいからです。
- 移動体通信網の場合、携帯電話が基地局に接続を申し込むと、基地局が携帯電話に、特定の周波数や時間枠を割り当てます。これが完了すると、接続は対向ネットワークと同様に動作します。

しかし、遠距離にある2台のホストが通信する場合、次のような理由で、上記のプロトコルでは不充分です。

- インターネット全体にわたってブロードキャストパケットを中継することはできない。
- 遠隔ホストのリンク層アドレスが得られたとしても、それだけではパケットの送信方法を判断できない。

このように、ARPパケットは物理LAN内に限られています。インターネットでは**経路制御**の仕組みにより、パケットを遠隔ホストに送信する方法を判断しています。具体的に言うと、**ルーター**は基盤機器の一種で、IPアドレスのある範囲から別の範囲に、データを送信する方法を知っています。

- **エッジルーター** — 単一の顧客サイトにサービスを提供する小型ルーター。通常、ネットワークの片側にあるIPアドレス範囲についてしか知りません。サイト内ネットワーク（狭い範囲のIPアドレス）に送ったパケットはある1つの物理ポート、それ以外のパケットは別のポートを通って送信されます（その経路上で、組み込み/外付けケーブルモデム、DSLモデム、その他のエンコーディングハードウェアを経由するかもしれません）。
- **コアルーター** — 大規模なインターネットバックボーン向けにサービスを提供するルーター。複数の物理接続を持ち、大規模な経路制御表を管理して、あるIP範囲内のパケットを捌きます。BGP（Border Gateway Protocol）、RIP（Routing Information Protocol）などを使って、経路情報を相互に広告し合います。新しい物理ケーブルが接続されると、その一端のルーターは隣接するルーターに対し、もう一端のルーターに到る経路がある旨を通知します。隣接するルーターはこの広告をさらに隣のルーターに伝え、順次経路情報が伝播していきます。

同様に、リンクが切断された（意図的に変更した、下水工事の際に不注意でファイババンドルを切断してしまった、など）場合も、いずれかの側のルーターが問題を検出し、パケットの経路を別のリンクに切り替えるので、利用者は何も気づかず従来通り利用できます。ハードウェア障害が起こっても動的に再設定することにより、インターネット全体の頑健性を確保する設計になっているのです。

ネットワークトポロジーは複雑で頻繁に変化するので、通信性能も複雑で頻繁に変わります。たとえば、ラップトップ機をWi-Fiルーターに接続し、2台のデスクトップ機をEthernetで接続した場合、デスクトップ機どうしの方が、ラップトップ機とデスクトップ機よりも高速に通信できます。そしてどちらも、公衆インターネット上のサイトとの接続よりは恐らく高速です。さらに、サイトとの接続はそれぞれ異なるルーターやケーブルを経由するので、速度も接続先によってさまざまです。

経路制御の詳細は、IPv4かIPv6かによって異なります。

IPv4の経路制御

IPv4アドレスは32ビットの数値で、ホスト部とネットワーク部に分かれます。ホスト部は、ある物理ネットワーク上にあるホストを一意に識別します。ネットワーク部は、当該ホストが接続されているネットワークを識別します。

> **注意:** IPアドレスのホスト部はホストを一意に識別しますが、1対1の関係とは限りません。あるホストが複数のインターフェイスを備え、複数のネットワークに接続し、それぞれにIPアドレスを割り当てていることもあります。それどころか、同じ物理ネットワークに接続する同じネットワークインターフェイスに、複数のIPアドレスを割り当てることさえ可能です。

IPアドレスのブロック分割方法に応じ、ネットワーク部は8～30ビットの範囲で設定できます。

ネットワーク経由で適切に送受信するためには、各ホストが、自分自身のIPアドレス、送信先ホストのIPアドレス、そして送信先ホストが自分自身のいずれかのアドレスと同じネットワーク上にあるか、の3つを認識できなければなりません。ホストはこの情報に基づき、パケットを送信先に直接送る（送信先が同じネットワーク上にある）か、ルーターを経由する（同じネットワーク上にない）かを判断します。

多くのホストは、ごく単純なアルゴリズムでこの判断をします。

- 自分自身と送信先の、IPアドレスのネットワーク部が完全に一致すれば、（ARPなどにより物理アドレスを取得して）直接送信します。
- そうでなければ、主物理ネットワーク上のルーターにパケットを送信します（ルーターの物理アドレスは、そのIPアドレスからARPなどにより取得）。このルーターを**デフォルトゲートウェイ**とも言います。

  IPv4アドレスについて特別な情報（たとえば送信先がVPN接続の他端にあること）を知っている場合、デフォルトゲートウェイ以外のルーターに送ることもありますが、これは例外的な取り扱いです。

ネットワーク部の長さは**ネットマスク**で表します。これは32ビットの数値で、IPアドレスのうち、ネットマスクが1であるビットの部分がネットワーク部、0である部分がホスト部になります。通常、ネットマスクもIPアドレスと同じ書式で表します。たとえばネットワーク部の長さが28ビットであれば、ネットマスクは`255.255.255.240`となります。表5-1に、`developer.apple.com`のIPアドレスをネットマスクに基づいてネットワーク部とホスト部に分ける様子を示します。

表 5-1　developer.apple.comのIPv4アドレスとネットマスクの例

	ネットワーク部	ホスト部		
IPアドレス	17.	254.	2.	129
IPアドレス（2進表記）	00010001	11111110	00000010	10000001
ネットマスク	255.	0.	0.	0
ネットマスク（2進表記）	11111111	00000000	00000000	00000000

> **注意:** ネットワーク部の長さはIPアドレスを所有するネットワーク管理者が方針を定めて管理するものです。表5-1に示したのは、理論的に`developer.apple.com`に設定可能なネットマスクの一例に過ぎません（実際とは違っています）。

ネットワーク部もホスト部も、（事実上常に）連続したビットに割り当てられるので、ネットワーク部の長さ（ネットマスクの「1」ビットの数）を数字で表し、スラッシュを前に置いて記述する略記法があります。たとえばネットワーク部が28ビット長であれば「`/28`」ネットワーク、表 5-1の例は「`/8`」ネットワーク、というように記述するのです。理論上は、「1」と「0」のビットをどのように組み合わせてネットマスクにしても構わないのですが、オペレーティングシステムやネットワーク管理ソフトウェアが、このように標準を外れた設定でも正常に動作する、という保証はありません。

ネットワーク部の値を固定したとき、ホスト部のみが異なるIPアドレスの範囲を、**サブネット**または**ネットブロック**と言います。たとえば、24ビットをネットワーク部に割り当て、8ビットをホスト部に残したサブネットが考えられます。これは256通りのIPアドレスから成るブロックです。

しかし実際には、256通りすべてが使えるわけではありません。サブネットには特別なIPアドレスを3つ確保しておく必要があります。ブロードキャストアドレス、ネットワークアドレス、ルーターアドレスの3つです。

- **ブロードキャストアドレス**とは、ホスト部がすべて「1」であるアドレスのことです。このアドレスにパケットを送信すれば、同じブロードキャストドメイン（通常はサブネット）に属するすべてのホストが受信します。LAN外に出て行くことはありません。

> **性能に関する注記:** 一般に、ブロードキャストアドレスにデータを送信することは避け、マルチキャストアドレスを使ってください。このアドレス宛のパケットは、本当に必要なホストが監視するだけでよく、また、サブネット境界を越えて送信することも可能です。

- **ネットワークアドレス**とは、ホスト部がすべて「0」であるアドレスのことです。古いオペレーティングシステムではブロードキャストアドレスとして使われていたので、歴史的な互換性の理由でこの番号が予約されています。
- **ルーターアドレス**は、ルーターのアドレスです。サブネット内のどのIPアドレスでも構いません（稀にはサブネット外であることも）が、経路制御がなくても到達可能でなければならず、したがって同じ物理ネットワーク上である必要があります。通常、ルーターには、ホスト部の最下位ビットが「1」、それ以外が「0」のアドレス（すなわち、ネットワークアドレスより1大きい値）を割り当てます。

> **注意:** 特別な例外として、対向ネットワーク（2台のホストから成り、その一方がルーターとして動作）の場合、ルーターアドレスとネットワークアドレスを別々に割り当てる必要はありません。

以上に加え、IPv4では特別な用途にいくつかアドレスを予約しています。以下に示すように、このようなアドレスは、上位ビットをパターンに当てはめて調べるだけで識別できるよう設計されています。

アドレスのタイプ	IPv4マスク
未指定アドレス アドレスがないことを表します。ホストに割り当てることはできません。	`0.0.0.0`
ループバックアドレス 自分自身（localhost）に接続するためのアドレスです。	`127.0.0.1`
マルチキャストアドレス 所定のホスト群に対してパケットを送信するために使います。	`224.0.0.0/4`
リンクローカルユニキャストアドレス 経路制御の対象にならないアドレスです。	`169.254.0.0/16`

サイトローカルユニキャストアドレス
顧客サイト内でのみ経路制御されるアドレスです。
	`10.0.0.0/8`
	`172.16.0.0/12`
	`192.168.0.0/16`

IPv6の経路制御

IPv6アドレスは次の部分に分かれます。

- 64ビットのネットワークID。これがさらに次の部分に分かれます。
  - グローバルルーティングプリフィックス - どのサービス事業者が所有する番号か、を表します。この部分を階層的に編成すれば実質的に分割管理が可能ですが、これはサービス事業者の方針によるので、外部からは見えません。
  - サブネットID - 顧客サイトの個々の物理ネットワークを識別します。顧客側で自由に編成して構いません。

  2つの分割位置は自由に決めることができます。ISPが大企業に、`/48`ブロック（48ビットのグローバルルーティングプリフィックス、16ビットのサブネット、64ビットのインターフェイス）を割り当てれば、顧客は最大2^{16}の独立したネットワークを作ることができます。一方、個々の家庭に`/64`ブロックを割り当て、ネットワーク部をすべてISPが管理する場合、顧客は単一のネットワークしか作れません。
- 64ビットのインターフェイスID。同じネットワークに属するホストを識別します（ホストのMACアドレスをもとに、プログラムで生成することもあります）。

表 5-2 IPv6アドレスの構造

ネットワーク部		ホスト部
グローバルルーティングプリフィックス nビット	サブネットID 64-nビット	インターフェイスID 64ビット

> **注意:** IPv4と同様、IPアドレスのホスト部はホストを一意に識別しますが、1対1の関係とは限りません。同じネットワークインターフェイスに、同じ物理ネットワーク上の複数のIPアドレスを割り当てることも普通です（“近隣探索(Neighbor Discovery)とIPv6アドレス割り当て” （31 ページ）を参照）。また、同じホストが複数のインターフェイスを持ち、それぞれ別のIPアドレスでネットワークに接続することも可能です。

IPv6の経路制御も、考え方はIPv4の場合に似ています。ただし、ブロードキャストアドレスやネットワークアドレスは予約されていません。代わりに特別に、「全ノード」を表すリンクローカルマルチキャストグループ（`ff02::1`）があって、同様の機能を果たします。同様に、可変長のサブネットもありません。インターフェイスIDは常に64ビット長です。

IPv6は特別な用途にいくつかアドレスを予約しています。以下に示すように、このようなアドレスは、上位ビットをパターンに当てはめて調べるだけで識別できるよう設計されています。

アドレスのタイプ	IPv6マスク	マスクのビットパターン
未指定アドレス アドレスがないことを表します。ホストに割り当てることはできません。	::/128	00000000....00000000
ループバックアドレス 自分自身（localhost）に接続するためのアドレスです。	::1/128	00000000....00000001
マルチキャストアドレス 所定のホスト群に対してパケットを送信するために使います。	FF00::/8	11111111
リンクローカルユニキャストアドレス 経路制御の対象にならないアドレスです。	FE80::/10	11111110 10
サイトローカルユニキャストアドレス 顧客サイト内でのみ経路制御されるアドレスです。	FEC0::/10	11111110 11

他のアドレスパターンはすべて、グローバルユニキャストアドレスと看做します。

ファイアウォールとネットワークアドレス変換

ルーターの多くは、どちらの側のホストから見ても、できるだけ透過的になるよう設定します。そうでないルーターを、ファイアウォールと言います。

ファイアウォールとは、ルーターのうち、トラフィックの中身を調べ、改変し、ある場合には遮断するもののことです。物理的な防火壁（火災が建物のある部分から拡大しないよう遮断）と同様に、エッジルーターとして動作し、外部の攻撃者と内部ネットワークとのやり取りを制限することにより、企業や家庭のネットワークのセキュリティを確保するために使います。

ファイアウォールは一般に、次のような目的で利用します。

- 特定のポート宛てのトラフィックを遮断する。たとえば、多くのファイアウォールが、SMB（Server Message Block）やAFP（Apple Filing Protocol）のようなファイル共有プロトコルに関係するポートを遮断して、LAN内でしかこのサービスが利用できないようにしています。
- 不正な/サービス妨害を試みるパケットを遮断する。不正なパケットは長年にわたり、Smurf、Ping of Death、Invite of Deathなどのサービス妨害（Denial-of-Service）攻撃に使われてきました。

- パケット内部を詳しく調べて、疑わしいトラフィックを検出、報告します。
- **NAT（Network Address Translation、ネットワークアドレス変換）** を実施する。パケットの送信元アドレスや送信先アドレスをつけ替える機能です。

ネットワークアドレス変換についてさらに詳しく説明しておきましょう。NATは、ファイアウォールの一方の側にあるホストから送出されたトラフィックを、別のIPアドレスから送出されたように見せかけるために、広く使われている技術です。目的に応じて構成は異なります。

- マスカレード：ファイアウォール内部からの接続を変換して、ファイアウォールから送出されたパケットのように装います。ファイアウォール内のホストが、公衆インターネット上のホストに接続しようとすると、ファイアウォールは一時的な変換規則により、送信元アドレスを当該ファイアウォールのアドレスに、送信元ポート番号をある大きな値に書き換えます。

  この構成の場合、ファイアウォール内のホストはインターネットに接続できますが、逆にインターネット側からファイアウォール内に接続することはできません。これによりセキュリティを大幅に向上できる代わりに、一部のネットワークプロトコルは利用できなくなります。
- 送信先NAT：ファイアウォール外からファイアウォール内のホストに接続する際、接続先として指定されているIPアドレスを書き換えます。これは、ある種の負荷分散を行い、複数のサーバが外部からは単一サーバに見えるようにするためによく使われます。

マスカレードを行うファイアウォールの中には、ファイアウォール内のホスト上で動作しているアプリケーションが、一時的にファイアウォールの設定を変更するよう要求して、外部から接続できるようにするものもあります。これに対応したプロトコルとしては、NAT-PMP（Network Address Translation Port Mapping Protocol）や、IGD（Internet Gateway Device）標準デバイス制御プロトコル（UPnP（Universal Plug And Play standard）の一部）がよく使われます。

OS XやiOSの場合、Bonjourに、NAT-PMPやUPnPに対応したファイアウォールを介して、ポートマッピングを生成する機能が組み込まれています。広域Bonjourを使って広告されたサービスは、自動的にマッピングされます。他の方法で広告されたサービスについては、`DNSServiceNATPortMappingCreate`を実行してマッピングを生成し、`DNSServiceRefDeallocate`で破棄することになります。マッピングは、生成したプロセスが終了したときにも、自動的に破棄されます。

動的アドレス割り当て

インターネットの黎明期には、各ホストに一意的なIPアドレスを割り当て、設定ファイルにその対応を記述していました。インターネットの普及に伴い、技術に通じていないユーザが増え、ディスクレス機器が一般的になると、ネットワークが変わるたびに人手でIPアドレスをつけ替える手間も無視できなくなりました。ラップトップ機や無線ネットワーク機器が普通に使われるようになったことも、その動きに拍車をかけています。

この問題を解決するため、さまざまなプロトコルが開発されましたが、現在ではDHCP（Dynamic Host Configuration Protocol）や、IPv6の近隣探索、DHCPv6、リンクローカルアドレシングに集約されています。以下、このプロトコルについて詳しく説明します。

DHCP（Dynamic Host Configuration Protocol）とDHCPv6

IPv4ではDHCPが、アドレス自動割り当ての仕組みを提供しています。コンピュータその他の機器は中央サーバに、当該ネットワークに適合したIPv4アドレスを要求します。サーバによって、割り当て可能なアドレスプールからランダムに割り当てる方式と、要求元ホストのMACアドレスに基づき管理者が手で割り当てる方式があります。

IPv6では、DHCPv6を使って同様に処理することもできますが、次の節で説明するSLAACの仕組みを使ってIPv6アドレスを取得することも可能です。

DHCPサーバは、サブネットマスクやルーターアドレスなど経路制御に必要な情報、あるいはDNSやディレクトリサーバのアドレス、さらに、プロトコル拡張として定義されたさまざまなデータを渡すこともできます。

DHCPの動作原理を大まかに説明すると、次のようになります。まず、クライアントはIPv4アドレス要求を、ブロードキャストで送信します。サーバがアドレスを割り当てた場合、クライアントは所定の期間、そのアドレスの「払い出し」を受けたと言います。時間切れになるまでの間であれば、更新も可能です。だからといって更新に応じる義務はないのですが、ほとんどのサーバが応じるようになっています。

一般に、初回の接続時には、DHCPサーバとの間で、ブロードキャストUDPパケットを使ってやり取りします。既存の（まだ有効な）IPアドレスを更新する場合、（クライアント側の判断で）ユニキャストUDPパケットでやり取りしても構いません。

クライアントは、さらに追加データ（DNSサーバ、ネットワークボリュームのマウント、Active Directoryのドメインコントローラなど）の要求を送ることもできます。これは、特別なオプションをつけた要求を送って応答を待つ、という形で行います。

近隣探索(Neighbor Discovery)とIPv6アドレス割り当て

IPv6プロトコルには、近隣探索の機能が（ICMPv6上に）実装されています。これには2つの目的があります。

- 同じ物理ネットワーク上にある他のホストの物理アドレスを調べる。これはARPに代わる機能です。
- SLAAC（StateLess Address AutoConfiguration、ステートレスアドレス自動設定）。これはDHCPの機能の大部分に代わるものです。

ホストが初めてネットワークに接続するときには、自ら割り当てたIPv6アドレス（設計上、広域的に一意）を使います。ホストはこのアドレスを使って、近隣探索要求をルーターに送ります。ルーターは、IPv4におけるDHCPと同様の情報を返します。すなわち、ルーターのIPアドレス、ネットワークプリフィックス（大雑把に言えばネットワークアドレスやサブネットマスクと同じようなもの）、DNSサーバのIPアドレスなどです。クライアントはこれをもとに、公衆インターネットとの通信に使えるIPv6アドレスを組み立てます。

自ら割り当てるリンクローカルアドレスのほかに、各ホストは少なくとも2つ、永久アドレスおよびプライバシーアドレスというIPv6アドレスを作成して割り当てます。

永久アドレスは、ホストの物理（リンク層）アドレスをもとに生成します。したがって一般に、別のネットワークに移動してもホスト部は変わりません。同じネットワーク上の他のホストが当該ホストに対してデータを送る、一貫した宛先を提供するために使います。

各ホストはほかに、プライバシーアドレスをいくつか割り当てます。これはランダムに生成され、時間の経過とともに変わります。したがって特定のハードウェアに結びついたものではありません。デフォルトでは、外向きの接続の際、送信元としてプライバシーアドレスを使うことになっています。

> **注意:** ただし、一般に公開するサービスを登録する場合、アプリケーションは（将来変わる可能性が高い）プライバシーアドレスを使うべきではありません。Bonjourでサービスを登録すれば、これは適切に処理されます。Bonjourについては次の節で説明します。

リンクローカルアドレスとBonjour

ドメイン名は、永続的なサーバにとっては理想的な仕組みですが、費用がかかるばかりでなく、技術に通じた人がその設定をする必要があります。また、一般に公開しないサーバであれば、世界中につながるドメイン名を用いる意味はほとんどありません。さらに、標準的なドメイン名は、DHCPサーバその他、動的にIPアドレスを割り当てる仕組みと相性がよくありません。動的に構成されるネットワークが増えてくるにつれ、別の解決策が必要になってきました。

その代替手段としてOS XやiOSには、Bonjourという、設定の手間が不要なネットワーク処理の仕組みを実装されています。Bonjourは次の3つの部分から成ります。

- IPv4におけるリンクローカルアドレス - DHCPサーバその他、IPアドレスを割り当てる仕組みが使えない場合に、自らIPアドレスを割り当てる仕組みです（IPv6の場合はこのプロトコル自身にアドレス割り当ての仕組みがあるので、Bonjourに同じ機能を実装する必要はありません）。
- マルチキャストDNS — 基盤サーバがない、あるいは管理されないローカル名の方が便利な場合に、DNSと同様のアドレス解決機能を提供する技術です。
- DNS Service Discovery — サービスを登録、探索する手段です。

上記の技術を組み合わせると、永続的なDNS基盤がなくても、同じ物理ネットワーク上の複数のホストがサービスを広告し、相互に探索できます。

通常、BonjourはマルチキャストDNSを使って、DNS Service Discoveryの問い合わせを、LAN上のホストすべてに送信し、あるサービスを提供していないかどうか訊ねます。件のサービスを提供しているホストは、問い合わせ元にその旨のメッセージを返します。その結果、問い合わせ元はサービスを提供するホストを見つけ、利用できることになります。各ホストに対して個別に問い合わせる必要はありません。

DNS Service Discoveryは、リンクローカルアドレスを組み合わせて使うことにより、事前の設定作業なしでネットワーク処理を行うことができます。設定済みのネットワーク基盤がなくても、ネットワーク機器を探索できるのです。たとえば、アドホック無線ネットワークを介して、2台のコンピュータがファイルを共有する、といったことが可能です。

リンクローカルアドレスはきわめて分かりやすい仕組みです。

- IPv4では、ホストがDHCPサーバからアドレスを取得できなかった場合、所定の範囲（`169.254.*.*`）からランダムにIPアドレスを選び、リンク層のプロトコル（ARPなど）を使って、そのアドレスを割り当てられたホストがないかどうか問い合わせます。応答があった場合、別のアドレスを選んで再度試みます。なければこれを、自分自身に割り当てられたアドレスとします。
- IPv6の場合、どのインターフェイスにも、`fe80::`プリフィックスの空間にリンクローカルアドレスがあります。SLAAC、DHCPv6その他の機構により、別のアドレスを割り当てられていても、これは変わりません。このアドレスは、インターフェイスに物理アドレス（EthernetのMACアドレスなど）があれば、これをもとに生成します。なければこのプリフィックス内からランダムに選び、SLAACの重複アドレス検出プロトコルにより一意性を確保します。詳しくは、RFC 4862を参照してください。

以上のIPアドレスはすべて同じサブネットにあるので、あるLAN上のリンクローカルアドレスが割り当てられたホストは、ルーターを介することなく相互に通信できます。したがって、他の基盤機能がなくても、相手を探索できるのです。

> **注意:** Bonjourの3つの技術は独立にも使われます。特にマルチキャストDNSは、サービスを探索する必要がない場合に、LAN上のホストの名前を調べるために使えます。同様に、旧来のDNSサーバは、Wide-Area Bonjourを介して、サービスに関する情報を提供できます。

書類の改訂履歴

この表は「*ネットワーキング コンセプト*」の改訂履歴です。

日付	メモ
2012-07-19	パケット、経路制御、動的アドレス割り当て、サービス探索などの基本事項について概要を説明した新規ドキュメント。

用語解説

ARP（Address Resolution Protocol、アドレス解決プロトコル） コンピュータその他の機器のIPアドレスをもとに、物理アドレスを調べるプロトコル。

アプリケーション層（application layer） ネットワークプロトコルスタックの最上位層。個々のアプリケーションに特有のデータ形式やプロトコルを定義しています。たとえばHTTP（HyperText Transport Protocol）はアプリケーション層のプロトコルです。

ブロードキャストアドレス（broadcast address） LAN上の全デバイス宛てに同時にパケットを送るための特別なアドレス。

コアルーター（core router） 主なインターネットバックボーン経路に対してサービスを提供するルーター。大量のトラフィックを捌き、同時に多数の経路を管理しなければならないので、処理能力の高い機器を使います。経路広告に参加して、ネットワークトポロジーの変化を探索あるいは広報します。

デフォルトゲートウェイ（default gateway） 宛先IPアドレスへの経路がシステム経路表に載っていない場合に、外向きトラフィックの送り先とするルーター。

ドメイン名（domain name） インターネット/イントラネットサイトを識別する、人が読める形の名前。「developer.apple.com」など。アプリケーションは、ドメイン名を解決することにより、当該サイトにデータを送るために必要なIPアドレス（IP address）を取得できます。

エッジルーター（edge router） 顧客サイトと上流のISPを接続するルーター。通常、2つか3つ程度のネットワーク間を結ぶだけであり、経路広告に参加することもありません。

カプセル化（encapsulation） パケットを（通常は種類の異なる）別のパケットに包む処理。たとえばLAN上で、IPパケットはEthernetパケットにカプセル化されます。Ethernetパケットには、LAN内の宛先に関する情報が書き込まれています。IPパケットには、当該パケットが公衆インターネットに出た後、何をすればよいか、に関する情報が書き込まれています。

ファイアウォール（firewall） 通過できるトラフィックの種類を制限するルーター。所定のポートを遮断する、ネットワークアドレス変換により一方の側にあるホストのIPアドレスを隠蔽する、不正なパケットを遮断する、その他さまざまなパケット書き換え処理を行う、などの機能があります。

断片化（fragmentation） 最大パケット長（MTU、Maximum Transmission Unit）が小さい箇所を通すために、パケットをより小さな部分に分割すること。

ヘッダ（header） パケットについては、その先頭（ペイロードに先行）に当たる、送信先に関する情報を記述した部分。HTTPについては、要求や応答の内容に関する情報（ホスト名、キャッシュ方針など）を伝える一連の値。

ホップ（hop） あるホストから別のホストに到る経路（route）上にある、個々の物理リンク。

ホスト（host） ネットワークに接続された機器。クライアントコンピュータ、サーバ、携帯電話、さらにはネットワーク対応プリンタまで、さまざまな機器があります。

ホスト名（hostname、host name） DNSで特定のホスト（あるいは単一ホストのように見せかけるホスト群）を表す名前。

基盤機器（infrastructure device） ネットワークの基本機能を提供する機器。ルーター、Wi-Fiアクセスポイント、Ethernetスイッチなど。

ICMP（Internet Control Message Protocol、インターネット制御メッセージプロトコル） 帯域外（out-of-band）制御メッセージをやり取りする低レベルのネットワークプロトコル。オペレーティングシステムがTCP接続を確立する際に使います。主として、接続失敗時の通知（接続拒否、ホスト不達など）に用います。また、`ping`、`traceroute`などのネットワーク診断ツールも利用します。

IP層（Internet Protocol layer） インターネットを経由してパケットを伝送するための基本機能を提供するネットワーク階層。物理層（ハードウェア相互接続）の上位、トランスポート層（TCP、UDPなど）の下位に位置します。

IPアドレス（IP address） インターネット上にある個々のホストを一意に識別する番号（「インターネットプロトコルアドレス」を短縮した呼称）。IPv4アドレス（IPv4 address）とIPv6アドレス（IPv6 address）という2種類の形態があります。

IPv4アドレス（IPv4 address） 4つの8ビット数値（全体で32ビット）から成るIPアドレス。たとえばdeveloper.apple.comのIPアドレスは17.254.2.129です。

IPv6アドレス（IPv6 address） 8組の16ビット16進数（全体で128ビット）から成るIPアドレス。すべて0である組の連なりを1箇所のみ、「::」で置き換えて略記できます。 たとえばexample.comのIPv6アドレスは2001:500:88:200::10です。

レイテンシ(latency, 遅延時間） パケットが宛先に到達するまでに要する時間（通常はミリ秒単位）。往復の待ち時間、すなわち、パケットが送信先に到達し、その応答が返ってくるまでの時間で表すのが普通です。これは2つの理由で重要です。第1に、接続を確立するために要する時間が増える原因となります。第2に、応答が返ってくるのを待って次の要求を送信するプロトコルの場合、性能が大きく抑えられます。

リンク（link） ネットワーク上にある2台のホスト間を結ぶ物理接続（またはそれをエミュレートする仮想接続）で、（リンク層スイッチを除き）中継ルーターがないもの。

リンク層（link layer） ネットワークプロトコルスタックの最下層。LANその他の物理リンク（link）を介して、あるホストから別のホストに、物理的にパケットを伝送する機能を提供します。

監視ソケット（listening socket、listen socket） 入ってくる接続を待ち受ける設定したソケット。

MTU（Maximum Transmission Unit、最大伝送単位） あるリンク（link）を介して配送できる最大パケット長。実際の通信ハードウェアによって制限され、通常、ハードウェアで処理できる最大長の物理パケットで伝送可能な、最大ペイロード長で表します。しかし（ギガビットEthernetなど）場合によっては、デフォルトの

MTUをソフトウェアでさらに制限し、大きなパケットを伝送できない旧ハードウェアとの後方互換性を確保することもあります。

マルチキャスト（multicast） ネットワーク上の複数のホストに同時配送する特殊なパケット。ただしすべてのホストを対象とするブロードキャストとは異なります。

NDP（Neighbor Discovery Protocol、近隣探索プロトコル） Ethernet上のIPv6で、物理ネットワーク上の他の機器に関する情報を収集するプロトコル。近隣機器の物理アドレスを調べ、ルーターやDNSを探索し、上流リンクに関する情報（MTU（Maximum Transmission Unit、最大伝送単位）その他）を取得するなど、多くの機能があります。

ネットブロック（netblock） サブネット（subnet）を参照。

ネットマスク（netmask） IPv4アドレスのうち、ネットワーク部とホスト部がどのビットに当たるかを表すもの。送信先アドレスのネットワーク部が送信元のそれと同じであれば、どちらも同じサブネット（subnet）にあると判断します。

ネットワークアドレス（network address） IPv4ネットワークで予約されている特別なアドレス。ホスト部がすべて0であるもの。旧オペレーティングシステムではブロードキャストアドレスとして使われていたので、互換性の理由で予約されています。

NAT（Network Address Translation、ネットワークアドレス変換） ファイアウォールで実行するパケット書き換えの一形態で、送信元または送信先のIPアドレスを書き換えて伝送するもの。複数の機器からのトラフィックを、単一の機器から出ているように見せかけて、セキュリティ保全や負荷分散に役立てる使い方が一般的です。

ネットワークインターフェイス（network interface） ハードウェア（または仮想ハードウェア）のうちリンク（link）のエンドポイントを表す部分。

パケット（packets） コンピュータネットワークを介して送信されるデータの単位。

パスMTU探索（path MTU discovery） ホストが送信先に断片化を起こすことなく送信できる、最大パケット長を調べる処理。適切な単位に断片化して送信することにより、再断片化することなく最終目的地まで送信できるようにする効果があります。「Don't Fragment」ビットを設定したパケットを送信する方法で調べます。「Fragmentation Needed」ビットが立ったICMPパケットが、パス上のルーターから返された場合、少しずつ短く設定して試み、これを送信先に到達するまで繰り返します。MTU（Maximum Transmission Unit、最大伝送単位）も参照。

ペイロード（payload） パケットで送信するデータ本体（パケット自体と区別する場合の呼び方）。

物理層（physical layer） リンク層（link layer）を参照。

ポート番号（port numbers） あるホストが提供する、個々のサービスを識別するための番号。これはさらに、TCPポートかUDPポートかに分かれます。

再帰（recursion） 再帰問い合わせを使うこと。クライアントがローカルDNSサーバに対して名前解決を要求する形態です。再帰問い合わせに応じる設定のDNSサーバは、ルートサーバに問い合わせて応答を得、それに基づいて次のサーバに問い合わせ、これを繰り返して最終的に、問いに対する回答を得る仕組みになっています。再帰（recursion）も参照。

経路（route） パケットがあるホストから別のホストに到る通り道。同じ物理ネットワーク上にあれば、単一のリンクが経路になります。そうでなければいくつかルーター（router）を経由することになります。

ルーター（router） 複数のネットワーク間でパケットを受け渡しする機器。経路制御規則に基づき、受け取った各パケットをどのネットワークに渡すか判断します。多くのルーターは他のルーターと情報をやり取りし、ネットワークリンクの増減に応じて経路制御規則を最適化します。

ルーターアドレス（router address） ルーター（router）のIPアドレス。

経路制御（routing） ある物理ネットワーク上のパケットを別の物理ネットワークに転送する処理。経路制御規則に基づいてどのネットワークに渡すかを判断します。これを実行する機器をルーター（router）と呼びます。

共有ネットワーク（shared network） ネットワーク上の各機器がすべてのパケットを受信するネットワーク。交換ネットワーク（switched network）と対立する方式です。

サブネット（subnet） IPアドレスの範囲。パケットの送信元と送信先が同じサブネットに属すれば、中継ルーターを介することなく、直接送信されます。

交換ネットワーク（switched network） 物理ネットワークのうち、「スイッチ」という基盤機器が、宛先に応じてパケットの送り先を変えるもの。宛先ホストだけがパケットを受信すればよいので、ネットワーク性能が向上します。共有ネットワーク（shared network）と対立する方式です。

トレイラ（trailer） パケットの（ペイロードに続く）末尾部分。通常、ペイロードデータのチェックサムを収容します。

TCP（Transmission Control Protocol、伝送制御プロトコル） トランスポート層のプロトコルのひとつ。両方向でストリームベースのデータ伝送、フロー制御、配送保証（自動再送）などの機能があります。UDP（User Datagram Protocol、ユーザデータグラムプロトコル）と対立する方式です。

トランスポート層（transport layer） IP層の上に位置するネットワーク階層。ポート番号、配送保証、フロー制御、チェックサムなどの機能を提供します。代表的なトランスポート層プロトコルとして、TCP（Transmission Control Protocol、伝送制御プロトコル）とUDP（User Datagram Protocol、ユーザデータグラムプロトコル）があります。

UDP（User Datagram Protocol、ユーザデータグラムプロトコル） トランスポート層のプロトコルのひとつ。単方向でパケットベースのデータ伝送、ベストエフォート型の配送（自動再送なし）などの特徴があります。TCP（Transmission Control Protocol、伝送制御プロトコル）と対立する方式です。

Apple Inc.
1 Infinite Loop
Cupertino, CA 95014
U.S.A.

Apple Japan
〒106-6140 東京都港区六本木 6
丁目10番1号 六本木ヒルズ
http://www.apple.com/jp