

コンピュータリテラシ#2 – インターネットの原理

久野 靖 (電気通信大学)

2017.5.19

1 今回の目標

今回の目標は次の通りです。

- Unix システムへのログインとコマンド実行ができるようになる — 本科目でも他の科目でも Unix 上で実行したり Unix を題材とすることは多いので Unix を使いこなせる必要があります。
- インターネットに関する主要な概念を理解する — ネットワークは使うだけでなく研究の対象や実験の土台ともなるので、原理や特性を知っておく必要があります。

2 Unix システムの利用

2.1 Unix システムへの接続

前回、コンピュータは「情報を処理する装置」ということをお話ししました。また後の回で改めて出て来ますが、コンピュータの動作はすべてプログラムを実行するかたちで行われます。

ということは、あなたが手元のコンピュータの電源を入れた後、マウスで画面を操作したりするのも、すべて「そのような動作をしてくれるプログラム」が動いて行われているわけです。この部分を OS (オペレーティングシステム — Operating System) と呼びます。コンピュータの操作方法や主要な機能は OS によって変わってきます。ここで説明する Unix もその 1 種別です。

皆様が普段目にする OS は PC (パソコン) 用の OS である Mac OS や Windows だと思いますが、Unix はネットワーク関係の機能が豊富に準備されていて、インターネット上でさまざまなサービスを実現するために多く使用されています。また、皆様が研究室で研究したり論文を書くのにも Unix が多く使われています。そういうわけで、この科目の実習を行う手段として Unix を主に用い、ついでに Unix に習熟して頂きます。

演習に使用する Unix システムは `sol.edu.cc.uec.ac.jp` という名称のシステムです。これにネットワーク経由で接続するためには、演習室の環境からであれば、「Terminal (`sol.edu.cc.uec.ac.jp`)」を起動し、続いてパスワードを入力することで行えます。¹

2.2 CUI と GUI

Unix に接続すると、まずコマンド (command、指令) が打ち込める状態になっています。ここでできることは、コンピュータにやらせたいことを、文字の並びであるコマンドの形でコンピュータに与えることだけです。そして、コマンドの出力も同じ窓に続いて出て来ます。このようなコンピュータの使い方を **CUI** (Character User Interface または Command User Interface) と呼びます (図 1)。

これに対して、普段皆様が PC で使っているような、画面にデスクトップやメニューなどが表示されていて、それをマウスやタッチパッドなどで指して操作するようなものは **GUI** (Graphical User Interface) と呼びます。

GUI の方が分かりやすそうなのに、なぜ CUI があるのでしょうか。それは、次の理由によります。

¹このシステムには学外からでも接続できますが、そのためには「SSH プロトコルに対応した遠隔ログインソフト」が必要です。Mac OS には `ssh` というコマンドが設置済みで、これを使うことができます。Windows の場合は、たとえば PuTTY というフリーソフトを導入して使うことができます。

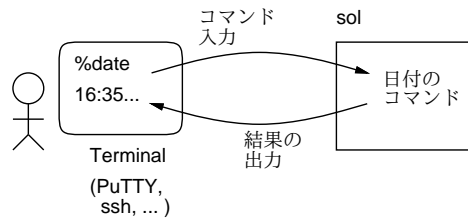


図 1: CUI の概念

- ネットワーク経由で遠隔地から操作する場合は、手前側と向こう側の間で通信に時間が掛かったり、両者のソフトウェア (たとえば OS) の違いがあり、GUI が使えないことがある。
- コンピュータを操作することの基本は「これこれの動作を下さい」とコンピュータに指示することであり、それにはコマンドで指示するのが一番直接的かつ応用範囲が広い。²
- コマンドは覚える手間が掛かる代わりに、いちど覚えてしまえば GUI で操作するよりも高速である。つまり CUI の方が「プロ向き」である。

コマンドを覚えるのが大変そうだと思うかもしれませんが、これから毎回テーマごとにいくつかのコマンドが出て来るだけです。演習をきちんとやっていたら問題ありません (皆様がこれまでに英単語を覚えて来た数と比べたら、合計でも 100 分の 1 未満だと思います)。

演習 1 `sol.cc.uec.ac.jp` に接続して下さい。すると「[...]\$」のような文字列が表示されると思います。これを「プロンプト」と言い、コンピュータからの「さあコマンドを入力してください」というメッセージになっています。その状態で次のことをやってみなさい。

- 「`date[RET]`」を実行する。`[RET]` は RETURN ないし ENTER と書かれたキーの意味。このキーの通常の用途は「1 行終り (改行)」だが、CUI では「コマンドを実行せよ」という意味になる。`date` は日付と時刻を出力するコマンドなので、画面に日付と時刻が表示され、その後再度プロンプトが表示される。(以後は `[RET]` の記載は省略します。)
- 「`exit`」を実行してみる。これはシステムの実行を終了するコマンドなので、接続が終る。(そのことを確認したら再度接続。)
- 「`sleep 5`」を実行。これは「5 秒間ただ待ってから終了する」コマンドなので、少しの間待たされ、そしてコマンドが終了すると (出力はないので) 次のプロンプトが現れる。このように、コマンドとコマンドに与える「追加指定」(オプションやパラメタと呼びます) との間は「 」(空白文字) で区切る。
- 「`sleep 100`」を実行する。そして (100 秒は待ちたくない)、`^C` (Control キーを押しながら C のキー) を打つと、コマンドが中止され次のプロンプトが現れる。間違っても長くかかるコマンドを開始してしまった際はこのように `^C` を使う。

なお、「`man` コマンド名」で指定したコマンドのマニュアルが見られます (`date`、`sleep` で試してください)。`man` は 1 画面より長い情報を見るためのコマンド `less` を呼び出します。`less` では次の 3 つのキーが使えます: `[SP]`(空白) → 次の画面を見る、`[b]` → 1 画面戻る、`[q]` → 終了。

3 ネットワークの基礎概念

3.1 イントロ: 地球の裏側までどのくらい? exam

今日の社会では、世界のすみずみまにでまたがるネットワーク (インターネット) は欠かせない機能です。しかし皆様は、単に便利に使っているだけで、仕組みを学んだり、調べたりしたことは無い

²GUI ではメニュー等に含まれていない機能は使えない。このため、Windows や Mac OS でも、頻繁に使わない機能はコマンドで起動する必要がある。

と思います。しかしこの科目は原理・仕組みまで考えることがテーマですから、そのようなことをしていきます。まず手始めに、次の疑問を考えてみることにしましょう。

インターネットで地球の裏側(ブラジル)まで行ってくるのに、どのくらいの時間で済むの?
少し手をとめて、片道どれくらいの時間だと思うか、あなたの予想をメモしてください。

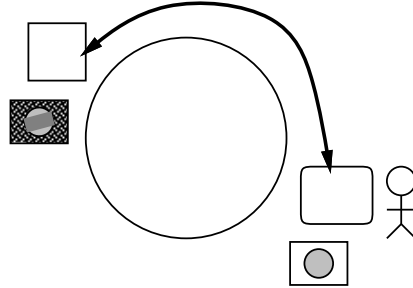


図 2: ブラジルまで行って帰ってくる時間は?

では実際に測ってみましょう。それには次のようにすればよいのです (図 2)。

- Google などで「ぜったいにブラジル現地にありそうな施設」の Web ページを検索して探す。ここでは地元のホテルっぽい施設を探し、「Royalty Hotel Barra」を見つけました。
- そのサーバ名(ブラウザのアドレス窓に出る名前)の部分にある名前「www.royaltybarra.com.br」を指定して ping コマンドを実行し、「パケット往復時間」を調べる(以下の表示にあるバイト(byte)というのはデータ量の単位で、1 バイトは 8 ビットです。パケットについては後述)。

```
[...]$ ping www.royaltybarra.com.br
PING www.royaltybarra.com.br (216.245.192.26): 56 data bytes
64 bytes from 216.245.192.26: icmp_seq=0 ttl=50 time=152.154 ms
64 bytes from 216.245.192.26: icmp_seq=1 ttl=50 time=152.103 ms
64 bytes from 216.245.192.26: icmp_seq=2 ttl=50 time=152.047 ms
64 bytes from 216.245.192.26: icmp_seq=3 ttl=50 time=152.153 ms
^C
--- www.royaltybarra.com.br ping statistics ---
4 packets transmitted, 4 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 152.047/152.114/152.154/0.044 ms
[...]
```

- 往復時間が 150msec 程度だから、片道は 75msec 程度と分かる。

演習 2 自分でも Royalty Hotel のサイトまでのパケット往復時間と経路を確認してみなさい。確認できたら、以下の課題をやってみなさい。

注意! 多人数で 1 つのマシンからあちこち ping すると攻撃と勘違いされるので、sol ではやらず、個人 PC かスマホ (ping や traceroute ができるアプリがあります) でやること。

- どこか「ネット的に遠い」と思う地域を選び、そこにある施設(複数)までのパケット往復時間を計測して「遠さ」を確認してみなさい。できれば、そこまで行くのにどこを通過しているのか予想してみて、その地域までの時間も計測して確認みるとなおよいでしょう。
- 世界の複数地域までのパケットの往復時間を調べ、およその傾向をまとめなさい。できればそれらを整理して世界のネットワークのつながり方の予測できるとなおよいでしょう。³

³後から出て来る traceroute を使ってその正しさを確認することができます。

- c. 本来なら遠くにあると思われる施設のサーバへの往復時間が異常に速いものの例を調べて挙げてみなさい。できれば、どのようなサイトについてそのようなことがなされているかの傾向を分析できるとなおよいでしょう。

3.2 ネットワークとその目的 exam

コンピュータネットワークとは何でしょうか？ 形としては、複数のコンピュータシステムが互いに通信できるように接続されたものがネットワークです。では、ネットワークを構成する目的は何でしょうか？ 次のものが挙げられます。

- a. 資源の共有 — あるコンピュータ内のデータを別のコンピュータでも利用できるようにする (データの共有)、あるコンピュータの能力が不足したら一部を別のコンピュータで処理する (処理能力の共有)、など。
- b. 信頼性 — 1 台のコンピュータだとそれが止まってしまうと処理が停止してしまうが、複数のコンピュータをネットワークで結合したのなら、1 台が壊れても残りで処理を進めるようにできる。
- c. 経済性 — 大きなコンピュータ 1 台で色々やらせるより、量産型のマシンを必要な台数だけネットワークで結合する方が廉価。
- d. 段階的成長 — 1 台のコンピュータで能力が不足したら、もっと大きいマシンに置き換える必要があり大変だが、複数台のネットワークであればマシンを追加することで能力を増やせる。
- e. 通信媒体 — 距離的に離れたシステムどうしを接続することにより、新しいタイプの応用が可能になる。

どの目的を主とするかによって、ネットワークの形態は異なります。たとえば、信頼性や経済性のためにネットワークを構成するのなら、その各コンピュータ間の距離は近く、それらの間を高速なネットワークで結ぶことになります (**LAN** — Local Area Network、局所ネットワーク)。一方、離れたところとのデータ共有や通信が目的なら、そのネットワークは長い距離を結ぶものでしょう (**WAN** — Wide Area Network、広域ネットワーク)。

3.3 回線交換とパケット交換 exam

古くからあるネットワークの代表例は固定電話です。電話はもともと、2つの機器の間を直接繋ぐと遠隔地で通信ができるというところから始まりました。そして、個人の家から電話が引けるようになった後も、電話局で交換手が配線を繋ぎ、最終的に自分と相手との間に線がつながると通話できる、という点は同じでした。その後技術の進歩により、番号をダイヤルすると自動で相手につながる等の変化はありますが、「最初に線を確保し、その後通話し、終わったら切断する」という点はずっと変わっていません。これを回線交換 (line switching) と呼びます。⁴

回線交換の弱点は、最初に線を確保する手間が必要で、線が確保できないと通信もできないことです。利点は、一度つながった線は自分たち専用に確保され、常に 100% 使えるということです。⁵

一方、インターネットの方式はこれとは違っていています。データは上限サイズが決まったパケット (packet) という単位で、ネットワークに常時送り出せます。各パケットには宛先が書いてあり、中継点ごとに宛先の方向に中継することで最終目的地まで到達します。これをパケット交換 (packet switching) と呼びます。パケットとは「葉書」「小包み」だと思えばよいでしょう (図 3)。先の ping コマンドはこのパケットの往復時間を測っていたのでした。

パケット交換の利点と弱点は先の裏返しです。パケット交換では接続の手間が無く、経路の途中では多数の通信のパケットが「相乗り」するので回線が有効活用できます。ただし、通信量が多くなり

⁴ 携帯電話も固定電話も今日では実態はパケット交換になりました。

⁵ 話している二人が沈黙している間も線は確保されたままなので、無駄があるとも言えます。

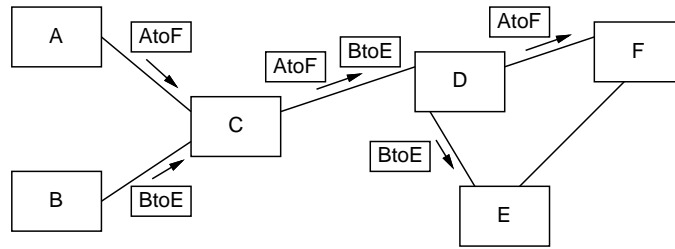


図 3: パケット交換の原理

すぎると、既に通信している相手どうしでも通信しにくくなります (到達時間が長くなったり、パケットが失われます)。また、1パケットに入らない大きなデータを送る場合には、複数のパケットに分けて送り、相手先で組み立てる必要があります。そして、パケットは独立に送られるので、一部だけ無くなったり、途中で追い越しが起きたりします (それにどう対処するかは、すぐに説明します)。

3.4 システムと階層構造 exam

コンピュータシステムもそうですが、多くのシステムは「階層構造」(layered architecture)を持ちます。その基本的な考え方は、「全体を層状に構成し」「下の層の詳細は上の層に影響しないようにする」ことです。

たとえば、あなたが遠方の相手方と取り引き契約の交渉をするために、その書類を秘書に送らせるとします。秘書が文書を送るのには「郵便」「クロネコメール」「FAX」など複数の手段がありますが、それはあなたは関知する必要がありません。秘書はどの手段を使うかで作業内容が異なりますが、郵便を使うとして、郵便ポストに入れたら、郵便局が集配に自動車を使うか、バイクかといったことは関知する必要がありません (図 4)。逆に秘書はあなたの検討の内容には関知しないし、郵便局は秘書が送ったものの内容には関知しないわけです。

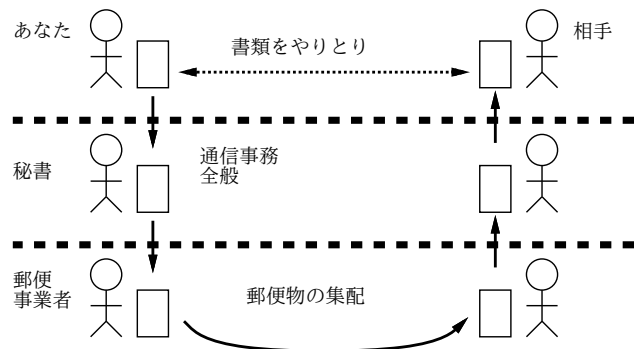


図 4: 階層構造

階層構造を用いると「それぞれの層の中をうまく構成する」と「隣接する層とのやりとりをする」ことだけに限定して考えれば済むので、問題の複雑さが小さくなります。ネットワークシステム全体もそういう風に構成されているわけです (そうしないと作れないくらい込み入っている)。

3.5 プロトコルとその考え方 exam

次にプロトコル (protocol) について説明しましょう。プロトコルとは「通信規約」などと約されることもあります。同じ階層の相手どうしでうまくやりとりするための約束ごとです。先の例では、値段の交渉のしかたは「あなた」と「相手」のプロトコルですし、宛先を指定して書類を受け渡すのはそれぞれの「秘書」さんどうしのプロトコルです。

ネットワークは非常に多数の機能の集まりですから、それらの機能を実現する多数のプロトコルを必要とします。それらのプロトコルは、上で説明したように階層構造になっています。1つの層を1つのプロトコルが占めている場合もありますし、層によってはさまざまな用途に応じて複数のプロトコルが使い分けられるものもあります。

先に「パケットは失われたり追い越しがあつたりする」という話をしました。しかし実際には、たとえば音楽ファイルを送った時に、曲の途中が抜けたりフレーズが入れ替わったりすることはありませんね？ これも、プロトコルの機能によってエラーのない伝送が実現されているからです。

どうしたらそんなことが可能になるのでしょうか それは例えば、次のようにするのです (図5)。

1. 送り側は送り出すパケットに一連番号をつける。
2. 受け側は相手からパケットが来たら確認パケットを送る。
3. 送り側は一定時間待っても確認が来なかったら再送する。
4. 受け側は一連番号の順にパケットを正式に受け取る。

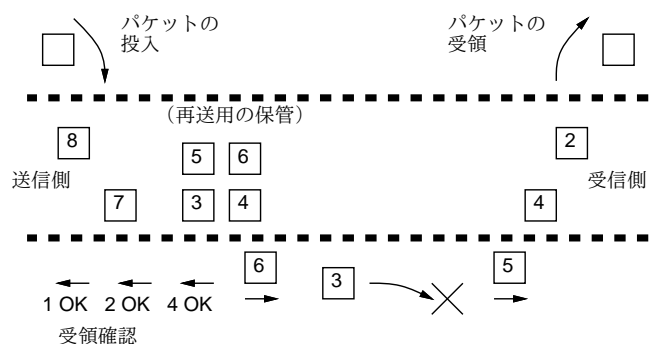


図 5: エラー制御のある伝送

このようにすると、下の階層でエラー (パケットの喪失等) があっても、上の階層にはエラーのないサービスが提供できます。ただし、下の階層でのエラーに対処するためには「再送」「確認待ち」などが必要であり、エラーが多くなればなるほど伝送は (上の階層から見て) 「遅く」なるように見えます。

演習 3 3~4人の1グループで、うち2人が送信側と受信側になって背中合わせに座り、英字10文字程度のメッセージを送るゲームをします。残った人はネットワーク役としてパケット (メモ紙) を送信側から受信側またはその逆に運びますが、時々 (2回に1回くらい?)、a~cにある「普通でないこと」を起こします。1パケットには「4文字」までしか書けず、紙の手渡し以外の動作 (合図等) は禁止です。a~cそれぞれについて、正しく送るための (送信側と受信側の間での) 約束を工夫しなさい。⁶

- a. 送ったパケットは必ず10秒以内に到着するが、時々「×」印だけのパケットに入れ替わっていることがある。
- b. 送ったパケットは到着することもしないこともある。到着する場合は10秒以内に到着する。
- c. 送ったパケットは到着することもしないこともあり、到着する場合も遅延することがある (後から送ったパケットに追い越されたりすることもある)。

⁶ヒント: 「正しく受信した」ことを示す確認パケット (ACK) を返送することはどの場合も必要になります。ACKが×になったり喪失・遅延する可能性もあることに注意。

4 インターネットのプロトコル

4.1 TCP/IP のプロトコル群 exam

ここからは、インターネットで使われているプロトコル群である TCP/IP を題材として見ていきます。TCP/IP にはこれまで長く使われてきた **IPv4** と、近年普及してきている IPv6 とがありますが、以下では IPv4 について説明します。ここで述べる範囲では、アドレス表記やプロトコルやプログラムの名前が違う程度でして、両者に原理的な違いはありません。

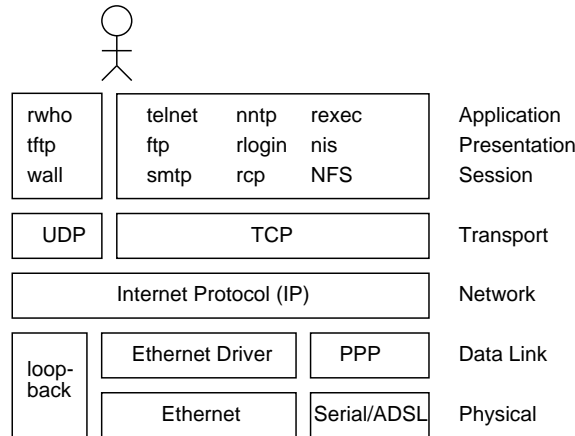


図 6: TCP/IP のプロトコル構成

TCP/IP の階層構造とそこに含まれる主要なプロトコルを図 6 に示します。中間の「ネットワーク層」として書かれている **IP**(Internet Protocol) はすべてに共通していて、その上や下では層ごとに複数のプロトコルがあります。

下側では、どのような媒体（無線 LAN とか光ファイバーとか）であるかで異なるプロトコルになります。IP より上側のプロトコルは「トランスポート層」のところで **UDP**(User Datagram Protocol) と **TCP**(Transmission Control Protocol) に分かれます。UDP は単独のパケットを送受するもので、少量のデータを迅速にやりとりしたい場合に向いています。一方 TCP は前の節で説明したような考え方でエラーの無い通信を実現しています。それより上では、電子メールとか、WWW とか、サービスによってプロトコルが違ってきます。

4.2 IP アドレス exam

先にパケットの説明で、各パケットに「宛先」が書いてあると述べました。TCP/IP ではこの宛先として指定するものを **IP アドレス** と呼びます。パケットを届ける先は個々のマシン（ホスト）ですから、IP アドレスはホストごとに別の値が割り当てられています。

TCP/IP が今の形になった 1980 年ごろには、ネットワークの規模はごく小さく、通信は遅く、コンピュータも非力でした。パケット方式ではすべてのパケットに宛先を入れるため、その形がコンパクトであることが大切でした。そこで IPv4 では、アドレスとして「32 ビットの値」を使うことにしました。我々が普段目にする「210.154.96.162」のような書き方は、32 ビットを 8 ビットずつに区切って、各バイトの値を 0~255 の範囲の数値として表記したものです (図 7 上)。

32 ビットの IP アドレスはネットワーク部 (network part) とホスト部 (host part) に分かれています。これはどういうことかということ、1 つの LAN では、そこに接続されている複数のホストや機器にそれぞれ違うホスト部を割り当て、ネットワーク部はすべて共通にします。これによって、外部からはネットワーク部だけ見てパケットを送れば済むようになるのです。

ネットワーク部とホスト部の境界はネットワークごとに違うので、必要のつど指定する必要があります。その指定は、ネットワーク部が全部「1」、ホスト部が全部「0」のビットになった 32 ビットの

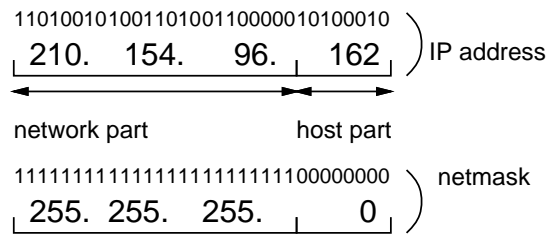


図 7: IP アドレスとネットマスク

値を使うのが普通です。これをネットマスク (netmask) ないしサブネットマスク (subnet mask) と言います。ネットマスクも通常、8 ビットずつに区切って各バイトを 0~255 の 10 進数として表現します (図 7 下)。⁷

IP アドレスは、世界中 (インターネット中) で重複がないように管理されていて、ある特定の IP アドレスを持つホストは世界中でただ 1 つしか存在しないようになっています (実際にはネットワーク番号を重複しないように各組織に割り当て、ホスト番号はその組織のネットワーク管理者が割り当てます)。これによって、世界中のどこからでも「あのコンピュータ」という指定をすればそのコンピュータに向かってパケットが転送されて来るようになるわけです。⁸

4.3 インターネットの構造 exam

IP アドレスが分かったところで、次はインターネットの全体像を見てみましょう。皆様は自分の使うマシンからインターネットまで、どうつながっているのか考えたことがありますか？

図 8 に簡単な概念図があります。左上の方にある「C」と書かれた箱のあたりが、自分のコンピュータだと思ってください。多数のホストを持っているところでは、それらのマシンはスイッチ (switch) ないしハブ (hub) という装置によって相互接続されていて、1 つの LAN を構成しています。個人の家のように、マシンが少数しかない場合でも、今日では無線 LAN を使うのでやはり LAN になっていることが多いでしょう。

LAN のような 1 つのネットワークと別のネットワーク (や外部の回線) とがつながるところでは、ハブやスイッチではなく、次節で述べる経路制御を行う中継装置が必要です。TCP/IP ではこのような装置のことをルータ (router) と呼びます。学校や家庭などの LAN が直接接続する相手先は ISP (Internet Service Provider) ないし「プロバイダ」と呼ばれる事業者のネットワークです。ISP は契約者から料金を取ってインターネットへの接続を提供するのが商売であり、そのために大規模な WAN を自社の設備によって (または他社の設備を論理的に借りるなどして) 構築しています。

しかしこれだけでは、私がプロバイダ A と契約していて、知合いがプロバイダ B と契約しているときに、相互に通信できないことになります。そこで、プロバイダはそれぞれの WAN を相互に接続しており、そこを通じてパケットを相互に流通させます。⁹A と B が直接繋がっていなくても、間接的につながってさえいれば互いに通信ができるのです。このようにして、多数のネットワークが相互につながってできた集合体がインターネットです。そういう意味で、インターネットは「草の根」的な構造をしており、どこかに「本部」があるわけではないのです。

4.4 IP と経路制御

先に説明したように、TCP/IP の中では IP というプロトコルがインターネットの中核となっています。IP の最大の「魔法」は、行き先を指定するだけで、それが世界中どこであつても、パケットを

⁷255.255.255.0 ではネットワーク部 24 ビット、(全体が 32 ビットなので残りの) ホスト部 8 がビットとなります。ネットワーク部が 29 ビットであれば、ネットマスクは 255.255.255.248 になります (16 進数に直してみてください)。

⁸このほか、ある LAN の中だけでパケットをやりとりする場合には自由につかっている「プライベートアドレス」も用意されています。プライベートアドレスを含んだパケットはその LAN から外に出さないようにしなければなりません。

⁹これを効率よく行うため、多数の ISP が相互に接続を取るための地点を設けることもあります。これを IX (Internet eXchange) と呼びます。

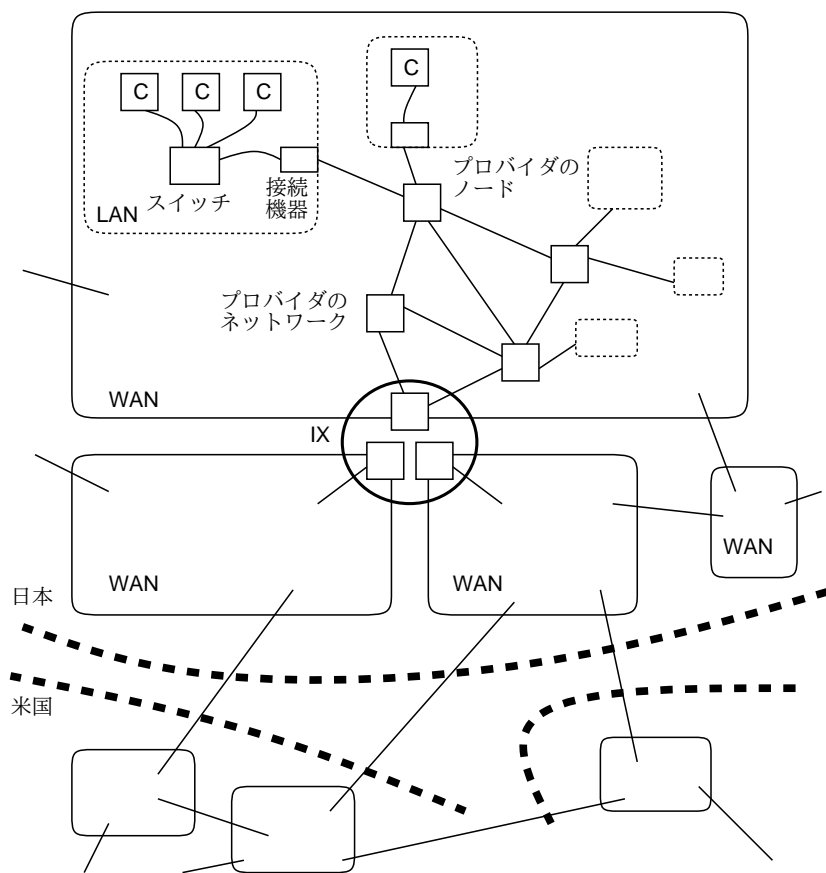


図 8: インターネットの構造

その行き先に届けてくれることです。この機能を経路制御 (routing) と呼びます。

IP の経路制御がなぜそんなに偉いのでしょうか？ 郵便物の場合を考えて見ましょう。ある郵便局に集まってきた葉書に「東京都調布市調布ヶ丘 1-5-1」という宛先が書いてあったとすると、そこが東京都以外の郵便局であれば、東京に「調布市」という市があるかどうか知らなかったとしても、とにかく東京中央郵便に送れば済みます。東京中央郵便局では、東京都のどの区や市はどの局の受け持ちか知っていますから、「調布ヶ丘」という地名を知っていてもいなくてもとにかく調布郵便局に送れば済みます。これが可能なのは、住所が「都道府県→区市町村→地名→番地」という階層構造になっているからです。アドレスが階層構造になっていれば、各中継地点では「自分の受け持ち範囲でないアドレスはとにかく上位の中継地点に送る」「受け持ち範囲のアドレスはより小さい受け持ち範囲の中継地点か個々の宛先に送る」という方法で経路制御が行えます。

しかし IP ではアドレスは 32 ビットの数値であり、上で述べたような階層構造にはなっていません。前述のように、32 ビット中で上位何ビットかがネットワークアドレス、残りがホストアドレスであり、LAN 内のホストはネットワークアドレスが共通ということは決っていますが、ネットワークアドレスのつけ方は任意です。¹⁰

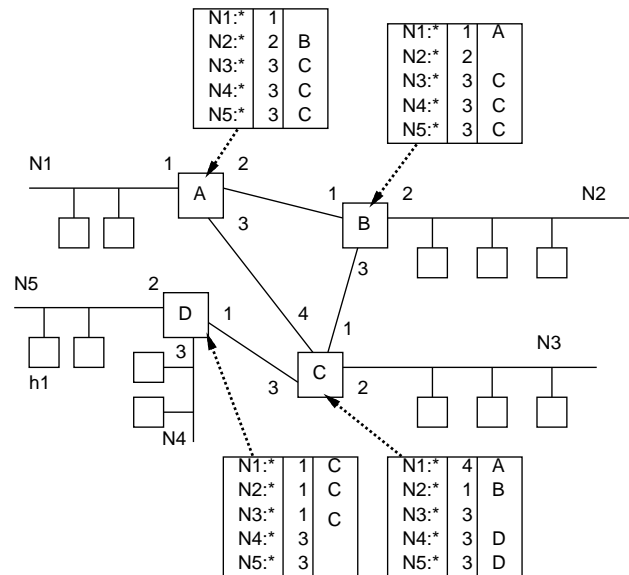


図 9: 経路表

つまり「深大寺元町」「布田」などの名前だけが与えられ、それを見て送り先を決める必要があります(布田 N 丁目、というのが近くにまとまっていることは保証されますが)。つまり、すべての主要な郵便局には全国のあらゆる地名の一覧表があり、その一覧表に「この地名はこちらの方に送る」と書かれていて、それに従って中継するわけです。このデータのことを経路表 (routing table) と呼びます。

たとえば図 9 のように 4 つのルータでネットワーク N1~N5 がつながっているとします。それぞれのルータは、自分が直接つながっているネットワークについてはそこにつながるインタフェースにパケットを送ればいいと分かりますが、それ以外については経路表を参照してパケットの転送先を決めます。たとえば、ルータ A に「N5:h1」あてのパケットが来たとすると、表を参照して「インタフェース 3 番を通じて C に送ればいい」と分かります。C にそのパケットが来ると、C も同様にして「インタフェース 3 番を通じて D に送る」と分かり、D に到達するとそこから直接 N5 に送るわけです。このような経路表は、インターネット上で経路情報を交換する複数のプロトコルを用いて、常に最新の状態に保たれています。

¹⁰ 厳密にはそれだと経路表が大きくなって大変なので、できるだけ隣接したネットワークには隣接した番号をつけ、外部からはあたかも 1 つのネットワークに見えるようにします。これを経路の集約 (aggregation) と呼びます。

先の ping では RTT しか分かりませんが、tracert というコマンドを使うと相手先までの経路を (できる範囲で) 調べることができます。ブラジルのホテルでやってみましょう。¹¹

```
[... ]$ traceroute royaltyhotel.com.br
traceroute to royaltyhotel.com.br (200.219.245.77), 30 hops max, 60 byte packets
 1 * * *
 2 tokyo2-RM-XE-4-0-10-4000.s5.sinet.ad.jp (150.99.198.253) 1.800 ms 1.502 ms 2.046 ms
 3 osaka-GM-ET-7-1-0-1115.s5.sinet.ad.jp (150.99.90.31) 8.872 ms
osaka-GM-ET-5-1-0-115.s5.sinet.ad.jp (150.99.71.233) 8.863 ms 9.124 ms
 4 ae-14.r01.osakjp02.jp.bb.gin.ntt.net (61.200.91.153) 9.213 ms 9.476 ms 9.773 ms
 5 ae-2.r22.osakjp02.jp.bb.gin.ntt.net (129.250.2.254) 9.803 ms 10.191 ms 9.312 ms
 6 ae-8.r20.sngpsi05.sg.bb.gin.ntt.net (129.250.3.110) 84.209 ms 82.437 ms 83.505 ms
 7 ae-2.r00.sngpsi02.sg.bb.gin.ntt.net (129.250.3.147) 82.071 ms 84.627 ms 83.823 ms
 8 ae0.singapore2.sin.seabone.net (93.186.133.12) 83.170 ms 81.711 ms 84.226 ms
 9 et-4-1-0.sanpaolo8.spa.seabone.net (195.22.219.75) 360.393 ms
ae7.sanpaolo8.spa.seabone.net (195.22.219.17) 353.827 ms
ae6.sanpaolo8.spa.seabone.net (195.22.219.3) 356.827 ms
10 et-4-1-0.sanpaolo8.spa.seabone.net (195.22.219.75) 370.325 ms
et-10-1-0.sanpaolo8.spa.seabone.net (195.22.219.145) 365.513 ms 357.196 ms
11 149.3.181.18 (149.3.181.18) 355.220 ms 350.302 ms 348.872 ms
14 * * *
15 186-230-241-10.ded.intelignet.com.br (186.230.241.10) 374.156 ms 362.217 ms 367.592 m
16 static.200.143.173.74.datacenter1.com.br (200.143.173.74) 360.658 ms
static.200.143.173.78.datacenter1.com.br (200.143.173.78) 364.463 ms
static.200.143.173.38.datacenter1.com.br (200.143.173.38) 367.192 ms
(途中略)
30 * * *
[... ]$
```

このコマンドは指定した宛先までの通信が通っている中継点とそこまでの往復時間を順次リストアップします (場所によっては中継点が複数マシンから構成されていてタイミングでどのマシンかが代わるものがあり、このような時は中継点が複数表示されます。また、セキュリティのため情報を返さない中継点もあり、その場合は「* * *」のような表示になります)。これを見ると、おおよそ次のようなことがわかります。

- 電通大のホスト SINET という会社 (ISP です) のネットに入り、そこの内部で数段の中継があり、なぜか大阪に行く。大阪までは 10 ミリ秒程度。
- SINET から NTT に渡され、NTT の所有している線でなぜかシンガポールに渡る (時間が 80 ミリ秒くらいに)。
- 次はまた海を渡ってブラジルのサンパウロに行き、そこのデータセンターに渡る。ここで 360 ミリ秒程度。そこからは残念ながら表示されない。

どうも、先にやった 140 ミリ秒と比べて遅いので、経路が違っているようですが、原因は不明です。

演習 4 演習 2 でパケット到達時間を調べたホストまでの経路を traceroute で確認してみなさい。それができたら、次の事項から 1 つ以上やってみなさい。

注意! 多人数で 1 つのマシンからあちこちに traceroute すると攻撃と勘違いされるので、sol ではやらず、個人 PC かスマホ (ping や traceroute ができるアプリがあります) でやること。なお、Windows では traceroute コマンドは tracert という名前になっています。

¹¹Windows ではコマンド名は tracert になっています)。

- a. 日本国内のさまざまな箇所までの経路を調べ、自分のいるところから国内がどのようにつながっているかを整理してまとめなさい。
- b. 世界の複数の地域までの経路を調べ、日本から世界各地にどのようにつながっているかを整理してまとめなさい。
- c. 大学から各地までの経路と、自宅 (ないし職場など) から各地までの経路を比較し、どこまでが違ってどこからは共通かなどを整理しなさい。

4.5 ドメイン名と DNS

ここまで見て来たように、IP での通信はあくまでも IP アドレスを用いて行われますが、人間が見て理解するにはもっと「普通の」(意味のある) 名前を使うことが望まれます。このため、インターネット上ではドメイン名 (domain name) と呼ばれる文字列の名前サポートし、ドメイン名から IP アドレスを検索するためのシステムとして **DNS**(Domain Name System) と呼ばれるサービスが運用されています。ドメイン名は複数の名前を「.」でつなげたもので、右側の名前がより広い範囲に対応する階層構造になります (日本での住所表記とは反対ですが、英語では住所、街、州、国の順で書くのでそれと同様です)。たとえばドメイン名「sol.cc.uec.ac.jp」は次のような階層に対応しています。

```
sol.cc.uec.ac.jp
      ↑日本
      ↑教育組織
      ↑電気通信大学
      ↑計算センター
      ↑ホスト名
```

DNS は内部的にもこの階層構造を利用して「最右側の名前 (**TLD**、Top-Level Domain) の管理元をすべて知っているサーバ (ルートサーバ)」、「各 TLD の管理をするサーバ (その子供のドメインの管理元をすべて知っている)」、「第3レベルのサーバ」等々のように分担してドメイン名と IP アドレスの対応を把握し、検索に応答できる状態を維持しています。

DNS に対する検索は通常、ユーザが (メールや Web などのアドレスの形で) ドメイン名を指定した時に自動的に行われますが、コマンド **nslookup** を使って次のようにユーザが直接検索することもできます。

```
[... ]$ nslookup www.yahoo.co.jp
Server:          130.153.26.5
Address:         130.153.26.5#53
Non-authoritative answer:
www.yahoo.co.jp canonical name = www.g.yahoo.co.jp.
Name: www.g.yahoo.co.jp
Address: 183.79.123.210
Name: www.g.yahoo.co.jp
Address: 183.79.196.239
Name: www.g.yahoo.co.jp
Address: 183.79.135.206
Name: www.g.yahoo.co.jp
Address: 124.83.155.249
[... ]$
```

上の例では IP アドレスが4つ返されていますが、これは大量のアクセスをさばく必要があるサーバでは複数マシンにアクセスを分散させるため、IP アドレスも複数になるのです。

本日の課題 **2A**

本日の課題は「演習 2」「演習 3」「演習 4」に含まれる小問 (合計で 9 個) の中から 1 つ以上を選択し、結果をレポートとして報告して頂くことです。LMS の「レポート # 2」の入力欄に直接入力してください (別途作成したものをコピーペーストで貼っても構いません)。以下の内容がこの順に含まれるようにしてください。

- 冒頭に題名「コンピュータリテラシレポート # 2」、学籍番号、氏名、提出日付を書く。(グループでやったものはグループのメンバー全員の氏名も別途書く。)
- 課題の再掲を書く (どんな課題であるかをレポートを読む人が分かる程度に要約する)。
- レポート本体の内容 (やったこととその結果) を書く。
- 考察 (課題をやった結果自分が新たに分かったことや考えたこと) を書く。
- 以下のアンケートに対する回答。

Q1. コンピュータネットワークの仕組みについてどれくらい知っていましたか。新たに知ったことで面白かったことは何ですか。

Q2. Unix システムを使って操作しながら調べたりするのは慣れましたか。

Q3. リフレクション (今回の課題で分かったこと) ・感想 ・要望をどうぞ。

なお、グループ課題以外の課題もグループでやって構いません。いずれの場合も、(メンバー全員の氏名を明記した上で) レポートは必ず各自で執筆してください。レポート文面が同一 (コピー) と認められた場合は同一であると認めた全員について点数にペナルティを科すことがあります。