

ネットワーク構成運用論

2004/05/31

芸術情報設計学科

藤村 直美

過去のネットワーク

- ◆ メーカー毎に独自のネットワークを構築
 - SNA (Systems Network Architecture)
 - DECNET (Digital Equipment Corporation)
 - N1
- ◆ ARPANETの発展 インターネット
- ◆ 中央集権的システム 分散処理システム
- ◆ 欧州では異なる製造業者の機器を接続
OSI参照モデルとして規格化

プロトコル

- ◆ コンピュータ同士が円滑な通信を行うための通信規約、約束事、ルール
- ◆ IP、TCP、UDP、SMTP、HTTPなど
- ◆ 物理的なケーブルやコネクタの形状、電気信号の規格なども含む

OSI 7 階層モデル

- ◆ 7層からなるインターフェイス構造
- ◆ 7つの層(レイヤー)を定義
 - アプリケーション層
 - プレゼンテーション層
 - セッション層
 - トランスポート層
 - ネットワーク層
 - データリンク層
 - 物理層
- ◆ それぞれの層では同じ水準の手順で通信

各層の意味

- ◆ アプリケーション層 = 相手と何について話すか
- ◆ プレゼンテーション層 = 相手とどんな言葉で話すか
- ◆ セッション層 = 相手は誰か
- ◆ トランスポート層 = 相手は何処にいるか
- ◆ ネットワーク層 = どのような経路で相手のいる場所にたどり着けるか
- ◆ データリンク層 = どんな道筋でいけばよいか
- ◆ 物理層 = 道筋の交通手段

階層化の意味、効果

- ◆ 階層ごとに独立している
- ◆ 他の階層の影響を受けない
- ◆ 開発時に他の階層のプロトコルを使用可能
- ◆ ある階層の仕様が変わっても影響されない

OSI 7階層モデル概念図



<http://www.atmarkit.co.jp/icd/root/05/5798905.html>

物理層(レイヤ1)

- ◆ 中継点に到達するまでの物理的な接続
 - 具体的には、ケーブルの材質やコネクタ形状、およびデータと電気信号の相互変換方式(電圧などの規定)など
 - RS-232Cなどのインターフェース規格や、リピータハブやケーブル類といった製品が物理層に相当
- ◆ 物理的な伝送媒体の電気、機械的な制御
 - 物理層ではデータの内容については一切関知しない
- ◆ ビット伝送サービス

データリンク層(レイヤ2)

- ◆ ネットワーク上で直結されている機器同士での通信方式を定めたもの
- ◆ 隣接ノード間でのフレームなどのビット列単位の伝送に関する制御
- ◆ 同期のための制御
- ◆ 誤り制御、フロー制御
- ◆ PPPなどの規格がデータリンク層に相当しており、スイッチングハブなどの製品がデータリンク層をカバーしている

ネットワーク層(レイヤ3)

- ◆ 通信経路を決める
- ◆ 発信者から受信者までの中継点およびネットワークの資源管理
- ◆ 上位層からのPDU (Protocol Data Unit)の分割と再組み立て
- ◆ PDUとはプロトコルが扱うデータの単位
 - TCP/IPのPDUは「パケット」、
 - EthernetのPDUは「フレーム」、
 - ATMのPDUは「セル」である。
- ◆ 輻輳制御

トランスポート層(レイヤ4)

- ◆ 通信すべき相手側とのデータの運送に関する業務
- ◆ データ転送の信頼性を確保するための方式を定めたもの
- ◆ 誤り制御、フロー制御
- ◆ ネットワーク層を通して送られてきたデータの整序や誤り訂正、および再送要求など
- ◆ TCP、UDPなどがトランスポート層に属する

セッション層(レイヤ5)

- ◆ 通信の開始時や終了時などに送受信するデータの形式などを規定したもの
- ◆ セッションと呼ばれる一連の会話のための接続手続きと管理
- ◆ 計算機へのログイン、ログアウトなどの制御
- ◆ この層で論理的な通信路が確立される

プレゼンテーション層(レイヤ6)

- ◆ アプリケーションなどでやり取りされる情報の構造を共通にする
- ◆ 圧縮方式や文字コードなど、データの表現形式を規定
- ◆ 個別のバイナリファイルをネットワークで通信できる形式に変換したり、逆にネットワーク経由で受信したデータをアプリケーションソフトが認識できる形式に復元したりする部分
- ◆ セッション層からアプリケーション層までの通信方式は単一のプロトコル(例えばHTTP)で定められていることが多い。

アプリケーション層(レイヤ7)

- ◆ プロセス間でやりとりする情報の表現形式について設定
 - データ構造
 - 符号化の方法
 - 文字の表現形式
- ◆ 電子メール、電子ニュース、WWW
- ◆ Mbone

各層のモデル

- ◆ 送受信者間でやりとりされるのはPDU (Protocol Data Unit)
- ◆ 各層のPDUは下位層のPDUのデータ部となる
- ◆ プロトコル制御情報 (PCI : Protocol Control Information)
- ◆ 下位層インターフェイスへの制御情報 (ICI : Interface Control Information)
IDU (Interface Data Unit)

TCP/IPにおける階層

- ◆ アプリケーション層 (HTTP, SMTP,...)
 - アプリケーション層、プレゼンテーション層、セッション層
- ◆ トランスポート層 (TCP, UDP)
- ◆ ネットワーク層 (IP, RIP, BGP4)
- ◆ リンク層 (PPP, Ethernet, ATM, ISDN,...)
 - データリンク層、物理層

WWWの通信例

- | | |
|----------------------------|--------------------|
| ◆ クライアント側 | ◆ サーバ側 |
| ■ URLからHTTP | ■ URLを復元 |
| ■ TCPヘッダーを付加 (TCPセグメント) | ■ TCPヘッダーを取り除く |
| ■ IPヘッダーを付加 (IPダイアグラム) | ■ IPヘッダーを取り除く |
| ■ イーサネットのヘッダーを付加 (MACフレーム) | ■ イーサネットのヘッダーを取り除く |
- ➡
ルータなどで中継

中継システムのモデル化

- ◆ 終端システム(7階層のOSIインターフェイスを持つシステム)
- ◆ 中継システム
 - リピータ 物理層の中継
 - ブリッジ データリンク層の中継
 - ルータ ネットワーク層の中継

中継システム(1)

- ◆ハブとスイッチ
 - ネットワークに機器を接続
 - ハブは関係ないデータも流す
 - スイッチはつながっていない機器のデータは流さない
- ◆リピータ
 - リピータは同一ネットワークを単純に接続
 - 電氣的な延長

中継システム(2)

- ◆ブリッジ
 - ブリッジは同じ種類のネットワークを接続
 - MACアドレスで中継
 - 無駄なトラフィックを中継しない(フィルタリング)
- ◆ルータ
 - ルータは異なる種類のネットワークも接続
 - 各種のインターフェイス(イーサ、FDDI、専用線、ATM等)
 - 経路制御の機能が必要

インターネット

- ◆ネットワークのネットワーク
- ◆異なる技術で構築されたネットワーク同士を相互接続
- ◆1970年代頃から盛んに研究
- ◆パケット形式の変換、宛先の識別

インターネットの相互接続法

- ◆プロトコル変換システム
- ◆ゲートウェイ(ルータ)をおき、それぞれのネットワーク層のプロトコルの上にインターネットプロトコル(IP)を使用
- ◆IPパケットを自分のネットワーク層で送る
- ◆セッション層とプレゼンテーション層はない

インターネットの構成要素

- ◆ホスト
 - ネットワークの終端システム
 - 情報の発信と受信
 - 自分宛でないパケットを廃棄
- ◆ルータ
 - IPパケットの蓄積交換作業
 - IPパケットを転送
 - 経路情報を持っている

IPヘッダー (IPv4)

バージョン	ヘッダー長	サービスタイプ	全長	
ID		フラグ	フラグメントオフセット	
TTL	プロトコルタイプ		ヘッダーチェックサム	
送信元アドレス				
宛先アドレス				
オプション(可変)			パディング(可変)	
データ				

IPヘッダーの情報

- ◆ IPの版(V4またはV6)
- ◆ ヘッダー部の長さ
- ◆ サービス品質
 - 優先度
 - 遅延指定(普通、遅い)
 - 速度指定(普通、速い)
 - 信頼性(普通、高信頼度)

パケット分割

- ◆ ネットワークの最大パケット長(MTU :Maximum Transmission Unit)
- ◆ 転送可能な最大パケットサイズ(MTU)はネットワークごとに異なる
- ◆ インターネットでは情報フィールドは最大1500バイト(最小は46バイト)
- ◆ 長すぎる時は分割と組み立て
- ◆ データリンク層で行うが、IP層にも機能がある

パケット通信の特徴

- ◆ 大きなデータを一塊で送信すると、
 - 他の通信をふさぐ
 - エラーが起きたときに全部再送しないとけない
 - 受け取り側の受信能力を超えることがある
- ◆ 小さなパケットに分解して通信すると、
 - 複数のホストが同じ通信線を共有可能
 - エラーに対して小さなパケットの復旧だけで済む
 - 少しずつ受信できる

生存時間(TTL)

- ◆ 宛先を見失うといつまでもさまよう可能性
- ◆ 経由するルータの数を設定
- ◆ ルータを経由するたびに1を引き、ゼロになったらパケットを破棄
- ◆ 破棄と同時にICMPで送信元に通知

ICMP

- ◆ Internet Control Message Protocol
- ◆ 用途
 - 宛先への到達性
 - ルータのバッファの空き状況
 - もっと良い経路がある
- ◆ ネットワーク管理に必須のプロトコル
 - 現在はネットワーク管理にはSNMPが使われている

ICMPの種類

- ◆ こだまメッセージと返事
- ◆ 宛先に到達不可能
- ◆ 時間切れ
- ◆ パラメータ問題
- ◆ パケット消失
- ◆ 再転送
- ◆ 時間のスタンプ要求/回答(ping)
- ◆ 情報要求/回答
- ◆ アドレスマスク要求/回答

pingとtraceroute

- ◆pingコマンドで到達性を確認
- ◆tracerouteで経由するルートを確認

pingの例

```
C:¥Documents and Settings¥fujimura>ping cs.kyushu-id.ac.jp
Pinging cs.kyushu-id.ac.jp [202.26.211.5] with 32 bytes of data:

Reply from 202.26.211.5: bytes=32 time=61ms TTL=50
Reply from 202.26.211.5: bytes=32 time=59ms TTL=50
Reply from 202.26.211.5: bytes=32 time=58ms TTL=50
Reply from 202.26.211.5: bytes=32 time=57ms TTL=50

Ping statistics for 202.26.211.5:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 57ms, Maximum = 61ms, Average = 58ms

C:¥Documents and Settings¥fujimura>
```

tracerouteの例

```
[fujimura@cosmos fujimura]$ /usr/sbin/traceroute ftp.kyushu-u.ac.jp
traceroute to ftp.kyushu-u.ac.jp (133.5.1.2), 30 hops max, 38 byte
packets
 1 202.26.222.1 (202.26.222.1) 0.314 ms 0.253 ms 0.237 ms
 2 202.26.211.65 (202.26.211.65) 0.398 ms 0.333 ms 0.325 ms
 3 kyushu-S1-GE6-2.sinet.ad.jp (150.99.208.62) 0.802 ms 0.705
ms 0.608 ms
 4 kyushu-u-LAN2.sinet.ad.jp (150.99.196.82) 0.635 ms 0.591 ms
0.578 ms
 5 133.5.3.1 (133.5.3.1) 1.736 ms 1.683 ms 1.662 ms
 6 fs1.nc.kyushu-u.ac.jp (133.5.1.2) 0.850 ms 0.695 ms 0.756 ms
[fujimura@cosmos fujimura]$
```

ヘッダーのエラーチェック

- ◆ヘッダーチェックサムで、IPヘッダーが壊れていないかチェック
- ◆IPヘッダーの内容のみ
- ◆データ部についてはTCPやUDPで保証

パケット転送方法

- ◆サブネットの意味(物理的、論理的ネットワークのまとめ)
- ◆通信時の手順
 - ドメイン名からネームサーバ(DNS :Domain Name System)に照会してIPアドレスを入手
 - ホストアドレスから同一ネットワークならアドレス変換テーブル参照して送信
 - 同一ネットワークでない場合にはIPアドレスから経路制御表を参照してルータ(GW)に送出

ARP

- ◆Address Resolution Protocol
- ◆MITの人工知能研究所で開発
- ◆C L型の問い合わせと回答で情報収集
- ◆仕組み
 - ホストAがホストBのMACアドレスを知りたい
 - ホストBのIPアドレスを含めたARPパケットを放送
 - ホストBは自分のMACアドレスを答える

arpの例

```
[fujimura@cosmos fujimura]$ /sbin/arp -a
? (202.26.222.1) at 00:D0:04:E4:40:0A [ether] on eth0
cp840.aid.kyushu-id.ac.jp (202.26.222.74) at 08:00:11:0A:55:3D [ether] on eth0
common.kyushu-id.ac.jp (202.26.222.75) at 00:D0:B7:21:C4:B9 [ether] on eth0
jearn.kyushu-id.ac.jp (202.26.222.82) at 00:D0:09:32:1F:52 [ether] on eth0
[fujimura@cosmos fujimura]$
```