ネットワーク
システム管理 #05

たかさきこうや

1限 (09:00-10:45)

1

課題(not お題)

- IPアドレス計算機のExcel
- 出せました？
- 以下提出が確認された人リスト(到着順)
- c140093 c240089 c240050 c240015 c240046 c240133 c040062 c240048 c240038 c140090 c140128 c240071 c240067 c240070 c240047 c240127 c240085 c240100 c140123 c240002

2

先週のおさらい

- コンピュータ同士は、自分と相手を識別するために IPアドレスを使って通信をする
- IPアドレスを覚えるのはしんどいので、名前(ドメイン名/ホスト名)を使う
- 名前とIPアドレスを紐づけるためにDNSという仕組みを使う

3

www.cuc.ac.jp

- 本名：bay.cc.cuc.ac.jp
- IPアドレス：202.244.38.102
- このコンピュータは一体「なんだろうか？」

4

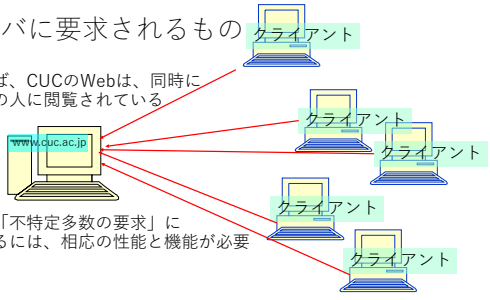
サーバ

- サーバ、と呼ばれるコンピュータ
  - サーバ(Server)＝サービスをするもの
    - Ex: ビアサーバ＝ビールをサービスするもの
- サービスを受ける側は、クライアントと言う
  - クライアント＝顧客

5

サーバに要求されるもの

- 例えば、CUCのWebは、同時に多くの人に閲覧されている



- この「不特定多数の要求」に応えるには、相応の性能と機能が必要

6

サーバの性能はピンキリだ

- 例えば、みんなが教室や自宅で使っている PCをサーバとして使うことも可能
- ただし、サーバはクライアントPCとは異なる点がある
  - ハードウェアとしての違い
  - OSとしての違い

7

ハードウェアとして

- CPUが「並列動作を得意としていること」
  - あるいはCPUがそもそも複数あること
- メモリが潤沢にあること
  - データ破損の訂正機能が付いていること (ECC=Error-correcting code)
  - CPUとの同期機能がついていること(Registered Buffer)
- HDDが高信頼性であること
  - サーバ用であること(SAS=Serial Attached SCSI)
- でも、個々の装置の信頼性を高めることには限界がある

8

冗長(Redundancy)

- 何らかの異常に備え、機器の一部機能 もしくは機器そのものを複数用意すること
  - 「話が冗長だ」という言い回しもあるのでネガティブな印象があるかもしれないけれど、機械設計では必須の概念
- 冗長化することで、可用性を高めた構成を High Availability(HA)構成　などと呼ぶ

9

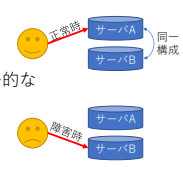
冗長化の例

- 電源冗長化
- ネットワーク冗長化
- ハードディスク冗長化
- サーバ冗長化
- スイッチ冗長化
- 経路冗長化

10

アクティブ-スタンバイ

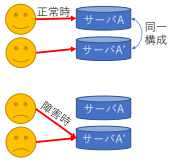
- ある機能を複数用意し、普段使いしているAが故障したら予備Bを稼働させる
  - 普段はJRで学校に通っているけど、人身事故が起きたら京成に切り替える…的な
- Aに常に最新情報が蓄積されている場合、Bに変えた時に情報の蓄積が無くなってしまうので何らかの工夫が必要
- 電源やネットワーク経路なら使いやすい



11

負荷分散

- 最初から同一機能のAとA'の2台用意し、両方に仕事をさせるが、どちらかが故障したら修理するまで縮退運転させる
  - 道路で片側2車線用意し、工事の時は1車線規制する…的な



12

レプリケーション

- あるサーバ上にあるデータを
  別のサーバにも同期して持たせる仕組み
- サーバそのものではなく、同一サーバ内でHDDのデータのみを
  同期する場合は「ミラーリング」と呼ばれる
  - Ex: RAID 1
-

13

サーバは

- 複数人での同時使用が前提
- クライアントは、それが壊れたからと
  言って全員の業務が止まることはない

- サーバは関係者全員の業務が派手に止まる可能性がある

14

OSとして

- OSにはライセンス条項が存在する
- ハードウェアに存在することもあるが…

- Windows10の場合、クライアントOSとして
  使用する場合に限って使用許諾が与えられている

- 「サーバとして使う性能がない」のではなく
  「サーバとして使うことが認められていない」

15

EULA

- End User License Agreement
  - 使用許諾契約
- 誰が、何台のPCで、どの程度の期間どういう契約形態で
  そのソフトウェアを使用できるか、という定義
  - 例えば、Windows 10の場合、20台までの接続しか許さない
    - https://www.microsoft.com/en-us/Useterms/OEM/Windows/10/Useterms_OEM_Windows_10_Japanese.htm
    - これでは、不特定多数のWebサービスは
      出来ないよね？

16

Windows Server

- クライアントにサービスする場合
  Windowsだと
  「Windows Server 2016」や
  「Windows Server 2019」などを選択する必要がある
  - この場合も、接続クライアント数に応じて
    CAL(Client Access License)を購入する
    必要がある
- また、世の中にはWindows以外のOSも存在する
  - Ex: UNIX

17

UNIXとは？

- WindowsのようなOS
  (オペレーティング・システム)の一種
  - パーソナルコンピュータとしては、
    WindowsやMacintoshに続く利用率
- 昨今有名な「Linux」もUNIXの一種
  - 厳密にはUNIX互換
    - iOS→FreeBSD(UNIX互換OS)を元に開発
    - Android→Linux(UNIX互換OS)を元に開発
- こんにちのネットワークサービスを支えているOSといえる

18

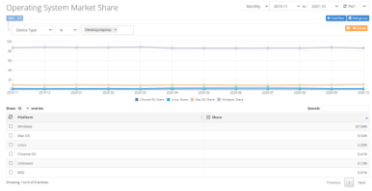
OSのシェア

- https://gs.statcounter.com/os-market-share
- 実は世界のOSシェアのトップを AndroidとWindowsが競っている
  - Androidは中身はLinuxの派生
  - iOSとmacOSは中身は FreeBSDの派生
- OSとしてはWindowsより Linuxの方が多いと言える

19

デスクトップOSシェア

- https://netmarketshare.com/
- デスクトップOSのシェアという点ではWindowsが圧倒的



20

余談

- 「ネットワークシステム管理」の世界においては 「世の中のシェア的にはどうなっている」かを知るのが 非常に重要になる
- なにかを作るなら、シェアが多いものを対象に作っていく
  - 脆弱性が狙われるとき、それはシェアが多いモノから 狙われていく
    - 脆弱性　〇ぜいじゃくせい　　×きじゃくせい

21

無料のOSもある

- Linux(CentOS、Ubuntu…)
- FreeBSD
- は、無料で配布されている(ただしサポートはない)
- OSの出来としては非常に良い
- UNIX互換
- UNIXというのは規格であり、準拠したOSは誰でも作れるが UNIXを名乗るにはパテント料を払わなければならない

22

では、そのサーバは

- どうやってサービスをするのだろうか？
- IPはあくまでもEnd to Endの通信を 確立するための手段に過ぎない
- これを行うためのプロトコルが 「TCP」や「UDP」である

23

TCP

- IP(Internet Protocol)は、通信を 相手まで届ける仕組み
- TCP(Transmission Control Protocol)は 「その相手に対して、信頼のおける通信を確立する仕組み」

- 送信側、受信側共にポート番号を用いて、 誰が誰にサービスを要求するのかを明確にする
- 通信を確立させたうえで接続状態を維持
- 届かなかった通信の再送　などを行う

24

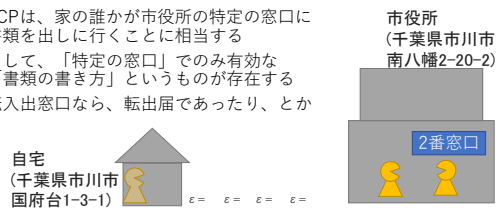
UDP

- TCPに対して「信頼性がないけど早い」プロトコル
- User Datagram Protocolの略
- TCP同様、ポート番号を用いて通信する

25

IPとTCPの関係

- 家と市役所の関係がIPの通信に相当するなら
- TCPは、家の誰かが市役所の特定の窓口に書類を出しに行くことに相当する
- そして、「特定の窓口」でのみ有効な「書類の書き方」というものが存在する
- 転入出窓口なら、転出届であったり、とか



26

この書類の定型書式が

- 更にTCPの上のプロトコルに相当する
- HTTPはそのうちの一つ
- Webコンテンツを要求に応じて表示するプロトコル
- 「クライアントはサーバにリクエスト(要求)を出し」
- 「サーバはクライアントにレスポンス(応答)を返す」

- 住民票発行届を出せば住民票を貰えるでしょ？

27

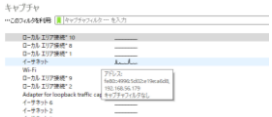
IPアドレスとインターフェース

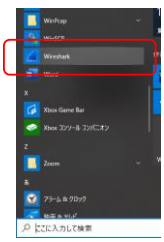
- 先週、コマンドプロンプトで ipconfigコマンドを実行した
- IPアドレスを有する インターフェースが 1個(もしくはそれ以上) あったはず
- 僕の場合は イーサネット　アダプターの 「イーサネット」に IPアドレスが付与されている

28

初回にインストールした

- wiresharkというプログラムを 起動してみよう
- インターフェースを選択するメニューが あるので、「イーサネット」を選択
  - 同じIPアドレスがついてるのがあるはず

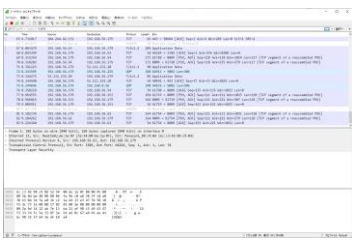




29

すると…

- なんかピロピロと 画面上に出てくる
- 放っておくとずっと 出続けるので、 左上の「■」を押して 一旦止める



30

これはなに？

- Wiresharkは「パケットキャプチャツール」

- 「パケット」＝英語で「小包」の意味
- 通信を行う際に、データは細切れにして送受信されるがその単位

31

パケットとフラグメント

- 例えば、1学年320人の学生を修学旅行に送り込みたい
- バスは1台につき40人しか乗れない
  - →じゃあ、8台のバスがあれば足りる？
- 1台のバスにつき、運転手1名とバスガイド1が必要
  - 通信にはヘッダというものが必要で、これに相当する(後述)
  - 実質38人しか乗れない
- 1クラス32人につき、引率の先生2人が必要
  - 実質引率の先生が20人なので320+20=340人だ
  - 実質38人定員のバス9台では足りないので10台用意する

32

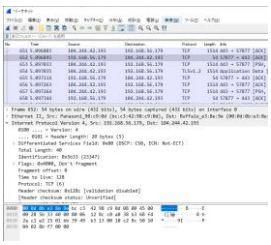
みたいなことをして

- データは、通信経路上の都合によって分割される
- これをフラグメント(断片)化と言いフラグメント化されたデータを「パケット」と言う
- インターネットの場合、大体1500byteで分割する

33

話を戻す

- SourceとDestinationという項目があるが、これが「送信元IPアドレス」と「宛先IPアドレス」の意味
- 送信元(Source)に自分のIPアドレスが入っている場合、「送っている」通信
- 宛先(Destination)に自分のIPアドレスが入っている場合、「受け取っている」通信



34

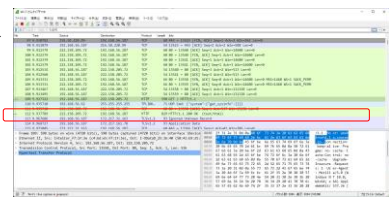
では試しに

- Wiresharkを起動した状態で
- http://abehiroshi.la.coocan.jp/
- にアクセスをしてみよう
- 阿部寛氏のWebサイトである

35

何処かに

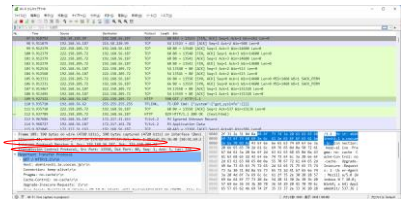
- HTTPプロトコルで、 HTTP/1.1 200 というものがあるはず
- また、どのリクエストに対するレスポンスなのかもわかる
- 実際にフラグメント通信もわかる



36

この通信は

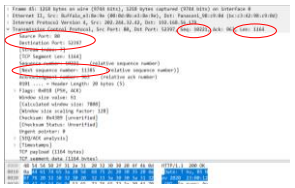
- IP(v4)通信であり、送信元IPアドレスと宛先IPアドレスがある
- どのプロトコルを使っているかも記されている



37

この通信は

- ポート番号80番からのレスポンスであり、中身が1164バイトであることも分かる
- シーケンス番号が10221であり、中身が1164(バイト)なので10221+1164=11385ということで次のデータのシーケンス番号が11385であろうことも分かる

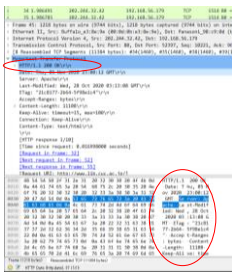


38

この通信は

- HTTPであり、バージョンが1.1であること
- ステータスコードが200であり、要求に成功したこと
- あれやこれやの実データが返ってきてること

- などが一通りわかる



39

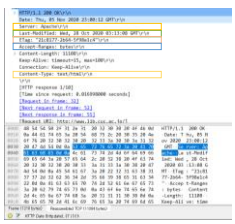
HTTP

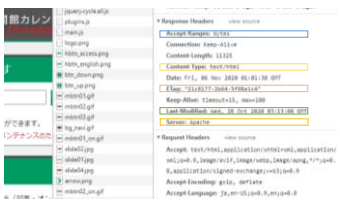
- Hyper Text Transfer Protocol
- Webコンテンツをやり取りするプロトコル
- 付帯情報をやり取りする「ヘッダ」と本文である「ボディ」を送受信する
- どんな情報をやりとりしてるかは、ブラウザの「デベロッパーツール」を開くと多少わかる

40

比較すると

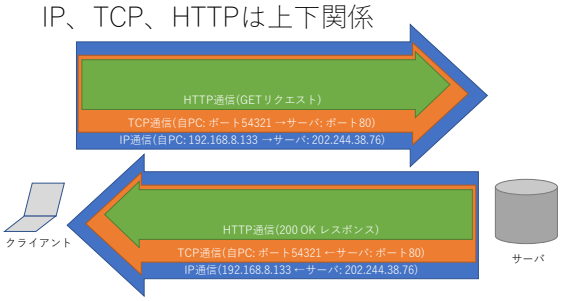
- 同じであることがわかる





41

IP、TCP、HTTPは上下関係



42

では同じことを

- https://www.cuc.ac.jp/ で
  やってみて、wiresharkと
  デベロッパーツールで比較してみよう

43

今度は

- それらしい通信が見つからない
- どうして？

44

TLS

- https:// でURLが始まる場合、
  ブラウザとサーバ間で暗号通信が
  行われており、その間はたとえ
  PCのネットワークインターフェース
  であっても盗聴することはできない
- 逆に言えば、httpは通信経路の
  何処でも盗聴可能

45

TLS

- ブラウザとWebサーバアプリケーションとの間で
  End to Endの暗号化を行う為、経路上では通信は
  盗聴できない

46

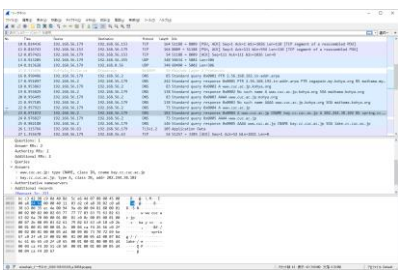
そうしたら今度は

- nslookup をした瞬間の通信を
  Wiresharkでキャプチャしてみよう

47

こちらも

- 暗号化はされて
  いないようだ
- ただ、1回
  問合せたに
  しては何度か
  やり取りが
  あるようだが…



48

これは

- www.cuc.ac.jp が実は「何かの名前の省略形」じゃないかと疑い、「自力で何かを補完した様子」やIPv4のアドレスと同時にIPv6のアドレスも名前解決しようとした痕跡と思われる

49

UNIXの使い方

- サーバとして発展してきた
  - 色々なサービスをしている
- 「ターミナル端末」というものからサーバにログインして使える
  - ちなみに、Windows側もターミナル端末になれる
  - メッセンジャーとかでサーバにログインする感じ

50

そんなことはまぁいいや

- 使ってみよう
- 依然自分のPCにインストールしたTeraTermを起動
- サーバ名(ホスト名)は login.cuc.ac.jpを指定

51

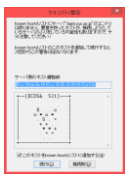
Teratermってなに？

- サーバにログインするためのターミナル端末のひとつ
- 暗号化された安全な通信が確保される
- 使用されるプロトコルはTCP
- ポート番号は22(CUCの場合)

52

ログインするために

- Teratermは、サーバ名を指定した後、ユーザ名とパスワードを入力する
  - ホストの鍵を確認されることもある

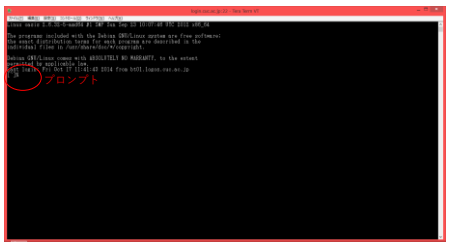


こんなのが↑
出てくることもあるが、リストに追加して続行してよい

53

ログインしたら

- こんな感じになるはず



54

この状態で

- サーバは命令待ち状態になる
  - チカチカしてるのが「プロンプト」
- 色んなアプリケーションをサーバ側で実行できる
  - ここがメッセンジャーとかFTPとかそういうのと違うところ
  - 遠隔地にいながらにして、色んなプログラムをサーバ側で実行できる
    - Windowsの「リモートデスクトップ」と同じ

55

サーバは

- プロンプトで、コマンド入力を待ち受ける
  - コマンド入力⇒コンピュータにさせたい仕事を、文字で入力する行為
  - コマンドを入力したら「改行」を押すとコマンドが実行される

56

試しに

- プロンプトのところで
- 「nslookup www.cuc.ac.jp」
  と入力してみよう

- 先日、Windowsのコマンドプロンプトでやったことと同じ結果が返ってくるはず！

  - 実は、Windowsで使われるnslookupコマンドは、UNIXのコマンドを移植したもの

57

コマンド：exit

- サーバからログアウトするコマンド
  - ログアウトする方法を理解したら
    再度ログインしてね

- なお、これをやらずにログアウトすると、
  色々と面倒なことになる（かも知れない）

- 追加履修の学生さんはまだアカウントが
  ないからログインできない(来週までに準備します)

58

軽くコマンドの練習(date)

- コマンド：date
  - 今日の日付時刻を出力するコマンド
  - date –u
  - で出てくるものはなんだろう？

59

コマンドの練習(cal)

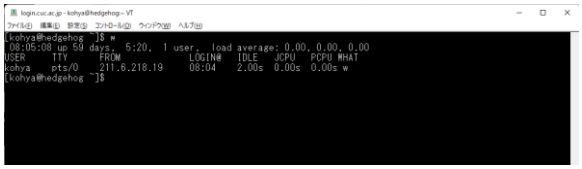
- コマンド：cal
  - カレンダーを表示するコマンド
    - cal 1 1980　　で出てくるものはなんだろう？

    - 1752年9月のカレンダーを出してみよう

60

コマンドの練習(w)

- 誰がログインしているかが分かる
- 何処からログインしているかが分かる



61

コマンド：ps (process)

- ps…サーバで動かしているプロセスを表示する
- ps -aux
- と打つと、全部の詳細なプロセスが表示される

- Windowsにおける「タスクマネージャ」に相当するが、あくまでこれは「psコマンドを実行した瞬間の全プロセス」のみ

62

プロセスとは？

- サーバの中で動いている、各々のプログラムのこと
- Windowsで言うところのタスク
- これが同時に複数動かせることが、マルチタスクコンピュータの証拠
- このコンピュータ(login.cuc.ac.jp)上の全プロセスが閲覧できる

63

コマンド：top

- サーバの中で動いている個々のプログラムが、どの程度資源を消費しているか
  - Windowsの「タスクマネージャ」に似ている
  - ※あまりみんなで同時に実行しないこと
- CPUの使用率やメモリの使用率も見られる
  - Windowsのパフォーマンスに相当
- 停止するには、「Ctrl + c」

64

top/psコマンドの意義

- ホストの状態を把握できる
  - タスクマネージャやパフォーマンスも然り
- 通常状態が把握できることで、異常な状態を把握できる
  - 健康診断だって「正常な状態」からどれだけ離れているかで異常を判断するでしょう？
- 「ネットワークやシステム」を「管理」するには「正常な状態の把握」が常日頃から必要！

65

コマンド：ifconfig

- Windowsでの「ipconfig」とほぼ同等
- サーバに付与されているIPアドレスやサブネットマスクを調べる

66

では、本日のお題

- https://forms.gle/jEnhuErMn562vyGj6

67