Key Issues

Fault-tolerance

The larger the network, the more things can go wrong.

E.g.: link/node failures, message corruption, software bugs

$\longrightarrow$ managing downtime: tier-1 providers

$\longrightarrow$ 99.999%

Two types of failures:

- independent
- correlated

In a network system with n components, assume a component fails with independent probability p

$\longrightarrow$ expected number of failures: $n \cdot p$

$\longrightarrow$ probability of no failures: $(1 - p)^n$

$\longrightarrow$ probability of k simulaneous failures: p^k

Thus correlated failures have miniscule probability.

$\longrightarrow$ exponentially small in k

In reality, failures are not independent.

$\longrightarrow$ e.g., power outage, natural disasters

We have:

$\longrightarrow$ Murphy's Law

- issue of reliable communication
- reliable network services

  $\rightarrow$ main principle: redundancy
- Examples:
  - routing of messages: alternate/back-up routes
  - domain name servers: duplication
  - transmission by space probes: forward error correction (FEC)

    $\rightarrow$ also used for multimedia traffic

Network security

Features:

- confidentiality: encryption
- integrity: message has not been tampered
- authentication: sender really is who she claims to be

$\longrightarrow$ "CIA"

$\longrightarrow$ foundation: cryptography

$\longrightarrow$ end-to-end

$\longrightarrow$ networking problem?

Modern security vulnerabilities:

- denial of service (DoS) attack

  → e.g., SYN flooding

- distributed DoS (DDoS) attack

  → e.g., commercial, personal, infrastructure

- worm attacks: e.g., CodeRed, Blaster, . . .

  → buffer overflow: mainly bugs in MS DLLs

- spam mail (security issue?)

- with fault-tolerance impacts QoS (quality of service)

  → Aug. 04: US broadband deployment exceeds dial-up

- security: trade-off with overhead

  → what is the desired operating point?

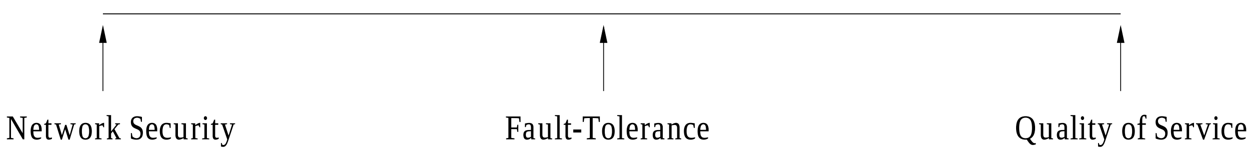
  → too much ⇒ too slow

  → too little ⇒ too vulnerable

For example: secure routing (S-BGP)

  ⟶ "BBN vs. Cisco"

Big picture:



$\longrightarrow$ points in the same spectrum

$\longrightarrow$ malicious (Byzantine) vs. non-malicious

$\longrightarrow$ availability

$\longrightarrow$ service assurances

Performance

Issues:

- excessive traffic can cause congestion (analogous to highways)
- traffic volume exhibits large fluctuations

  $\rightarrow$ burstiness
- multimedia traffic is voluminous (even for single user)
- ubiquitous access

  $\rightarrow$ wired/wireless Internet

Potential for bottleneck development

$\longrightarrow$ spontaneous or persistent

$\longrightarrow$ similar consequences as failures

Different applications require different levels of service quality.

Challenges:

$\longrightarrow$ how to provide customized QoS

$\longrightarrow$ many users and applications: scalability

$\longrightarrow$ must interoperate with legacy Internet

Current state:

- overprovisioning

  → "throw bandwidth at the problem"

  → tier-1 ISPs use sophisticated traffic engineering

- still no Internet QoS

  → changing with VoIP and content deployment

- not economic

  → few tier-1 providers make money

Data networking, telephony, and content convergence

⟶ Y2K+ trend

- VoIP (Voice-over-IP): wired world

  → traditional TDM-based telephony system is entirely separate network

  → economic factors are dictating convergence

  → from KaZaA to Skype

- cellular voice networks: 2G, 2.5G, 3G

  → what is 4G?

  → telcos/cellular providers are concerned

  → take-over by WLAN + IP?

  → strategy: active participation

- peer-to-peer: rampant content dissemination

  → from audio to movies

  → content providers need to get into the action

  → do not want to get into the action

$600 question:

⟶ what will the wireless/wireline future hold?

Mixture of high bandwidth/low bandwidth networks, wireline/wireless, . . .

Network performance

An overview of key concepts.

Three yardsticks or performance measures:

- throughput: bps or b/s (bits-per-second)
- latency: msec, ms (millisecond)

  $\longrightarrow$ signal propagation speed
- delay: msec

  $\longrightarrow$ includes software processing overhead
- jitter: delay variation

  $\longrightarrow$ standard deviation etc.

Bandwidth vs. throughput:

bandwidth—maximum data transmission rate achievable at the hardware level; determined by signalling rate of physical link and NIC.

throughput—maximum data transmission rate achievable at the software level; overhead of network protocols inside OS is accounted for.

reliable throughput—maximum reliable data transmission rate achievable at the software level; effect of recovery from transmission errors and packet loss accounted for.

$\longrightarrow$ true measure of communication speed

$\longrightarrow$ "goodput" or "effective throughput"

$\longrightarrow$ vs. "raw throughput"

Trend on protocol implementation and overhead side:

migration of protocol software functionality into NICs; NIC is becoming a powerful, semi-autonomous device

network processors: programmable NICs and more such as forwarding between NICs, i.e., router

⟶ as opposed to ASIC based devices

⟶ trade-off between hardware & software

⟶ boundary between hardware & software blurred

With proliferation of wireless networks, lower layers have become important in network programming and system design

⟶ possible project topic using iPAQs

Meaning of "high-speed" networks:

- signal propagation speed is bounded by SOL (speed-of-light)

  $\rightarrow$ $\sim$300K km/s or $\sim$186K miles/s

  $\rightarrow$ optical fiber, copper: nearly same

  $\rightarrow$ coast-to-coast latency

  $\rightarrow$ geostationary satellites: $\sim$22.2K miles/s

  $\rightarrow$ limitation of sending a single bit (e.g., as photon)

- can only increase "bandwidth"

  → analogous to widening highway, i.e., more lanes

  → simulatenous transmission

  → a single bit does not travel faster

  → "high-speed" ⇔ "many lanes"

  → completion time of large files faster

Key issue:

$\longrightarrow$ fat and long pipes

$\longrightarrow$ a lot of traffic in transit

$\longrightarrow$ large delay-bandwidth product (transit traffic)

$\longrightarrow$ significant damage before recovery

$\longrightarrow$ reactive cost

$\longrightarrow$ limitation of feedback systems (e.g., TCP)

Some units:

Gbps (Gb/s), Mbps (Mb/s), kbps (kb/s):

10^9, 10^6, 10^3 bits per second; indicates data transmission rate; influenced by clock rate (MHz) of signalling hardware; soon Tbps.

$\longrightarrow$ communication rate: factors of 1000

Common bit rates:

- 10 Mbps (10BaseT), 100 Mbps (100BaseT)
- 11 Mbps (and 5, 2, 1 Mbps) for 802.11b WLAN
- 100 Mbps (FDDI)
- 64kb/s (digitized voice)
- 144kb/s (ISDN line 2B + D service)
- 1.544 Mbps (T1), 44.736 Mbps (T3)
- 155.52 Mbps (OC-3), 622.08 Mbps (OC-12)
- OC-24, OC-48

GB, MB, kB:

2^{30}, 2^{20}, 2^{10} bytes; size of data being shipped; influenced by the memory structure of computer; already TB.

$\longrightarrow$ data size: factors of 1024

$\longrightarrow$ byte over bit

Common data sizes:

- 512 B, 1 kB (TCP segment size)
- 64 kB (maximum IP packet size)
- 53 B (ATM cell)
- 810 B (SONET frame)

Packet, frame, cell, datagram, message, etc.

$\longrightarrow$ *packet* most generic term

Conventional usage

- frame: LAN-level
- datagram: IP packets
- cell: ATM packets
- packet: generic
- PDU (protocol data unit): generic
- message: high-level (e.g., e-mail)