Data Marshalling

CSci551: Computer Networks
SP2006 Thursday Section
John Heidemann

Data Structures and Network Encodings

- a problem from time immemorial
- see Danny Cohen's "On Holy Wars and a Plea for Peace", IEN-137, 1980
- what are the problems:
  - (higher level issues: security, authentication, authroization)
  - computer architecture issues: ex. endianness
    - if you send the number 0x0019
    - 0: 0x00 1: 0x19 or 0: 0x19 1: 0x00
  - "25"
  - also packing rules: for complex data structure, how are they arranged

Endianness

- in memory, how are 32-bit words laid out?
- given 0x12345678 0x9abcdef0
- big-endian: (SPARC, 68k, PPC, Internet protocols)
  - 0x12 0x34 0x56 0x78 0x9a 0xbc 0xde 0xf0
- little-endian: (x86)
  - 0x78 0x56 0x34 0x12 0xf0 0xde 0xbc 0x9a
- when you're on the same computer, it doesn't matter
- when you send data *between* computers, it does

Structure Packing

- how big is the structure:
  - struct s { char a; int b; char c; };
- answers:
  - 6 bytes: 1 + 4 + 1 (typical); 1+2+1?
  - maybe 12 (assuming 4-byte alignment) or maybe 9 = 1+3 + 4 + 1, or maybe 6
- why?
  - depends on processor and the compiler and maybe compiler options
  - most processors require *alignment*

Why care?

- what happens?
  - struct s; write(socket_fd, &s, sizeof(s)); (could output 9 or 12 bytes)
- it means you can't assume a structure is the same as a packet format
  - must understand the host computer
  - and the particular compiler!
- alternative to just writing structures?
  - to guarantee layout on the wire, maybe explicitly copy each field into a memory buffer

Some Partial Solutions

- Internet protocols
  - structure + manually handle byte ordering with HTONL() HTONS() (but then you need to understand your compiler)
  - OR use compiler directive __attribute(packed)
    - controls on-the-wire format, but makes all in-memory accesses slow
- other ways
  - explicit copies
  - (distributed) shared memory (but then common cpu type)
  - RPC: Remote Procedure Calls, provides RPC complier (ex: XDR, "external data representation")
  - CORBA or DCOM (OLE): fancy RPC systems that support objects

Other Data Transmission Issues

- (endianness, structure packing)
- RPC systems might have procedures and even objects
- arrays? lists? pointers? trees?
  - think about how you would marshal these things
  - ex: for a tree: maybe put the data in an array, and then make the pointers be array indicies

Where have you seen these issues?

- project a
- homework 2
- headers of protocols like TCP