Advanced Unix System Administration

Lecture 14
April 9, 2008

Steven Luo
<sluo+decal@OCF.Berkeley.EDU>

Network Address Translation

- Parts of the IPv4 address space are designated non-public
- We can alleviate the IPv4 address crunch if we find a way to route traffic to and from these hosts
- NAT is a clever hack to do this
  - In its simplest form, just map public IPs to private ones one-to-one with some mangling
  - Not useful for the conserving IPs application

Network Address Translation

- Port translation
  - We need some way of keeping track of who sent the outbound traffic, if we're to route the replies correctly
  - Solution:
    - Mangle the source port
    - Keep track of the source ports corresponding to each client connection, and route traffic accordingly
  - This limits the number of outbound connections per client, but that's usually not an issue

Network Address Translation

- Problems with NAT
  - Applications (FTP) frequently include IPs and port numbers in their protocols, so we need to mangle these too
  - Incoming connections can't be handled, so direct connection protocols have a hard time
- Philosophical objections to NAT
  - Hosts behind NAT on the Internet aren't really full peers anymore
  - Only delays the inevitable

The Domain Name System

- People aren't very good at remembering numbers
  - And they're definitely no good at remembering IPv6 addresses!
- Classic solution: the hosts file
  - Domains maintain hosts files, which are distributed and synchronized via FTP
  - Simple, but absolutely does not scale
- DNS provides a way of providing names in a scalable, distributed way

The Domain Name System

- Structure of DNS
  - Hierarchical system, each part of hierarchy separated by dots
  - DNS servers are delegated authority over parts of the DNS zone by lower DNS servers
  - 13 “root” DNS servers store the delegations for the lowest level
- Name resolution
  - Start at root, inquire for name at each level until we get to what we want

The Domain Name System

- DNS resource records (RRs)
  - Fields: name, type (2 bytes), class (2 bytes), TTL (32 bit integer), data length (16 bits), data
  - DNS can store many different types of information; each is assigned a number
    - Types: A (1, IPv4 address), AAAA (28, IPv6 address), CNAME (5, alias), MX (15, mail exchanger), PTR (12, reverse DNS), TXT (16), SOA (6, start of authority), NS (2, name server)
  - Records also have a class – the only useful one nowadays is IN (1, Internet)

The Domain Name System

- DNS over-the-wire format
  - Queries can go over UDP or TCP
    - UDP is recommended, but queries are limited to 512 bytes
  - Fields
    - Transaction ID (2 bytes), flags (2 bytes: query/response, opcode, authoritative, truncated, recursion desired, recursion available, reserved, answer authenticated, reply code), questions, answers, authoritative answers, additional answers