

Network Services

Hands-On UNIX System Administration DeCal

Week 8 — 14 March 2011

Last time

- **Last time:** *(deep breath)* SSH key authentication, GNU screen, RCS version control, PGP encryption, a mail client that sucks less (mutt), netcat, OpenSSL's s_client, and makefiles!
- Any questions about anything so far? (If you're confused, come to OH! The only dumb question is an unasked question.)

Announcements

- **New project servers** are up!
 - blizzard.OCF.Berkeley.EDU, port 22XY (where XY is your group number)
 - SSH keys (from last week) required
- **Final project proposals** are due at the start of next class, **28 March** (after spring break). See the website for details.

Too Many TLAs

- Look up the **OSI reference model** and the **TCP/IP model** if you're curious about networking. We'll take most of it as given and focus on the application layer.
- The **Transmission Control Protocol (TCP)** provides reliable transmission of data over inherently unreliable media. Most network services use TCP (web, mail, chat, SSH...).

Too Many TLAs

- The **User Datagram Protocol (UDP)** is a counterpart to TCP, used when reliability is less important than speed. Used by DNS, TFTP, VoIP, streaming media...
- TCP and UDP transmissions use **ports**, numbered between 1 and 65536 (2^{16}). Ports below 1024 are well-known, and using them on UNIX requires root access.

DNS

- DNS — the **Domain Name System** — is part of core internet functionality. In a nutshell: it's the Internet's phonebook.
- You can ask your DNS server for *records* (do this with `nslookup`, `host`, or `dig`).
- This is done automatically by client software (e.g., Mozilla Firefox) when you attempt to connect to a remote server.



http://www.youtube.com/watch?v=hcaJ1Vp_Ntg
video by CENTR, the Council of European National Top-Level Domain Registries

DNS Records

There are dozens of kinds of DNS records, and a domain can have more than one record.

Some of the most common ones:

- **A** record: an IPv4 address.
- **AAAA** record: an IPv6 address.
- **CNAME** (Canonical Name) record: an alias for another domain (think “symlink”). CNAMEs and all other types of records are mutually exclusive.

DNS Records

- **PTR** record: points to a canonical name (but doesn't continue processing as with a CNAME record). Usually used for reverse DNS lookups.
- **MX** record: specifies mail servers for a domain. Domains can have multiple MX records, listing different mail servers by priority.
- **SRV** record: similar to MX records, but not specific to mail. Can be used by LDAP, Jabber, ...

“Netcraft confirms it...”

- The **Hyper-Text Transfer Protocol** is a simple, text-based protocol. A basic web server can be implemented in [a 25-line bash script](#) — you’ll be writing a basic HTTP client in today’s lab.
- Popular servers: Apache, IIS (“Internet Information Services”), lighttpd...
- You can see this in action with netcat (nc).

More Protocols

- There are dozens of other protocols you probably use on a day-to-day basis: SMTP and IMAP/POP for e-mail, XMPP/OSCAR/MSNP/YMSG for instant messaging, SSH and SCP on EECS Instructional machines...
- Others you may not know about: NFS (Network File System), LDAP (Lightweight Directory Access Protocol), Kerberos...

Server Daemons

- A program that runs in the background is called a *daemon*; many services run as daemons, including Apache and SSH.
- Services are frequently controlled by *init scripts* — e.g., `/etc/init.d/ssh restart` will restart your SSH server. Also, some daemons have their own management tools (e.g., `apache2ctl`).



One to rule them all

(and in the darkness bind them)

- *A super-server daemon* listens on multiple ports, spawning other server processes to handle incoming requests.
- `inetd` connects network sockets to server processes' `stdin` and `stdout`. This makes it really easy to write your own server!
- Other super-servers: `xinetd`, `launchd`...