Experimental Networking (ECSE 4963)

Lab 1, Basic Linux and Networking Commands

Shiv Kalyanaraman

Yong Xia (TA)

shivkuma@ecse.rpi.edu

http://www.ecse.rpi.edu/Homepages/shivkuma

Aug 29, 2002

Basic Commands

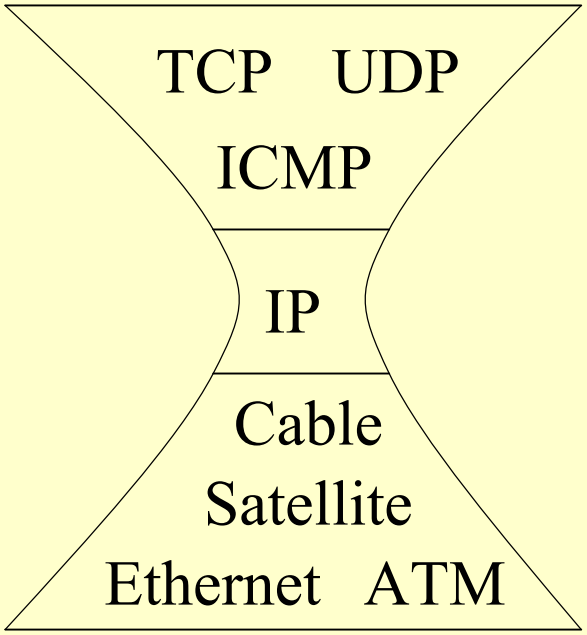
- Linux commands
- Networking commands and tools
- Socket programming
- An introduction project
- Preview of the next lab

Basic Linux Commands

- man – manual page (help info)
- pwd – present working directory
- ls – list content in current directory
- mv / cp / rm – move(rename) / copy / delete file
- mkdir / rmdir – create / delete a new directory
- chmod – change w/r/x modes of file
- ps – check process status
- kill – terminate process
- pipe operator >, >>, |

Internet Protocols Stack

- Packet-switched network
- IP is the glue
- Hour-glass architecture
  - all hosts and routers run IP
  - IP runs over everything
- Common Intermediate Representation



TCP / UDP / IP

- IP
  - Unreliable, best-effort service
  - Connectionless: no per-packet or per-session state information inside network, each IP packet is delivered independent of all other packets
  - Like post-office (USPS) mail

TCP / UDP / IP

- UDP
  - Datagram service: ***explicit boundary*** between packets
  - What's more than IP?
    - Port number: multiplexing for applications
    - Checksum: weak error detection (not correction!)

TCP / UDP / IP

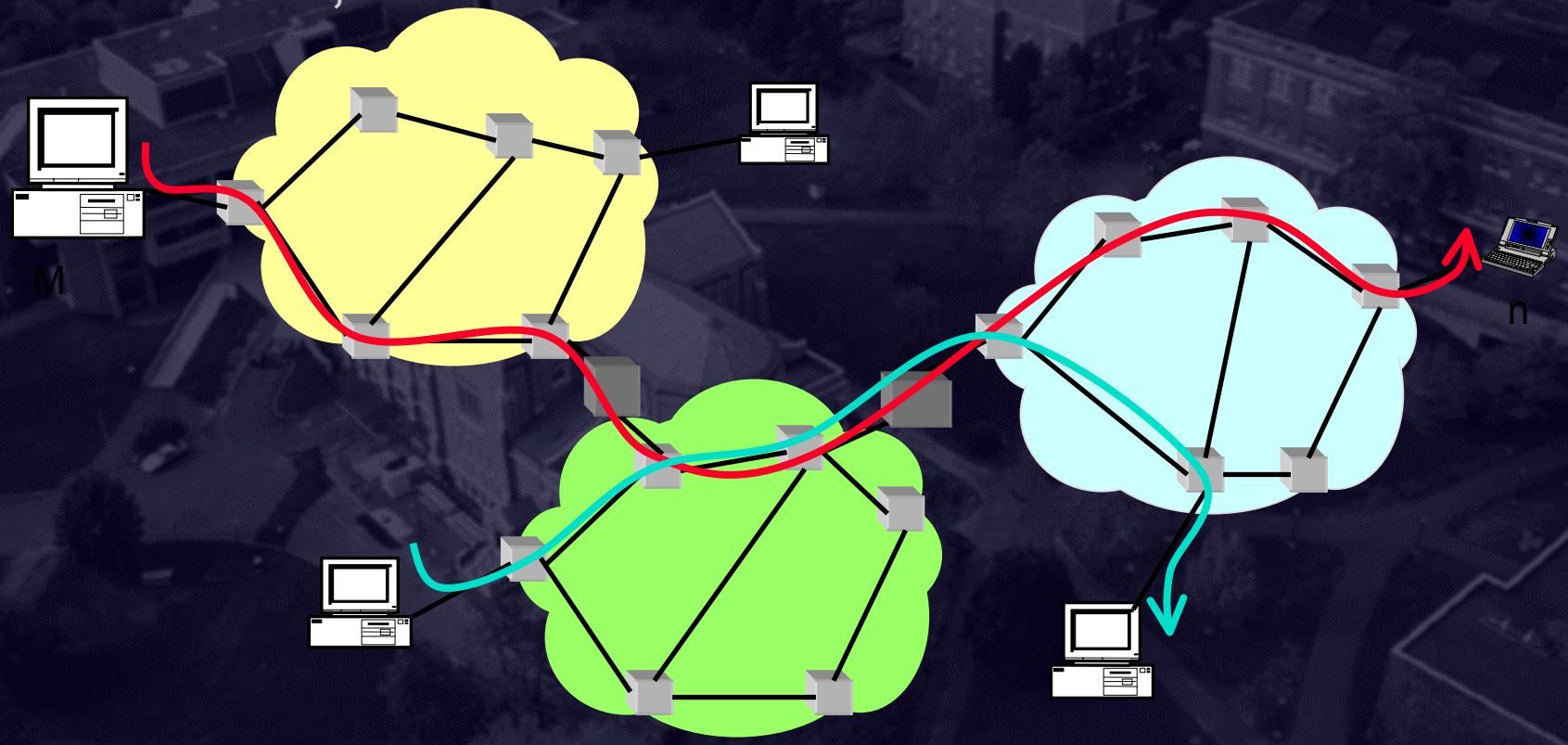
- TCP
  - Many versions: Tahoe, Reno, SACK, Vegas
  - Connection-oriented: ***per-session*** state variables maintained at end-hosts (but not in network, unlike circuit-switched and virtual-circuit approaches)
    - Aka ***end-to-end*** "connection" or "association"
    - Reliably setup and tear down the end-to-end association

TCP (contd)

- Reliable: uses:
  - ACKs ("sender: receiver correctly got this packet"),
  - checksum ("receiver: is this packet is correct or wrong?") and
  - window (multiple packets in flight: pipelined)
- Byte-stream: no application-packet boundary like UDP
- Congestion control: reduce demand during overload, to ensure stable statistical multiplexing of the network

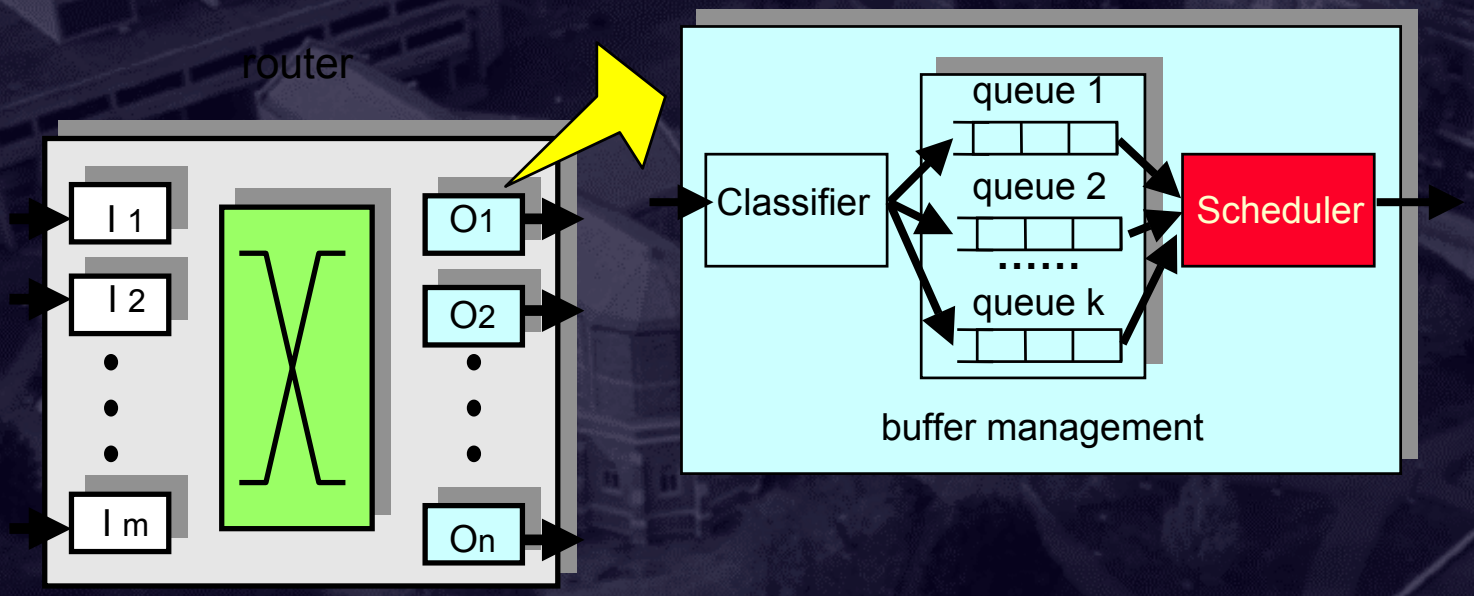
Basic Networking Commands

- How to detect if another machine is alive, the path to it, and resolve its DNS name to ip address, etc. …?



A Router Model

- Input and output queues (buffers)
- Switch fabric (forwarding / routing)



Basic Networking Commands

- ping – check if machine is alive
- ifconfig – interface (ip addr / mask) configuration
- arp – link / network layer address mapping
- netstat – status info of network configuration
- telnet – remote terminal
- ftp – file transfer tool
- route – set static route of a machine
- traceroute – gather route information
- tcpdump – dump packet header
- nslookup – resolve DNS name of target hostname

Key : What can you <u>infer</u> about the network or end-to-end <u>properties</u> with these commands?

Packet's Life in a Router

- Packet is processed at a router:
  - Packet enters at IP input queue from an interface (ifconfig)
  - Calculate next hop router based on longest-prefix match of packet header destination IP address with routing table entry
    - Routing table is maintained dynamically by a daemon (e.g. gated), or statically by route command
    - View routing info etc. via netstat command
  - If it's an ICMP packet, further processing
    - ICMP echo → ping
  - Decrement Time_To_Live (TTL), dicard packet if it's zero and send ICMP error back to source → traceroute
  - Send packet to output queue (forwarding)
  - Nslookup: gives ip address of a host (or router I/f) name
  - Tcpdump can be used to view the whole packet!

Some Miscellaneous Hints

- To check default gateway/static routes without being dependent on DNS:

  netstat -rn

- If you get an error saying command not found, set the path:

  export PATH=$PATH:<new dir>

- Common directories where programs are found:

  /usr/sbin, /sbin, /usr/bin, /usr/local/bin

- If you have root and want to start/stop/restart the networking modules :

  /etc/init.d/network start

  /etc/init.d/network stop

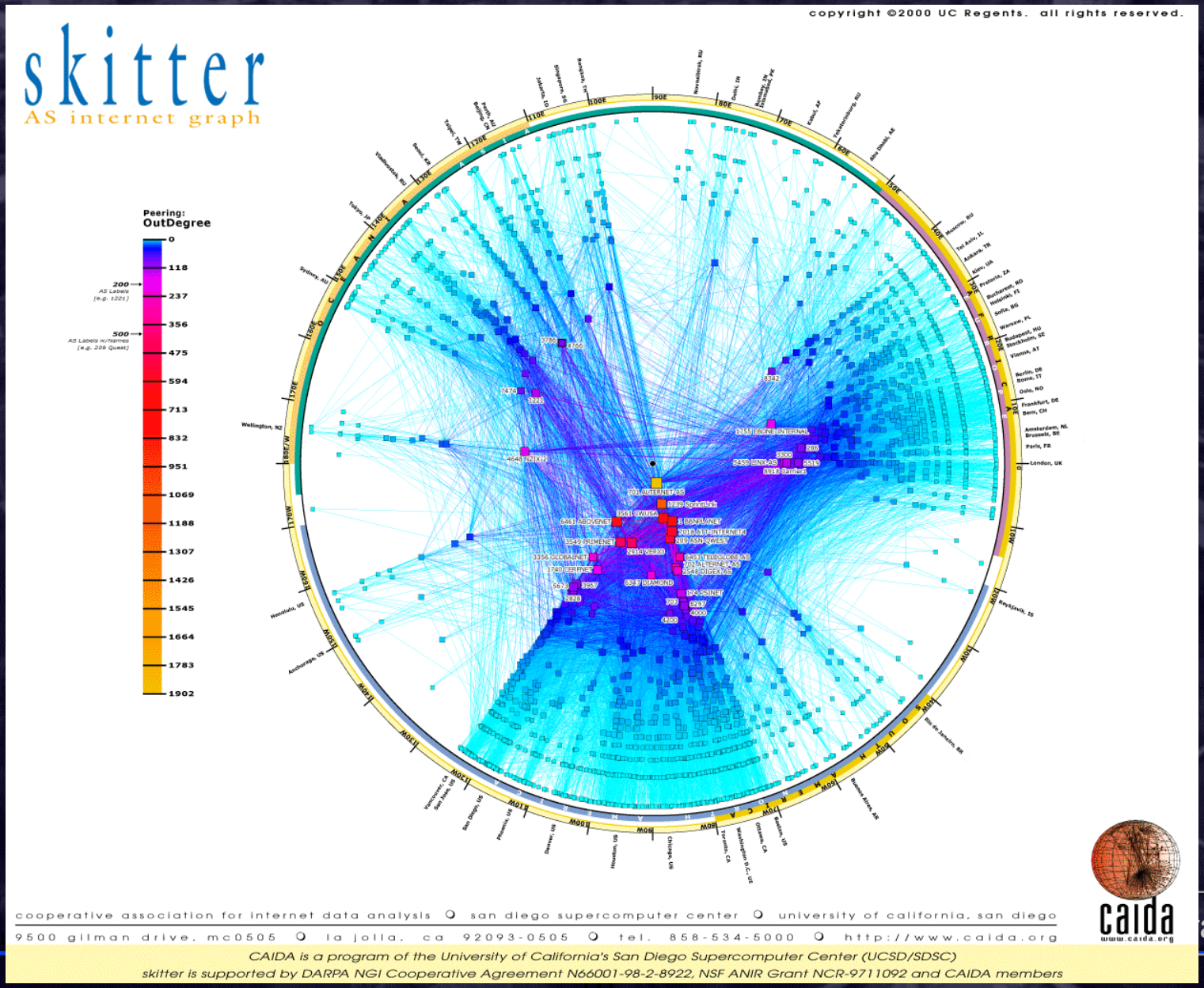
  /etc/init.d/network restart

Project: Measuring RTT

- Write program to measure round-trip time between two end host on the Internet;
  - Refer to ping: write a simple wrapper program..

- Propose a model for RTT prediction, i.e., give a sequence of RTT measures, estimate the next RTT value.
  - Measure several values of RTT. What can you say about the samples? If they are variable, what can you do to reduce the variability of the RTT estimate?
  - Time series model

Advanced Ideas: For Fun!

- Can you correlate info across measurements using different techniques?
  - "Rocketfuel": ISP maps
    - http://www.cs.washington.edu/research/networking/rocketfuel/
  - Skitter: Internet Maps
    - http://www.caida.org/tools/measurement/skitter/
  - Pathchar: bandwidth measurement
    - http://www.caida.org/tools/utilities/others/pathchar/
  - Pathrate/Pathload: load/available bandwidth measurement
    - http://www.pathrate.org/
  - Visualization tools & utilities:
    - http://www.caida.org/tools/visualization/
    - http://www.caida.org/tools/utilities/
  - King: end-to-end latency estimator
    - http://www.cs.washington.edu/homes/gummadi/king/
  - Internet Traffic Archive:
    - http://ita.ee.lbl.gov/index.html

Internet AS Map: caida.org