Router Construction

Workstation-Based

- Aggregate bandwidth
  - 1/2 of the I/O bus bandwidth
  - capacity shared among all hosts connected to switch
  - example: 800Mbps bus can support 8 T3 ports

- Packets-per-second
  - must be able to switch small packets
  - 100,000 packets-per-second is achievable
  - e.g., 64-byte packets implies 51.2Mbps
Switching Hardware

- **Design Goals**
  - throughput (depends on traffic model)
  - scalability (a function of \( n \))

- **Ports**
  - circuit management (e.g., map VCLs, route datagrams)
  - buffering (input and/or output)

- **Fabric**
  - as simple as possible
  - sometimes do buffering (internal)

---

Router Architecture Overview

Two key router functions:

- run routing algorithms/protocol (RIP, OSPF, BGP)
- *switching* datagrams from incoming to outgoing link

---

Input Port Functions

- **Physical layer:**
  - bit-level reception

- **Data link layer:**
  - line termination

**Decentralized switching:**

- given datagram dest., lookup output port using routing table in input port memory
- goal: complete input port processing at ‘line speed’
- queuing: if datagrams arrive faster than forwarding rate into switch fabric
Input Port Queuing

- Fabric slower than input ports combined -> queuing may occur at input queues
- Head-of-the-Line (HOL) blocking: queued datagram at front of queue prevents others in queue from moving forward
- Queuing delay and loss due to input buffer overflow!

Output Ports

- Buffering required when datagrams arrive from fabric faster than the transmission rate
- Scheduling discipline chooses among queued datagrams for transmission

Output port queueing

- Buffering when arrival rate via switch exceeds output line speed
- Queuing (delay) and loss due to output port buffer overflow!
Three Types of Switching Fabrics

Switching Via Memory
First generation routers:
- packet copied by system’s (single) CPU
- speed limited by memory bandwidth (2 bus crossings per datagram)

Modern routers:
- input port processor performs lookup, copy into memory
- Cisco Catalyst 8500

Switching Via a Bus
- datagram from input port memory to output port memory via a shared bus
- bus contention: switching speed limited by bus bandwidth
- 1 Gbps bus, Cisco 1900: sufficient speed for access and enterprise routers (not regional or backbone)
Switching Via An Interconnection Network

- Overcome bus bandwidth limitations
- Banyan networks, other interconnection nets initially developed to connect processors in multiprocessor
- Advanced design: fragmenting datagram into fixed length cells, switch cells through the fabric.
- Cisco 12000: switches Gbps through the interconnection network

Crossbar Switches

- Example crossbar
- Concentrator
  - select 1 of n packets
- Complexity: $n^2$

Knockout Switch
Knockout Switch (cont)

- Output Buffer

Self-Routing Fabrics

- **Banyan Network**
  - constructed from simple $2 \times 2$ switching elements
  - self-routing header attached to each packet
  - elements arranged to route based on this header
  - no collisions if input packets sorted into ascending order
  - complexity: $n \log_2 n$

Self-Routing Fabrics (cont)

- **Batcher Network**
  - switching elements sort two numbers
  - some elements sort into ascending (clear)
  - some elements sort into descending (shaded)
  - elements arranged to implement merge sort
  - complexity: $n \log^2 n$

- Common Design: Batcher-Banyan Switch
IPv6

• Initial motivation: 32-bit address space completely allocated by 2008.
• Additional motivation:
  – header format helps speed processing/forwarding
  – header changes to facilitate QoS
  – new “anycast” address: route to “best” of several replicated servers
• IPv6 datagram format:
  – fixed-length 40 byte header
  – no fragmentation allowed

Major Features

• 128-bit addresses
• Multicast
• Real-time service
• Authentication and security
• Auto-configuration
• End-to-end fragmentation
• Enhanced routing functionality, including support for mobile hosts
IPv6 Addresses

- Classless addressing/routing (similar to CIDR)
- Notation: x:x:x:x:x:x:x (x = 16-bit hex number)
  - contiguous 0s are compressed: 47CD::A456:0124
  - IPv6 compatible IPv4 address: ::128.42.1.87
- Address assignment
  - provider-based
  - geographic

IPv6 Header

- 40-byte “base” header
- Extension headers (fixed order, mostly fixed length)
  - fragmentation
  - source routing
  - authentication and security
  - other options

IPv6 Header (Cont)

Priority: identify priority among datagrams in flow
Flow Label: identify datagrams in same “flow.”
(concept of “flow” not well defined).
Next header: identify upper layer protocol for data
Other Changes from IPv4

- **Checksum:** removed entirely to reduce processing time at each hop
- **Options:** allowed, but outside of header, indicated by “Next Header” field
- **ICMPv6:** new version of ICMP
  - additional message types, e.g. “Packet Too Big”
  - multicast group management functions

Transition From IPv4 To IPv6

- Not all routers can be upgraded simultaneous
  - no “flag days”
  - How will the network operate with mixed IPv4 and IPv6 routers?
- Two proposed approaches:
  - **Dual Stack:** some routers with dual stack (v6, v4) can “translate” between formats
  - **Tunneling:** IPv6 carried as payload in IPv4 datagram among IPv4 routers

Dual Stack Approach
**Tunneling**

Logical view:

A \(\rightarrow\) B:
IPv6

B \(\rightarrow\) C:
IPv6 inside IPv4

C \(\rightarrow\) D:
IPv6 inside IPv4

D \(\rightarrow\) E:
IPv6

E \(\rightarrow\) F:
IPv6

Physical view:

A \(\rightarrow\) B:
IPv6

B \(\rightarrow\) C:
IPv6 inside IPv4

C \(\rightarrow\) D:
IPv6 inside IPv4

D \(\rightarrow\) E:
IPv6

E \(\rightarrow\) F:
IPv6

---

**IPSec**

- Implements network layer encryption and authentication
  - Provides an end-to-end security solution in the network architecture itself
    - Confidentiality, integrity and authenticity of IP datagrams
  - End systems and applications do not need any changes
  - Encrypted packets look like ordinary IP packets and can be easily routed through any IP network
- Included in IPv6 specifications
IPSec Technologies

- Diffie-Hellman key exchange for deriving key material between peers on a public network
- Public key cryptography for signing Diffie-Hellman exchanges to guarantee the identity of two parties and avoid man-in-the-middle attacks
- Bulk encryption algorithm such as DES, 3DES, Blowfish, IDEA, RC4, AES etc.
- Message digest algorithms for ensuring authenticity
- Digital certificates for authentication

IPSec Details

- Refers to several related protocols
  - Described in RFCs 2401 – 2411 and 2451
- Includes
  - IP Security Protocol proper
    - Defines the information to add to an IP packet to enable confidentiality, integrity and authenticity controls
  - Internet Key Exchange
    - Negotiates the security association between two entities and exchanges the keys
- Does not specify any particular encryption technology to use
**IPSec Transport Mode of Operation**

- Only the IP payload is encrypted, the headers are left intact
  - Adds only a few bytes to each packet
  - Allows devices to see the source and destination addresses
    - Enables intermediate routers to provide special services based on IP header
    - Allows attacker to perform certain traffic analysis based on this information

**IPSec Tunnel Mode of Operation**

- The entire IP datagram is encrypted, including the IP headers
  - Source and destination addresses are also hidden
  - Prevents traffic analysis by attacker
- Used in VPNs

**IPSec Security Associations**

- A Security Association (SA) is a statement of the negotiated security policy between two communicating devices
  - Which algorithms have been used for security services?
  - What are the keys used?
- IPSec uses a SA to track down the parameters in a given session.
  - For a bi-directional communication between A and B two SAs are established
### Security Association

- A one-way relationship between sender & receiver that affords security for traffic flow
- Defined by 3 parameters:
  - Security Parameters Index (SPI)
  - IP Destination Address
  - Security Protocol Identifier
- Has a number of other parameters
  - Seq no, AH & EH info, lifetime etc

### Authentication Header

- Used to ensure the authenticity of the data
  - Provides support for data integrity & authentication of IP packets
  - Data includes the entire IP payload including transport layer headers and also the invariant data in the IP header (like source address, destination address etc.)
  - Prevents address spoofing attacks by tracking sequence numbers
- Uses keyed message digest algorithms rather than digital signatures to ensure authenticity
- Does not provide any confidentiality protection to the payload
Encapsulating Security Payload

- Used to ensure confidentiality, integrity and authenticity of data
  - Data includes the entire IP payload but not any portion of IP header (unlike AH)
- Uses encryption algorithms like DES, IDEA, Blowfish, RC4 and the more recent AES for confidentiality, and message digest algorithms like MD5 and SHA for integrity and authenticity

Encapsulating Security Payload (ESP)

IPSec Key Management

- Handles key generation & distribution
- Typically need 2 pairs of keys
  - 2 per direction for AH & ESP
- Manual key management
  - Sysadmin manually configures every system
- Automated key management
  - Automated system for on demand creation of keys for SA's in large systems
  - Has Oakley & ISAKMP elements
Internet Key Management Protocol (IKMP/ISAKMP)

- IKMP negotiates security association
  - Internet Key Exchange (IKE) is the standard method used.
  - First creates an authenticated secure tunnel between two entities and then negotiates the security association for IPSec over this tunnel

IKE Steps

- Two step process
  - Authentication of peers
    - Using Pre-shared keys
      - Requires manual configuration
    - Using Public key cryptography
      - Does not ensure non-repudiation
    - Using Digital signatures and public key certificates
      - Ensures non-repudiation
  - Key exchange to generate secure tunnel
    - Uses Oakley (authenticated Diffie-Hellman) key exchange