Internet Peering, IPv6, and NATs

Mike Freedman

V22.0480-005

Networks

Internet Peering

Review: Routing

- Internet has a loose hierarchy of domains
  - Hosts now local router
  - Local routers know site routers
  - Site routers know core routers
  - Core routers know everything
- Organized by Autonomous System (AS)
- Two-level route propagation
  - Interior gateway protocol (each AS uses own)
  - Exterior gateway protocol (Internet-wide)

Types of ASes

- Local traffic – packets with src or dst in local AS
- Transit traffic – passes through an AS

- Stub AS
  - Connects to only a single other AS
- Multihomed AS
  - Connects to multiple ASes
  - Carries no transit traffic
- Transit AS
  - Connects to multiple ASes and carries transit traffic

How do ASs establish routing?

- Purchase transit services from provider
  - Small ISP receives access to all of transit providers routes, thus has view of entire Internet
  - Simple from customer perspective
  - Often good service due to customer relationship
  - Can become expensive ($100Ks / month)

How do ASs establish routing?

- Establish peering relationships
  - ISPs reciprocally provide access to each others' customers (announce routes)
  - Peering is not transitive

Three-steps to peering

- Identification of Potential Peers
- Peering Negotiation
- Implementation Discussions

Why peer?

- Lower transit costs
  - Equivalent peering transport is ~10x cheaper
- Lower Latency
- Usage-based traffic billing
  - Lower latency, lower packet loss

    → Handle more traffic from customers

    due to TCP congestion control, hence more $

Why not peer?

- Traffic asymmetry and investment
- May be potential for transit sales
- Peering consumes resources
- Don’t help business competitors
- Peers have not Service Level Agreement to guarantee problems get repaired

- Paid peering model emerging if traffic asymmetry reaches ratio (4:1)

With whom to peer?

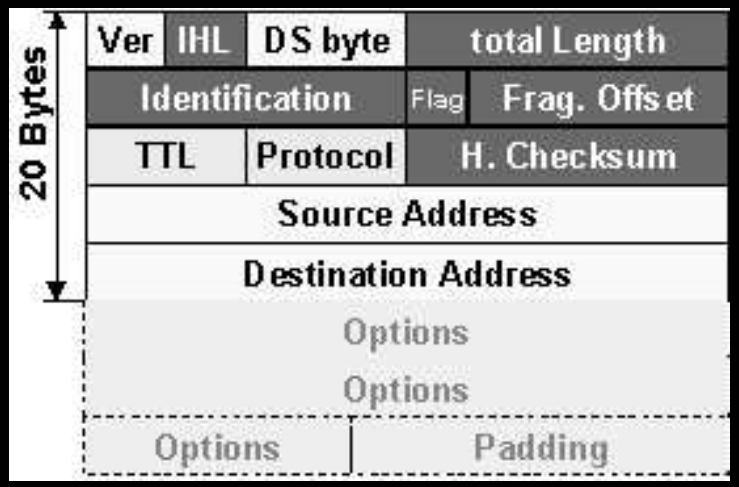
- Calculate quantities of traffic involved
- Business arrangements / peering policies
- Migration path from transit to peering
  - Access internet via transit from global provider
  - Pursue peering arrangements on public switches
  - Migrate high-traffic public peering to private interconnections (e.g., fiber)
  - Attempt to negotiate peering with former transit provider, although difficult

IPv6

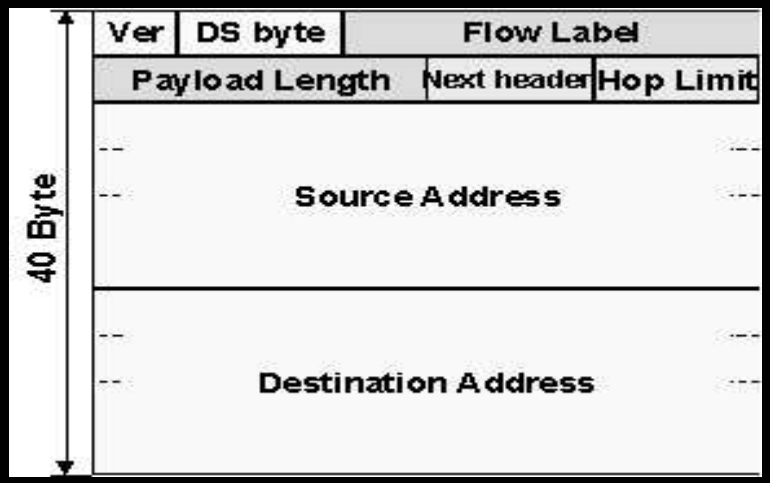
IPv6

- Motivation: running out of IP addresses
  - Subnetting and “classless” routing have helped
  - But still hard to achieve 100% address utilization
- Other goals:
  - Header format simplification, reduce common-case processing costs
  - Improved support for extensions and options
  - Flow labeling capability for flows that request special handling, such as QoS or real-time service
  - Authentication, data integrity, data confidentiality

Packet headers



IPv4
32-bit address space



IPv6
128-bit address space

Address Allocation

- Internet organizing as autonomous systems (AS), seek to provide ways to hierarchically aggregate routing information
- Nontransit AS as subscriber (think NYU)
- Transit AS as provider (think UUNET, AT&T)
- Goal is to aggregate routing information to reduce the burden on intradomain routers
  - AT&T needs to know about UUNET, not about NYU

Address Allocation

- Flexible address assignment like CIDR (Classless Inter-Domain Routing for IPv4)
- Before CIDR:

  126 Class A (/8), 16,777,214 hosts each
  Plus 65,000 Class B (/16), 65,534 hosts
  Plus 2 million+ Class C (/24), 254 hosts

- After CIDR:

  /15 512 Class C 131,072 hosts
  /14 1,024 Class C 262,144 hosts
  /13 2,048 Class C 524,288 hosts

IPv6 address format

3	m	n	o	p	125-m-n-o-p
010	RegistryID	ProviderID	SubscriberID	SubnetID	InterfaceID

- Assign sizes of “m” based on size of provider
  - Short m allows more subdivisions
- Could aggregate at geographical boundaries
  - Europe assigned one RegistryID, it handles sub-allocation to European providers
  - What if subscriber has more than one provider?
    - 2 providers could share joint prefix

47CD:1234:4422:AC02:0022:1234:A456:0124

Autoconfiguration

- IPv6 supports stateless autoconfiguration
- Two-part problem
  - Obtain interface ID that is unique on link
    - Use Ethernet 48-bit address

:1234:A456:0124

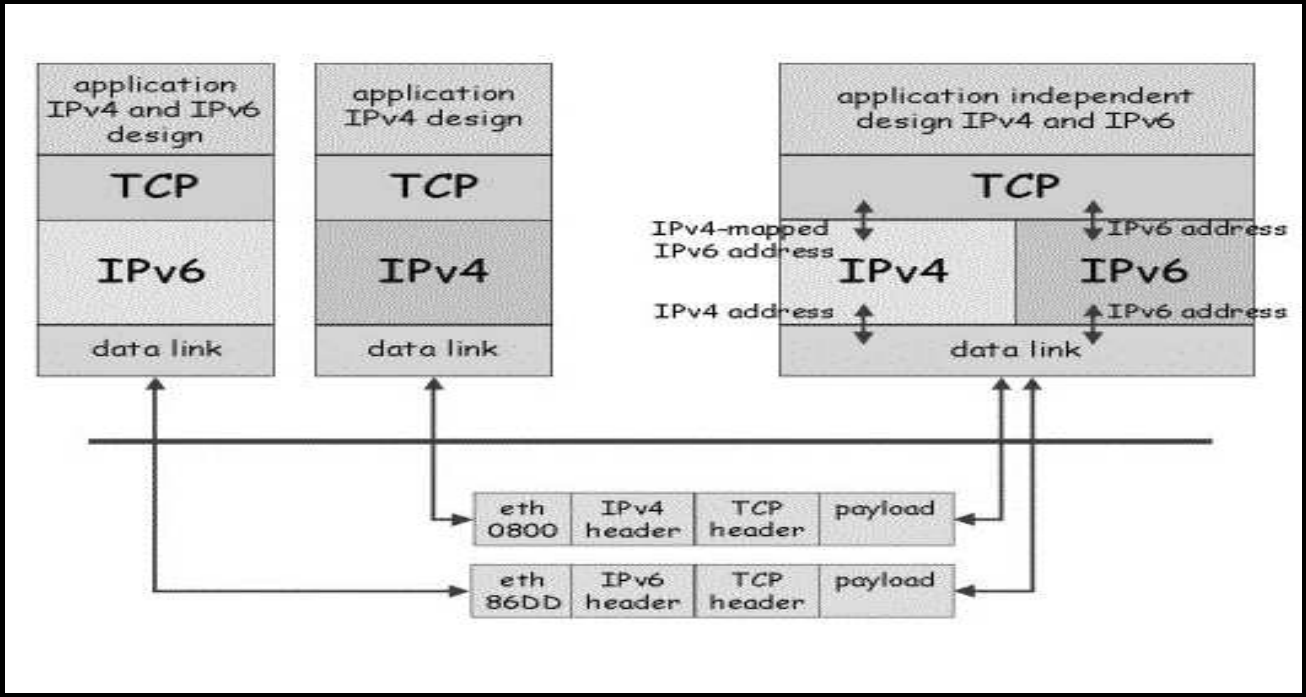
  - Obtain correct address prefix for that subnet
    - Find prefix from subnet's router

47CD:1234:4422:AC02:0022:1234:A456:0124

IPv4 to IPv6 Transition

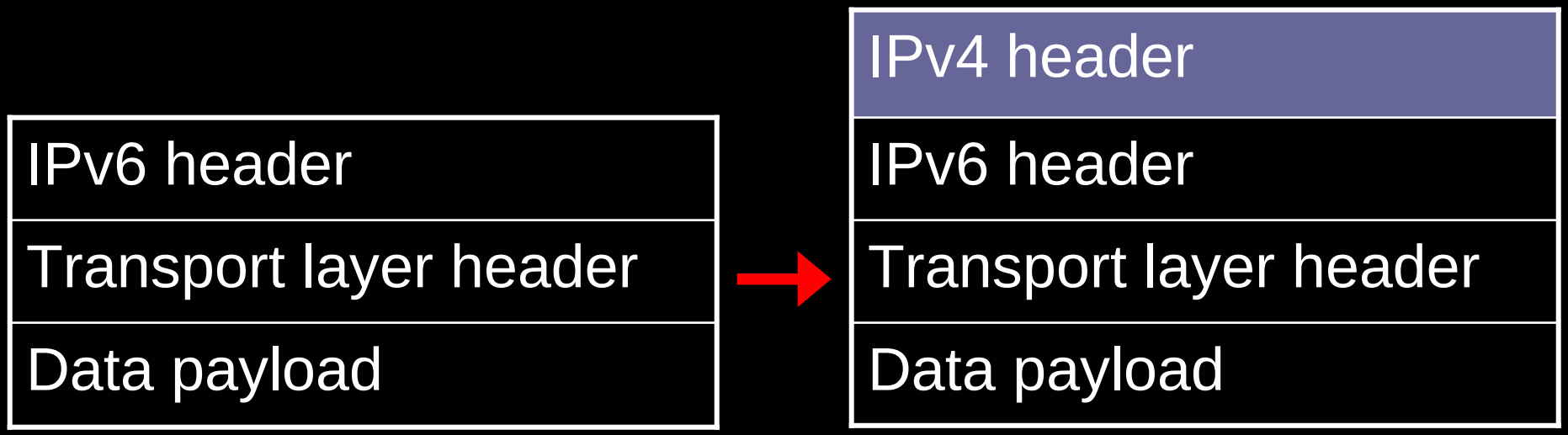
- Deployment must be incrementally
  - IPv4 speak to IPv4 or IPv6
  - IPv6 speak to IPv6 over IPv4 network
- Aggregated Global Unicast Addresses (prefix 001) are like classless addresses
- Two mechanisms for transition

IPv4 to IPv6 Transition: Dual-Stack



- Nodes runs both protocols and use Version field
- IPv4 and IPv6 addresses may be unrelated
- If IPv6 client connects to IPv4 server, server needs IPv4 packet, but client could use IPv4-mapped IPv6 address
  - 2 leading bytes of 1's, 32-bit address, pad with 0's
  - ::FFFF:128.96.33.81

IPv4 to IPv6 Transition: Tunneling



- Encapsulate IPv6 as payload of IPv4 pkt
  - Handle TTL correctly (IPv6 sees as single hop)
  - Handle fragmentation carefully
  - Handle IPv4 ICMP msgs and pass back error to IPv6 initiator

Network Address Translation:

A Stop-Gap Mechanism

Network Address Translation

- Ad-hoc way to conserve IP address space currently used while IPv6 not deployed
- Key idea:
  - All hosts do not need globally-unique addresses
  - Hosts could be assigned private, locally-unique addresses, i.e., within NYU (10.*, 192.168.*)
  - If wish to communicate to globally, NAT box translates from private address to global address
- Static NAT, Dynamic NAT
- Overloading (using ports), Overlapping

Network Address Translation

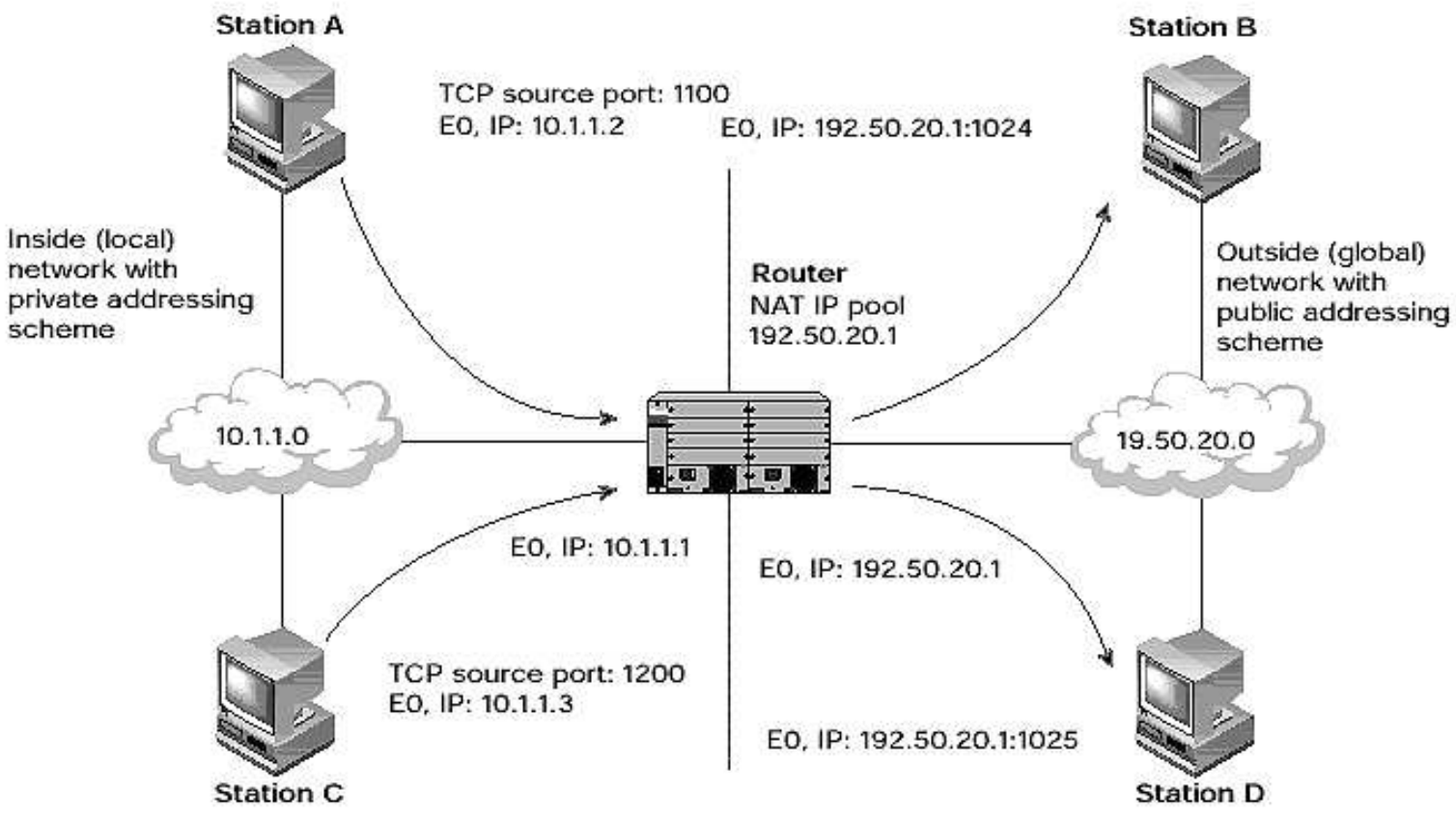


Figure 6: NAT with Port Address Translation (PAT) of global addressing

No global addressability, uniqueness

- Protocols often carry IP address in messages (e.g., active FTP, RTSP), must be specially handled
- Some apps/servers use IP address as identifier: rlogin/rsh, Kerberos, IPsec, ONC RPC/NFS.
- Almost any distributed or ``cluster computing'' system (e.g., PVM, MPI) use IP as routing id
- Cannot associate state with IP addresses alone
- Peer-to-peer systems

Other problems with NATs

- Persistence of host-to-address binding
  - NATs and application sessions are not aware of one another
- Some applications assume same address, same host (e.g., load balancing)
- Deployment of new apps (e.g., peer-to-peer)
- Reliability, Scalability
- Private address spaces may conflict, esp. coupled with virtual private networks (VPNs)