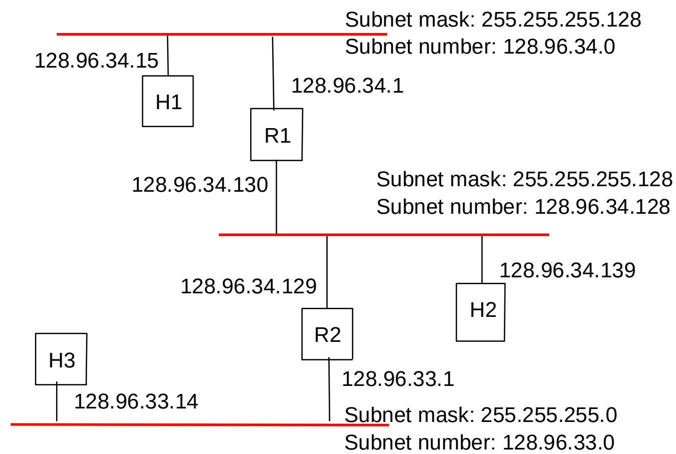
CSE/EE 461 Lecture 9
Interdomain Routing

Tom Anderson
tom@cs.washington.edu
Peterson, Chapter 4.3

Scalability Concerns

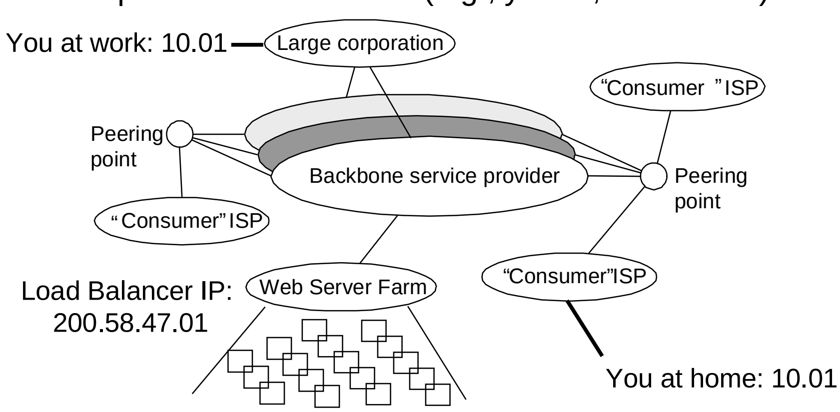
- Size of routing tables
  - Solution: Hierarchical IP addresses
    - Allocate addresses to match network structure
    - Aggregate addresses dynamically
    - Virtual IP addresses
- Unscalable routing algorithms
  - Solution: Interdomain routing (BGP)
    - Separately route inside an organization vs. between domains
    - Explicit policy knobs for crossing organizational boundaries

Subnet Example



NAT/Load Balancer Example

Private IP address spaces; no more global connectivity except for visible services (e.g., yahoo, kmart.com)



Network Address Translation

- Every network, organization, or ISP can have its own private IPv4 address space
  - Example: hosts assigned 10.01, 10.02, …
  - Internal communication occurs normally
  - All external communication goes through NAT
  - NAT transforms each packet to maintain illusion of global Internet addresses

NAT Mechanics

- Host wants to access an external web service
  - Sends request; From: 10.01; To: 200.58.47.01
- NAT transforms outgoing packets
  - Rewrite source address so reply comes back
    - From: 128.80.40.01, port 5736; To: 200.58.47.01
  - Return address is dynamically allocated on connection setup; torn down on connection termination
- NAT transforms incoming packets
  - Match incoming reply to internal host making request
  - Rewrite destination address so reply goes to host
    - From: 200.58.47.01; To: 10.01

Load Balancers

- A highly available, scalable web service can require thousands of servers
- Original approach:
  - Use DNS to translate web service name to IP addresses of individual servers
  - Load balance by “round robin” – give different clients different server addresses
- Want server failures to be transparent
  - Use DNS to translate name to a single address
  - Load balancer forwards incoming requests to servers
  - Translates each incoming/outgoing packet, as in NAT

Load Balancer Mechanics

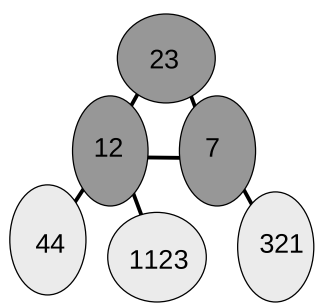
- Client requests arrive at load balancer
  - From: 221.xx.yy.zz; To: 200.58.47.01
- Load balancer chooses which server to use
  - Can be based on packet contents (e.g., URL)
  - Rewrites packet to appear to be directed to server
  - From: 221.xx.yy.zz; To: 10.53
- Server replies to client
  - From 10.53; To: 221.xx.yy.zz
- Load balancer rewrites packet to appear to come from service, not individual server
  - From 200.58.47.01; To: 221.xx.yy.zz

NAT/Load Balancer Failover

- Are NATs/load balancers transparent?
  - some protocols put IP addresses in payload
  - what about failures?
- On failure, current connections => lost
  - NAT/load balancers have "hard" state
  - Clients can retry; want new connections to work
- Hot standby NAT/load balancers on same subnet
  - Too slow to change DNS translation to point global name to a different NAT/load balancer
  - Instead, replacement NAT/LB uses ARP to pretend to be IP address of failed machine

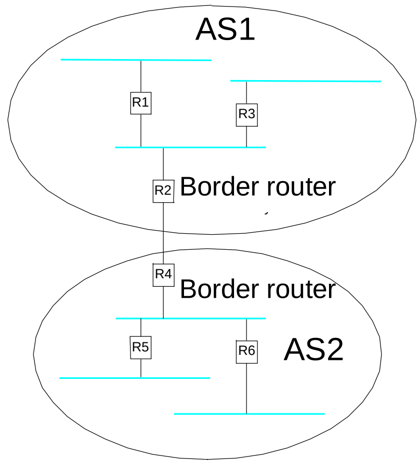
Inter-Domain Routing

- Network comprised of many Autonomous Systems (ASes) or domains
- To scale, use hierarchy: separate inter-domain and intra-domain routing
- Also called interior vs exterior gateway protocols (IGP/EGP)
  - IGP = RIP, OSPF
  - EGP = EGP, BGP



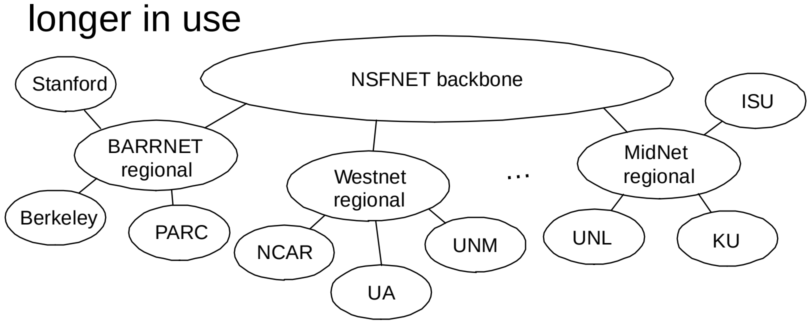
Inter-Domain Routing

- Border routers exchange AS paths to reach each destination
- Route to border, through AS to next AS, …, to destination AS, to host
- Internal routers can use notion of default routes
- Core is “default-free”; routers must have a route to all networks in the world



Exterior Gateway Protocol (EGP)

- First major inter-domain routing protocol
- Constrained Internet to tree structure; no longer in use

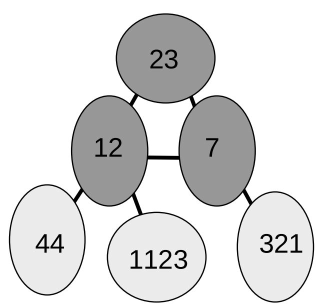


Border Gateway Protocol (BGP-4)

- Interdomain routing protocol used in the Internet backbone today
- Features:
  - Path vector routing
  - Policy knobs to provide operators control over routing
  - Operates over reliable transport (TCP)
  - Uses route aggregation (CIDR)

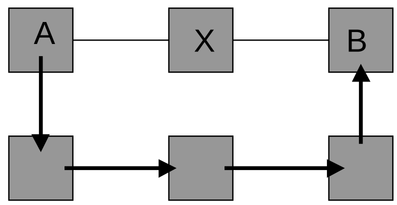
Path Vectors

- Similar to distance vector, except send entire paths
  - Ex: 321 hears [7,12,44]
  - Helps avoid loops
  - Supports policy knobs
- Shorter paths (# of AS's) are chosen in preference to longer ones



Policies

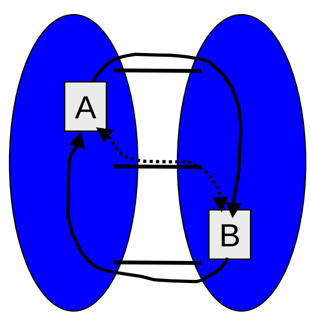
- Choice of routes may depend on owner, cost, acceptable use policy, …
  - Business considerations
- Local policy dictates what route will be chosen and what routes will be advertised!
  - Example: X doesn't provide transit for B, or A prefers not to use X



a

Impact of Policies – Example

- Early Exit / Hot Potato
  - "if it's not for your customers, get rid of it"
- Combination of local policies not globally best
- Side-effects: asymmetry, persistent loops, …



Examples of Policies

- Direct preferences
  - Which provider (and which link at each provider) should I favor for a specific destination?
  - Example: use free peering over paid transit; use fat pipes over thin ones; measure and optimize; …
- Reverse path preferences
  - Which link does my provider prefer me to use?
    - where does my provider have extra capacity?
  - Which destinations does my provider serve directly and/or peer with?
    - which destinations does my provider want me to send to it?
  - Each AS is free to ignore

Operation over TCP

- Most routing protocols operate over UDP/IP
  - Use periodic updates to recover from lost routing packets
- BGP uses TCP
  - TCP handles error control; reacts to congestion
  - Allows for incremental updates; only notify about changes (new links, retracted links)
- Issue: data vs. control plane
  - Shouldn't routing messages be higher priority than data?

Interdomain Routing Summary

- Route aggregation (CIDR) improves scalability
  - Many large organizations connect to multiple providers; impedes route aggregation

Scalable Routing Summary

- Hierarchical address allocation helps routing scale
  - Addresses are constrained by topology
  - Only need to advertise and compute routes for network aggregates
- Internet is a collection of Autonomous Systems (ASes)
  - Policy dominates routing at the AS level
- Structural hierarchy helps make routing scalable
  - BGP routes between autonomous systems (ASes)