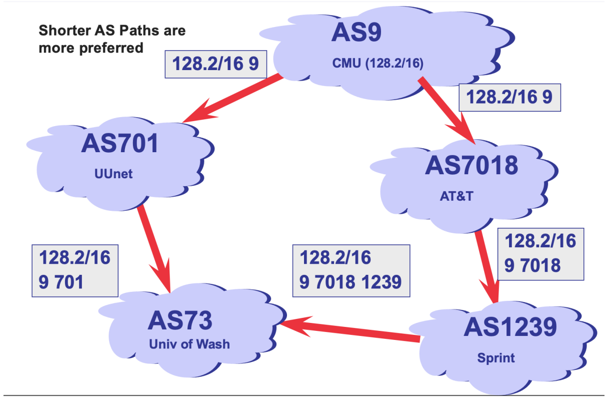
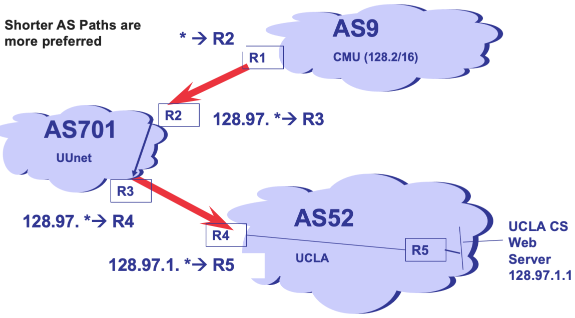
08 - Border Gateway Protocol

Border Gateway Protocol (BGP)

- border router ⟺ edge router
- protocol for inter-AS comms (AS = Autonomous System)
- ASes have AS ids bc ASes may have many prefixes
- routing is done hierarchically with shortest AS paths



- hierarchical routing because not all routers store routes to all ASes or even networks, so make routers hierarchical and AS edge routers jut route all to root/major routers which know dest routing and proceed with hops



- however, because weaker edge routers don't know abt all domains/ASes, they use the BGP to optimize routes without knowing abt all paths
  - BGP uses path vector protocol instead of distance vector to know shortest path instead of storing all possible distances
- e.g., only allow govt packets through ARPANET, if don't know the dest domain → just forward to 701 (image above)
- multihoming - multiple ISPs service the domain
- peer-to-peer - usually bw ISPs to share paths

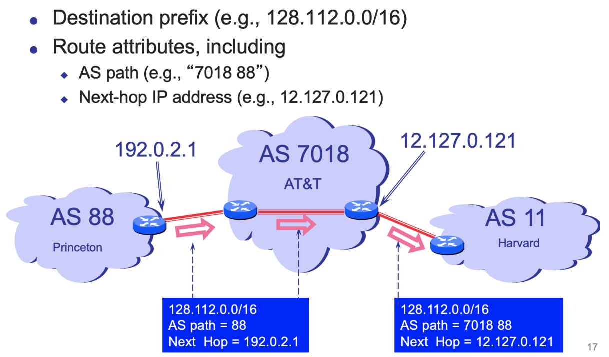
BGP Session

- basic operation steps:

1. Establish session
   1. requires TCP connection between edge routers of 2 domains
2. exchange all active routes
3. while connection is true, exchange updated routes

- nodes learn multiple routes between domains and store in a routing table - w/ incremental updates

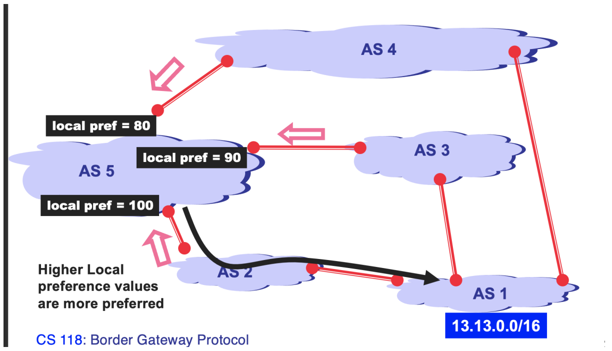
- routing packets (not packet forwarding) sent to fill routing table



- Destination prefix (e.g., 128.112.0.0/16)
- Route attributes, including
  - AS path (e.g., “7018 88”)
  - Next-hop IP address (e.g., 12.127.0.121)

- generally speaking, most of the time just dump to IP which knows AS paths to get to dest
- edge routers only need to know next hop addr
- once packet is on the line, it ARPs to get MAC of router and propagate
- origin - route from inside (IGP) or outside (EFP)
- local pref - stat ranked paths within AS (preferred entry)
- multi-exit discriminator - decide which router to exit from
- community - opaque data used for tag routes that are treated equivalently?

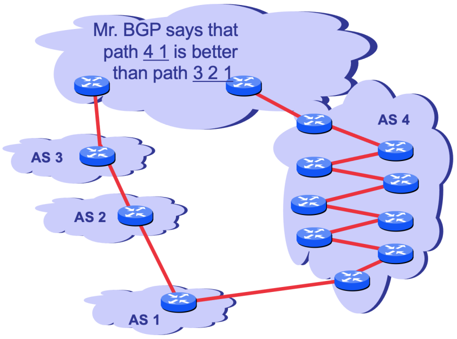
BGP Decision Tree

- Default decision for route selection
  - Highest local pref, shortest AS path, lowest MED, prefer eBGP over iBGP, lowest IGP cost, router id
    - prefer eBGP over iBGP bc eBGP is more direct edge router from AS to AS
- Many policies built on default decision process, but…
  - Possible to create arbitrary policies in principal
    - Any criteria: BGP attributes, source address, prime number of bytes in message, …
    - Can have separate policy for inbound routes, installed routes and outbound routes
  - Limited only by power of vendor-specific routing language
- 

Higher Local preference values are more preferred

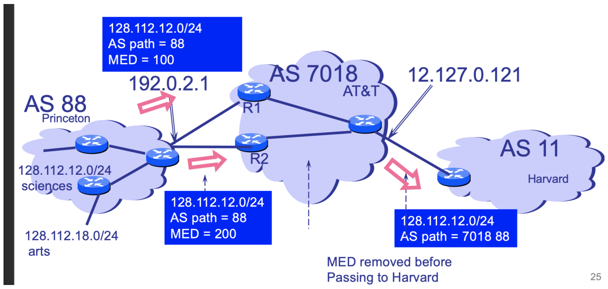
CS 118: Border Gateway Protocol

- shortest path preference is AS context not router/hop context ⇒ greedy relative to AS paths but may not be optimal in number of router hops WITHIN an AS



Optimizations

- MEDs - router-level load balancing via MEDs to prefer router delivery



  - example Cisco config to set MEDs

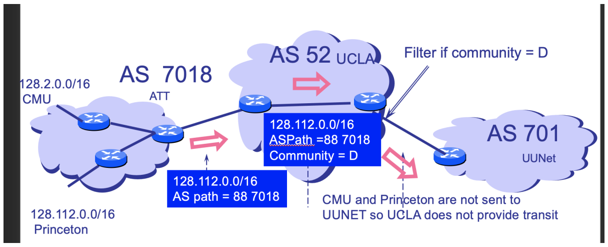
```
neighbor R1 route-map setMED-R1 out
neighbor R2 route-map setMED-R2 out

access-list 1 permit 128.112.12.0 255.255.255.0 //sciences
access-list 2 permit 128.112.18.0 255.255.255.0 // arts

route-map setMED-R1 … match ip address 1 set metric 100
// for R1 send science prefix with lower MED priority
route-map setMED-R1 … match ip address 2 set metric 200
// for R1 send arts prefix with higher MED prioriity

route-map setMED-R2 … match ip address 1 set metric 200
// for R2 send science prefix with higher MED priority
route-map setMED-R2 … match ip address 2 set metric 100
// for R2 send arts prefix with lower MED priority
```

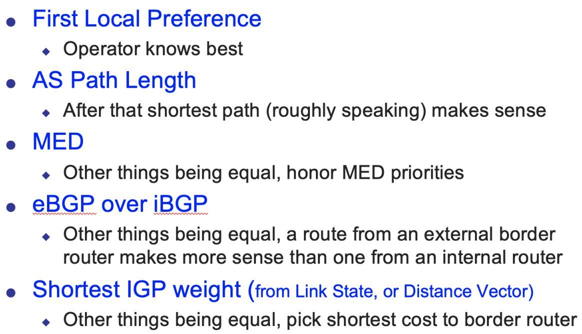
- community - way to tag multiple equivalent routes with same tag value
  - remote routers can filter via tag
  - e.g., `NOTRANSIT` if not in ISP network
  - add community tag to routing packets which is the same tag for all ISPs the AS pay



    for so enable routes if `community == D`
- route aggregation - combine paths to the same AS to reduce cached routes

  - now create routes as sets of route via union

BGP Optimization Preference

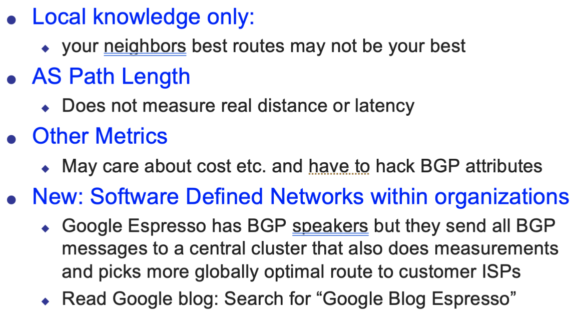


- First Local Preference
  - Operator knows best
- AS Path Length
  - After that shortest path (roughly speaking) makes sense
- MED
  - Other things being equal, honor MED priorities
- eBGP over iBGP
  - Other things being equal, a route from an external border router makes more sense than one from an internal router
- Shortest IGP weight (from Link State, or Distance Vector)
  - Other things being equal, pick shortest cost to border router

BGP Drawbacks

- Instability
  - Route flapping (network x.y/z goes down… tell everyone)
  - Long AS-path decision criteria defaults to DV-like behavior (bouncing)
  - Not guaranteed to converge, NP-hard to tell if it does
- Scalability still a problem
  - > 500,000 network prefixes in default-free table today
  - Tension: Want to manage traffic to very specific networks (eg. multihomed content providers) but also want to aggregate information.
- Performance
  - Non-optimal, doesn’t balance load across paths
- multi-homing gaming
  - extra reliability but vulnerable to gaming by switching ISP networks depedning on cost
  - ISP usually charge at 95th percentile traffic/usage

BGP is Suboptimal

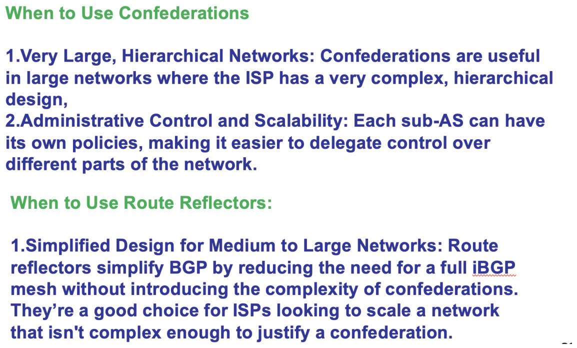


- Local knowledge only:
  - your neighbors best routes may not be your best
- AS Path Length
  - Does not measure real distance or latency
- Other Metrics
  - May care about cost etc. and have to hack BGP attributes
- New: Software Defined Networks within organizations
  - Google Espresso has BGP speakers but they send all BGP messages to a central cluster that also does measurements and picks more globally optimal route to customer ISPs
  - Read Google blog: Search for “Google Blog Espresso”

- Google Espresso, use central SDN to determine which BGP router to forward to external outside of WAN
- "hack" others' BGP by calculating latency across external WANs and store in central SDN to forward externally

[Optional] Scaling iBGP

- The default way of a full mesh between all border routers has O(N^2) overhead, where N is # border routers

- Two common ways to scale IBGP in large ISPs: confederations and route reflectors
    - In confederations, we divide a large AS into stub AS’s hierarchically, so stub AS’s don’t know internals of each
    - In route reflectors, leaf border routers send BGP messages to a central reflector that sends to all clients. Can generalize to a tree of reflectors.
    - 

When to Use Confederations

1.Very Large, Hierarchical Networks: Confederations are useful in large networks where the ISP has a very complex, hierarchical design,
2.Administrative Control and Scalability: Each sub-AS can have its own policies, making it easier to delegate control over different parts of the network.

When to Use Route Reflectors:

1.Simplified Design for Medium to Large Networks: Route reflectors simplify BGP by reducing the need for a full iBGP mesh without introducing the complexity of confederations. They’re a good choice for ISPs looking to scale a network that isn't complex enough to justify a confederation.