Mechanism vs. Policy

• Routing Mechanism
  - Path discovery for end-to-end connectivity
  - Hop-by-hop forwarding along a path

• Routing Policy
  - Deciding which routes to advertise
    - For which destinations, to whom?
  - Determining which packets to forward
    - Over what links, at what rate, for whom?
Wide-Area Routing

- **Control Plane**
  - Each AS computes paths to destinations using received advertisements
    - Actual path selection based upon tuning parameters
  - Selectively exports routes to neighbors based upon business relationships
    - Often changes/removes/rewrites tuning parameters

- **Data Plane**
  - Next hop selected according to local information
    - Destination addresses, current router, arrival link, etc.
  - Possibly filter inappropriate traffic
    - Drop traffic that “shouldn’t” be here
Some Current Frustrations

- BGP is extremely difficult to configure
  - Forced to use ‘assembly language’ to express mechanism and local business policy
- Poor performance
  - Recovery from failure can take a long time
  - Despite the existence of workable routes
- Poor flexibility
  - ASes can’t control routing outside of their network
  - Special-case modifications on human time-scales

All symptoms of policy-mechanism link
The Goal

- Enforce all policies (only) while forwarding
  - We need some amount of filtering anyway
  - Removes complexity from control plane
  - Route discovery becomes policy neutral

❌ Could need lots of information at each router
  - Need descriptions of all applicable policies
  - Information required as input to policy decision

✅ Instead, compute policy decisions offline
  - Stamp each packet with a proof of compliance
  - Forwarding check reduces to stamp verification
Network Capabilities

- Verifiable attestation of policy compliance
  - Valid for a particular portion of the network
  - “Signed” by an authorized party
  - Designates a resource (billable) principal

- Capabilities are composable & transferable
  - Capabilities can be exchanged between entities
  - To use, need to *bind* to a particular packet
  - Packets can carry more than one capability
Capability Binding

- Authorization agent has a secret symmetric key, $k$, shared with routers in the region
- Define a per-capability secret, issued with $c$
  - $s = \text{MAC}_k(c)$
- Compute a per-packet binding
  - $B = \text{MAC}_s(p)$
- Routers can verify packet bindings
  - $B' = \text{MAC}_{\text{MAC}_k(c)}(p)$
Platypus

- For now, loose source routing is an out
  - Capabilities to attest to policy compliance
  - (We don’t handle route discovery)
- Allow Intra-AS traffic engineering
  - Each ISP engineers its own network
  - ISPs can decide granularity of control
- Support accountability & (gasp!) billing
  - Capabilities identify a resource principal
Efficient Overlay Construction
Intra-AS Router Variation

![Graph showing intra-AS router variation with different delay values for AS3549 (GBLX) to Lulea, Sweden, and Anaheim, CA to AS3549 (GBLX), with regions marked for West Coast, Mid West, East Coast, and Western Europe.]
Intra-AS Router Variation

![Graph showing delay variations between different regions and ASes.](image-url)
Ongoing work

- **Capability Distribution**
  - Broadcast encryption
  - Lightweight capability revocation

- **Performance**
  - Flow-based authentication
  - Probabilistic verification

- **Accounting**
  - Hierarchical resource principal naming
  - Distributed token buckets
  - Windowed Bloom filters?