CSE 534
Intra-domain Routing

Vyas Sekar
Stony Brook University

Based in part on content borrowed from:
David Maltz and Franck Le
Recap: Internet Architecture

- Packet-switched datagram network
- IP is the “compatibility layer”
  - Hourglass architecture
  - All hosts and routers run IP
- Stateless architecture
  - no per flow state inside network
Routing Design in Operational Networks
A look from the Inside
SIGCOMM 2004

Why are these words highlighted?
Likes/dislikes
- Border routers speak eBGP to external peers
- Run intra-domain protocol inside the domain
In theory, intra-domain is simple?

• Link state vs. Distance Vector

• Run a single protocol within the domain

• Everyone can reach everyone else
A Study of Operational Production Networks

- Obtained anonymized configuration files for 31 active networks (>8,000 configuration files)

- Networks include:
  - 6 Tier-1 and Tier-2 Internet backbone networks
  - 25 enterprise networks
  - Sizes between 10 and 1,200 routers
  - 4 enterprise networks significantly larger than the backbone networks

- Networks created by diverse set of designers and companies
However, huge gap between models and routing in real world.

**Theoretical models**
- Studied routing protocols in isolation

**Routing in real world**
- Complex routing designs
- 10s to 100s protocol instances per network
Network broken up into compartments, each with only 1 to 4 routers
Each compartment has its own AS number
Hub and spoke logical topology
Why? Lots of control over how spokes communicate
Too simplistic, reality is more complex

• Why is there complexity?
  – Scaling (LS for whole network may take a while)
  – Even a single domain is not homogeneous
  – Policies (Access control)
  – Evolution (mergers/acquisitions/splits)
  – Convenience (design templates)
  – Practical effects (router memory)
The Problem of Routing Design

- HostA
- HostB
  - HostC
  - HostD

Routing Protocols:
- iBGP
- eBGP
- OSPF
- EIGRP
- RIP
- ACLs
- Policy
Routing Design

- Selecting routing protocols
- Configuring their boundaries
- Setting the policies that control their interaction
- Adding packet filters, other mechanisms

- Routing design fundamentally establishes the network’s properties

- Topology doesn’t say much about reachability
What do Designers do Today?

• Network designers balance many goals
  • Scalability
  • Resiliency to failure
  • Make it easy to expand network

• Many “rules of thumb” in use
  • Instability results from overloaded routers
  • Too much routing state is bad
  • Use routing boundaries to control spread of change

• Routing Design is currently an art – can we add more science?
How Are Routing Designs Expressed Today?

1. interface Ethernet0
   - ip address 6.2.5.14 255.255.255.128

2. interface Serial1/0.5 point-to-point
   - ip address 6.2.2.85 255.255.255.252

3. ip access-group 143 in

4. frame-relay interface-dlci 28

5. router ospf 64

6. redistribute connected subnets

7. redistribute bgp 64780 metric 1 subnets

8. network 66.251.75.128 0.0.0.127 area 0

9. router bgp 64780

10. redistribute ospf 64 match route-map 8aTzlvBrbaW

11. neighbor 66.253.160.68 remote-as 12762

12. neighbor 66.253.160.68 distribute-list 4 in

access-list 143 deny 1.1.0.0/16
access-list 143 permit any
route-map 8aTzlvBrbaW deny 10
match ip address 4
route-map 8aTzlvBrbaW permit 20
match ip address 7
ip route 10.2.2.1/16 10.2.1.7
Prevalence of Route Redistribution

99.9% networks rely on route redistribution

1. From IGP and local routes to BGP

2. From BGP into IGP (78% of networks with 15+ routers)

3. From IGP into IGP (35% of networks with 15+ routers)
Single Hub

[Diagram of a network with nodes and connections labeled with BGP numbers and other details.]
Dual Hub - Hubs
Need for abstractions!

• How do you infer “intent” from this mess?

• Raw visualizations are useful but don’t tell the full picture
Four key abstractions

• Routing Process Graph
  – each router, process is a node

• Routing Instance Graph
  – Collapse all router processes that are same logical grp

• Route Pathway
  – Track provenance of routes for a given router

• Address Space Structure
  – Plan for address assignment
Router Model
Route Redistribution

Routing policy 1

Routing policy 2

OSPF

BGP

OSPF

Route Selection

Route Table

Router 1
Route Redistribution

**Diagram:**
- **A** (P: 128.2.1/24, Hop-count: 1) connected to **B** via RIP
  - **B** (Type: External 2, Cost: 200) connected to **C** via OSPF

**Configuration of Router B:**
```
... router rip
  ...
! router ospf 200
  redistribute rip metric 100
!```

**Notes:**
- Route Redistribution
- RIP routing protocol
- OSPF routing protocol
- **P:** 128.2.1/24
- **A** to **B** through RIP
- **B** to **C** through OSPF
- Cost: 200
Route Selection

- Router RIP
  - Distance 90
  - AD: 120
- Router OSPF
  - AD: 110

Configuration of Router D

```plaintext
... router rip
distance 90
!
router ospf 200
...
```
Routing Protocol Adjacencies

Router 2

Router 1

Routing policy 1

Routing policy 2

OSPF

Route Table

RS

OSPF

BGP

OSPF

Route Selection

Route Table
Reverse-Engineering Overview

Configuration files

Find links

Construct Layer 3 Topology

Find adjacent routing processes

Construct Routing Process Graph

Condense adjacent routing processes

Construct Routing Instance Graph

OSPF #1

BGP AS1

OSPF #2
Reconstruct the Layer 3 Topology

Router 1 Config

interface Serial1/0.5
ip address 1.1.1.1 255.255.255.252

Router 2 Config

interface Serial2/1.5
ip address 1.1.2 255.255.255.252
Construct the Routing Process Graph

OSPF

Policy1

BGP

Policy2

OSPF

RT

Internet

Route Table

RT

OSPF

RT

OSPF
Abstract to a Routing Instance Graph

- Pick an unassigned Routing Process
- Flood fill along process adjacencies, labeling processes
- Repeat until all processes assigned to an Instance
Why are these abstractions useful?

• Discover properties of routing design in the wild
  – “common language” to describe routing design
  – Expose common patterns in routing designs

• Expose dependencies in how routes are learned
  – Can there be loops/oscillations

• Possible anomalies/unexpected configs
  – Does the intent deviate from config?

• Enable new verification/correctness guidelines
Routing Protocol Instances

% networks with ≤ n instances

Number of routing protocol instances (n)
Myth: Policy Enforced at Edge of Network

- Conventional wisdom:
  - Place packet filters on the edge to defend infrastructure
  - Routing policy applied where networks touch
Reality: Policy Exists Throughout Networks

- Packet filters commonly used on *internal* links
- Protect routers from attack
- Implement reachability matrix
  - Prevent some hosts from communicating with others
  - Localize traffic, particularly multicast
Reachability Example

- Enterprise with two remote offices
- Only A&B should be able to talk to server C
Reachability Example

- Network designers add two links for robustness
- Configure routing protocols to use new links in failure
Reachability Example

- Designers apply packet filters to new links
Reachability Example

- Permit A->C
- Permit B->C
- Permit A->C

A

B

C

Internet
• Packet from B->C dropped!
• Testing under normal conditions won’t find this error!
Route Redistribution & Loops

Configured route redistribution can result in persistent forwarding loops.
Key takeaways

• Real-world routing is more complex
  – Policies
  – Multiple routing protocols
  – Route redistribution, route selection

• Methodology for “white-box” reverse engineering

• Exposing this complexity spawned a lot of follow up work
What would be some interesting follow up?

- Modeling complexity?
- How much of this is fundamental/artifact?
- Tackling this complexity?
- Configuration verification?
- Why are you doing ACL in routing layer?
Summary of Anomalies

• Wider range than BGP and caused by
  – Improper Route Redistribution (RR)
  – Improper assignments of AD values alone
  – Interplays between the two primitives (nondeterministic routing behaviors)

• Difficult to look for them from net configs (NP hard)