
CSCI-1680
Network Layer:
Inter-domain Routing – Policy and Security

Nick DeMarinis

Based partly on lecture notes by Rachit Agarwal, Rodrigo Fonseca, Jennifer Rexford,
Rob Sherwood, David Mazières, Phil Levis, John Jannotti

Warmup for discussion

Given this routing table, to which prefix would a router map each IP?

- 1.2.3.4
- 138.16.100.5
- 138.16.10.200
- 12.34.5.120
- 12.34.18.5

	Prefix	Next Hop
①	1.0.0.0/8	...
②	12.34.0.0/16	...
③	12.34.16.0/20	...
④	138.16.0.0/16	...
⑤	138.16.100.0/24	...

Warmup for discussion


Given this routing table, to which prefix would a router map each IP?

- 1.2.3.4 \Rightarrow ①
- 138.16.100.5 \Rightarrow ⑤
- 138.16.10.200 \Rightarrow ④
- 12.34.5.120 \Rightarrow ②
- 12.34.18.5 \Rightarrow ③

	Prefix	Next Hop
①	1.0.0.0/8	...
②	12.34.0.0/16	...
③	12.34.16.0/20	...
④	138.16.0.0/16	...
⑤	138.16.100.0/24	...

Administrivia

Upcoming deadlines

- HW2: Out later today
 - Next Thursday: HW2 due, Midterm out
 - Next Friday: Midterm due
 - IP deadline moved to Tuesday, March 22
- 

Administrivia

Upcoming deadlines

- HW2: Out later today
- Next Thursday: HW2 due, Midterm out
- Next Friday: Midterm due
- IP deadline moved to Tuesday, March 22

- Want to help rebuild this course? Apply to HTA/UTA in the fall!
 - Also looking for summer hires!

Today

- BGP Continued
 - Policy routing, instability, vulnerabilities

Longest Prefix Match

When performing a forwarding table lookup, select the most specific prefix that matches an address

- Eg. 12.34.18.5

IN BINARY:

0000 1100 0010 0100

TWO POSSIBLE
MATCHES:

0001 0010 0000 0101

① 0000 1100 0010 0100 XXXX XXXX XXXX XXXX

② 0000 1100 0010 0100 0001 XXXX XXXX XXXX

Prefix	Next Hop
1.0.0.0/8	...
12.34.0.0/16	① ...
12.34.16.0/20	② ...
138.16.0.0/16	...
138.16.100.0/24	...

→ MOST SPECIFIC MATCH
WINS ⇒ LONGEST PREFIX MATCH.

Longest Prefix Match

(NOTE: DON'T NEED THIS FOR IP PROJECT, UNLESS DOING CAPSTONE.)
When performing a forwarding table lookup, select the most specific prefix that matches an address

- Eg. 12.34.18.5

COULD ALSO DO
IN SOFTWARE W/
A TREE DATA STRUCTURE

Prefix	Next Hop
1.0.0.0/8	...
12.34.0.0/16	...
12.34.16.0/20	...
138.16.0.0/16	...
138.16.100.0/24	...

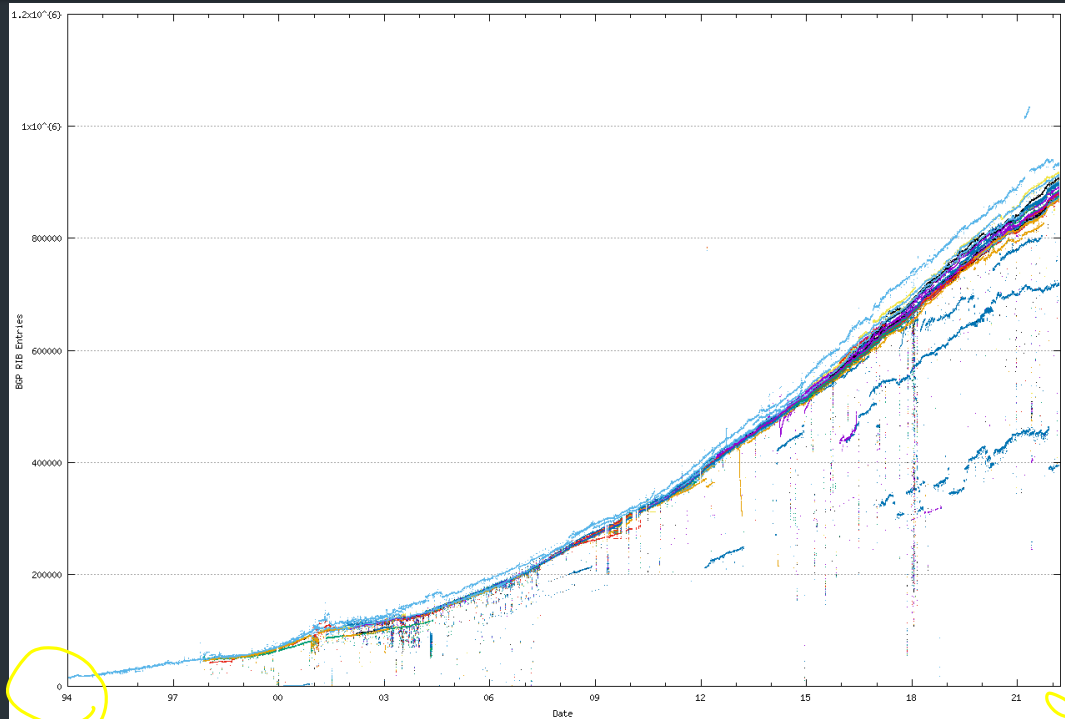
Internet routers have specialized memory called TCAM (Ternary Content Addressable Memory) to do longest prefix match fast (one clock cycle!)

Goal: forward at line rate (as fast as link allows)

Prefixes

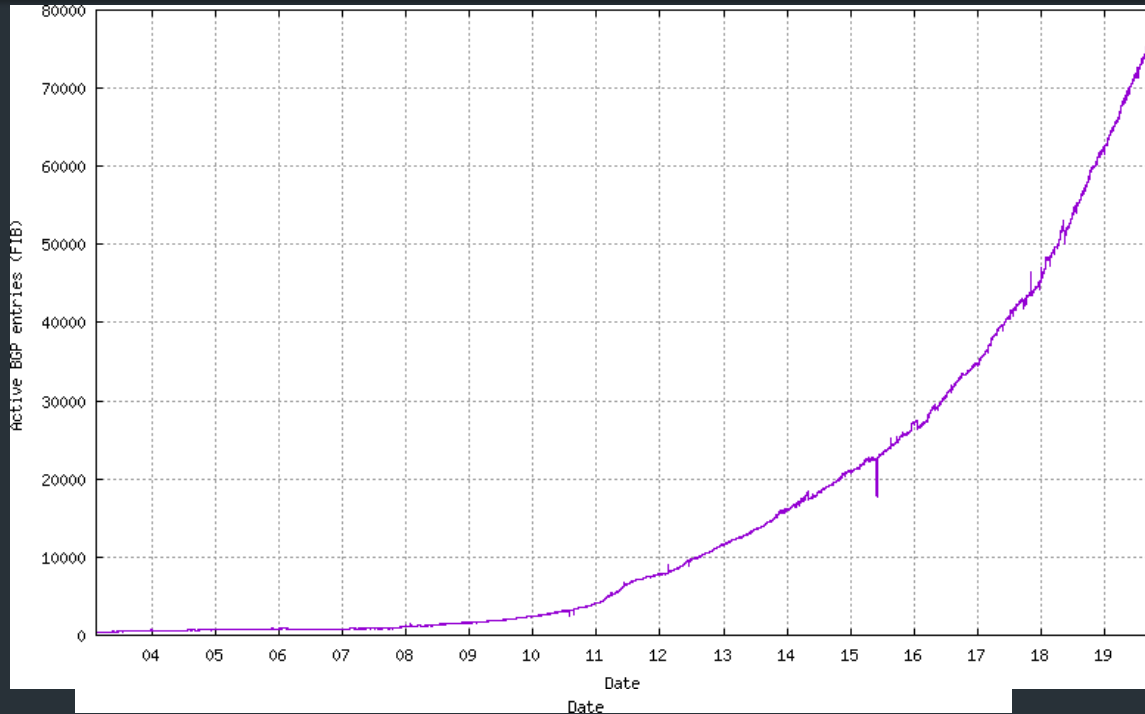
- Nodes in local network share prefix
 - Key to decide whether to send message locally
- Prefixes can also aggregate multiple networks
 - E.g., 100.20.33.128/25, 100.20.33.0/25 -> 100.20.33.0/24
- If networks connected hierarchically, can have significant aggregation
- But allocations aren't so hierarchical... what does this mean?

BGP Table Growth



Source: bgp.potaroo.net

BGP Table Growth for v6



Source: bgp.potaroo.net

512k day

- On August 12, 2014, the full IPv4 BGP table reached 512k prefixes
- Many older routers had only 512k of TCAM, had to fall back to slower routing methods
- Caused outages in Microsoft Azure, ebay, others...

What can lead to table growth?

- More addresses being allocated
- Fragmentation
 - Multihoming
 - Change of ISPs
 - Address re-selling

Recall: BGP mechanics

- Path-vector protocol
- Exchange prefix reachability with neighbors (ASes)
 - E.g., "I can reach prefix 128.148.0.0/16 through ASes 44444 3356 14325 11078"
- Select routes to propagate to neighbors based on routing policy, not shortest-path costs
- **Today: Policies and implications**

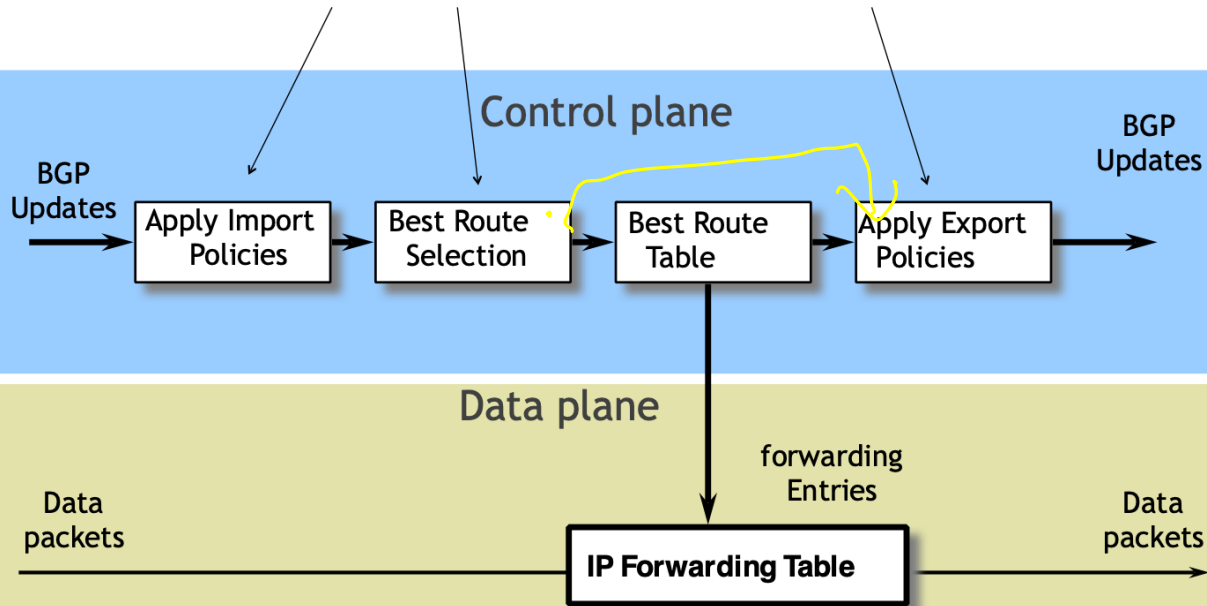
Where do we use policies?

Policies are imposed in how routes are selected and exported

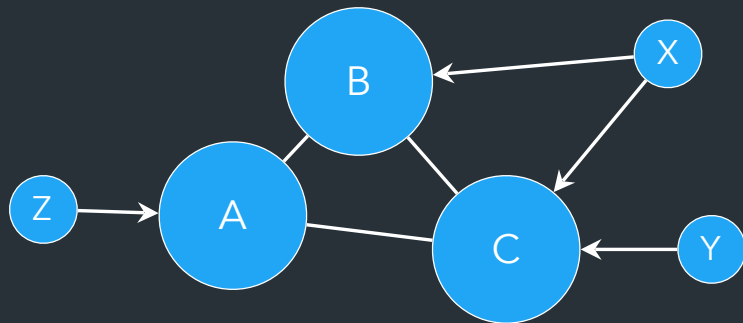
- Selection: which path to use in your network
 - Controls if/how traffic leaves the network
 - Export: which path to advertise
 - Controls how/if traffic enters the network
- (How you route traffic out)
- TELLS NETWORK WHAT TRAFFIC TO SEND YOU

Update processing

*Open ended programming.
Constrained only by vendor configuration language*



AS Relationships

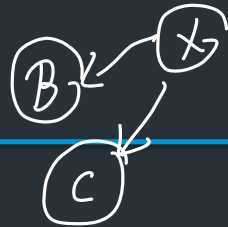


X PAYS B, C
FOR ITS
TRAFFIC

Policies are defined by relationships between ~~ASes~~ ASes

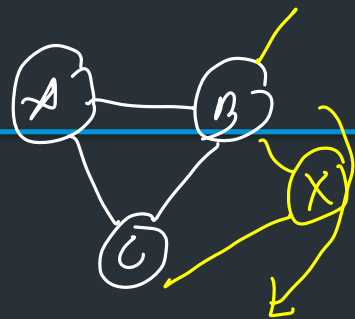
- Provider
- Customer
- Peers

AS relationships



- Customer pays provider for connectivity
 - E.g. Brown contracts with OSHEAN — *SMALL ISP FOR EDUCATION IN RI*
 - Customer is stub, provider is a transit
- Many customers are multi-homed
 - E.g., OSHEAN connects to Level3, Cogent
- Typical policies:
 - Provider tells all neighbors how to reach customer
 - Provider wants to send traffic to customers (\$\$\$) (*WHO PAY FOR IT*)
 - Customer does not provide transit service

Peer Relationships



- Peer ASs agree to exchange traffic for free
 - Penalties/Renegotiate if imbalance
- Tier 1 ISPs have no default route: all peer with each other
- You are Tier $i + 1$ if you have a default route to a Tier i
- Typical policies
 - AS only exports customer routes to peer
 - AS exports a peer's routes only to its customers
 - Goal: avoid being transit when no gain

↳ YOU'RE NOT GETTING PAID FOR IT!
(NOT YOUR PROBLEM)

Typical route selection policy

In decreasing priority order:

PAYS YOU !!

1. Make or save **money** (send to customer > peer > provider)

← YOU PAY THEM !!

2. Try to maximize **performance** (smallest AS path length)

3. Minimize use of my **network bandwidth** ("hot potato routing")

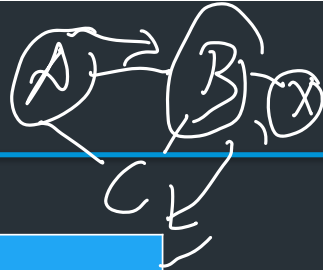
4. ...

IF YOU CAN PASS TRAFFIC TO ANOTHER NETWORK, DO SO - THIS SAVES YOUR BANDWIDTH.

Gao-Rexford Model

- (simplified) Two types of relationships: peers and customer/provider
- Export rules:
 - Customer route may be exported to all neighbors
 - Peer or provider route is only exported to customers
- Preference rules:
 - Prefer routes through customer (\$\$)
- If all ASes follow this, shown to lead to stable network

Typical Export Policy



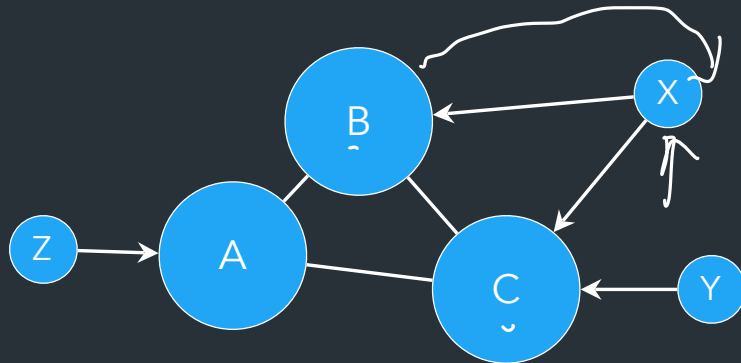
Destination prefix advertised by...	Export route to...
<u>Customer</u>	<u>Everyone</u> (providers, peers, other customers...)
<u>Peer</u>	Customers only
Provider	Customers only

*NOT OTHER PEERS,
OR PROVIDERS.
OTHERWISE,*

*YOUR AS WILL BE TRANSIT FOR A
PEER OR PROVIDER!*

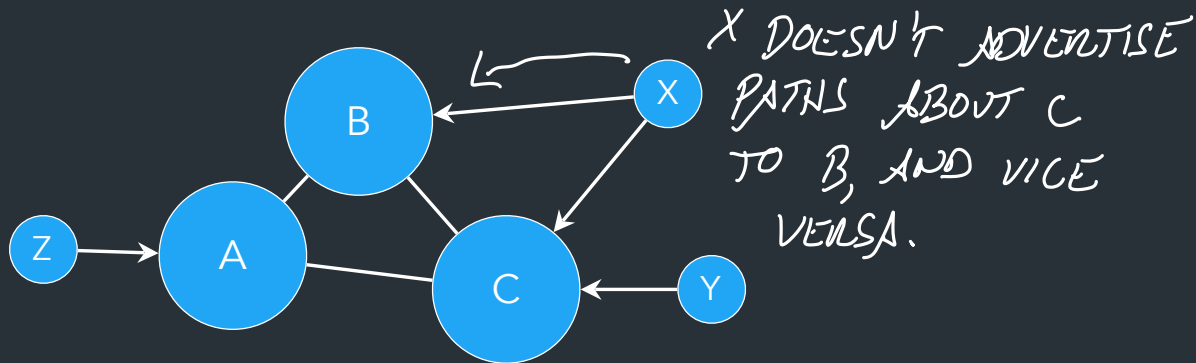
Known as Gao-Rexford principles: define common practices for AS relationships *(SHOWN TO CREATE STABLE RELATIONSHIPS)*

AS Relationships



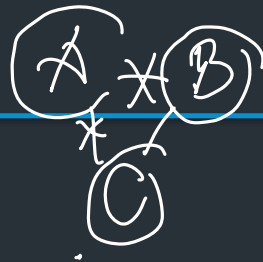
- How to prevent X from forwarding transit between B and C?

AS Relationships



- How to prevent X from forwarding transit between B and C?
- How to avoid transit between CBA ?
 - B: BAZ \rightarrow X
 - B: BAZ \rightarrow C ? (\Rightarrow Y: CBAZ and Y:CAZ)

Peering Drama



- Cogent vs. Level3 were peers
- In 2003, Level3 decided to start charging Cogent
- Cogent said no
- **Internet partition**: Cogent's customers couldn't get to Level3's customers and vice-versa
 - Other ISPs were affected as well
- Took 3 weeks to reach an undisclosed agreement

BGP can be fragile

- Individual router configurations and policy can affect whole network
- Consequences sometimes disastrous...

Some BGP Challenges

- Convergence \Rightarrow MINUTES, OR LONGER
- Traffic engineering
 - How to assure certain routes are selected
- Misconfiguration
- Security

BGP can be fragile! One router configuration can affect a large portion of the network

Recent Notable incidents

- October 4 2021: Facebook accidentally removed routes for its DNS servers

— Outside world couldn't resolve facebook.com, and neither could Facebook!

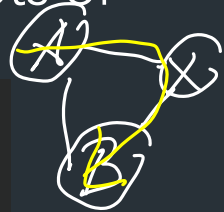
→ 1.2.3.4

- June 24, 2019: Misconfigured ^{SMALL CUSTOMER ROUTER} router accepted lots of transit traffic

Jérôme Fleury

[URGENT] Route-leak from your customer

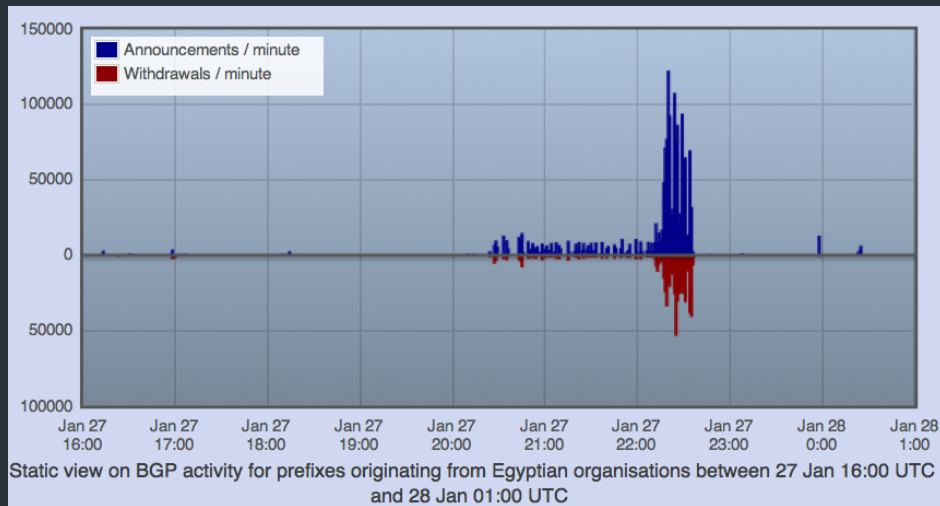
To: CaryNMC-IP@one.verizon.com, peering@verizon.com, help4u@verizon.com,



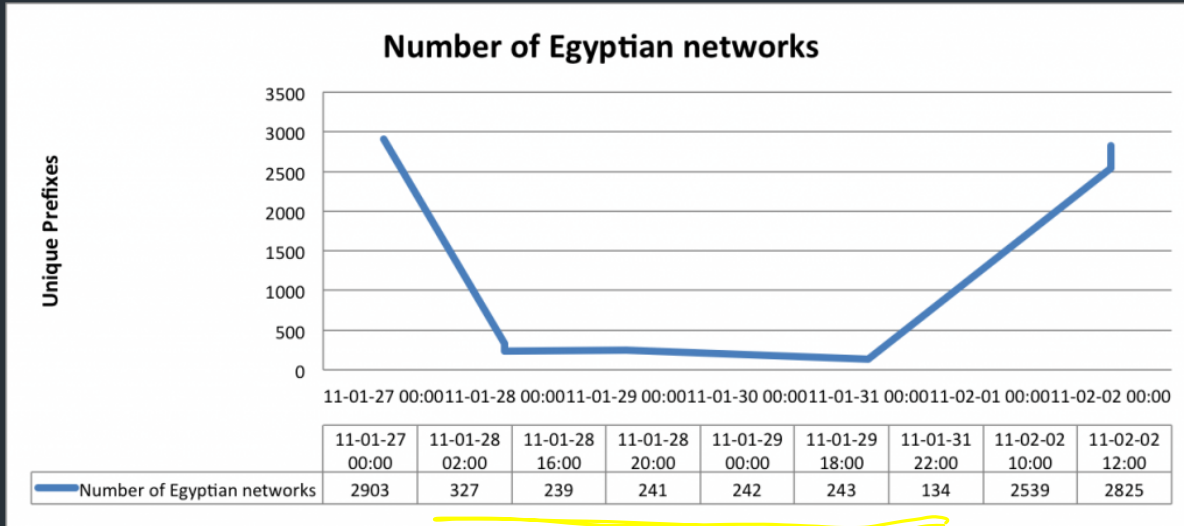
→ HOW TO SOLVE? SYSADMINS NEEDED TO COORDINATE TO FIND + FIX PROBLEMS (HUMAN INTERVENTION!)

"Shutting off" the Internet

- Starting from Jan 27th, 2011, Egypt was disconnected from the Internet
 - 2769/2903 networks withdrawn from BGP (95%)!



Egypt Incident



BGP Security Goals

- Confidential message exchange between neighbors
- Validity of routing information
 - Origin, Path, Policy
- Correspondence to the data path

→ A BGP SPEAKER CAN LIE ABOUT
PREFIXES, PATHS

Origin: IP Address Ownership and Hijacking

- IP address block assignment
 - Regional Internet Registries (ARIN, RIPE, APNIC)
 - Internet Service Providers
- Proper origination of a prefix into BGP
 - By the AS who owns the prefix
 - ... or, by its upstream provider(s) in its behalf
- However, what's to stop someone else?
 - Prefix hijacking: another AS originates the prefix
 - BGP does not verify that the AS is authorized
 - Registries of prefix ownership are inaccurate

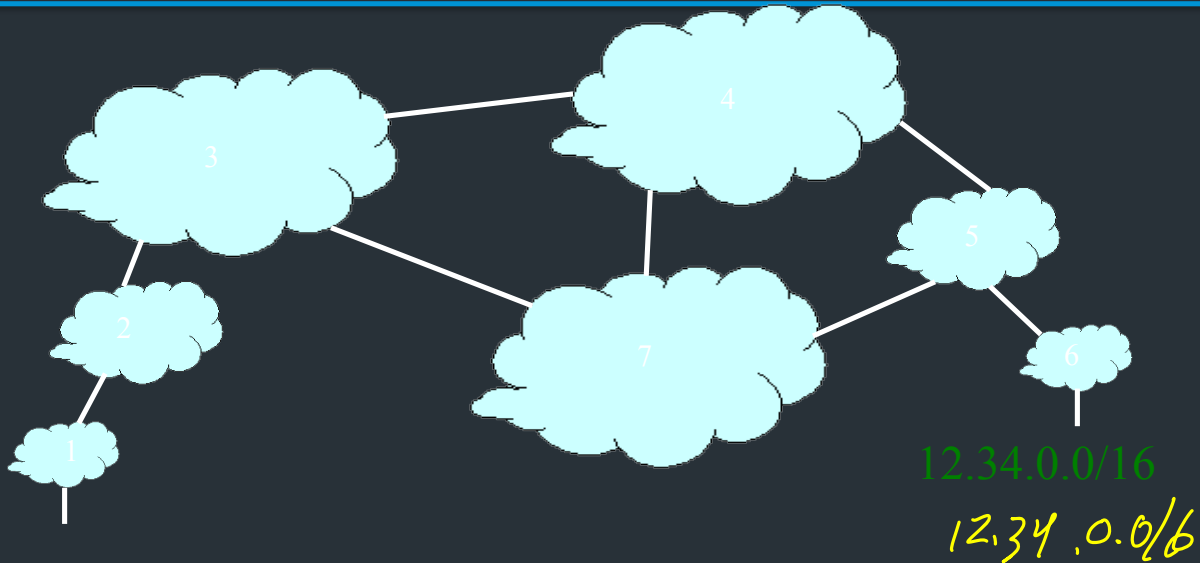
138.16.0.0/16

BROWN → 138.16.0.0/16

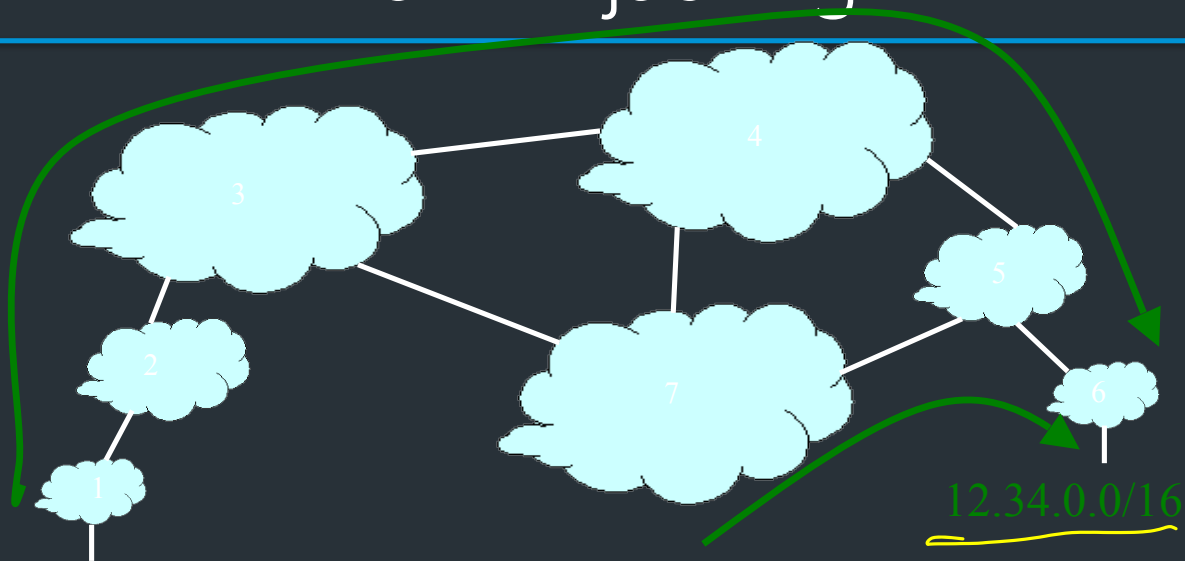
↳

→ BY DEFAULT,
NO VALIDATION.

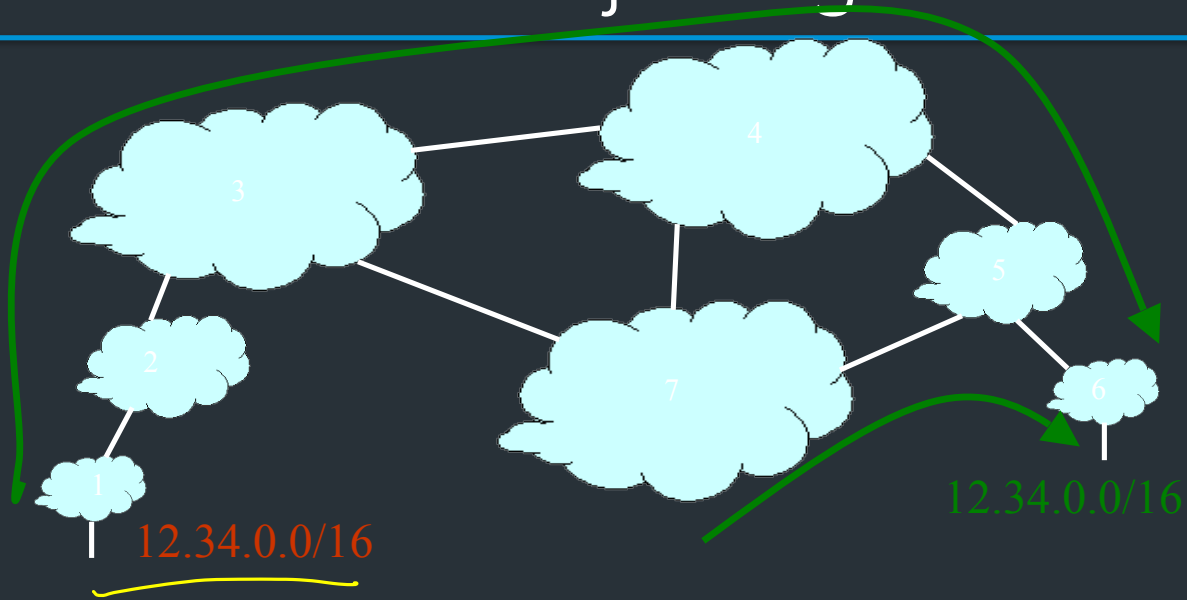
Prefix Hijacking



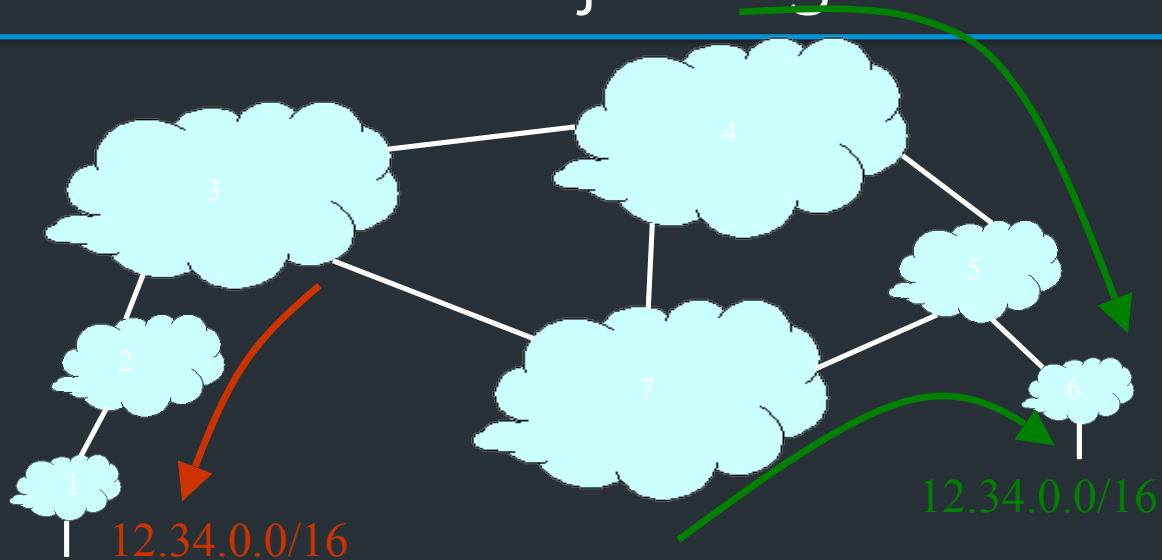
Prefix Hijacking



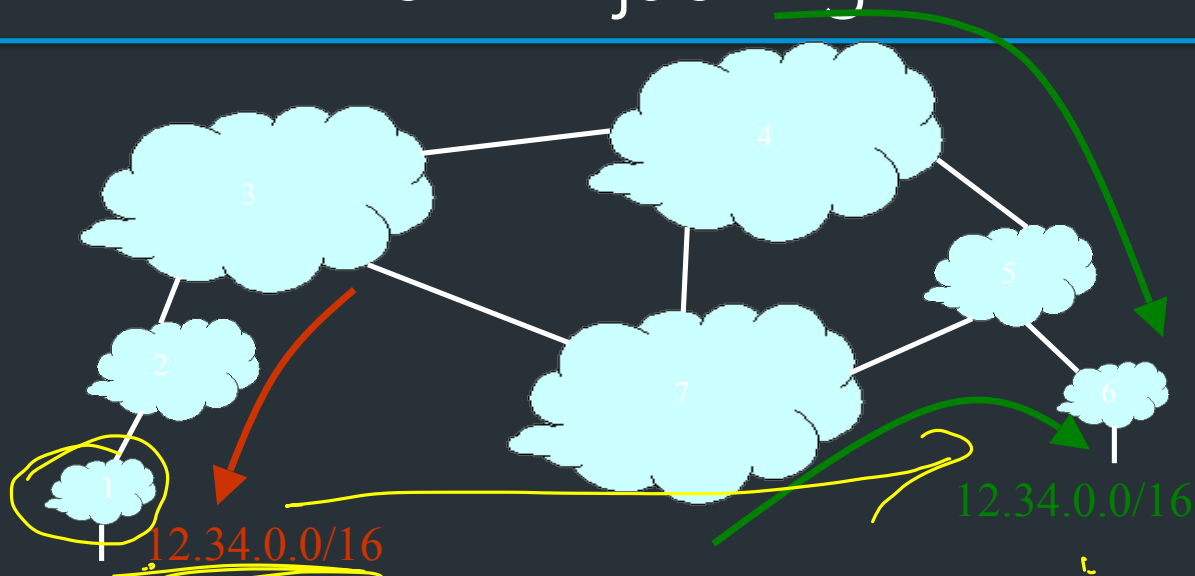
Prefix Hijacking



Prefix Hijacking



Prefix Hijacking



- Consequences for the affected ASes
 - Blackhole: data traffic is discarded
 - Snooping: data traffic is inspected, and then redirected
 - Impersonation: data traffic is sent to bogus destinations

Hijacking is Hard to Debug

- Real origin AS doesn't see the problem

Hijacking is Hard to Debug

- Real origin AS doesn't see the problem
 - Picks its own route

Hijacking is Hard to Debug

- Real origin AS doesn't see the problem
 - Picks its own route
 - Might not even learn the bogus route

Hijacking is Hard to Debug

- Real origin AS doesn't see the problem
 - Picks its own route
 - Might not even learn the bogus route
- May not cause loss of connectivity

Hijacking is Hard to Debug

- Real origin AS doesn't see the problem
 - Picks its own route
 - Might not even learn the bogus route
- May not cause loss of connectivity
 - E.g., if the bogus AS snoops and redirects

Hijacking is Hard to Debug

- Real origin AS doesn't see the problem
 - Picks its own route
 - Might not even learn the bogus route
- May not cause loss of connectivity
 - E.g., if the bogus AS snoops and redirects
 - ... may only cause performance degradation

Hijacking is Hard to Debug

- Real origin AS doesn't see the problem
 - Picks its own route
 - Might not even learn the bogus route
- May not cause loss of connectivity
 - E.g., if the bogus AS snoops and redirects
 - ... may only cause performance degradation
- Or, loss of connectivity is isolated

Hijacking is Hard to Debug

- Real origin AS doesn't see the problem
 - Picks its own route
 - Might not even learn the bogus route
- May not cause loss of connectivity
 - E.g., if the bogus AS snoops and redirects
 - ... may only cause performance degradation
- Or, loss of connectivity is isolated
 - E.g., only for sources in parts of the Internet

Hijacking is Hard to Debug

- Real origin AS doesn't see the problem
 - Picks its own route
 - Might not even learn the bogus route
- May not cause loss of connectivity
 - E.g., if the bogus AS snoops and redirects
 - ... may only cause performance degradation
- Or, loss of connectivity is isolated
 - E.g., only for sources in parts of the Internet
- Diagnosing prefix hijacking

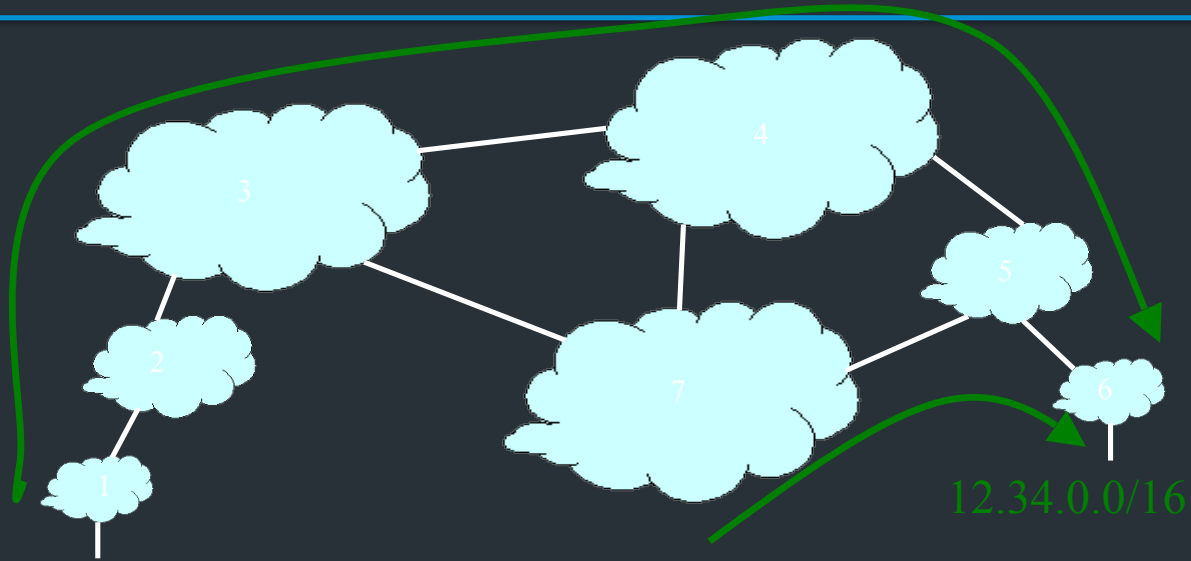
Hijacking is Hard to Debug

- Real origin AS doesn't see the problem
 - Picks its own route
 - Might not even learn the bogus route
- May not cause loss of connectivity
 - E.g., if the bogus AS snoops and redirects
 - ... may only cause performance degradation
- Or, loss of connectivity is isolated
 - E.g., only for sources in parts of the Internet
- Diagnosing prefix hijacking
 - Analyzing updates from many vantage points

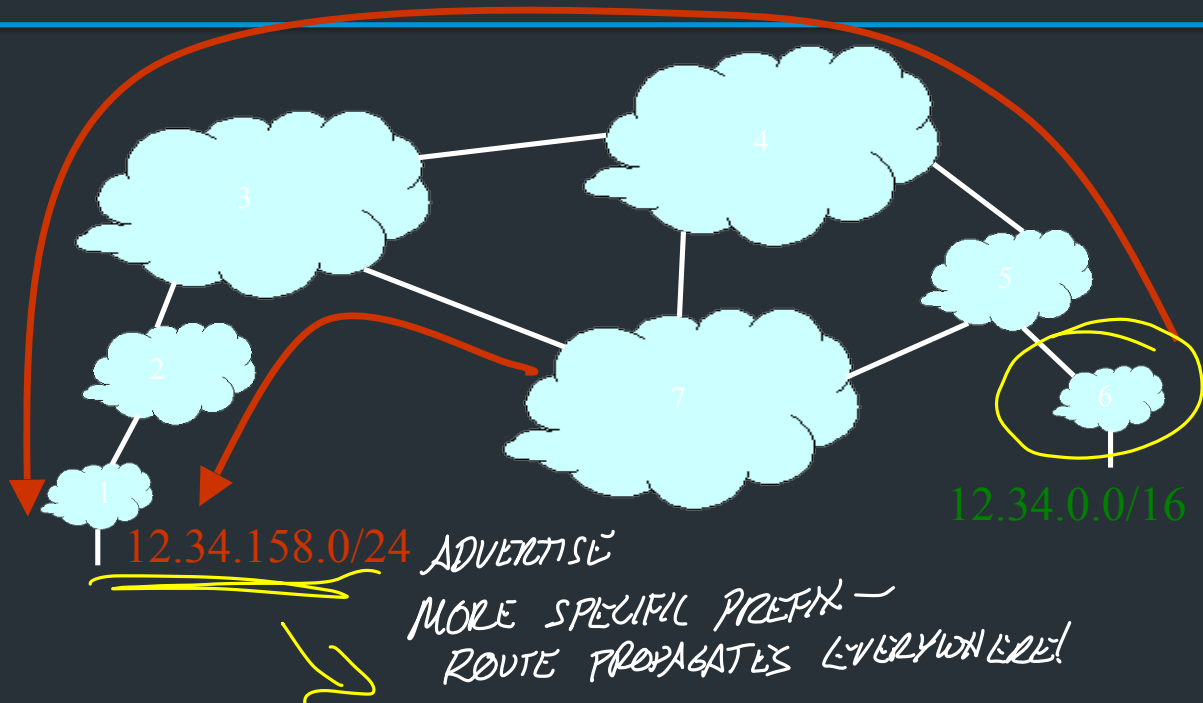
Hijacking is Hard to Debug

- Real origin AS doesn't see the problem
 - Picks its own route
 - Might not even learn the bogus route
- May not cause loss of connectivity
 - E.g., if the bogus AS snoops and redirects
 - ... may only cause performance degradation
- Or, loss of connectivity is isolated
 - E.g., only for sources in parts of the Internet
- Diagnosing prefix hijacking
 - Analyzing updates from many vantage points
 - Launching traceroute from many vantage points

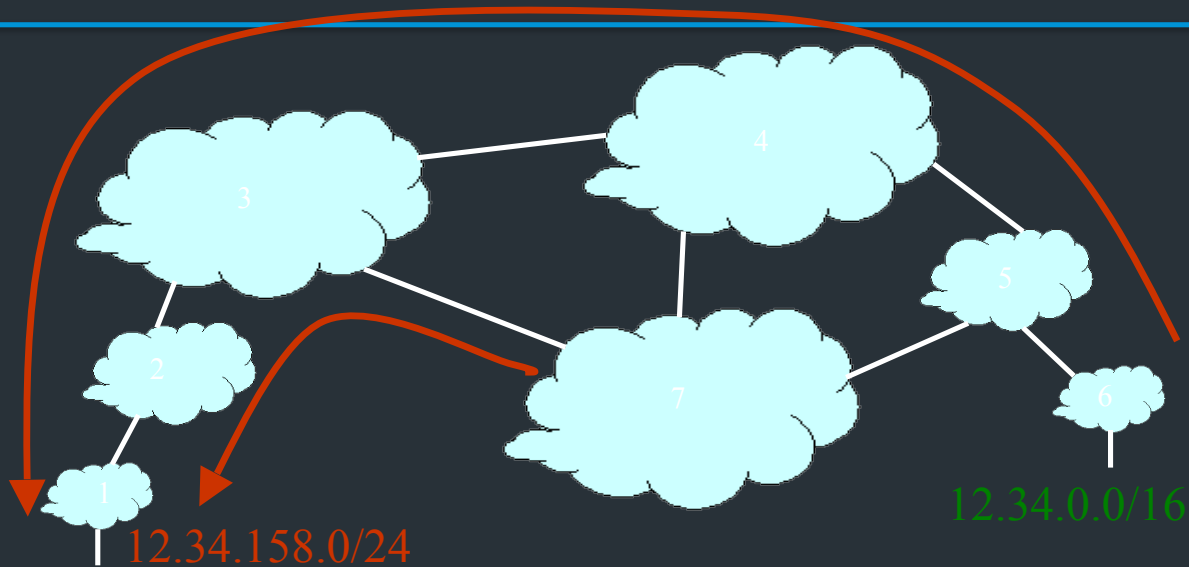
Sub-Prefix Hijacking



Sub-Prefix Hijacking



Sub-Prefix Hijacking



- Originating a more-specific prefix
 - Every AS picks the bogus route for that prefix
 - Traffic follows the longest matching prefix

How to Hijack a Prefix

- The hijacking AS has
 - Router with eBGP session(s)
 - Configured to originate the prefix
- Getting access to the router
 - Network operator makes configuration mistake
 - Disgruntled operator launches an attack
 - Outsider breaks into the router and reconfigures
- Getting other ASes to believe bogus route
 - Neighbor ASes not filtering the routes ←
 - ... e.g., by allowing only expected prefixes
 - But, specifying filters on peering links is hard

Pakistan Youtube incident

- Youtube's has prefix 208.65.152.0/22
- Pakistan's government order Youtube blocked
- Pakistan Telecom (AS 17557) announces 208.65.153.0/24 in the wrong direction (outwards!)
- Longest prefix match caused worldwide outage
- <http://www.youtube.com/watch?v=IzLPKuAOe50>

↘ "BLACKHOLE"

Many other incidents

- Spammers steal unused IP space to hide
 - Announce very short prefixes (e.g., /8). Why?
 - For a short amount of time
- China incident, April 8th 2010
 - China Telecom's AS23724 generally announces 40 prefixes
 - On April 8th, announced ~37,000 prefixes
 - About 10% leaked outside of China
 - Suddenly, going to www.dell.com might have you routing through AS23724!

Attacks on BGP Paths

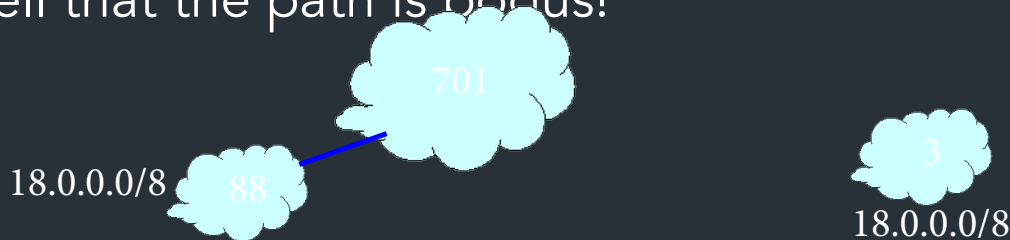
- Remove an AS from the path
 - E.g., 701 3715 88 -> 701 88
- Why?
 - Attract sources that would normally avoid AS 3715
 - Make path through you look more attractive
 - Make AS 88 look like it is closer to the core
 - Can fool loop detection!
- May be hard to tell whether this is a lie
 - 88 could indeed connect directly to 701!

Attacks on BGP Paths

- Adding ASes to the path
 - E.g., 701 88 -> 701 3715 88
- Why?
 - Trigger loop detection in AS 3715
 - This would block unwanted traffic from AS 3715!
 - Make your AS look more connected
- Who can tell this is a lie?
 - AS 3715 could, if it could see the route
 - AS 88 could, but would it really care?

Attacks on BGP Paths

- Adding ASes at the end of the path
 - E.g., 701 88 into 701 88 3
- Why?
 - Evade detection for a bogus route (if added AS is legitimate owner of a prefix)
- Hard to tell that the path is bogus!



Proposed Solution: S-BGP

- Based on a public key infrastructure
- Address attestations
 - Claims the right to originate a prefix
 - Signed and distributed out of band
 - Checked through delegation chain from ICANN
- Route attestations
 - Attribute in BGP update message
 - Signed by each AS as route along path
- S-BGP can avoid
 - Prefix hijacking
 - Addition, removal, or reordering of intermediate ASes

S-BGP Deployment

- Very challenging
 - PKI (RPKI)
 - Accurate address registries
 - Need to perform cryptographic operations on all path operations
 - Flag day almost impossible
 - Incremental deployment offers little incentive
- But there is hope! [Goldberg et al, 2011]
 - Road to incremental deployment
 - Change rules to break ties for secure paths
 - If a few top Tier-1 ISPs
 - Plus their respective stub clients deploy simplified version (just sign, not validate)
 - Gains in traffic => \$ => adoption!

FAILURE

Your ISP (Verizon, AS701) does not implement BGP safely. It should be using RPKI to protect the Internet from BGP hijacks. [Tweet this →](#)

▼ Details

```
fetch https://valid.rpki.cloudflare.com
```

✓ correctly accepted valid prefixes

```
fetch https://invalid.rpki.cloudflare.com
```

✗ incorrectly accepted invalid prefixes

Data Plane Attacks

- Routers/ASes can advertise one route, but not necessarily follow it!
- May drop packets
 - Or a fraction of packets
 - What if you just slow down some traffic?
- Can send packets in a different direction
 - Impersonation attack
 - Snooping attack
- How to detect?
 - Congestion or an attack?
 - Can let ping/traceroute packets go through
 - End-to-end checks?
- Harder to pull off, as you need control of a router

BGP Recap

- Key protocol that holds Internet routing together
- Path Vector Protocol among Autonomous Systems
- Policy, feasibility first; non-optimal routes
- Important security problems

Next Class

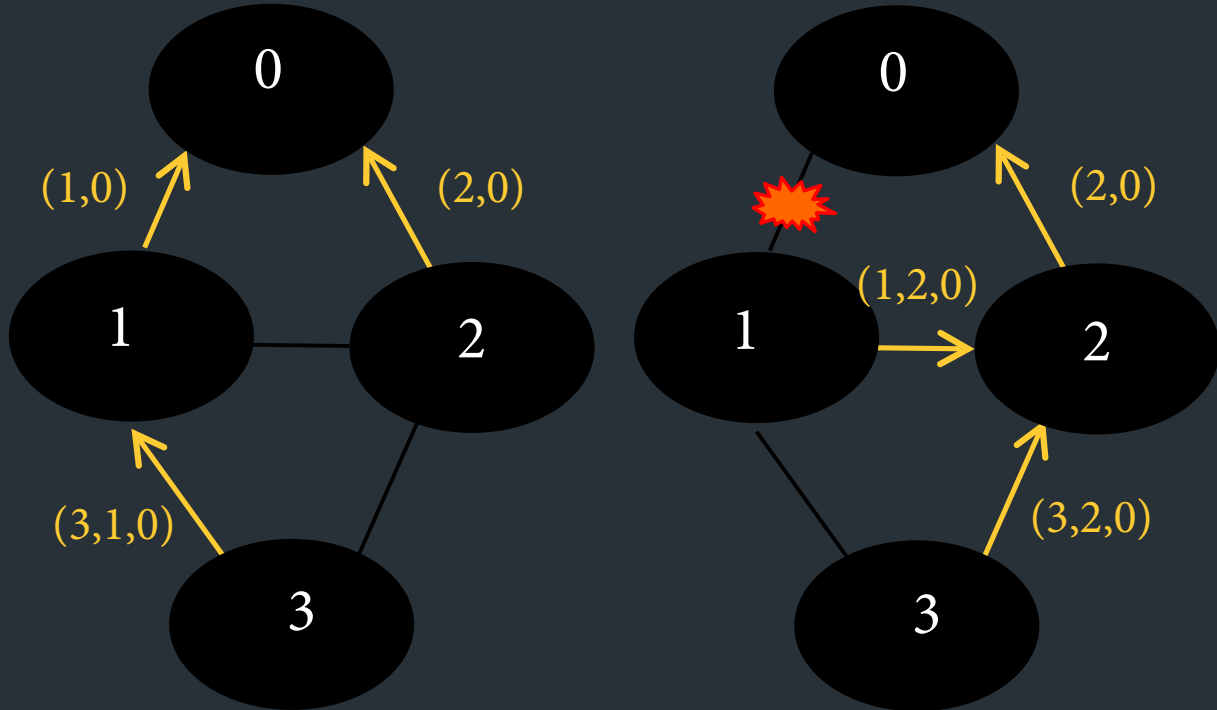
- Network layer wrap up

Following slides not covered, but
interesting

Convergence

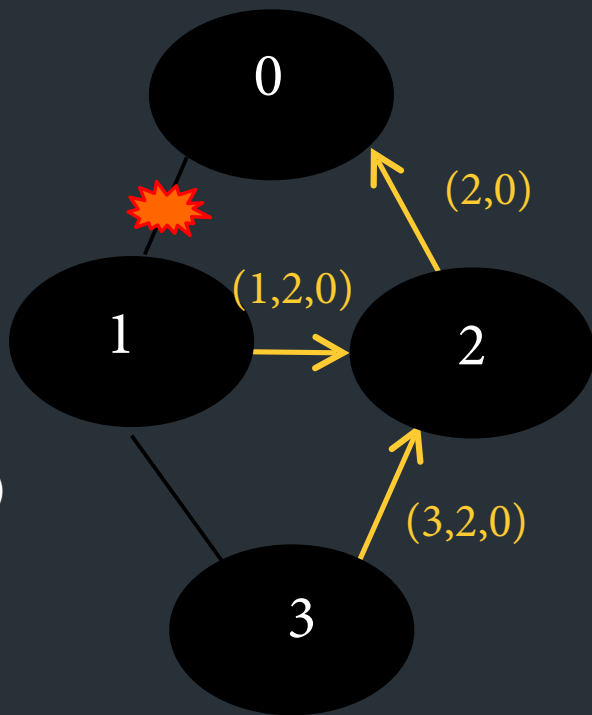
- Given a change, how long until the network re-stabilizes?
 - Depends on change: sometimes never
 - Open research problem: “tweak and pray”
 - Distributed setting is challenging
- Some reasons for change
 - Topology changes
 - BGP session failures
 - Changes in policy
 - Conflicts between policies can cause oscillation

Routing Change: Before and After

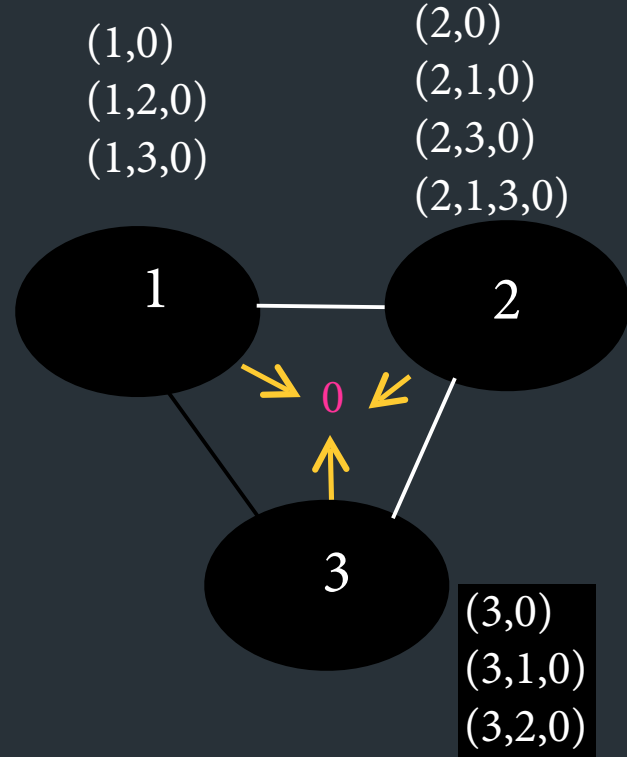


Routing Change: Path Exploration

- AS 1
 - Delete the route (1,0)
 - Switch to next route (1,2,0)
 - Send route (1,2,0) to AS 3
- AS 3
 - Sees (1,2,0) replace (1,0)
 - Compares to route (2,0)
 - Switches to using AS 2

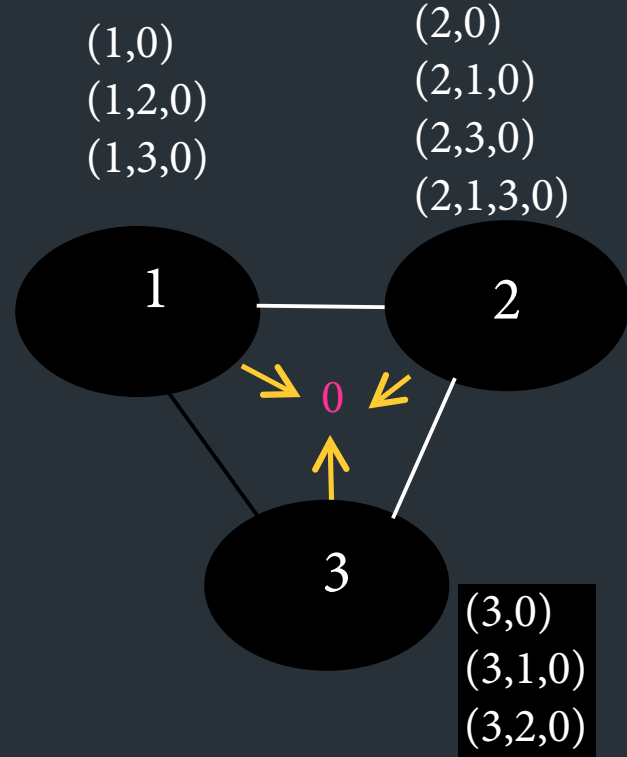


Routing Change: Path Exploration



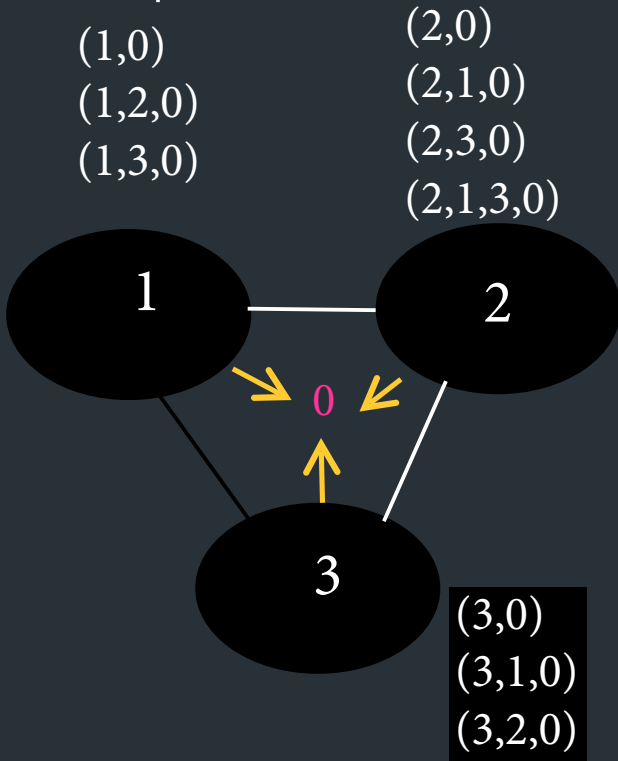
Routing Change: Path Exploration

- Initial situation
 - Destination 0 is alive
 - All ASes use direct path



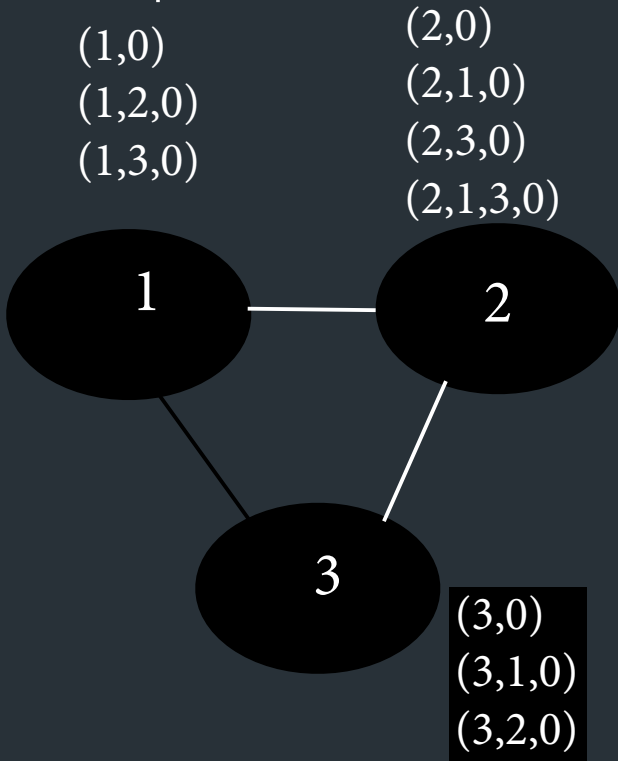
Routing Change: Path Exploration

- Initial situation
 - Destination 0 is alive
 - All ASes use direct path
- When destination dies
 - All ASes lose direct path
 - All switch to longer paths
 - Eventually withdrawn



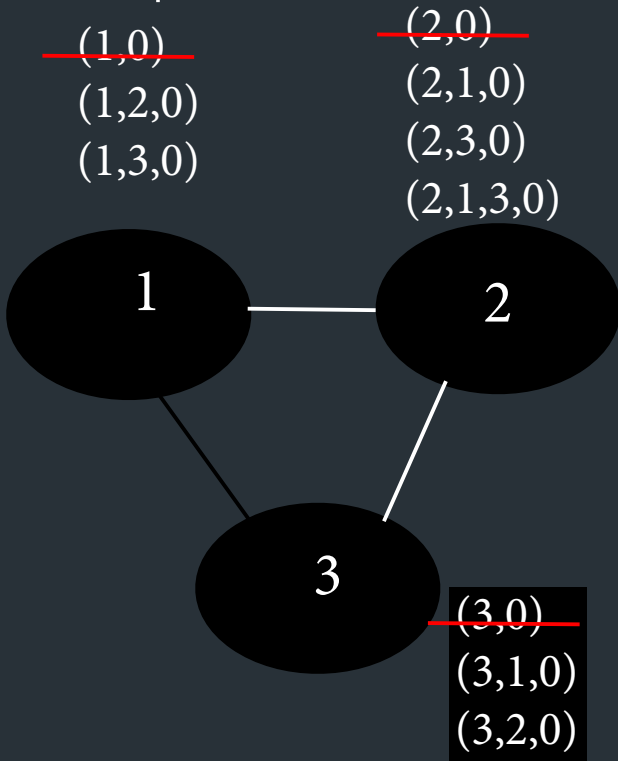
Routing Change: Path Exploration

- Initial situation
 - Destination 0 is alive
 - All ASes use direct path
- When destination dies
 - All ASes lose direct path
 - All switch to longer paths
 - Eventually withdrawn



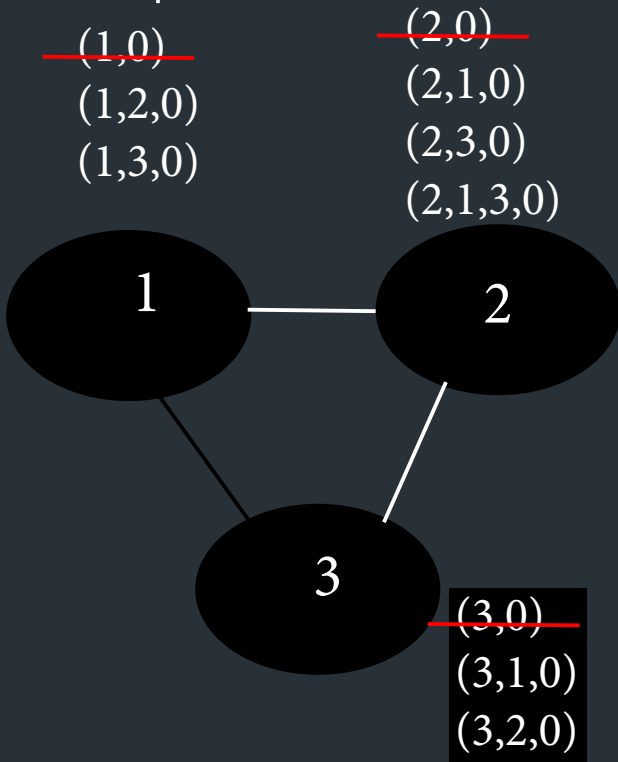
Routing Change: Path Exploration

- Initial situation
 - Destination 0 is alive
 - All ASes use direct path
- When destination dies
 - All ASes lose direct path
 - All switch to longer paths
 - Eventually withdrawn



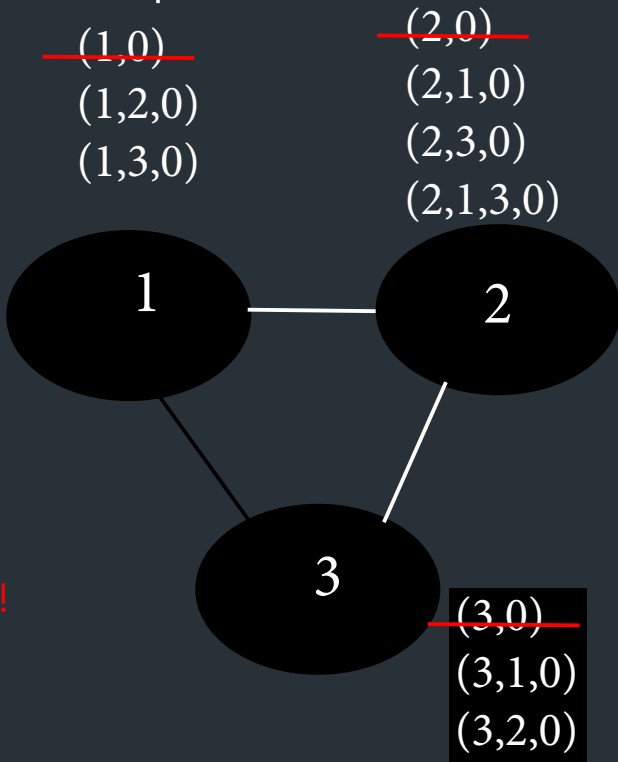
Routing Change: Path Exploration

- Initial situation
 - Destination 0 is alive
 - All ASes use direct path
- When destination dies
 - All ASes lose direct path
 - All switch to longer paths
 - Eventually withdrawn
- E.g., AS 2
 - $(2,0) \rightarrow (2,1,0)$
 - $(2,1,0) \rightarrow (2,3,0)$
 - $(2,3,0) \rightarrow (2,1,3,0)$
 - $(2,1,3,0) \rightarrow \text{null}$



Routing Change: Path Exploration

- Initial situation
 - Destination 0 is alive
 - All ASes use direct path
- When destination dies
 - All ASes lose direct path
 - All switch to longer paths
 - Eventually withdrawn
- E.g., AS 2
 - $(2,0) \rightarrow (2,1,0)$
 - $(2,1,0) \rightarrow (2,3,0)$
 - $(2,3,0) \rightarrow (2,1,3,0)$
 - $(2,1,3,0) \rightarrow \text{null}$
- Convergence may be slow!

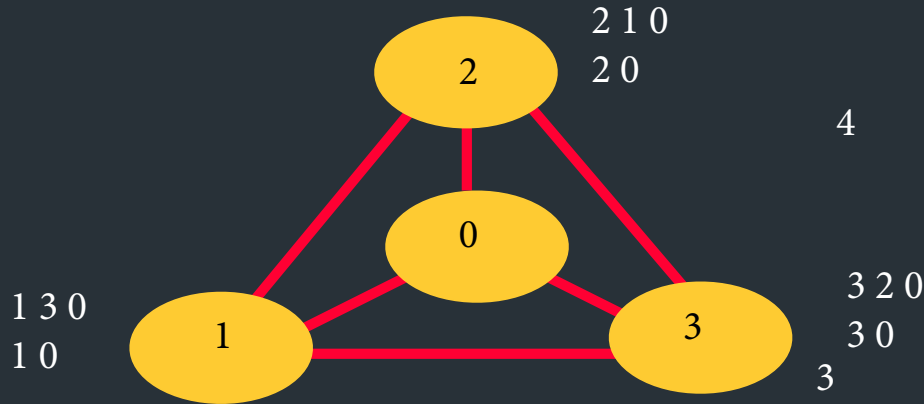


Route Engineering

- Route filtering
- Setting weights
- More specific routes: longest prefix
- AS prepending: "477 477 477 477"
- More of an art than science

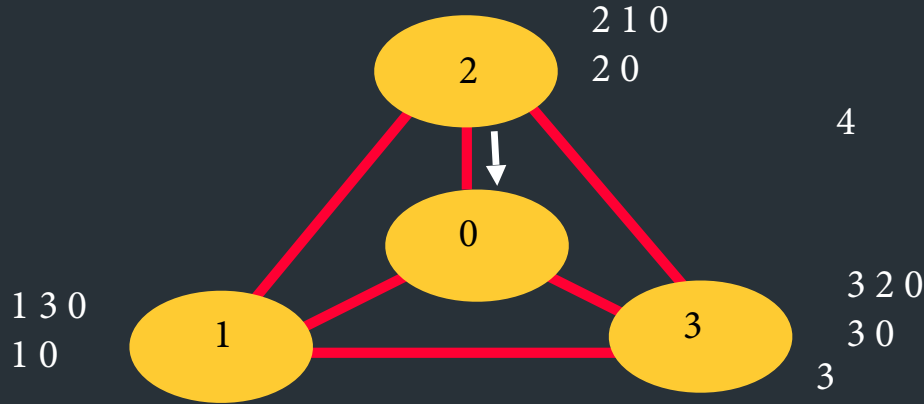
Unstable Configurations

- Due to policy conflicts (Dispute Wheel)



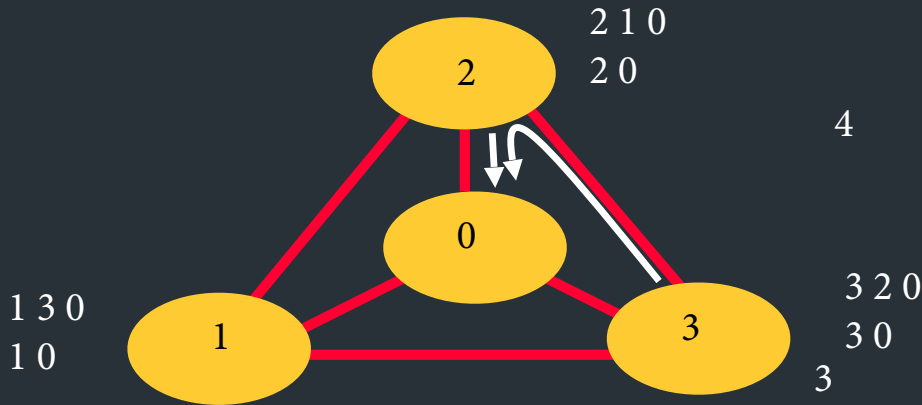
Unstable Configurations

- Due to policy conflicts (Dispute Wheel)



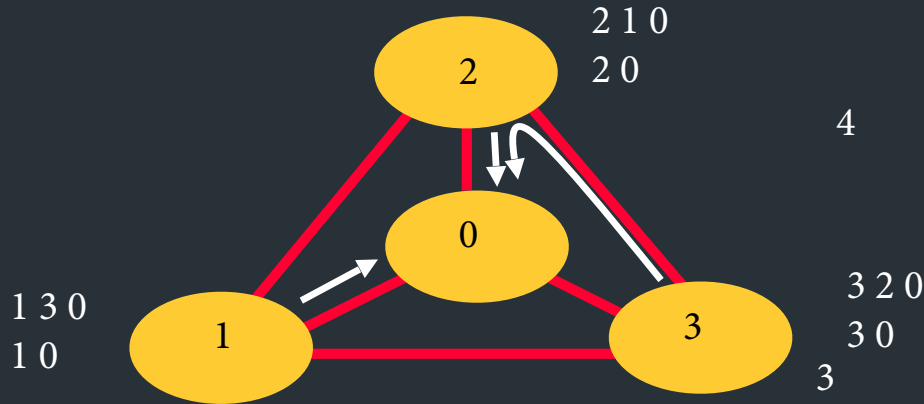
Unstable Configurations

- Due to policy conflicts (Dispute Wheel)



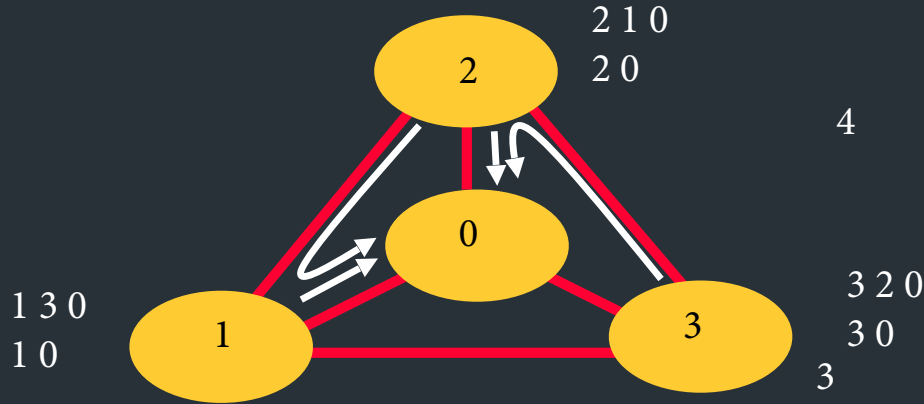
Unstable Configurations

- Due to policy conflicts (Dispute Wheel)



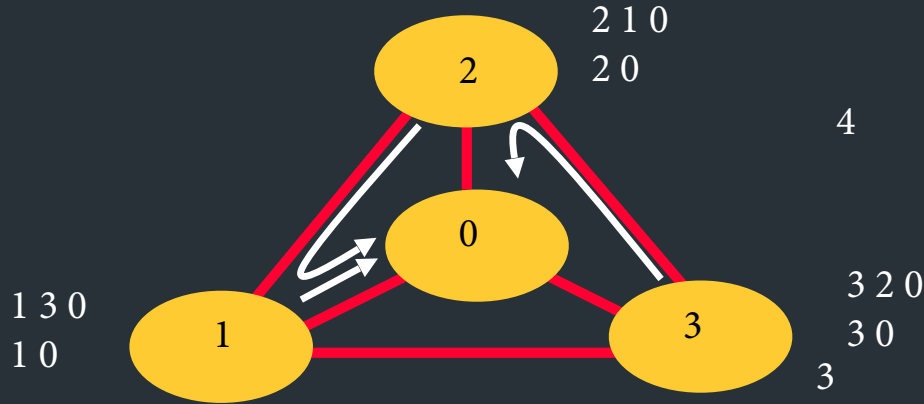
Unstable Configurations

- Due to policy conflicts (Dispute Wheel)



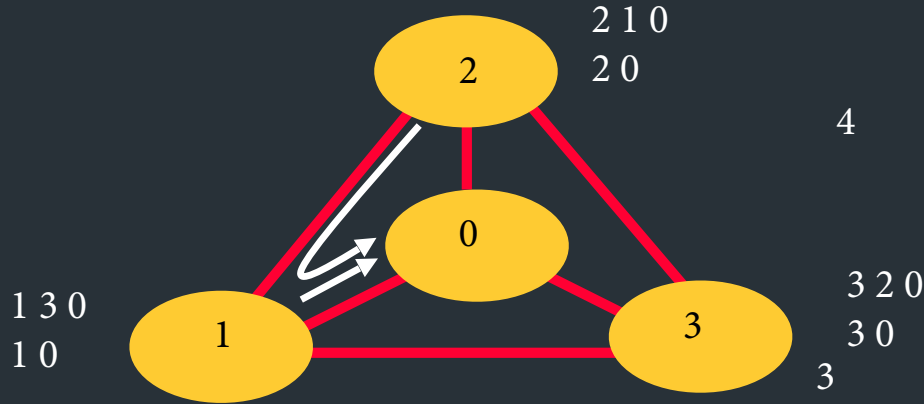
Unstable Configurations

- Due to policy conflicts (Dispute Wheel)



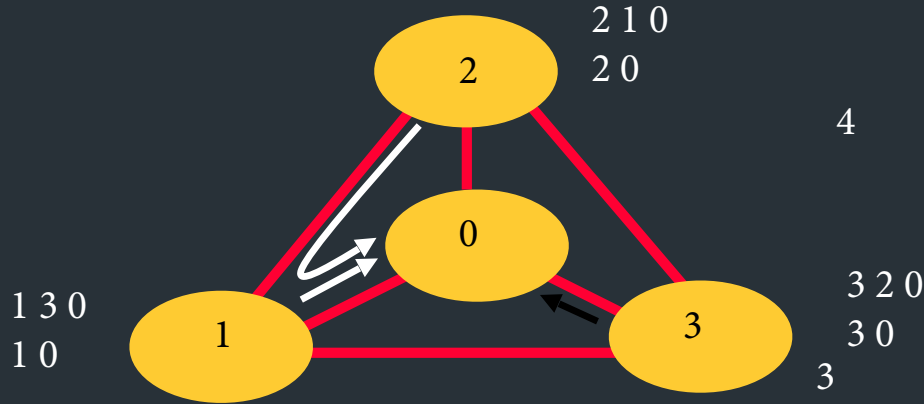
Unstable Configurations

- Due to policy conflicts (Dispute Wheel)



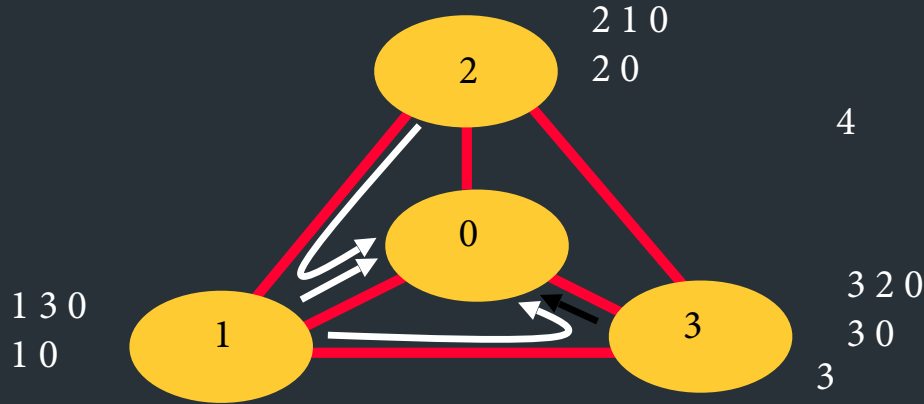
Unstable Configurations

- Due to policy conflicts (Dispute Wheel)



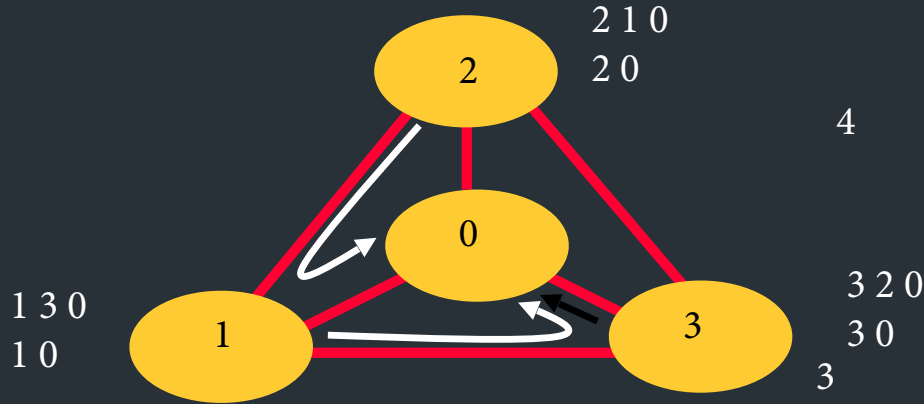
Unstable Configurations

- Due to policy conflicts (Dispute Wheel)



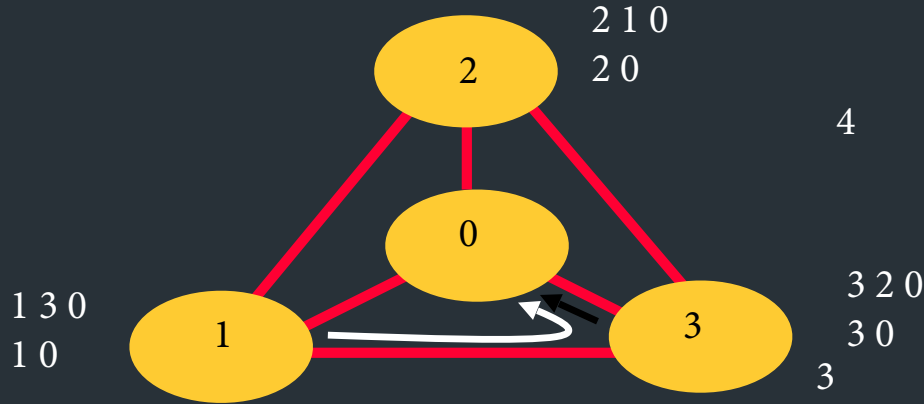
Unstable Configurations

- Due to policy conflicts (Dispute Wheel)



Unstable Configurations

- Due to policy conflicts (Dispute Wheel)

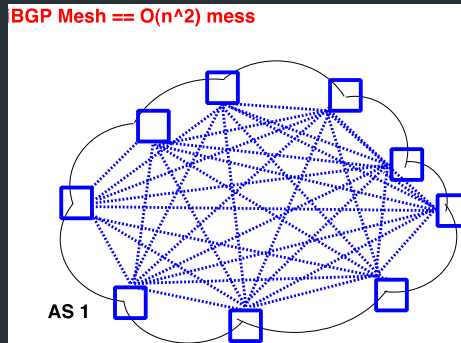


Avoiding BGP Instabilities

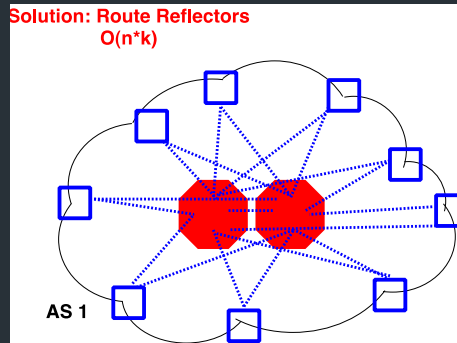
- Detecting conflicting policies
 - Centralized: NP-Complete problem!
 - Distributed: open research problem
 - Requires too much cooperation
- Detecting oscillations
 - Monitoring for repetitive BGP messages
- Restricted routing policies and topologies
 - Some topologies / policies proven to be safe*

* Gao & Rexford, "Stable Internet Routing without Global Coordination", IEEE/ACM ToN, 2001

Scaling iBGP: route reflectors



Scaling iBGP: route reflectors

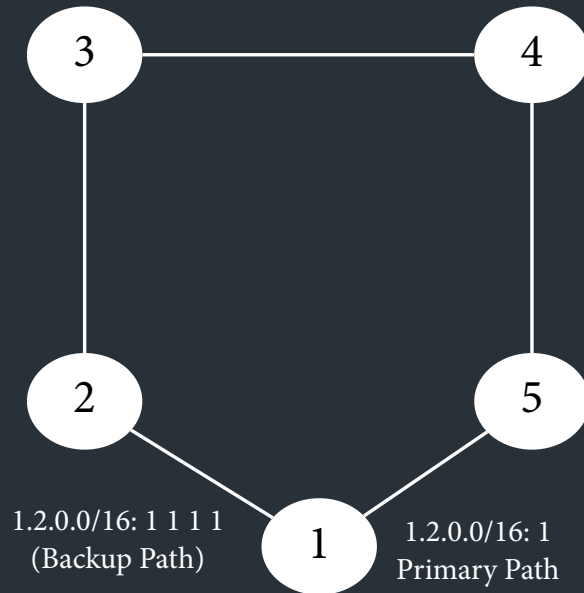


Multiple Stable Configurations

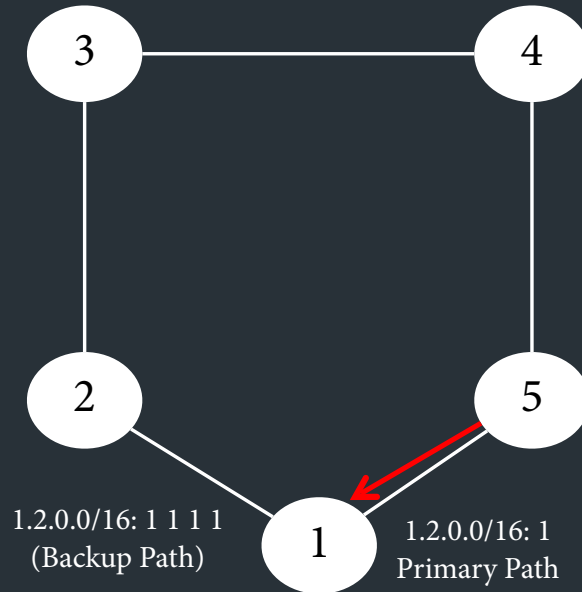
BGP Wedgies [RFC 4264]

- Typical policy:
 - Prefer routes from customers
 - Then prefer shortest paths

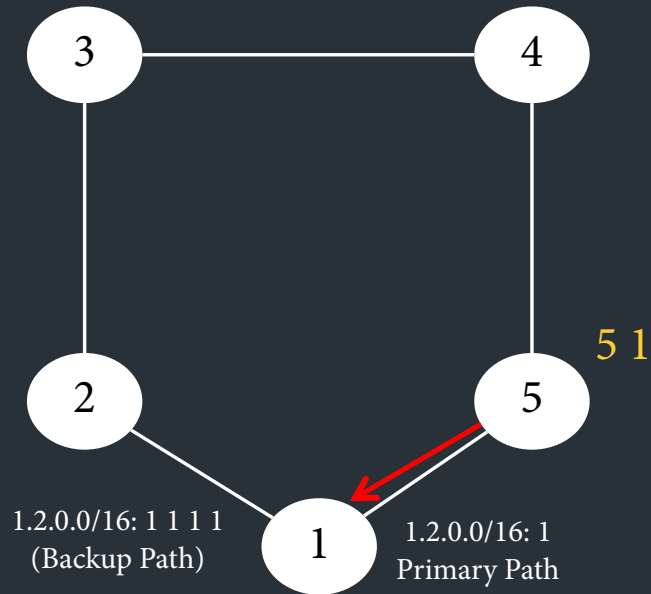
BGP Wedgies



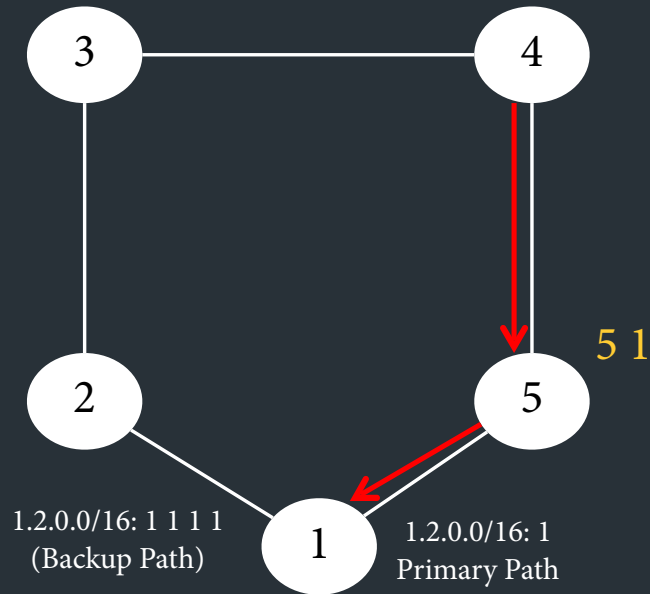
BGP Wedgies



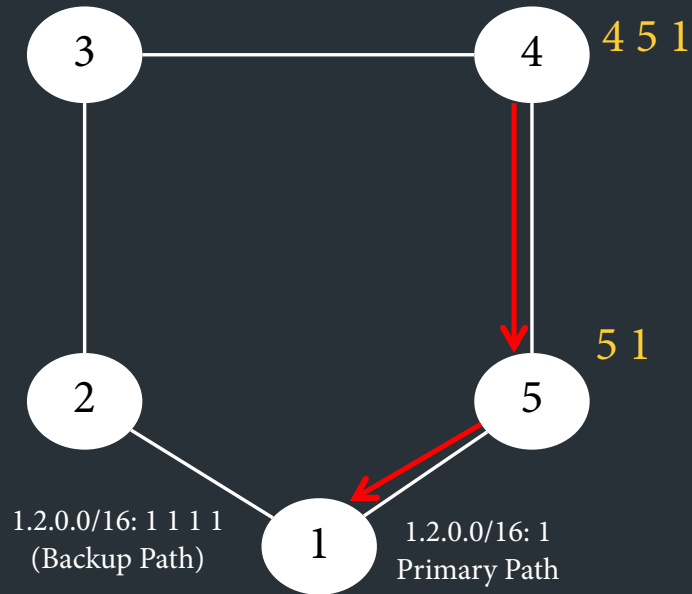
BGP Wedgies



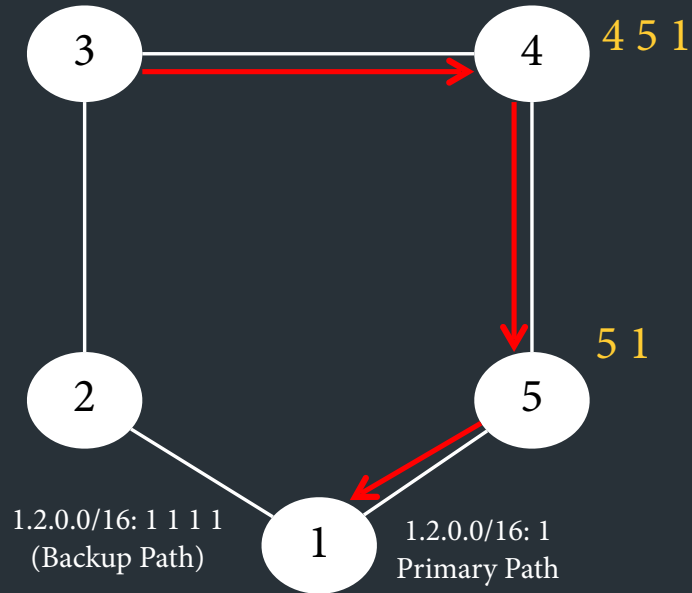
BGP Wedgies



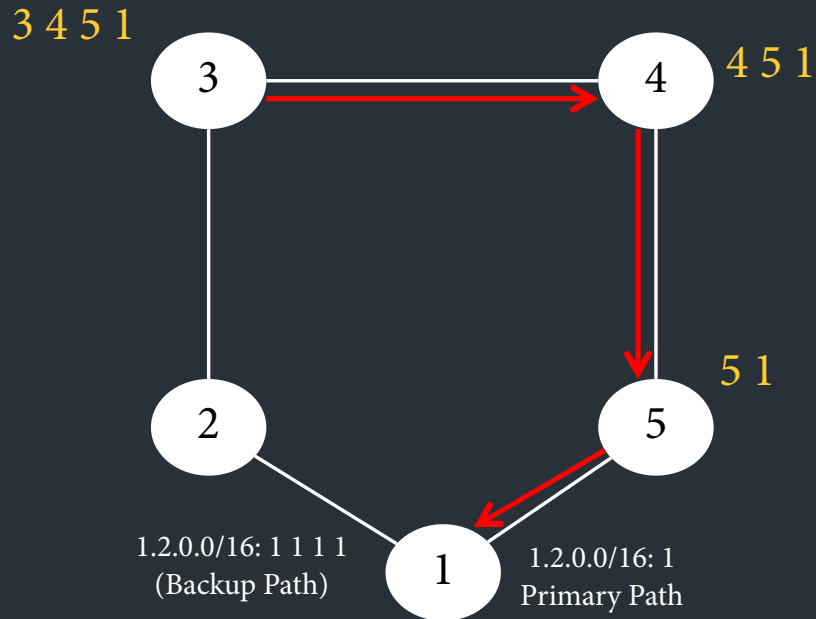
BGP Wedgies



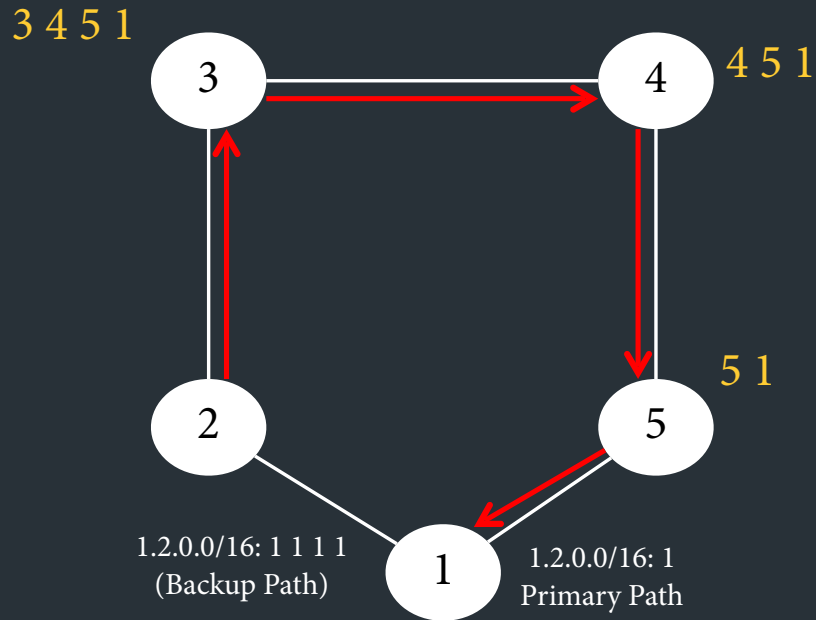
BGP Wedgies



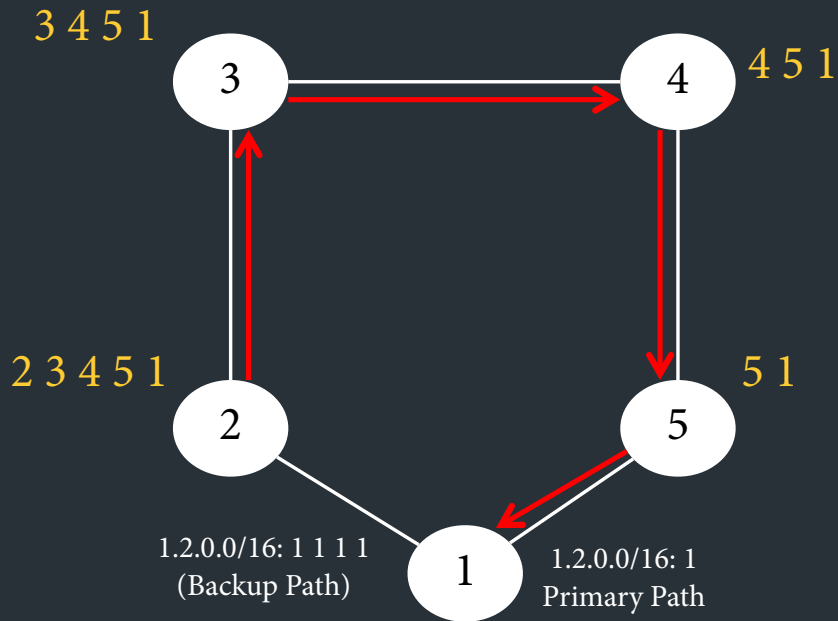
BGP Wedgies



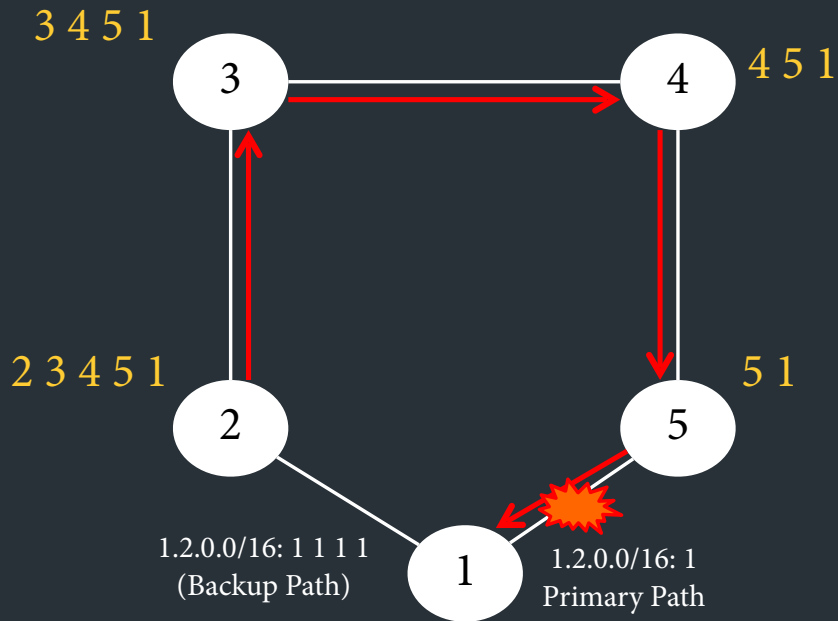
BGP Wedgies



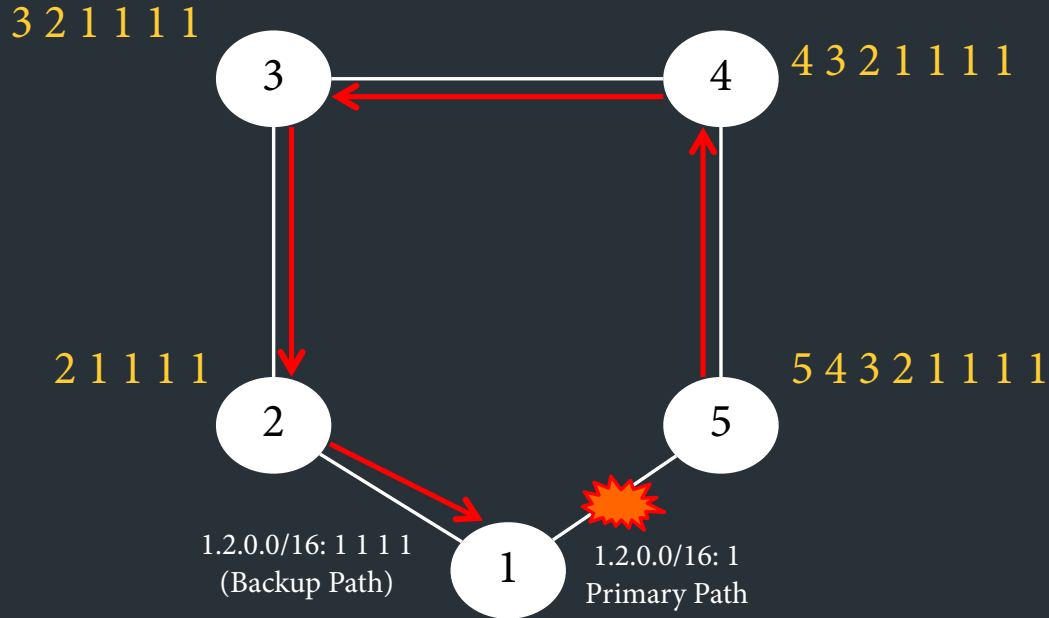
BGP Wedgies



BGP Wedgies



BGP Wedgies



BGP Wedgies

3 prefers customer route: stable configuration!

3 2 1 1 1 1

