

# Best Practices in IPv4 Anycast Routing

**Gaurab Raj Upadhaya**

**PCH**

# What *isn't* Anycast?

- Not a protocol, not a different version of IP, nobody's proprietary technology.
- Doesn't require any special capabilities in the servers, clients, or network.
- Doesn't break or confuse existing infrastructure.

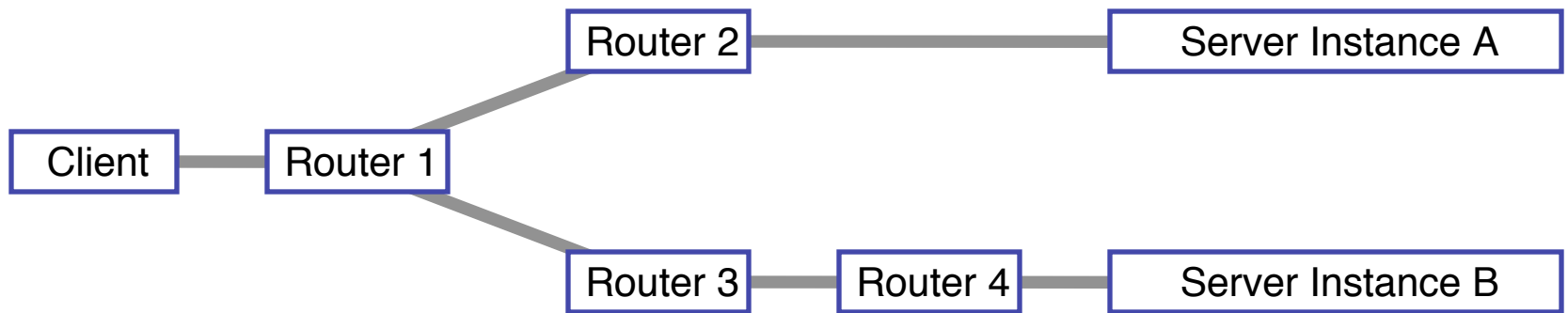
# What *is* Anycast?

- Just a configuration methodology.
- Mentioned, although not described in detail, in numerous RFCs since time immemorial.
  - RFC 1546 “Host Anycasting Service” 1993,
  - New Recent Draft “Operation of Anycast Services” `draft-ietf-grow-anycast-03`; Expires 28 July, 2006.
- It’s been the basis for large-scale content-distribution networks since at least 1995.

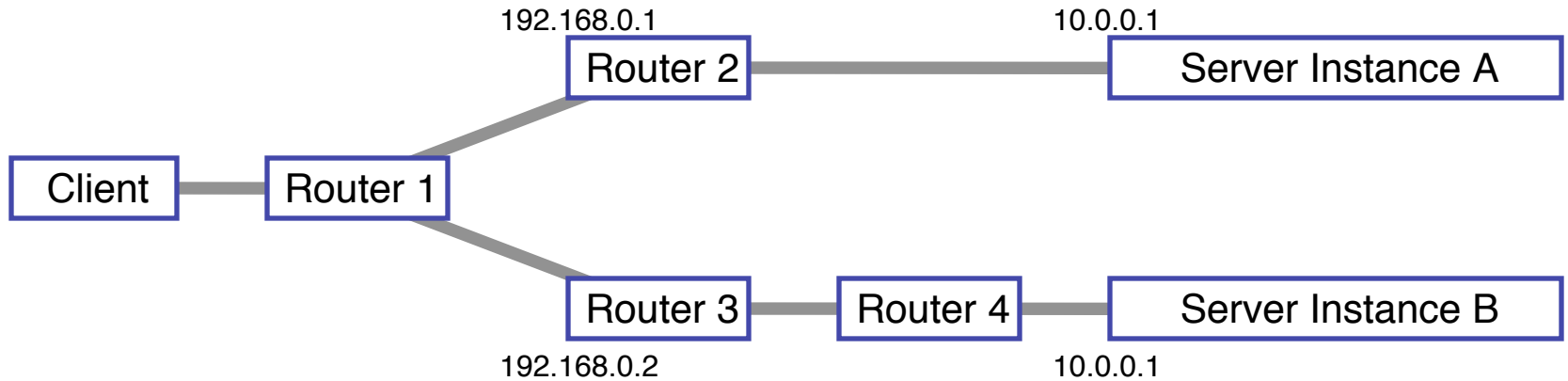
# How Does Anycast Work?

- The basic idea is extremely simple:
- Multiple instances of a service share the same IP address.
- The routing infrastructure directs any packet to the topologically nearest instance of the service.
- What little complexity exists is in the optional details.

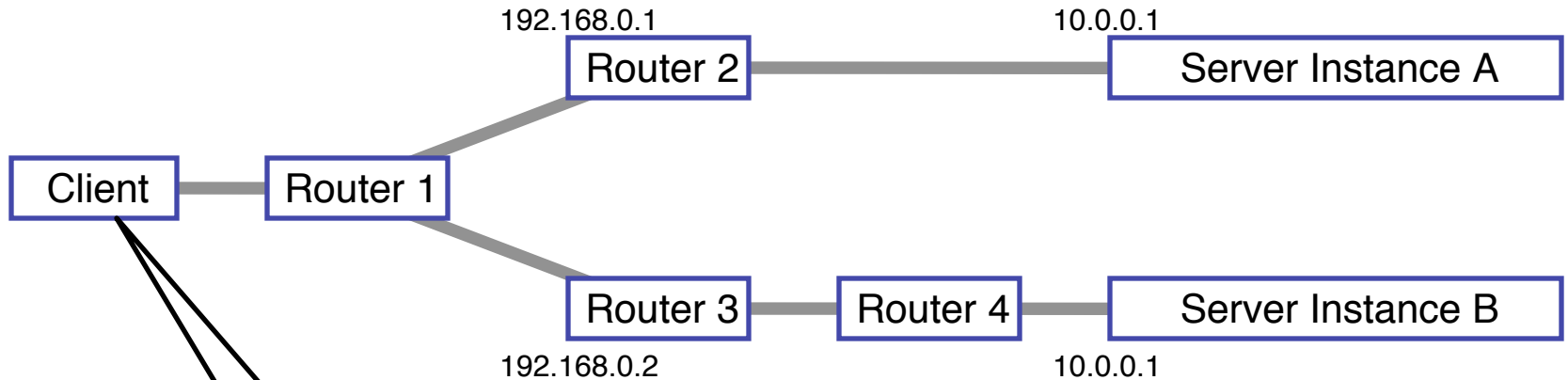
# Example



# Example



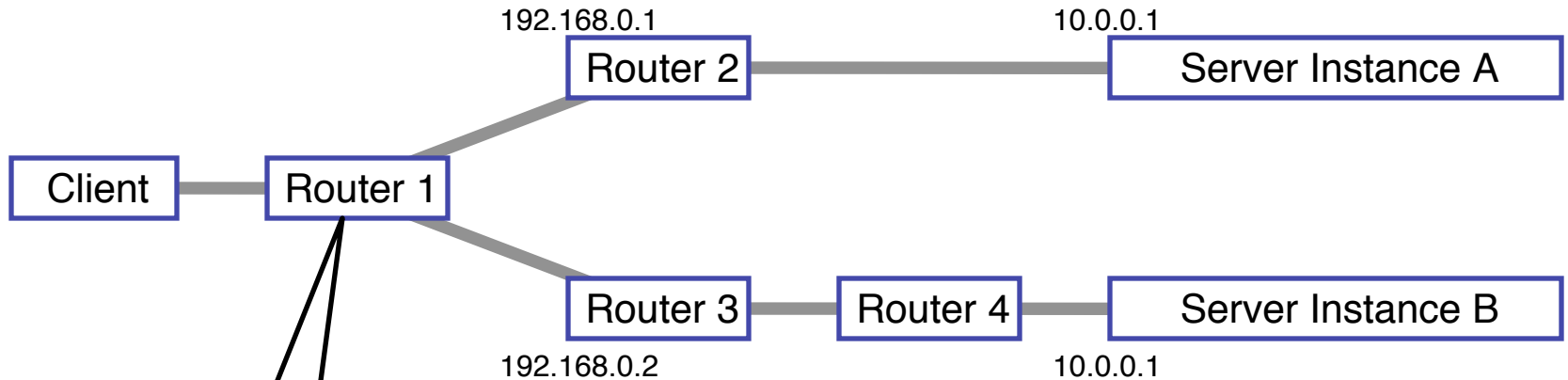
# Example



DNS lookup for `http://www.server.com/`  
produces a single answer:

```
www.server.com. IN A 10.0.0.1
```

# Example

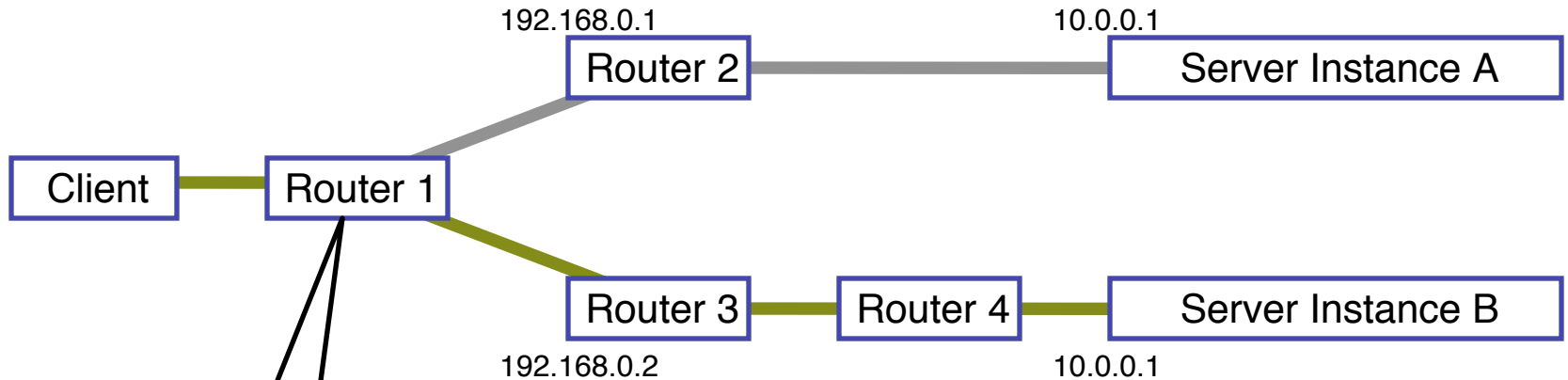


Routing Table from Router 1:

| Destination | Mask | Next-Hop    | Distance |
|-------------|------|-------------|----------|
| 192.168.0.0 | /29  | 127.0.0.1   | 0        |
| 10.0.0.1    | /32  | 192.168.0.1 | 1        |
| 10.0.0.1    | /32  | 192.168.0.2 | 2        |



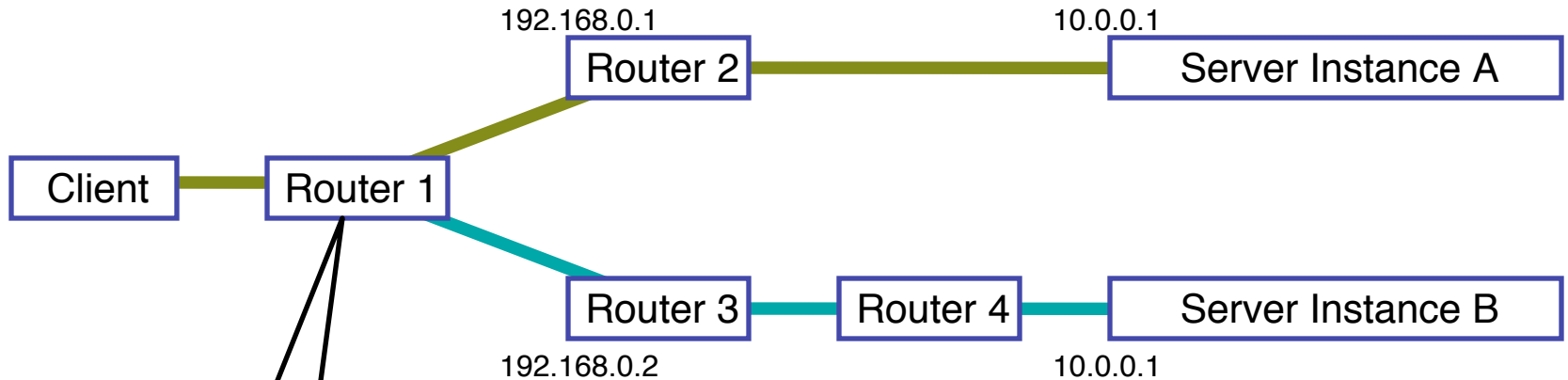
# Example



Routing Table from Router 1:

| Destination | Mask | Next-Hop    | Distance |
|-------------|------|-------------|----------|
| 192.168.0.0 | /29  | 127.0.0.1   | 0        |
| 10.0.0.1    | /32  | 192.168.0.1 | 1        |
| 10.0.0.1    | /32  | 192.168.0.2 | 2        |

# Example

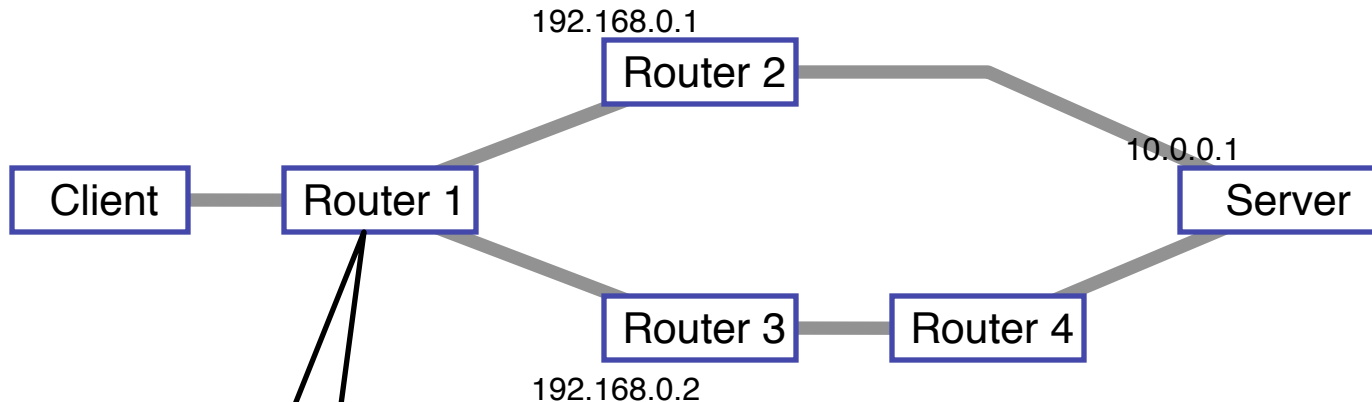


Routing Table from Router 1:

| Destination | Mask | Next-Hop    | Distance |
|-------------|------|-------------|----------|
| 192.168.0.0 | /29  | 127.0.0.1   | 0        |
| 10.0.0.1    | /32  | 192.168.0.1 | 1        |
| 10.0.0.1    | /32  | 192.168.0.2 | 2        |

# Example

What the routers think the topology looks like:



Routing Table from Router 1:

| Destination | Mask | Next-Hop    | Distance |
|-------------|------|-------------|----------|
| 192.168.0.0 | /29  | 127.0.0.1   | 0        |
| 10.0.0.1    | /32  | 192.168.0.1 | 1        |
| 10.0.0.1    | /32  | 192.168.0.2 | 2        |

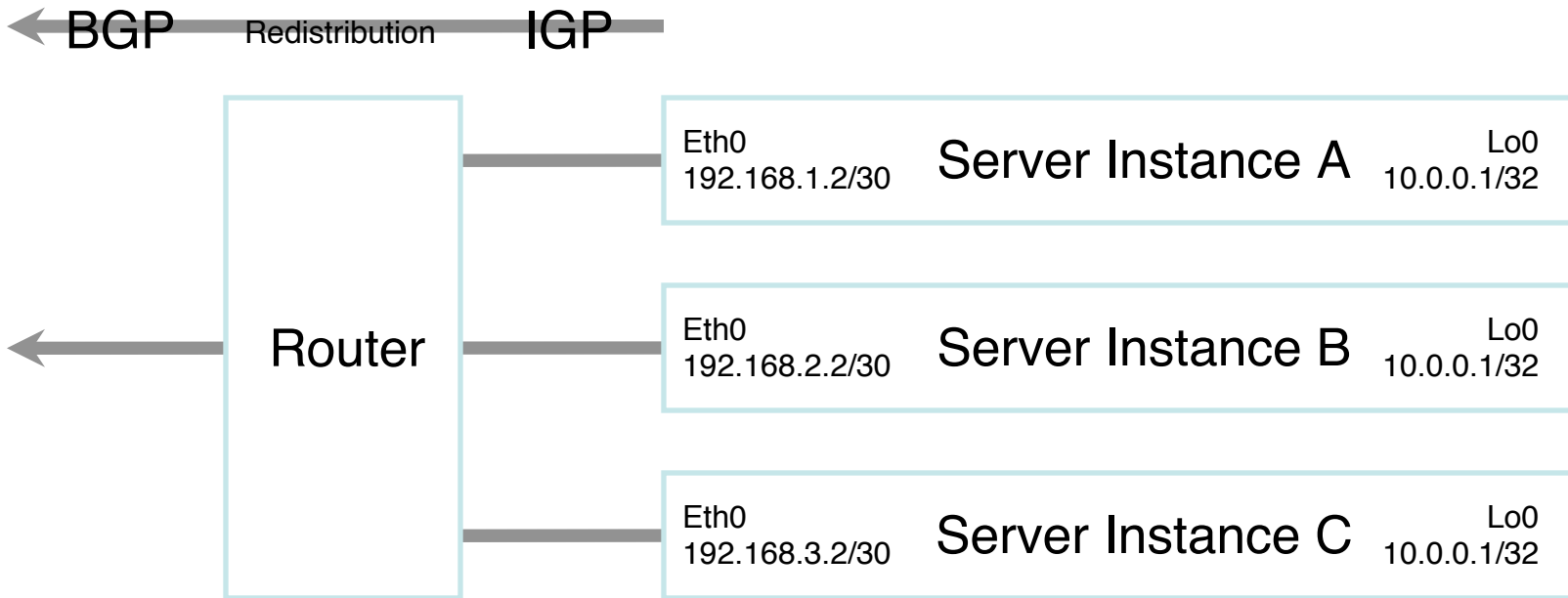
# Building an Anycast Server Cluster

- Anycast can be used in building either local server clusters, or global networks, or global networks of clusters, combining both scales.
- F-root is a local anycast server cluster, for instance.

# Building an Anycast Server Cluster

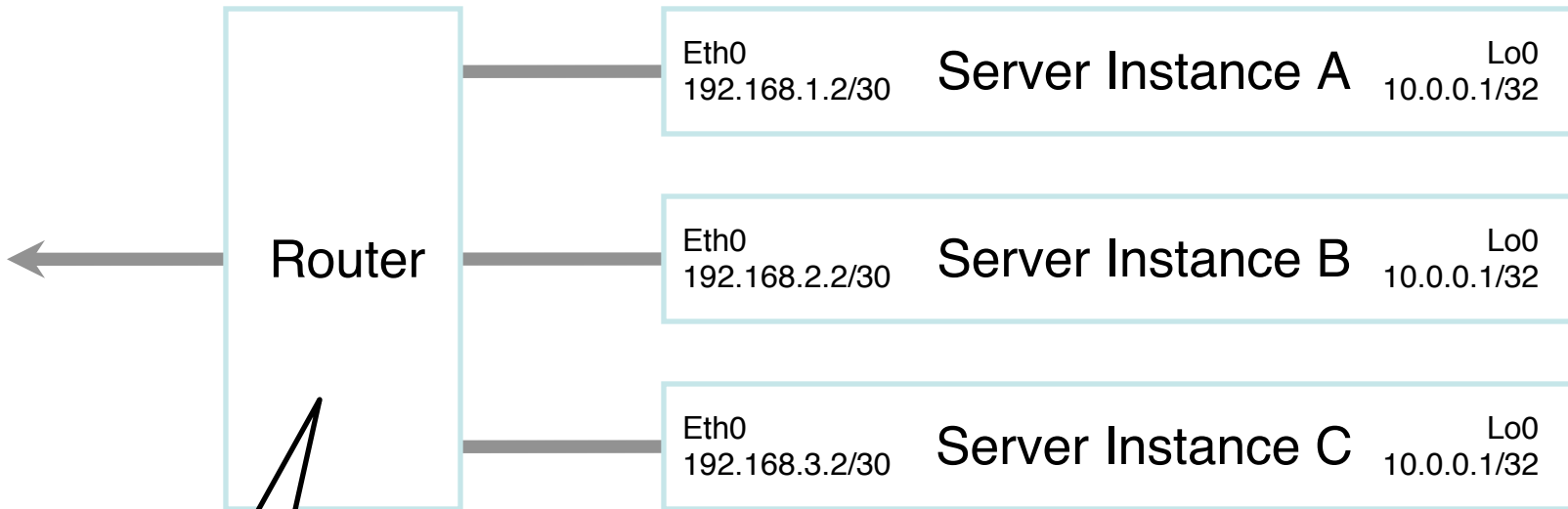
- Typically, a cluster of servers share a common virtual interface attached to their loopback devices, and speak an IGP routing protocol to an adjacent BGP-speaking border router.
- The servers may or may not share identical content.

# Example



# Example

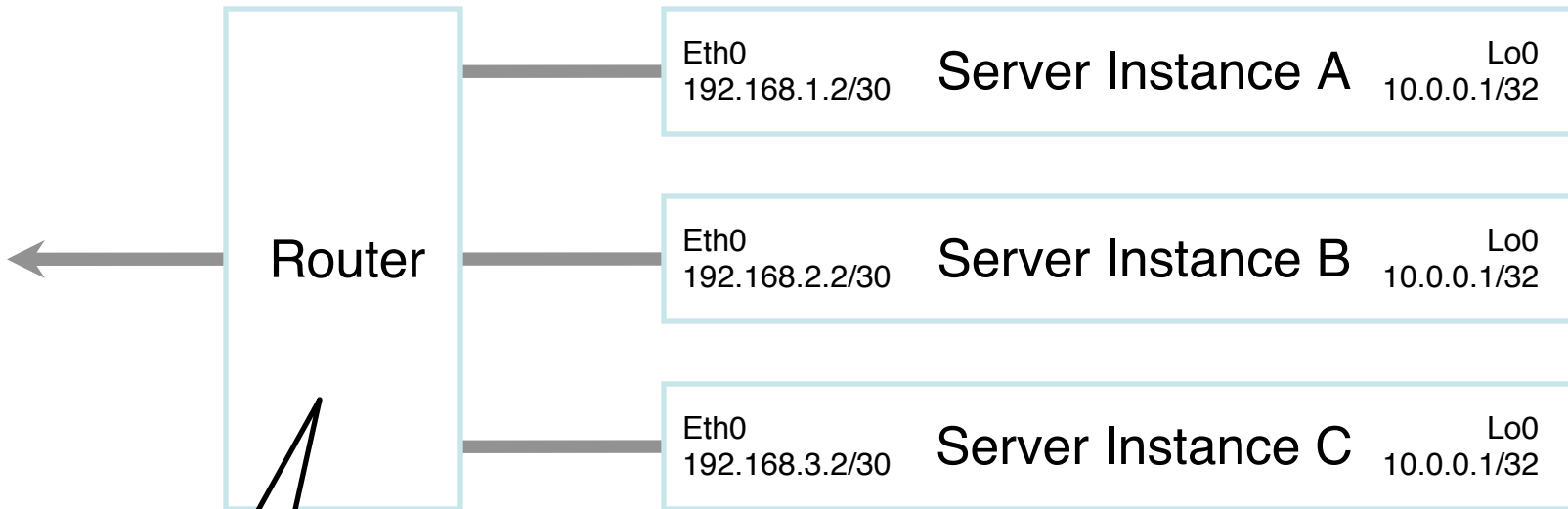
← BGP — Redistribution — IGP



| Destination | Mask | Next-Hop    | Dist |
|-------------|------|-------------|------|
| 0.0.0.0     | /0   | 127.0.0.1   | 0    |
| 192.168.1.0 | /30  | 192.168.1.1 | 0    |
| 192.168.2.0 | /30  | 192.168.2.1 | 0    |
| 192.168.3.0 | /30  | 192.168.3.1 | 0    |
| 10.0.0.1    | /32  | 192.168.1.2 | 1    |
| 10.0.0.1    | /32  | 192.168.2.2 | 1    |
| 10.0.0.1    | /32  | 192.168.3.2 | 1    |

# Example

← BGP — Redistribution — IGP



| Destination | Mask | Next-Hop    | Dist |
|-------------|------|-------------|------|
| 0.0.0.0     | /0   | 127.0.0.1   | 0    |
| 192.168.1.0 | /30  | 192.168.1.1 | 0    |
| 192.168.2.0 | /30  | 192.168.2.1 | 0    |
| 192.168.3.0 | /30  | 192.168.3.1 | 0    |
| 10.0.0.1    | /32  | 192.168.1.2 | 1    |
| 10.0.0.1    | /32  | 192.168.2.2 | 1    |
| 10.0.0.1    | /32  | 192.168.3.2 | 1    |

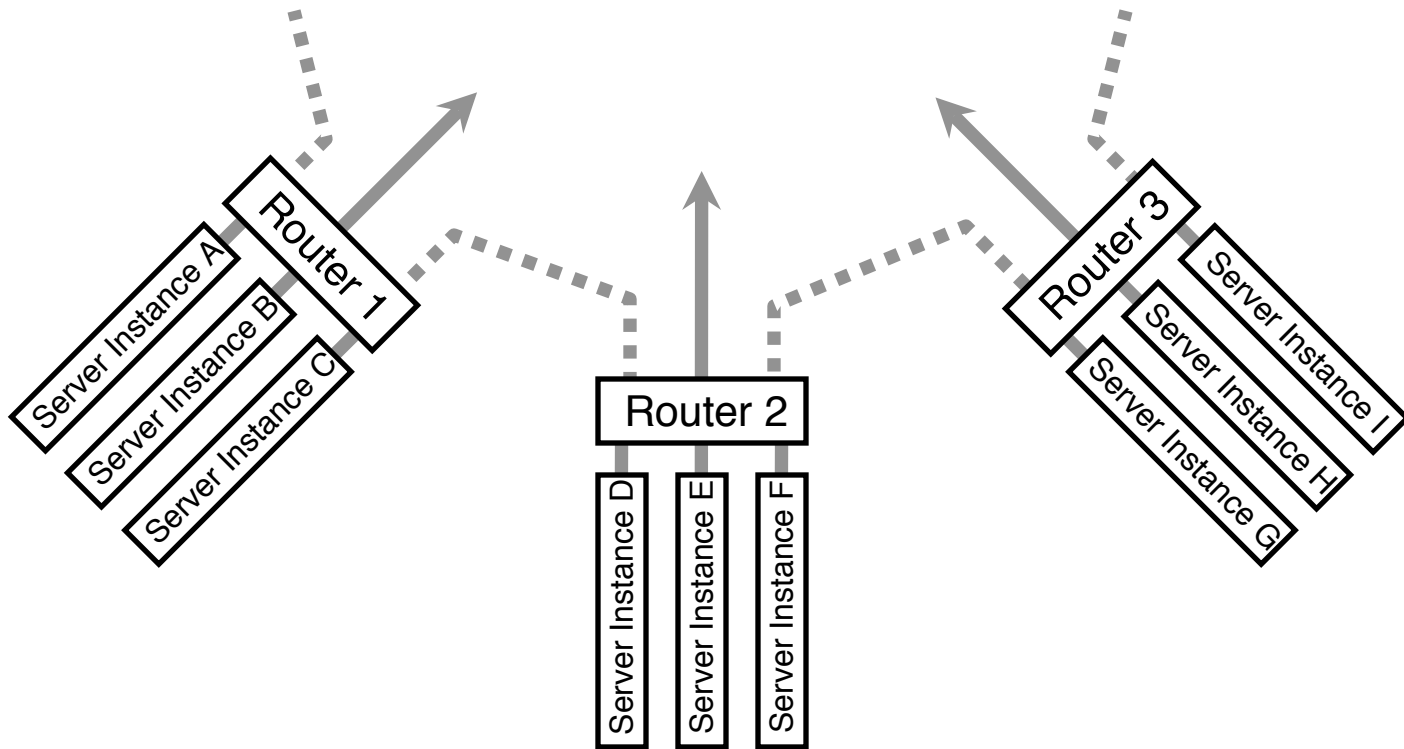
} Round-robin load balancing



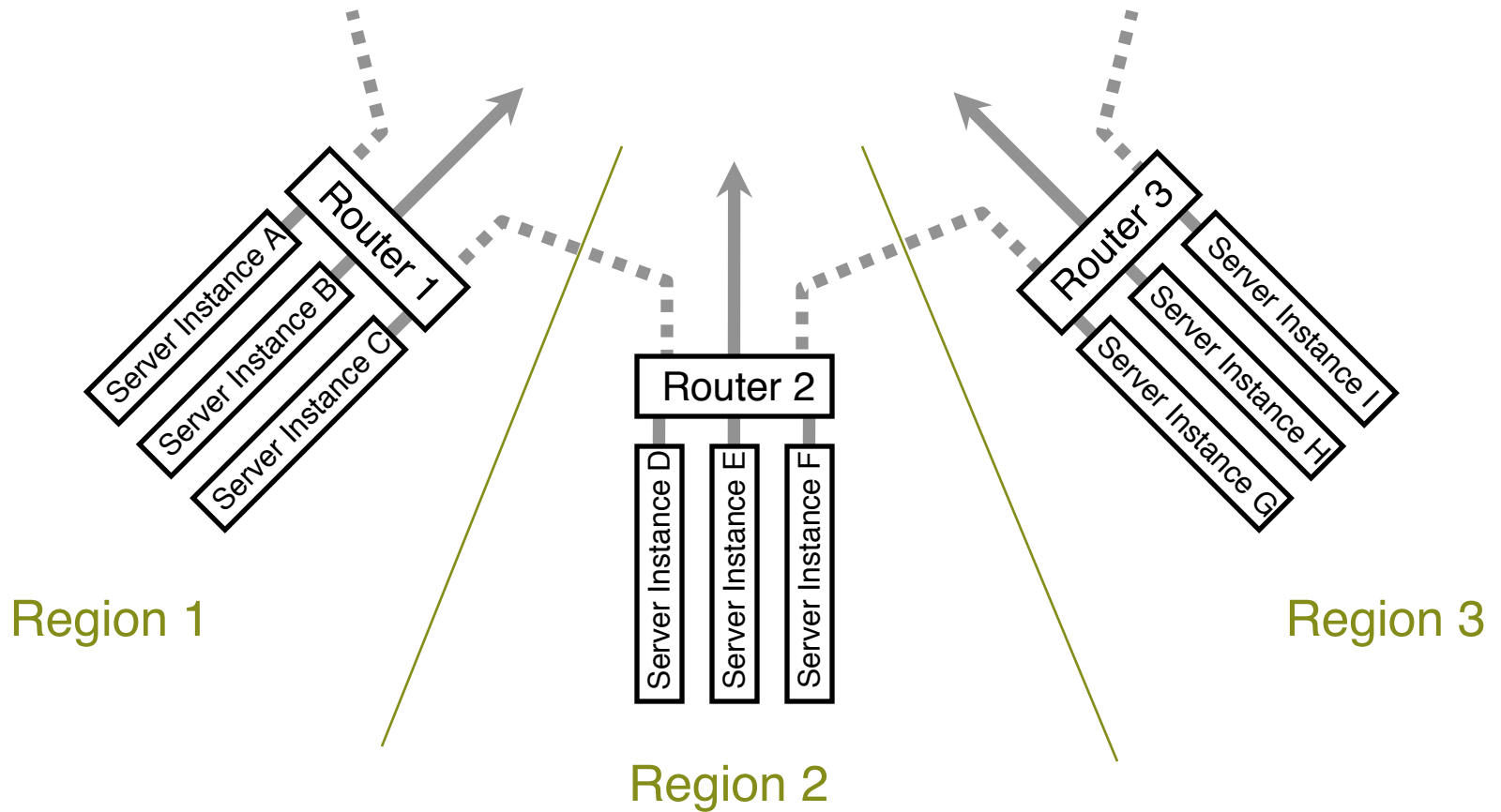
# Building a Global Network of Clusters

- Once a cluster architecture has been established, additional clusters can be added to gain performance.
- Load distribution, fail-over between clusters, and content synchronization become the principal engineering concerns.

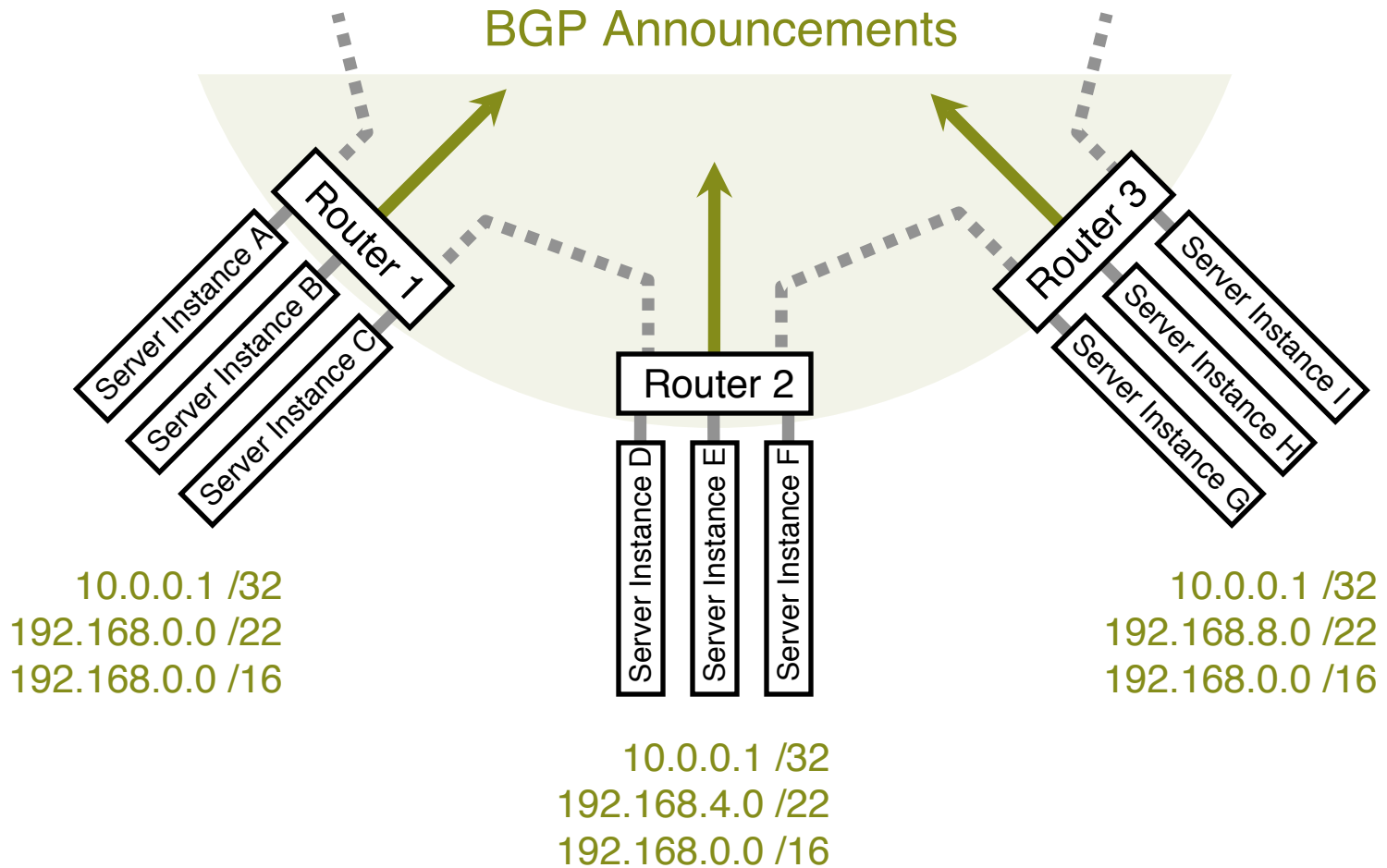
# Example



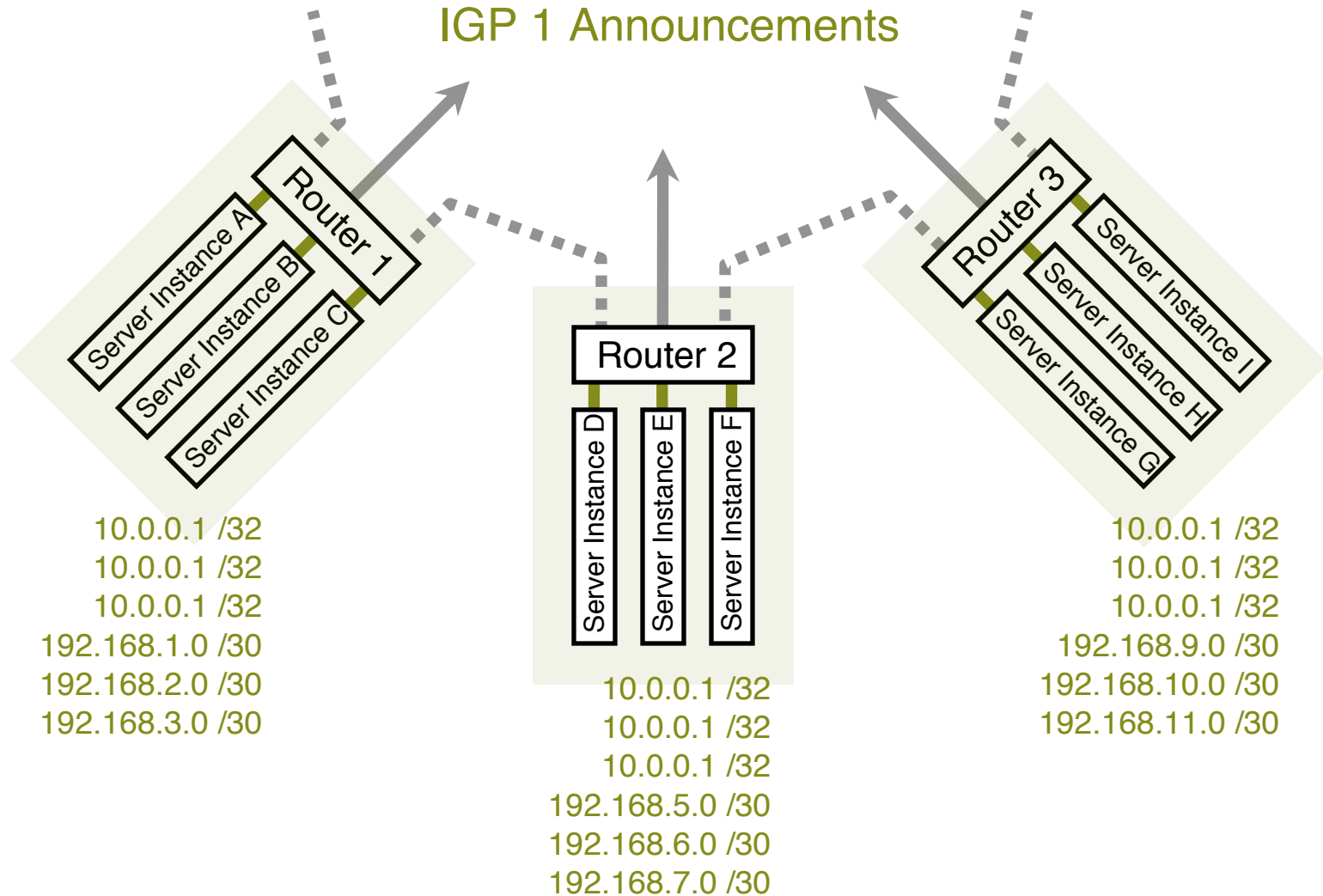
# Example



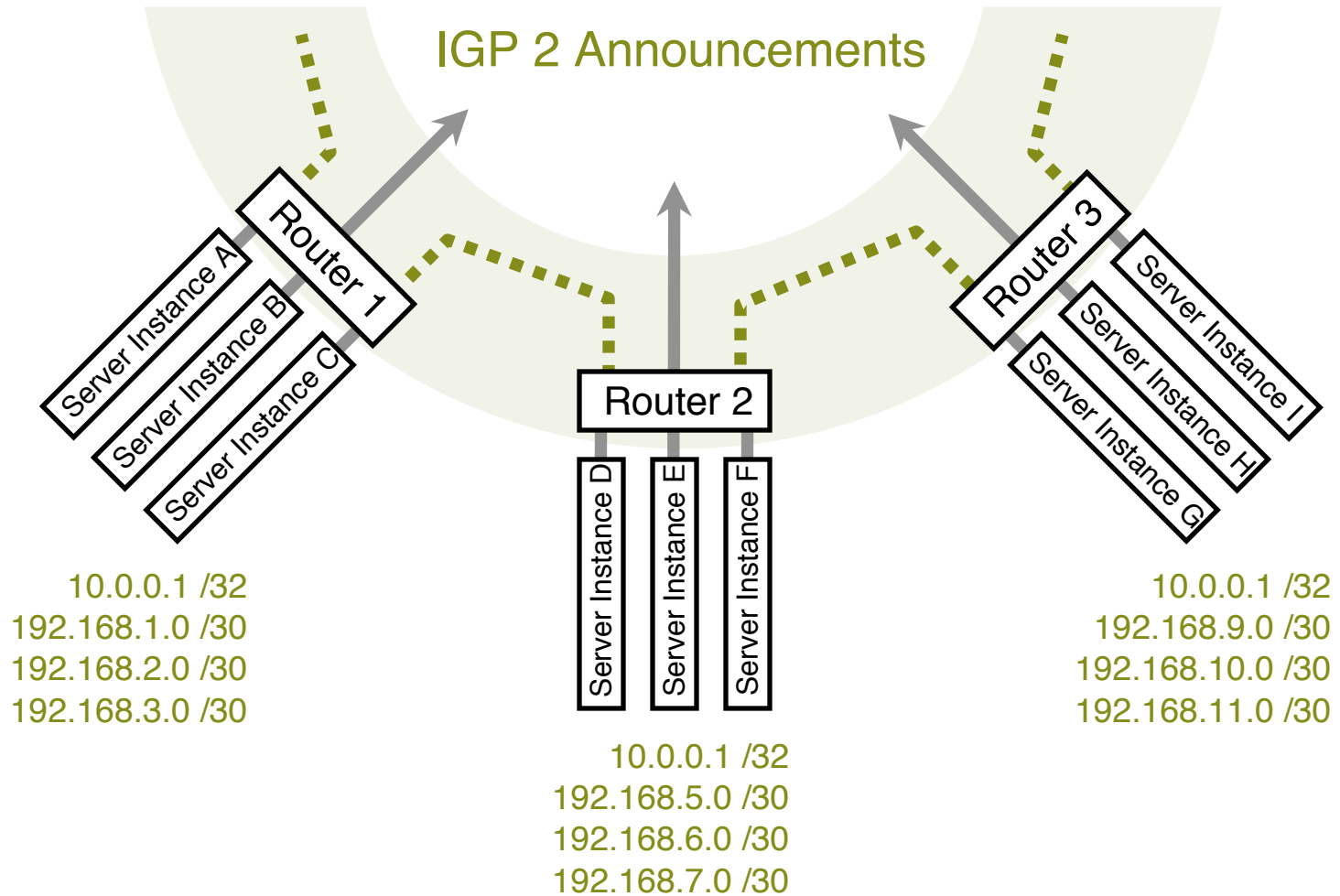
# Example



# Example



# Example



# Performance-Tuning Anycast Networks

- Server deployment in anycast networks is always a tradeoff between absolute cost and efficiency.
- The network will perform best if servers are widely distributed, with higher density in and surrounding high demand areas.
- Lower initial cost sometimes leads implementers to compromise by deploying more servers in existing locations, which is less efficient.

# Caveats and Failure Modes

- DNS resolution fail-over
- Long-lived connection-oriented flows
- Identifying which server is giving an end-user trouble



# DNS Resolution Fail-Over

- In the event of poor performance from a server, DNS servers will fail over to the next server in a list.
- If both servers are in fact hosted in the same anycast cloud, the resolver will wind up talking to the same instance again.
- Best practices for anycast DNS server operations indicate a need for two separate overlapping clouds of anycast servers.

# Long-Lived Connection-Oriented Flows

- Long-lived flows, typically TCP file-transfers or interactive logins, may occasionally be more stable than the underlying Internet topology.
- If the underlying topology changes sufficiently during the life of an individual flow, packets could be redirected to a different server instance, which would not have proper TCP state, and would reset the connection.
- This is not a problem with web servers unless they're maintaining stateful per-session information about end-users, rather than embedding it in URLs or cookies.
- Web servers HTTP redirect to their unique address whenever they need to enter a stateful mode.
- Limited operational data shows underlying instability to be on the order of one flow per ten thousand per hour of duration.

# Identifying Problematic Server Instances

- Some protocols may not include an easy in-band method of identifying the server which persists beyond the duration of the connection.
- Traceroute always identifies the *current* server instance, but end-users may not even have traceroute.

# A Security Ramification

- Anycast server clouds have the useful property of sinking DOS attacks at the instance nearest to the source of the attack, leaving all other instances unaffected.
- This is still of some utility even when DOS sources are widely distributed.

# Routing Caveats

- **NO\_EXPORT**
  - Does it work ? May be it does, may be it doesn't.
  - Added complexity of announcing covering prefixes
  - Some further work done by the RIPE NCC folks, more measurements are needed.

Questions ?

# Thank You

Gaurab Raj Upadhaya

gaurab @ lahai. Com

The anycast tutorial can be found at

[http:// www.pch.net / resources / tutorials / anycast](http://www.pch.net/resources/tutorials/anycast)