

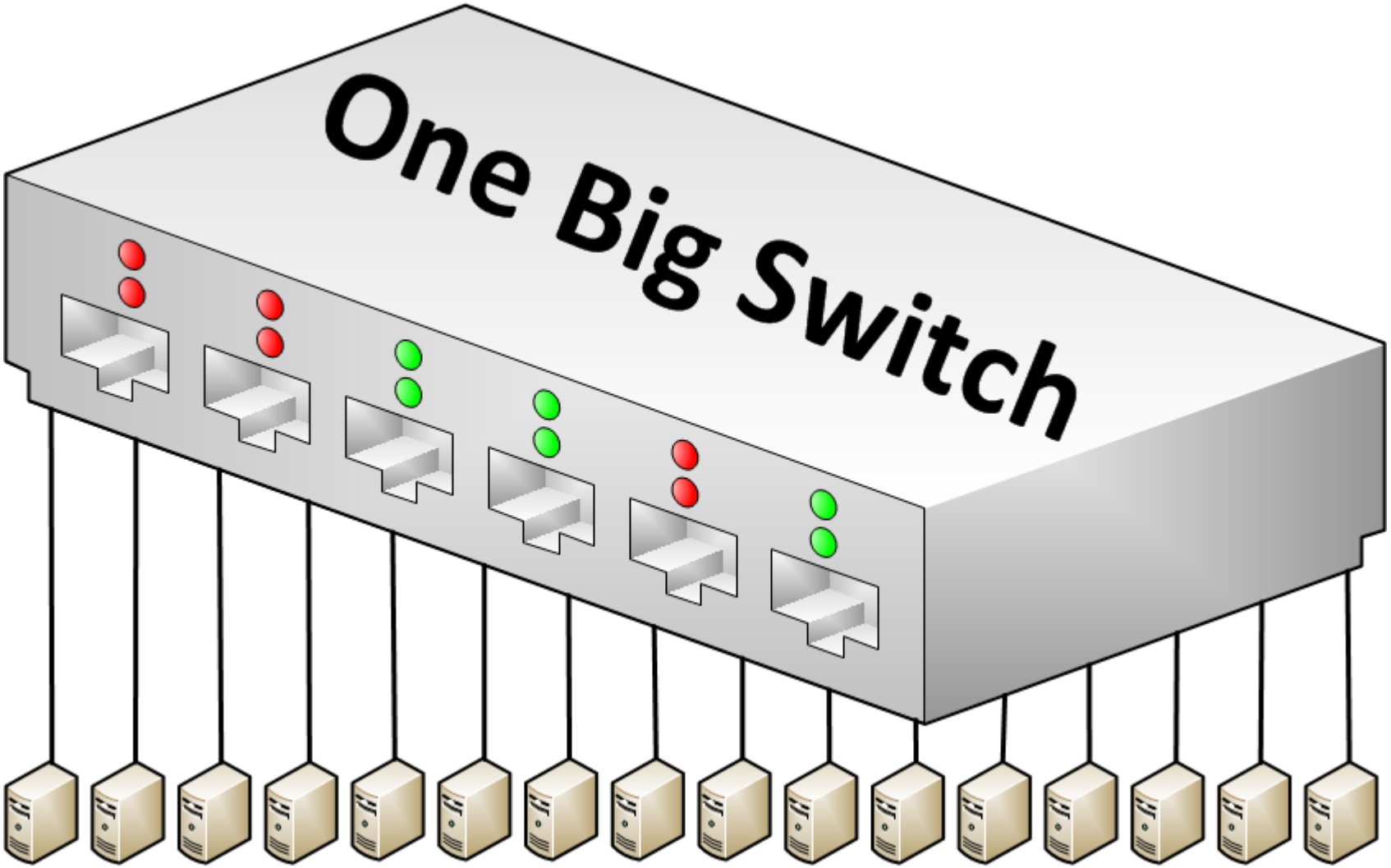
GRIN: Utilizing the Empty Half of Full Bisection Networks

Alexandru Agache and Costin Raiciu

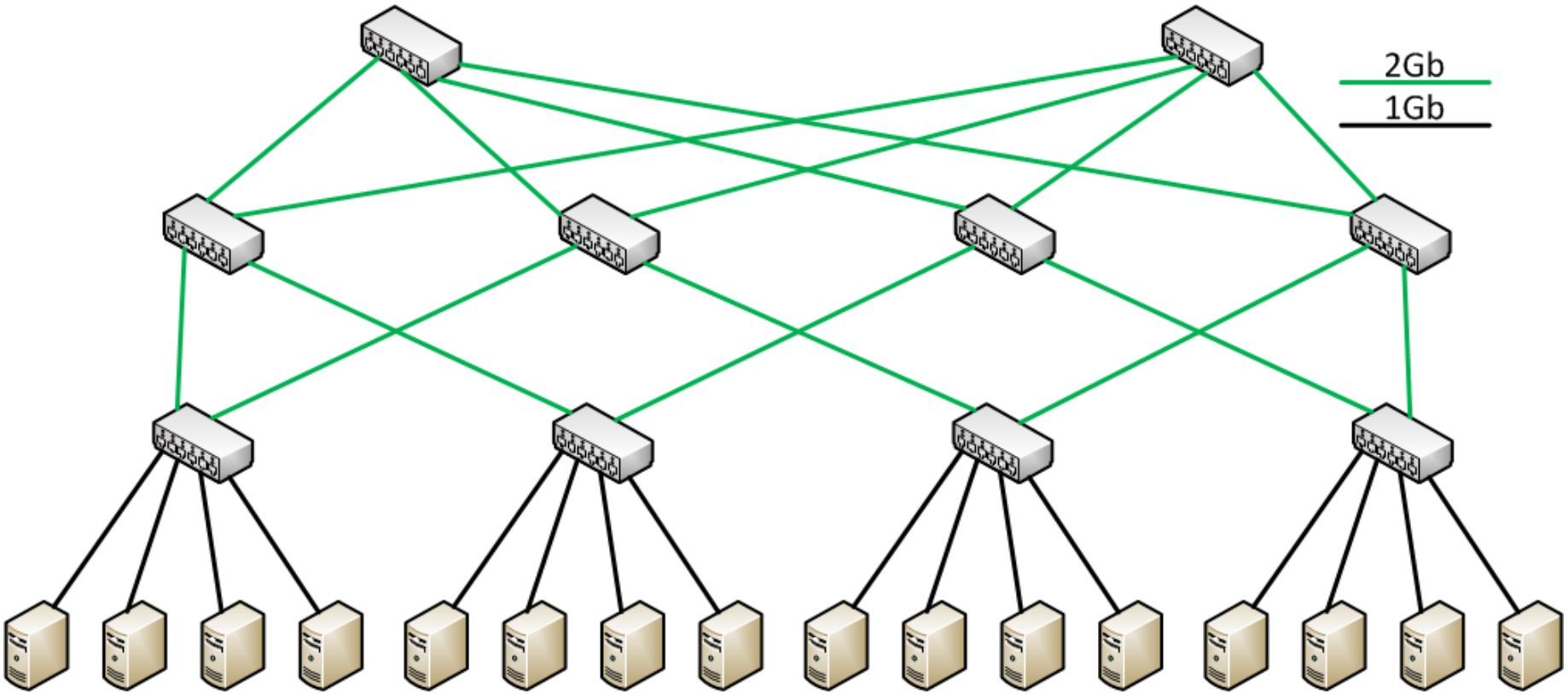
University Politehnica Bucharest

Thanks to CISCO

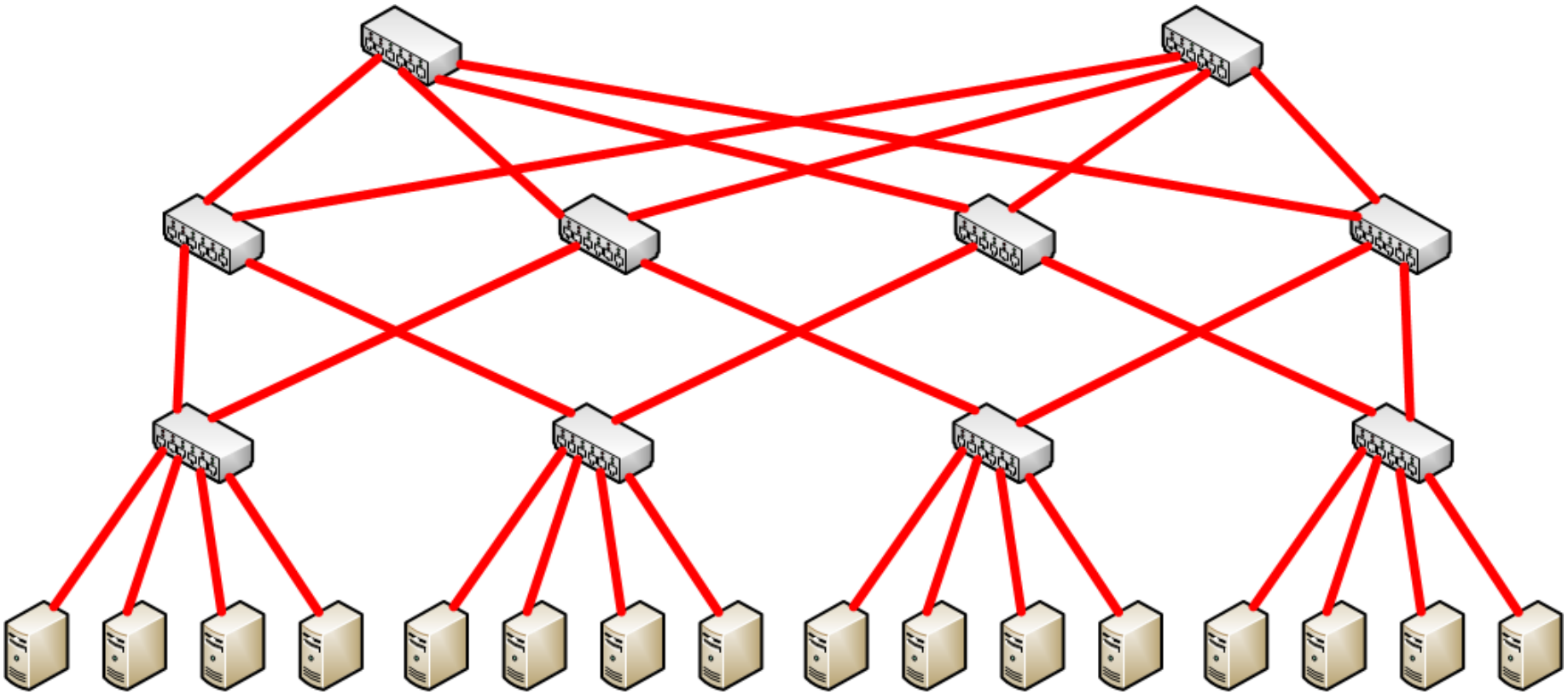
What is the best datacenter network topology ?


One Big Switch

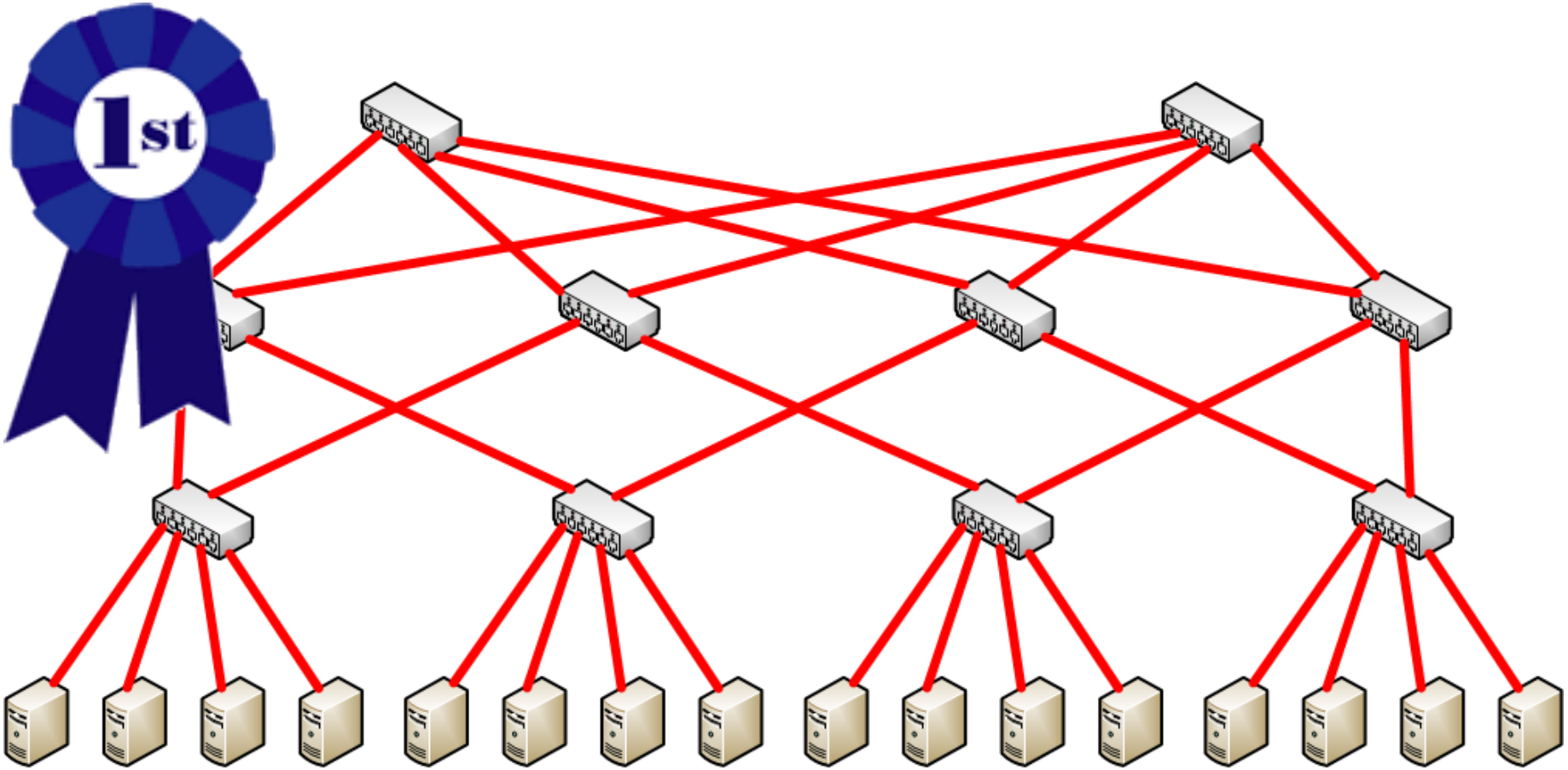
A Full-Bisection Network

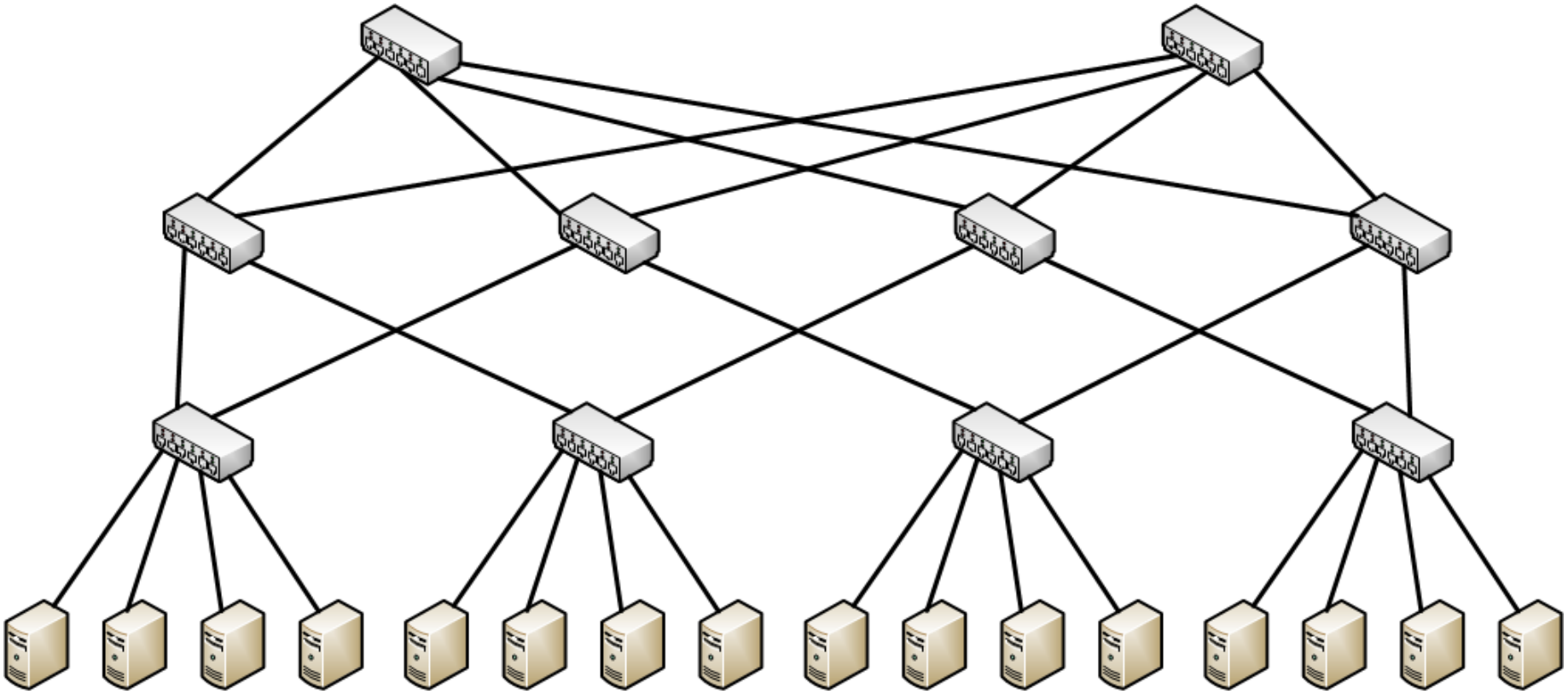


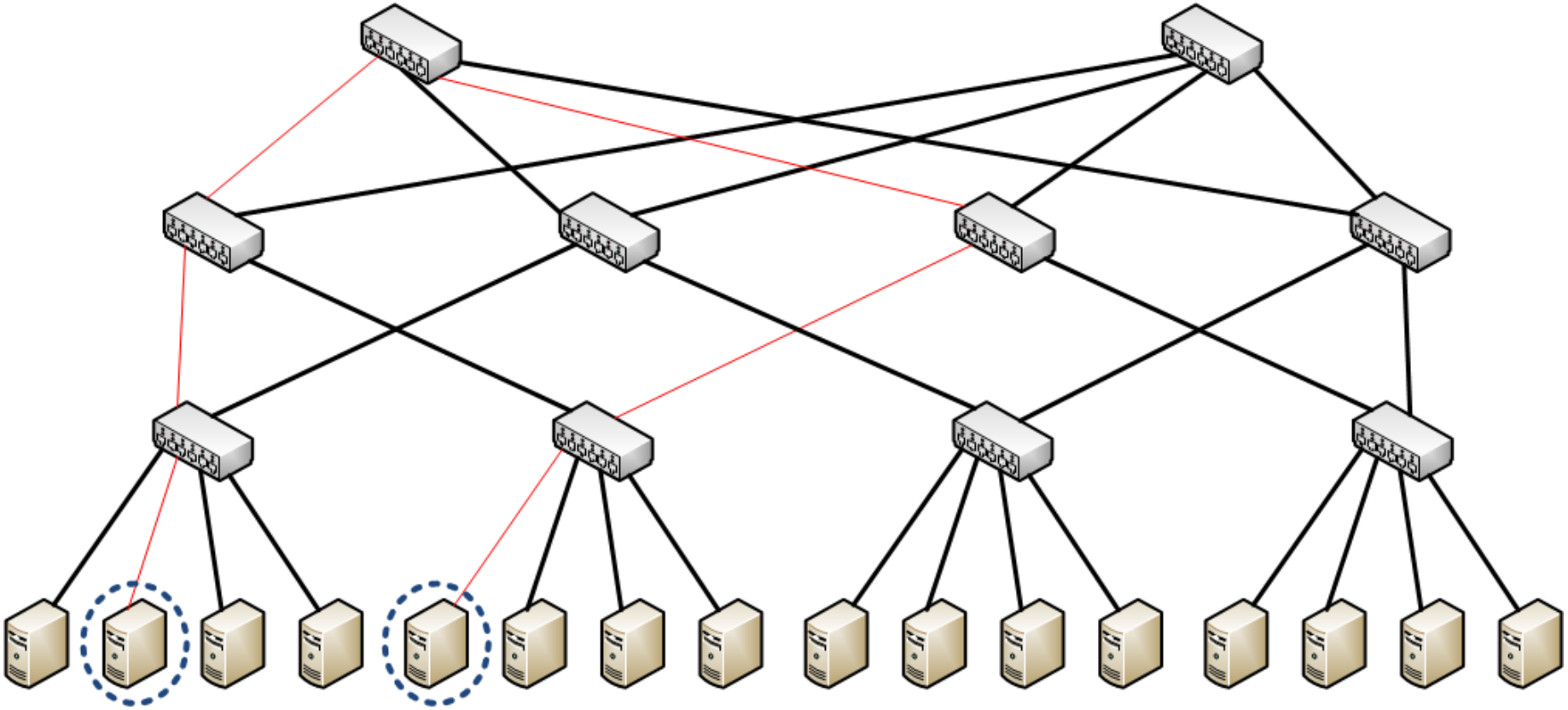
All-to-all Heavy Traffic

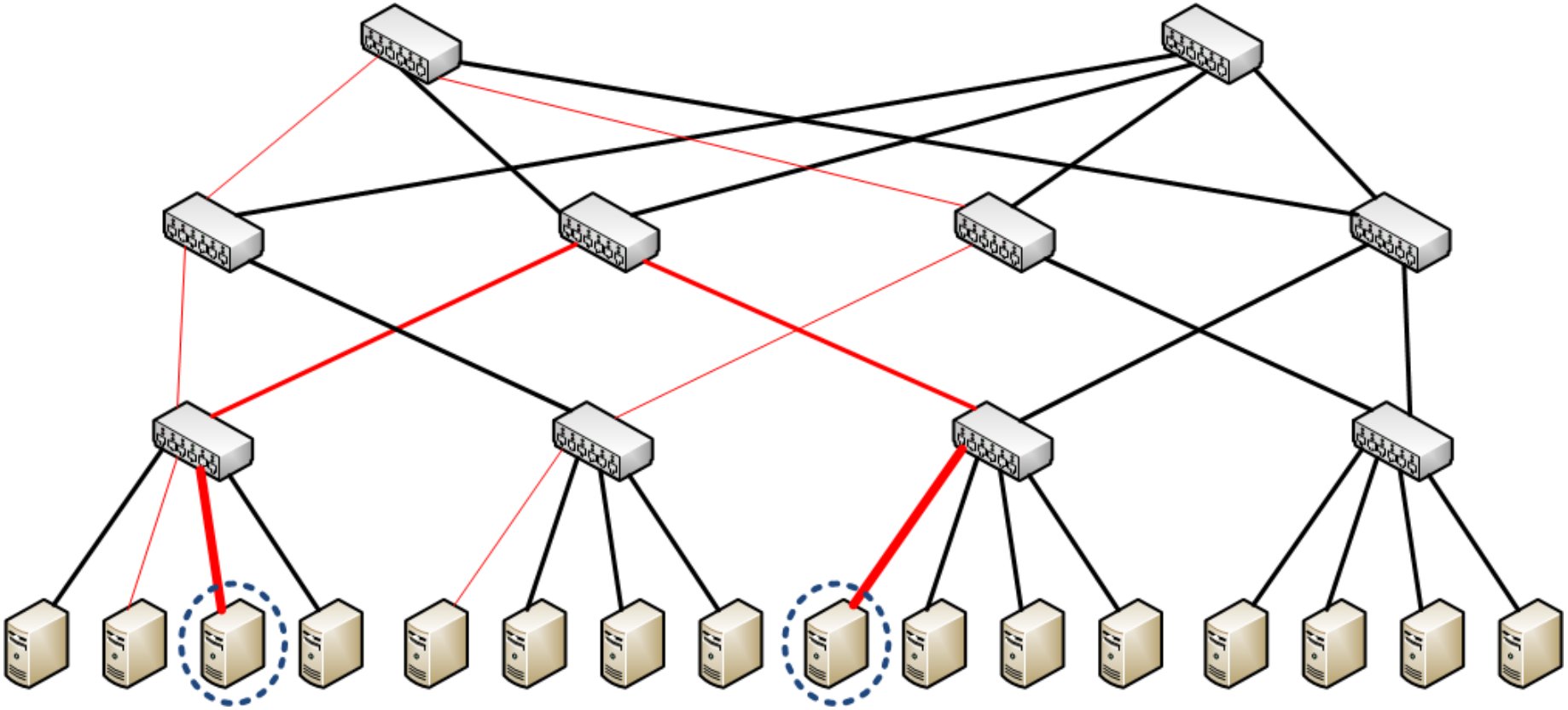


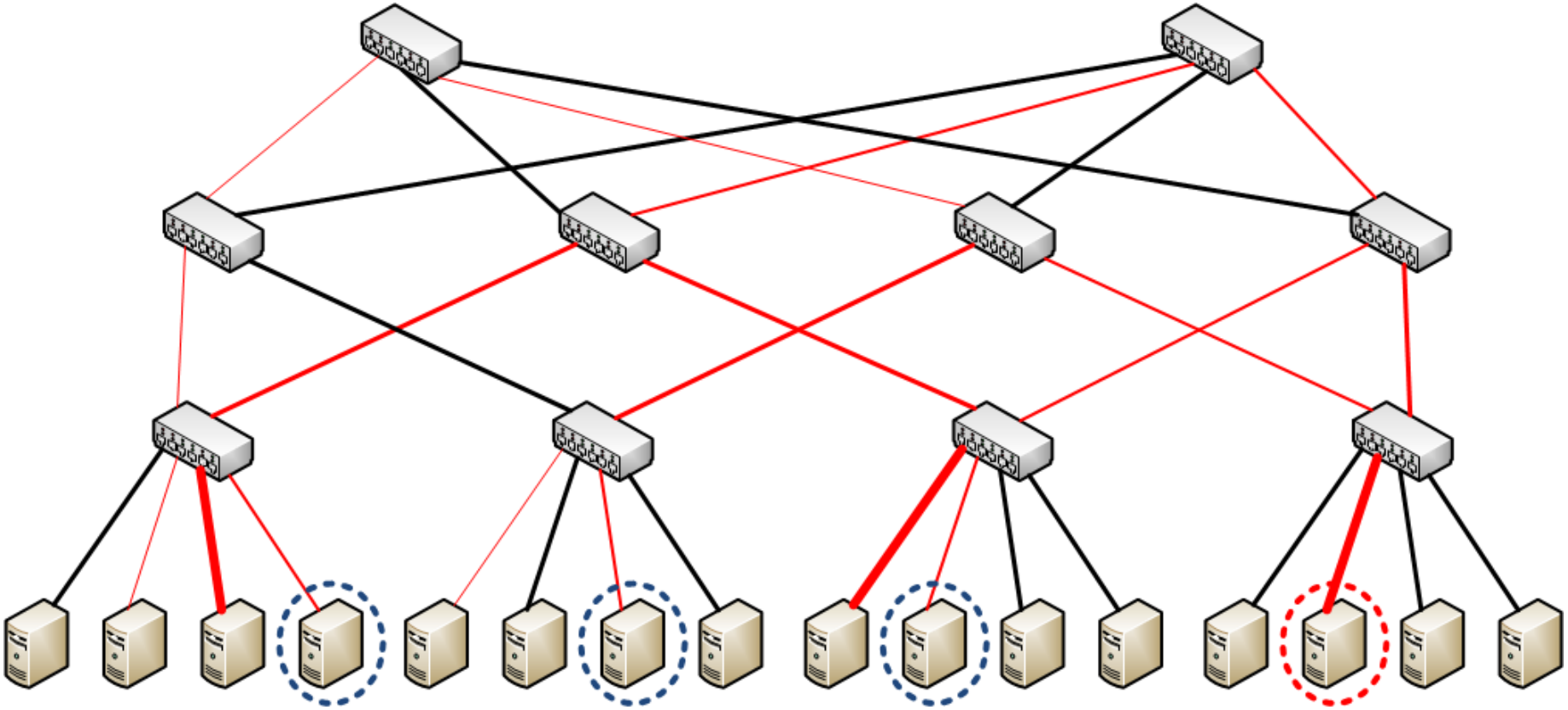
All-to-all Heavy Traffic

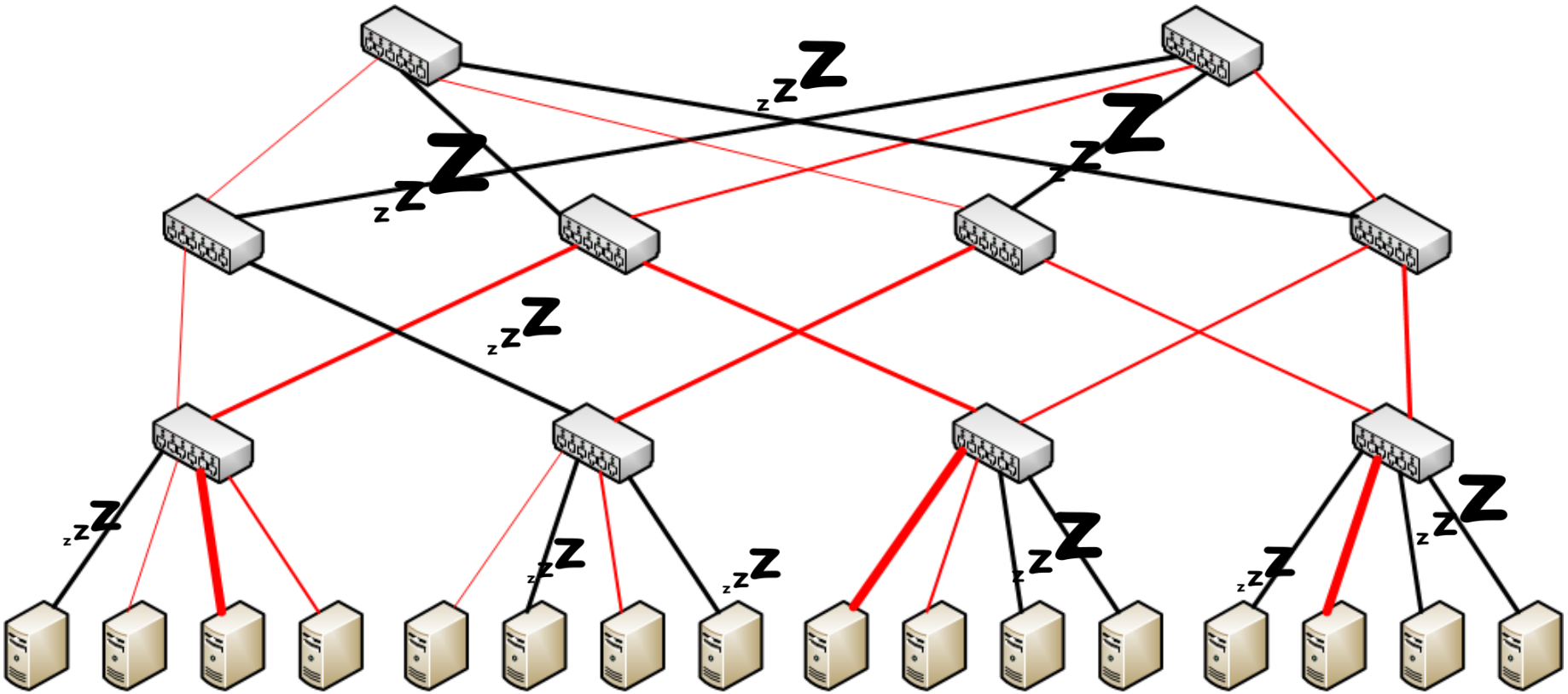




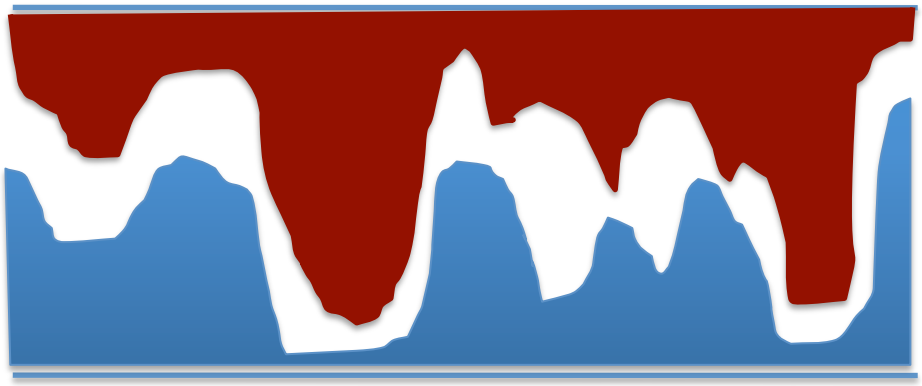






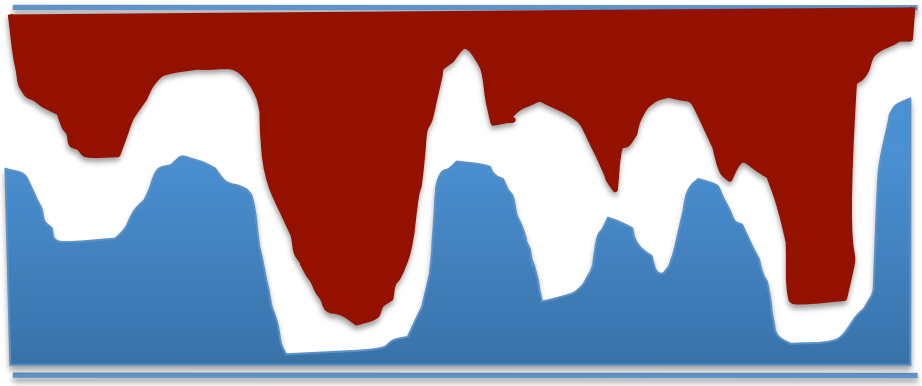


Resource Pooling



- different applications
- different loads at different times

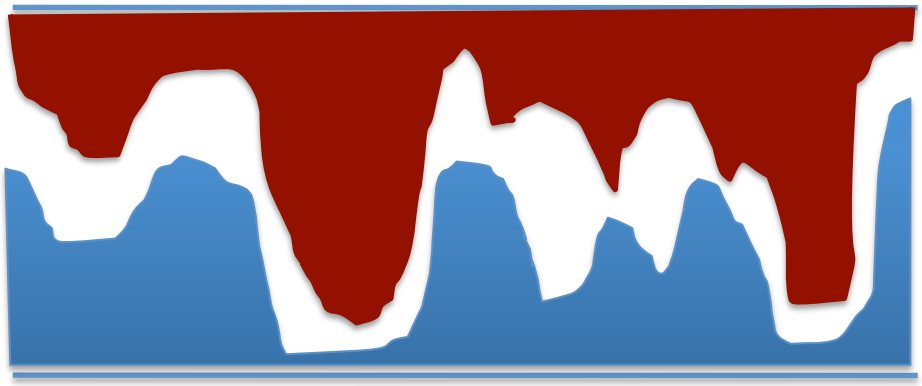
Resource Pooling



- different applications
- different loads at different times

- CPU ✓
- Memory ✓
- Storage ✓

Resource Pooling



- different applications
- different loads at different times

- CPU ✓
- Memory ✓
- Storage ✓
- Network capacity ✗

Our goal: Extend the resource pooling principle to datacenter networks.

Desirable Properties

Desirable Properties

- As cheap as possible



Desirable Properties

- As cheap as possible



- Widely applicable



Desirable Properties

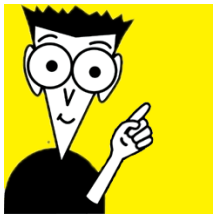
- As cheap as possible



- Widely applicable



- Easily and incrementally deployable



Requirements

Requirements

- Additional server network ports



Requirements

- Additional server network ports



(datacenter servers nowadays ship with dual or quad port NICs)

Requirements

- Additional server network ports



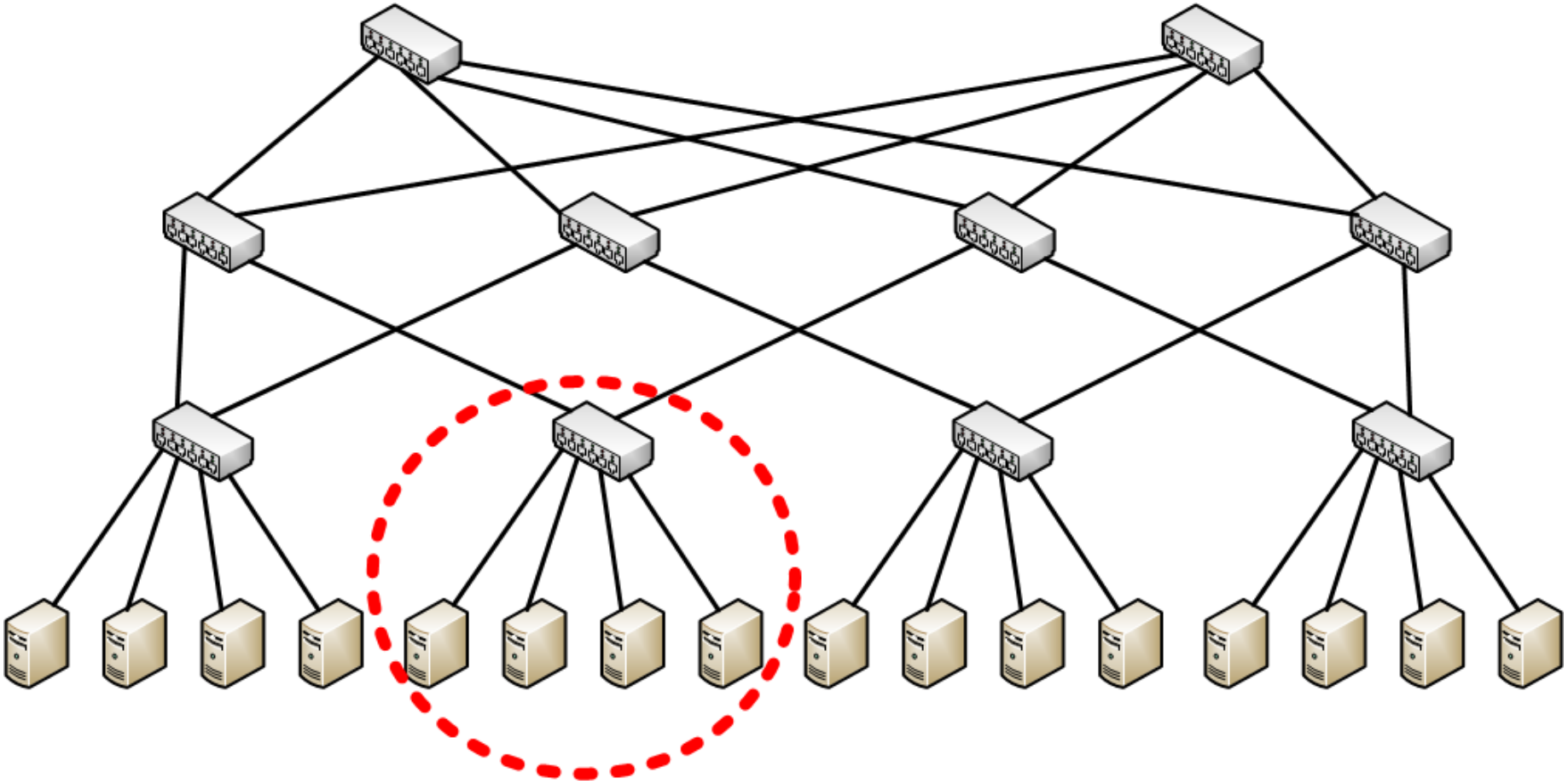
(datacenter servers nowadays ship with dual or quad port NICs)

- Multipath TCP

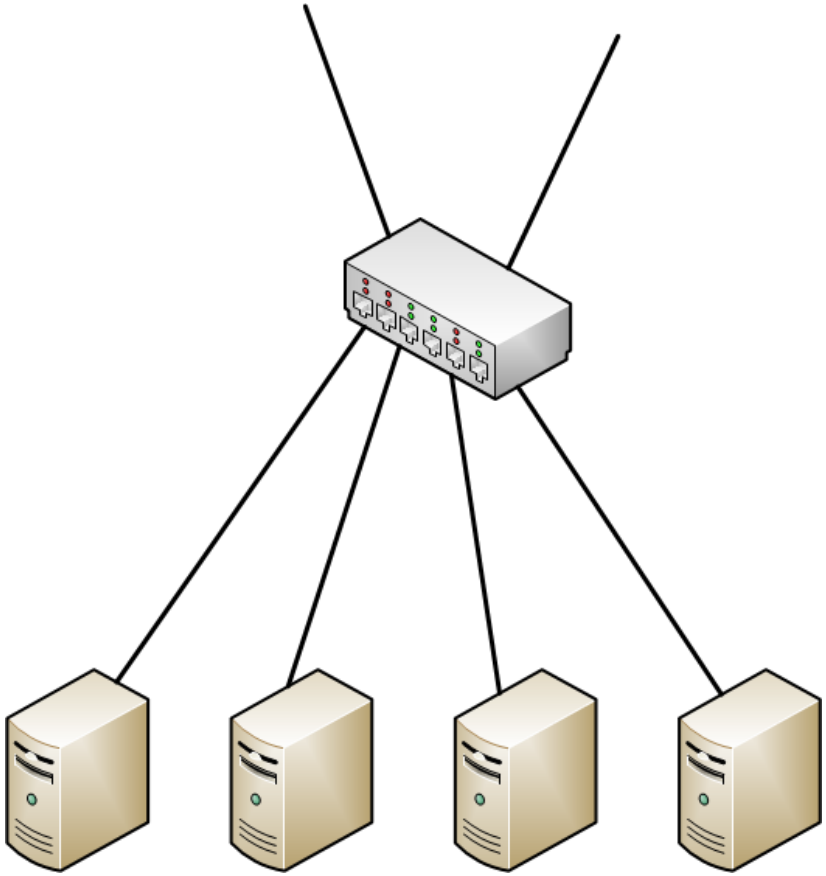


Solutions

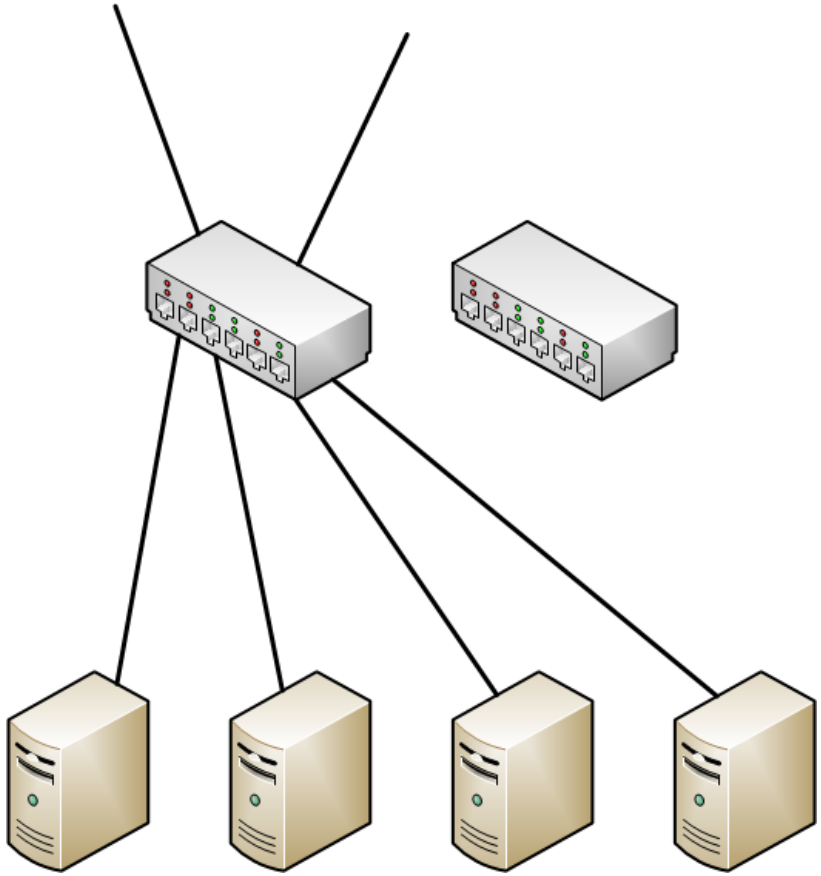
Multihoming



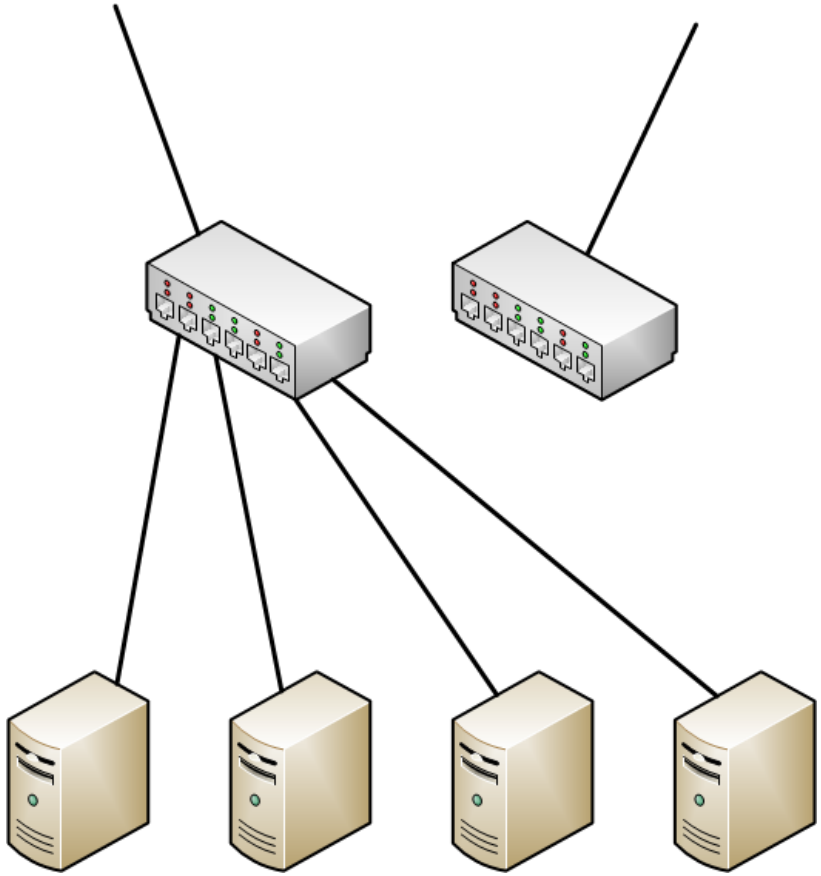
Multihoming



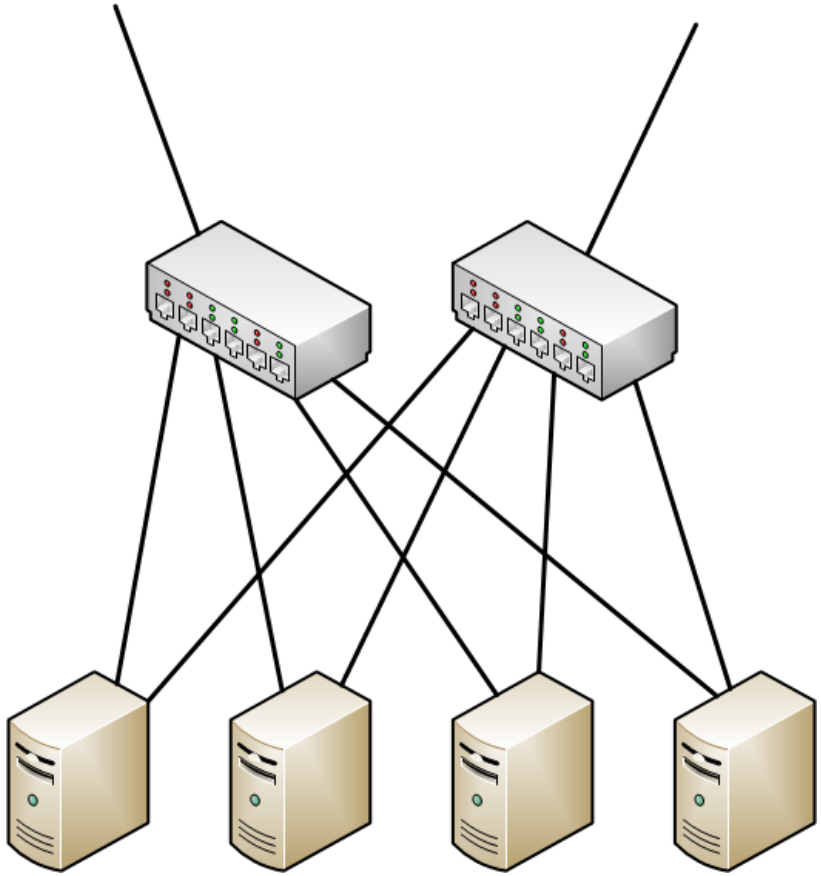
Multihoming



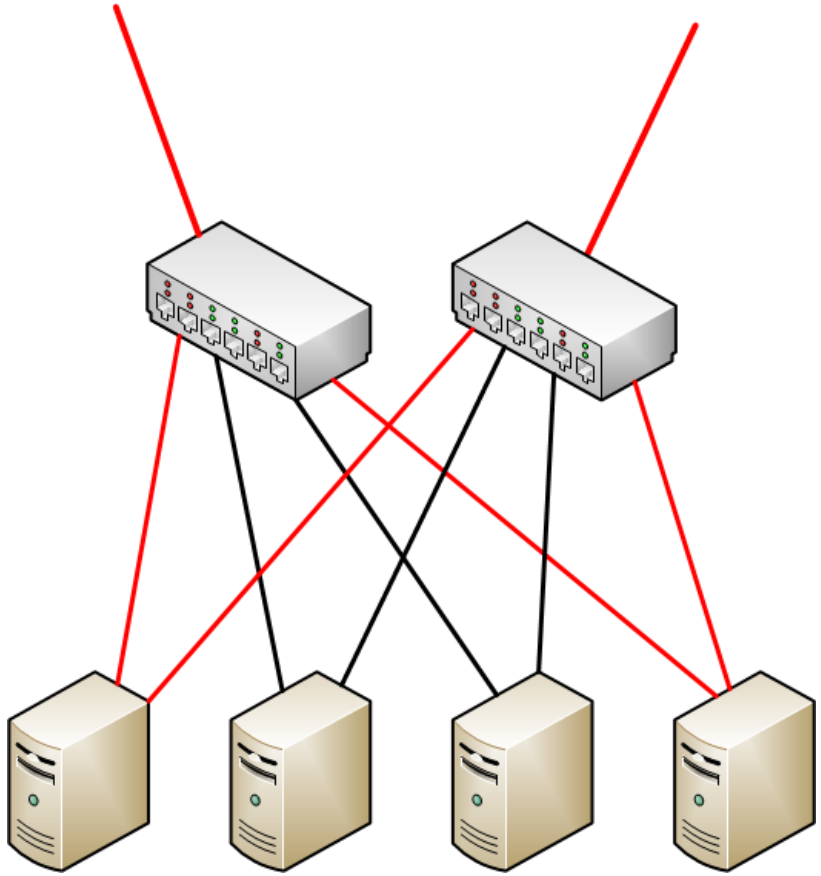
Multihoming



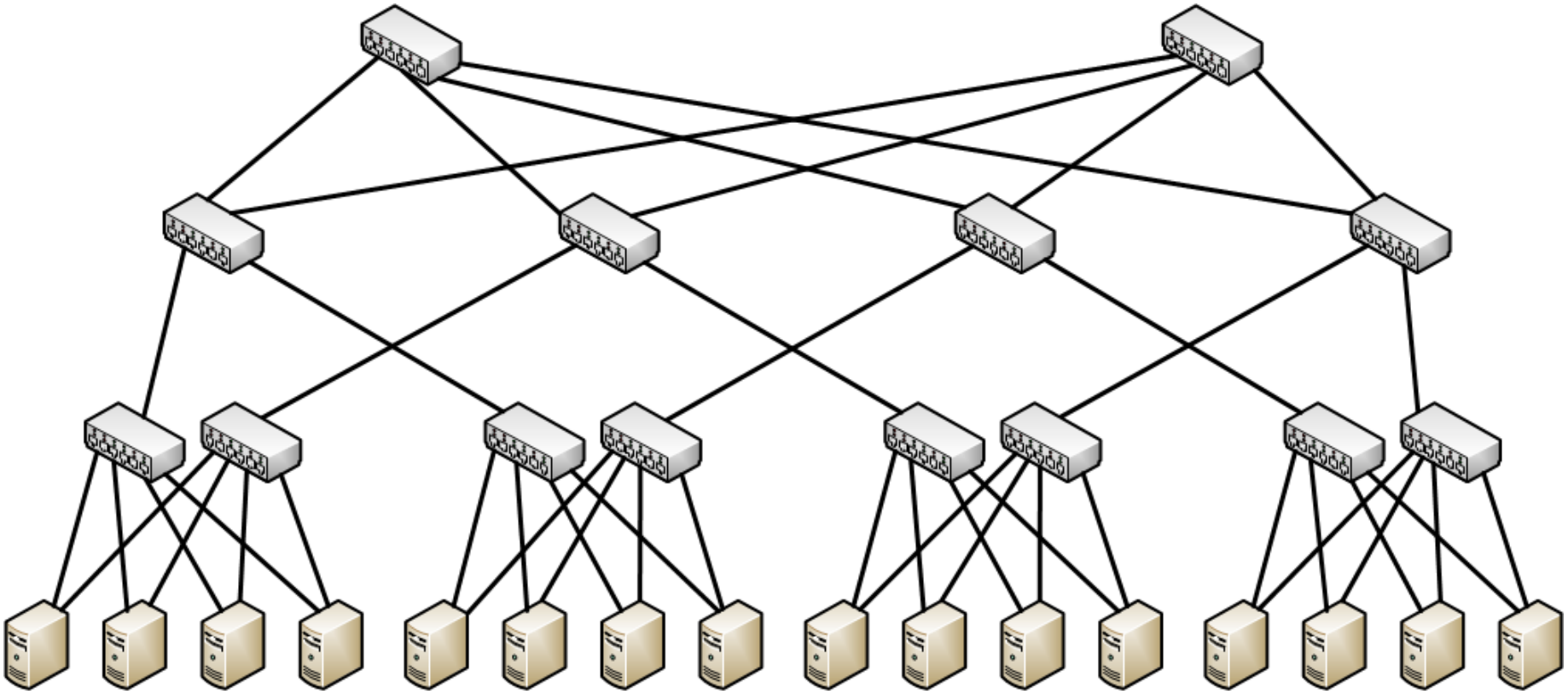
Multihoming



Multihoming



Multihoming

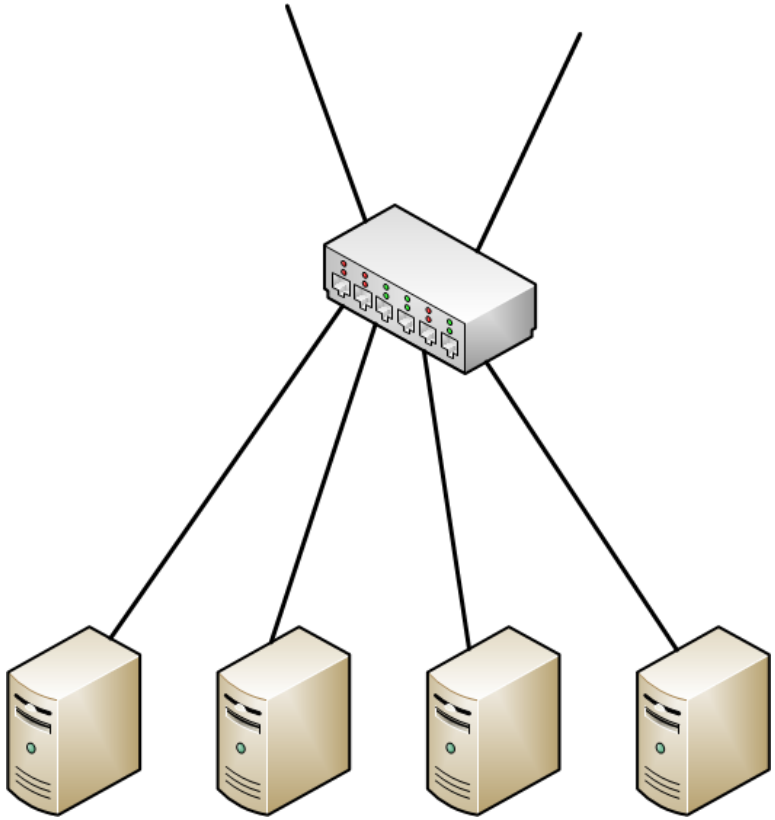


Multihoming

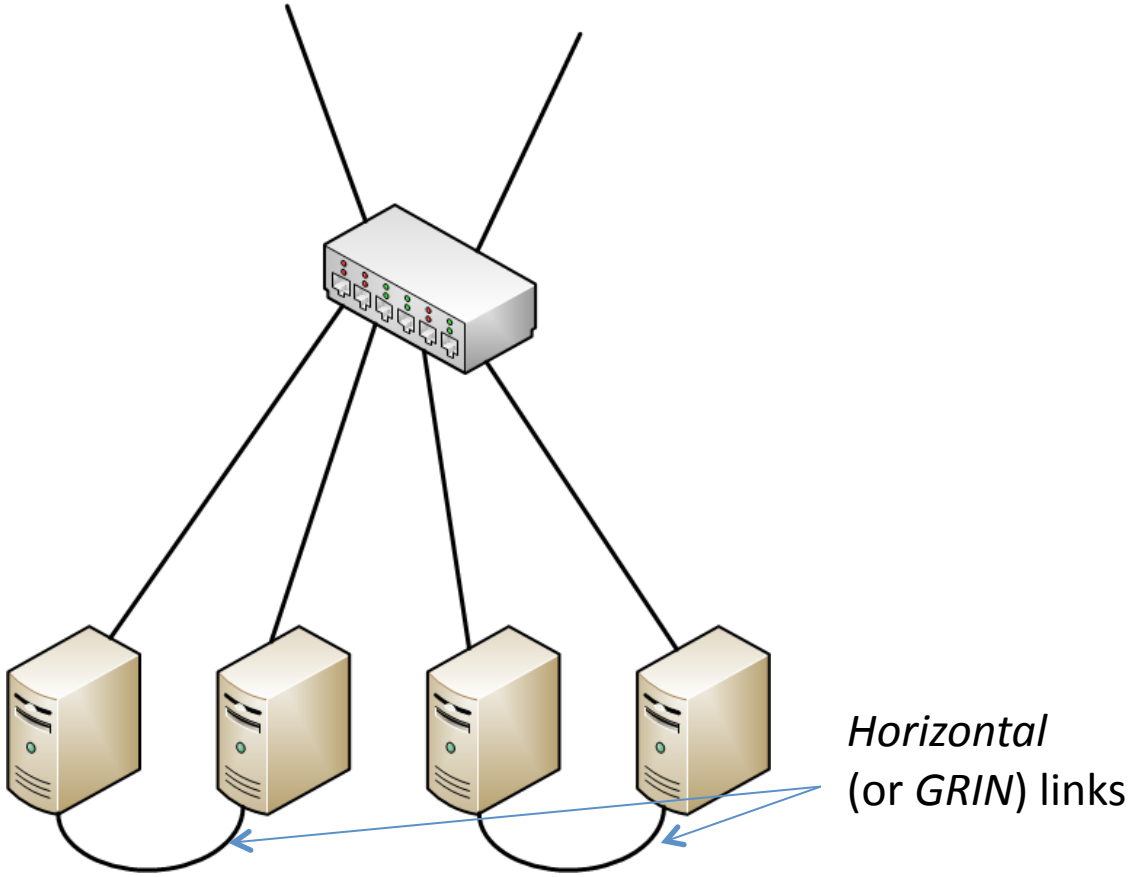
- Increased performance

- Rather expensive
  - needs additional switches
  - needs additional rack space
  - increases maintenance costs

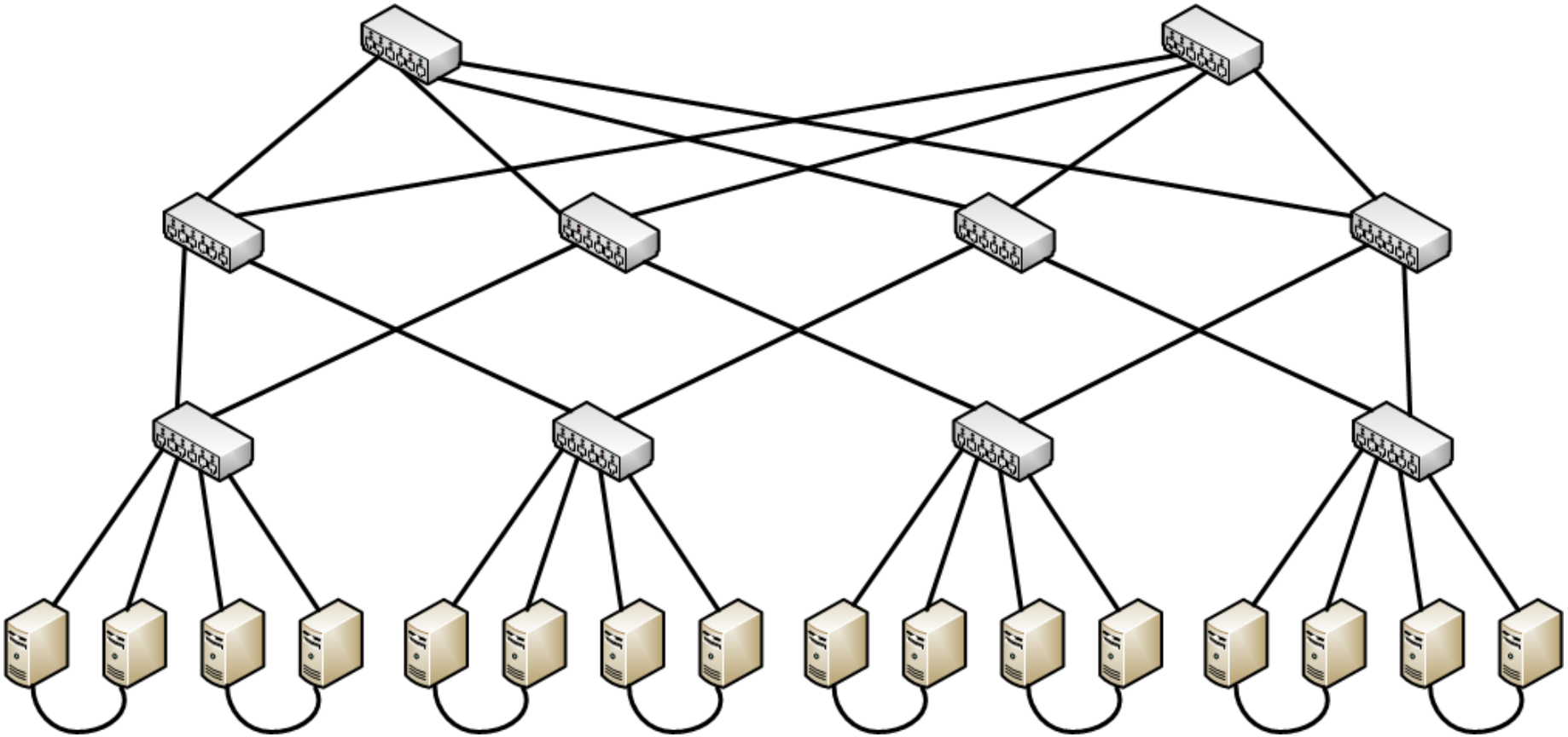
GRIN



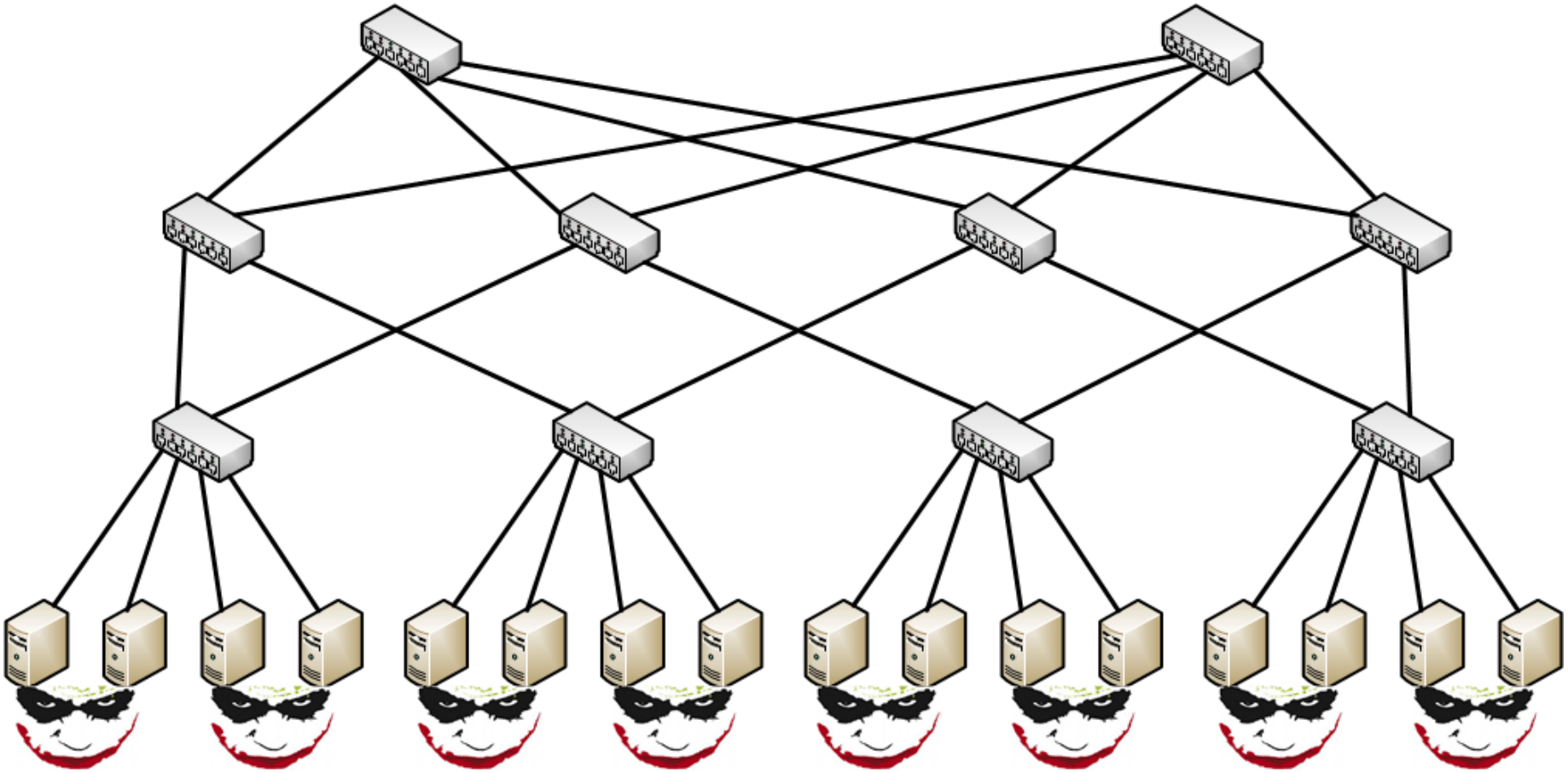
GRIN



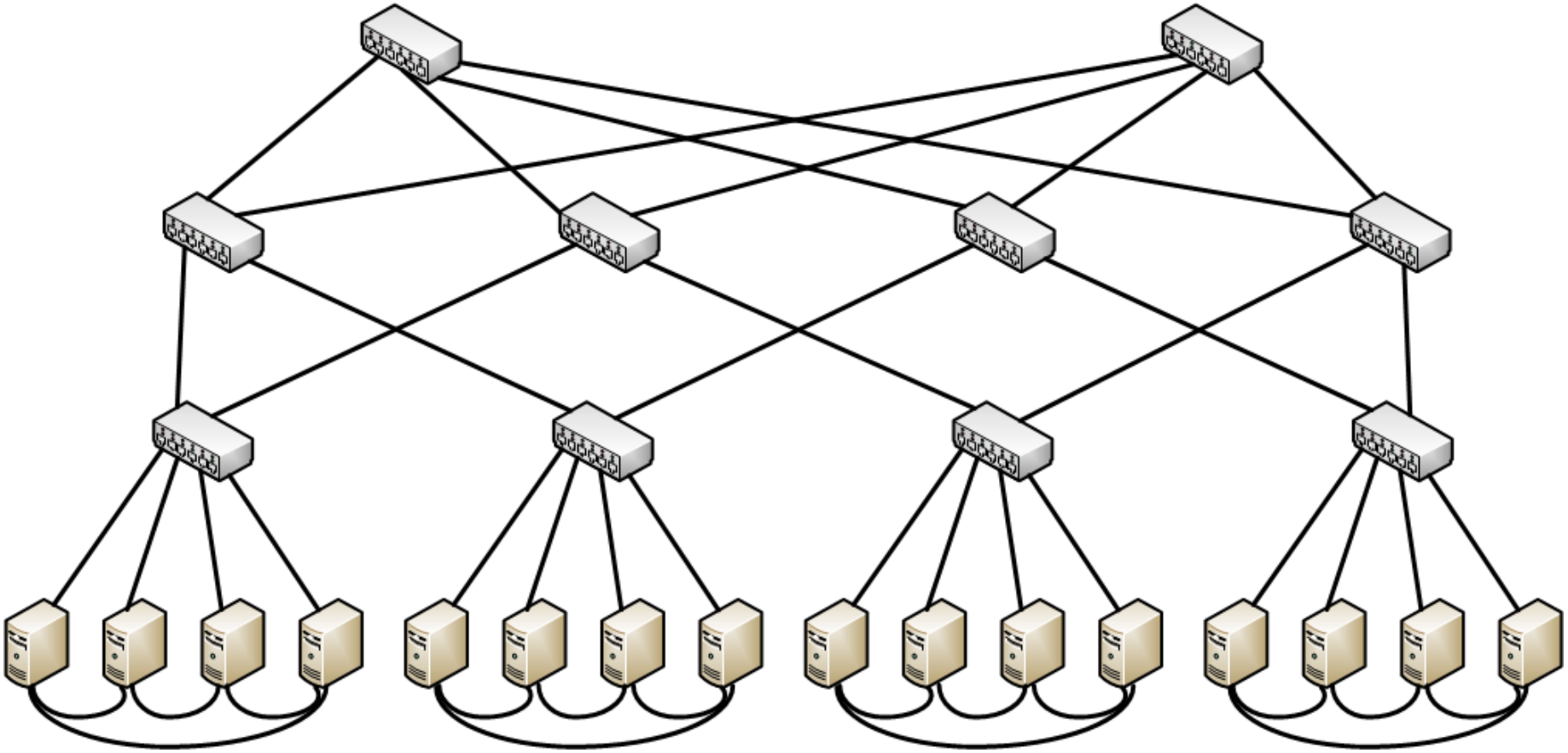
GRIN1



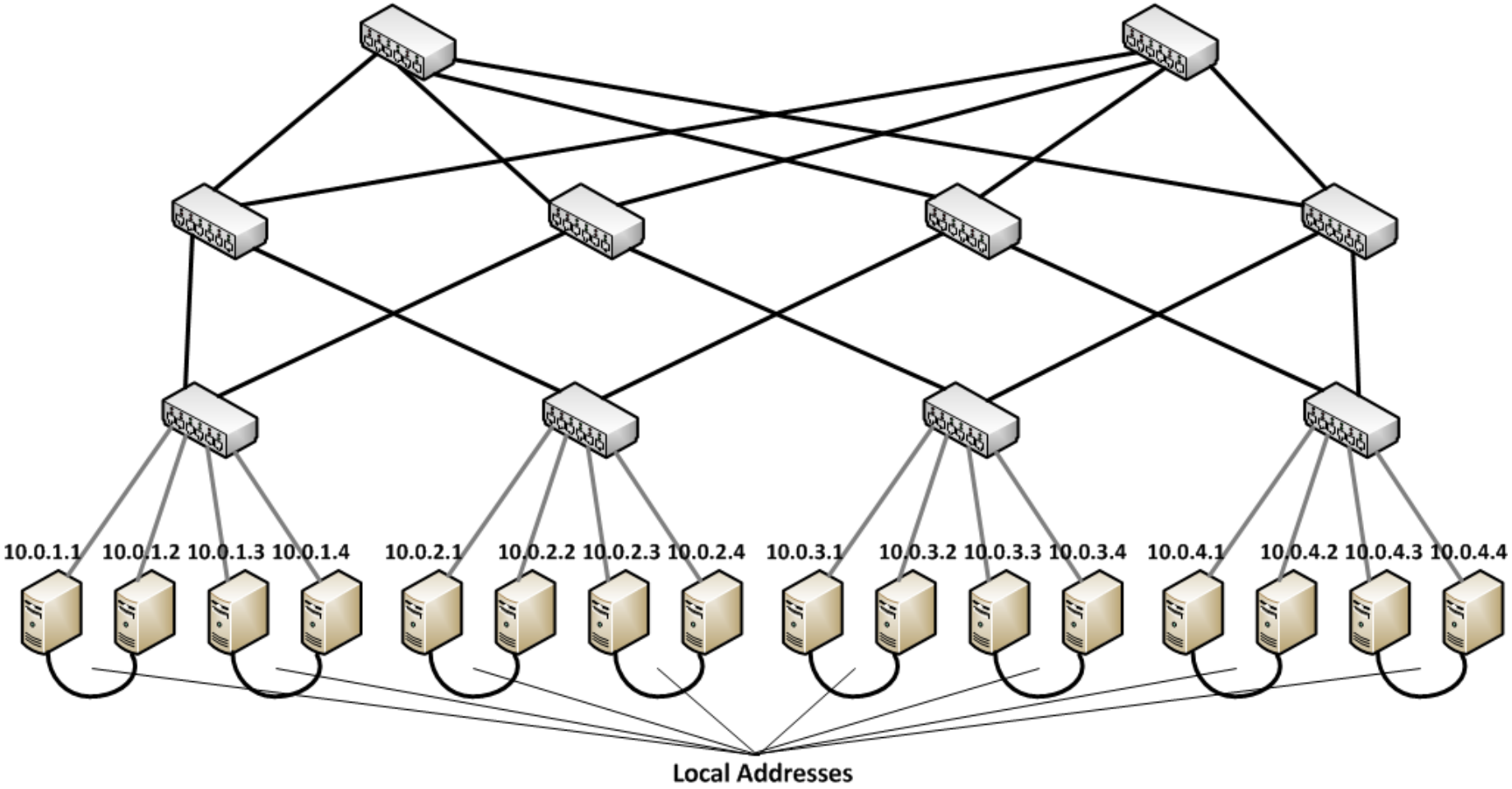
Why So Underutilized ?



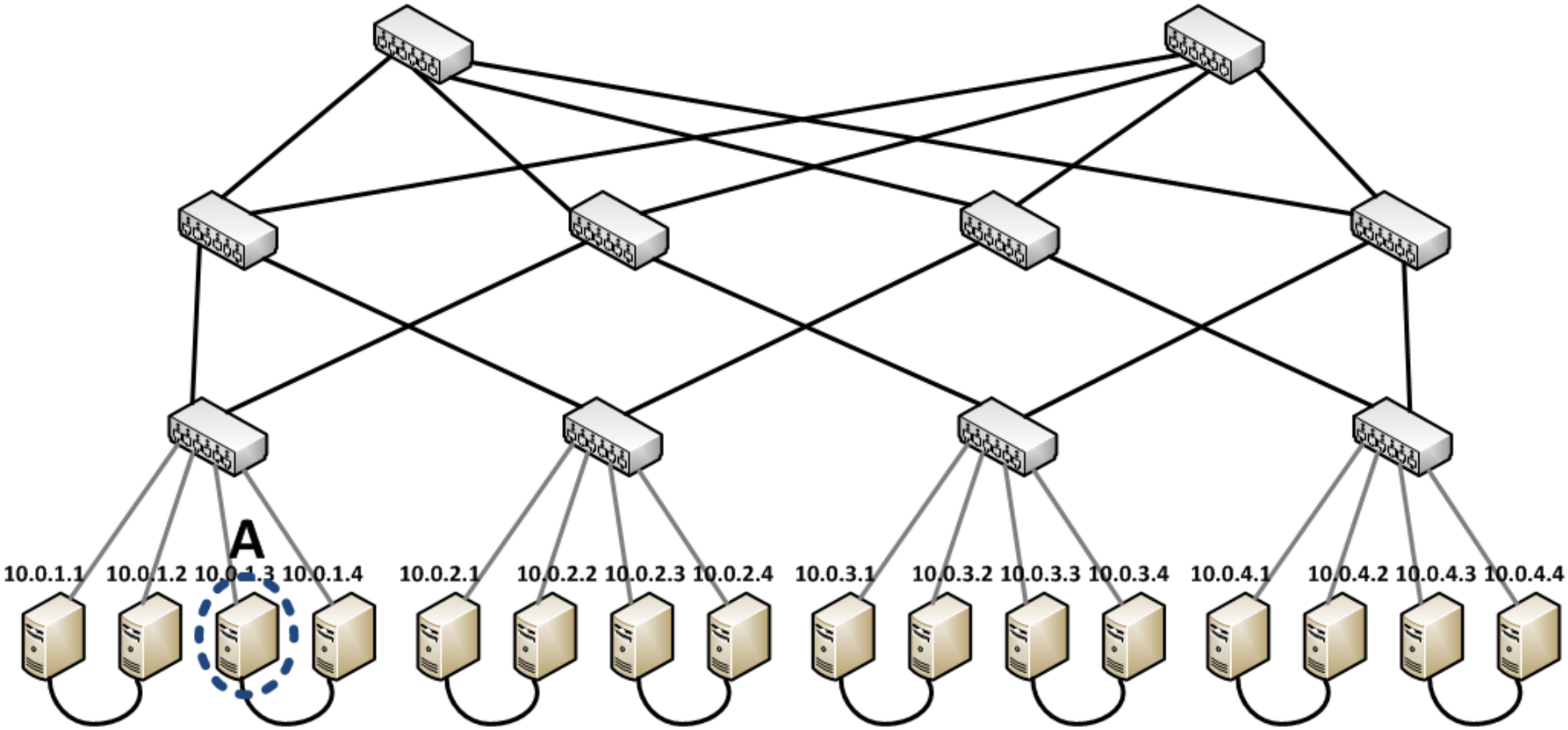
GRIN2



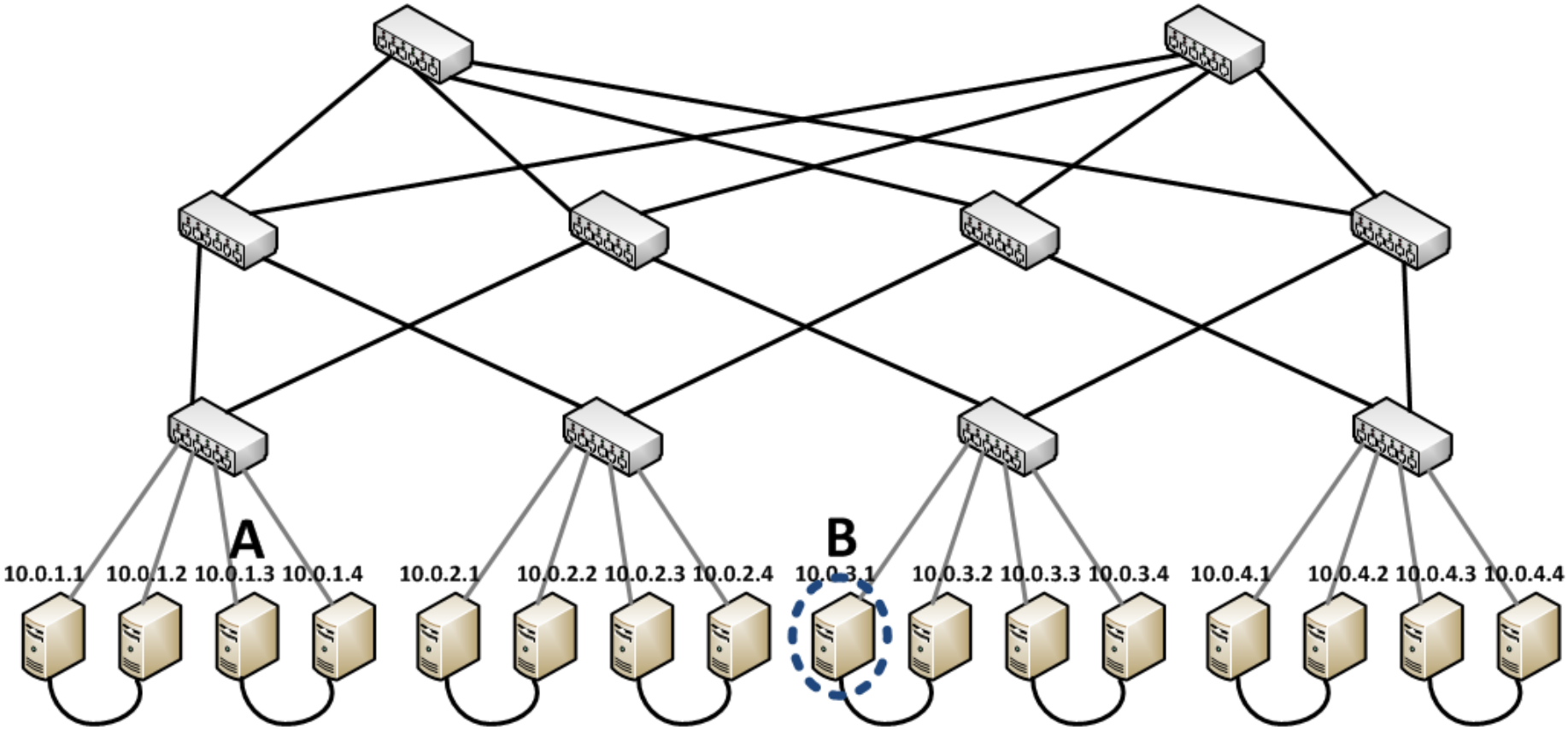
GRIN Routing Example



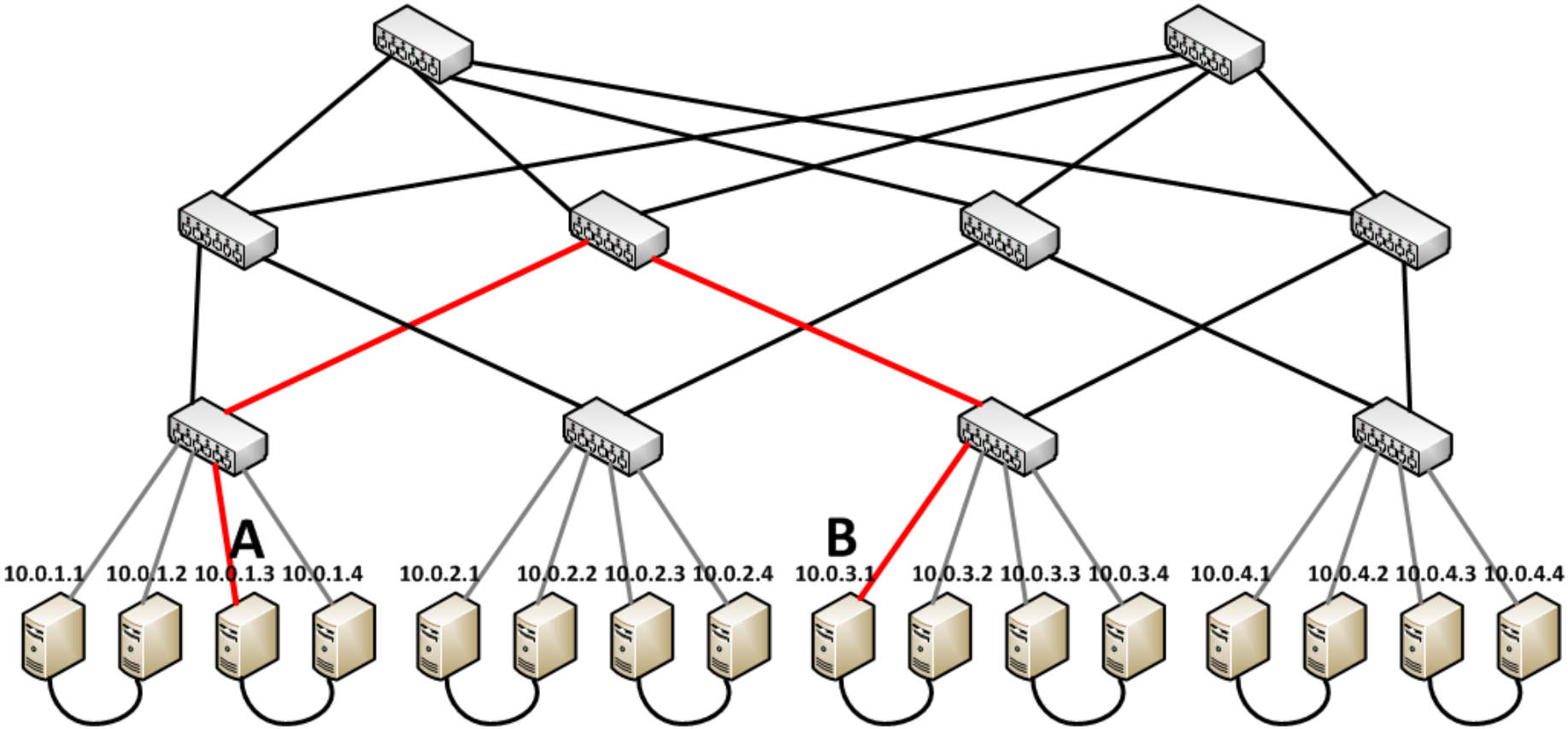
GRIN Routing Example



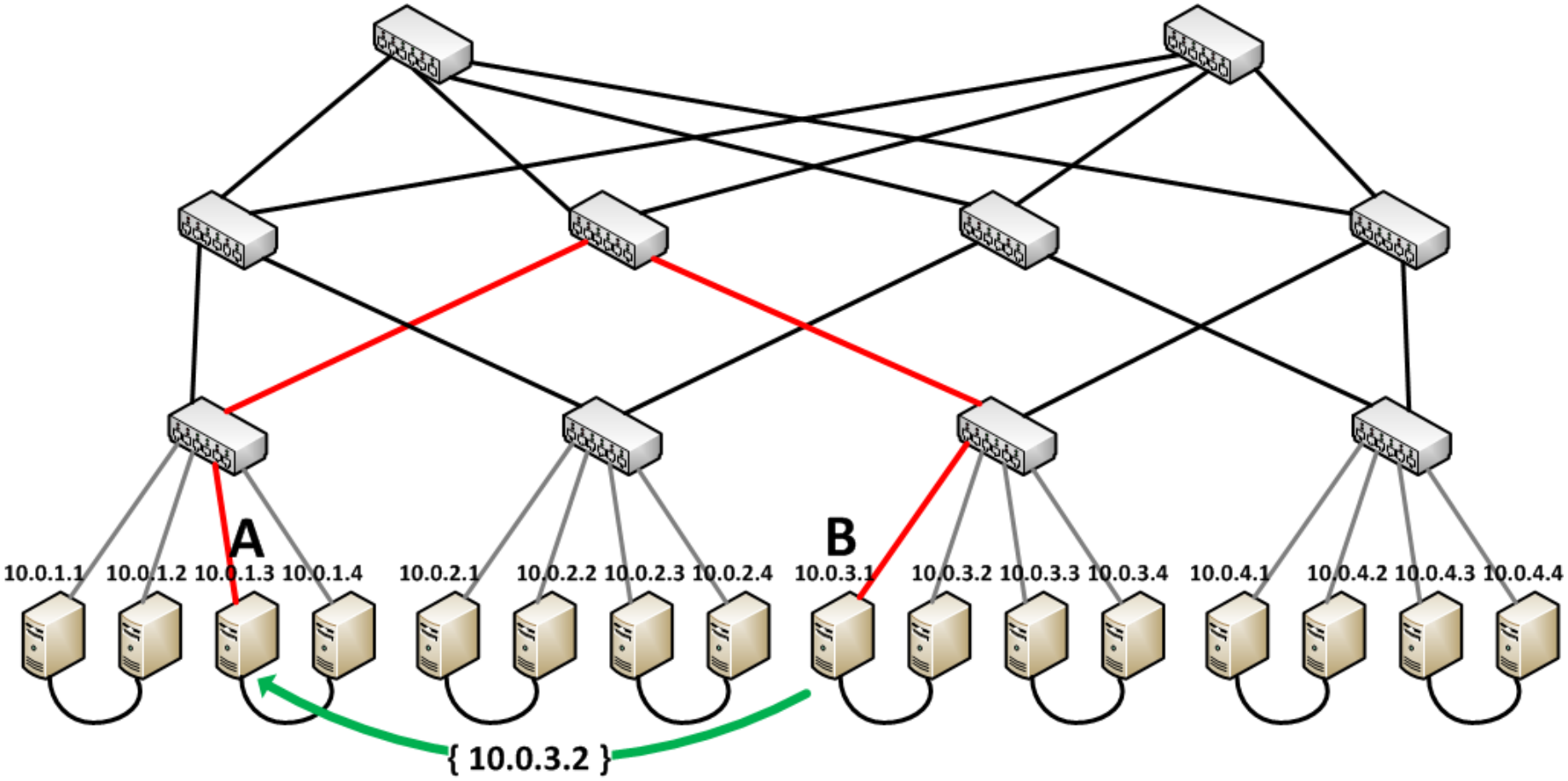
GRIN Routing Example



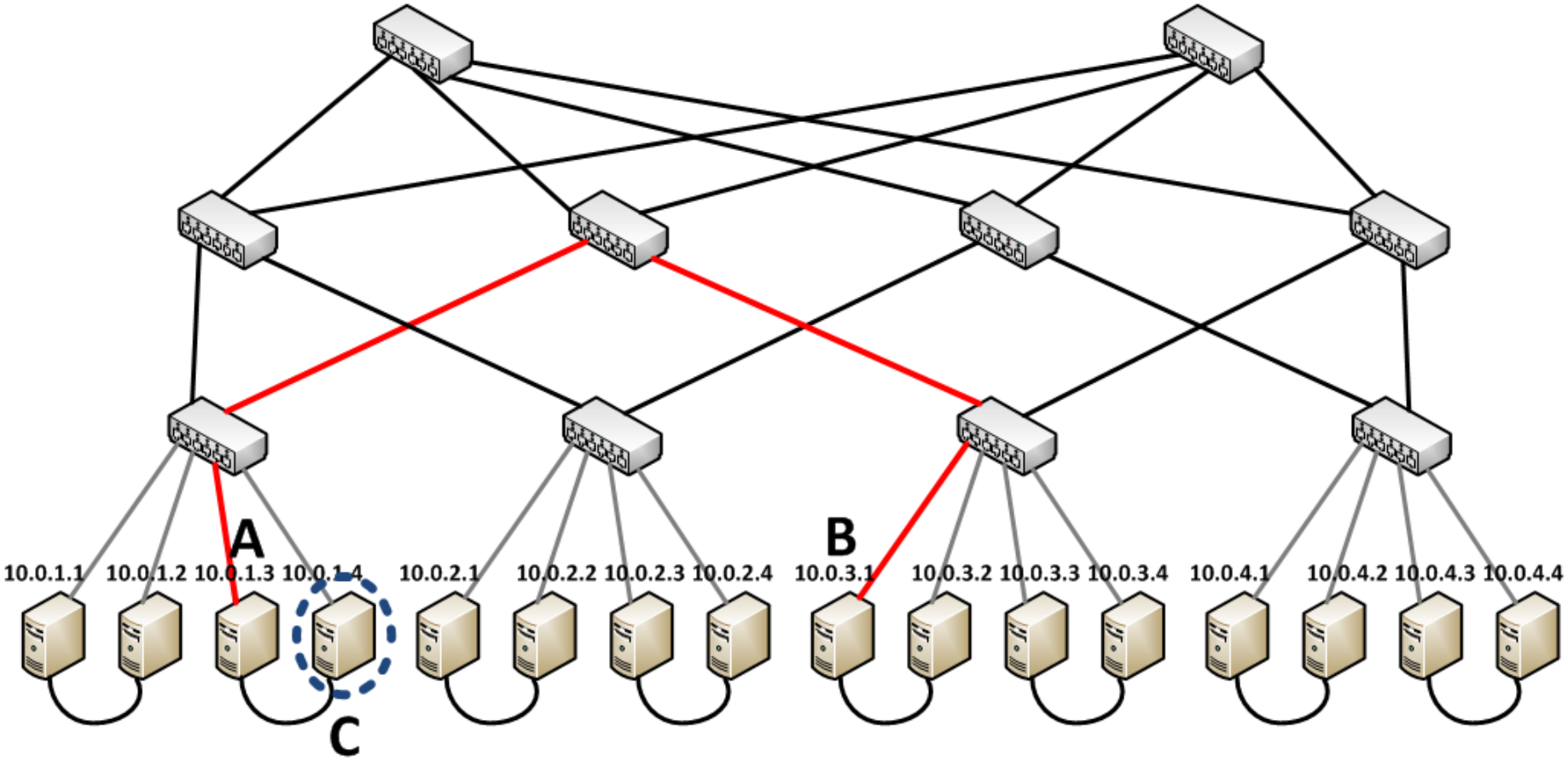
GRIN Routing Example



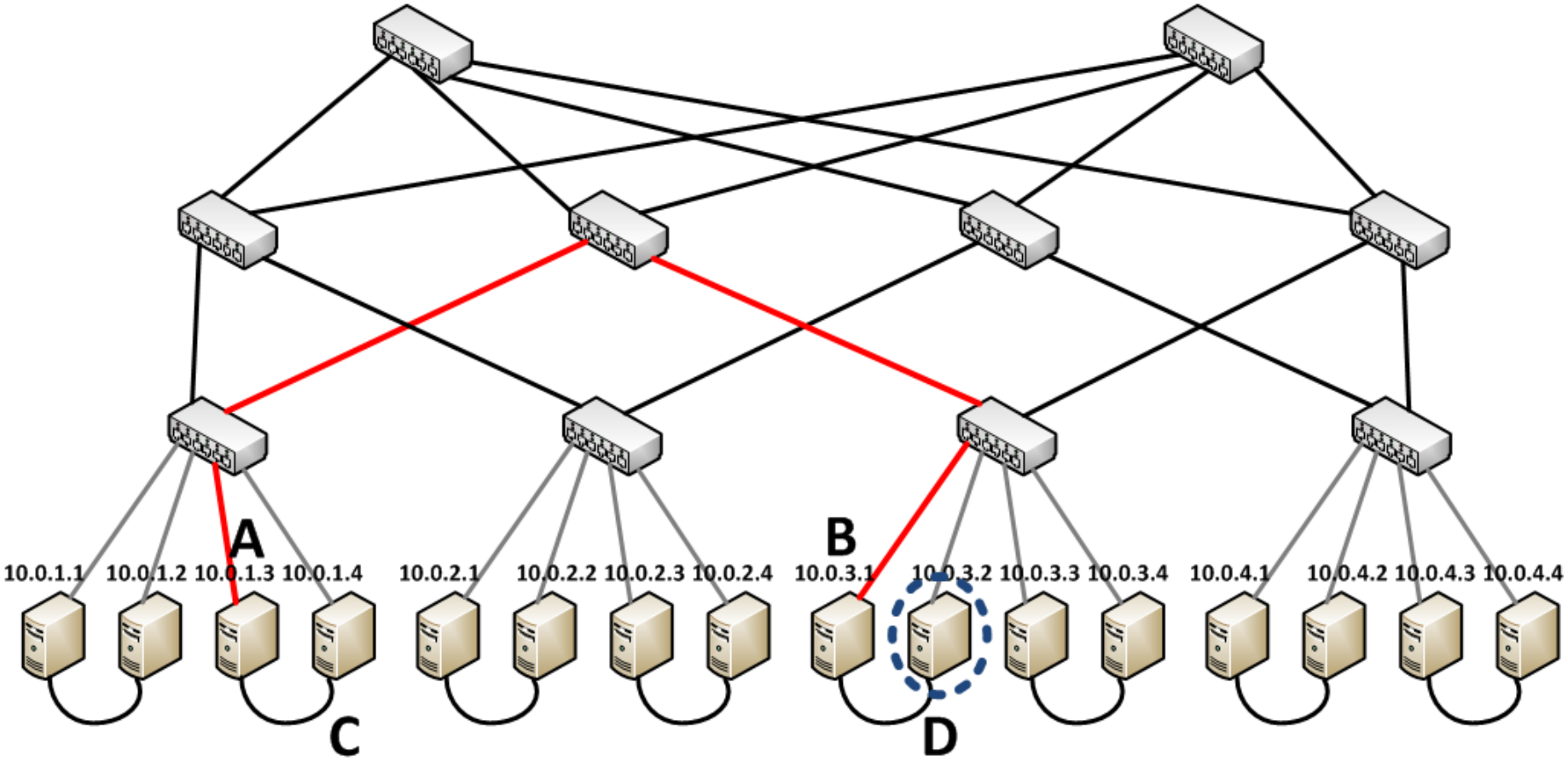
GRIN Routing Example



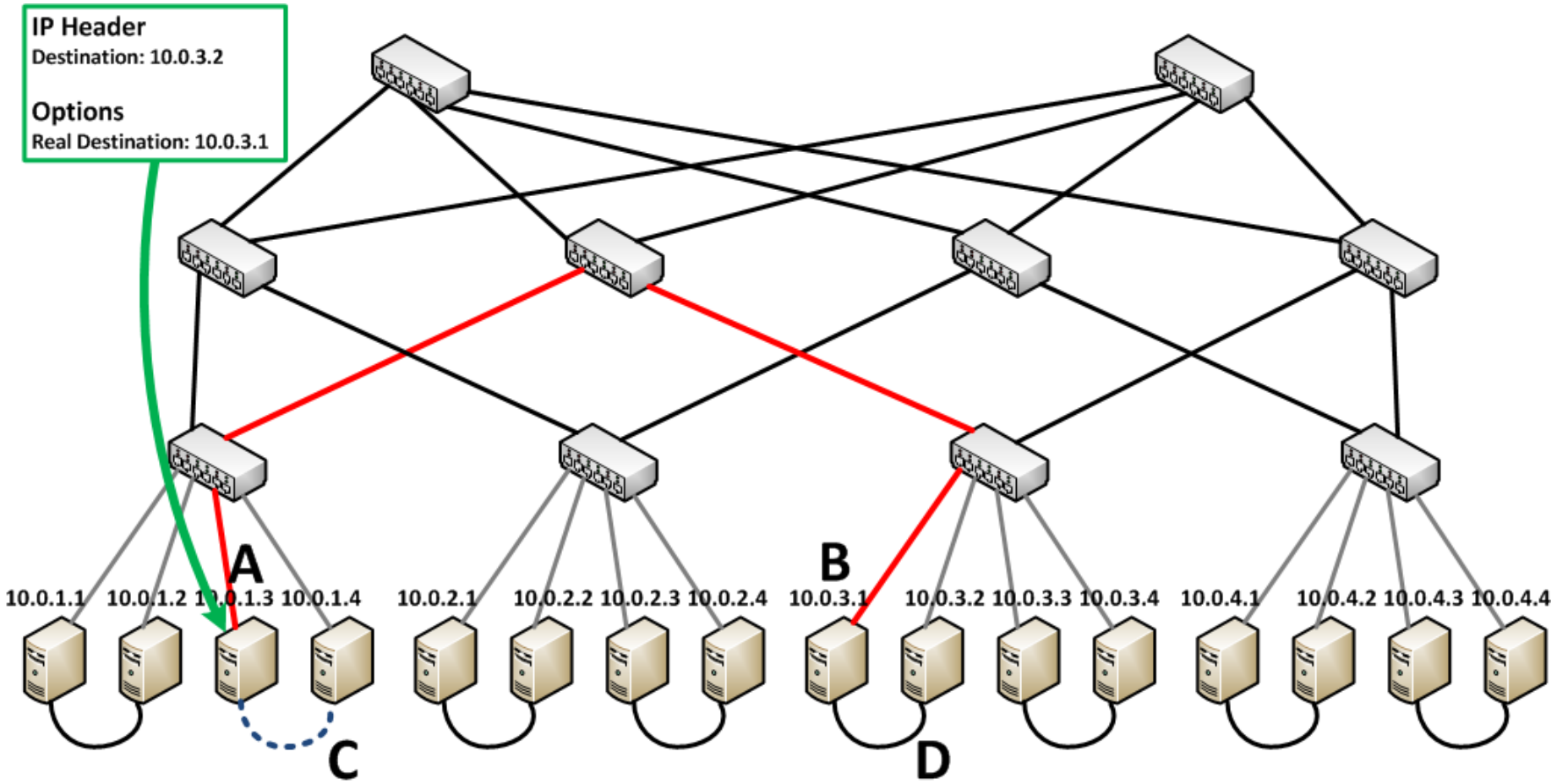
GRIN Routing Example



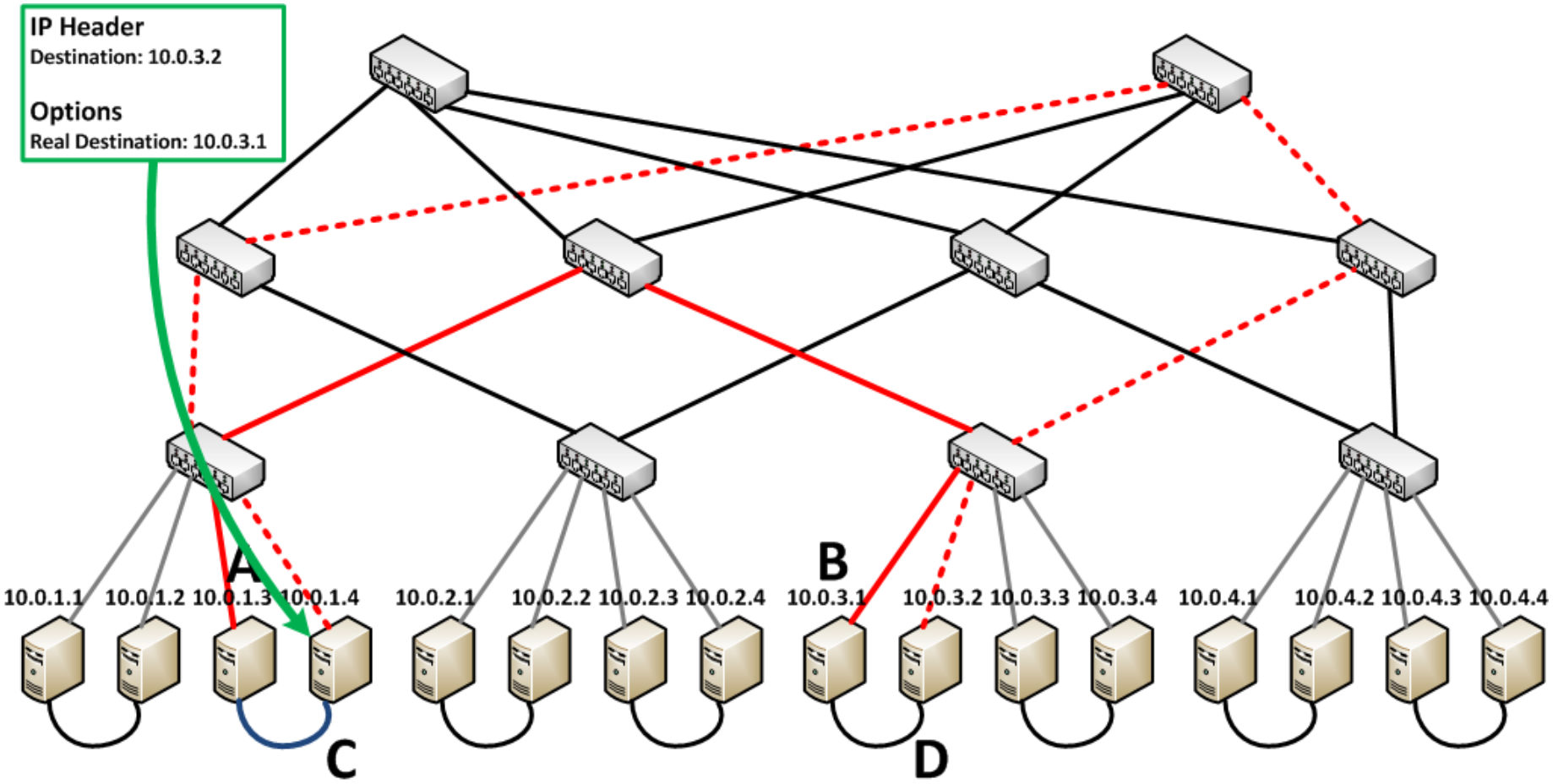
GRIN Routing Example



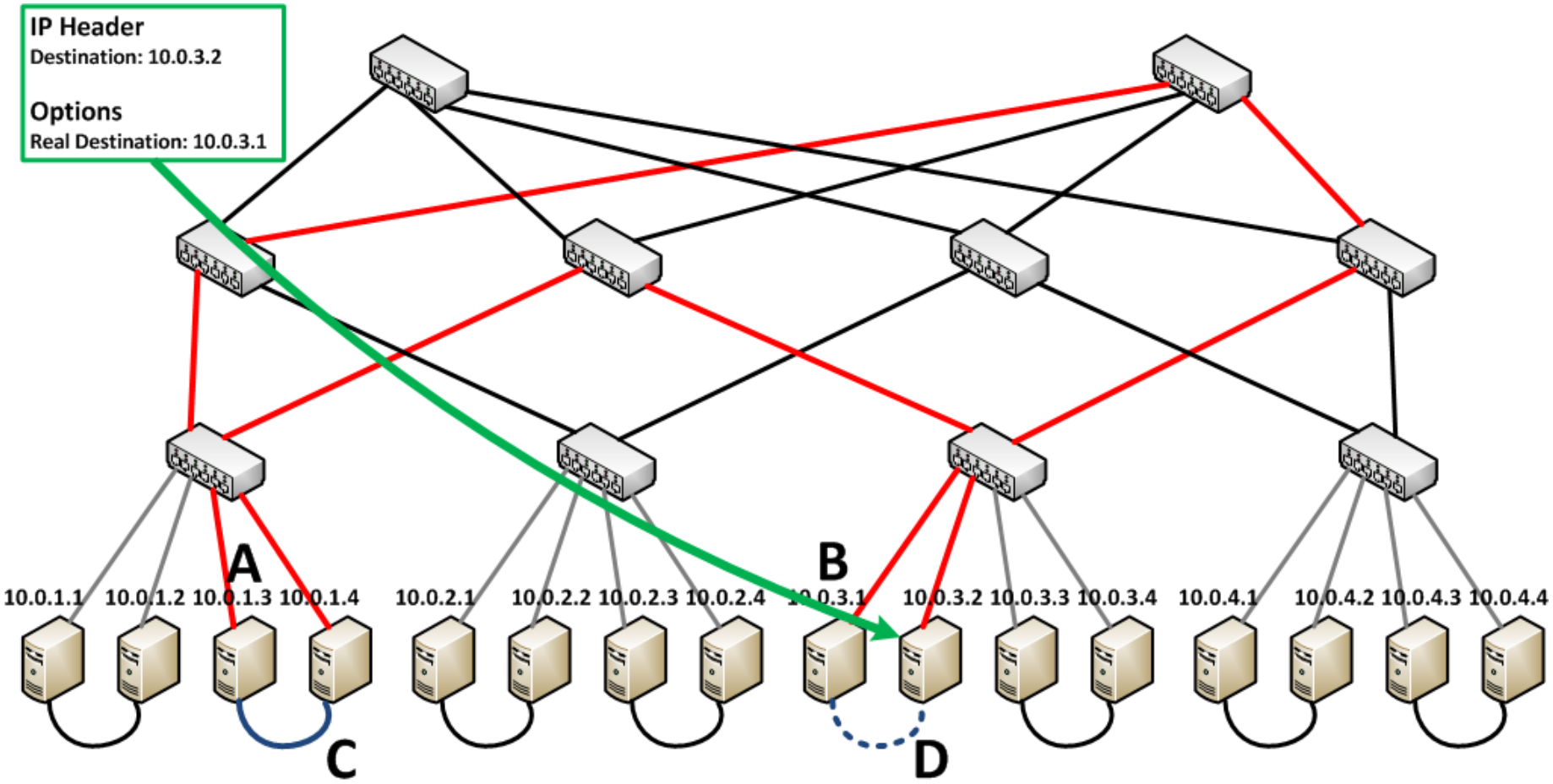
GRIN Routing Example



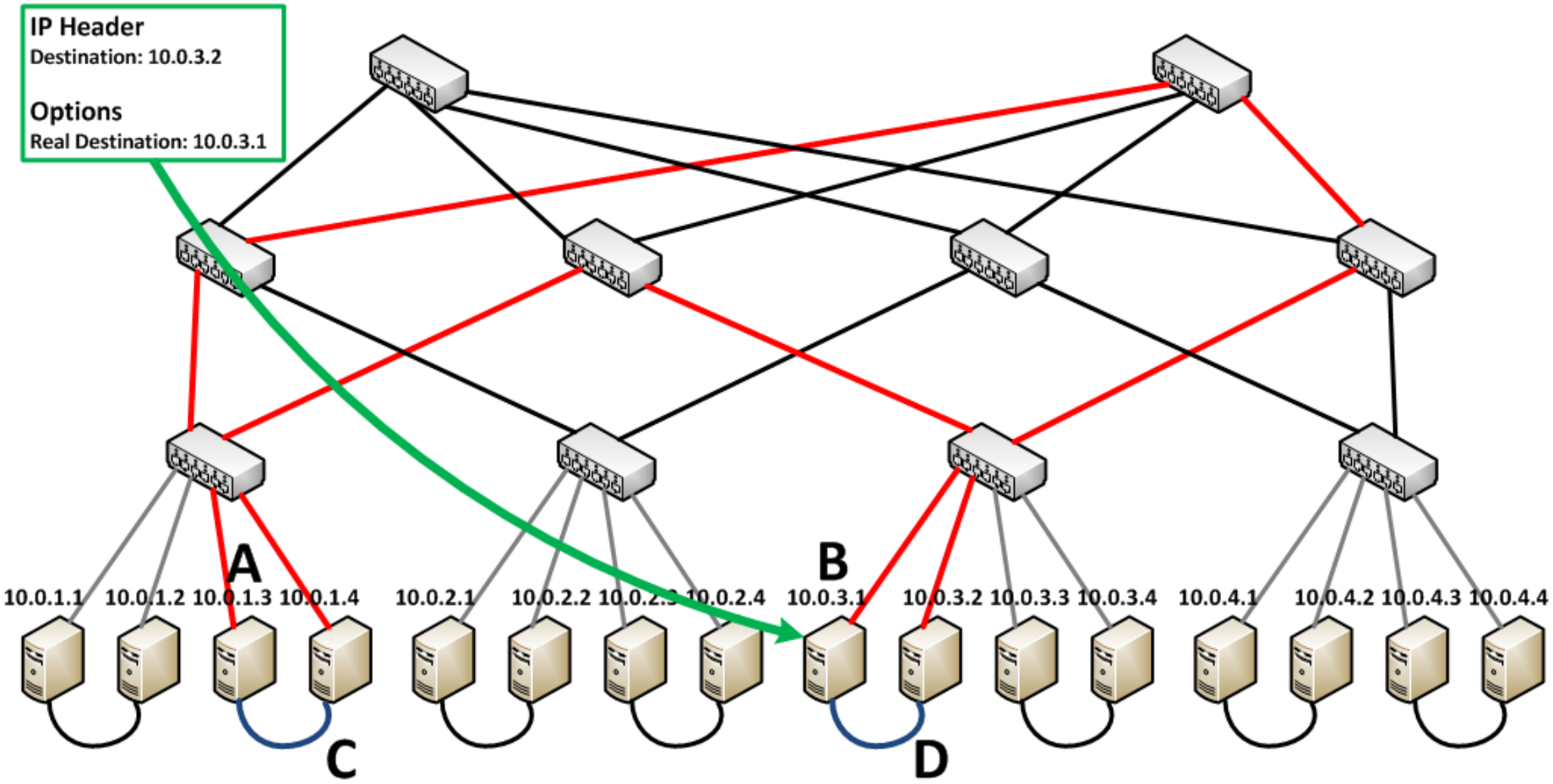
GRIN Routing Example



GRIN Routing Example



GRIN Routing Example



GRIN Routing

How do we determine a path from server A to server B ?

GRIN Routing

How do we determine a path from server A to server B ?

-pick C = A or any of A's neighbors

GRIN Routing

How do we determine a path from server A to server B ?

-pick C = A or any of A's neighbors

-pick D = B or any of B's neighbors

GRIN Routing

How do we determine a path from server A to server B ?

-pick C = A or any of A's neighbors

-pick D = B or any of B's neighbors

-pick a path ***p*** between C and D from the original topology

GRIN Routing

How do we determine a path from server A to server B ?

-pick C = A or any of A's neighbors

-pick D = B or any of B's neighbors

-pick a path ***p*** between C and D from the original topology

The GRIN path is going to be A → C ***p*** D → B.

(as long as A, B, C and D represent distinct servers)

GRIN

- Increases performance
- Cheap and easy to deploy
  - we just need to wire up existing ports

- Significant server forwarding overhead ?

GRIN

- Increases performance
- Cheap and easy to deploy

- we just need to wire up existing ports

- ~~Significant server forwarding overhead ?~~

(when forwarding 1400B packets at full speed, in both directions, we utilize 10% of a single core)

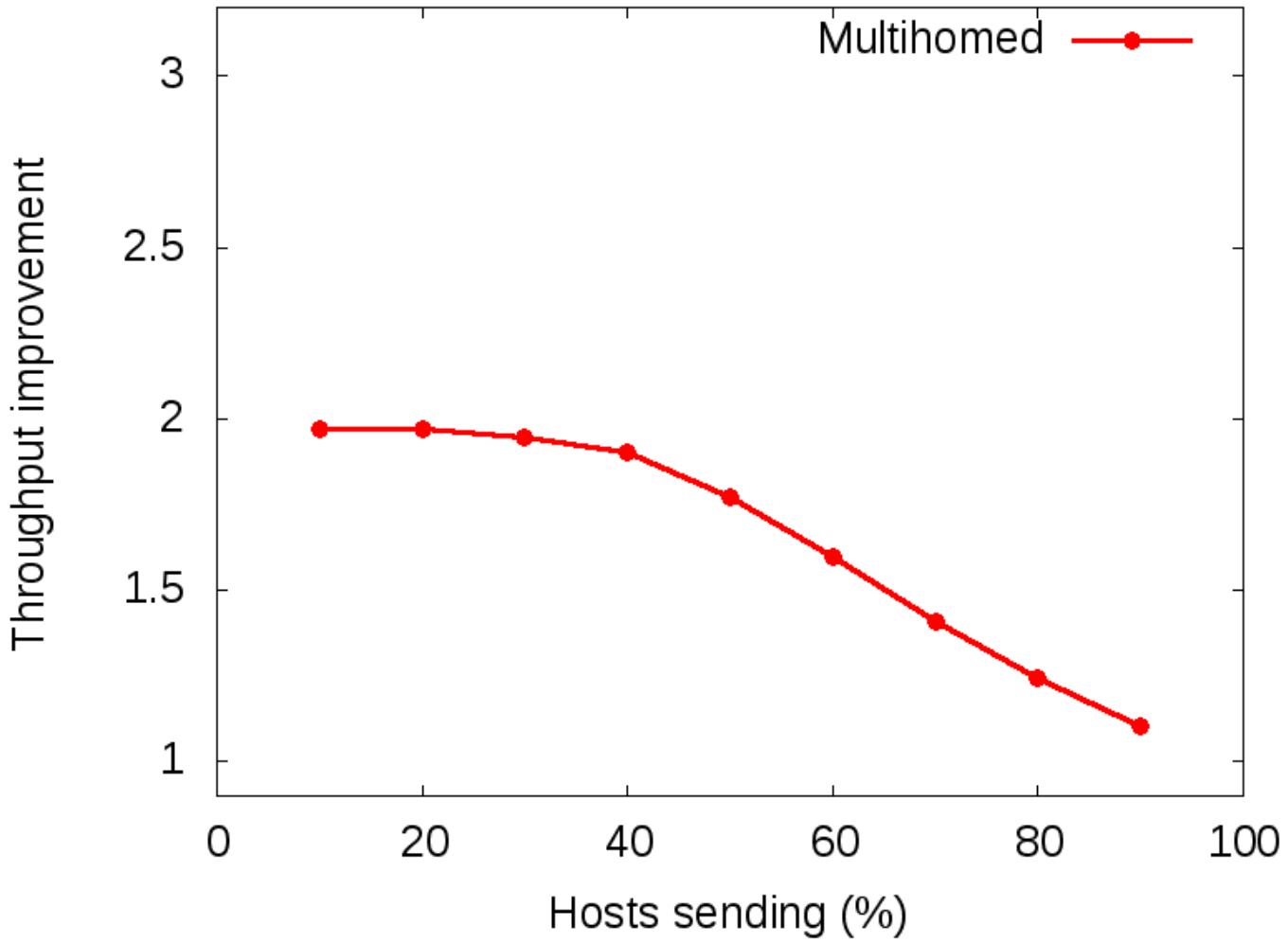
Evaluation

- Simulated Vl2 topology consisting of 120 servers arranged in 6 racks
- One multihomed topology (using a single additional switch) vs. GRIN1, GRIN2 and GRIN3
- Long flows
- Non-local connections only
- One connection matrix for each experiment

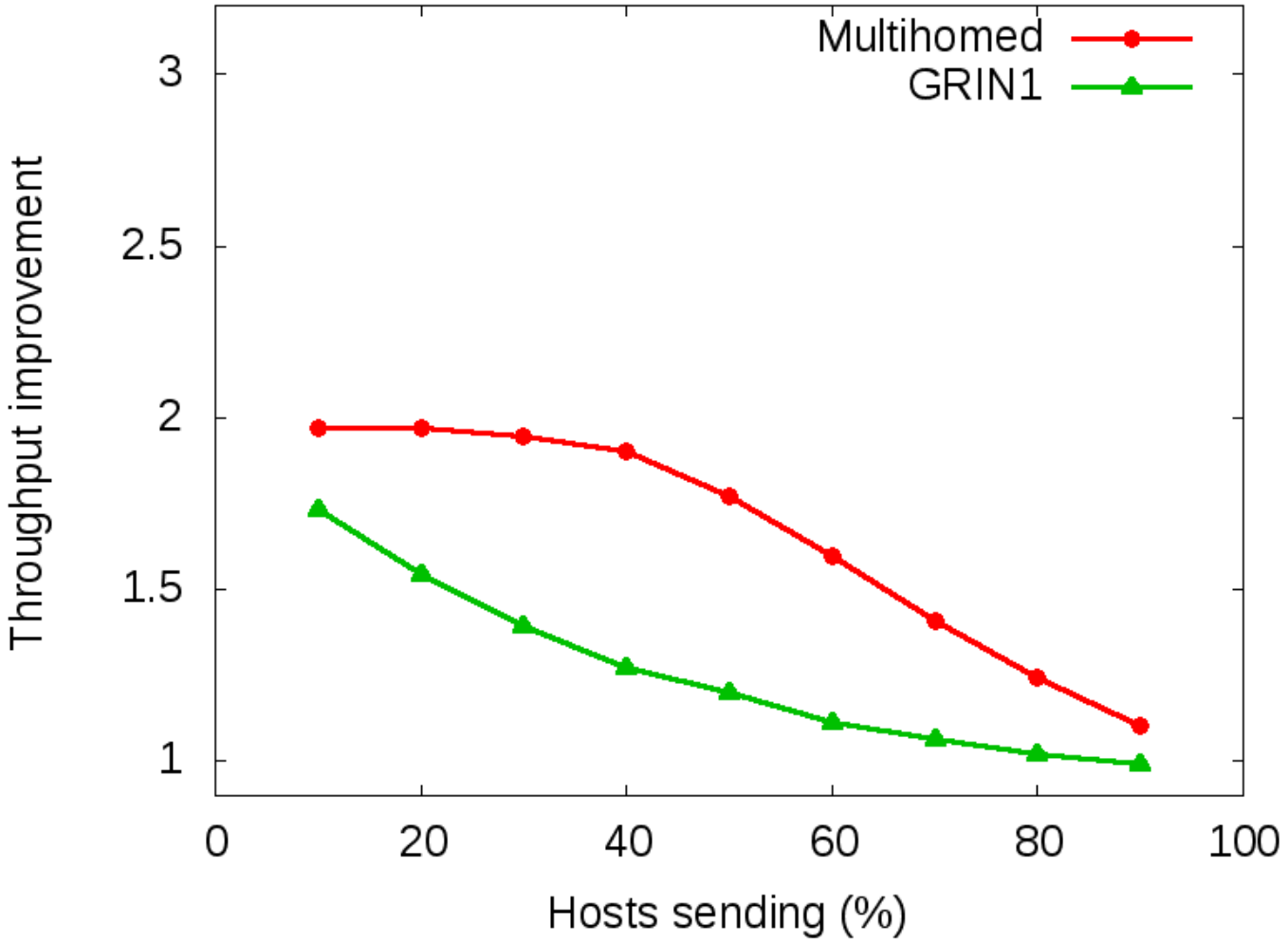
Permutation Traffic Matrix

- Each active server picks a single destination at random and tries to send data at the fastest possible rate
- Each destination may only receive data from a single sender
- Different numbers of active servers for different experiments

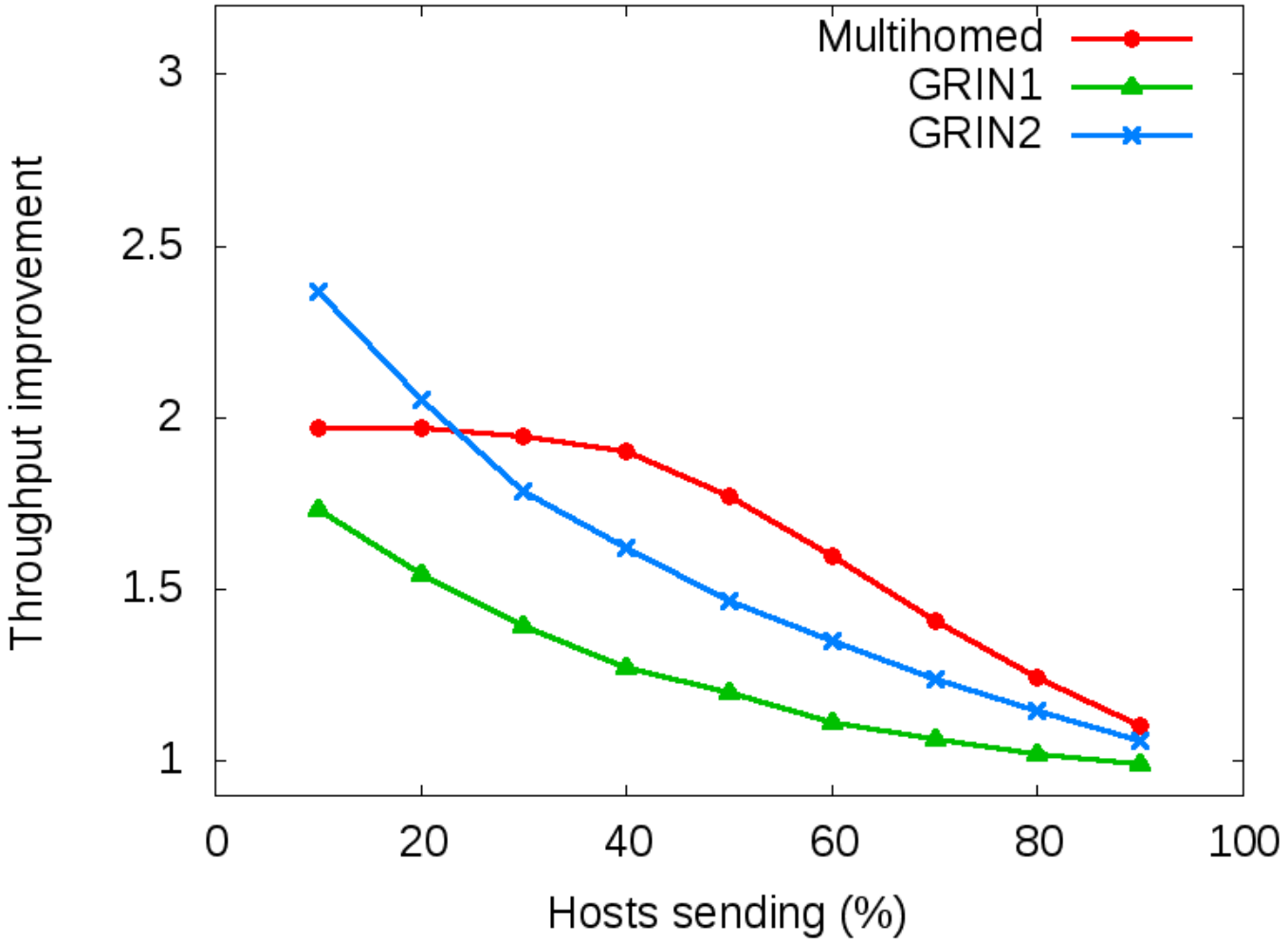
Permutation Traffic Matrix



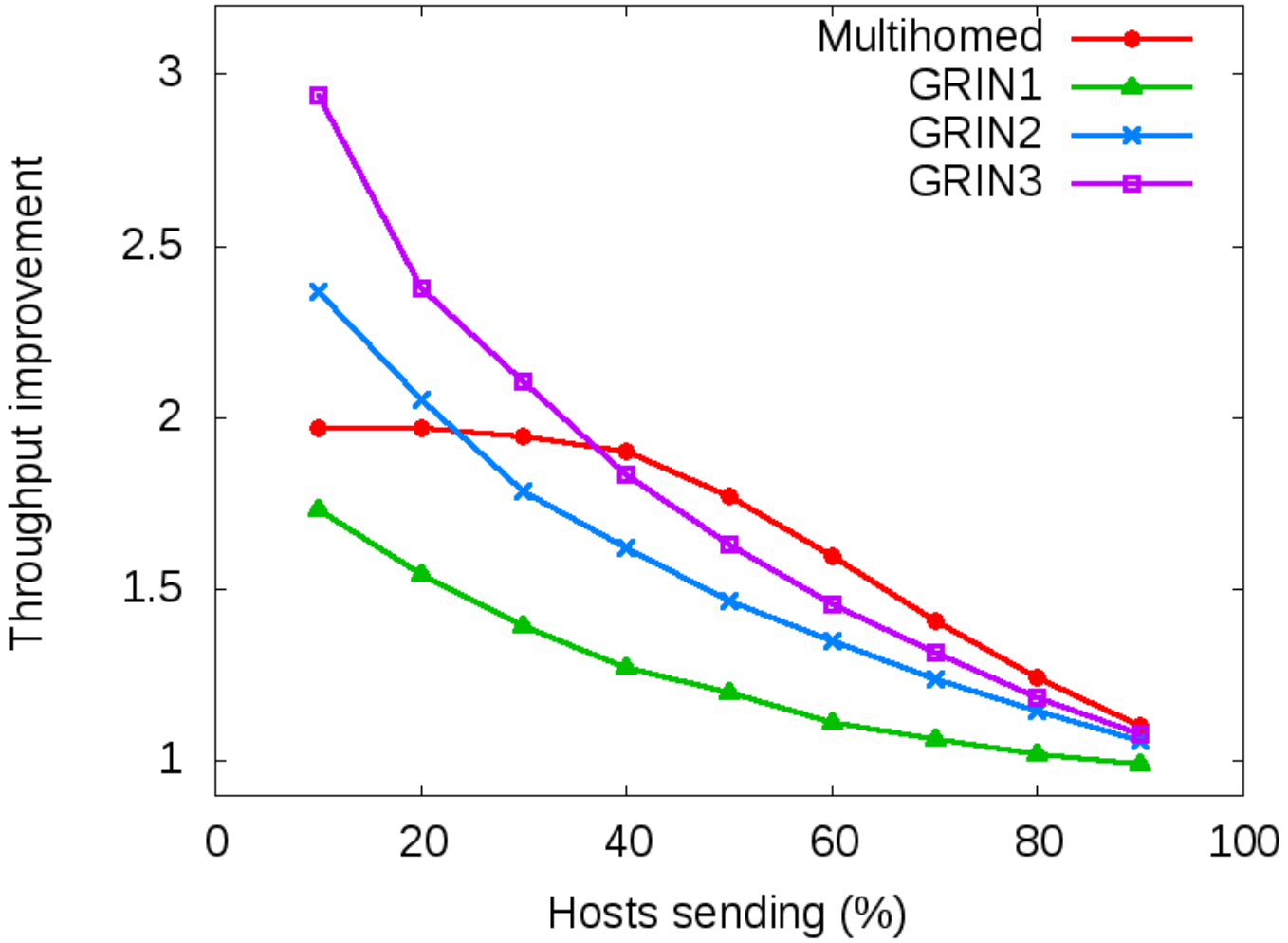
Permutation Traffic Matrix



Permutation Traffic Matrix



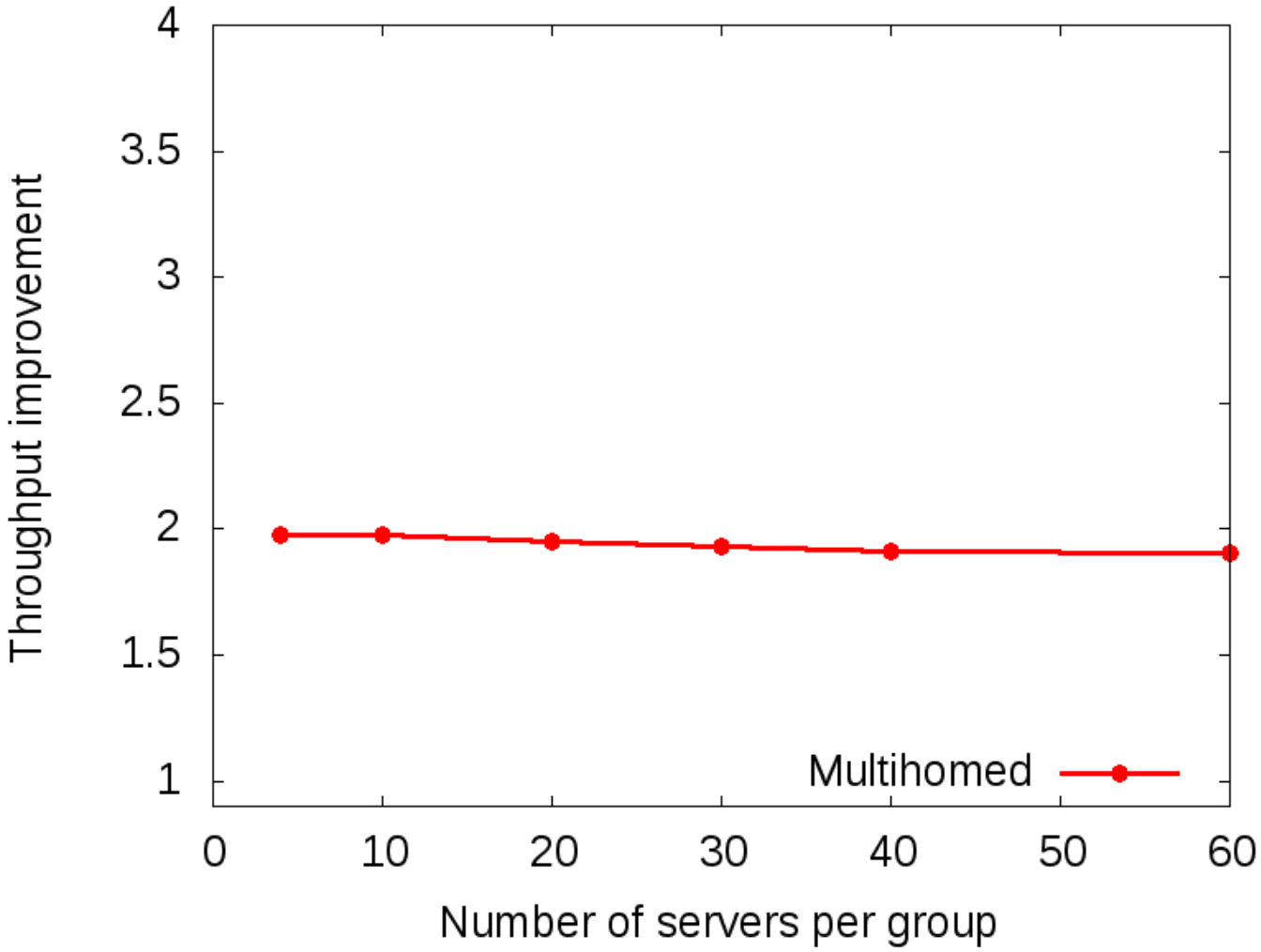
Permutation Traffic Matrix



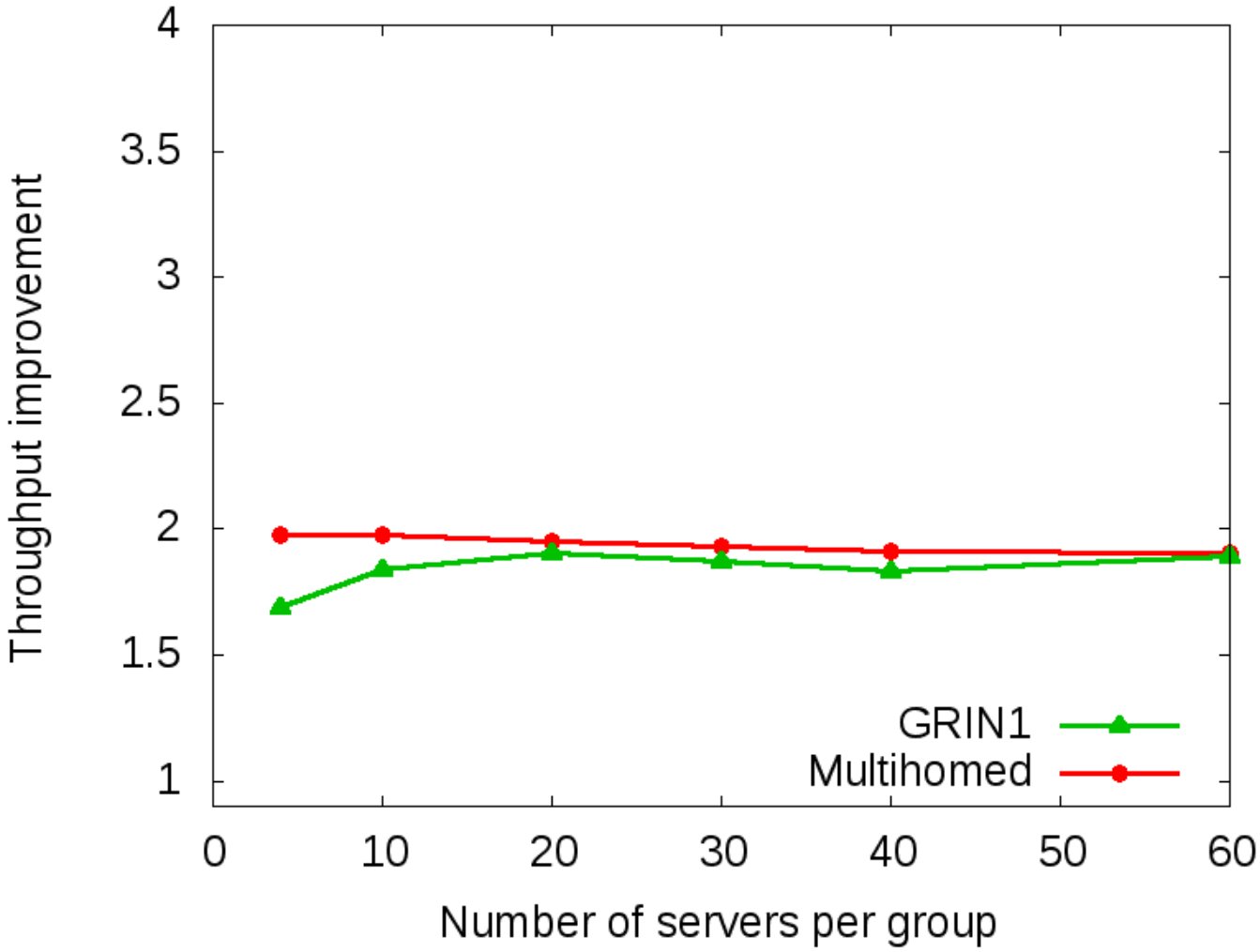
Group Traffic Matrix

- Every server is randomly assigned to a group
- All the groups have the same size
- Within each group we randomly pick a destination
- Each server that is not a destination tries to send flat out to the destination from its group
- Best case scenario for GRIN topologies

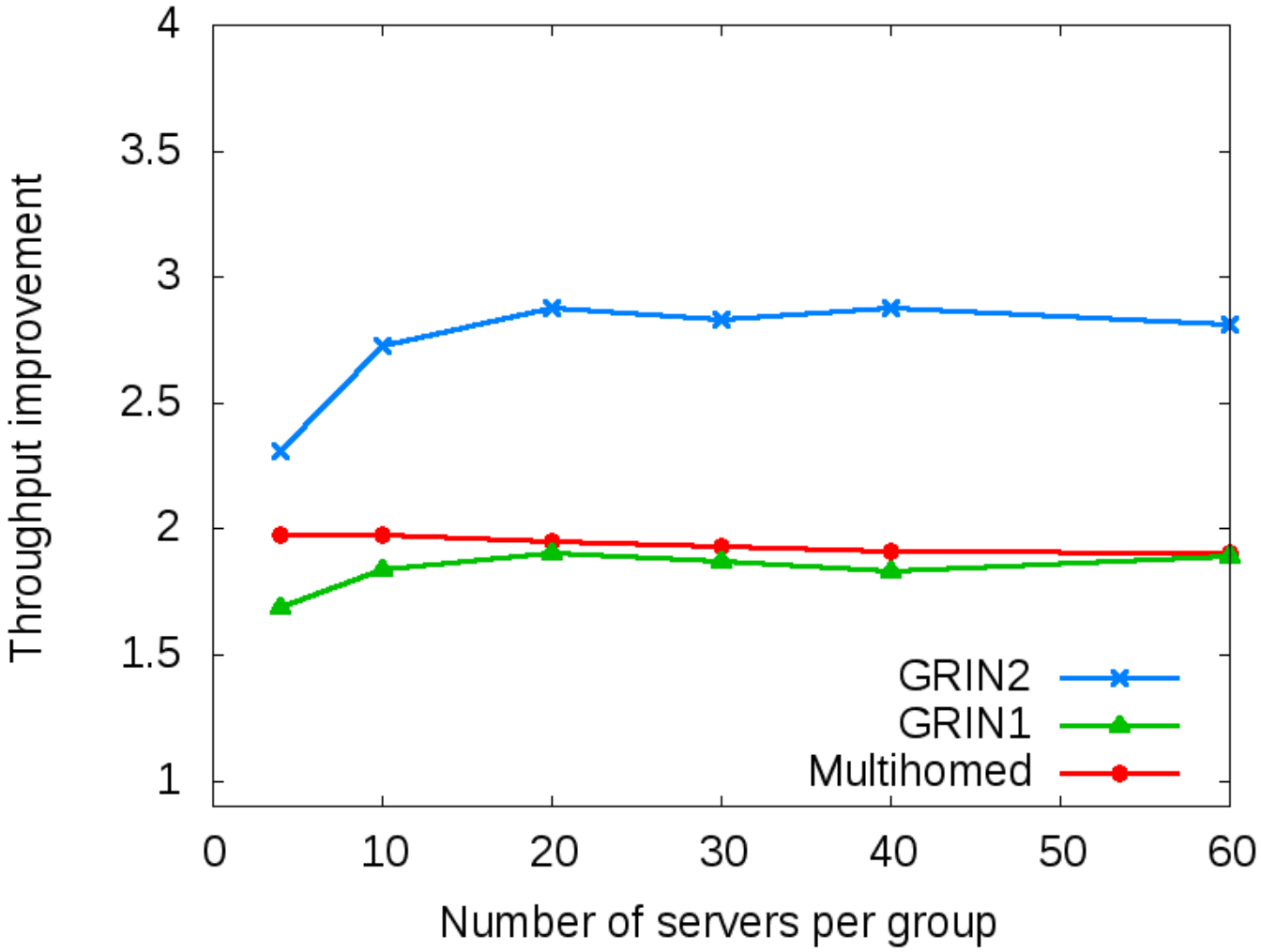
Group Traffic Matrix



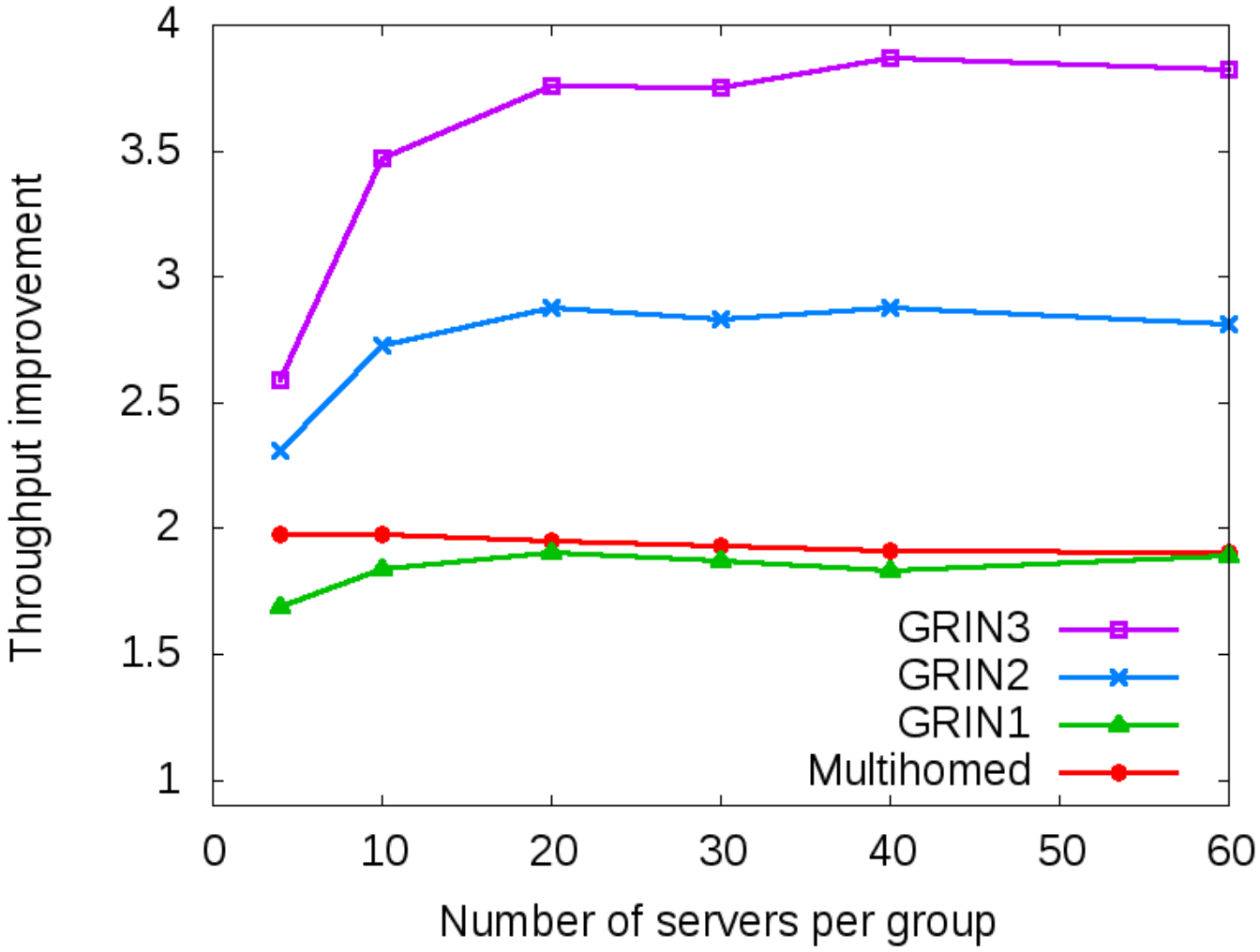
Group Traffic Matrix



Group Traffic Matrix



Group Traffic Matrix



Conclusions

GRIN is a simple, cheap and easily deployable solution to utilize available bandwidth in datacenter networks

Future research:

- build the GRIN topology
- influence on low-latency flows and ways to mitigate any undesirable effects