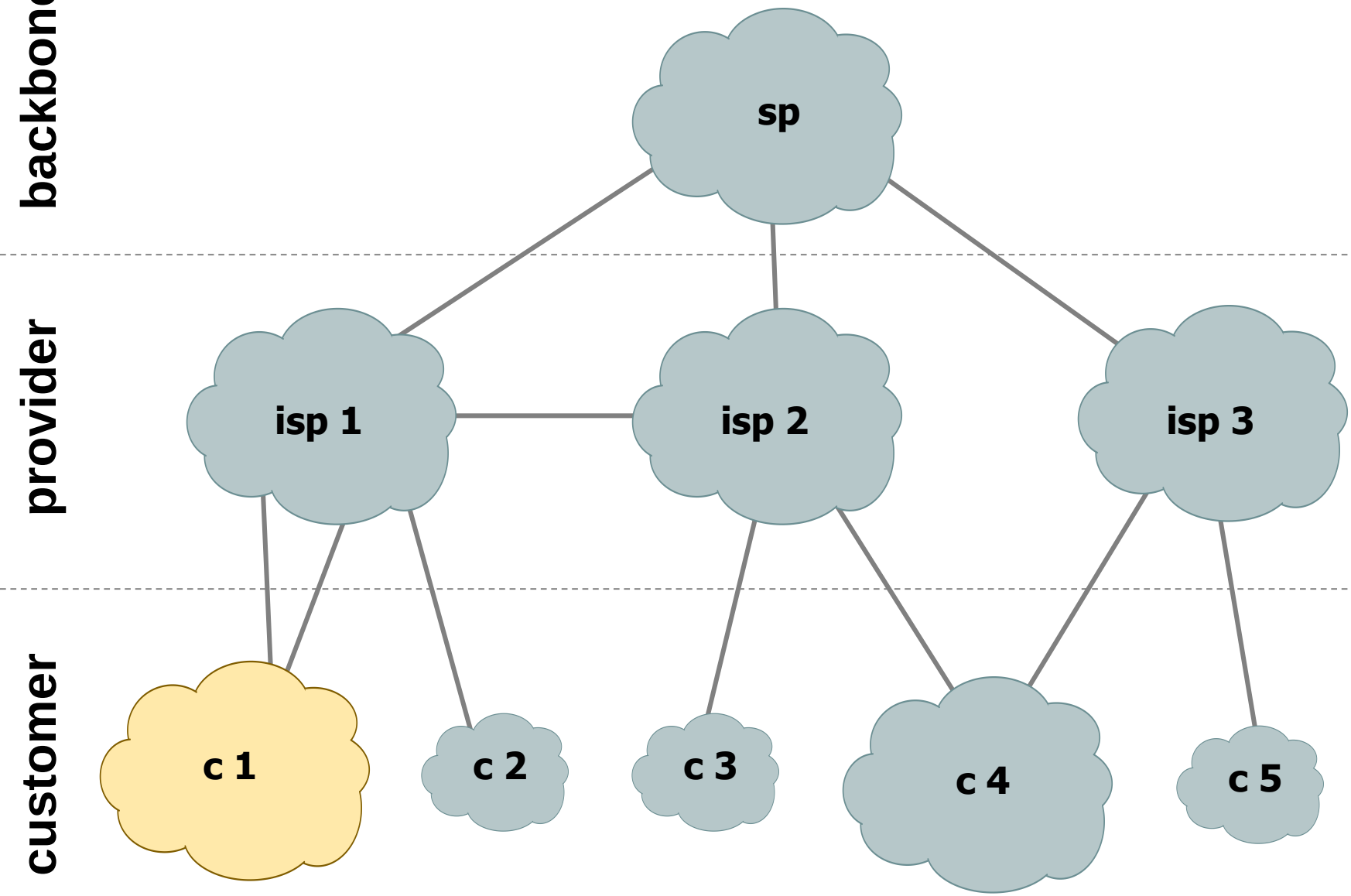
kathara lab

bgp: multi-homed-stub

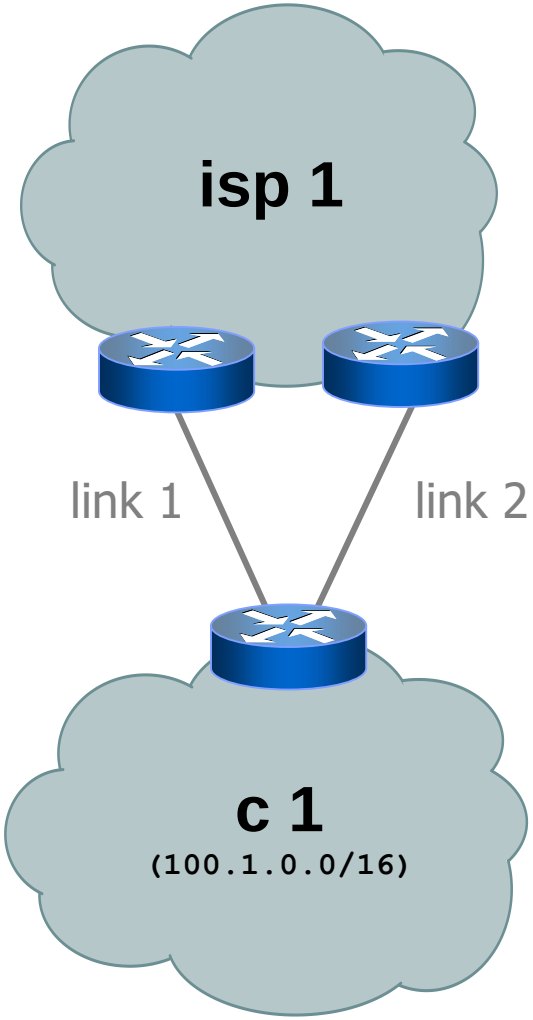
Version	1.0
Author(s)	G. Di Battista, M. Patrignani, M. Pizzonia, F. Ricci, M. Rimondini
E-mail	contact@kathara.org
Web	http://www.kathara.org/
Description	configuration of a multi-homed stub network with backup – kithara version of a netkit lab

copyright notice

multi-homed stub network

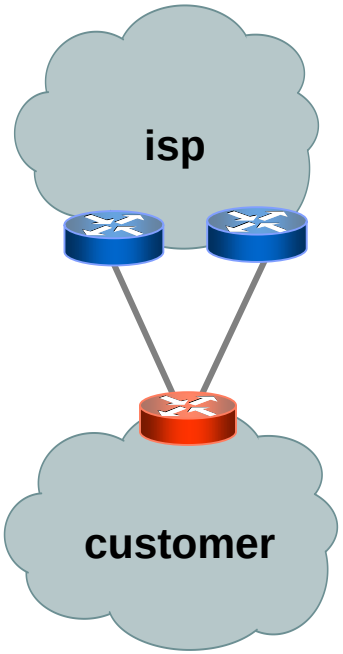


multi-homed stub network

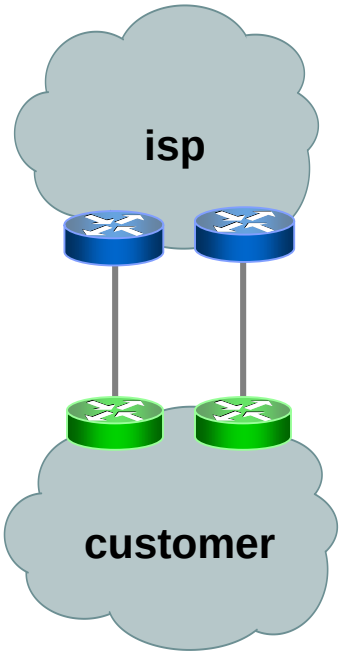


- two links to the same isp
- generally two routers of the customer as are involved

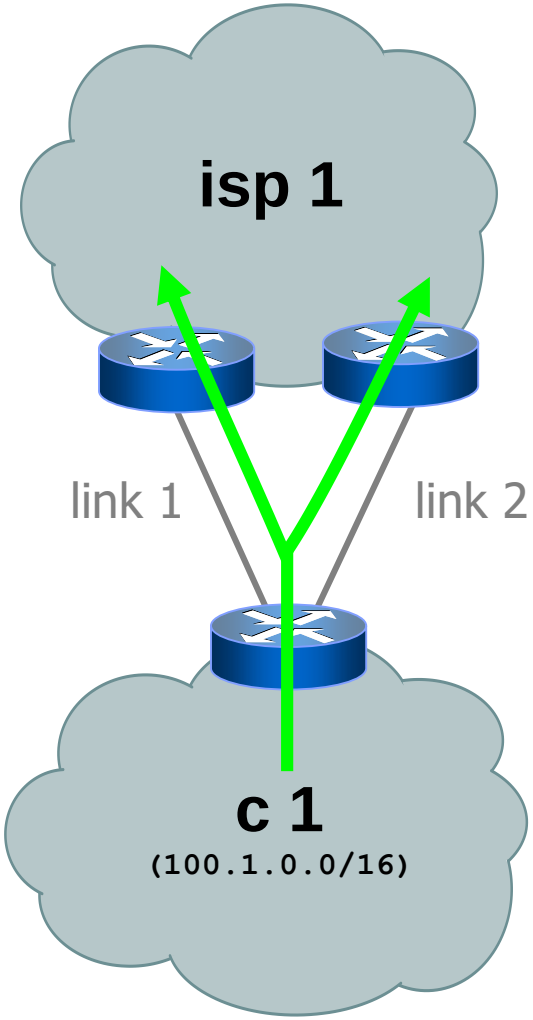
single point of failure



augmented redundancy

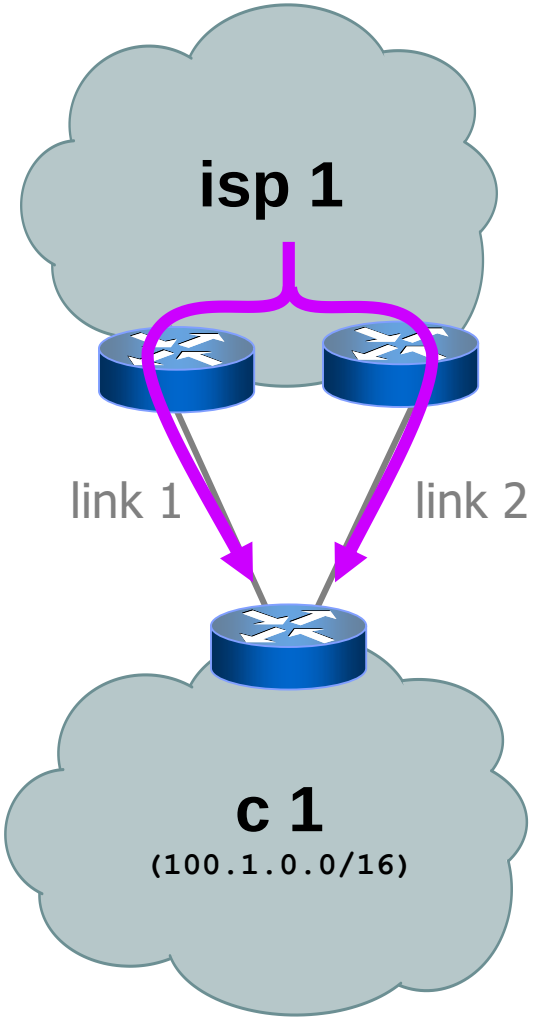


degrees of freedom



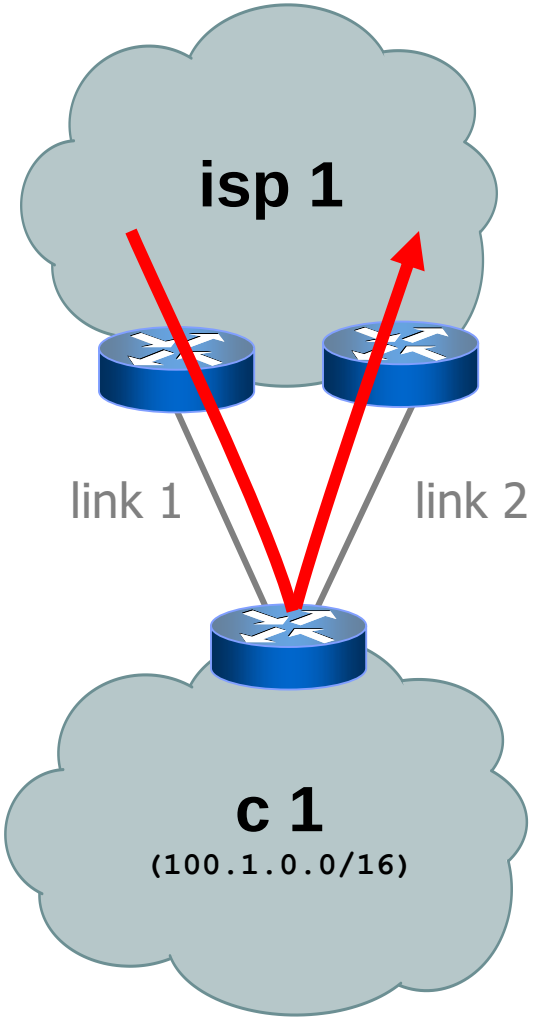
- an outbound packet may be sent through one of the two links in order to reach the internet

degrees of freedom



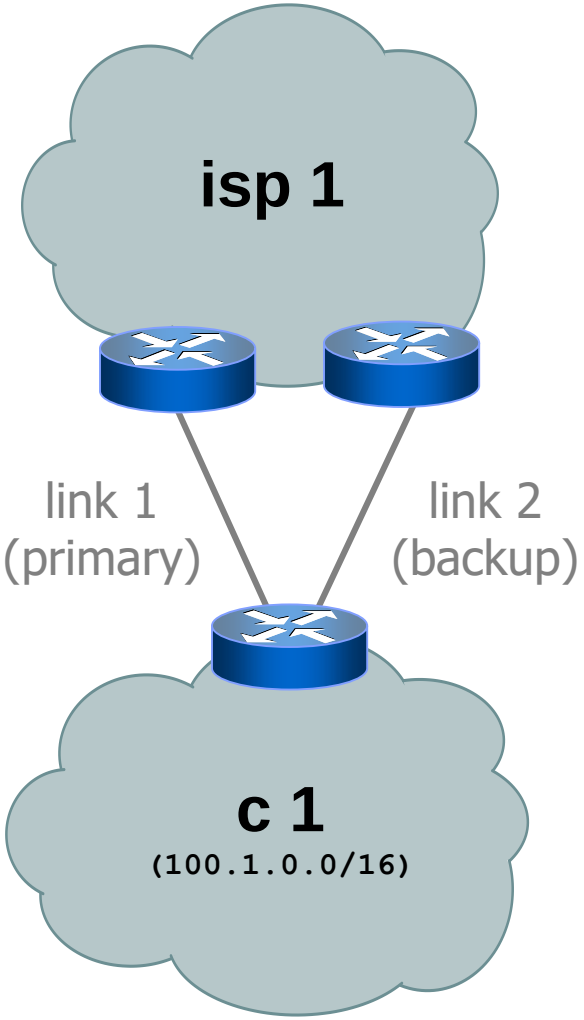
- an outbound packet may be sent through one of the two links in order to reach the internet
- an inbound packet may use any of the two links in order to reach the network

degrees of freedom



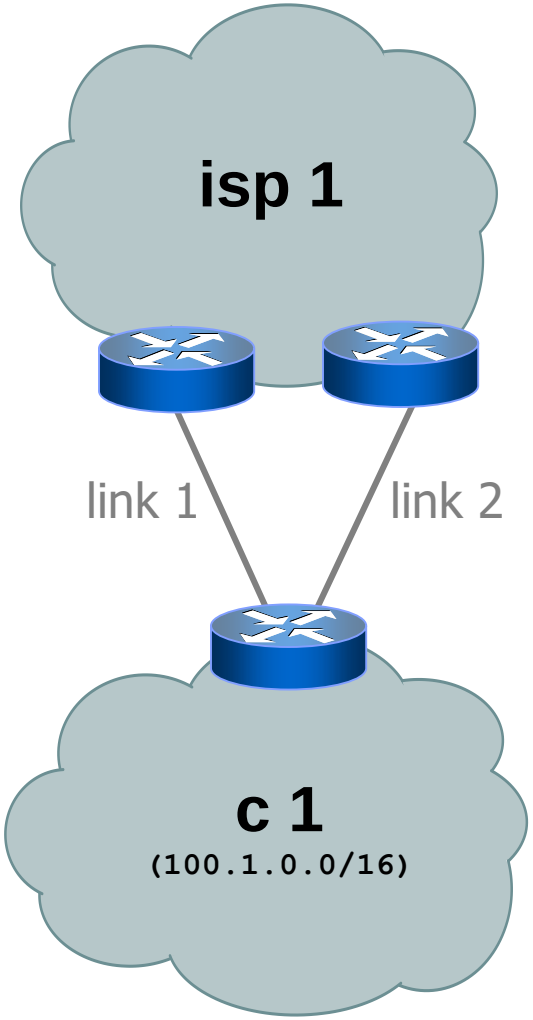
- an outbound packet may be sent through one of the two links in order to reach the internet
- an inbound packet may use any of the two links in order to reach the network
- an internet packet may traverse link 1 and link 2 (or vice versa)

desired policy: backup



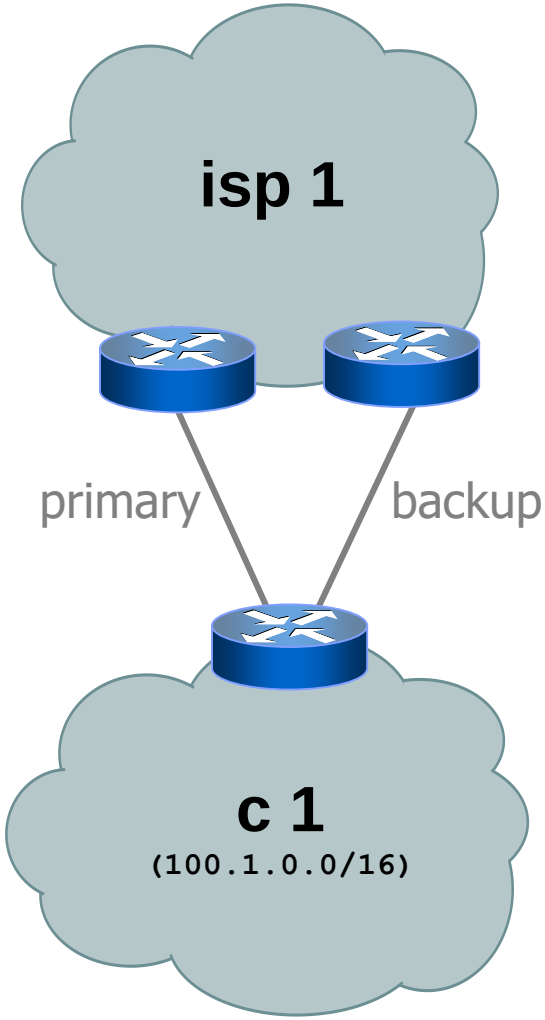
- rule out transit flows
- inbound traffic:
  - use link 1
  - use link 2 when link 1 is unavailable
- outbound traffic:
  - use link 1
  - use link 2 when link 1 is unavailable

alternatives to using bgp



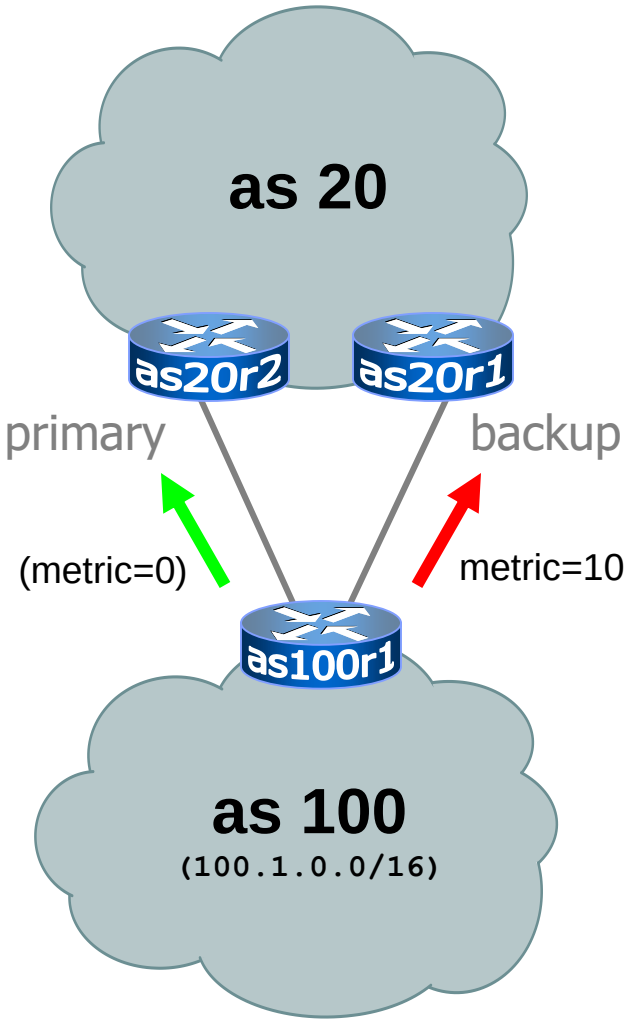
- using an igp (is-is, ospf, rip,...)
  - packets use link 1 or link 2 depending on the shortest path to customer c 1
  - there is no way to rule out transit packets when link 1 and link 2 are on the minimum path between a source and a destination
- using static routes
  - both the routers of the isp and the network have to be coherently configured by hand
  - there is no way to manage an automatic backup mechanism

using bgp



- announce /16 aggregate on each link
  - primary link makes standard announcements
  - backup link increases metric on outbound announcements, and reduces local-pref on inbound announcements
- when one link fails, the announcement of the /16 aggregate via the other link ensures continued connectivity

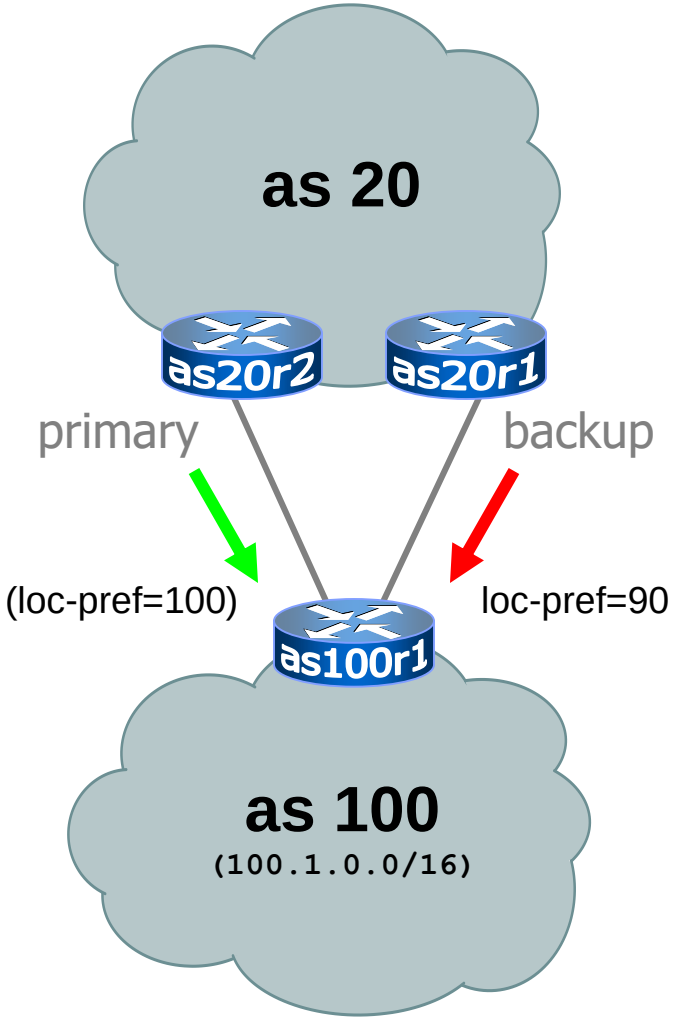
setting metric



- the value of the "multi-exit-discriminator" attribute is called "metric"
- upon receiving the same announcement with two different meds, the provider will (hopefully) adopt the one with the smaller one
- the metric is set on outgoing announcements and manages inbound traffic flows
- metrics are comparable only among announcements coming from the same neighboring as

default value: 0

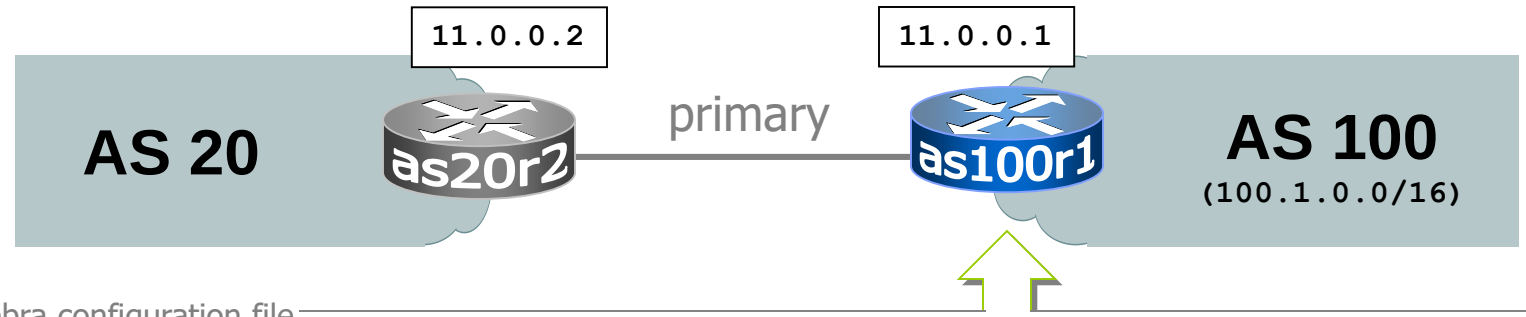
setting local-preference



- the customer assigns a lower local-preference to the announcement coming from the backup peer
- the local-preference attribute is checked before as-path length in the route selection process
- local-preference applies to incoming announcements and manages outbound traffic flows

default value: 100

router as100r1 configuration



zebra configuration file

```
! router as100r1 (primary, customer side)
!
router bgp 100
network 100.1.0.0/16
!
neighbor 11.0.0.2 remote-as 20
neighbor 11.0.0.2 description Router as20r2 (primary)
neighbor 11.0.0.2 prefix-list mineOutOnly out
neighbor 11.0.0.2 prefix-list defaultIn in
!
… next slide
```

router as100r1 configuration

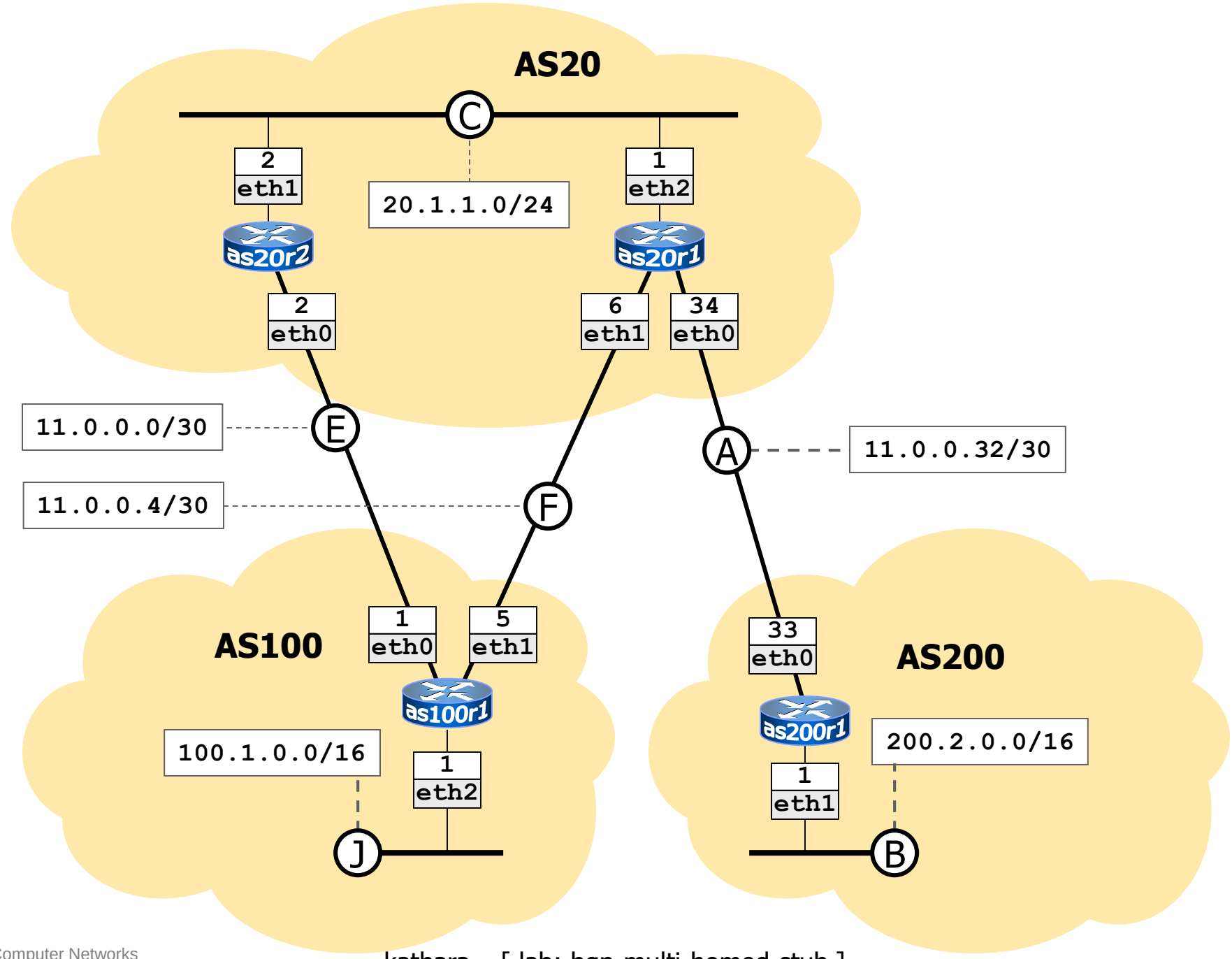
zebra configuration file

```
!
neighbor 11.0.0.6 remote-as 20
neighbor 11.0.0.6 description Router as20r1 (backup)
neighbor 11.0.0.6 prefix-list mineOutOnly out
neighbor 11.0.0.6 route-map metricOut out
neighbor 11.0.0.6 prefix-list defaultIn in
neighbor 11.0.0.6 route-map localPrefIn in
!
… next slide
```

router as100r1 configuration

zebra configuration file

```
ip prefix-list mineOutOnly permit 100.1.0.0/16
!
ip prefix-list defaultIn permit 0.0.0.0/0
!
route-map metricOut permit 10
match ip address myAggregate
set metric 10
!
route-map localPrefIn permit 10
set local-preference 90
!
access-list myAggregate permit 100.1.0.0/16
```



multi-homed stub

- start the lab

host machine

```
user@localhost:~$ cd kathara-lab_bgp-multi-homed-stub
user@localhost:~/kathara-lab_bgp-multi-homed-stub$ lstart
```

- ping as100r1 from as200r1

as200r1

```
as200r1:~# ping 100.1.0.1
PING 100.1.0.1 (100.1.0.1) 56(84) bytes of data.
64 bytes from 100.1.0.1: icmp_seq=1 ttl=62 time=1.39 ms
64 bytes from 100.1.0.1: icmp_seq=2 ttl=62 time=1.88 ms

--- 100.1.0.1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1022ms
rtt min/avg/max/mdev = 1.398/1.642/1.886/0.244 ms
```

- everything seems to work fine, but...

multi-homed stub

- there are strange things happening

as200r1

```
as200r1:~# traceroute 100.1.0.1
traceroute to 100.1.0.1 (100.1.0.1), 64 hops max, 40 byte packets
 1  11.0.0.34 (11.0.0.34)  2 ms  1 ms  1 ms
 2  100.1.0.1 (100.1.0.1)  2 ms  2 ms  2 ms
```

- we set up the routing to prefer passing through as20r2! we are not traversing that router! why?
- even more strange:

as100r1

```
as100r1:~# ping 200.2.0.1
PING 200.2.0.1 (200.2.0.1) 56(84) bytes of data.
From 11.0.0.2 icmp_seq=1 Destination Net Unreachable
From 11.0.0.2 icmp_seq=2 Destination Net Unreachable

--- 200.2.0.1 ping statistics ---
2 packets transmitted, 0 received, +2 errors, 100% packet loss,
time 999ms
```

multi-homed stub

- let us have a look at bgp

as20r1

```
as20r1:~# telnet localhost bgpd
.....
bgpd> show ip bgp
BGP table version is 0, local router ID is 20.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
* i0.0.0.0          20.1.1.2                 0    100      0 i
*>                  0.0.0.0                  0         32768 i
*>i11.0.0.0/30      20.1.1.2                 0    100      0 i
*> 11.0.0.4/30      0.0.0.0                  0         32768 i
*> 11.0.0.32/30     0.0.0.0                  0         32768 i
* i20.1.1.0/24      20.1.1.2                 0    100      0 i
*>                  0.0.0.0                  0         32768 i
* i100.1.0.0/16     11.0.0.1                 0    100      0 100 i
*>                  11.0.0.5                10             0 100 i
*> 200.2.0.0/16     11.0.0.33                0             0 200 i

Total number of prefixes 7
```

why is bgp choosing to pass through 11.0.0.5 rather than 11.0.0.1?

multi-homed stub

- the point of view of as20r2

as20r2

```
as20r2:~# telnet localhost bgpd
.....
bgpd> show ip bgp
BGP table version is 0, local router ID is 20.1.1.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
* i0.0.0.0          20.1.1.1                 0    100      0 i
*>                  0.0.0.0                  0         32768 i
*> 11.0.0.0/30      0.0.0.0                  0         32768 i
*>i11.0.0.4/30      20.1.1.1                 0    100      0 i
*>i11.0.0.32/30     20.1.1.1                 0    100      0 i
* i20.1.1.0/24      20.1.1.1                 0    100      0 i
*>                  0.0.0.0                  0         32768 i
*> 100.1.0.0/16     11.0.0.1                 0              0 100 i
* i                 11.0.0.5                10    100      0 100 i
* i200.2.0.0/16     11.0.0.33                0    100      0 200 i

Total number of prefixes 7
```

200.2.0.0 is in the table (just 1 entry) but is not selected as the best

multi-homed stub

- the configuration is wrong; ibgp and igp do not interplay properly in as20
  - no igp tells as20r1 how to reach next-hop 11.0.0.1
  - no igp tells as20r2 how to reach next-hop 11.0.0.33
  - since the next-hops learned via ebgp are not reachable (i.e., the recursive lookup fails), bgp does not use them
- notice that the ping from as200r1 to 100.1.0.1 works
  - forward path: 11.0.0.34, 11.0.0.5
  - backward path: 11.0.0.2, 20.1.1.1, 11.0.0.33 (have a look with a sniffer placed inside as20r1)

multi-homed stub

- how to fix?
- several possible solutions
  - activate rip in as20
  - add static routes in as20r1 and as20r2
  - …
- the rip solution; on both as20r1 and as20r2 do:
  - configure rip (edit **`/etc/zebra/ripd.conf`**)
    - **`router rip`**
    - **`network 20.1.1.0/24`**
    - **`redistribute connected`**
  - activate rip (edit **`/etc/zebra/daemons`**)
  - restart zebra (**`/etc/init.d/zebra restart`**)

multi-homed stub

- how to check that it works?
  - perform a **`show ip bgp`** on all routers
  - check with **`route`** on all routers
  - perform **`ping`**s and **`traceroute`**s from/to several sources/destinations
- example:

as100r1

```
as100r1:~# traceroute 200.2.0.1
traceroute to 200.2.0.1 (200.2.0.1), 64 hops max, 40 byte packets
 1  11.0.0.2 (11.0.0.2)  1 ms  2 ms  1 ms
 2  20.1.1.1 (20.1.1.1)  2 ms  2 ms  2 ms
 3  200.2.0.1 (200.2.0.1)  2 ms  2 ms  2 ms
```

as100r1 is reaching 200.2.0.1 via as20r2 (as it should)

multi-homed stub

- now shut down the primary connection on as100r1

as100r1

```
as100r1:~# telnet localhost bgpd
.....
User Access Verification

Password: zebra
bgpd> enable
Password: zebra
bgpd# configure terminal
bgpd(config)# router bgp 100
bgpd(config-router)# neighbor 11.0.0.2 shutdown
bgpd(config-router)# quit
bgpd(config)# quit
bgpd# write file
Configuration saved to /etc/zebra/bgpd.conf
bgpd# show ip bgp summary
BGP router identifier 100.1.0.1, local AS number 100
2 BGP AS-PATH entries
0 BGP community entries

Neighbor        V    AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
11.0.0.2        4    20      12      21        0    0    0 00:00:22 Idle (Admin)
11.0.0.6        4    20      12      20        0    0    0 00:07:33            1

Total number of neighbors 2
bgpd#
```

multi-homed stub

- check the backup

as100r1

```
as100r1:~# route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
11.0.0.4        0.0.0.0         255.255.255.252 U     0      0        0 eth1
11.0.0.0        0.0.0.0         255.255.255.252 U     0      0        0 eth0
100.1.0.0       0.0.0.0         255.255.0.0     U     0      0        0 eth2
default         11.0.0.6        0.0.0.0         UG    0      0        0 eth1
```

as100r1

```
as100r1:~# traceroute 200.2.0.1
traceroute to 200.2.0.1 (200.2.0.1), 64 hops max, 40 byte packets
 1  11.0.0.6 (11.0.0.6)  2 ms  2 ms  2 ms
 2  200.2.0.1 (200.2.0.1)  2 ms  2 ms  1 ms
```

multi-homed stub

- restart the primary connection and check that the primary link is back

as100r1

```
as100r1:~# telnet localhost bgpd
.....
User Access Verification

Password: zebra
bgpd> enable
Password: zebra
bgpd# configure terminal
bgpd(config)# router bgp 100
bgpd(config-router)# no neighbor 11.0.0.2 shutdown
bgpd(config-router)# quit
bgpd(config)# quit
bgpd# write file
Configuration saved to /etc/zebra/bgpd.conf
bgpd# quit
Connection closed by foreign host.
as100r1:~# traceroute 200.2.0.1
traceroute to 200.2.0.1 (200.2.0.1), 64 hops max, 40 byte packets
 1  11.0.0.2 (11.0.0.2)  1 ms  1 ms  1 ms
 2  20.1.1.1 (20.1.1.1)  1 ms  2 ms  2 ms
 3  200.2.0.1 (200.2.0.1)  2 ms  2 ms  2 ms
as100r1:~#
```