IPv6 Deployment in Sofia University Network (2007-2013)

Vesselin Kolev (Technion) and Vasil Kolev

Who was involved?

Who was involved?

Former crew of "Networks and Communications" who designed, implemented and supported the IPv6 connectivity at Sofia University between 2007 and 2013:

Vesselin Kolev (VESS-RIPE) – *now at* Technion (Israel Institute of Technology)

Nikolai Nikolov, Vladislav Rusanov – *now at* Tradeo

Hristo Dragolov – *now at* Manson

Radoslav Buchakchiev – *now at* Aalborg University

Ivan Yordanov – *now at* Interoute

Georgi Naidenov – *now is* a freelancer

Stefan Dimitrov, Vladislav Georgiev, Mariana Petkova – *still at* Sofia University

Global Unicast Address Allocation

Global Unicast Address Allocation

Currently Used Global Unicast Address Allocation
(since February 11, 2011)

```
inet6num:       2001:67c:20d0::/47
netname:        BG-SUNET
descr:          Sofia University "St. Kliment Ohridski"
descr:          Autonomous IPv6 Address Space
country:        BG
org:            ORG-UoS32-RIPE
admin-c:        NCC123-RIPE
tech-c:         NCC123-RIPE
status:         ASSIGNED PI
mnt-by:         RIPE-NCC-END-MNT
mnt-lower:      RIPE-NCC-END-MNT
mnt-by:         AS5421-MNT
mnt-routes:     AS5421-MNT
mnt-domains:    AS5421-MNT
source:         RIPE
```

In order to secure better the Sofia University's maintainer object AS5421-MNT in RIPE DB *only* OpenPGP digitally signed authentication is allowed (since 2012).

Global Unicast Address Allocation

NO MORE IN USE IN SOFIA UNIVERSITY NETWORK

Previously Used Global Unicast Address Allocation

(between February 12, 2007 and February 11, 2011)

```
inet6num:        2a01:288:8000::/35
netname:         BG-SUNET
descr:           Sofia University
descr:           BG-1164 Sofia
country:         BG
admin-c:         KS2437-RIPE
tech-c:          SD2427-RIPE
tech-c:          GN1498-RIPE
tech-c:          VK1242-RIPE
notify:          as5421@uni-sofia.bg
status:          ASSIGNED
mnt-by:          AS5421-MNT
mnt-domains:     AS5421-MNT
source:          RIPE
```

This address allocation was given to Sofia University on a temporary basis. It was used for the deployment until February 11, 2011 in ASSIGNED PI manner. On that date this allocation was returned to the Mobiltel IPv6 address pool.

Address Allocation Policy

Global Unicast Address Allocation Policy

Source segment: `2001:67c:20d0::/47`

Documented in: RFC4291

Purpose: Access to IPv6 Internet

1. Initial allocation:

- `/60` for each faculty network
- `/64` for each backbone network
- `/64` for each server farm Ethernet segment
- `/64` for each virtual machine internal bridge in server farms (if requested).

2. Additional allocations: on-request, following “Initial allocation” sizes.

3. Special allocations (sub `/64`): on-request

Unique Local Address Allocation Policy

Source segment: `fc00::/7`

Documented in: **RFC4193**

Purpose: **Local access to restricted resources in Sofia University network**

1. **Initial allocation (per-request):** `/32`
2. **Additional allocations (on-request):** `/32`

Inter- and Intra-AS Routing

BGP4+: Equipment

Application Software:

`Quagga 0.99.15 (http://www.quagga.net/)`

Running on:

`Linux  2.6.32/CentOS 6.x (http://www.centos.org)`

Inter- and Intra-AS Routing

BGP4+: Connectivity

1. Global Unicast Connectivity:

```
AS6802  (UNICOM-B-AS)
AS8717  (SPECTRUMNET)
```

2. Local Unicast Connectivity (Peering):

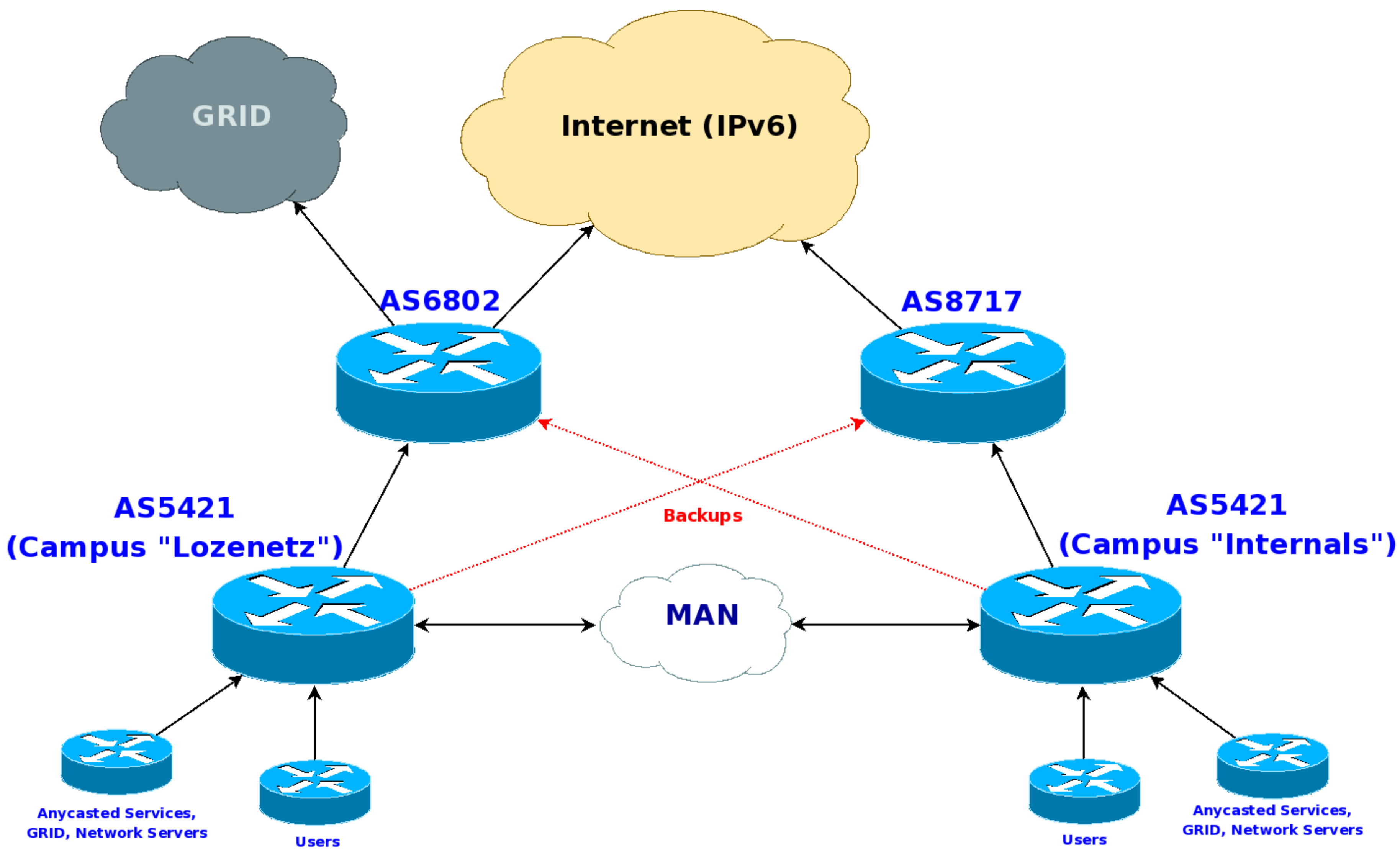
```
AS3245  (DIGSYS-AS)
AS8262  (LIREXNET-AS)
AS9070  (ITD)
AS9127  (NETISSAT-AS)
AS34224 (NETERRA-AS)
```

3. Local Stub-AS Connectivity (Anycast):

```
AS112   (ISC-AS112)
```

Inter- and Intra-AS Routing

BGP4+: Global Unicast Connectivity



Inter- and Intra-AS Routing

BGP4+: Originated Prefixes

1. Unicast prefixes:

```
2001:67c:20d0::/48 (as-path ... 5421)
2001:67c:20d1::/48 (as-path ... 5421)
2001:67c:20d0::/47 (as-path ... 5421)
```

2. Anycast prefixes (origin AS112):

```
2620:4f:8000::/48  (as-path ... 5421 112)
```

Inter- and Intra-AS Routing

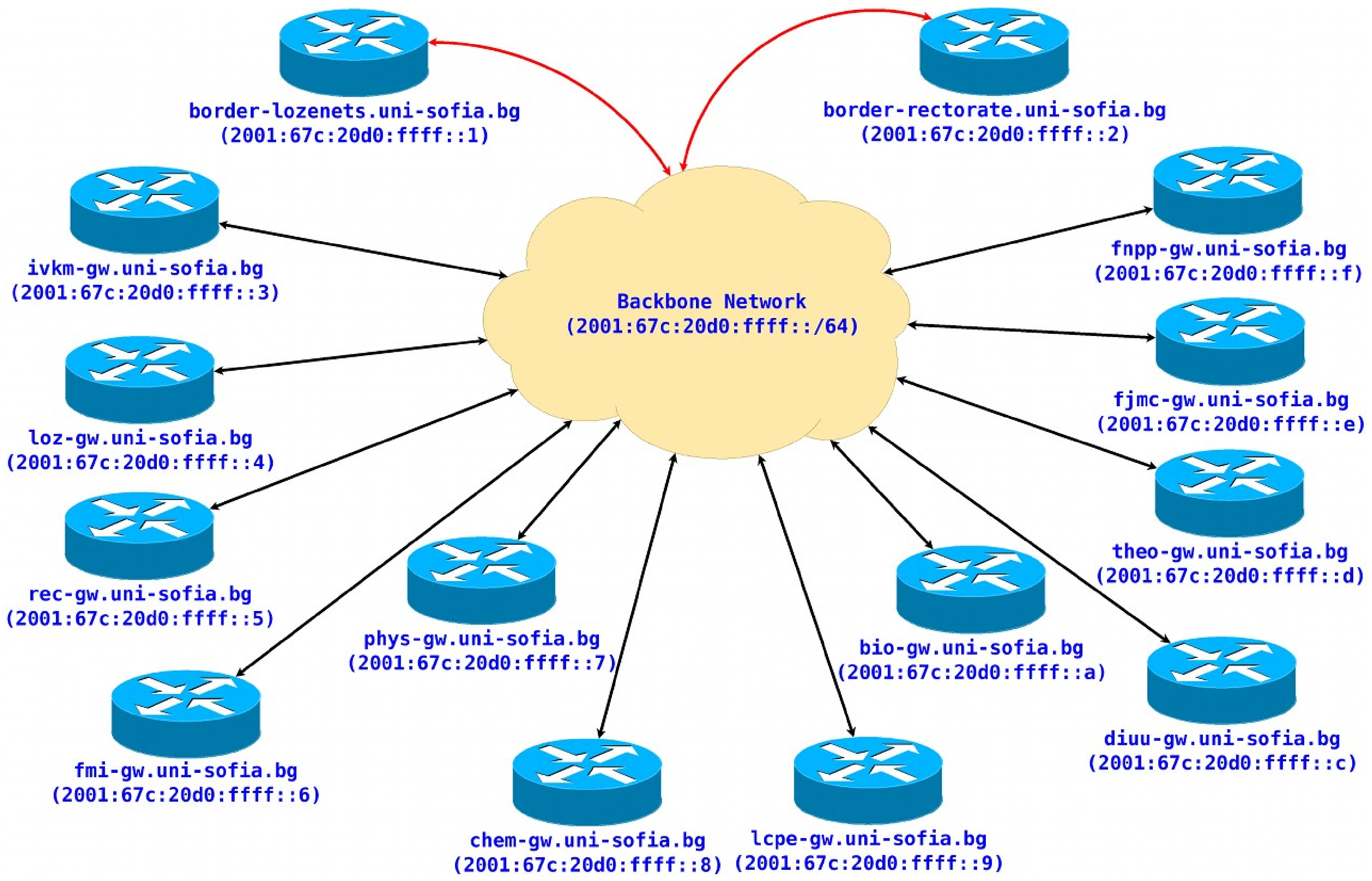
AS5421 BGP4+ incoming unicast prefix filters

```
if prefix(origin_as) not in range(64496-64511)
                 and not in range(64512-65535)
                 and not in range(65536-65551)
                 and not in range(65552-131071)
                 and not in range(4200000000-4294967295)
     continue
else
     reject


if prefix(net) in 2000::/3
           and (prefix(len) gt /3 and lt /49)
     continue
else
     reject
```

Inter- and Intra-AS Routing

BGP4+: Intra-AS routing - Route Reflector Schema



Inter- and Intra-AS Routing

BGP4+: Intra-AS routing – Route Reflector Schema

1. Roles in Route Reflector (RR) Schema:

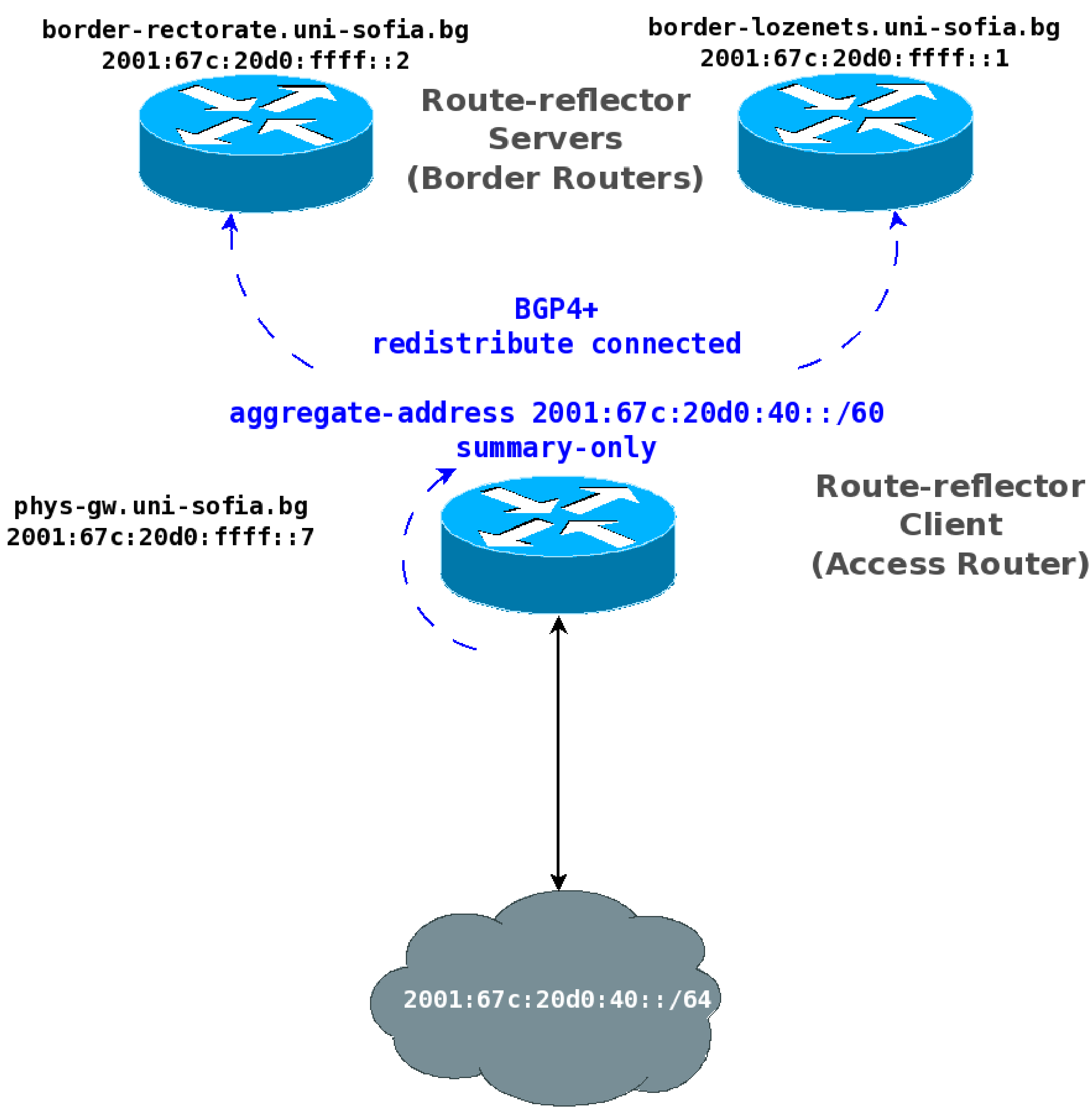
`Route Reflector Server == Border Router`

`Route Reflector Client == Access Router`

2. RR reduces the total number of BGP4+ sessions:

- ✔ **each access router supports only 2 BGP4+ sessions (one session per border router);**
- ✔ **each border router supports one BGP4+ session per access router.**

Intra-AS Routing

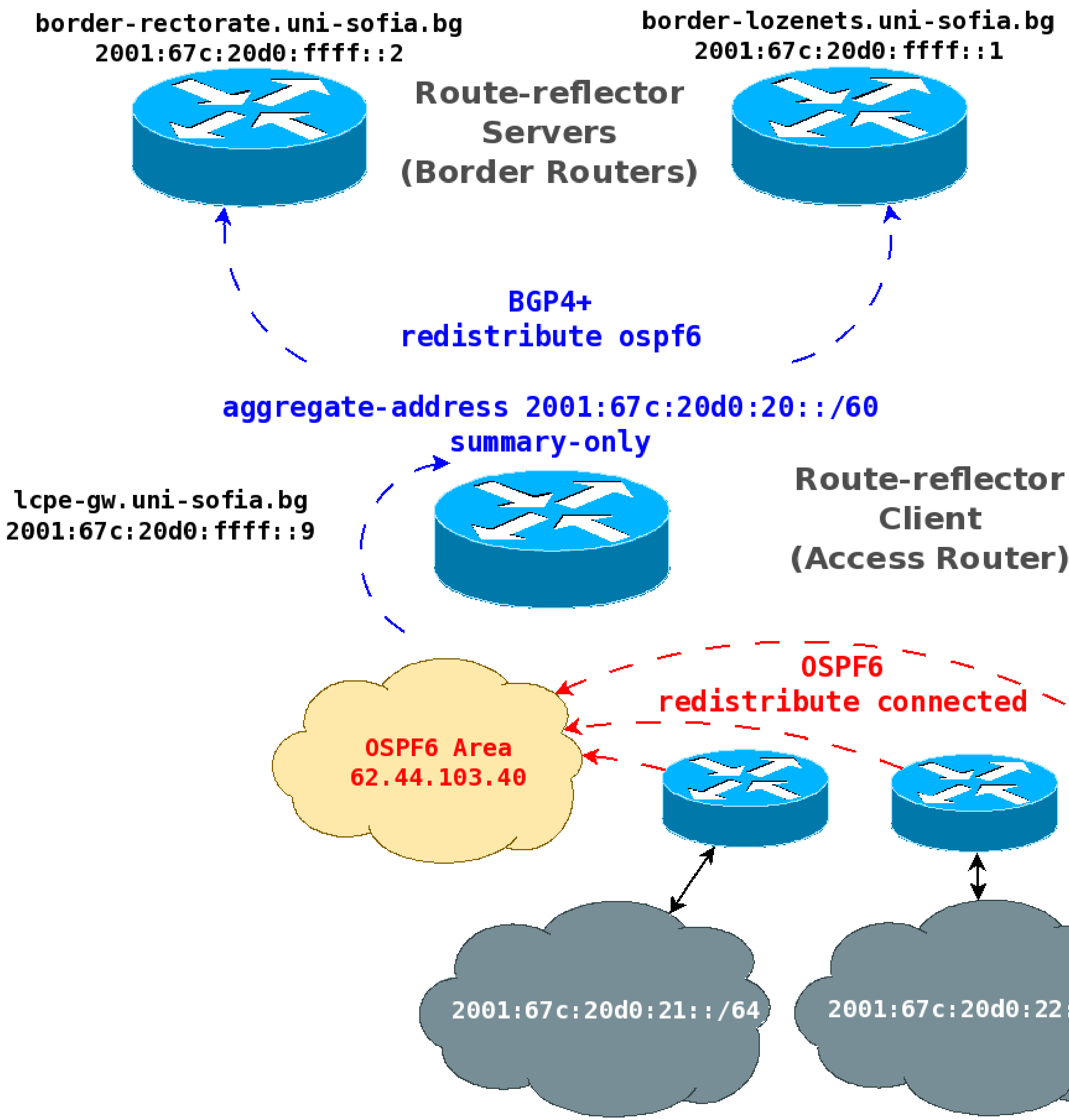


BGP4+: Intra-AS routing – Prefix Origination

Prefix origination within AS5421 is organized to be *"link-state"* - like.

So *no* BGP4+ origination through "network" declaration is used. Only BGP4+ redistribution and aggregation is applied.

Intra-AS Routing



BGP4+: Intra-AS routing – Prefix Origination

Prefix origination within AS5421 is organized to be *“link-state”* - like.

So *no* BGP4+ origination through “network” declaration is used. Only BGP4+ redistribution and aggregation is applied.

Intra-AS Routing (behind the access routers)

Securing OSPFv3 and RIPng as multicast services with IPsec (1/2)

1. Configuration in setkey.conf if ipsec-tools are used (kernel 2.6.x, KAME IPsec implementation):

```
spdadd fe80::/64[0] ff02::5[0] any -P in ipsec esp/transport//require ;
spdadd fe80::/64[0] ff02::5[0] any -P out ipsec
esp/transport//require ;
spdadd fe80::/64[0] ff02::6[0] any -P in ipsec esp/transport//require ;
spdadd fe80::/64[0] ff02::6[0] any -P out ipsec
esp/transport//require ;

################ OSPF multicast group ff02::5 #####################

add fe80::230:18ff:feba:106f ff02::5 esp 0x962005 -m transport -E
blowfish-cbc 0x27c... -A hmac-sha256 0x3f7... ;

################ OSPF multicast group ff02::6 #####################

add fe80::230:18ff:feba:106f ff02::6 esp 0x962006 -m transport -E
blowfish-cbc 0x27c... -A hmac-sha256 0x3f7... ;
```

Intra-AS Routing (behind the access routers)

Securing OSPFv3 and RIPng as multicast services with IPsec (2/2)

2. Why use IPsec for securing OSPFv3 and RIPng:

- ✔ OSPFv3 and RIPng applications receives packages only from secured and trusted sources filtered by kernel (not by internal software routine in ospf6d or ripngd);
- ✔ It is impossible for intruders to send malformed packages to the multicast groups and ospf6d and ripngd if they are not part of IPsec transport communication;

3. Some problems with the implementation:

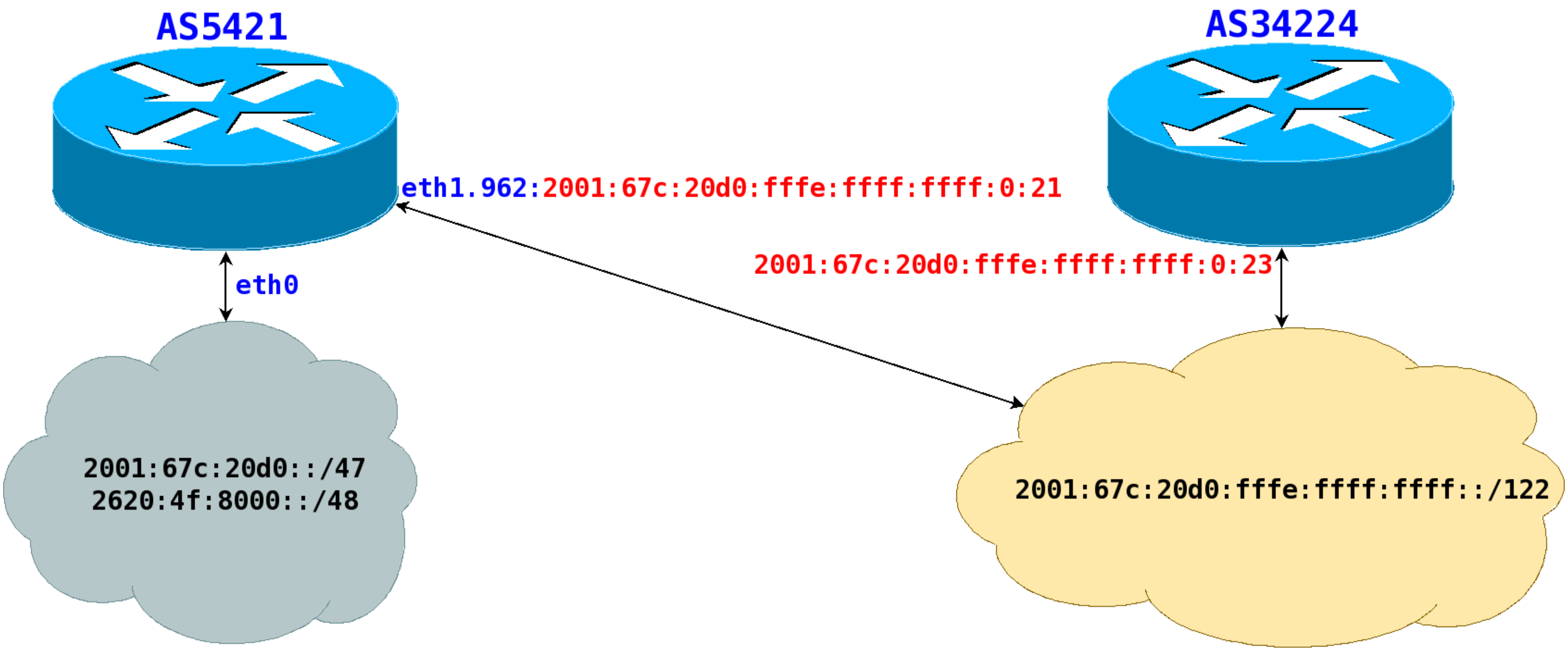
- ✔ Preshared keys are used;
- ✔ Not a standard;
- ✔ Requires very qualified staff to implement and support it.

Note: *RFC5374 defines the mechanism for key-exchange multicast based IPsec communication but it is still unimplemented in major open source IKEv1 software.*

Packet Filtering

Packet Filtering

Bi-directional packet filtering applied on border routers



Packet Filtering

Bi-directional packet filtering applied on border routers

Policy related to the incoming traffic on **eth0:**

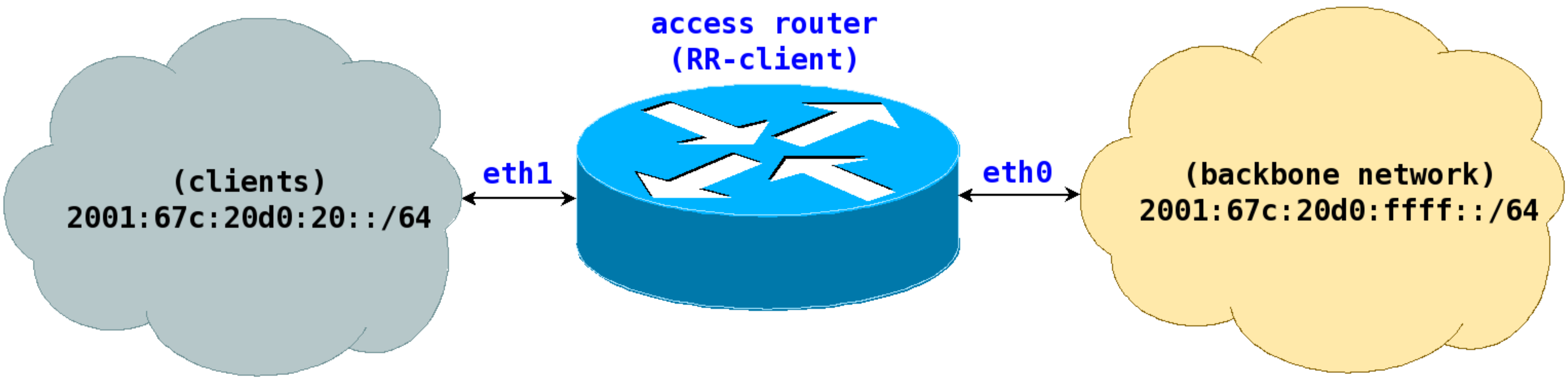
```
# ip -6 route add default dev lo table blackhole
# ip -6 rule add from 2001:67c:20d0::/47 dev eth0 table main prio 10
# ip -6 rule add from 2620:4f:8000::/48 dev eth0 table main prio 10
# ip -6 rule add from ::/0 dev eth0 table blackhole prio 100
# ip -6 rule add to 2000::/3 dev eth0 table main prio 10
# ip -6 rule add to ::/0 dev eth0 table blackhole prio 100
```

Policy related to the incoming traffic on **eth1.962:**

```
# ip -6 rule add from 2001:67c:20d0:fffe:ffff:ffff::/122 dev eth1.962
  table main prio 10
# ip -6 rule add from 2001:67c:20d0::/47 dev eth1.962 table blackhole
  prio 20
# ip -6 rule add from 2000::/3 dev eth1.962 table main prio 30
# ip -6 rule add from ::/0 dev eth1.962 table blackhole prio 100
# ip -6 rule add to 2001:67c:20d0::/47 dev eth1.962 table main prio 10
# ip -6 rule add to 2620:4f:8000::/48 dev eth1.962 table main prio 10
# ip -6 rule add to ::/0 dev eth1.962 table blackhole prio 100
```

Packet Filtering

Bi-directional packet filtering applied on access routers



Policy related to the incoming traffic on **eth0:**

```
# ip -6 route add default dev lo table blackhole
# ip -6 rule add to 2001:67c:20d0:20:/60 dev eth0 table main prio 10
# ip -6 rule add to ::/0 dev eth0 table blackhole prio 100
```

Policy related to the incoming traffic on **eth1:**

```
# ip -6 rule add from 2001:67c:20d0:20:/60 dev eth1 table main prio 10
# ip -6 rule add from ::/0 dev eth1 table blackhole prio 100
```

Packet Filtering

Protocol-specific packet filtering on border-routers

Port 179/tcp must be accessible only to the designated BGP4+ neighbors (border or access-routers):

```
# ip6tables -A INPUT -p tcp -m tcp --dport 179 -j REJECT --reject-with
 tcp-reset
# ip6tables -I INPUT -s 2001:67c:20d0:ffff::2 -p tcp -m tcp --dport 179
 -j ACCEPT
# ip6tables -I INPUT -s 2001:67c:20d0:ffff::3 -p tcp -m tcp --dport 179
 -j ACCEPT
...
```

Local ntpd (123/udp) must not be open for new requests and have to support only the responses from requested NTP servers:

```
# ip6tables -A INPUT -p udp -m conntrack ! --ctstate
RELATED,ESTABLISHED -m udp --dport 123 -j DROP
# ip6tables -I INPUT -i lo –dport 123 -j ACCEPT
...
```

Packet Filtering

Protocol-specific packet filtering on access-routers (1/2)

Port 179/tcp must be accessible only to the designated BGP4+ neighbors (border-routers):

```
# ip6tables -A INPUT -p tcp -m tcp --dport 179 -j REJECT --reject-with
 tcp-reset
# ip6tables -I INPUT -s 2001:67c:20d0:ffff::2 -p tcp -m tcp --dport 179
 -j ACCEPT
# ip6tables -I INPUT -s 2001:67c:20d0:ffff::3 -p tcp -m tcp --dport 179
 -j ACCEPT
...
```

Local ntpd (123/udp) must not be open for new requests and have to support only the responses from requested NTP servers:

```
# ip6tables -A INPUT -p udp -m conntrack ! --ctstate
RELATED,ESTABLISHED -m udp --dport 123 -j DROP
# ip6tables -I INPUT -i lo –dport 123 -j ACCEPT
...
```

Packet Filtering

Protocol-specific packet filtering on access-routers (2/2)

Preventing the direct access to the workstations:

```
# ip6tables -I FORWARD -o eth1 -p tcp -m conntrack ! --ctstate
RELATED,ESTABLISHED -j REJECT --reject-with tcp-reset
# ip6tables -I FORWARD -o eth1 -p udp -m conntrack ! --ctstate
RELATED,ESTABLISHED -j DROP
```

This way no service on the workstations could be accessed *directly*. Only workstations can initiate TCP sessions or UDP streams.

IPv6 Enabled Services

IPv6 Enabled Services

1. Web

Most of the web-servers in Sofia University Network serve content over IPv6 protocol since 2007:

http://www.uni-sofia.bg
http://mailbox.uni-sofia.bg
http://www.lcpe.uni-sofia.bg
http://www.phys.uni-sofia.bg
...

2. DNS

All DNS servers of Sofia University support IPv6 since 2007. BIND9 is used. Hierarchical anycast is implemented in Intra-AS routing to keep DNS service fault tolerant.

3. SMTP/IMAP

Major SMTP/IMAP servers in Sofia University Network (mailbox.uni-sofia.bg, for exampe) support IPv6 *since 2007*. Sendmail, Cyrus IMAPd, and Dovecot are used. The first SPAM message received from IPv6 based source was received by mailbox.uni-sofia.bg on December 12, 2007.

IPv6 Enabled Services

4. FTP

ftp://ftp.uni-sofia.bg supports IPv6 since 2010.

5. OpenVPN

Two implementations of OpenVPN servers in Sofia University Network supports IPv6. Pools of IPv6 addresses are used to address automatically the clients (if the client version of OpenVPN supports IPv6 addressing/pushing).

Addressing and Servicing the Clients

1. Coverage

- ✔ In the end of 2008 more than 50% of the workstations in Campus “Lozenets” was connected to IPv6 Internet.
- ✔ Today over 85% of workstations in Sofia University Network have global unicast IPv6 addresses!
- ✔ 17-23% of the traffic of the traffic is IPv6

2. Methods for addressing and configuring IPv6 hosts

Each access router in Sofia University Network provides:

- ✔ RADVD – Stateless method for numbering the workstations by advertising a network prefix and router address information. Zeroconf is used in addition to provide DNS cache servers addresses.
- ✔ DHCP6 – Statefull method for numbering the workstations and supplying the hosts with information about the accessible DNS cache servers.

Default routes as prefixes and their origination and redistribution

Default routes as prefixes and their origination and redistribution

IPv6 default routes

`::/0`

Default route to the whole IPv6 address space.

✔ Must *not* appear in Inter-AS prefix redistribution.
✔ May appear in Intra-AS prefix redistribution.
✔ Mandatory for an access to all IPv6 Global and Unique Local Unicast address spaces.
✔ The *next-hop* is the default gateway.
✔ Manually installed (static route configuration) or automatically installed in RIB if the client accepts and processes Router Advertisement (RA) messages.

`2000::/3`

Default route to the whole IPv6 Global Unicast address space (IPv6 Internet).

✔ Must *not* appear in Inter-AS prefix redistribution.
✔ May appear in Intra-AS prefix redistribution.
✔ Might be set to allow specific access only to IPv6 Global address spaces.
✔ The *next-hop* is not the default gateway.
✔ Manually installed (static route configuration) or automatically installed in RIB if the client speaks RIPng, OSPFv3, or BGP4+. It is possible that a *fence* device could manage the installation and removal of that route (clustering, maintenance).

Default routes as prefixes and their origination and redistribution

Self-origination: creation and redistribution of the default route `::/0` into OSPFv3 (similar is for RIPng)

It does *not* block the IPv6 transport because bgpd supplies RIB with *more specific* prefixes (::/0 is the least specific)

Create route `::/0` in `/etc/quagga/zebra.conf`:

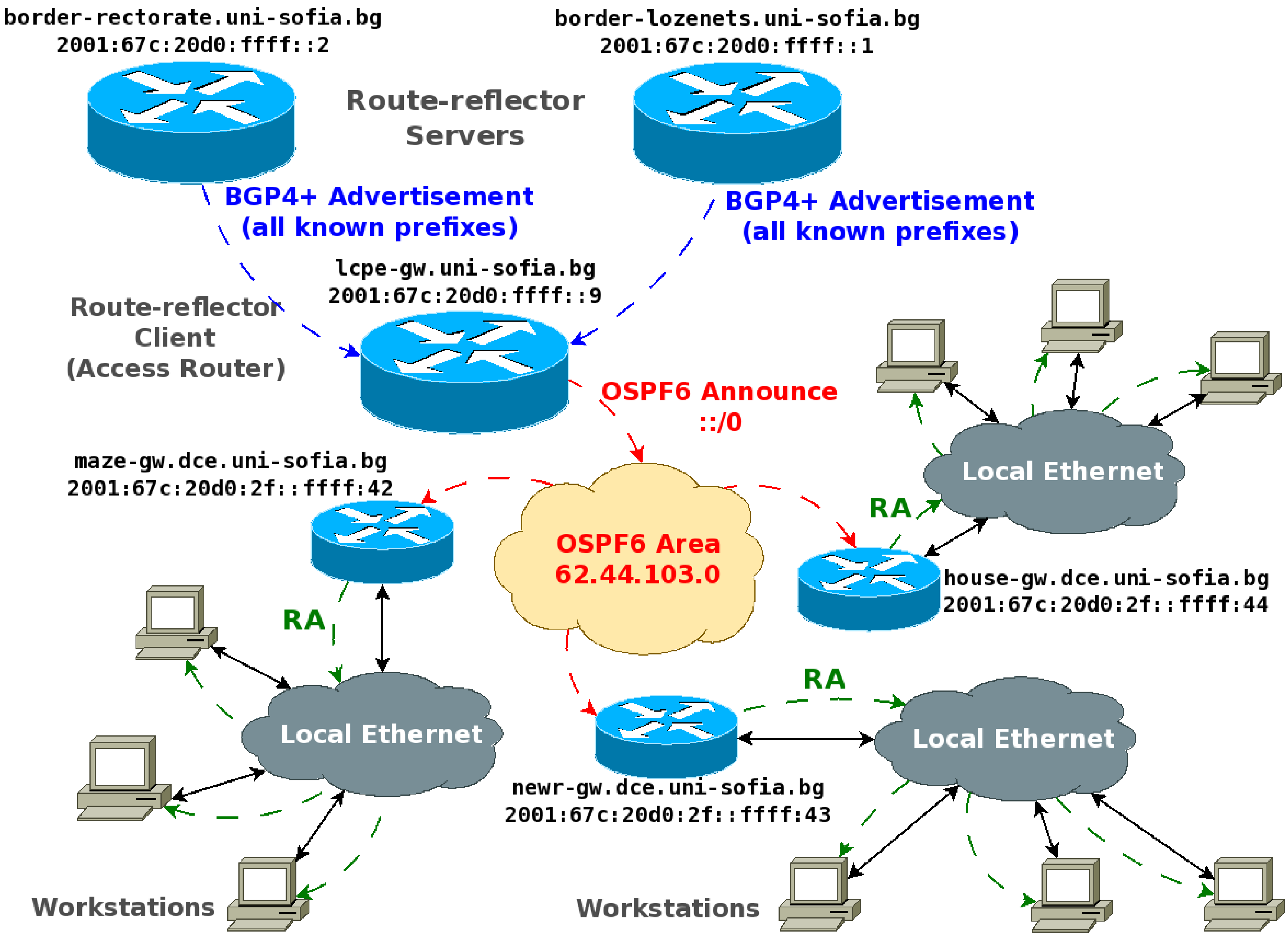
```
ipv6 route ::/0 lo blackhole
```

and describe it for redistribution in `/etc/quagga/ospf6d.conf`:

```
router ospf6
 router-id 62.44.103.41
 redistribute static route-map REDISTRBUTE_STATIC
 interface eth1.100 area 62.44.103.40
!
ipv6 prefix-list REDISTRBUTE_STATIC seq 5 permit ::/0
!
route-map REDISTRBUTE_STATIC permit 10
 match ipv6 address prefix-list REDISTRBUTE_STATIC
!
```

Default routes as prefixes and their origination and redistribution

Self-origination: distribution of the default route ::/0 into OSPF6 stub area



Default routes as prefixes and their origination and redistribution

Self-origination: creation and redistribution of the default route `2000::/3` into OSPFv3 (similar is for RIPng)

Create route **`2000::/3`** in `/etc/quagga/zebra.conf`:

```
ipv6 route 2000::/3 lo blackhole
```

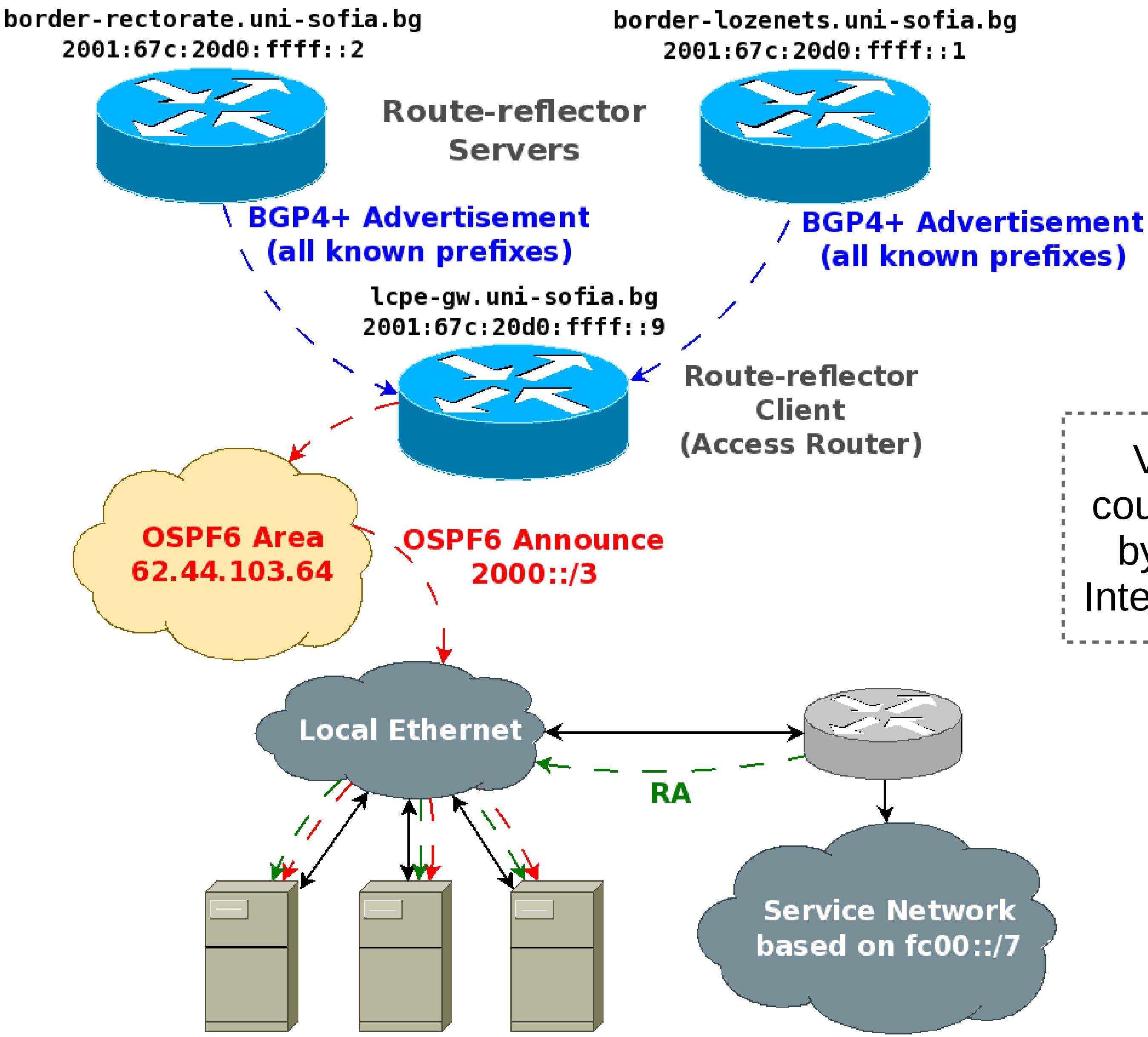
It does *not* block the unicast IPv6 transport because bgpd supplies RIB with *more specific* prefixes (2000::/3 is the least specific unicast prefix)

and describe it for redistribution in `/etc/quagga/ospf6d.conf`:

```
router ospf6
 router-id 62.44.103.41
 redistribute static route-map REDISTRBUTE_STATIC
 interface eth1.100 area 62.44.103.40
!
ipv6 prefix-list REDISTRBUTE_STATIC seq 5 permit 2000::/3
!
route-map REDISTRBUTE_STATIC permit 10
 match ipv6 address prefix-list REDISTRBUTE_STATIC
!
```

Default routes as prefixes and their origination and redistribution

Self-origination: distribution of the default route 2000::/3 into OSPF6 stub area

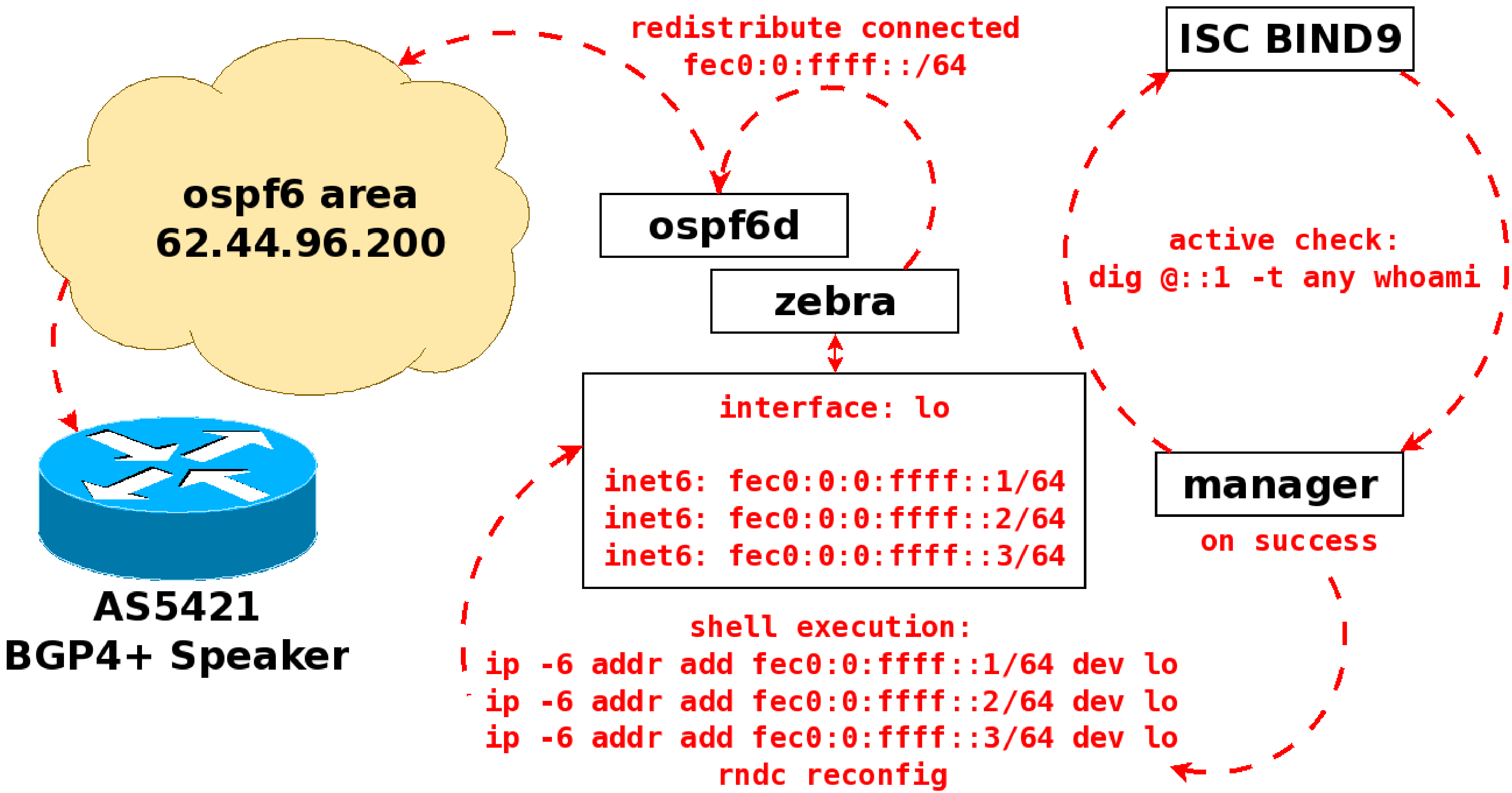


Virtual Machine environment could access the service network by default. The access to IPv6 Internet is through different router.

IPv6-based DNS Anycast

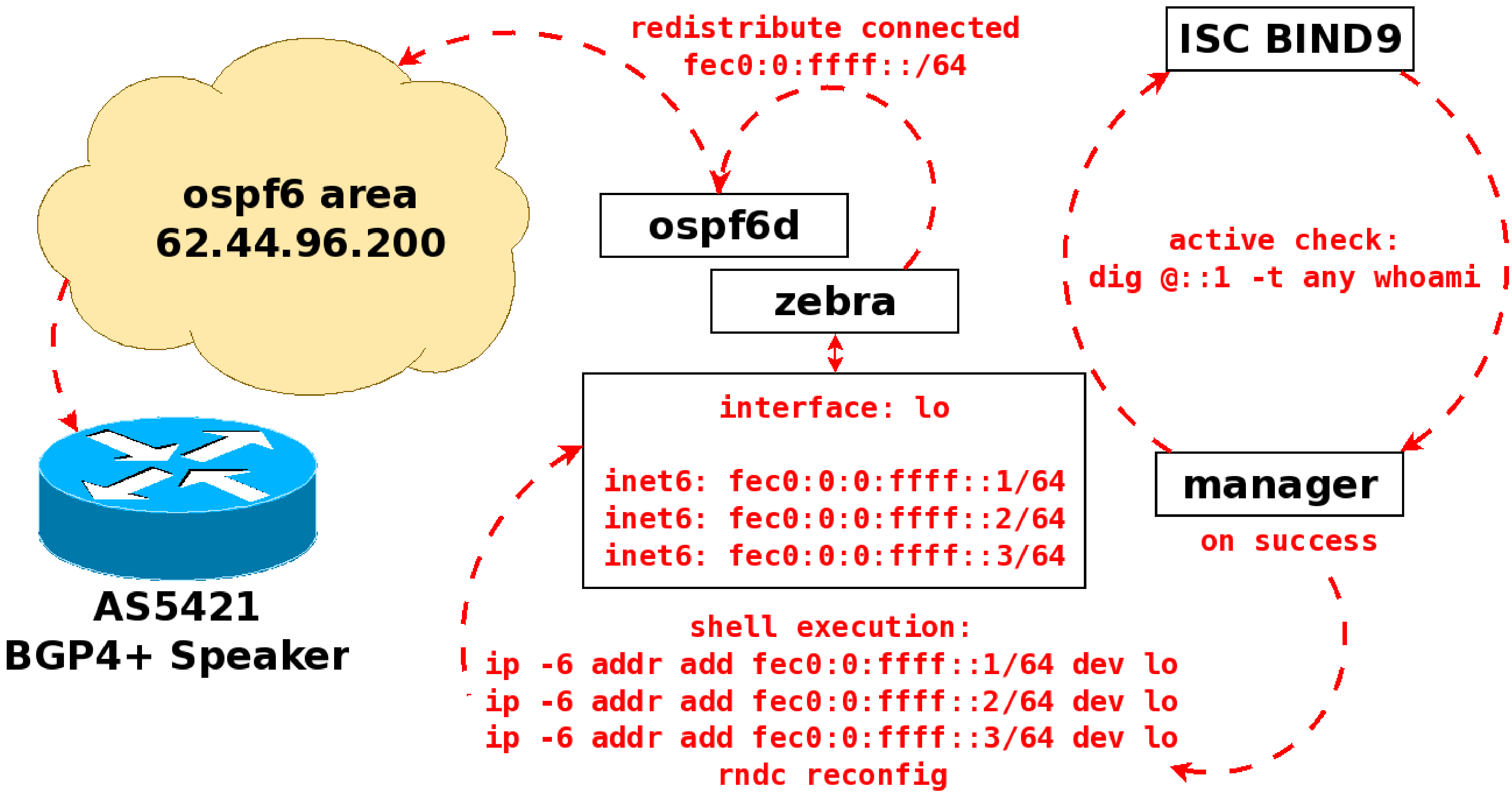
IPv6-based DNS Anycast

How does DNS anycast node work: (re-)starting the prefix redistribution



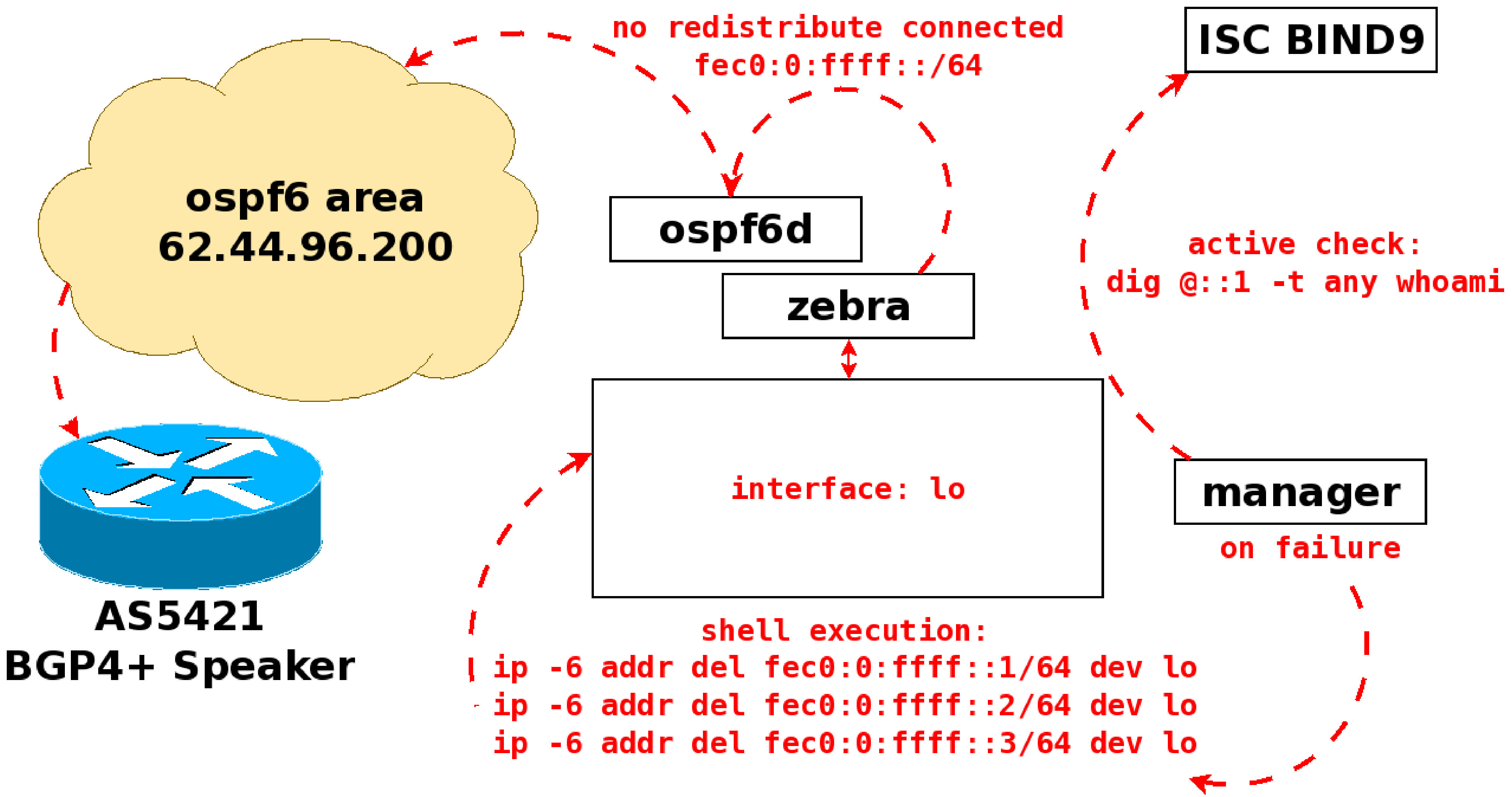
IPv6-based DNS Anycast

How does DNS anycast node work: (re-)starting the prefix redistribution



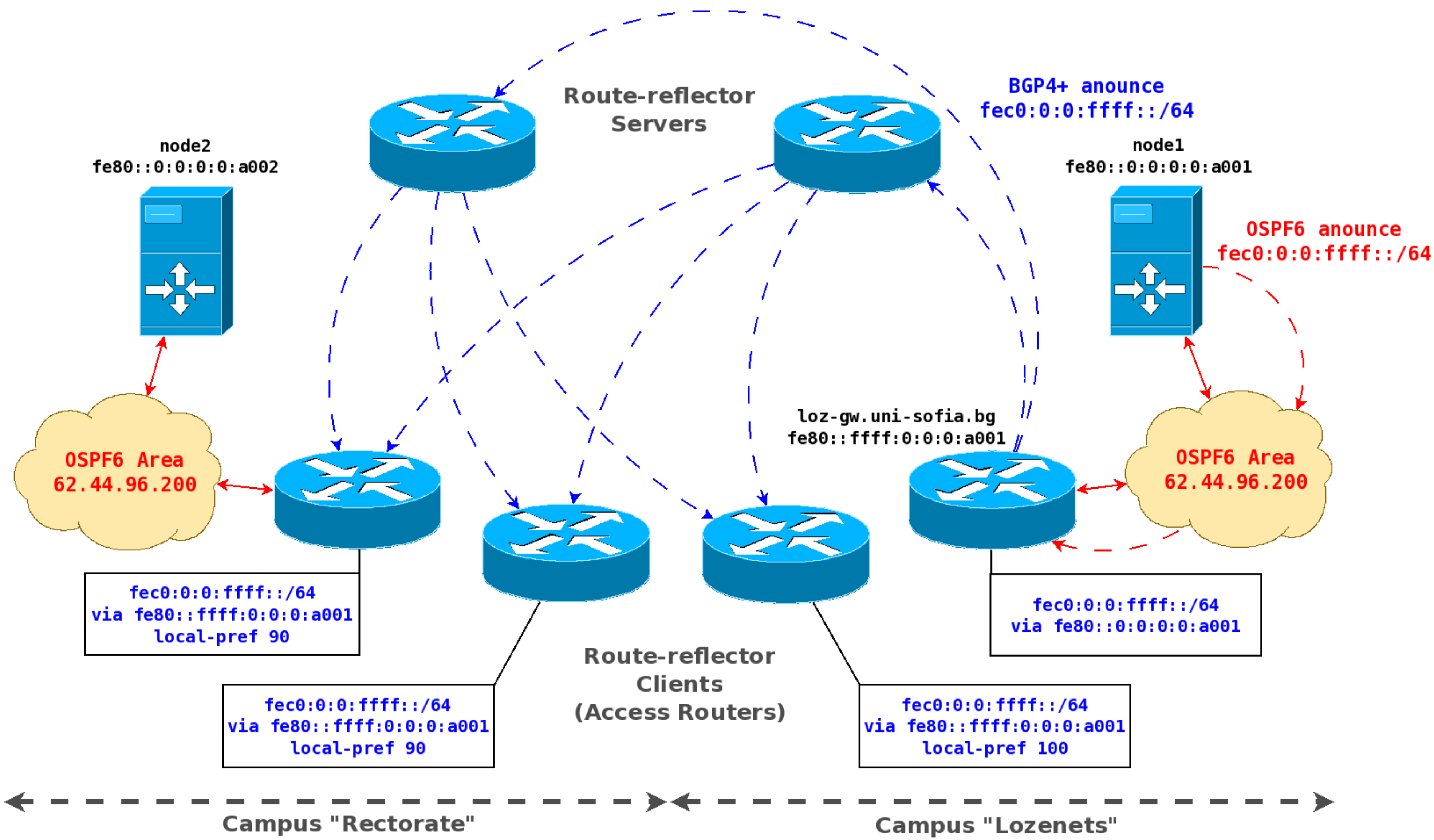
IPv6-based DNS Anycast

How does DNS anycast node work: stopping the prefix redistribution on BIND9 failure



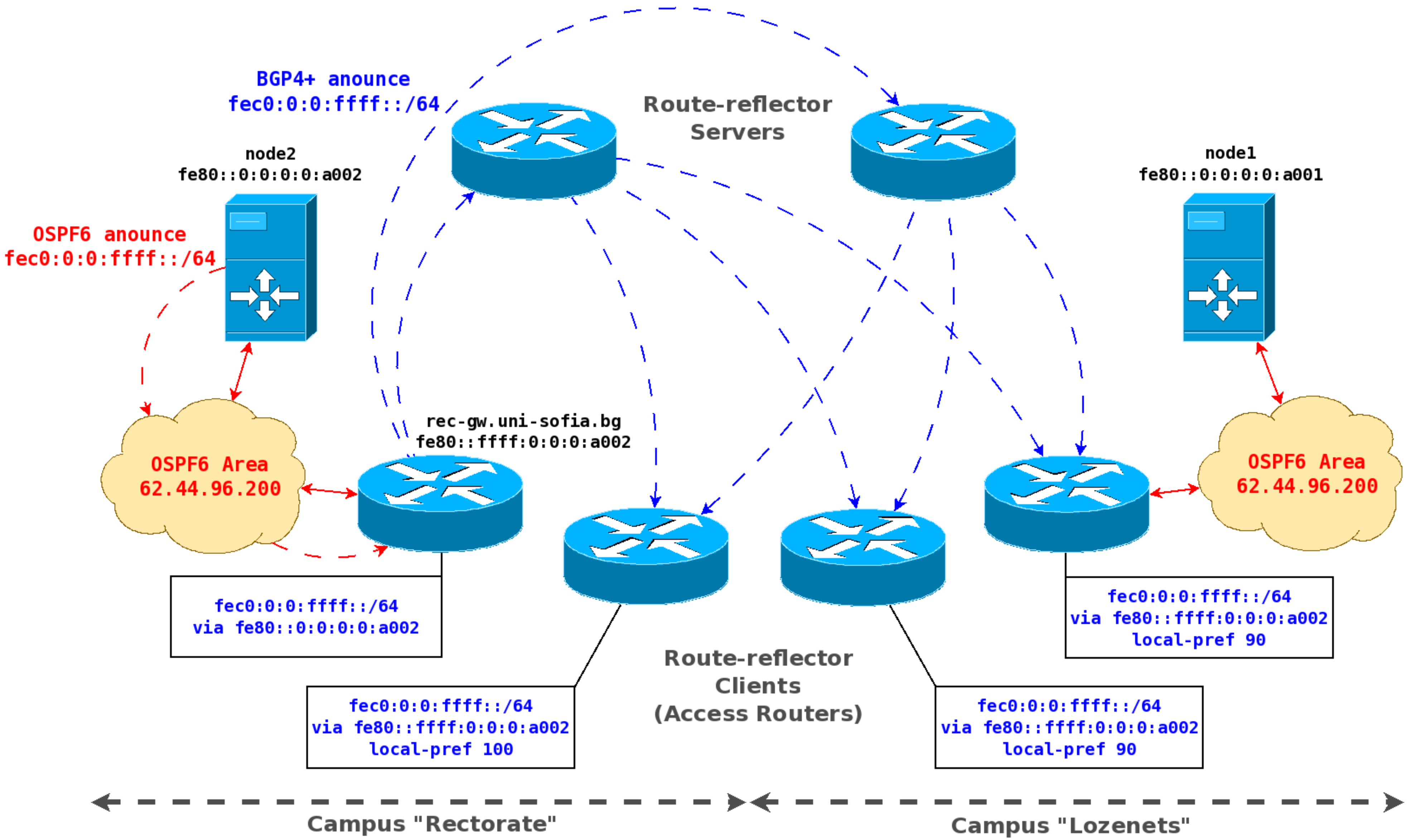
IPv6-based DNS Anycast

How does DNS anycast node work: high-availability



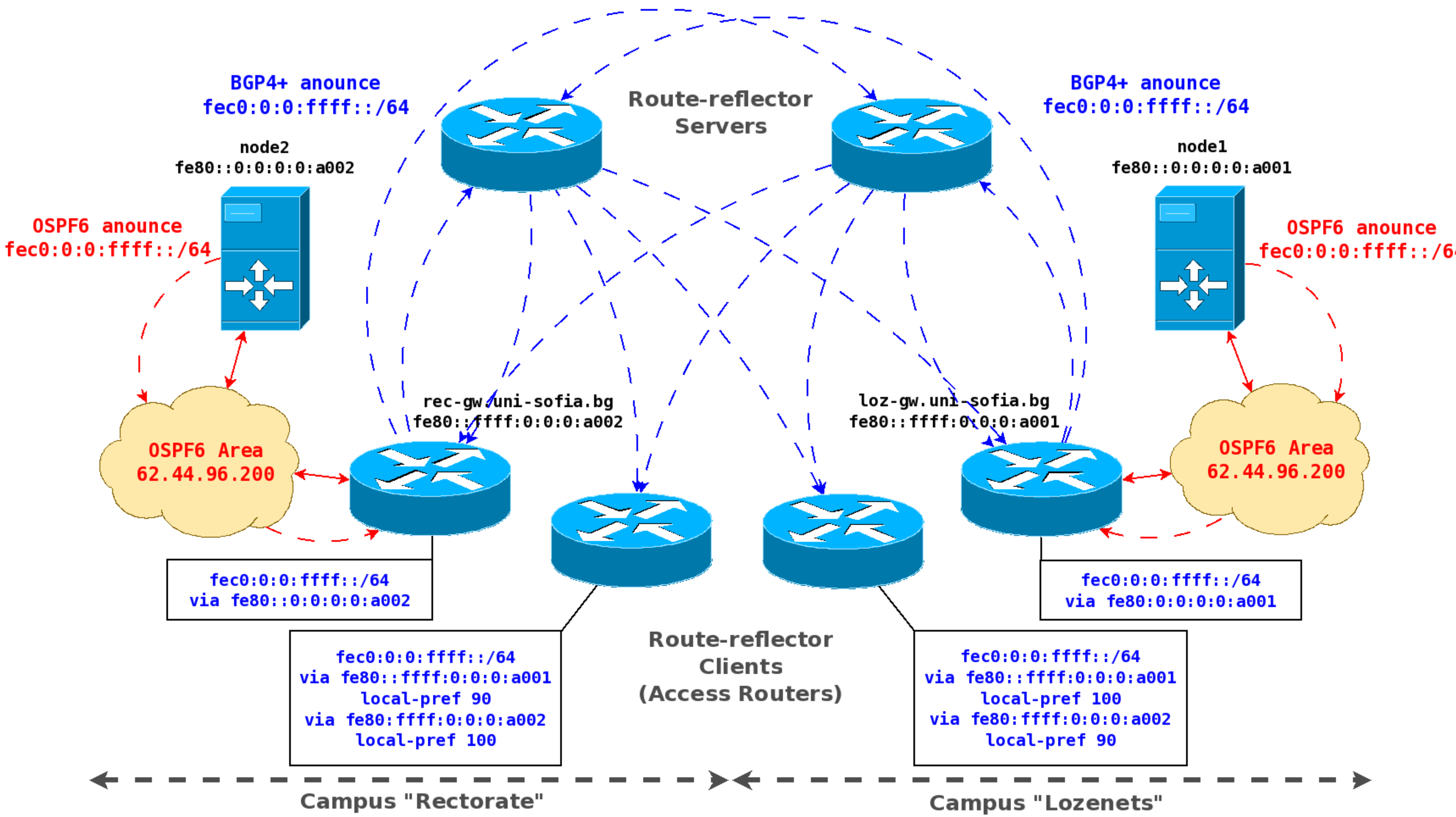
IPv6-based DNS Anycast

How does DNS anycast node work: high-availability



IPv6-based DNS Anycast

How does DNS anycast node work: high-throughput



Thank You very much for Your attention!
Questions?