

# Směrovací démon BIRD

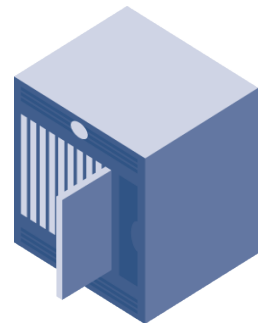
CZ.NIC z. s. p. o.

Ondřej Filip / [ondrej.filip@nic.cz](mailto:ondrej.filip@nic.cz)

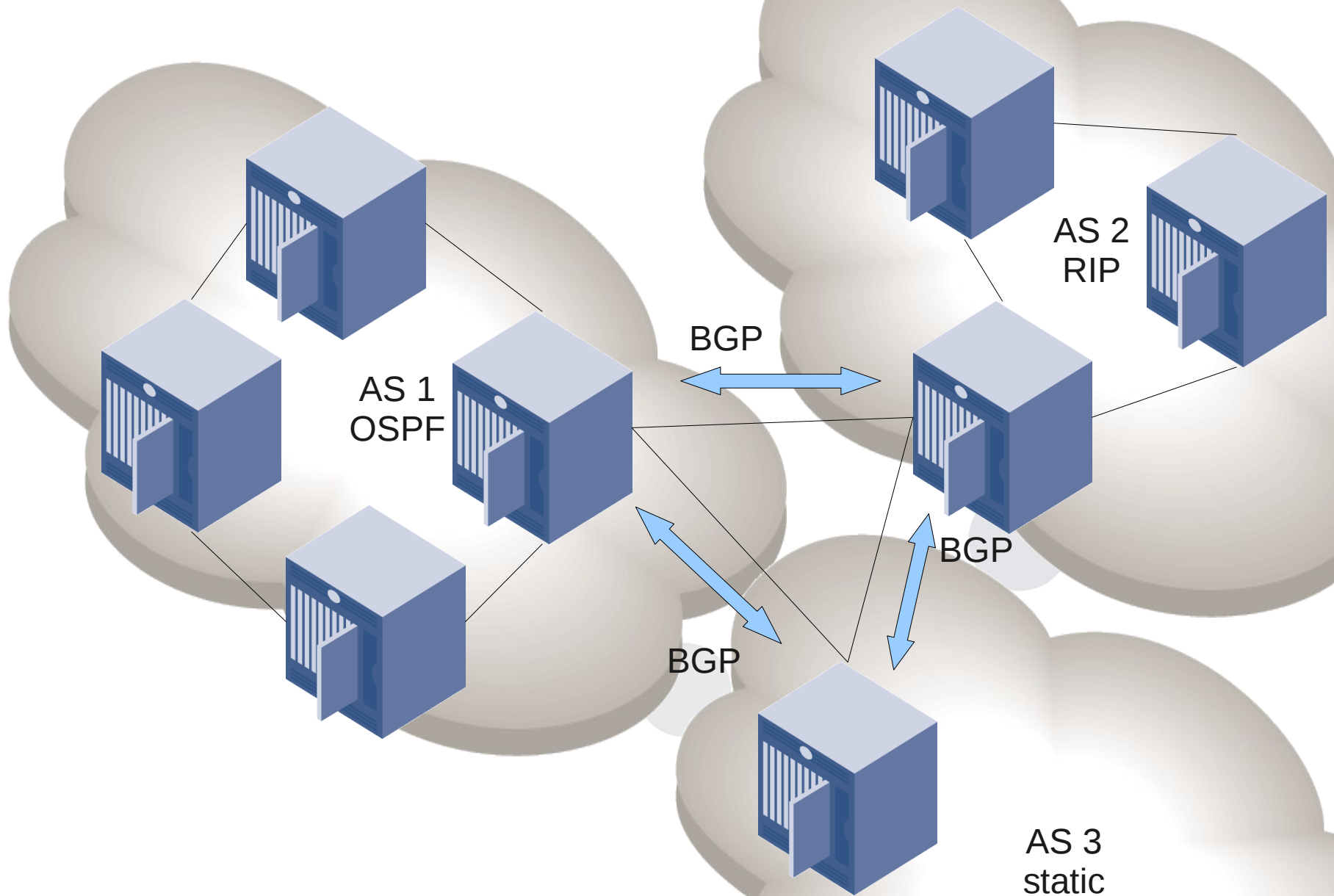
8. 6. 2010 – IT10

# Směrování a forwarding

- Router - zařízení připojené k více sítím
- Umí přeposlat „cizí“ zprávu - forwarding
- Cestu pozná podle směrovací (routovací) tabulky
- Sestavování routovací tabulky – routing
  - Statické
  - Dynamické
    - Interní - uvnitř AS rychlé, důvěřivé, přesné
      - RIP, OSPF
    - Externí (mezi AS, pomalé, filtering, přibližné – pouze BGP)



# Rozdělení směrovacích protokolů



# Směrovací démon

- Na Linuxu (a ostatních UNIXech) – uživatelská aplikace mimo jádro, forwarding v jádře
- Obvykle implementuje více směrovacích protokolů
- Směrovací politika - filtrování
- Quagga (Zebra) – Cisco syntax <http://www.quagga.net>
- OpenBGPD - <http://www.openbgpd.org>
- GateD – zastaralý, ne volná licence
- BIRD

# Historie projektu

- Start projektu v roce 1999
- Seminární projekt – MFF UK Praha
- Projekt uspán
- Drobné probuzení v letech 2003 a 2006 (CESNET)
- Plně obnoveno na přelomu 2008/2009 v rámci Laboratoří CZ.NIC - <http://labs.nic.cz>



# Cíle projektu

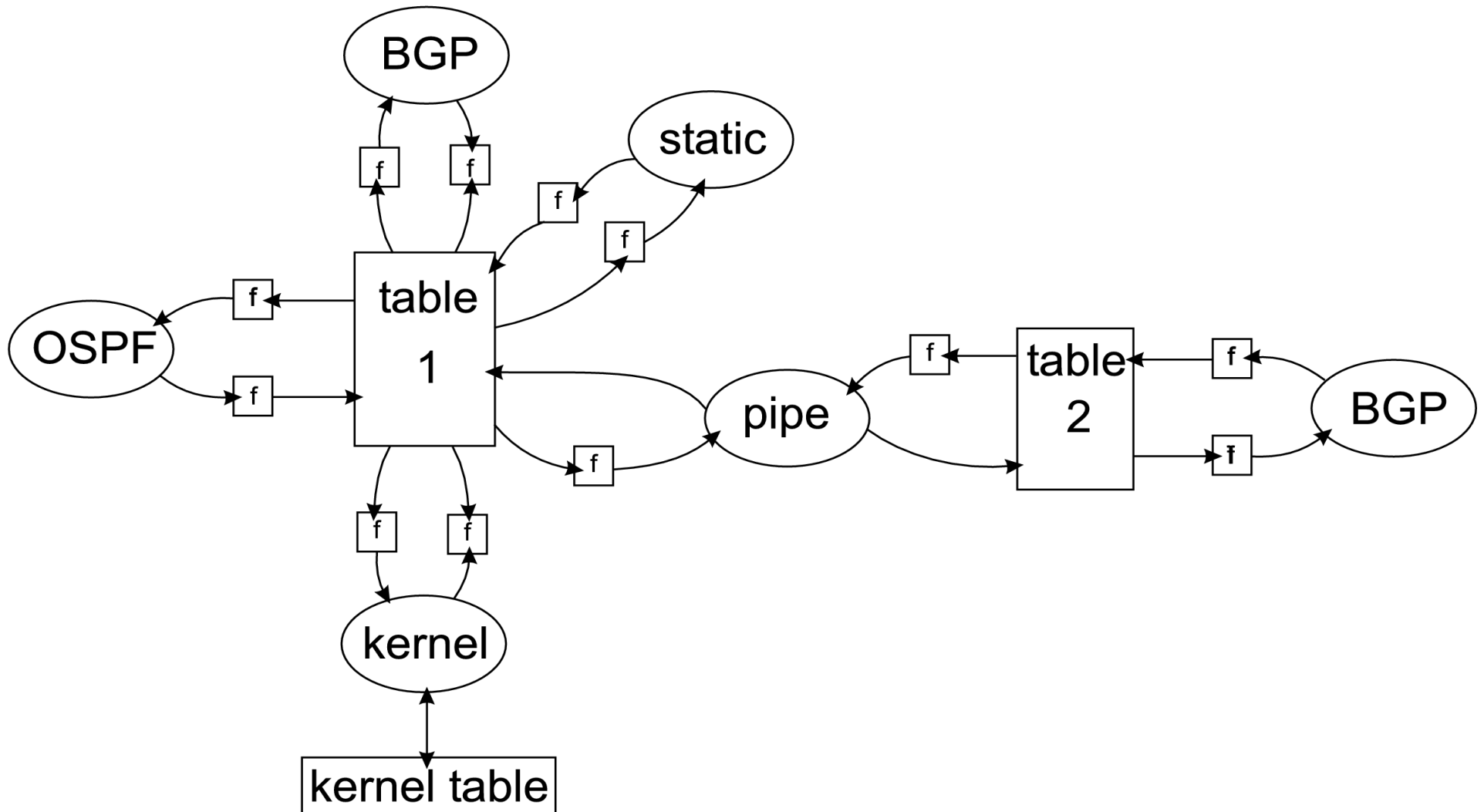
- Opensource směrovací démon – alternativa k tehdejšímu démonu Quagga/Zebra (GateD)
- Rychlý a efektivní
- Portabilní, modulární
- Podpora současných směrovacích protokolů
- IPv6 a IPv4 v jednom zdrojovém kódu (dvojitý překlad)
- Snadná konfigurace a rekonfigurace (!)
- Silný filtrovací jazyk

# Vlastnosti



- Portabilní – Linux, FreeBSD, NetBSD, OpenBSD
- Podpora IPv4 i IPv6
- Static, RIP, OSPF, BGP - Route reflektor, Směrovací server (Route server) ASN32 (ASPLAIN), MD5
- Více směrovacích tabulek (RIBs) (interně, ale také synchronizace s OS, pokud to podporuje)
- Protokol PIPE (!)
- Více směrovačů, route serverů a pod. na 1 systému
- Efektivní, silná konfigurace, Silný jazyk pro filtrování
- Příkazová řádka (show, restart, ...)

# Design





# Příklad konfigurace

```
log "/var/log/bird.log" all;

router id 193.51.100.238;

protocol static {
    route 10.0.0.0/8 drop;
    route 172.16.0.0/12 drop;
    route 192.168.0.0/16 drop;
}
filter bgp_out {
    if (net = 192.175.48.0/24 ) &&
        (source = RTS_DEVICE) then accept;
    else reject;
}
protocol bgp NIX_1 {
    local as 112;
    neighbor 193.51.100.235 as 6981;
    import all;
    export filter bgp_out;
}
```

# Příkazová řádka



```
bird> show route
10.0.0.0/8      via 200.30.10.3 on eth2 [ospf1 13:10] E2 (150/5/1000)
127.0.0.0/8    dev lo [direct1 13:09] (240)
200.30.20.0/24 via 200.30.10.3 on eth2 [ospf1 13:10] I (150/10)
200.30.10.0/24 dev eth2 [direct1 13:09] (240)
                dev eth2 [ospf1 13:10] I (150/5)
200.0.10.0/24  dev eth0 [direct1 13:09] (240)
                dev eth0 [ospf1 13:09] I (150/5)
172.16.0.0/16  via 200.30.10.3 on eth2 [ospf1 13:10] E2 (150/5/1000)
195.47.235.0/24 via 194.50.100.246 on eth1 [NIX2 Apr11] (100)[AS688i]
                via 194.50.100.245 on eth1 [NIX1 Apr25] (100)[AS688i]

bird>
bird> show route protocol ospf1
10.0.0.0/8      via 200.30.10.3 on eth2 [ospf1 13:10] E2 (150/5/1000)
200.30.20.0/24  via 200.30.10.3 on eth2 [ospf1 13:10] I (150/10)
200.30.10.0/24  dev eth2 [ospf1 13:10] I (150/5)
200.0.10.0/24   dev eth0 [ospf1 13:09] I (150/5)
172.16.0.0/16   via 200.30.10.3 on eth2 [ospf1 13:10] E2 (150/5/1000)
```

# Příkazová řádka



```
bird> show route for 127.0.0.1
127.0.0.0/8      dev lo [direct1 13:09] (240)
```

```
bird> show route filter bgp_out
192.175.48.0/24 dev dummy0 [direct1 Apr1] (240)
```

```
bird> show route count
1469 of 1469 routes for 849 networks
```

```
bird> show route export NIX_1
192.175.48.0/24 dev dummy0 [direct1 Apr1] (240)
```

```
bird> show route where 127.0.0.5 ~ net
0.0.0.0/0      via 195.47.235.1 on eth0 [static1 Apr1](200)
127.0.0.0/8    dev lo [direct1 Apr1] (240)
```

```
bird> show route filter {if 127.0.0.5 ~ net then accept;}
0.0.0.0/0      via 195.47.235.1 on eth0 [static1 Apr1](200)
127.0.0.0/8    dev lo [direct1 Apr1] (240)
```

# BIRD vs Quagga vs OpenBGPD

- Import plné IPv4 BGP tabulky ~300k položek
- Porovnání na Linuxu
- Více měření
- Spotřeba CPU (CPU time, sec) a paměti (MB)
- Test 1 – se synchronizací s směrovací tabulkou OS (Quagga - bgpd+zebra)
- Test 2 – bez synchronizace (i OpenBGPD)

# BIRD vs Quagga vs OpenBGPD

- Test 1

Daemon	Memory (MB)	CPU (sec)
Zebra	90 + 77 = <b>167</b>	32 + 120 = <b>152</b>
BIRD	<b>30</b>	<b>14</b>

- Test 2

Daemon	Memory (MB)	CPU (sec)
Zebra	<b>87</b>	<b>30</b>
BIRD	<b>30</b>	<b>7</b>
OpenBGP	33 + 18 = <b>51</b>	10 + 7 = <b>17</b>

# Příklad filtru

```
function avoid_martians()  
prefix set martians;  
{  
    martians = [ 169.254.0.0/16+, 172.16.0.0/12+,  
                192.168.0.0/16+, 10.0.0.0/8+, 224.0.0.0/4+,  
                240.0.0.0/4+, 0.0.0.0/32-, 0.0.0.0/0{25,32},  
                0.0.0.0/0{0,7} ];  
  
    # Avoid RFC1918 networks  
    if net ~ martians then return false;  
  
    return true;  
}
```

# Příklad filtru - Quagga

```
ip prefix-list avmart seq 10 deny 169.254.0.0/16 le 32
ip prefix-list avmart seq 20 deny 172.16.0.0/12 le 32
ip prefix-list avmart seq 30 deny 192.168.0.0/16 le 32
ip prefix-list avmart seq 40 deny 10.0.0.0/8 le 32
ip prefix-list avmart seq 50 deny 224.0.0.0/4 le 32
ip prefix-list avmart seq 60 deny 240.0.0.0/4 le 32
ip prefix-list avmart seq 70 deny 0.0.0.0/0 le 7
ip prefix-list avmart seq 80 deny 0.0.0.0/0 ge 25
ip prefix-list avmart seq 90 permit any
```

# Příklad filtru - OpenBGPD

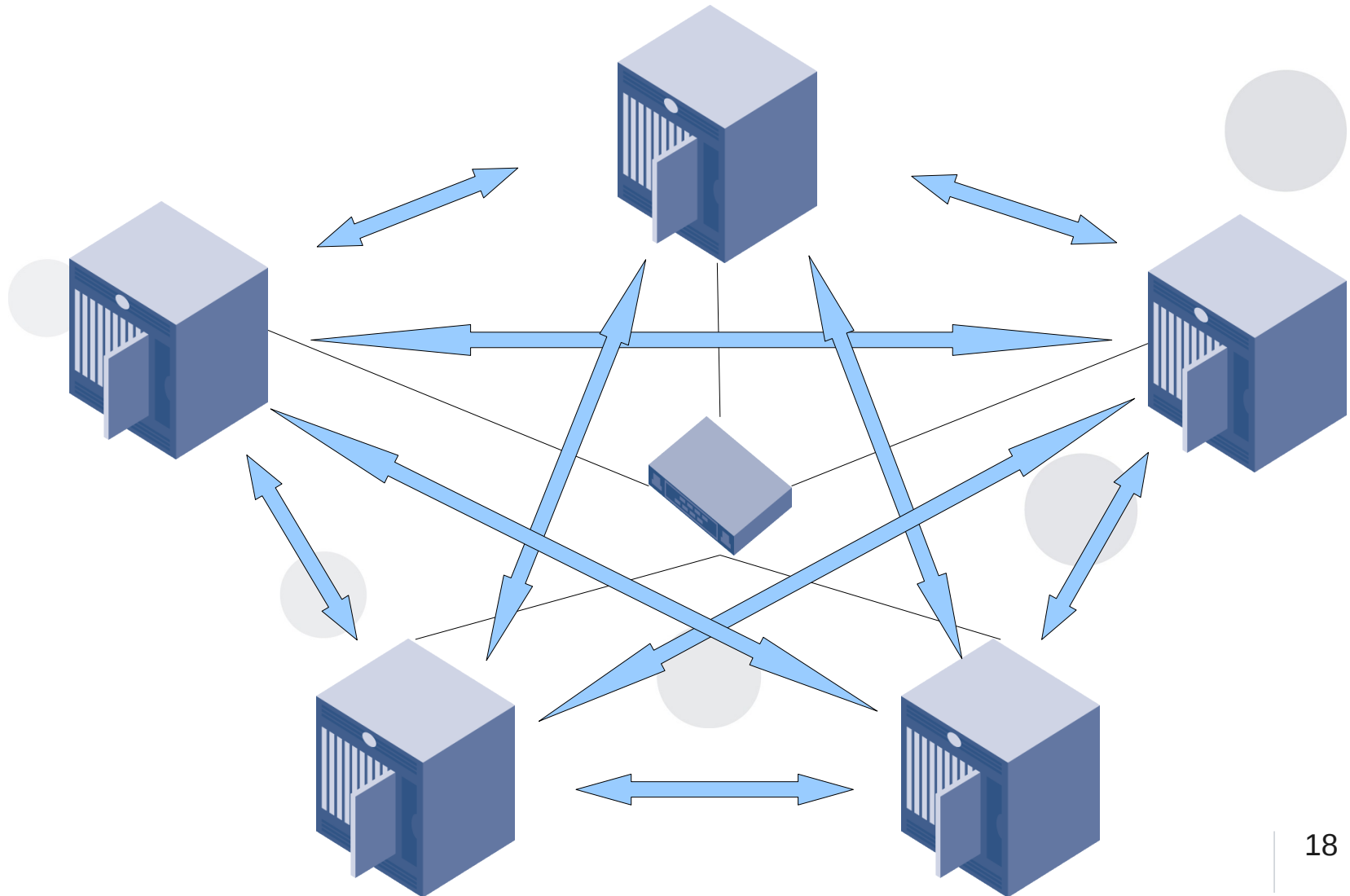
```
# filter bogus networks
allow from any prefixlen 8 – 24
deny from any prefix 0.0.0.0/0
deny from any prefix 10.0.0.0/8 prefixlen >= 8
deny from any prefix 172.16.0.0/12 prefixlen >= 12
deny from any prefix 192.168.0.0/16 prefixlen >= 16
deny from any prefix 169.254.0.0/16 prefixlen >= 16
deny from any prefix 192.0.2.0/24 prefixlen >= 24
deny from any prefix 224.0.0.0/4 prefixlen >= 4
deny from any prefix 240.0.0.0/4 prefixlen >= 4
```



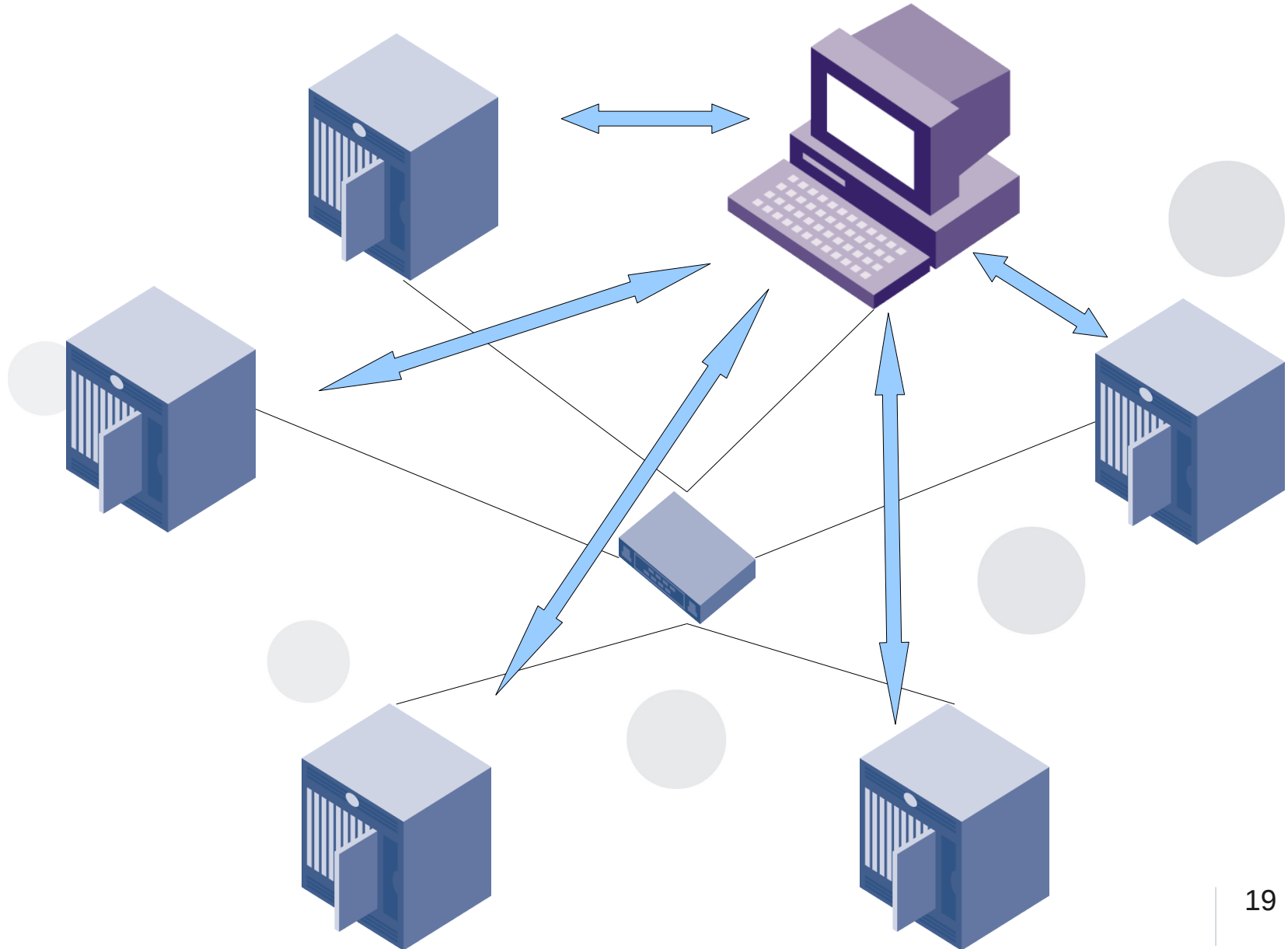
# Příklad filtru

```
function asmatch()  
int set asnums;  
{  
    asnums = [ 11111, 22222, 33333, 44444, 55555,  
              66666, 77777, 88888, 99999, 100..200 ];  
  
    # Check originating AS number  
    if bgp_path.last ~ asnums then return true;  
  
    return false;  
}
```

# IXP bez route serveru



# IXP s route serverem



# Router server

- Route server – snížení počtu BGP relací
- Každý člen IXP –  $n-1$  BGP relací - celkem  $n*(n-1)/2$
- Nutnost konfigurace s každým novým členem
- Zátěž CPU routeru
- Řešením je route server – všichni jsou pouze připojeni k RS – 1 relace na router –  $n$  celkem
- RS musí skrýt svou existenci
- Kvalitní filtrování – tisíce položek
- Ačkoliv jsou všichni propojeni k route serveru, nemusí všichni být propojeni se všemi – signalizace RS
- Kvalita RS, efektivita RS

# Příklad filtru – router server

- Politika route serveru - NIX.CZ

Pořadí	Komunita	Akce
1	0:<peer-as>	Nepropaguj routu <peer-as>
2	47200:<peer-as>	Propaguj routu <peer-as>
3	0:47200	Nepropaguj nikomu
4	47200:47200	Propaguj všem

# Příklad filtru – router server

- Každý ISP zapojený do route serveru neztrácí svobodu definice své vlastní směrovací politiky
- Příklad 1 – chci posílat svou síť pouze CZ.NICu
  - 10.0.0.0/8 community 47000:25192, 0:47000
- Příklad 2 – chci posílat všem krom CZ.NICu
  - 10.0.0.0/8 community 0:25192, 47200:47200

# Router server - Quagga

```
ip community-list standard C-0-10001 permit 0:10001
ip community-list standard C-0-47200 permit 0:47200
ip community-list standard C-47200-10001 permit 47200:10001
```

```
route-map Policy10001 deny 10
  match community C-0-10001
!
route-map Policy10001 permit 20
  match community C-47200-10001
!
route-map Policy10001 deny 30
  match community C-0-47200
!
route-map Policy10001 permit 40
!
```

# Příklad filtru – route server

```
define myas = 47200;
```

```
function bgp_out(int peeras)
```

```
{
```

```
    if ! (source = RTS_BGP ) then return false;
```

```
    if (0,peeras) ~ bgp_community then return false;
```

```
    if (myas,peeras) ~ bgp_community then return true;
```

```
    if (0, myas) ~ bgp_community then return false;
```

```
    return true;
```

```
}
```

```
protocol bgp R25192x1 {
```

```
    local as myas;
```

```
    neighbor 194.50.100.13 as 25192;
```

```
    import where bgp_in(25192);
```

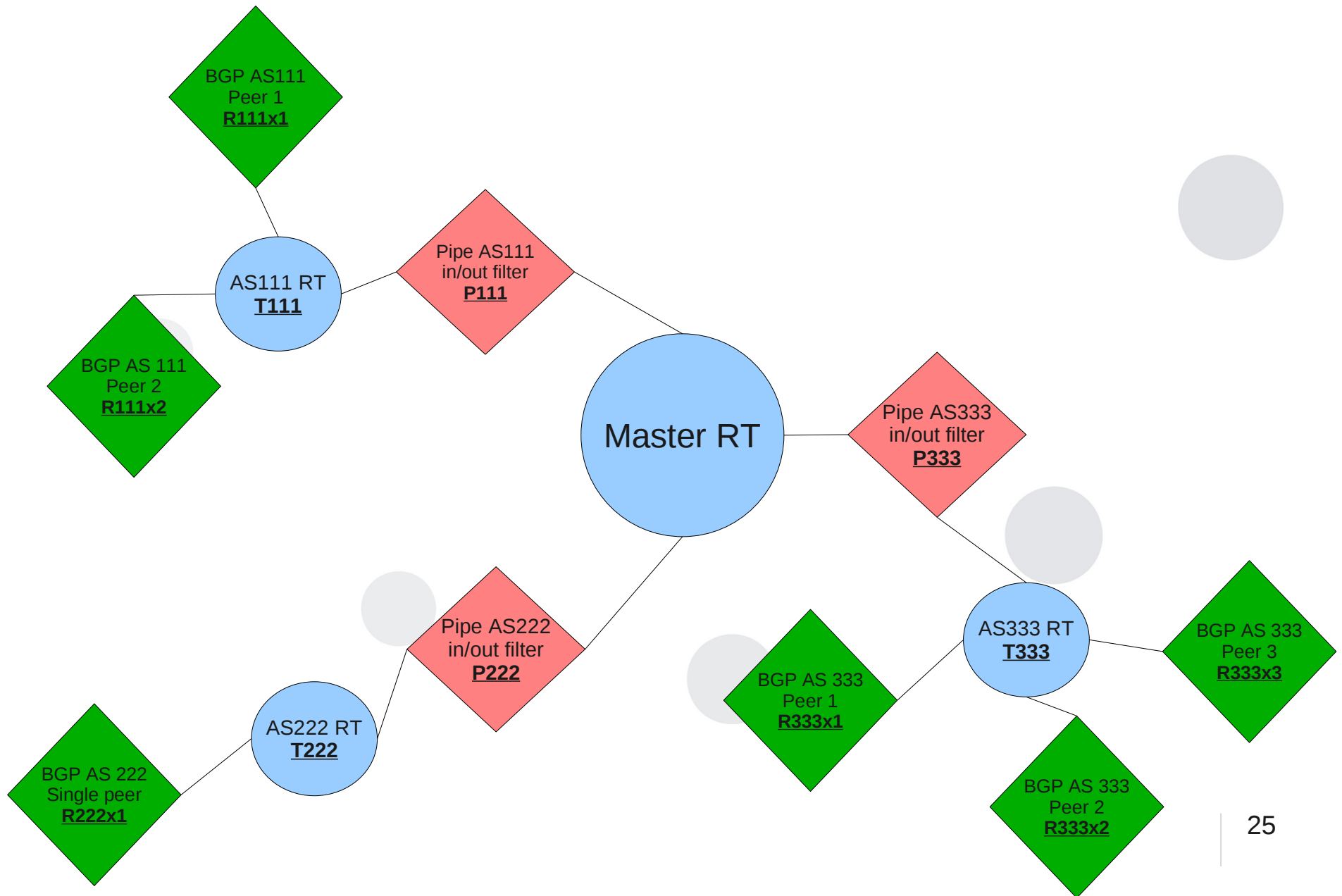
```
    export where bgp_out(25192);
```

```
    rs client;
```

```
}
```



# BIRD jako route server



# Nasazení BIRDa

Detaily na RIPE60



PAIX - (PAO, SEA, CHI, DAL, NYC, ATL, TOR, TYS)

# Ocenění



Conspicuous Contribution to LINX Award 2010  
To Ondrej Filip and his talented Czech colleagues,  
supported by NIX.CZ and CZ.NIC,  
for outstanding effort in creating BIRD, an open source  
route server platform of great benefit to IXPs



# Jiné aplikace

- AS112 server v NIX.CZ
- BGP/OSPF router u menších ISP
- Router pro některé anycastové uzly CZ.NICu
- Používán v malých embedded systémech – součást firmware některých WiFi AP (OpenWRT)
- Součást analyzátoru BGP tabulek

# Budoucí vývoj

- Nová verze cca jednou měsíčně
- Nové webové stránky
- IPv6 Route advertisement
- Lightweight CLI (OpenWRT)
- Route flap dampening
- Solaris (CZ.NIC DNS anycasting)
- ...
- Záleží na přáních uživatelů

# Závěr



- Český projekt – ale pro celosvětovou komunitu – kvalitní dokumentace v AJ
- Alternativa k dnes nejrozšířenějšímu démonu Quagga
- Efektivní, mocný filtrovací jazyk
- Nasazen v nejdůležitějších centrech Internetu
- Jeden z OpenSource projektů Laboratoří CZ.NIC – příspěvek ke stabilitě a rozvoji Internetu



# ¿Dotazy?

Děkuji za pozornost!  
<http://labs.nic.cz>

