Bod. 11 Detekce routeru a problémy

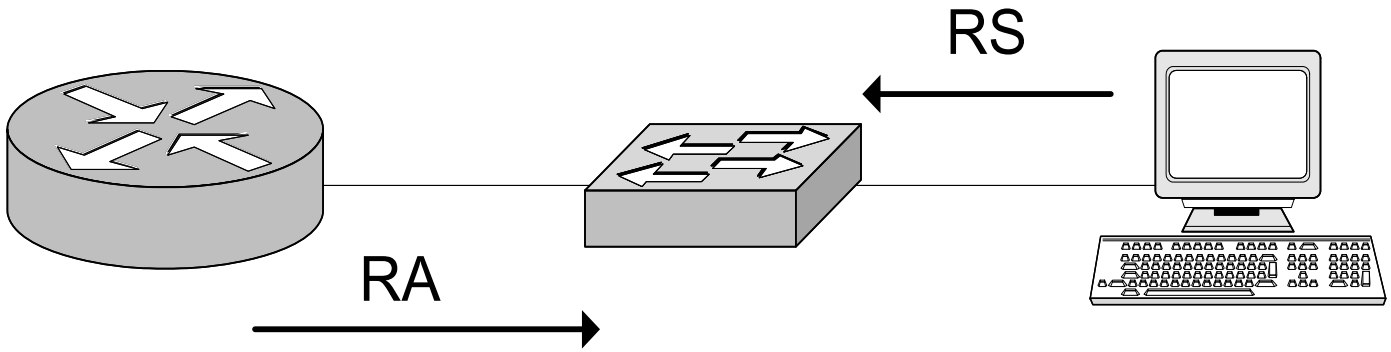
Konfigurace IPv6 uzlu:

- Statická
  - Plně statická, onlink prefix a router napevno
    - doporučeno pro servery, aktivní prvky
  - Pouze adresa + prefix, detekce routeru z RA (problém onlink prefixů)
- Dynamická
  - Automatická bezestavová (SLAAC), detekce routeru z RA
  - Automatická stavová (DHCPv6), detekce routeru z RA

Jak funguje detekce routeru?

- Součást protokolu **Neighbor Discovery** (RFC4861, 97 stran): RS, RA, NS, NA, Redirect
  - IPv4 ARP, RDISC a ICMP Redirect
- Více cílů:
  - Získat prefix(y) lokální sítě (pro vygenerování automatické adresy) a jejich parametr(y)
  - Získat adresu(y) routerů pro lokální síť
  - Získat parametry pro automatickou konfiguraci: M,O flag, hop limit, MTU
  - RFC 5006 doplňuje adresu RDNS (rekurzivního DNS)
  - RFC XXXX (5006bis) doplňuje DNS search list

Jak funguje detekce routeru?



- Uzel vyšle ICMPv6 paket Router Solicitation (vyžádání routeru) na multicast adresu *all-routers* (FF02::2)
- Všechny směrovače v síti odpoví paketem Router Advertisement (inzerce routeru) buď na unicast link-local adresu nebo na *all-nodes* (FF02::1)
  - Relying on unauthenticated broadcast packet to determine where host should send traffic to
  - Hacking all WS 2008 & Vista in zero-time (viz MS10-009)

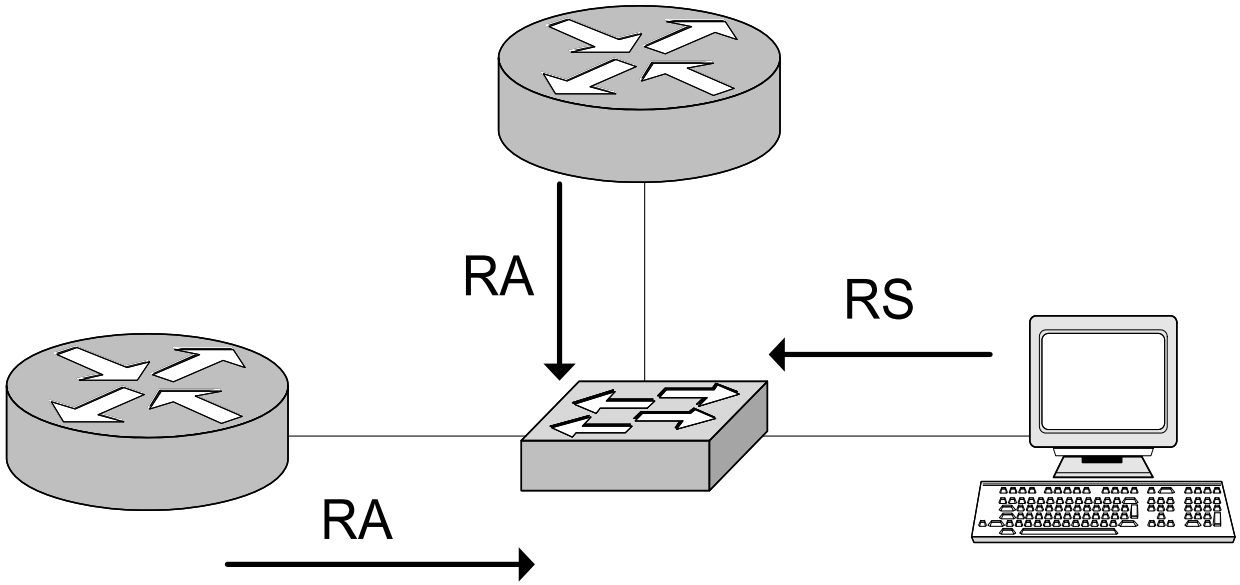
Jak funguje detekce routeru?

- RA je volitelné, musí být na routeru zapnuté
- Klient může RA ignorovat (obvykle přijímá)
- Router může inzerovat více IPv6 prefixů, každý může mít jiné parametry:
  - On-link=0/1 (lokální sítě, lze posílat přímo)
  - On-link Valid lifetime (0-inf)
  - Auto=0/1 – lze použít pro automatickou konfiguraci (SLAAC)
  - Auto Preferred lifetime (0-inf)
- Router se může zrušit (router lifetime=0)
  - problém kontroly platnosti – obvykle nevadí jiná MAC adresa (zdrojová LLA musí být stejná)

Co s tím?

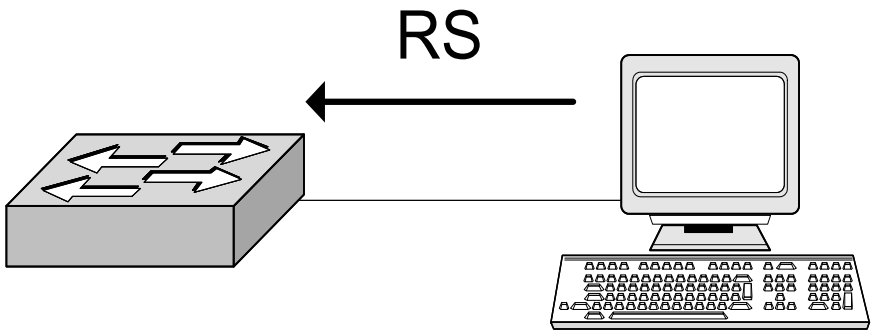
- Uzel shrnuje všechny přijaté informace:
  - Link-local adresy (LLA) platných routerů uloží do seznamu implicitních routerů (s časem platnosti)
  - Pokud je čas platnosti 0, zruší router v seznamu
  - Uzel musí udržovat minimálně dvouprvkový seznam (doporučeno více)
    - co když přeteče?
  - Inzerované prefixy s nenulovým časem platnosti uloží do **seznamu prefixů** (onlink, auto), ostatní vyhodí
    - pro každý Auto=1 vygeneruje bezestavovou adresu

Výběr routeru



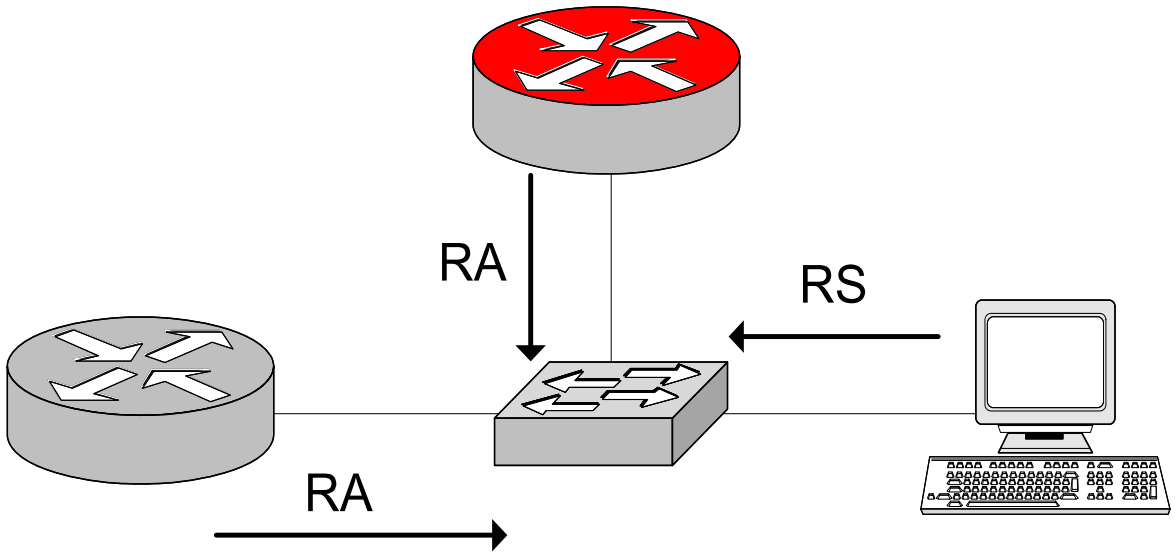
- Dostupnost routeru je ověřována pomocí ND/NUD
- Preferovány jsou ty, u nichž je dostupnost ověřena
- Pokud je jich více, je doporučeno round-robin
- Reakce na výpadek – 5 až 38 sekund

Co když router není?



- RFC 2461 5.2: The sender performs a longest prefix match against the Prefix List to determine whether the packet's destination is on- or off-link. If the destination is on-link, the next-hop address is the same as the packet's destination address. Otherwise, the sender selects a router from the Default Router List. **If the Default Router List is empty, the sender assumes that the destination is on-link. (zrušeno v RFC4861)**

Co když je falešný?



- Rogue IPv6 Router Advertisement Problem Statement (draft RFC, verze 01 z 7.6.2010)
  - Chyba správce (např. špatně nastavená VLAN)
  - Chyba uživatele (NB s Win/ICS jako 6to4 gateway)
  - Záměrné podvržení (hacker)

Co s tím?

1. Statická konfigurace:
   - Určitě u aktivních prvků a serverů!
   - Není reálná u NB, přenosných zařízení
   - Většinou není reálná u klientských PC
2. Router Preference Option (RFC4191)
3. ACL na L2 prvcích
4. RA snooping, RA guard
5. Monitorování a obrana (*rafixd*, *ramond*, *ndpmon*)
6. Paketové filtry u klientů (zahazovat neplatné RA)
7. Doplnění default router/prefix do DHCPv6
8. SeND (RFC3971)

Router Preference Option (RFC4191)

- Doplňuje do parametrů RA preferenci routeru a nový option Route Information s preferencí:
  - 01 High
  - 00 Medium (default)
  - 11 Low
  - 10 Reserved - MUST NOT be sent
- When a type B (rozumí *pref v RA*) host does next-hop determination and consults its Default Router List, it primarily prefers reachable routers over non-reachable routers and secondarily uses the router preference values.
- Type C (rozumí *Route Info*) hosts use a Routing Table instead of a Default Router List. Entries in the Routing Table have a prefix, prefix length, preference value, lifetime, and next-hop router. Type C hosts use both the Default Router Preference value in the Router Advertisement header and Route Information Options.

Konfigurace preference routeru

- Rtadvd (rtadvd.conf):
  `eth0:raflags#136: (M flag=128+High=8)`
- HP E-Series (ProCurve) – neumí
- HP A-Series (H3C/3COM ComWare) – neumí
- Dell 8024F – neumí
- Extreme Networks (XOS 12.4) – neumí
- Cisco
  `ipv6 nd router-preference high/medium/low`
- Další?

ACL na L2 prvcích

Je třeba filtrovat neplatné RA a DHCPv6 Reply:

a) podle MAC adresy routeru nebo
b) podle link-local adresy routeru (relay) nebo
c) podle portu (nejpraktičtější)

- Source MAC (a), Source IPv6 (b), Any (c)
- Destination IPv6 (FF02::1, FE80::/64)
- Next Header = 58 (ICMPv6) nebo 17 (UDP)
- ICMPv6 Type = 134, ICMPv6 Code = 0
- UDP source port = 547, dest port = 546

HP E-series K.14.XX

```
ipv6 access-list block-ra-dhcp
  10 deny icmp any any 134 0
  20 deny udp any eq 547 fe80::/64 eq 546
  30 permit ipv6 any any
exit
interface 1-44
  ipv6 access-group block-ra-dhcp in
exit
show statistics aclv6 block-ra-dhcp port 20
```

HP A-Series (H3C)

```
acl ipv6 number 3000 name block-ra-dhcp
     rule 10 deny icmpv6 icmpv6-type router-advertisement
     rule 20 deny udp destination fe80::/64 destination-port
     eq 546 source-port eq 547
quit
traffic behavior b_brd
     filter deny
quit
traffic classifier c_brd
     if-match acl ipv6 3000
quit
qos policy p_brd
     classifier c_brd behavior b_brd
quit
interface GigabitEthernet 1/0/20
     qos apply policy p_brd inbound
quit
```

RA snooping

- L2 prvky standardně podporují DHCP snooping (blokování podvržených DHCP odpovědí)
- RA snooping bude logicky taky časem doplněn
- IPv6 RA guard – draft RFC (verze 06 z 17.6.2010)
  - Bezestavové – jako ACL/RA snooping s definicí:
    - povolené/zakázané LLA adresy RA
    - povolené/zakázané porty přijímající RA
      - Cisco:
```
inteface fastethernet 0/0
        ipv6 nd raguard
```
    - povolené/zakázané prefixy (tohle obvykle v HW ACL nejde)
  - Stavové:
    - Autodetekce – detekuje routery, pak přejde do filtrování
    - SeND based – validuje RA a pouští jen validované (problém patentu Cisco 20080307516)

Monitorování a obrana

- **Ndpmon** - http://ndpmon.sourceforge.net/
  - Monitoruje ICMPv6 (*BSD, Linux)
  - Konfigurace v XML
  - Detekuje:
    - Nové uzly (IPv6 adresy)
    - Změny adres, DAD
    - Podvržení adresy (NS/NA)
    - Podvržený router (MAC, IP, prefix)
    - Router redirect
  - Problémy:
    - Hlášení nejsou konfigurovatelné
    - Práce s pamětí (scházející free)
    - Konfigurace se občas přepíše
    - Jen jeden interface

Rafixd

- **Rafixd** – KAME projekt
  - http://www.kame.net/dev/cvsweb2.cgi/kame/kame/kame/rafixd/
  - http://github.com/strattg/rafixd (upravený pro Linux)
  - Jednoúčelový, pouze monitoruje (v debug) a odinzeruje zadaný prefix (lze na více if):

```
rafixd –f –D –p 2002::/16 eth0
recv_ra: Jun/07/2010 16:37:39 received a packet from
   fe80::a6ba:dbff:fe6d:9a11%em0 to ff02::1 on em0
recv_ra: Jun/07/2010 16:37:39 RA prefix:
   2001:718:802:808::/64
recv_ra: Jun/07/2010 18:53:26 RA prefix:
   2002:93e5:8fa:1::/64
recv_ra: Jun/07/2010 18:53:26 received a bogus prefix
   2002:93e5:8fa:1::/64 from fe80::230:48ff:fed6:7d0a%em0
add_router: Jun/07/2010 18:53:26 added a bogus router
   fe80::230:48ff:fed6:7d0a on em0 expiring in 1344msec
purge_router: Jun/07/2010 18:53:28 sent a purge packet on
   em0, len = 70
```

Ramond

- **Ramond**-0.4 – založen na rafixd
  - Monitoruje jen RA
  - Umí více interface
  - Na neplatné routery reaguje RA s nulovou platností (clear)
  - Vyžaduje apr-1 a libxml2
  - Konfigurace v XML, podobná ndpmon

Funguje skutečně?

- NOTE: while this utility appears to work as described (sending a RA with lifetime=0 for the router), it does not appear to make a real effect on Windows XP or Linux stations...
- `./ra6 eth0 fe80::230:48ff:fed6:7d0a 2002:93e5:08fa:1::/64 1500`
- Linux: netstat –A inet6 –r (BSD netstat –rn6)
  `*/0 fe80::230:48ff:fed6:7d0a UGDA 1024 eth0`
  - vzápětí zmizela, **funguje OK**
- ipconfig/all (Win XP, Vista, Win7)
  `Default Gateway...fe80::230:48ff:fed6:7d0a%10`
  - taky zmizela, **funguje OK**
- ALE: zůstala nakonfigurovaná SLAAC adresa (lifetime infinite)!

Monitorování a obrana

- Proč tak složitě?

```
tcpdump -n -i eth0 dst host ff02::1 |
  grep "router advertisement" | grep –v
  LL-adresa-routeru
```

Scapy (Python + IP protokoly)

- Generuje, dekóduje cokoli, lze použít interaktivně:

```
>>> p=IPv6(dst="ns.cesnet.cz")/ UDP()/ DNS(rd=1,
   qd=DNSQR(qname="www.cesnet.cz"))
>>> p
<IPv6 nh=UDP dst=<Net6 ns.cesnet.cz> |<UDP
   sport=domain |<DNS  rd=1 qd=<DNSQR
   qname='www.cesnet.cz' |> |>>>
>>> p.show()
...
>>> sr1(p)                                  odeslat/přijmout odpověď
<IPv6  version=6L tc=0L fl=0L plen=253 nh=UDP hlim=61
   src=2001:718:1:1::2 dst=2001:718:802:8b0::93e5:b013
   |<UDP  sport=domain dport=domain len=253
   chksum=0x50fc |<DNS  id=0 qr=1L opcode=QUERY aa=1L
   tc=0L rd=1L ra=1L z=0L rcode=ok qdcount=1 ancount=1
   nscount=3 arcount=6 qd=<DNSQR  qname='www.cesnet.cz.'
   qtype=A qclass=IN |> an=<DNSRR
   rrname='www.cesnet.cz.' type=A rclass=IN ttl=86400
   rdata='195.113.144.230' |> ...
```

Vygenerování RA

```
>>> fam,mac = get_if_raw_hwaddr(conf.iface)
>>> p=Ether()/IPv6()/ICMPv6ND_RA()/
  ICMPv6NDOptPrefixInfo(prefix="2002:93e5:c010:1::",
  prefixlen=64)/ICMPv6NDOptSrcLLAddr(lladdr=mac)
>>> p
<Ether  type=0x86dd |<IPv6 nh=ICMPv6 hlim=255
  dst=ff02::1 |<ICMPv6ND_RA  |<ICMPv6NDOptPrefixInfo
  prefixlen=64 prefix=2002:93e5:c010:1:: [6to4 GW:
  147.229.192.16] |<ICMPv6NDOptSrcLLAddr
  lladdr=00:30:48:d6:ad:54 |>>>>>
>>> sendp(p, loop=1, inter=60)
```

Odinzerování routeru:

```
>>> send(IPv6(src=router_lla)/
  ICMPv6ND_RA(routerlifetime=0), loop=0, inter=0)
```

Poor man rafixd

```
#!/usr/local/bin/python
from scapy.all import *
def ra_monitor_callback(pkt):
   if ICMPv6ND_RA in pkt and ICMPv6NDOptPrefixInfo in pkt
    and pkt[ICMPv6NDOptPrefixInfo].prefix[0:4] == "2002":
    send(IPv6(src=pkt[IPv6].src)/
           ICMPv6ND_RA(routerlifetime=0) )
   u = pkt.sprintf(" rogue %Ether.src% %IPv6.src% >
                  %IPv6.dst%
%ICMPv6ND_RA.routerlifetime%")
   return time.asctime() + u

sniff(prn=ra_monitor_callback,
   filter="dst host ff02::1", store=0, iface="eth0")
```

Doplnění default router do DHCPv6

- Default Router and Prefix Advertisement Options for DHCPv6
  - Draft RFC z 2.3.2009, expirovaný
- Nezapadá do automatické konfigurace, nelze očekávat akceptaci (bylo vypuštěno záměrně)
- Diskuse RA x DHCPv6, we will see iff
  - RFC 5006 gets widely accepted (or not).
  - DHCPv6 supports default route + prefix lengths (or not).
  - SLAAC gets widely accepted (or not).
  - Mac OS X supports DHCPv6 (or not).
  - RA Guard is widely implemented.
  - DHCPv6 Snooping is widely implemented.
  - SeND gets widely accepted (or not).