

Strach

Z

IPv6?

Radek Zajíc  
LinuxDays 2014

# Něco málo o mně

Radek Zajíc, @zajdee

## IPv6 evangelista

Šírím povědomí o IPv6

Vyvracím mýty a vysvětluji rozdíly

Jako aktivní blbec nasazují IPv6 už od roku 2008 a ukazují, že „to jde“ :-)

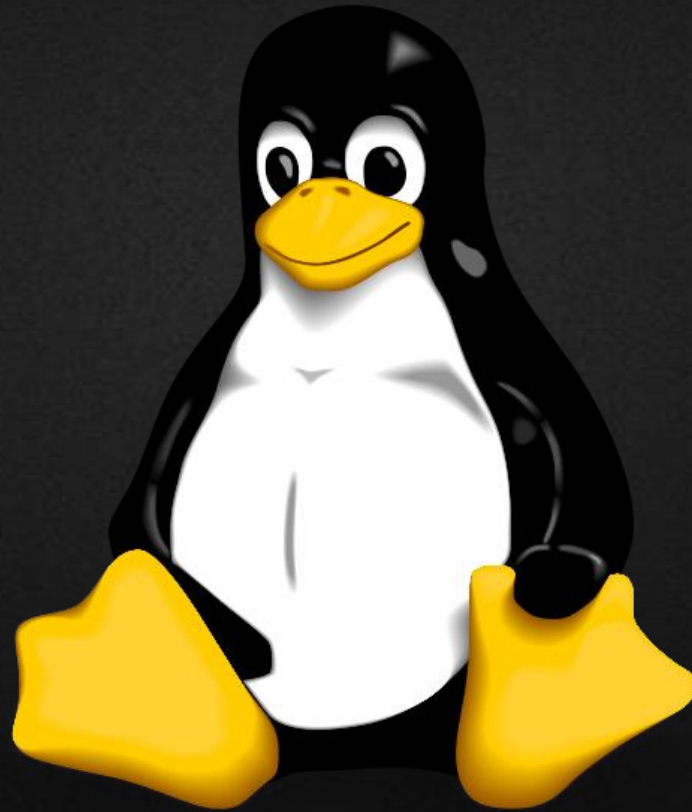
## Nejvýznamnější IPv6 aktivity:

Seznam.cz – obsah (2008 – 12)

T-Mobile.cz – konektivita (2013 – 14)

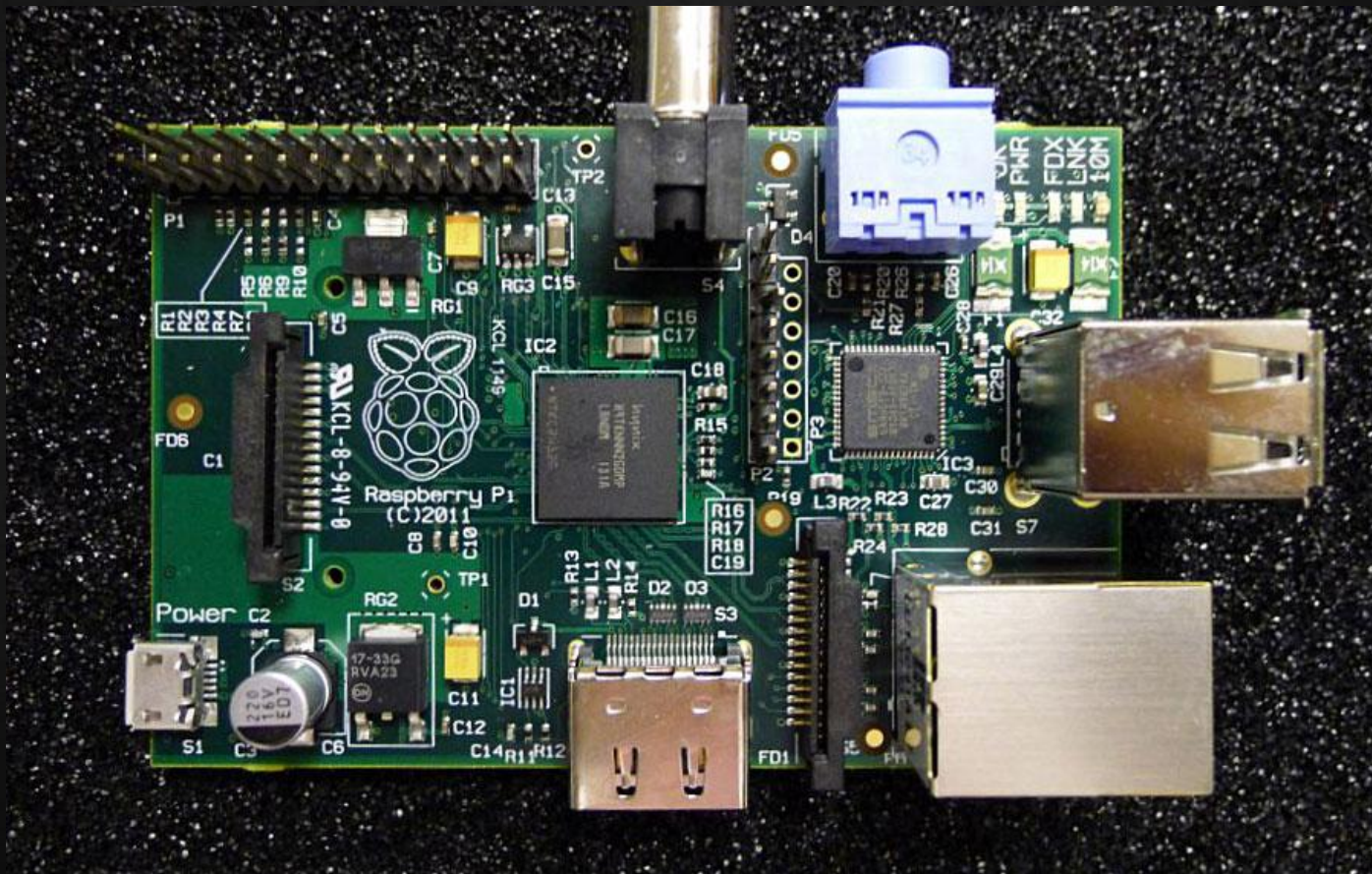
# Malá anketa

Kolik z nás používá Linux?



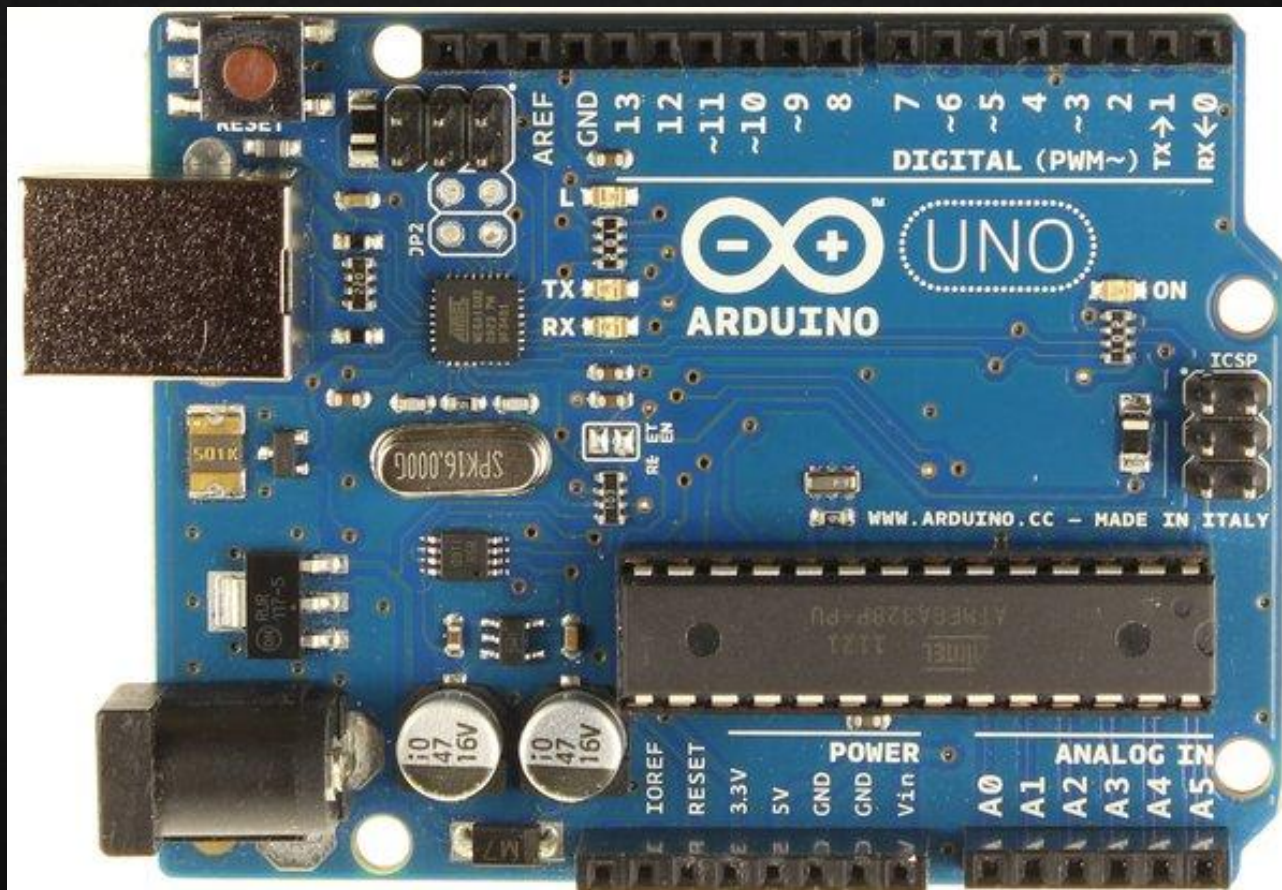
# Malá anketa

Kolik z nás používá Raspberry Pi?



# Malá anketa

Kolik z nás používá Arduino?



# Malá anketa

Kolik z nás používá OpenWRT?



# Malá anketa

A kolik z nás používá IPv6?



# Co je IPv6?

IPv6 (internetový protokol verze 6) je označení nastupujícího protokolu pro komunikaci v současném Internetu. Nahrazuje dosluhující protokol IPv4.

(Heslo [IPv6](#) na české Wikipedii)



# Proč „Strach z IPv6“?

IPv6 se rozšiřuje velmi pomalu.

Problém slepice nebo vejce?

Velcí hráči už IPv6 často  
podporují

I přesto se objevuje plno hlasů,  
které IPv6 zavrhnou, odmítají.

Je za tím strach z nového,  
neznámého?

# Proč IPv6? – Vyčerpání IPv4

## Depletion Dates

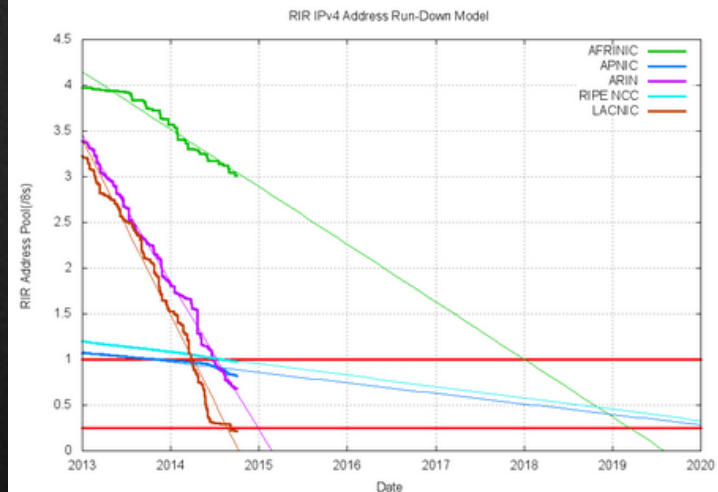
- Assigned Class "B" network numbers Mar. 11, 1994
- NIC "connected" Class B network numbers Apr. 26, 1996
- NSFnet address space\* Oct. 19, 1997
- Assigned Class "A-B" network numbers Feb 17, 1998
- NIC "connected" Class A-B network numbers Mar. 27, 2000
- BBN snapshots\* May 4, 2002

\* all types: may be earlier if network class address consumption is not equal.

IANA Unallocated Address Pool Exhaustion:  
03-Feb-2011

Projected RIR Address Pool Exhaustion Dates:

RIR	Projected Exhaustion Date	Remaining Addresses in RIR Pool (/8s)
APNIC:	19-Apr-2011 (actual)	0.8226
RIPE NCC:	14-Sep-2012 (actual)	0.9758
LACNIC:	10-Jun-2014 (actual)	0.2184
ARIN:	15-Mar-2015	0.6809
AFRINIC:	06-Jun-2019	3.0116



Projection of consumption of Remaining RIR Address Pools

Tenkrát v devadesátém...

...jsme věděli, že to přijde.

# IPv4 na kapačkách

- 1993 – RFC 1517–8 – zavedení beztrždního adresování
  - 1996 – RFC 1918 – privátní IP sítě 10.0.0.0/8, 192.168.0.0/16, 172.16.0.0/12
  - 2001 – RFC 3022 – NAT/NAPT – překlad adres a portů
  - 2011 – RFC 6264 – „operátorský“ NA(P)T
  - 2012 – RFC 6598 – „Sdílené“ adresy 100.64.0.0/10 pro operátorský NA(P)T
  - 2011 – 2014 – trh s adresami (~\$ 12 za adresu)
- A další způsoby, jak „ušetřit“ adresy – DHCP, vícenásobný NAT, rozmach privátních sítí uvnitř organizací



Je IPv6 řešením?

Ano...

Ale jen  
částečně. Zatím?

# World IPv6 Day & Launch



2011

IPv6 na webserverech  
na jeden den



2012

Na webserverech  
napořád  
Zapojeni i výrobci  
domácích routerů a  
poskytovatelé připojení

# IPv6 obsah ve světě a u nás

Google



NETFLIX  
NETFLIX



IHNED.cz

SEZNAM.CZ

centrum.cz



Justice.cz

LUPA<sup>CZ</sup>

Prima

cz.nic

MATY.CZ

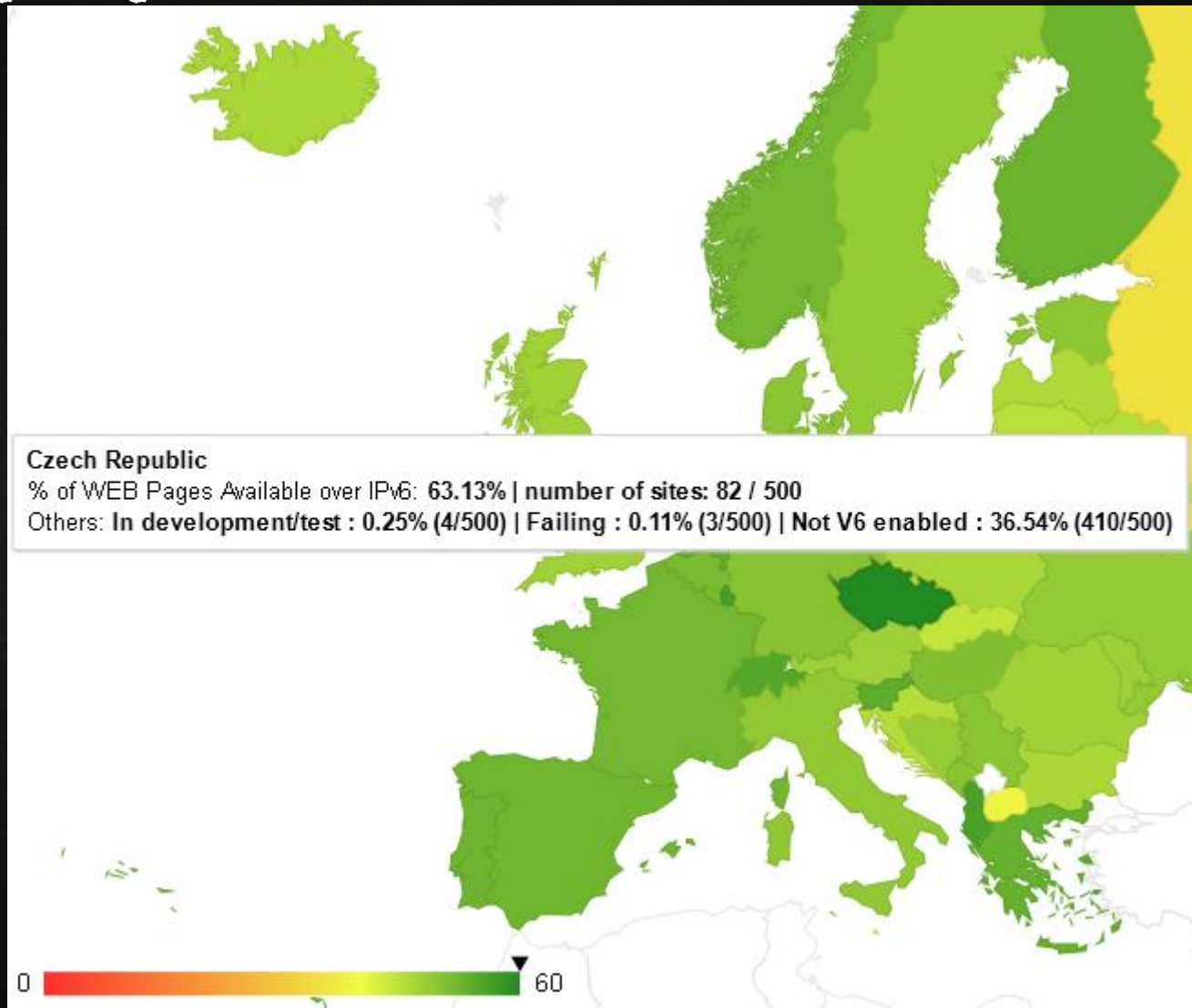
ROOT.CZ

Sport.cz

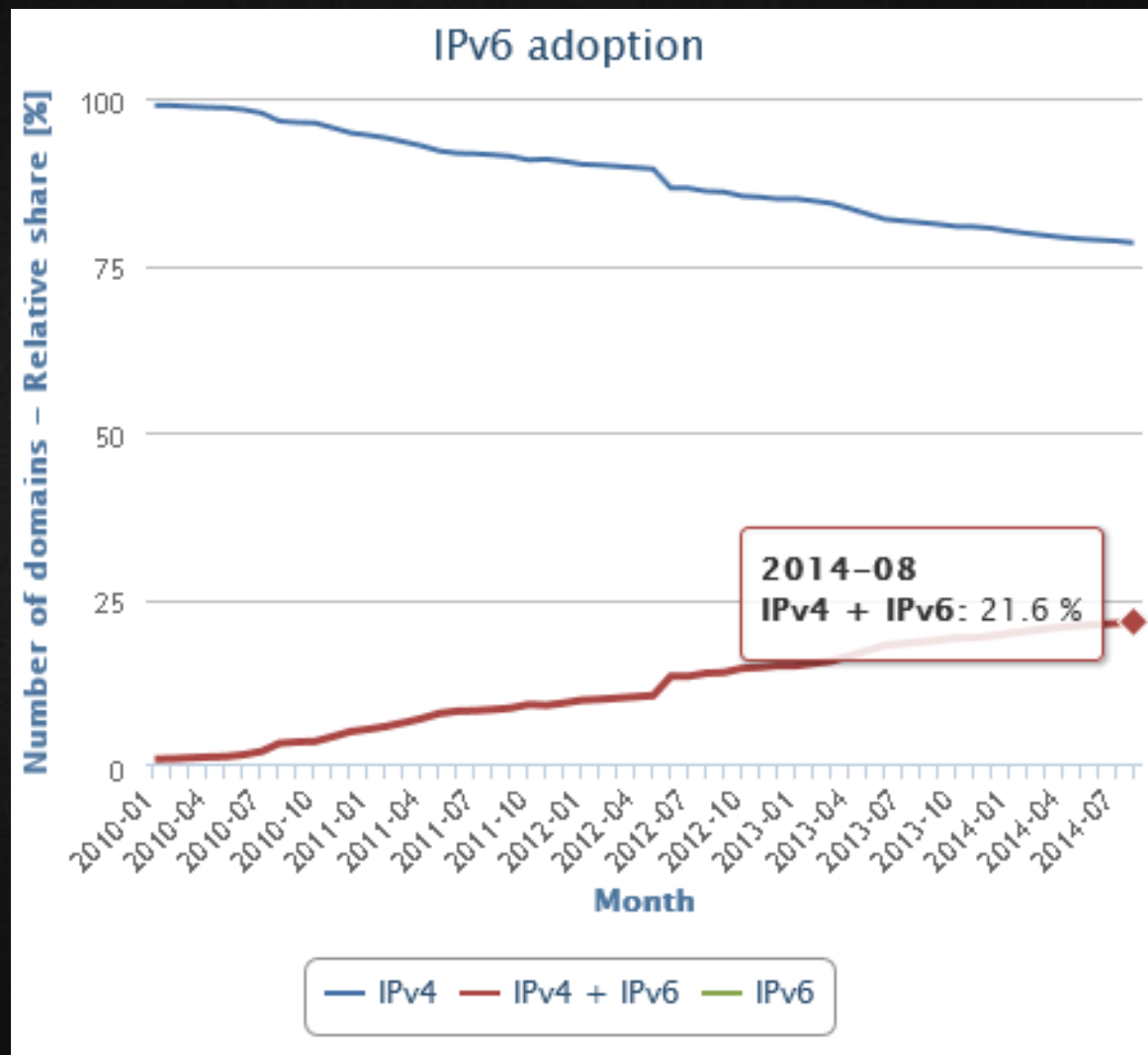


Hosting90  
systems s.r.o.

# ČR je jedničkou v IPv6 obsahu...



...ale jen 21 procent .cz webů  
podporuje IPv6

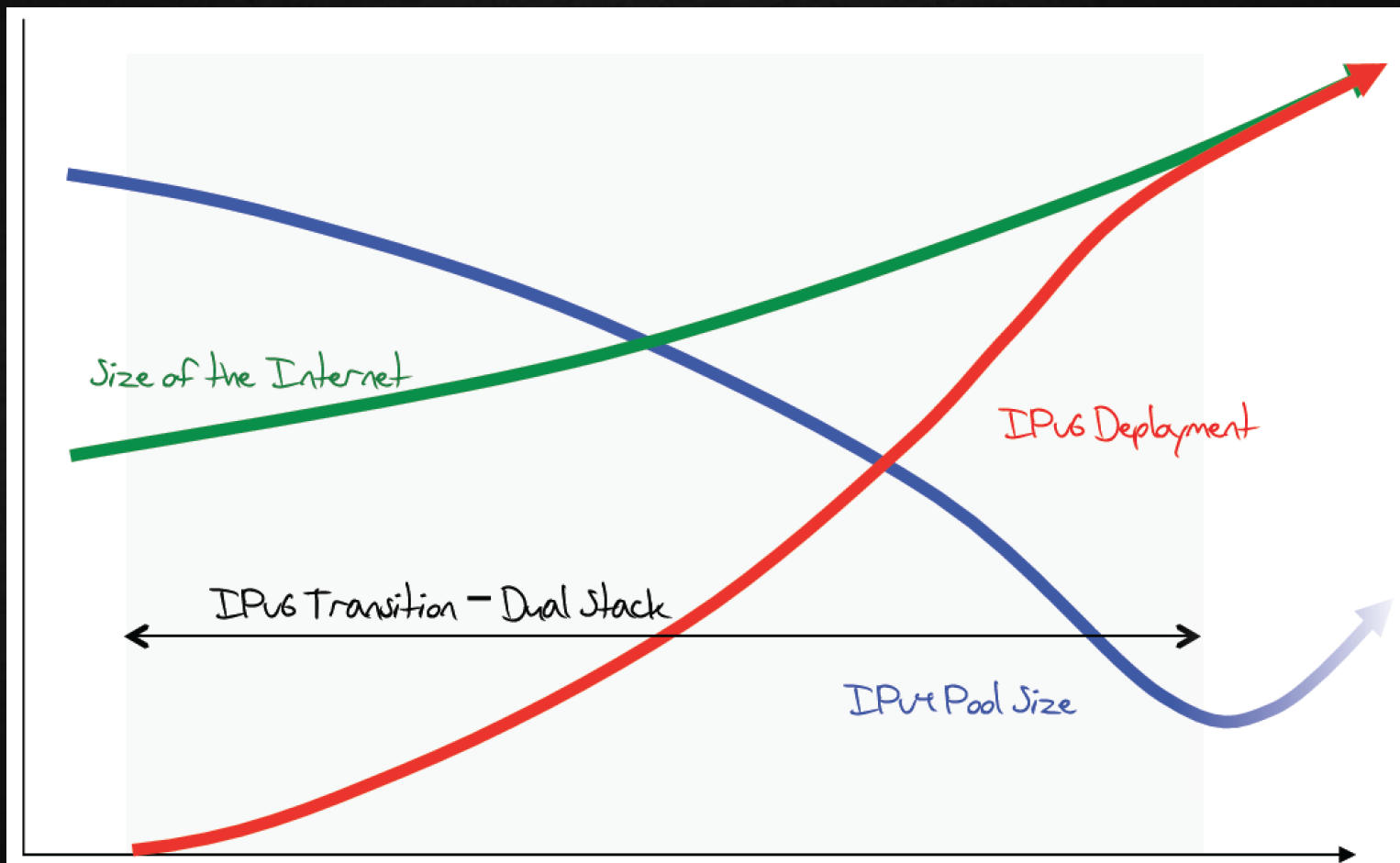




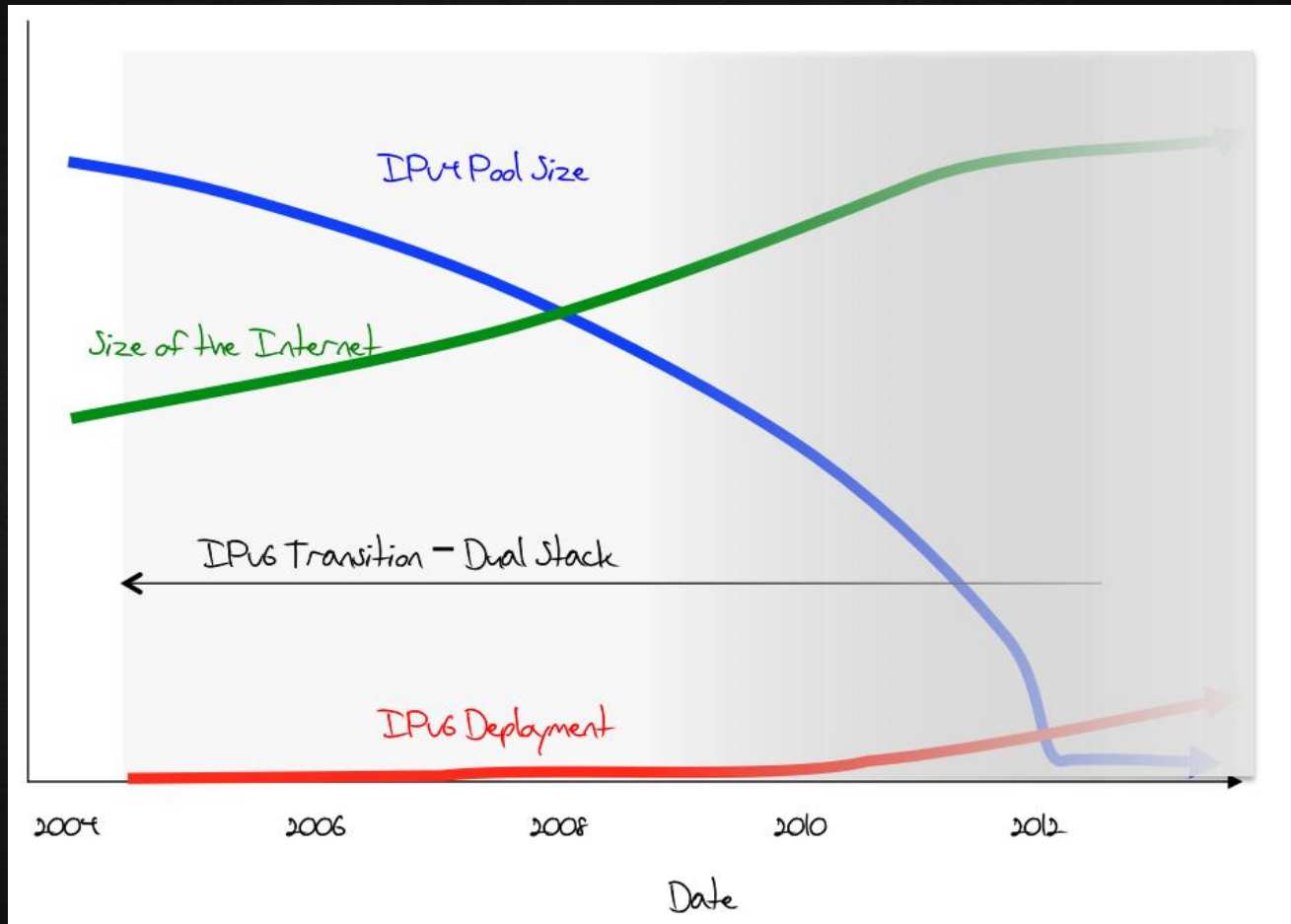
# IPv6 připojení ve světě a u nás



# IPv6 očekávání pre-2000



# IPv6 očekávání 2012



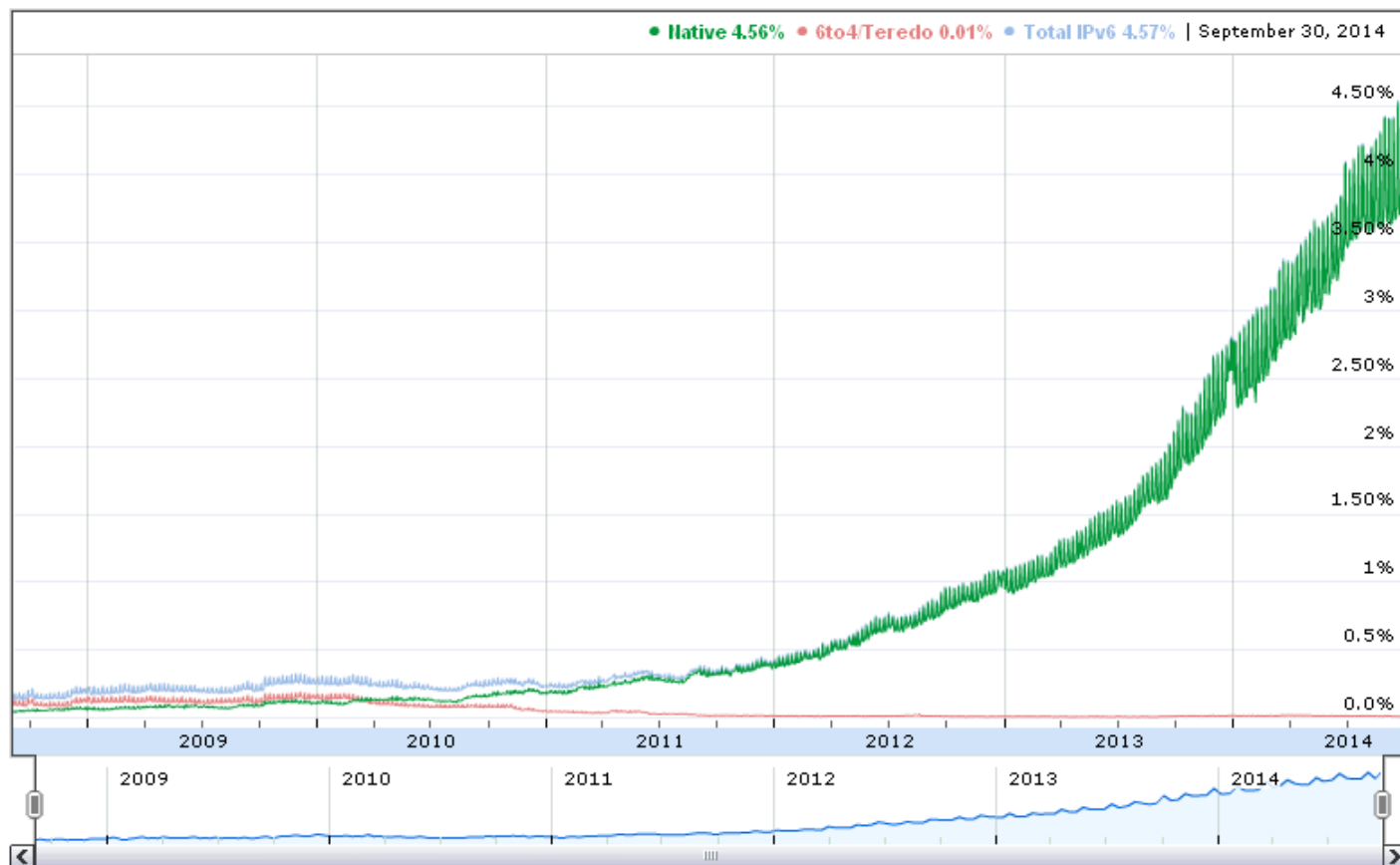
# Jak to vidí Google

## IPv6 Adoption


Per-Country IPv6 adoption

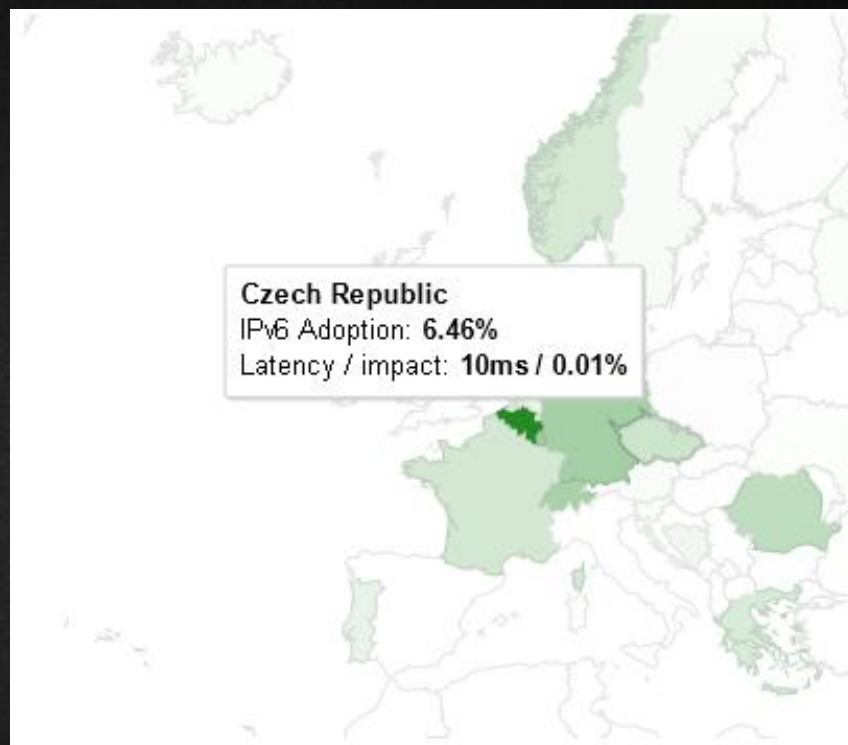
### IPv6 Adoption

We are continuously measuring the availability of IPv6 connectivity among Google users. The graph shows the percentage of users that access Google over IPv6.



# Jak to vidí Google

Rank	Country	%-age
1	 <a href="#">Belgium</a>	28.65 %
2	 <a href="#">Luxembourg</a>	11.75 %
3	 <a href="#">Germany</a>	11.64 %
4	 <a href="#">Switzerland</a>	11.06 %
5	 <a href="#">United States of America</a>	9.56 %
6	 <a href="#">Peru</a>	8.98 %
7	 <a href="#">Romania</a>	8.36 %
8	 <a href="#">Czech Republic</a>	6.46 %
9	 <a href="#">France</a>	5.61 %
10	 <a href="#">Japan</a>	5.55 %



<http://google.com/ipv6/>

<https://www.vyncke.org/ipv6status/>

# Pojďte vyzkoušet IPv6!

Chtějte od svého poskytovatele připojení IPv6! Rozumný přiděl je o velikosti /60 až /48 – to je 16 až 65536 koncových sítí.

Pokud poskytovatel IPv6 nepodporuje, zkuste tunelované IPv6 (ale zapomeňte na Teredo a 6to4)

Zajímaví poskytovatelé jsou např. [tunnelbroker.net](http://tunnelbroker.net) nebo [SixXS.net](http://SixXS.net)



# IPv4 vs. IPv6

IPv4	IPv6
32 bitů	128 bitů
192.0.2.1	2001:db8::1
Používá broadcast	Broadcast nepoužívá
Autokonfigurace pouze pomocí DHCPv4	Autokonfigurace více způsoby (PITA)
Objevování sousedů jako samostatný protokol (ARP)	Objevování sousedů součástí ICMPv6
IP Hlavička obsahuje kontrolní součet a nepovinná pole	Nepovinná data vysunuta do dodatečných hlaviček, kontrolní součet si řeší vyšší vrstvy
Fragmentovat může odesílatel i router na cestě	Fragmentuje jen odesílatel

# Zápis IPv6 adres a sítí

Osm segmentů oddělených dvojtečkou, každý segment má 16 bitů

Segmenty zapisujeme hexadecimálně

Dlouhý zápis adresy

2001:0db8:0000:0002:0000:0000:0000:000a  
2001:0DB8:0000:0002:0000:0000:0000:000A

Vypouštíme „leading zeros“

2001:db8:0:2:0:0:0:a

Sekvenci dvou a více nulových segmentů nahradíme „:“

2001:db8:0:2::a

Síťovou masku píšeme jako počet bitů adresy sítě za lomítko (CIDR zápis)

2001:db8:0:2::a/64 (Adresa stroje v síti)  
2001:db8:0:2::/64 (Síť jako celek)



# Základní zásady IPv6

Obvykle  $2^{16}$  (/48) až  $2^{32}$  (/32) koncových sítí  
pro každou organizaci/ISP

Autokonfigurace vyžaduje  $2^{64}$  adres (maska /64)  
pro každou koncovou síť

Autokonfigurace způsobuje boolehav

Výchozí brána zjištěna z oznámení  
směrovače

Adresy vytvořeny automaticky dle oznámení  
směrovače

nebo přiděleny z DHCPv6

DNS přiděleno v oznámení směrovače nebo  
pomocí DHCPv6

# Základní zásady IPv6

Hierarchické adresování a  
směrování

Organizace dostane /48

+– Každá pobočka /52

+– Budova v pobočce /56

+– Koncová síť /64

Nepoužívá se NAT!\*

# Základní zásady IPv6

Samotná IPv6 zatím nestačí, musí tedy být doplněna IPv4

Provozu obou protokolů zároveň říkáme dual-stack - je to dnes nejčastější režim používání IPv6 na koncových stanicích

IPv4 lze ale k uživateli dopravit i jako službu! (464XLAT, DS-Lite)

# Co je třeba pro IPv6 obecně

OpenWRT 14.07<sup>1</sup> – splňuje RFC 7084  
(detaily zítra mezi 11.30 – 12.30)

Běžná Linuxová Distribuce vyžaduje  
ruční konfiguraci. Měla by být  
zároveň dostatečně nová (letošní,  
loňská)

Podpora IPv6 je součástí kernelu i  
běžných nástrojů – ip6tables, ppp,  
radvd, wide-dhcpv6, freeradius, ip,  
ping6, traceroute, ...

# Co je třeba pro připojení

## WAN rozhraní:

- IPv6 adresa a výchozí brána
- maska sítě (obvykle /64)
- V případě 6in4 tunelů: IPv4 adresa tunnelbrokera

## Pro vaše LAN rozhraní:

- Delegovaný IPv6 prefix (/64 až /48)
- Přidělí provider
- Nezapomeňte: Nepoužívá se NAT

# Podpora v kernelu

Podpora v kernelu – dnes běžná

IPv6 forwarding

```
net.ipv6.conf.all.forwarding=1
```

IPv6 je aktivní „by default“

# Firewall a IPv6

ip6tables součástí distribucí, fungují jako  
u IPv4

Firewall patří na koncový uzel

Nezahazujte na hranici sítě spojení, která  
přicházejí zvenku

Proč? V IPv6 není UPnP ani prorážení  
NATu!

Výjimka: rizikové protokoly

ICMPv6 klíčové pro fungování protokolu –  
filtrujte opatrně, vizte RFC4890

# Autokonfigurace - RA

Router advertisement (oznámění směrovače) - autokonfigurace  
koncových uzlů, distribuuje informace o síti

Příznaky v RA zprávě - per link

Managed Flag on/off - z DHCPv6 si ber úplně vše

Other Flag on/off - z DHCPv6 si ber pouze DNS/PD/atp.

Příznaky v RA zprávě - per prefix

OnLink Flag on/off - je-li off, všechnen provoz z uzlu jde skrz  
router

Autonomous Flag on/off - je-li on, kernel vytvoří na základě  
tohoto prefixu adresy na interface

Přijímání router-advertisementů:

```
net.ipv6.conf.$IFACE.accept_ra=$X
```

\$X==0 vždy vypnuto (výchozí bránu nutno nastavit ručně)

\$X==1 zapnuto, pokud nedělám forwarding; jinak vypnuto

\$X==2 zapnuto, i pokud dělám forwarding



# Autokonfigurace – RA

Jak vypadá autokonfigurovaná adresa?

Horních 64 bitů získáno z oznámení směrovače

Je-li vytvořena z MAC adresy (EUI-64)

12:12:34:34:56:56 → 1012:34ff:fe34:5656

2001:db8:0:1:1012:34ff:fe34:5656/64

Jsou-li adresy generovány podle zásad

„privacy extensions“ (zachování

soukromí), je 64 bitů opravdu náhodných

2001:db8:0:1:e115:cda6:108b:f7f1/64

# Autokonfigurace - DHCPv6

DHCPv6 - zapomeňte na isc-dhcp,  
zajímavější jsou wide-dhcpv6 a  
dibbler

Rozhodnutí KDY použít DHCPv6 pro jaký  
účel je závislé na příznacích z RA :-(

Jeden DUID vládne všem rozhraním

# Delegování prefixů

Pokud váš ISP podporuje delegování prefixů pomocí DHCPv6, opět vám pomůže `wide-dhcpv6` nebo `dibbler`  
V OpenWRT 14.09 podporováno nativně

# Delegování prefixů

```
# Příklad pro wide-dhcpv6 a ppp
interface ppp0 {
    send ia-pd 1;
};

id-assoc pd 1 {
    prefix-interface eth1 {
        # Přiděl subnet ID 1 síťovce eth1
        sla-id 1;

        # Suffix adresy
        ifid 1;

        # Počet bitů síťové části adresy, které vám ISP přiděluje
        # Odečtete velikost přiděleného prefixu od 64 a doplňte zde.
        sla-len 8;
    };
};

# Přidělený blok: 2001:db8:ffff:ab00::/56
# na eth1 bude nakonfigurovaná adresa 2001:db8:ffff:ab01::1/64
```

# Ruční konfigurace interfaces

/etc/network/interfaces:

```
iface eth1 inet6 static
```

```
    address 2001:db8:ffff:ab01::1
```

```
    netmask 64
```

# Příklad tunelu – SixXS, OpenWRT, Debian

/etc/config/network:

```
config interface 'wan6'  
    option proto '6in4'  
    option peeraddr '217.31.57.16'  
    option ip6addr '2a01:8c00:db00:db::2/64'
```

/etc/network/interfaces:

```
auto sixxs  
iface sixxs inet6 v4tunnel  
    address 2001:db8:100:100::2  
    netmask 64  
    endpoint 217.31.57.16  
    up ip -6 route add default via 2001:db8:100:100::1
```

# ip -6

Zapomeňte na ifconfig & route,  
použijte ip (ip -6 address & ip -6  
route)

```
# ip -6 addr show dev br0.2
```

```
7: br0.2: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500
```

```
    inet6 2001:db8:1234::1/64 scope global
```

```
        valid_lft forever preferred_lft forever
```

```
    inet6 fe80::922b:34ff:fed1:ed8/64 scope link
```

```
        valid_lft forever preferred_lft forever
```

```
# ip -6 route
```

```
2001:db8:1234::/64 dev br0.2 proto kernel metric 256
```

```
fe80::/64 dev br0.2 proto kernel metric 256
```

```
default via fe80::da58:d7ff:fe00:a05 dev br0.3 src 2001:db8:1234::1  
metric 1
```

# A je to.

```
# traceroute -6 -I -n www.seznam.cz
```

```
traceroute to www.seznam.cz (2a02:598:1::3), 30 hops max, 80 byte packets
```

```
(...)
```

```
6 2001:4de8:b0ba:29:134::1 11.133 ms 10.807 ms 10.831 ms
7 2001:4de8:d1a1:1111:21::2 10.082 ms 13.514 ms 13.517 ms
8 2001:4de8:b0ba:deac::2 13.766 ms 12.793 ms 12.782 ms
9 2a02:598:6262:9::2 13.059 ms 12.701 ms 12.391 ms
10 2a02:598:1::3 12.056 ms 12.140 ms 11.796 ms
```

```
# telnet -6 www.seznam.cz 80
```

```
Trying 2a02:598:1::3...
```

```
Connected to www.seznam.cz.
```

```
# ping6 -c 1 -n www.seznam.cz
```

```
PING www.seznam.cz(2a02:598:1::3) 56 data bytes
```

```
64 bytes from 2a02:598:1::3: icmp_seq=1 ttl=57 time=9.99 ms
```

```
--- www.seznam.cz ping statistics ---
```

```
1 packets transmitted, 1 received, 0% packet loss, time 0ms
```

```
rtt min/avg/max/mdev = 9.998/9.998/9.998/0.000 ms
```



# Konektivita funguje

Ale co když fungovat přestane?

„Pomůže“ technika zvaná Happy Eyeballs  
(Podpora v běžných brouserrech, jinde zatím ne.)

Klient se pokusí připojit přes IPv4 i IPv6  
paralelně

První navázané spojení použije  
(v průběhu času může střídat protokoly)  
různé implementace v různých OS/aplikacích

# Zpřístupníme konektivitu

```
Daemon radvd
```

```
/etc/radvd.conf:
```

```
interface eth1
```

```
{
```

```
    AdvSendAdvert on; # zasílej oznámení
```

```
    AdvManagedFlag off; # ne DHCPv6 adresy
```

```
    AdvOtherConfigFlag off; # ne DHCPv6 ostatní
```

```
    prefix 2001:db8:101:101::/64
```

```
    {
```

```
        DeprecatePrefix on; # zruš při vypnutí
```

```
        AdvOnLink on; # generuj adresy
```

```
        AdvAutonomous on; # autonomní prefix
```

```
    }; # konec definice prefixu
```

```
}; # konec definice rozhraní
```

# Zpřístupníme služby

net.ipv6.bindv6only=1 (doporučené Debianisty)

IPv6 socket akceptuje pouze IPv6 spojení

V konfiguraci aplikace nutný Listen pro oba protokoly:

*Listen 0.0.0.0*

*Listen ::*

Výchozí ve Squeeze, Wheezy

net.ipv6.bindv6only=0 (obvyklé ve většině distro)

IPv6 socket akceptuje i IPv4 spojení

Zdrojová IPv6 adresa ve tvaru `::ffff:192.0.2.1`

V konfiguraci aplikace stačí Listen pro IPv6

*Listen ::*

Aplikace může toto chování předefinovat (umí např. nginx)

# Zpřístupníme služby

Zvyšte net.ipv6.route.max\_size

Default je 4096, což pro masivní IPv6 provoz nestačí. U IPv4 je default v milionech až miliardách, nebo kernel škáluje Ipv4 max\_size automaticky.

Doporučení: zvyšte max\_size, aby obsloužila vaše uživatele. Stav zaplnění lze sledovat pomocí

```
ip -6 route show cache
```

(Díky Martinu Duspivovi za doplnění!)

# Co je tu špatně?

 **Ondřej Caletka**  
@Oskar456  **Following**

@zajdee BTW \$ telnet [zajic.v.pytli.cz](http://zajic.v.pytli.cz) 80  
Trying 2001:4de8:deaf::b:1... a nic :(

 Reply  Retweet  Favorite  More

2:44 PM - 2 Oct 2014

217.69.133.249

2a00:1028:96ca:9

37.58.100.153

37.58.100.162

# Zpřístupníme služby

Vyzkoušejte, že služby (www, dns, mail) na IPv6 poslouchají a správně vyřizují požadavky

Ověřte, že aplikace fungují (!)

častý problém je ukládání IP adres klientů jako varchar(16), případně jako unsigned int(4) :-)

pokud vaše aplikace ověřuje sessions i pomocí IP adresy, bude mít v happy-eyeballs prostředí problém

Publikujte AAAA záznamy v DNS

Nezapomeňte na glue záznamy. A na monitoring!

# Shrnutí: postupujte po krocích

Není nutné vše rozchodit najednou

Začněte konektivitou na hranici sítě

Pokračujte konektivitou k jednotlivým serverům a  
stanicím

Zajistěte, aby služby podporovaly IPv6

Nezapomeňte, že IPv6 musí podporovat i aplikace

V první fázi stačí základní podpora na stanicích  
(připojení) a front-endech (obsah)

# Shrnutí: Servery & Routery

Autokonfigurace ani DHCPv6 se nepoužívá (accept\_ra=0) s výjimkou WAN rozhraní některých routerů (accept\_ra=2)

Všechny služby využívané mimo server by měly naslouchat (i) na IPv6 socketu

Služby, které se někam připojují, by měly využívat obou protokolů (kde to jde)



# Shrnutí: Domácí routery

Použijte OpenWRT. :-)

V jiných distribucích nakonfigurujte šestku ručně.

## WAN

zjistěte konfiguraci od svého ISP nebo tunnelbrokera

## LAN

aktivujte přidělování adres pomocí router advertisementů

vysílejte DNS přes bezstavové DHCPv6

volitelně zapněte přidělování adres pomocí stavového DHCPv6

Nepoužívejte pravidlo firewallu, které zabrání průchodu všeho provozu zvenčí

# Shrnutí: Koncové stroje

IPv6 v síti máte – naučte se s ní pracovat

Neblokujte IPv6

Povolte přijímání router advertisementů a  
aktivujte DHCPv6 klient  
tím zvýšíte šanci, že se připojíte ve všech  
sítích

(nebo použijte NetworkManager 0.9.6+)

Nezapomeňte na firewall

# Android 4+

Chybí podpora IPv6-only Wi-Fi

Chybí podpora DHCPv6 (celkově)

Naopak dobře funguje IPv6-only  
mobilní síť

Android jako první mobilní OS  
podporoval 4G-LTE, tedy IPv4 jako  
službu

# (Blízká) budoucnost?

IPv6-only datacentra



s branou do IPv4 světa

keywords: draft-anderson-v6ops-siit-dc-00

IPv6-only konektivita

do firem i domácností

keywords: NAT64/DNS64

IPv4 jako služba nad IPv6 sítí

keywords: 464XLAT, MAP-E/MAP-T, DS-Lite



# A ten strach?

IPv6 skutečně funguje. Nebojte se.

Přistupte k IPv6 ze začátku jako k Raspberry Pi.

Zkuste ji. Věřte jí. Překvapí vás.



Díky

radek@zajic.v.pytli.cz