



RIPE NCC

RIPE NETWORK COORDINATION CENTRE

IPv4-mapované IPv6 adresy

Ondřej Caletka | 6. června 2024 | Den IPv6 2024

Co je IPv4-mapovaná IPv6 adresa?



::ffff:192.0.2.1

- IPv6 adresa jako každá jiná
- Pevný prefix ::ffff:0:0/96 + IPv4 adresa
- Používá se pro kompatibilitu IPv6 socketů s **IPv4**



Socket API

Jak programovat pro IPv4 a
IPv6 zároveň

Klient

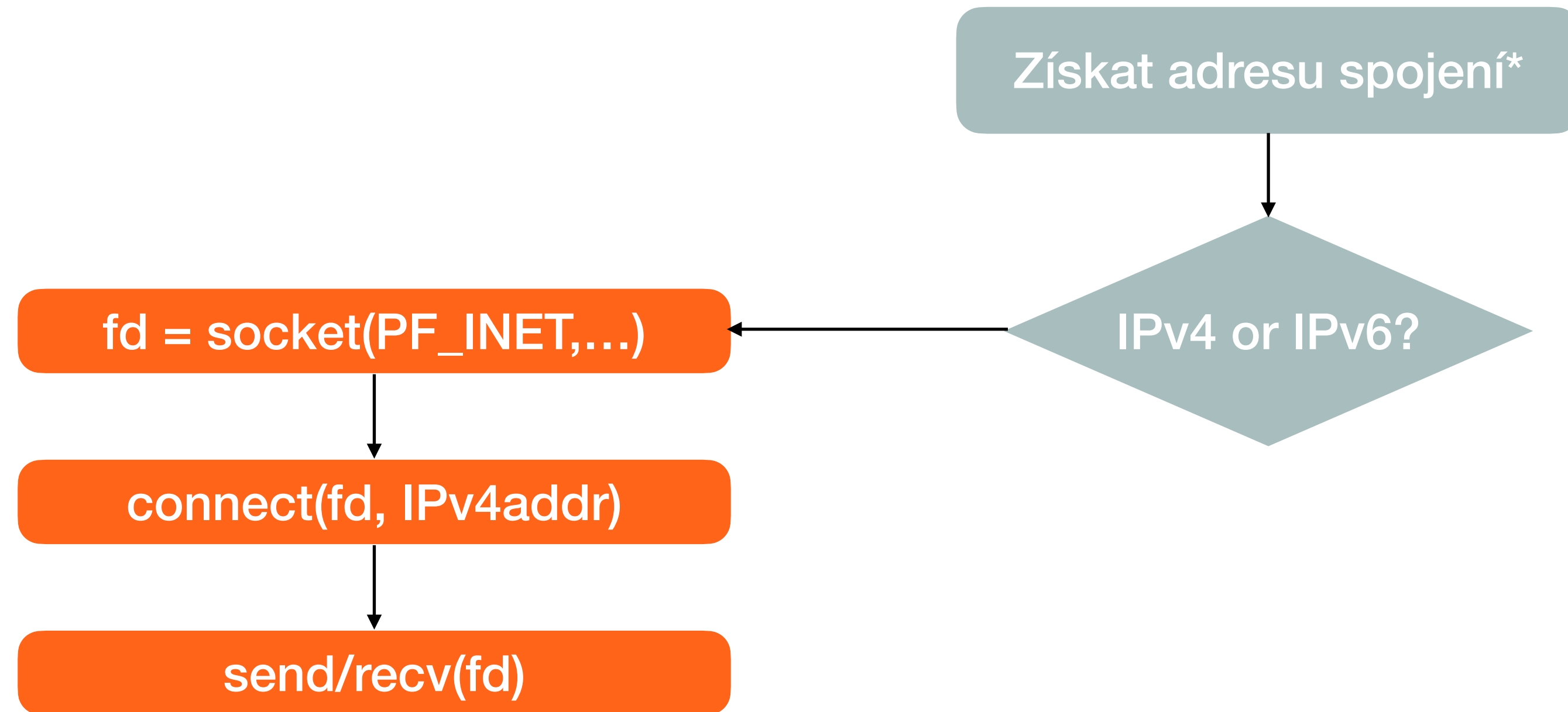


fd = socket(PF_INET,...)

connect(fd, IPv4addr)

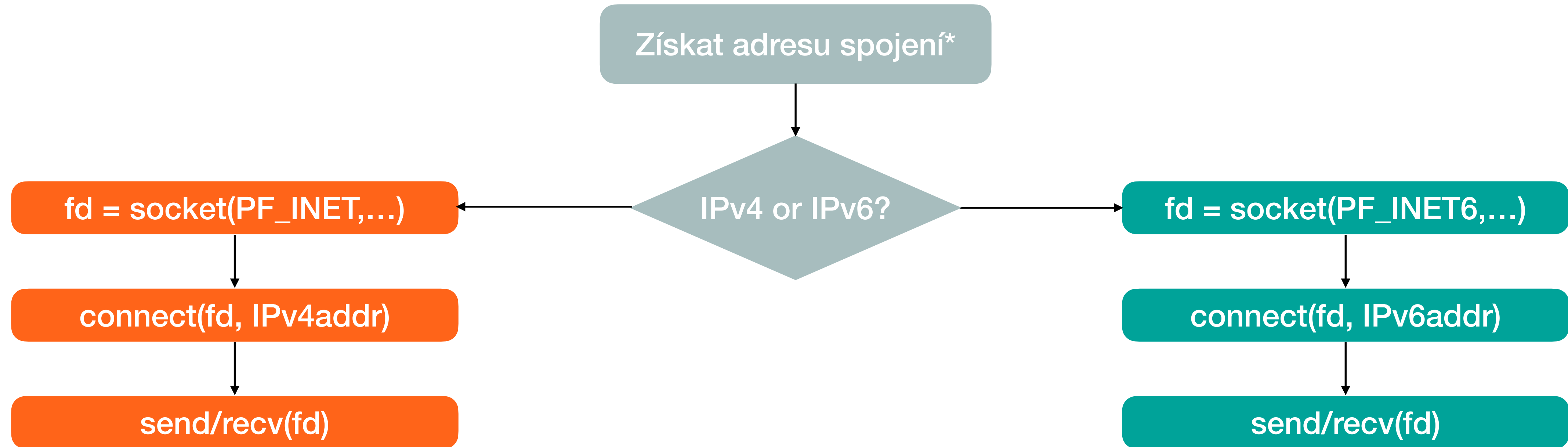
send/recv(fd)

Klient



***například pomocí `getaddrinfo(3)`**

Klient



*například pomocí `getaddrinfo(3)`

Server



```
fd4 = socket(PF_INET,...)
```

```
bind(fd4, IPv4addr)
```

```
listen(fd4, backlog)
```

```
cs = accept(fd4, *cIPv4add)
```

```
send/recv(cs)
```

Server



```
fd4 = socket(PF_INET,...)
```

```
bind(fd4, IPv4addr)
```

```
listen(fd4, backlog)
```

```
cs = accept(fd4, *cIPv4add)
```

```
send/recv(cs)
```

```
fd6 = socket(PF_INET6,...)
```

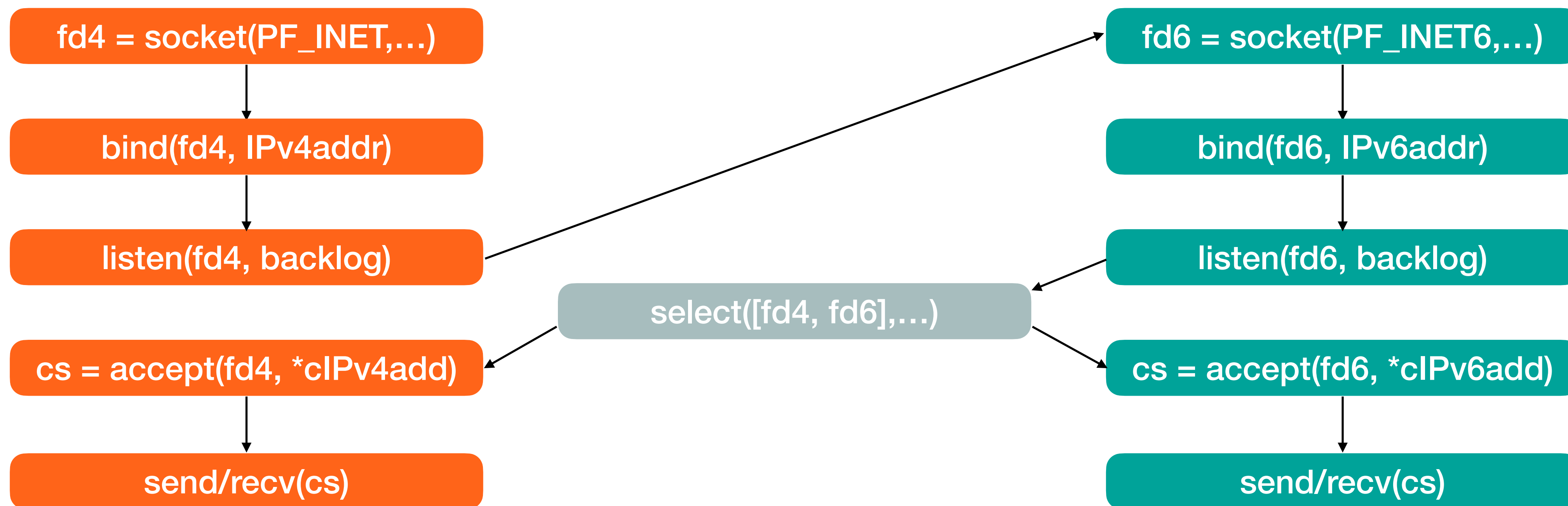
```
bind(fd6, IPv6addr)
```

```
listen(fd6, backlog)
```

```
cs = accept(fd6, *cIPv6add)
```

```
send/recv(cs)
```


Server



Kompatibilita IPv6 soketů s IPv4



- Sokety pro IPv6 mohou *někdy* pracovat i s IPv4
- Tím lze **zjednodušit přidání podpory IPv4** do starých aplikací
- Aplikace používá **pouze IPv6 sokety** = vidí **pouze IPv6 adresy**
- Provoz IPv4 je mapován pomocí **IPv4-mapovaných adres IPv6**
- Nedochází k žádnému překladu paketů

Kompatibilita IPv6 soketů s IPv4



- Sokety pro IPv6 mohou *někdy* pracovat i s IPv4
- Tím lze **zjednodušit přidání podpory IPv4** do starých aplikací
- Aplikace používá **pouze IPv6 sokety** = vidí **pouze IPv6 adresy**
- Provoz IPv4 je mapován pomocí **IPv4-mapovaných adres IPv6**
- Nedochází k žádnému překladu paketů
- Používání kompatibility s IPv4 se **nedoporučuje**:

Kompatibilita IPv6 soketů s IPv4



- Sokety pro IPv6 mohou *někdy* pracovat i s IPv4
- Tím lze **zjednodušit přidání podpory IPv4** do starých aplikací
- Aplikace používá **pouze IPv6 sokety** = vidí **pouze IPv6 adresy**
- Provoz IPv4 je mapován pomocí **IPv4-mapovaných adres IPv6**
- Nedochází k žádnému překladu paketů
- Používání kompatibility s IPv4 se **nedoporučuje**:
 - správně napsané aplikace raději použijí **dva samostatné sokety**

Kompatibilita IPv6 soketů s IPv4



- Sokety pro IPv6 mohou *někdy* pracovat i s IPv4
- Tím lze **zjednodušit přidání podpory IPv4** do starých aplikací
- Aplikace používá **pouze IPv6 sokety** = vidí **pouze IPv6 adresy**
- Provoz IPv4 je mapován pomocí **IPv4-mapovaných adres IPv6**
- Nedochází k žádnému překladu paketů
- Používání kompatibility s IPv4 se **nedoporučuje**:
 - správně napsané aplikace raději použijí **dva samostatné sokety**
 - ani dnes nelze spoléhat na univerzální přítomnost podpory IPv6

Kompatibilita IPv6 soketů s IPv4



- Sokety pro IPv6 mohou *někdy* pracovat i s IPv4
- Tím lze **zjednodušit přidání podpory IPv4** do starých aplikací
- Aplikace používá **pouze IPv6 sokety** = vidí **pouze IPv6 adresy**
- Provoz IPv4 je mapován pomocí **IPv4-mapovaných adres IPv6**
- Nedochází k žádnému překladu paketů
- Používání kompatibility s IPv4 se **nedoporučuje**:
 - správně napsané aplikace raději použijí **dva samostatné sokety**
 - ani dnes nelze spoléhat na univerzální přítomnost podpory IPv6
 - OpenBSD kompatibilitu s IPv4 **nepodporuje**

Volba soketu IPV6_V6ONLY



Volba soketu IPV6_V6ONLY



`IPV6_V6ONLY=0`
kompatibilita povolena

výchozí pro Linux a macOS

Volba soketu IPV6_V6ONLY



IPV6_V6ONLY=0
kompatibilita povolena

výchozí pro Linux a macOS

IPV6_V6ONLY=1
kompatibilita zakázána

**výchozí pro Windows a BSD
vynuceno na OpenBSD**

Volba soketu IPV6_V6ONLY



IPV6_V6ONLY=0
kompatibilita povolena

výchozí pro Linux a macOS

IPV6_V6ONLY=1
kompatibilita zakázána

**výchozí pro Windows a BSD
vynuceno na OpenBSD**

- Přenositelné aplikace **musí hodnotu vždy nastavit**

Volba soketu IPV6_V6ONLY



IPV6_V6ONLY=0
kompatibilita povolena

výchozí pro Linux a macOS

IPV6_V6ONLY=1
kompatibilita zakázána

**výchozí pro Windows a BSD
vynuceno na OpenBSD**

- Přenositelné aplikace **musí hodnotu vždy nastavit**
- Povolená kompatibilita blokuje otevření podobného IPv4 soketu



IPv4-mapované IPv6 adresy na Internetu



IPv4-mapované IPv6 adresy

- Repräsentují IPv4 adresu v **IPv6 socket API**
- Neměly by **nikdy opustit hostitele**
- Neměly by se nikdy objevit v žádném **IPv6 paketu**
- Vložit je třeba do DNS **by byla hloupost**

IPv4-mapované IPv6 adresy



- Reprezentují IPv4 adresu v IPv6 socket API
- Neměly by nikdy opustit hostitele
- Neměly by se nikdy objevit v žádném IPv6 paketu
- Vložit je třeba do DNS by byla hloupost
- **Přesto to lidé dělají**

```
$ host bam.nr-data.net
bam.nr-data.net is an alias for bam.cell.nr-data.net.
bam.cell.nr-data.net is an alias for fastly-tls12-bam.nr-data.net.
fastly-tls12-bam.nr-data.net has address 162.247.243.29
fastly-tls12-bam.nr-data.net has IPv6 address ::ffff:162.247.243.29
```

Proč by někdo něco takového dělal?



We did this to drive down the cost with our DNS provider. Queries for AAAA records that didn't exist, followed by queries for A records, was costing us significantly and we needed to alleviate that.

Our AAAA answers follow the standards, and our local dual-stack testing has shown no issues. The IPv4 addresses embedded in the IPv6 answers should be accurate, and should match the A record requests, and should all be routable in the IPv4 space.

Source: [New Relic support forum](#), shared by [Thomas Schäfer](#)

To je nesmysl. Nebo ne?



- Vytvořil jsem **dvě testovací stránky**:

To je nesmysl. Nebo ne?



- Vytvořil jsem **dvě testovací stránky**:

<https://ipv4-mapped.0skar.cz>

- pouze AAAA záznam mířící na IPv4-mapovanou IPv6 adresu
- neměla by být **nikdy dostupná**

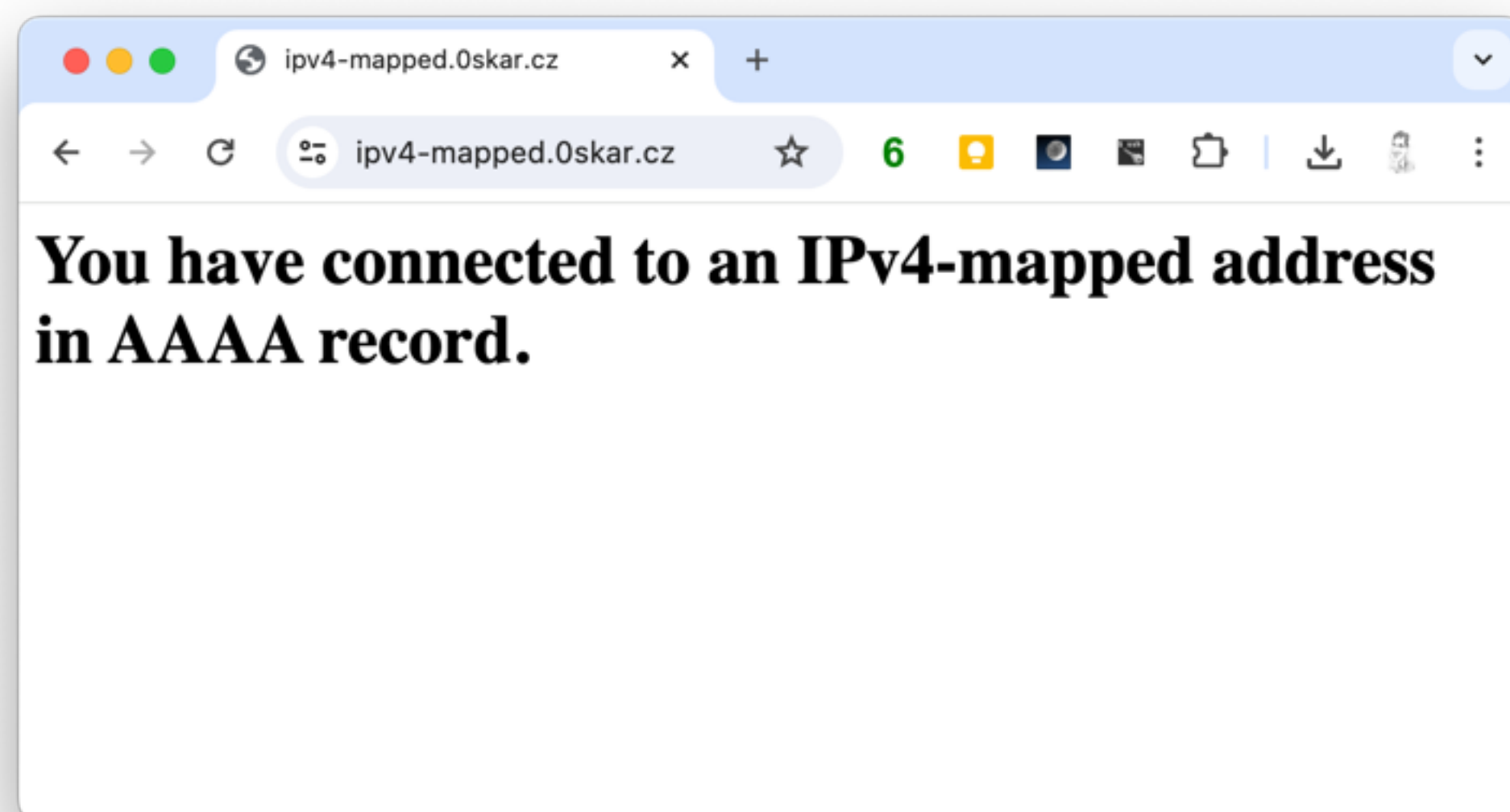
To je nesmysl. Nebo ne?



- Vytvořil jsem **dvě testovací stránky**:

<https://ipv4-mapped.0skar.cz>

- pouze AAAA záznam mířící na IPv4-mapovanou IPv6 adresu
- neměla by být **nikdy dostupná**



To je nesmysl. Nebo ne?



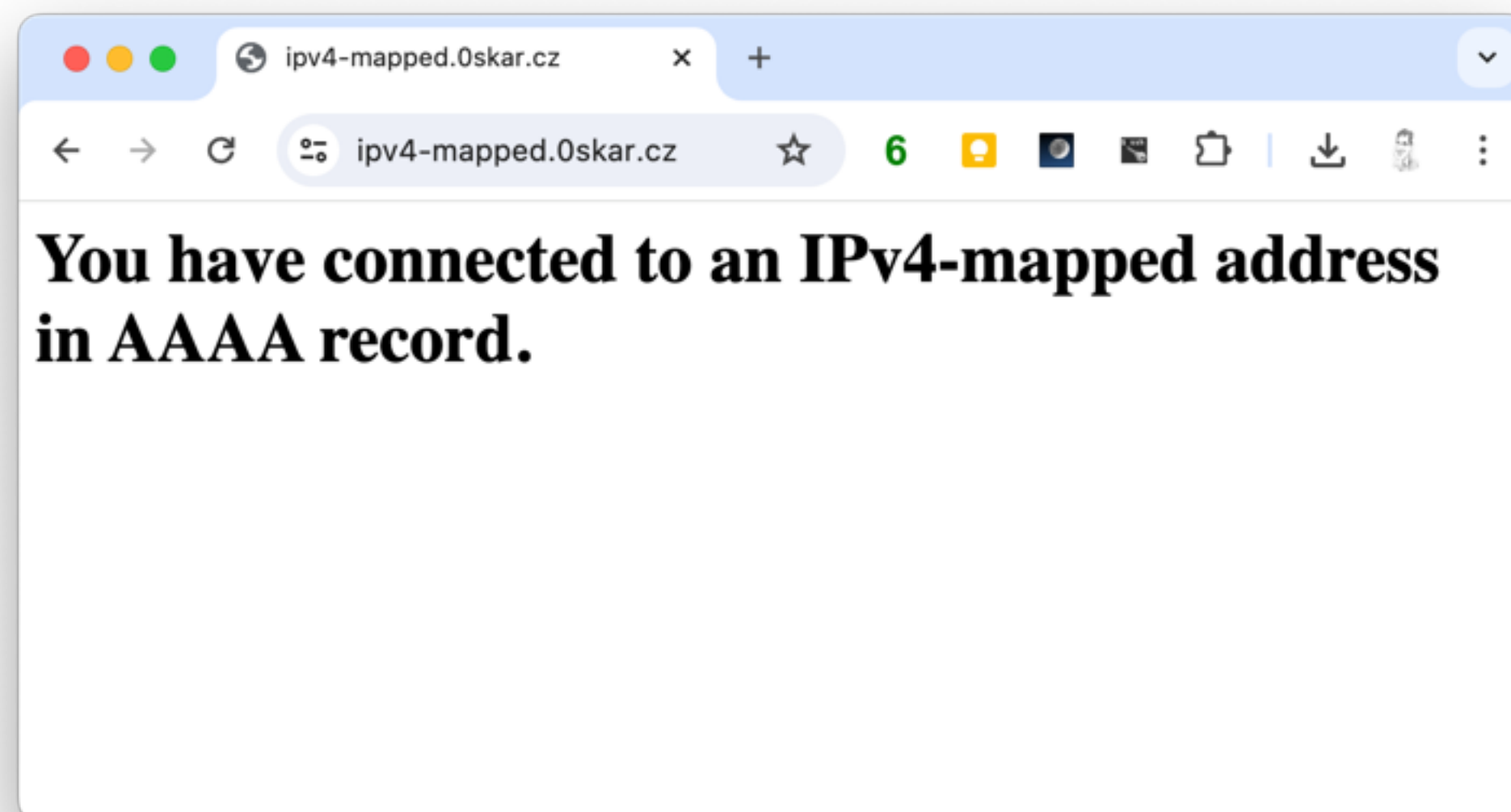
- Vytvořil jsem dvě testovací stránky:

<https://ipv4-mapped.0skar.cz>

- pouze AAAA záznam mířící na IPv4-mapovanou IPv6 adresu
- neměla by být **nikdy dostupná**

<https://ipv4-mapped-pref.0skar.cz>

- dual stack s AAAA záznamem mířícím na IPv4-mapovanou IPv6 adresu
- měla by vždy **ukázat obsah A-záznamu**



To je nesmysl. Nebo ne?



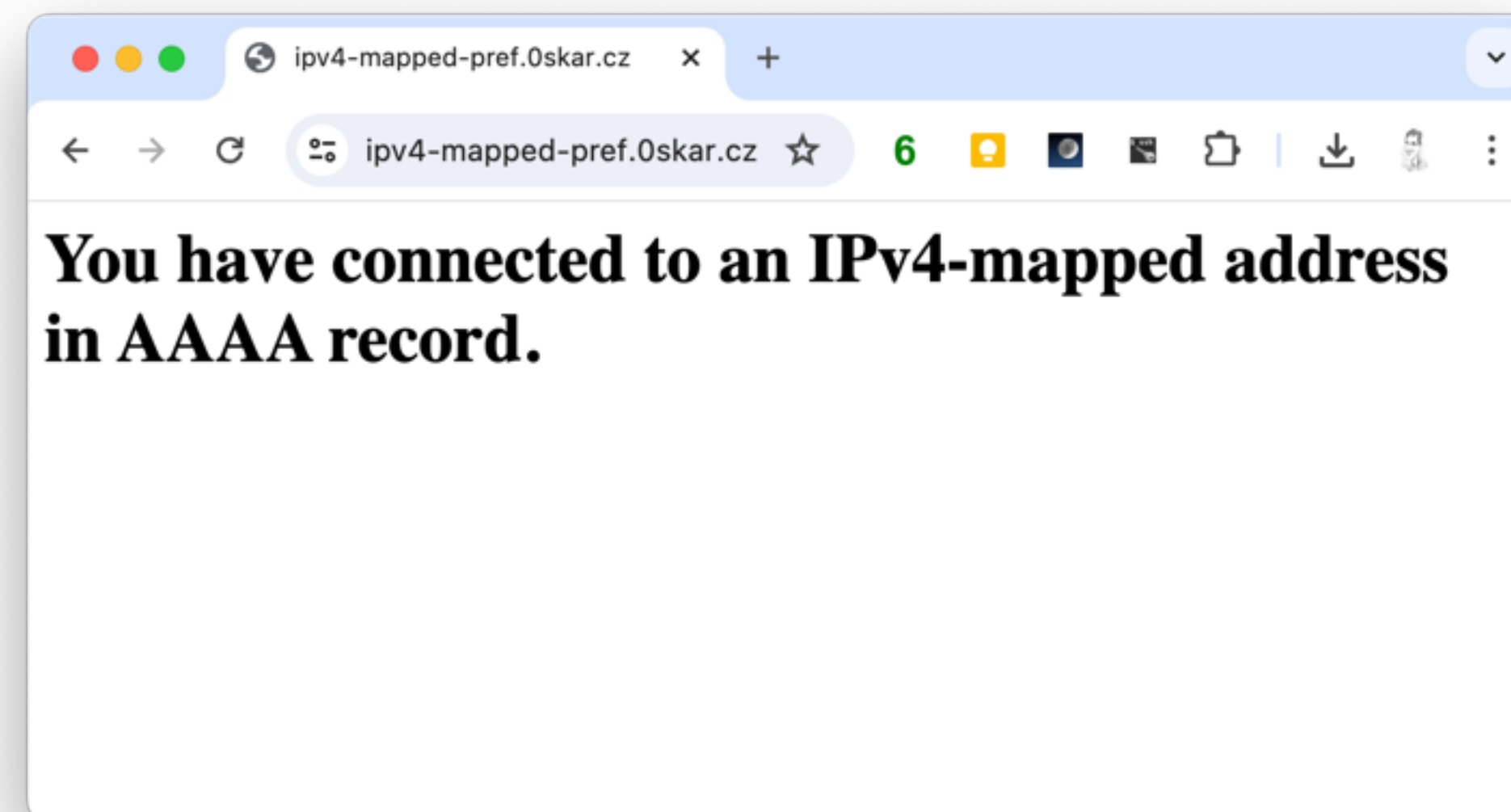
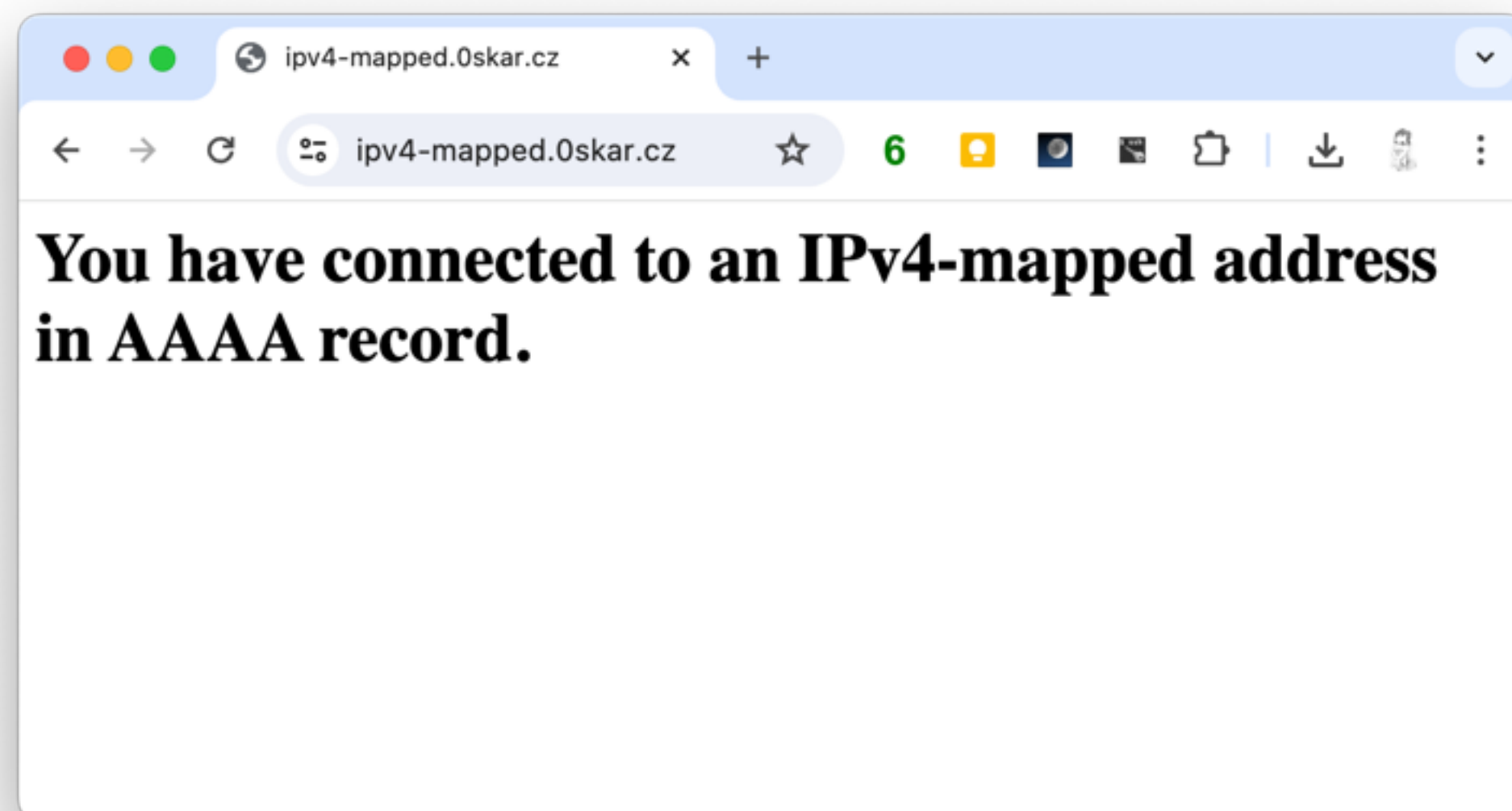
- Vytvořil jsem dvě testovací stránky:

<https://ipv4-mapped.0skar.cz>

- pouze AAAA záznam mířící na IPv4-mapovanou IPv6 adresu
- neměla by být **nikdy dostupná**

<https://ipv4-mapped-pref.0skar.cz>

- dual stack s AAAA záznamem mířícím na IPv4-mapovanou IPv6 adresu
- měla by vždy **ukázat obsah A-záznamu**



Máme tu problém



- Výsledky závisí na:
 - operačním systémem
 - prohlížeči
 - síti (dual-stack vs. IPv6-only)
- Všechna zařízení se ptala jak na **AAAA** tak i na **A** záznamy
 - nelze tedy ušetřit množství dotazů vložením IPv4 adresy do AAAA záznamu

Máme tu problém



- Výsledky závisí na:
 - operačním systému
 - prohlížeči
 - síti (dual-stack vs. IPv6-only)
- Všechna zařízení se ptala jak na **AAAA** tak i na **A** záznamy
 - nelze tedy ušetřit množství dotazů vložením IPv4 adresy do AAAA záznamu

```
$ curl https://ipv4-mapped.0skar.cz
curl: (6) Could not resolve host: ipv4-mapped.0skar.cz
$ curl https://ipv4-mapped-pref.0skar.cz
<h1>You have connected to an IPv4 address in A record.</h1>
```

macOS na síti typu dual-stack

Máme tu problém



- Výsledky závisí na:
 - operačním systému
 - prohlížeči
 - síti (dual-stack vs. IPv6-only)
- Všechna zařízení se ptala jak na **AAAA** tak i na **A** záznamy
 - nelze tedy ušetřit množství dotazů vložením IPv4 adresy do AAAA záznamu

```
$ curl https://ipv4-mapped.0skar.cz
curl: (6) Could not resolve host: ipv4-mapped.0skar.cz
$ curl https://ipv4-mapped-pref.0skar.cz
<h1>You have connected to an IPv4 address in A record.</h1>
```

macOS na síti typu dual-stack

```
$ curl https://ipv4-mapped.0skar.cz
<h1>You have connected to an IPv4-mapped address in AAAA record.</h1>
$ curl https://ipv4-mapped-pref.0skar.cz
<h1>You have connected to an IPv4-mapped address in AAAA record.</h1>
$ curl https://ipv4-mapped-pref.0skar.cz
<h1>You have connected to an IPv4 address in A record.</h1>
```

macOS na síti typu IPv6-only

Je to opravdu problém?



- Happy Eyeballs úspěšně **skrývá podobné problémy**
- Nefunkční AAAA záznam **rozbije DNS64**
 - to lze řešit nastavením DNS64 tak, aby **ignorovalo AAAA záznamy** s adresami mimo Global Unicast
- Lze předpokládat možnosti obejití nejrůznějších aplikačních firewallů pomocí IPv4-mapované IPv6 adresy
 - Například GitHub incident report z ledna 2024



Co se s tím dá dělat?

- **Provozovatelé DNS64:**
 - ignorujte adresy mimo 2000::
- **Výrobci operačních systémů a/nebo prohlížečů:**
 - možná filtrujte IPv4-mapované IPv6 adresy v resolveru?
- **Provozovatelé autoritativního DNS hostingu:**
 - nepožadujte větší platbu za prázdné odpovědi
- **Kdokoli:**
 - navrhněte v IETF nový dokument o *nepřípustném použití* IPv4-mapovaných IPv6 adres



Otázky

Ondrej@Caletka.cz
<https://Ondrej.Caletka.cz>



IPv6 Fundamentals E-learning Course

- ✓ Free online course
- ✓ Study at your own pace
- ✓ Interactive learning



academy.ripe.net

