Domain Name System

Domain Name System (DNS) slouží pro pohodlí uživatelů. **Umožňuje používat místo IP adres symbolická jména počítačů, uspořádaná do hierarchické struktury**.

Jeho dvě nejzákladnější funkce jsou:

- převod jména na IP adresu
- a naopak převod IP adresy na odpovídající jméno.

V případě IPv6 je role DNS velmi významná, protože zdejší adresy jsou dlouhé a obtížně se pamatují i zapisují. Bohužel příběh IPv6 a DNS je historií plnou omylů, která jen velmi velmi pomalu spěje ke šťastnému konci.

IPv6 v DNS má dvě stránky:

- za prvé je třeba ukládat IPv6 adresy do DNS, aby bylo možné získávat ke jménům adresy obou protokolů a naopak (IPv6 obsažené v DNS záznamech)

- za druhé se pak musí DNS servery zpřístupnit pro IPv6, aby se jich klienti hovořící novým protokolem vůbec měli jak zeptat (IPv6 jako transport pro DNS).

Tyto dva problémy jsou nezávislé. Ten první vyžaduje standardizaci ze strany IETF, ten druhý je otázkou implementací DNS serverů a úsilí jejich správců.

V roce 1995 vyšlo RFC 1886: *DNS Extensions to* historie *support IP version 6*, které definovalo jednoduché, ale nepříliš silné konstrukce pro ukládání IPv6 informací do DNS.

Po pěti letech následovalo RFC 2874: *DNS Extensions to Support IPv6 Address Aggregation and Renumbering*, které zavedlo jiné (podstatně vypečenější) mechanismy. Zároveň prohlásilo předchozí specifikaci za zastaralou.

Začala zuřivá debata a vázla jeho implementace a správci řady sítí odmítali nově definované DNS záznamy používat. Schizma se prohloubilo v létě 2001, kdy setkání IETF vyhlásilo novější specifikaci za experimentální a pro produkční sítě doporučilo vrátit se k RFC 1886.

Potěšilo zejména ty, kteří se od původních mechanismů nikdy neodklonili. Tento názor o rok později potvrdilo RFC 3363: *Representing Internet Protocolversion 6 (IPv6) Addresses in the Domain Name System (DNS)*.

Snad poslední slovo vneslo v roce 2003 RFC 3596: *DNSExtensions to Support IP Version 6*. **Představuje návrat na původní pozice (s velmi drobnými změnami),** o RFC 2874 se vůbec nezmiňuje. To má statut experimentálního protokolu, stejně jako jeho souputník RFC 2673 pro zápis binárních prefixů v reverzních adresách.

Kromě ukládání IPv6 informací do DNS je také třeba zajistit jeho přenos novým protokolem. Do horních pater doménové hierarchie proniká IPv6 mimořádně pomalu. V polovině roku 2008 podporuje nový protokol osm ze třinácti kořenových serverů, tedy zhruba dvě třetiny. Ovšem třeba pro doménu *com* se k IPv6 hlásí jen dva z jejích třinácti serverů, ze šestice serverů norské domény *no* pak ani jeden. Doména *cz* je v tomto směru přímo vzorová – pět ze šesti jejích autoritativních serverů hovoří oběma protokoly[1].

IPv6 adresy v DNS

Ukládání IPv6 informací do DNS řeší RFC 3596. Je jednoduché a drží se celkem přímočaře řešení, které se používá pro IPv4.

Pro dopředné dotazy (tedy zjišťování dopředné dotazy (AAAA) adresy k danému jménu) zavedlo nový typ záznamů nazvaný *AAAA*. V IPv4 k tomuto účelu slouží záznamy *A* a jelikož je délka IPv6 adresy čtyřnásobná, odrazila se tato skutečnost v názvu nového záznamu.

Příklad:
Má-li počítač *pc.kdesi.cz* adresu 2001:db8:89ab:1:123:45ff:fe67:89ab, bude v zónovém souboru (Zónový soubor obsahuje definici dané domény. Tento pojem má původ v programu BIND, nejpoužívanějším DNS serveru) pro doménu *kdesi.cz* obsažen záznam

```
pc          AAAA   2001:db8:89ab:1:123:45ff:fe67:89ab
```

Celá definice domény *kdesi.cz*, v níž se nacházejí dva autoritativní DNS servery ve dvou různých podsítích a jeden počítač, by mohla vypadat následovně:

```
$ORIGIN kdesi.cz.
@           SOA ns1.kdesi.cz. root.ns1.kdesi.cz. (
            2008011200 ; serial
            28800      ; refresh
            14400      ; retry
            3600000    ; expire
            86400      ; default_ttl
            )


;DNS servery
             NS   ns1
             NS   ns2

;adresy počítačů
ns1          AAAA   2001:db8:89ab:1:2a0:ecff:fe12:3456
ns2          AAAA   2001:db8:89ab:2:2a0:ecff:fe12:7890
pc           AAAA   2001:db8:89ab:1:123:45ff:fe67:89ab
```

Nepříjemnou vlastností tohoto přístupu je, že pokud síť používá několik prefixů a počítače tedy mají několik adres, musí mít i odpovídající počet AAAA záznamů.

Kdyby například síť z předchozího příkladu kromě prefixu 2001:db8:89ab::/48 používala navíc třeba prefix 2002:a00:1::/48 pro automatické tunelování 6to4 zónový soubor by rázem nabotnal:

```
$ORIGIN kdesi.cz.
@         SOA ns1.kdesi.cz. root.ns1.kdesi.cz. (
          2008011200 ; serial
          28800      ; refresh
          14400      ; retry
          3600000    ; expire
          86400      ; default_ttl
          )

;DNS servery
          NS  ns1
          NS  ns2

;adresy počítačů
ns1       AAAA  2001:db8:89ab:1:2a0:ecff:fe12:3456
          AAAA  2002:a00:1:1:2a0:ecff:fe12:3456
ns2       AAAA  2001:db8:89ab:2:2a0:ecff:fe12:7890
          AAAA  2002:a00:1:2:2a0:ecff:fe12:7890
pc        AAAA  2001:db8:89ab:1:123:45ff:fe67:89ab
          AAAA  2002:a00:1:1:123:45ff:fe67:89ab
```

Zpětné dotazy (PTR)

Zpětný dotaz vychází ze známé IPv6 adresy a snaží se k ní získat jméno.Stejně jako ve světě IPv4 se používají záznamy typu PTR. Dotaz je položen prostřednictvím doménového jména sestaveného tak, že se obrátí pořadí šestnáctkových číslic v adrese a na konec se připojí **doména *ip6.arpa*** (Původně byla v RFC 1886 pro tento účel použita doména *ip6.int*. Pozdější RFC 2874 použilo pro tento účel *ip6.arpa*, což navíc odpovídá praxi z IPv4. Aby se domény uvedly do souladu, prohlásilo RFC 3152: *Delegation of IP6.ARPA* doménu *ip6.int* za zavrženou a předepsalo použití *ip6.arpa* pro veškeré reverzní domény IPv6. Na tom trvá i RFC 3596).

Nuly se nesmí vynechávat, adresa musí být kompletní. Takže pro výše zmiňovanou adresu by reverzní dotaz měl tvar

b.a.9.8.7.6.e.f.f.f.5.4.3.2.1.0.1.0.0.0.b.a.9.8.8.b.d.0.1.0.0.2.ip6.arpa

Díky obrácenému pořadí číslic se **obecná část adresy (prefix) dostává na konec a lze realizovat distribuovanou správu reverzních domén**.

Například síť instituce vlastnící doménu *kdesi.cz* má prefix 2001:db8:89ab::/48 a tudíž dostane do správy reverzní doménu *b.a.9.8.8.b.d.0.1.0.0.2.ip6.arpa*.Pro počítač *pc.kdesi.cz* by její zónový soubor obsahoval záznam

```
b.a.9.8.7.6.e.f.f.f.5.4.3.2.1.0.1.0.0.0  PTR  pc.kdesi.cz.
```

Celá reverzní zóna odpovídající výše uvedené doméně *kdesi.cz* by musela obsahovat:

```
$ORIGIN b.a.9.8.8.b.d.0.1.0.0.2.ip6.arpa.
@         SOA ns1.kdesi.cz. root.ns1.kdesi.cz. (
          2008011200 ; serial
          28800      ; refresh
          14400      ; retry
          3600000    ; expire
          86400      ; default_ttl
          )

;DNS servery
          NS  ns1.kdesi.cz.
          NS  ns2.kdesi.cz.

;reverzní záznamy
6.5.4.3.2.1.e.f.f.f.c.e.0.a.2.0.1.0.0.0  PTR  ns1.kdesi.cz.
0.9.8.7.2.1.e.f.f.f.c.e.0.a.2.0.2.0.0.0  PTR  ns2.kdesi.cz.
b.a.9.8.7.6.e.f.f.f.5.4.3.2.1.0.1.0.0.0   PTR   pc.kdesi.cz.
```

Reverzní domény jsou dlouhé, což je důsledkem dlouhých IPv6 adres. Není třeba psát je úplně celé (díky $ORIGIN), ale i tak zbyde v zónovém souboru 16 až 20 číslic.

Pracné, ale robustní. To je krédo, v jehož duchu je postaveno RFC 3596 (a jeho předchůdce RFC 1886).

Rozsah IPv6 záznamů si de facto vynucuje strojové generování zónových souborů. Pokud není cílová síť opravdu malá a adresy a jména v ní se mění obvyklou rychlostí, **je ruční editace zónových souborů nepříjemná a riziko zanesení chyb vysoké**. Vřele doporučuji poohlédnout se po vhodném programu, který povede databázi počítačů a zónové soubory podle ní na přání vygeneruje.

Obsah domén

Technické řešení pro poskytování IPv6 informací v DNS je celkem jednoduché. Přesto bylo třeba řešit následující:

- **rozhodnutí, jaké IPv6 adresy vlastně do DNS ukládat**
- **jak koncipovat vztah jmen a adres obou protokolů.**

Rozhraní počítače adresy v DNS nese několik IPv6 adres s různým dosahem i životností. Které z nich patří do DNS a které nikoli?

Do DNS rozhodně zařaďte:

- všechny globální individuální adresy stroje s dlouhodobější platností
- dlouhodobě platné adresy přechodových mechanismů, jako je třeba 6to4.

Do DNS naopak nepatří:

- **Lokální linkové adresy** – mají platnost jen pro místní linku, DNS klient pochází odkudkoli a nemá žádnou šanci zjistit, která linka je ta pravá. Obecně platí, **že adresy omezeného dosahu nemají v DNS co dělat.**

- **Náhodně generované krátkodobé adresy** pro zachování soukromí – jednak by bylo třeba obsah zóny neustále aktualizovat, ale především by se vazbou na stejné jméno zcela zabil jejich účel zamezit sledování pohybu stroje v Internetu.

- **Otevřenou otázkou zůstávají adresy, které správce sítě nemá pod přímou kontrolou.** Jedná se o adresy získané bezstavovou automatickou konfigurací či přidělované liberálním DHCP serverem nastaveným ve stylu „dám nějakou adresu každému, kdo si o ni řekne“. Vznikají víceméně náhodně a navíc mívají krátkodobý charakter. Pro ně neexistuje univerzální doporučení. Nic se nestane, když v DNS nebudou. Pokud je tam chcete mít, bude zřejmě třeba nasadit dynamické aktualizace DNS, jejichž pomocí si klient se serverem dohodne doménové jméno a server je následně zařadí do DNS.

Strategie pojmenování

Pokud má **počítač jen IPv4 nebo IPv6 adresu, je jeho zařazení do DNS jednoznačně** dáno. Jak se ale zachovat, když komunikuje oběma protokoly a má samozřejmě jejich adresy?

Jsou dva základní přístupy:

Stejné jméno

IPv4 i IPv6 adresu přidělíte stejnému jménu. Pokud naše experimentální *pc.kdesi.cz* bude mít IPv4 adresu 10.1.2.3, budou jeho dopředné záznamy následující (v IPv6 jsem zachoval dvojici prefixů):

```
pc          A     10.1.2.3
            AAAA  2001:db8:89ab:1:123:45ff:fe67:89ab
            AAAA  2002:a00:1:1:123:45ff:fe67:89ab
```

Pokud někdo chce komunikovat s *pc.kdesi.cz*, dodá mu DNS všechny tři adresy a jeho stroj si podle svého připojení vybere. Komunikace novým protokolem proběhne zcela transparentně, aniž by se kdo co dozvěděl.

Mohou se ale, objevit **nepříjemnosti.**

Současné operační systémy totiž při dostupnosti obou protokolů dávají většinou přednost IPv6. Když nebude dobře fungovat IPv6 spojení mezi oběma stroji, dopadne to špatně. Taková situace bohužel v současnosti není vzácná.
Různé dohledové programy a systémy upozorňující na nefunkčnosti v síti si často hledí jen IPv4. Pokud nastane takový problém, spojení se nepodaří navázat. To ovšem musí zjistit až transportní protokol TCP, protože IP si doručování dat nijak nepotvrzuje. Čeká se tedy na vypršení trpělivosti TCP, což trvá desítky vteřin až minuty, teprve pak vzdálený počítač zkusí jinou alternativu a nakonec se snad dostane k něčemu fungujícímu.

Dále se mohou objevit **zádrhele na aplikační úrovni**.
Ilustrační příklad: Pro vzdálený přístup k serveru používám SSH. Nezadávám heslo, mám vygenerován soukromý a veřejný klíč, jejichž prostřednictvím se můj klient autentizuje. Najednou se začal dožadovat hesla. Důvodem bylo, že jsme serveru výše popsaným způsobem přiřadili IPv6 adresu. Můj stroj také disponuje IPv6 adresou, SSH tedy automaticky přešlo na nový protokol, ale já jsem u svého veřejného klíče na serveru měl nastaveno, že platí jen z IPv4 adresy mého počítače. Musel jsem přidat také jeho IPv6 adresu, abych mohl klíč nadále používat.

Odlišná jména

Druhou možnou strategií je používat pro IPv6 adresy odlišná jména. Bývá obvyklé zavést speciální poddoménu (často nazvanou *ip6* nebo *ipv6*) a do ní je zařadit. V našem experimentálním případě by tedy staré adresy zůstaly v doméně *kdesi.cz*, jména přiřazená IPv6 by končila *ip6.kdesi.cz*. Konkrétně pro stroj *pc* by definice vypadala následovně:

```
pc          A       10.1.2.3

pc.ip6      AAAA    2001:db8:89ab:1:123:45ff:fe67:89ab
            AAAA    2002:a00:1:1:123:45ff:fe67:89ab
```

V tomto uspořádání **z použitého jména přímo vyplývá komunikační protokol.** Zadá-li někdo DNS k řešení jméno *pc.kdesi.cz*, dostane pouze IPv4 adresu. Jestliže použije *pc.ip6.kdesi.cz*, získá dvojici IPv6 adres.

Tahle cesta představuje sázku na jistotu. Použil ji například *Google*, když začal poskytovat svůj slavný vyhledávač po IPv6. Případné prodlevy při problémech se síťovým protokolem, které jsem popsal výše, jsou zde zcela neakceptovatelné. Proto *www.google.com* vede pouze na IPv4 adresy, zatímco za jménem *ipv6.google.com* se skrývají IPv6 adresy téže služby.

Oddělení adres samozřejmě znamená útlum IPv6 provozu – většina uživatelů bude používat stará známá jména, která mohou být i vestavěna v aplikacích.

Může vadit (nebo vyhovovat),že z údajů poskytnutých DNS se nedá nijak zjistit, že *pc.kdesi.cz* a *pc.ip6.kdesi.cz* mají něco společného. Jsou to dvě zcela oddělená jména se svými reverzními záznamy. Je *www.google.com* a *ipv6.google.com* stejný stroj? To vědí jen u Google.

Nikde není řečeno, že všechna jména v doméně musí používat stejnou koncepci.
Příklad:
V síti TU v Liberci kombinujeme oba postupy: Pro běžné uživatelské počítače používáme oddělená jména, protože na jedné straně chceme umožnit experimenty s IPv6, ale na straně druhé se u nich na kvalitní
podporu IPv6 nelze spolehnout. Pouze u vybraných serverů, u kterých za dostupnost IPv6 ručíme a chceme transparentně používat nový protokol kdykoli to je možné, používáme společné jméno pro A i AAAA záznamy.
Variantu s oddělenými jmény lze také vnímat jako dočasné řešení. Až si budete jisti, že IPv6 služba je dostatečně robustní a dostupná, opustíte její samostatné jméno a přejdete na první variantu.

Provozní záležitosti

Podporovat také IPv4

Několik informativních RFC se zabývá pragmatickými otázkami soužití IPv6 sDNS. Příjemně minimalistické je RFC 3901: *DNS IPv6 Transport Operational Guidelines*, jehož hlavní myšlenka by se dala shrnout do věty „Nebuďte nadměrně progresivní, nedělejte DNS závislé na IPv6.“

Dvě doporučení.

První se týká rekurzivních serverů v koncových sítích, které řeší dotazy místních klientů. Podle RFC 3901 by měly mít možnost komunikovat oběma protokoly, aby nezůstaly bezmocné, pokud pro některou doménu dostanou jen IPv4 adresy autoritativních serverů.

Třeba pro výše zmiňované Norsko. V polovině roku 2008 nemá žádný z autoritativních serverů domény *no* IPv6 adresu. To samozřejmě neznamená, že v celém Norsku nemají IPv6, ani že někde v norských doménách nejsou AAAA záznamy. Ale aby se k nim server dostal, musí po cestě pohovořit

s některým ze serverů pro doménu první úrovně *no*, a to protokolem IPv4 – jiný není k mání. Pokud by váš server hovořil jen IPv6, nedokázal by to a celé norské DNS by pro něj zůstalo nedostupné.

Dostupnost IPv4 pro místní rekurzivní server lze zařídit i zprostředkovaně. Pokud se jedná o čistou IPv6 síť, mohlo by být přivedení IPv4 k jejímu serveru zbytečně komplikované. Naštěstí **implementace DNS serverů obvykle počítají s alternativou, že dotyčný počítač nemá přímý přístup k Internetu -** přistupují přes **prostředníka (forwarder)**.

Koncový DNS server pak spravuje pouze svou vyrovnávací paměť a jakékoli dotazy, které její pomocí není schopen zodpovědět, předává prostředníkovi.

V našem případě by komunikace mezi koncovým serverem a jeho prostředníkem probíhala protokolem IPv6, zatímco prostředník by hovořil oběma protokoly a dokázal se domluvit s kterýmkoli DNS serverem potřebným k vyřešení dotazu.

Druhým konkrétním doporučením obsaženým v RFC 3901 je nedělat opačnou věc. Nevytvářet domény, jejichž všechny autoritativní servery hovoří pouze IPv6. V Internetu existuje řada klientů a serverů podporujících pouze IPv4. Alespoň jeden ze serverů každé domény by měl podporovat starší verzi IP. Ideální samozřejmě je poskytnout dostatečný počet autoritativních serverů jak pro IPv4 tak pro IPv6, aby nikdo neměl problém domluvit se s nimi svým nativním protokolem.

AAAA záznamy fungujícím strojům

Nejrozsáhlejší a nejčerstvější soubor doporučení a úvah ohledně provozování IPv6 DNS najdete v RFC 4472: *Operational Considerations and Issues with IPv6 DNS*. Probírá komplexně celou problematiku, mimo jiné i **otázky adres vhodných pro zařazení do DNS či strategii pojmenování.**

Zdůrazňuje se zde, že AAAA záznam je vhodné ke jménu přiřadit, až když IPv6 na daném stroji skutečně funguje. Tedy má nakonfigurovánu příslušnou adresu, je připojen do dosažitelné sítě s fungujícím IPv6 a nový protokol podporují i provozované služby. Toto doporučení platí dvojnásob, pokud používáte strategii společného jména pro oba protokoly.

Dozvíte se také, co se stane, když v DNS nastavíte různou dobu životnosti (TTL) záznamům typu A a AAAA. Prozradím rovnou ponaučení celé bajky: nedělejte to.

Dost zevrubně jsou diskutovány otázky dynamické aktualizace DNS při stavové či bezstavové konfiguraci klientů. A pokud byste měli ambice napsat DNS klienta, najdete zde kapitolu s doporučeními, jak by se měl vůči IPv6 chovat.