Securityworld, 3. června 2011

DNSSEC část první aneb je potřeba začít od píky

Dnešní internet je po fyzické stránce změť kabelů, ať už metalických nebo optických, nějaké ty směrovače a přepínače a k tomu hromada různých serverů, na kterých běží služby. Vzhledem k tomu, že si lidé hůře pamatují čísla než jména a naopak počítače (a směrovací protokoly) nerozumí jménům, ale číslům (říkejme jim třeba IP adresa), musel vzniknout systém na překlad jmenných názvů na číselné IP adresy. Úvodní článek ze série o DNSSEC bude tedy o základních principech fungování DNS.

Princip DNS

Jak tedy systém DNS pracuje? Obecně se na celý systém DNS dá pohlížet jako na decentralizovanou hierarchickou databázi, jejíž hlavní smysl je poskytovat informace nutné pro překlad jmen na IP adresy. Informace uložené v DNS se ukládají ve formě, která se nazývá Resource Record (dále též RR záznam). Protože data uložená v RR záznamech jsou spíše statického charakteru, mají samotné informace i specifikaci maximální doby platnosti záznamu, tzv. TTL (Time to Live). Celá struktura jednoho RR záznamu obsahuje vlastníka (owner), třídu (class), typ (type), ttl a data (rdata – resource data). Třídy byly v návrhu specifikovány dvě: IN – pro Internet, a CH – pro Chaos. Třída Chaos byla vytvořena pro experimentální účely a v praxi se s ní můžete potkat pouze při speciálním typu dotazu na verzi DNS serveru (CH TXT version.bind.) nebo na identifikátor serveru v cloudu (CH TXT server.id.). Mezi základní typy RR záznamu patří: A – pro IPv4 adresy, MX – pro směrování pošty, AAAA – pro IPv6 adresy, a další. Tyto typy jsou registrovány u organizace IANA a jsou postupně doplňovány dle potřeby dalších protokolů, které s DNS pracují. Např. rozšíření DNSSEC zavádí hned několik nových typů RR záznamů. V datech RR záznamu jsou pak uloženy informace, které se liší dle typu záznamu, např. již zmiňovaný A záznam v datech obsahuje IPv4 adresu. A ještě jedna technická poznámka. RR záznamy jsou sdruženy podle názvu, třídy a typu do tzv. RRsetů, které sdílejí stejné TTL. Pokud například máte více poštovních serverů, tak nelze při zadávání MX záznamů mít pro každý server jiné TTL.

Formát RR záznamu:

Název	**Popis**
Vlastník (Owner Name)	Doménové jméno vlastníka RR záznamu
Typ (Type)	Typ RR záznamu (2 bajty)
Třída (Class)	Třída RR záznamu (2 bajty)

TTL	Délka platnosti záznamu (4 bajty)
Délka sekce RDATA (RDLENGTH)	Délka sekce RDATA (2 bajty)
RDATA	Data RR záznamu

Programy, které s DNS pracují, můžeme rozdělit na dvě základní kategorie: klient a server. Klient je program, který umí procházet hierarchickou strukturu DNS a zjišťovat informace uložené v RR záznamech. Server je pak program, který umí odpovídat na dotazy klientů, pro konkrétní doménová jména, pro které je nakonfigurován. Protože by implementace obecného DNS klienta do každého programu byla složitá, je v systémové knihovně implementovaný jednoduchý DNS klient (stub resolver), který se umí zeptat jenom na konkrétní jméno a očekává odpověď v jednoduché formě. Z tohoto důvodu může být DNS klient zároveň i serverem, který poskytuje informace tomuto stub resolveru. DNS server, který se chová jako klient, budeme nazývat resolver nebo také rekurzivní DNS (rDNS). Protože rDNS má většinou (nikoli nutně) vyrovnávací paměť, do které si ukládá výsledky již provedených dotazů, můžeme se také setkat s označením kešující DNS. DNS server, který odpovídá na dotazy, nazveme autoritativní DNS (aDNS). A aby to nebylo příliš jednoduché, tak DNS server může vystupovat v obou rolích zároveň. Na dotazy na doménová jména, pro které je server autoritativní, odpovídá přímo – chová se jako aDNS, ostatní dotazy přeposílá jiným autoritativním serverům, tedy se chová jako rekurzivní DNS server. Nicméně tato konfigurace není doporučována a z různých důvodů je lepší mít tyto dvě role – rekurzivní a autoritativní – odděleny.

Pomalu jsme se propracovali přes definice k tomu, co se vlastně stane, když program potřebuje např. přeložit jmenný název na IP adresu. Překlad jména na IP adresu se provede voláním systémové funkce `getaddrinfo`, která vytvoří strukturu `addrinfo`. Struktura `addrinfo` umí IPv6 i IPv4 adresy a program ji může využít dále pro vytváření IP spojení. Co se vlastně všechno stane ve chvíli, kdy dojde k volání `getaddrinfo`? Otevřete prohlížeč a chcete se podívat na stránky čtvrtletníku Securityworld, napíšete tedy do svého prohlížeče www.securityworld.cz. Prohlížeč zavolá funkci:

```
getaddrinfo("www.securityworld.cz", 80, &hints, &result).
```

Voláním této funkce se předá kontrola stub resolveru, který se nejprve podívá do lokálního souboru /etc/hosts (na systémech Windows je soubor uložený jako %SystemRoot%\system32\drivers\etc\hosts.txt) a pokud informaci nenalezne zde, zeptá se nakonfigurovatných rekurzivních DNS serverů. Moderní

systémy mezi tyto dvě metody zařazují navíc ještě multicast DNS (neboli také Bonjour protokol). Ale pojďme zpátky k našemu dotazu na www.securityworld.cz. Rekurzivní DNS server přijme dotaz a podívá se, zda-li odpověď nemá ve vyrovnávací paměti, pokud ano, vrátí rovnou tuto odpověď. V opačném případě začne hierarchicky od kořenové úrovně (jinak také root, v DNS je reprezentovaný nepovinnou tečkou úplně napravo v doménovém jméně) prohledávat DNS databázi. Aby ale vůbec mohl začít musí mít staticky nakonfigurované DNS servery pro kořenovou úroveň. Takových serverů je v současné době 13 a jejich kompletní seznam včetně rozmístění po světě můžete nalézt na stránkách www.root-servers.org. Mimochodem sdružení CZ.NIC poskytuje prostor pro umístění hned dvou serverů pro kořenovou zónu (označenými písmeny F a L). Náš rekurzivní DNS se zeptá náhodně vybraného kořenového serveru na „www.securityworld.cz“, tento server ovšem tuto informaci neví (protože je systém hierarchický, distribuovaný a decentralizovaný), ale ví, že správcem domény .cz je CZ.NIC, resp. jeho DNS servery. Proto odpoví „já to nevím, ale zeptej se serverů CZ.NICu“, prakticky to pak vypadá takto:

```
;; QUESTION SECTION:
;www.securityworld.cz.        IN A

;; AUTHORITY SECTION:
cz.               172800 IN NS a.ns.nic.cz.
cz.               172800 IN NS b.ns.nic.cz.
cz.               172800 IN NS c.ns.nic.cz.
cz.               172800 IN NS d.ns.nic.cz.
cz.               172800 IN NS f.ns.nic.cz.

;; ADDITIONAL SECTION:
a.ns.nic.cz.            172800 IN A 194.0.12.1
b.ns.nic.cz.            172800 IN A 194.0.13.1
c.ns.nic.cz.            172800 IN A 194.0.14.1
d.ns.nic.cz.            172800 IN A 193.29.206.1
f.ns.nic.cz.            172800 IN A 193.171.255.48
a.ns.nic.cz.            172800 IN AAAA 2001:678:f::1
b.ns.nic.cz.            172800 IN AAAA 2001:678:10::1
c.ns.nic.cz.            172800 IN AAAA 2001:678:11::1
d.ns.nic.cz.            172800 IN AAAA 2001:678:1::1
f.ns.nic.cz.            172800 IN AAAA 2001:628:453:420::48
```

Rekurzivní DNS server si přečte, že se má zeptat serverů CZ.NICu a pošle dotaz jednomu z těchto serverů. Zde se situace opakuje a rekurzivní DNS je odkázán na servery dns1.idg.cz a dns2.idg.cz, které obsluhují doménu securityworld.cz. Výsledek vypadá takto:

```
; QUESTION SECTION:
;www.securityworld.cz.  IN A

;; AUTHORITY SECTION:
securityworld.cz. 18000 IN NS dns1.idg.cz.
securityworld.cz. 18000 IN NS dns2.idg.cz.

;; ADDITIONAL SECTION:
dns1.idg.cz.            18000 IN A 78.41.22.67
dns2.idg.cz.            18000 IN A 78.41.22.68
```

Opět je vyslán další DNS dotaz, tentokrát již na doménové servery domény securityworld.cz, tedy například na server dns2.idg.cz. Tento server již požadovanou informaci ví a našemu dotazujícímu se serveru odpoví:

```
; QUESTION SECTION:
;www.securityworld.cz.  IN A

;; ANSWER SECTION:
www.securityworld.cz.   43200 IN A 83.167.250.183
```

Nyní náš rekurzivní DNS server konečně dostal odpověď, kterou potřeboval, a na úplně původní dotaz stub resolveru může odpovědět s informací 83.167.250.183. Veškeré informace, které při získávání IP adresy pro server www.securityworld.cz dostal, si uloží do vyrovnávací paměti, včetně informace o TTL. Příště, pokud dostane stejný dotaz v časovém limitu, který je pro doménu www.securityworld.cz 43200 sekund neboli 12 hodin, bude moci odpovědět přímo z vyrovnávací paměti.

V tuto chvíli je potřeba se ještě na chvíli zastavit u sekce ADDITIONAL. V této sekci posílá autoritativní DNS tzv. GLUE záznamy. Představme si situaci, kdy kořenové nameservery pošlou odpověď: „nedokáži ti odpovědět, co je www.securityworld.cz, ale možná to ví a.ns.nic.cz.“ Pokud by v další sekci DNS zprávy nepřišly informace o IP adrese serveru a.ns.nic.cz, musel by rekurzivní DNS zjistit IP adresu serveru a.ns.nic.cz sám. Jak by nejspíš vypadala odpověď kořenového serveru na dotaz ohledně IP adresy a.ns.nic.cz? „Nedokáži ti odpovědět, ale možná to ví a.ns.nic.cz“. V tu chvíli bychom se dostali do nekonečné smyčky, a původní dotaz by zůstal nezodpovězen.

Formát DNS zpráv

Nyní se zanoříme ještě hlouběji do protokolu DNS a podíváme se jak vypadá samotný DNS protokol. DNS servery mezi sebou komunikují pomocí DNS zpráv, které mají vždy standardní formát. Tento formát je v nejvyšší úrovni rozdělený na pět základních sekcí:

Název sekce	**Popis**
Hlavička (Header)	Hlavička DNS zprávy
Dotaz (Question)	Dotaz pro DNS server
Odpověď (Answer)	Odpověď DNS serveru
Autorita (Autority)	Sekce ukazující na autoritativní servery
Další (Additional)	Sekce obsahující další záznamy

Hlavička je v DNS zprávě vždy přítomna a obsahuje informace o příznacích zprávy, návratovou hodnoty odpovědi zprávy, informaci o přítomnosti dalších částí a identifikátor transakce. Bohužel hlavička DNS zprávy má fixní formát a velikost, z čehož vyplývá první omezení DNS protokolu, tak jak byl definovaný v RFC1035 – hlavička DNS zprávy může obsahovat pouze omezené množství příznaků. Toto omezení lze naštěstí obejít, na což se podíváme na závěr tohoto článku.

V tuto chvíli udělám malou odbočku k příznakům, které jsou definovány v hlavičce DNS zprávy. Jsou to QR, AA, TC, RD a RA. QR (QueRy) je příznak, který určuje, zda-li je zpráva dotaz nebo odpověď. AA (Authoritative Answer) je příznak, který vrací autoritativní servery u odpovědí na dotazy, které vedly do zón, které obsluhují. Dotaz, který položíte rekurzivnímu serveru, by nikdy neměl obsahovat tento příznak. TC (TrunCation) je příznak, který označuje, že DNS zpráva byla zkrácena, a dotazující se má zeptat znovu přes TCP protokol. RD a RA jsou dva příznaky, které spolu souvisí. Příznak RD (Recursion Desired) posílá klient (například stub resolver), který se ptá rekurzivního serveru a vyžaduje od něj, aby provedl rekurzivní doptávání na jeho dotaz. RA (Recursion Available) je pak příznak, který posílá zpátky dotazovaný server, aby dal najevo, že je ochotný toto rekurzivní doptávání provést. Pokud se zeptáte serveru, který je pouze autoritativní, tak příznak RA nebude nastaven.

Další sekcí DNS zprávy je dotaz. Dotaz zprávy obsahuje tři části – QNAME, QTYPE a QCLASS. QNAME (z Query Name) obsahuje doménové jméno na které se dotazujeme. QNAME se skládá s jednotlivých labelů (label je vždy část mezi dvěma tečkami), resp. samotný label předchází jeho délka reprezentovaná jedním bajtem. Pro samotnou délku je využíváno pouze nižších 6 bitů a z toho vyplývá, že maximální délka

jednoho labelu je 63 bajtů (tj. samé jedničky na nižších 6 bitech). Celý QNAME je ukončen speciálním označením pro kořenovou zónu – labelem o délce 0. Maximální délka sekce je 255 bajtů včetně jejich bajtů s délkami labelů a včetně labelu pro kořenovou zónu. QTYPE a QCLASS odpovídají Typu a Třídě RR záznamu, nicméně jsou lehce rozšířeny o některé další hodnoty, např. 255 znamená: vrať mi libovolný DNS záznam (tzv. dotaz ANY).

Další tři sekce obsahují vždy pouze RR záznamy. Rozdíl mezi sekcí dotazu a ostatními je jen v zápisu – doménové jméno vlastníka záznamu je rozloženo na jednotlivé labely, tak jak to bylo rozebráno v sekci Dotaz pro QNAME a samotná data záznamu v DNS zprávě předchází jejich délka vyjádřená 16-bitový číslem.

Pro úsporu místa v DNS zprávě byl vymyšlen mechanismus komprese doménových jmen. Pokud délka labelu na horních dvou bitech obsahuje jedničky, pak dolních šest bitů neobsahuje délku, ale ukazatel na předchozí výskyt doménového jména. Protože je doménové jméno rozkouskováno na jednotlivé oktety, může jeho zápis v DNS zprávě začínat jedním nebo několika labely (např. „www“) a končit ukazatelem na předchozí výskyt (např. „dnssec.cz“).

DNS odpověď bude vždy obsahovat alespoň jeden RR záznam, nebo v hlavičce DNS zprávy v sekci návratového kódu bude obsahovat důvod, proč nemohl být dotaz vyřízen. Mezi nejčastější důvody je neexistence doménového jména (NXDOMAIN), odmítnutí dotazu (REFUSED) nebo chyba na straně serveru (SERVFAIL). Jednotlivé sekce budou naplněny podle toho, zda-li server posílá přímo odpověď (sekce Odpověď), posílá-li informaci o delegaci na jiné servery (sekce Autorita) nebo posílá-li např. IP adresy DNS serverů tzv. GLUE záznamy (sekce Další). DNSSEC následně tyto sekce využívá také pro informace o vlastních RR záznamech, ale k tomu se dostaneme v další sekci našeho seriálu.

Na závěr si ještě rychle řekneme něco o omezeních DNS protokolu. RFC 1035 jich definuje hned několik a některá jsme zmínili už v průběhu článku, jsou to:

- maximální délka labelu je 63 oktetů
- maximální délka doménového jména je 255 oktetů
- TTL je 32-bitové číslo

- pro zachování interoperability byl label striktně omezen na alfanumerické znaky a pomlčku
- maximální velikost UDP paketu je 512 oktetů
- počet použitelných příznaků v hlavičce DNS zprávy je omezený

Maximální velikosti labelu, doménového jména a TTL zůstaly zachovány do dnešních dní. TTL změnit asi už nepůjde a ani se nezdá, že by existovala potřeba toto pole zvětšit. Maximální délka labelu již také nejspíš zvětšit nepůjde. Nejméně problematická je maximální délka doménového jména. Toto omezení nevychází ze struktury DNS zprávy, ale bylo definováno, aby lidé, kteří implementují DNS protokol měli jednodušší práci (bavíme se o roce 1987). Dnes je toto omezení spíše historické. Velká část systémů má v sobě toto omezení „zadrátováno“ a změnit toto omezení by byl úkol téměř nadlidský.

Omezení znaků v labelu již dávno neplatí. Plně osmibitové znaky do něj sice stále vkládat nelze, ale např. SRV záznamy používané pro směrování různých služeb obsahují podtržítko na začátku záznamu a nikdo se nad tím nepozastavuje. Čistě teoreticky lze do DNS stromu ukládat libovolná binární data, protože návrh byl hodně obecný, nicméně prakticky to moc nebude fungovat.

Poslední dvě omezení – maximální velikost DNS zprávy posílané přes UDP a počet příznaků v hlavičce – byly vyřešeny ve standardu pojmenovaném Extension Mechanisms for DNS, zkráceně také EDNS0, který je popsán v RFC2671. Toto RFC definuje speciální typ RR záznamu pojmenovaný OPT, který se „schovává“ v sekci Další a přetěžuje jednotlivé pole RR záznamu takto:

Název pole	**Popis**
Vlastník	Vždy kořenová zóna
Typ	OPT
Třída	Maximální akceptovaná velikost UDP paketu
TTL	Rozšířený návratový kód a verze EDNS
RDATA	Rozšířené příznaky

Standard EDNS0 vznikl v roce 1999 a letos oslaví již dvanáct let. I proto je vcelku pozoruhodné, že více než 30 % DNS serverů, které se ptají autoritativních serverů pro domény .cz, nepoužívá EDNS0. Z hlediska technologie DNSSEC je podpora EDNS0 nutná pro samotné fungování DNSSEC, protože jsou využívány rozšířené příznaky i větší velikost DNS zpráv.

Securityworld, 3. června 2011

Dnešní část seriálu věnovaná DNS zprávě končí. Příště si už konečně povíme něco technologii DNSSEC.

O autorovi:

Ondřej Surý je vedoucím Laboratoří CZ.NIC, výzkumného a vývojového centra správce české národní domény