Vysoké učení technické v Brně, Fakulta Informačních technologií

Programování síťové služby – Filtrující DNS resolver

Síťové služby a aplikace

Monika Rosinská
18.11.2020

Obsah

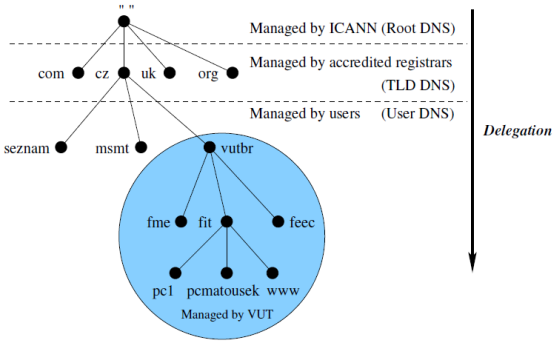
Uvedení do problematiky

Informace nastudované z literatury

Domain Name System DNS

DNS je systém sloužící pro vyhledávání a převod doménových jmen na shodnou IP adresu a naopak. Můžete si ho představit jako globální adresář všech serverů, počítačů a dalších zdrojů v síti.

Globální jmenný prostor adres a jejich mapování na IP adresy je hierarchicky rozdělen na několik částí a má strukturu invertovaného stromu. Strom se skládá z kořene (prázdný řetězec), domén prvního řádu (com, cz, …), domén nižšího řádu a z doménových jmen (seznam, idnes, …). Pokud chceme získat celé doménové jméno, postupujeme od odpovídajícího listu stromu až k jeho kořeni. Pro vyhledávání IP adresy se používá reverzní strom.



DNS protoksol

DNS protokol je aplikačním protokolem nad UDP/TCP. Používá se u dvou typů komunikace – DNS rezoluce a u přenosu zón.

DNS rezoluce se používá v případě, kdy klient vyhledává nějakou informaci. Klient položí serveru dotaz a ten mu odpoví. Server tak prohledává svůj stavový prostor, a to od kořene.

Přenos zón je synchronizace dat mezi primárním a sekundárním (záložním) serverem. Sekundární server si tak u primárního ověřuje, zda proběhla nějaká změna. Popřípadě existuje i varianta, kdy primární server pošle DNS notify, kterým vyzve druhý server k onomu dotazu.

	2 bytes	2 bytes	Example: answer to www.fit.vutbr.cz query
Header	Identification	Flags	59058 | qr, rd, ra, ad
	no. of questions	no. of answer RRs	1 | 1
	no. of authority RRs	no. of additional RRs	4 | 6
Question	Questions for the name server		www.fit.vutbr.cz IN A
Answer	RRs answering the question		www.fit.vutbr.cz 13963 IN A 147.229.9.23
Authority	RRs pointing towards an authority		fit.vutbr.cz. 13869 IN NS guta.fit.vutbr.cz. fit.vutbr.cz. 13869 IN NS kazi.fit.vutbr.cz fit.vutbr.cz 13869 IN NS rhino.cis.vutbr.cz fit.vutbr.cz 13869 IN NS gate.feec.vutbr.cz
Additional	RRs holding additional information		guta.fit.vutrbr.cz. 14062 IN A 147.229.8.11 kazi.fit.vutbr.cz. 13869 IN A 147.229.8.12 rhino.cis.futbr.cz. 86211 IN A 147.229.3.10 guta.fit.vutbr.cz. 14156 IN AAAA 2001:67c:1220:809 rhino.cis.vutbr.cz. 43692 IN AAAA 2001:67c:1220:e000:

DNS záznam

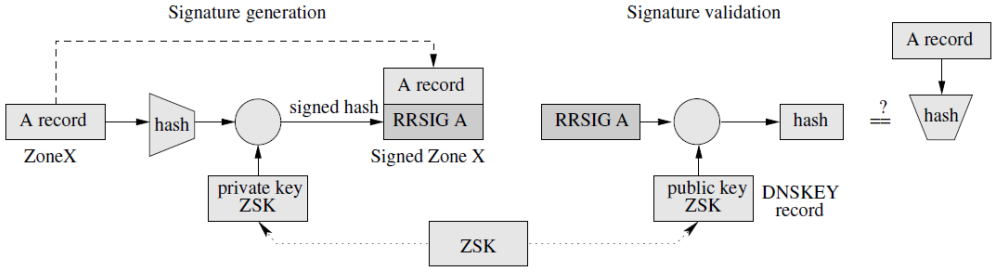
DNS záznamy obsahují jméno, typ záznamu, třídu, TTL (platnost záznamu), délku dat a samotná data. Součástí záznamu je i speciální záznam SOA, který kromě jména domény obsahuje další informace, jako kdy záznam expiruje, za jak dlouho je třeba kontrolovat změnu apod. Mimo to existuje několik záznamů, které se běžně používají a které obsahují další informace (např. A záznam, který převádí doménu na IPv4 adresu).

Resource Records Format	Example
Name (variable length)	email.fit.vutbr.cz
Type (16 bits)	CNAME
Class (16 bits)	IN (0x0001)
TTL (32 bits)	4106 (1 h 8 min 26 s)
RDLENGTH (16 bits)	10
RDATA (variable length)	hermina.fit.vutbr.cz

Zabezpečení DNS

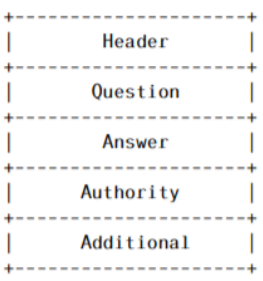
Jelikož se jedná o veřejný, distribuovaný systém potřebný pro komunikaci na internetu, je třeba zajistit integritu dat, autentizaci zdroje a šifrování. Zároveň je třeba zamezit podvržení odpovědi (útočník odpovídá místo serveru), podvržení dat v paměti cache a DoS útokům.

Pro zajištění integrity dat a autentizaci se používá DNSSEC.

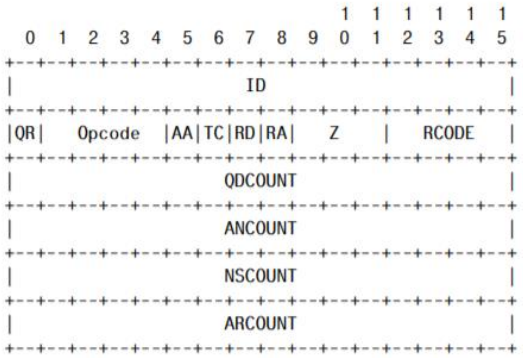


DNS packet

Struktura DNS packetu je následující:

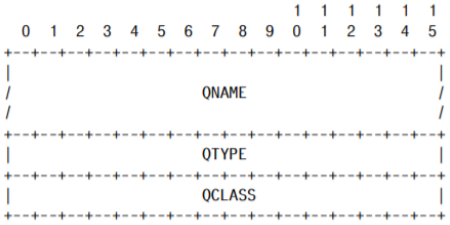


Hlavička (header) určuje typ packetu, která pole packet obsahuje, počet dotazů a odpovědí.



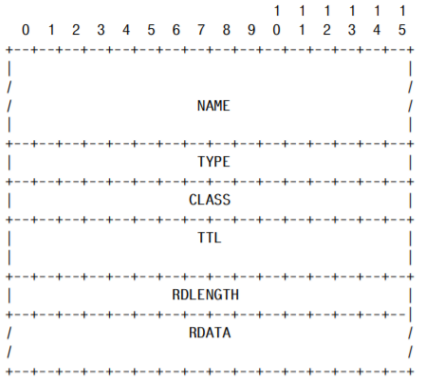
- ID pole - obsahuje ID dotazu
- QR - označuje zda se jedná o dotaz či odpověď
- Opcode – určuje typ dotazu; 4b
- AA – zda je odpověď autotirativní
- TC – zda bylo zpráva zkomprimována
- RD – zda je povolena rekurze
- RA – zda je zakázána rekurze
- Z – 3 rezervované bity
- RCODE – při odpovědi udává zda byla chybová zpráva a jaká (0 pokud neproběhl error, 1 pokud je zpráva špatného nebo neznámého formátu, 2 pokud server neodpovídá, 3 pokud dotazované doménové jméno neexistuje, 4 pokud server nepodporuje daný typ dotazu a 5, pokud server odmítne dotaz zpracovat – v projektu signalizuje zakázanou doménu)
- QDCOUNT – 16bit integer udávající počet dotazů
- ANCOUNT - 16bit integer udávající počet odpovědí
- NSCOUNT - 16bit integer udávající počet záznamů v autoritativní sekci
- ARCOUNT - 16bit integer udávající počet záznamů v neautoritativní sekci

DNS dotaz (question):



- QNAME – doménové jméno skládající se vždy z jednoho bytu (lable), který určuje počet oktetů, ze kterých se skládá daný úsek doménového jména; mezi tyto úseky se poté při zpracování „vkládá“ tečka (např. 6google3com -> google.com); konec domény se indikuje lable hodnotou 0
- QTYPE – 16b které určují typ dotazu
- QCLASS – 16b které určují třídu dotazu

DNS odpověď (answer)



- NAME – doménové jméno, na které se dotaz vztahoval
- TYPE - 16b určující kódování; určuje význam dat v RDATA poli
- CLASS – 16b určující kódování; určující třídu dat v RDATA poli
- TTL – doba (v sekundách), po kterou mohou být data cachována
- RDLENGTH – délka dat v RDATA
- RDATA – data odpovědi

Návrh aplikace

Aplikace je navržena tak, aby zpracovávala dotazy postupně (nikoli paralelně). Pokud přijde další dotaz ve chvíli, kdy zpracovává dotaz předchozí, dotaz si uloží do fronty.

Přijatý dotaz zkontroluje, a pokud je validní, odešle jej serveru, a následně přepošle odpověď tazateli. Jinak odešle tazateli odpověď s odpovídajícím chybovým kódem. Pokud server neodpovídá, tak program odešle odpovídající chybovou zprávu.

Program je členěn do logických celků (souborů). Jelikož je program psán v C++, měly by jednotlivé soubory obsahovat třídy. Jelikož jsem ale dříve psala jen v C, automaticky jsem třídy nepoužívala. Uvědomila jsem si to až pozdě, kdy jsem již nebyla schopna přepracování s jistotou stihnout.

Kromě souboru main, kde je jádro programu, je projekt členěn do následujících souborů:

- createConnection – obsahuje funkce pro vytváření spojení s klientem a serverem
- createResponse – obsahuje funkce pro vytvoření odpovědi v případě, že dotaz nebyl (kvůli jakékoli chybě) přeposlán serveru nebo v případě, kdy server neodpovídá
- filterDomain – obsahuje funkce pro filtrování zakázaných domén
- processArgs – obsahuje funkci na zpracování a kontrolu argumentů
- processRequest – obsahuje funkce pro zpracování dotazu, získání domény atd.

Implementace

Po spuštění programu jsou nejdříve zpracovány argumenty – zadaný DNS server, soubor se zakázanými doménami a případně i číslo portu, na kterém má soubor zachytávat komunikaci. Program kontroluje jak to, zda je server i soubor zadán, ale ověřuje, zda některý z parametrů není zadán víckrát. U DNS serveru zároveň zkontroluje, zda je daný server validní (tedy že taková stránka vůbec existuje).

Následuje načtení souboru se zakázanými doménami. Pokud se soubor nepodaří otevřít, program skončí s odpovídajícím upozorněním. Pokud je soubor prázdný (či neobsahuje žádnou doménu), program uživatele upozorní, ale neukončí se. Při procházení se odstraňují prázdné řádky a komentáře, zadané domény se uloží do seznamu.

Poté proběhne připojení k zadanému serveru a následně ke klientovi (na portu 53 nebo na zadaném portu). Pokud se připojení nezdaří, program skončí s odpovídajícím upozorněním.

Jelikož jsou některé porty rezervovány, nemusí se připojení podařit. V tom případě program vypíše upozornění, že v takovém případě je nutné program spustit jako správce.

Následně začne samotné odposlouchávání na zadaném portu. Příchozí dotaz se uloží do bufferu, zkontroluje se délka dotazu, a poté se předá ke zpracování. Jelikož všechny DNS dotazy mají stejnou strukturu, a umístění typu dotazu je až za doménovým jménem, jehož délka je proměnná, se nejdříve zjistí doménové jméno. Pokud v průběhu zpracování doménového jména dojde k chybě, program zpracovávání ukončí a pošle odpověď s chybou 1, což označuje nevalidní data.

Pokud zpracování doménového jména proběhne v pořádku, ověří se typ dotazu, a pokud se jedná o dotaz typu A, pokračuje se filtrováním domén.

Filtrování domén

Domény se filtrují pomocí while cyklu, který postupně porovnává dotazovanou doménu s doménami v seznamu (list). Pokud daná doména není nalezena, odstraní se první subdoména, respektive nalezne se první tečka v doméně, a část před ní se odstraní (např. z images.google.com se stane google.com) a filtrování proběhne znovu. Tento cyklus se opakuje, až není nalezena další tečka (tedy všechny subdomény a domény nižšího řádu jsme již odstranili).

Pokud není nalezena shoda dotazované domény s žádnou zakázanou doménou, dotaz se přepošle DNS serveru. Jinak se sestaví odpověď s chybou 5, což značí odmítnutí zpracování požadavku.

```
bool checkDomain(string domainName, list<string> bannedDomains){
    size_t pos; //position of dot (to remove subdomain)
    do{
        //go through list with banned domains
        for(string bannedDomain : bannedDomains){
            if (domainName == bannedDomain){
                return false;
            }
        }
        pos = domainName.find_first_of('.'); //get next dot in domain name
        pos++; //move to next char
        domainName = domainName.substr(pos); //split domain at first dot and save second part
    } while(pos != 0); //in pos will be 0 if no dot is found

    return true;
}
```

Pokud je dotaz přeposlán DNS serveru, program čeká na odpověď. Zde by ovšem mohl program uváznout, neboť server nemusí odpovídat (ať už z důvodu, že chybí připojení k internetu, server je zaneprázdněn nebo uživatel zadal špatnou adresu serveru). Proto je u spojení se serverem, respektive v socketu, aktivován timeout. Po vypršení timeoutu je čekání na odpověď přerušeno a tazateli je vrácena odpověď s návratovou chybou 2 značící nezastihnutí serveru.

Pokud odpověď od serveru obdrží, přepošle ji zpět tazateli a začne zpracovávat další požadavek.

Základní informace o programu

Program slouží k filtrování nežádoucích domén při dotazu na jejich přeložení.

Program zachytává komunikaci na zadaném portu. Zpracovává jen dotazy typu A, ostatní dotazy zahazuje. Pokud je dotaz typu A, zkontroluje požadovanou doménu. Pokud je tato doména zakázána, program to sdělí uživateli. Jinak je tento dotaz přeposlán určenému DNS serveru a odpověď je přeposlána původnímu tazateli.

Více informací se nachází v předchozích sekcích.

Návod na použití

Program je třeba přeložit příkazem *make* v daném adresáři.

Po přeložení se program spustí příkazem *dns* s parametry:

-s název_serveru

-f název_(cesta)_souboru_se_zakázanými_doménami

volitelně: –p číslo_portu (udává, na jakém portu má program zachytávat komunikaci)

Příklady spuštění:

./dns –s 8.8.8.8 –f file.txt –p 1100

./dns –s dns.google. –f ./ISA/ban.txt

./dns –s one.one.one.one –f ./ISA/ban.txt –p 84

Pokud není zadán parametr –p, automaticky se číslo portu nastaví na 53.

Pokud je v parametrech nějaká chyba, program se ukončí s odpovídajícím popisem.

Chybové zprávy

- odpověď s RCODE 1 – zaslaná data nejsou validním DNS dotazem
- odpověď s RCODE 2 – DNS server neodpovídá
- odpověď s RCODE 4 – nepodporovaný typ dotazu
- odpověď s RCODE 5 – zakázaná doména
- Can't create socket. – nezdařený pokus o vytvoření socketu
- Bind failed. – nezdařený pokus o napojení k socketu; obsahuje i dodatečné informace o možných důvodech selhání
- Connection failed. – nezdařený pokus o připojení k DNS serveru
- Invalid address. Address is not supported. – pokus o připojení k serveru selhal
- Error during opening file. – otevření souboru se zakázanými doménami se nezdařilo
- Please insert only one DNS server. – uživatel zadal více DNS serverů
- DNS server is invalid. – jméno DNS serveru se nezdařilo přeložit
- Please insert only one port number. - uživatel zadal více portů
- Port number is invalid. – zadané číslo portu není validní
- Please insert only one file with banned domains. - uživatel zadal více souborů se zakázanými doménami
- Unknown parameter. – uživatel zadal neznámý parametr
- Please insert file with banned domains. – uživatel nezadal soubor se zakázanými doménami
- Please insert DNS server. - uživatel nezadal DNS server
  - Upozornění: Note that file is empty. No domains will be filtered.\n Execute program if you want to use another file. – nevede k zastavení výstupu, vypisuje se na standartní výstup

Zvolená řešení při chybějící specifikaci

Více dotazovaných domén v jednom dotazu

Struktura DNS packetu umožňuje zadat více dotazovaných domén do jednoho dotazu. Jelikož ale není v zadání uvedeno, jestli s touto možností máme počítat, rozhodovala jsem se podle skutečných DNS serverů.

Při zkoumání jsem zjistila, že ačkoli je takové dotazování možné, žádný z velkých serverů jej nepodporuje a odpovídá pouze na první dotaz. Z toho důvodu jsem se rozhodla kontrolovat jen první dotazovanou doménu. Ostatní dotazované domény zodpovězeny totiž nebudou.

Komprimace

Existuje i možnost packety komprimovat – využití právě například u dotazování na více domén. Rozhodla jsem se komprimaci nepodporovat.

Literatura

vlastní poznámky z přednášek předmětu ISA na VUT FIT z roku 2020/2021 od Grégr Matěj, Ing., Ph.D. a Matoušek Petr, Ing., Ph.D., M.A.

https://www2.cs.duke.edu/courses/fall16/compsci356/DNS/DNS-primer.pdf?fbclid=IwAR1GHELCBjbVcxHuGu2deJCwpzA0nE0FwEjsCTW0IhisZI0GYzspxPpdMOY (18. 11. 2020)

http://www-inf.int-evry.fr/~hennequi/CoursDNS/NOTES-COURS_eng/msg.html?fbclid=IwAR1MuJT0jeJcHG3HPkl1f72-QHTzjbXv9JG1PLVuimQqXbhFJwGDNcnJjNY (18. 11. 2020)