«باسمه تعالی»

گزارش کار پروژه‌ی اول درس شبکه‌های کامپیوتری

آشنایی با DNS

طراحی و تدوین:

مهدی رحمانی

9731701

«بخش اول: سوالات تشریحی»

(1) از پروتکل DNS چه استفاده ای میشود؟

ابتدا یک مقدمه میگوییم : ما انسانها راه های مختلفی برای شناسایی یکدیگر داریم: نام شناسنامه ای، شماری ملی، و شمارهی گواهینامهی رانندگی می توانند هویت هر یک از ما را به صورت منحصر به فرد مشخص کنند. اگر چه هر یک از این شناسه ها می توانند برای شناسایی افراد به کار روند، در هر شرایط مشخص یکی از آنها مناسب‌تر از بقیه است.

در مورد host های اینترنت هم مانند انسانها روشهای مختلفی برای شناسایی وجود دارد. یکی از این روش ها، استفاده از نام میزبان (host) میباشد. به خاطرسپردن نام‌هایی مانند google.com و www.yahoo.com برای ما راحت تر است اما این نام‌ها هیچ اطلاعاتی درباره‌ی مکان دقیق این host ها در اینترنت به دست نمی‌دهند. از طرفی router های اینترنت با پردازش نام host مشکل دارند چون طول این نام‌ها میتواند متغیر باشد. به همین دلیل برای شناسایی host های اینترنت از شناسه‌ای موسوم به آدرس IP استفاده میشود.

آدرس IP یک عدد 4 بایتی با ساختاری سلسله مراتبی است. آدرس های IP به صورت چهار عدد که با نقطه از یکدیگر جدا شده اند، نمایش داده می شوند مانند: 121.7.106.83 هر یک از این چهار عدد می تواند مقداری بین 0 تا 255 بگیرد. سلسله مراتبی بودن آدرس IP به این دلیل است که هر چه از چپ به راست برویم، اطلاعات دقیق‌تری از مکان host صاحب این آدرس در اینترنت به دست می آوریم.

حال به سراغ اصل مطلب میرویم: پروتکل DNS مانند FTP HTTP و SMTP ، یک پروتکل لایه‌ی کاربرد است چون : 1) با استفاده از مدل client-server بین دو سیستم انتهایی ارتباط برقرار می کند، و 2) برای انتقال پیامهای DNS بین این سیستم ها به پروتکل های انتقال لایه های زیرین متکی است. با این حال، از جنبه های دیگر نقش DNS با کاربردهایی مانند وب، انتقال فایل و ایمیل بسیار تفاوت دارد، چون بر خلاف آن برنامه ها کاربر هرگز ارتباط مستقیم با DNS ندارد. به جای آن، DNS یک عملکرد کلیدی اینترنت را در اختیار برنامه های کاربردی اینترنت می گذارد. این عملکرد که وظیفه‌ی اصلی DNS میباشد اینگونه است که میتواند نام‌های host را به آدرس‌های IP ترجمه کند.

DNS علاوه بر تبدیل نام host به آدرس IP چند سرویس مهم دیگر نیز ارایه می کند:

Host aliasing(نام مستعار host): گاهی نام یک میزبان بسیار طولانی و پیچیده می شود، که در این صورت تعریف یک نام مستعار (alias) برای آن می تواند مناسب باشد. برای مثال، اگر نام یک میزبان چیزی مانند: relayl.west-coast.enterprise.com باشد(که به آن canonical hostname میگویند) میتوان به

کمک DNS دو نام مستعار مانند enterprise.com و www.enterprise.com برای آن تعریف کرد که به خاطر سپردن آن ها آسان‌تر است. برنامه های کاربردی می توانند علاوه بر آدرس IP برای به دست آوردن نام مستعار host نیز از DNS کمک بگیرند.

Mail server aliasing(نام مستعار سرویس دهنده‌ی پست): به دلایل مشابه، داشتن نام‌های مستعار ساده برای سرویس دهنده های پست نیز بسیار مطلوب است. برای مثال، اگر باب عضو سرویس ایمیل هات میل شود، آدرس ایمیل او چیزی ساده مانند bo@hetmail.o خواهد بود. با این حال، نام میزبان سرویس دهنده‌ی پست هات میل معمولا به سادگی hotmail.com نیست، و نام متعارف آن می تواند چیزی مثل relayl.west-coast. hotmail.com باشد. یک mail application میتواند به کمک DNS برای یک نام مستعار میزبان، canonical hostname آن را نیز(همانطوری که IP address آن را میابد) پیدا کند. (درواقع رکورد MX از DNS اجازه میدهد تا company's mail server وWeb server یک hostname مستعار یکسان داشته باشند.)

Load distribution(توزیع بار): برای توزیع بار بین replicated servers ، مانند replicated Web servers ، نیز می توان از DNS کمک گرفت. سایت های پر بازدید مانند yahoo.com معمولا از چندین replicated server استفاده می کنند، که این server ها در مناطق مختلف دنیا پراکنده اند و آدرس های IP متفاوتی دارند. در replicated server ها چند آدرس IP مختلف با یک alias hostname متناظر هستند. پایگاه داده‌ی DNS همه ی این آدرس های IP را در اختیار دارد، و زمانی که یک client این نام host را پرس و جو می کند، سرور DNS فهرستی از تمامی آدرس های IP آن به مشتری بر می‌گرداند. ولی در هر پاسخ ترتیب این آدرس ها را می چرخاند. از آنجا که client معمولا درخواست HTTP خود را به اولین درس IP این فهرست می فرستد، چرخاندن ترتیب آدرس ها توسط DNS باعث توزیع بار بین replicated serverها خواهد شد. DNS همین کار را برای mail server ها نیز می تواند انجام دهد، به طوری که چندین server می توانند دارای یک نام مستعار واحد باشند.

(2) رکوردهای مختلف DNS را نام ببرید و هر یک را به صورت مختصر توضیح دهید.

سرورهای DNS یک رکورد DNS ایجاد می کنند تا اطلاعات مهم در مورد یک دامنه یا hostname ، به ویژه آدرس IP فعلی آن را ارائه دهند.

سرویس دهنده های DNS همگی با هم یک پایگاه داده‌ی توزیع شده DNS را پیاده سازی می کنند که رکوردهای مرجع (Resource Record-RR) شامل رکوردهای نگاشت hostname به IP، را در خود ذخیره میکند. هر پیام پاسخ DNS یک یا چند رکورد مرجع را در خود حمل می کند. هر رکورد مرجع از چهار فیلد زیر تشکیل شده است:

(Name, Value, Type, TTL)

فیلد TTL طول عمر رکورد مرجع را مشخص می کند، یعنی زمانی که این رکورد باید از حافظه‌ی نهان سرورهایی که آن را ذخیره کرده اند، پاک شود. در مثال های انواع رکوردها که آمده این فیلد را نادیده میگیریم. معنای فیلدهای Name وvalue به فیلد Type بستگی دارد.(برخی تایپ های معروف در ادامه معرفی شده اند)

- اگر **Type=A**، آنگاه Name همان hostname است و Value همان IP address مربوط به آن hostname است. بنابراین، رکورد نوع A همان نگاشت استاندارد hostname به IP address است. برای مثال، رکوردی به شکل (A , 147.37.93.126, relay1.bar.foo.com) یک رکورد نوع A میباشد. در ضمن در این منظور از آدرس همان آدرس IPv4 میباشد.
- اگر **Type=AAAA** ، آنگاه این رکورد همانند رکورد نوع A میباشد با این تفاوت که آدرس IP که استفاده میشود همان IPv6 یک دامنه میباشد.
- اگر **Type=NS**، آنگاه Name همان نام دامنه (مانند foo.com) و Value درواقع همان hostname مربوط به یک authoritative DNS server است که میداند چگونه به آدرس‌های IP مربوط به host های این دامنه برسیم. از این رکورد برای هدایت پرس و جوهای DNS در زنجیره‌ی پرس و جو(query chain) استفاده میشود. درواقع به صورت خلاصه میتوان گفت این رکورد مشخص‌کننده‌ی DNS server معتبری است که می‌تواند به درخواست‌های DNS مربوط به یک دامنه‌ی خاص و بعضی زیردامنه‌های آن پاسخ بدهد. برای مثال، رکوردی به شکل (NS , dns.foo.com , foo.com) یک رکورد نوع NS است.
- اگر **Type=CNAME**، آنگاه Value همان canonical hostname مربوط به alias hostname (نام مستعار) Name است. این رکورد می تواند نام متعارف(canonical name) یک host را در اختیار یک پرسو جو کننده یا درواقع query قرار دهد. به عبارت دیگر از رکورد cname برای هدایت اتومات یک نام به نام دامنه دیگر استفاده می‌شود. شناخته شده‌ترین رکورد CNAME همان www می‌باشد

که www.yourdomain.comرا به آدرس yourdomain.com ارجاع می‌دهد و باعث می‌شود هر دو آدرس یک محتوا را نمایش دهند. یا مثلا (foo.com, relay1.bar.foo.com, CNAME) هم یک رکورد نوع CNAME میباشد.

- اگر **Type=MX** آنگاه Value همان canonical name مربوط به یک mail server با alias hostname (نام مستعار) Name است. برای مثال رکوردی به شکل (MX , mail.bar.foo.com , foo.com)یک رکورد نوع MX است. این رکورد اجازه می دهد تا mail server ها نام‌های مستعار ساده داشته باشند. به عبارتی میتوان گفت که این رکورد ایمیل ها را به یک mail server مخصوص هدایت میکند. توجه کنید که با این رکورد به شرکت‌ها اجازه میدهد از یک نام مستعار واحد همزمان برای mail serverو برای یکی از سرورهای دیگرش مانند Web Server استفاده کنید. اگر یک DNS client بخواهد canonical name یک mail server را بداند باید برای رکورد MX آن دامنه درخواست دهد؛ برای یافتن canonical name سرورهای دیگر باید برای رکورد CNAME درخواست دهد

همچنین تایپ های دیگری نیز در ادامه معرفی شده‌اند:

- **TXT** : این رکورد اجازه می‌دهد تا متن دلخواه خود را به رکورد DNS اضافه کنید. معمولاً این رکورد به سایر سرویس ها درباره عمل‌کرد دامنه اطلاعاتی را می‌دهد و اجازه میدهد که یک متن دلخواه را به یک دامنه متناظر کنیم.
- **PTR** : رکورد ptr که یک رکورد DNS معکوس نامیده می‌شود، یک آی پی را به یک آدرس دامنه ارجاع می‌دهد. دقیقا برعکس همان کاری که رکورد A انجام می‌دهد. در واقع این رکورد یک اتصال صحیح بین دامنه و آی پی برقرار می‌کند تا درخواست‌ها اشتباها به سرورهای دیگر ارسال نشود.
- **SOA** : SOAیا به اختصار Start Of Authority رکوردی است که ضروری ترین بخش از یک فایل zone را تشکیل میدهد. SOA رکوردی است که به مدیران Domain اطلاعات پایه ای از دامنه را می‌دهد نظیر : چه زمان هایی Update میشود ، زمان آخرین آپدیت چه وقت بوده ، آدرس ایمیل ادمین و غیره.توجه کنید که هر فایل zone شامل یک رکورد SOA میباشد. آپدیت شدن SOA بین Authoritative Name Server ها به خوبی در افزایش پهنای باند و بهینه سازی آن نقش موثری دارد. و همچنین در افزایش سرعت دسترسی به وب سایت ها هنگامی که حتی DNS سرور اصلی Down شود یا از کار بیفتد را نیز بر عهده دارد.
- **SRV** : رکورد SRV مشخص‌کننده‌ی هاستی پشتیبانی‌کننده از یک سرویس خاص است. به بیان ساده اگر کاربری (اپلیکیشنی) درخواست دسترسی به یک سرویس خاص را برای سرور DNS ارسال کند، در پاسخ برای آن رکورد SRV حاوی نام دامنه و شماره پورتی که سرویس روی آن فعال است، ارسال می‌شود. SRV

با استفاده از یک پورت مقصد خاص ، یک دامنه را به نام دامنه دیگر می‌نگارد. رکورد SRV اجازه می دهد سرویس های خاص مانند VOIP یا IM به مکان جداگانه هدایت شوند.

- **SPF**: رکورد SPF یکی از رکوردهای DNS است که با مشخص کردن فهرستی از سرورهای مجاز به ارسال ایمیل به یک دامنه، سبب کاهش فعالیت‌های اسپم می‌شود .یکی از حملات رایج در دنیای اینترنت IP Spoofingاست که در آن مهاجم با تغییر آدرس مبدا یک پکتIP ، از آن پکت برای اهداف خرابکارانه‌ی خود استفاده می‌کند. برای نمونه، ممکن است مهاجم ایمیلی را برای یک دامنه ارسال کند که به نظر از یک دامنه‌ی مطمین ارسال شده است، ولی در عمل این‌گونه نیست .وظیفه‌ی رکوردSPF ، جلوگیری از بروز چنین اتفاقاتی است. در این رکورد می‌توان مشخص کرد که چه سرورهایی مجاز به ارسال ایمیل به یک دامنه هستند. به این ترتیب، سرویس ارایه‌دهنده‌ی ایمیل آن دامنه با بررسی رکورد SPF تشخیص می‌دهد که آیا ایمیل دریافتی از یک مرجع معتبر است یا خیر.

در زیر نام تایپ تعداد بیشتری از رکوردها که در کد پروژه استفاده شده است، آمده است. لازم به ذکر است که تعداد این تایپ ها حتی بیشتر از این میباشد که در برخی سایت ها مانند ویکی پدیا میباشد.

```
types = ["ERROR", "A", "NS", "MD", "MF", "CNAME", "SOA", "MB", "MG", "MR", "NULL", "WKS", "PTR", "HINFO",
         "MINFO", "MX", "TXT", 'RP', 'AFSDB', 'X25', 'ISDN', 'RT', 'NSAP', 'NSAP-PTR', 'SIG', 'KEY', 'PX',
         'GPOS', 'AAAA', 'LOC', 'NXT', 'EID', 'NB', 'NBSTAT', 'ATMA']
```

(3) DNS server چیست و آدرس سه مورد از معروفترین DNSها server را نام ببرید.

جهت اینکه یک دامنه بتواند به وب سایت اشاره نماید در قدم اول بایستی در DNS Server اضافه شود. اما کار dns server چیست ؟ DNS سرور یک دیتابیس بزرگ است که شامل مجموعه ای از domain ها و IP های مرتبط می باشد. به عنوان مثال اگر دامنه google.com باشد سایت به IP با آدرس 64.233.167.99 مرتبط می شود. درواقع ها سرورهایی هستند که به Query های DNS پاسخ می‌دهند. همچنین می توانید سرور DNS را به عنوان دفترچه تلفن در نظر بگیرید. وقتی از رایانه خود می خواهید یک وب سایت بارگیری کند ، سرور DNS نام وب سایت را با آدرس IP مناسب مطابقت می دهد. این اجازه می دهد تا رایانه شما آن را به درستی پیدا و بارگیری کند.

DNS سرور های زیادی در شرکت های هاستینگ و سازمان‌ها وجود دارد. این DNS Server ها با یکدیگر در ارتباط هستند. بنابراین تنها کافی است شرکت هاستینگ شما نام دامنه شما را در سرور dns اضافه نماید تا تدریجا (در حدود ۴۸ ساعت) با سایر DNS ها در سراسرجهان هماهنگ شود.

برای حل مشکل مقیاس پذیری(مثلا طراحی متمرکز داشته باشیم که یک DNS Server تمامی نگاشت‌های موردنیاز کاربران سراسر دنیا را انجام دهد و خب اینطوری مشکلات زیادی باتوجه به زیاد بودن query ها در اینترنت امروز به وجود می‌آید)، DNS از تعداد زیادی سرورکه به صورت سلسله مراتبی سازماندهی شده و در سرتاسر دنیا توزیع شده اند، استفاده کند. هیچ کدام از این سرور های DNS به تنهایی نگاشت نام - آدرس تمامی hostهای موجود در اینترنت را در خود ندارند، و این نگاشت‌ها در بین سرور های DNS کل دنیا توزیع شده است. در اولین تقریب، سه طبقه سرور DNS در اینترنت وجود دارند: سرویس دهنده های DNS ریشه، سرویس دهنده های DNS دامنه های سطح بالا (موسوم به TLD)، و سرویس دهنده های DNS سرپرستی.(البته یک دسته‌ی دیگر هم گاها ازش نام برده میشود به نام Recursive Resolver .)

در حالت کلی این روند انجام میشود که تابع تحلیل‌گر بعد از تلاش برای ترجمه نام‌دامنه به صورت محلی(Local) ، اگه قادر به ترجمه نام‌دامنه نبود مجبور به ارتباط با DNS Server میشه. پس تابع یک بسته درخواستی (Query Packet)به صورت UDP تشکیل میده و به DNS Server ارسال می‌کنه و منتظر بسته دریافتی (Response Packet)می‌مونه. تا IP مورد نظر رو بگیره.

آدرس 3 مورد از معروف‌ترین DNS server ها در زیر آمده است:(همچنین IPv4 DNS address مربوط به آن‌ها نیز روبرویشان ذکر شده است)

- Cisco **OpenDNS**: 208.67. 222.222 and 208.67. 220.220;
- Cloudflare 1.1. 1.1: 1.1. 1.1 and 1.0. 0.1;
- **Google Public DNS**: 8.8. 8.8 and 8.8. 4.4; and.
- Quad9: 9.9. 9.9 and 149.112. 112.112.

(4) پورت پیشفرض مورد استفاده در پروتکل DNS چیست؟

پورت پیشفرض مورد استفاده در پروتکل DNS هم برای TCP و هم برای UDP پورت 53 میباشد.

پروتکل DNS از پورت 53 به منظور ارائه خدمات خود استفاده می نماید . بنابراین یک سرویس دهنده DNS به پورت 53 گوش داده و این انتظار را خواهد داشت که هر سرویس گیرنده ای که تمایل به استفاده از سرویس فوق را دارد از پورت مشابه استفاده نماید .

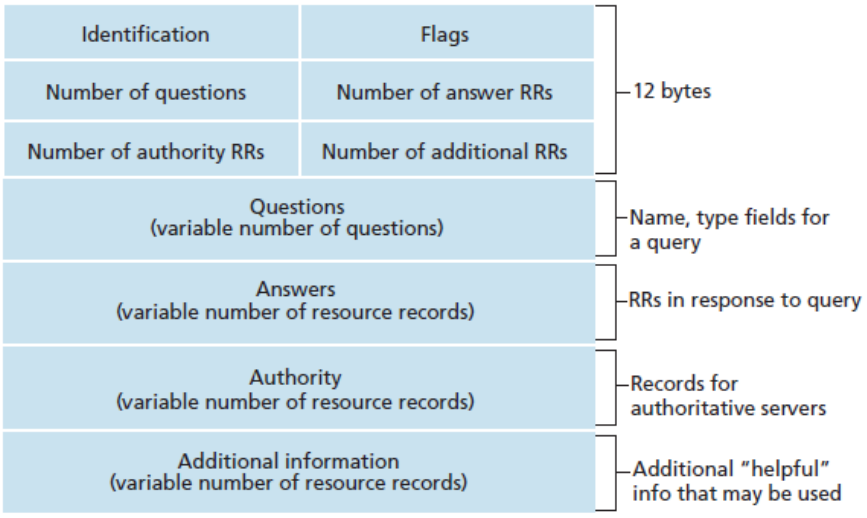
(5) ساختار بسته های DNS به چه شکل می باشد؟

به صورت کلی بسته‌های DNS ساختاری به شکل زیر دارند:

DNS Message Format

Header	Information about the message
Question	Question for the name server
Answer	Answer(s) to the question
Authority	Pointers to other name servers
Additional	Additional information

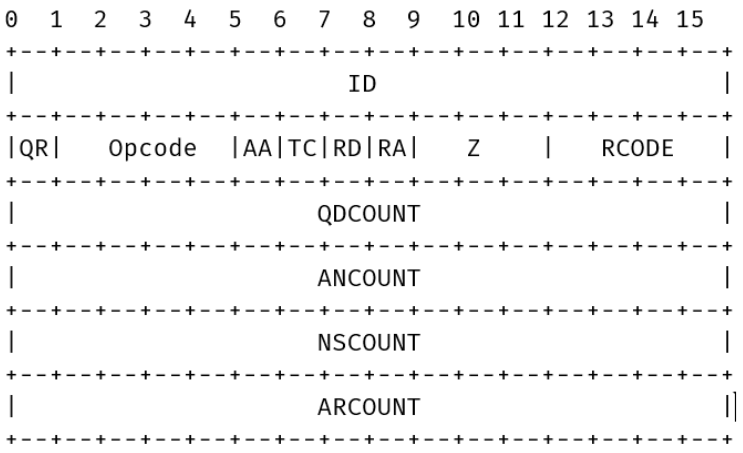
همچنین به صورت دقیق تر میتوان در شکل زیر نیز مشاهده کرد:



حال به بررسی دقیق‌تر هر قسمت میپردازیم:

Header :

Headerدر بستهDNS ، نوع بسته و بخش های اون رو توصیف میکند این قسمت 12 بایت اول بسته را تشکیل میدهد. قسمت هاب مختلفی دارد که در ادامه توضیح مختصری راجع به هرکدام میدهیم:



ID : این فیلد 16 بیتی میباشد. در یک ماشین ممکن هست در لحظه چند درخواست در حال بررسی باشد. درون این فیلد یک شناسه قرار میگیرد. این شناسه توسط ماشین درخواست‌دهنده ساخته میشود. این مقدار در سرور DNS ذخیره میشه و در بسته پاسخ (Response) هم همان مقدار درج میشود تا ماشین درخواست‌دهنده بتواند تشخیص دهد که کدام پاسخ (Response) برای کدام درخواست (Query) هست. این فیلد به نام TXID هم شناخته میشود.

QR: این فیلد 1بیتی میباشد. درون این فیلد نوع بسته مشخص میشود که میتواند از نوع درخواست (Query)یا پاسخ (Response) باشد . معنی و مفهوم مقادیر به شرح زیر است:

صفر: بسته از نوع درخواستی (Query) است.

یک: بسته از نوع پاسخ (Response) است.

Opcode : این فیلد 4 بیتی میباشد. این فیلد چهار بیتی مشخص‌کننده نوع بسته‌ی درخواست هست. مقادیر این فیلد به شرح زیر میباشد:

- صفر: بسته یک درخواست استاندارد و ساده دارد.(Standard query)
- یک: بسته درخواست معکوس دارد .(Inverse query) به این معنی است که درخواست‌دهنده قصد دارد نام‌دامنه رو با توجه به آدرس IP پیدا کند.
- دو: یعنی درخواست‌دهنده قصد دارد وضعیت سرور را چک کند.(Server status request)
- چهار: این مقدار به این معنی هست که یک سرور DNS اصلی(Master) ، به DNS فرعی (Slave) خودش هشدار میدهد که اطلاعات عوض شده است و باید خودش را بروزرسانی کند.(Notify)

- پنج: به این معنی هست که سرور DNS قصد بروزرسانی داردو به اصطلاح zone transfer شکل می‌گیرد.
- سه و شش تا پانزده: بدون کاربرد هستند و برای استفاده های احتمالی رزرو شده اند. البته درRFC های جدید تر از این مقادیر استفاده شدهاست.

AA : این فیلد 1 بیتی میباشد. مشخص‌کننده نوع پاسخ هست و مقادیر آن به شرح زیر هست:

- یک: به این معنی است که پاسخ Authoritative است.
- صفر: یعنی non-Authoritative است.

TC : این فیلد 1 بیتی میباشد. این مقدار مشخص کننده این هست که آیا بسته کوتاه شده است یا خیر. وقتی که حجم بسته درخواستی بیشتر از حجم پشتیبانی شده باشد مقدار این فیلد یک میشود. در TCP ما محدودیتی برای حجم نداریم ولی در UDP حداکثر مقدار 512 بایت هستش. در این مواقع ممکن هست که Clientنیاز به یک TCP Session داشته باشد.

RD: این فیلد 1 بیتی میباشد. مقدار این فیلد مشخص میکند که آیا برای ترجمه، پرسش و پاسخ از سرورهای دیگه صورت گرفته یا نه (بازگشتی بودن پاسخ). مقادیر این فیلد هم به شرح زیر هست:

- صفر: ارتباطی با دیگر سرورها صورت نگرفته.(Recursion not desired)
- یک: برای ترجمه با دیگر سرورها ارتباط صورت گرفته.(Recursion desired)

RA : این فیلد 1 بیتی میباشد. مقدار این فیلد مشخص کننده این هست که آیا سرور از بسته های درخواست بازگشتی (Recursive query) پشتیبانی میکند یا خیر. که مشخصا مقدار یک به معنی پشتیبانی کردن از این نوع درخواست هست.

Z : این فیلد 3 بیتی میباشد. این فیلد بی استفاده هست و برای استفاده های احتمالی رزرو شده، و مقدار آن همیشه صفر میباشد.

RCODE : این فیلد 4 بیتی میباشد. برای خطایابی در نظر گرفته شده است. به این صورت که در بسته درخواستی مقدار آن صفر تعیین میشود و سرور پاسخ دهنده آن را با توجه به وضعیت تغییر میدهد. مقادیر استفاده شده در این فیلد به شرح زیر هست:

- صفر: اگر بدون هیچ مشکلی درخواست پاسخ داده شود این مقدار صفر باقی میماند.

- یک: به این معنی هست که یک خطای ساختاری رخ داده (Format Error) و سرور نمیتواند بسته درخواستی را بخواند.
- دو: این مقدار یعنی سرور مشکل دارد و قادر به پاسخ‌گویی نیست.(Server Failure)
- سه: به معنی این هست که نام دامنه وجود ندارد.(Name Error)
- چهار: نوع بسته درخواستی توسط سرور پشتیبانی نمیشود.(Not Implemented)
- پنج: به این معنی هست که سرور درخواست را رد کرده است که اغلب به علت سیاست های امنیتی رخ میدهد.(Refused)

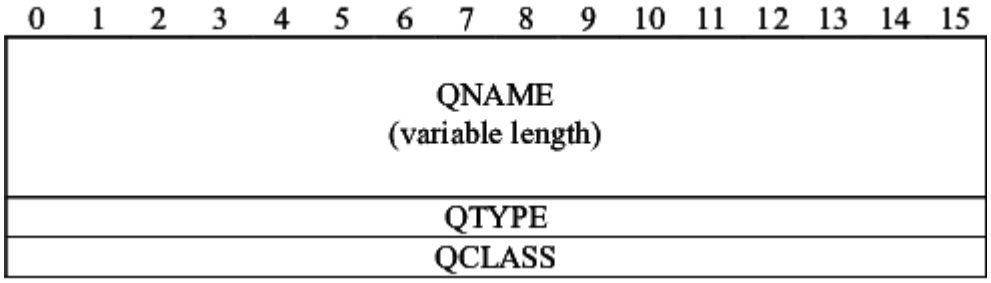
QDCOUNT : 16 بیتی میباشد. این فیلد تعداد درخواست های بسته را مشخص میکند.

ANCOUNT : 16 بیتی میباشد. این فیلد تعداد پاسخ های موجود در بسته را مشخص میکند.

NSCOUNT : 16 بیتی میباشد. مشخص کننده تعداد پاسخ های authoritative در بسته هست.

ARCOUNT : 16 بیتی میباشد. تعداد پاسخ های فرعی در بسته را مشخص میکند.

Question :



قسمت پرسش ها حاوی اطلاعاتی درباره‌ی پرس و جوی در دستِ انجام است. این قسمت شامل 3 بخش است: 1) یک فیلد «نام» حاوی نام مورد پرس و جو، و 2) یک فیلد «نوع» که نوع پرسش مورد نظر برای این نام را مشخص می کند،و 3) یک فیلد «کلاس» کلاس query موردنظر را به ما میدهد.

Answer :

در پیام پاسخ سرویس دهنده DNS قسمت جواب‌ها حاوی رکوردهای مرجع یافت شده برای پرس و جوی متناظر است. به یاد دارید که هر رکورد مرجع شامل فیلدهای Type (مانند CNAME، NS، A و...) و Value و TTL است. از آنجا که یک hostname می تواند چندین آدرس IP داشته باشد، پیام پاسخ می تواند حاوی چند رکورد مرجع باشد.

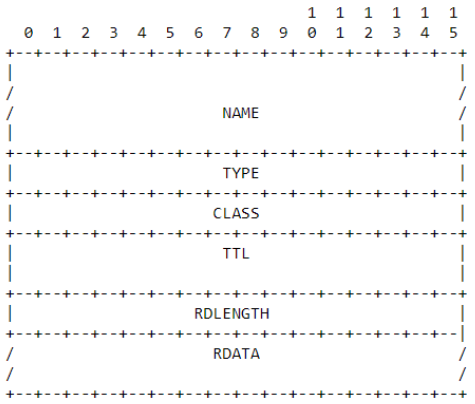
Authority:

قسمت سرپرستی شامل رکوردهای دیگر سرورهای سرپرستی است.

Additional Information :

سرویس دهنده های DNS رکوردهای سودمند دیگر را در قسمت اطلاعات اضافی پیام پاسخ قرار می دهند برای مثال، فیلد جواب در پیام پاسخ یک پرس و جوی MX حاوی رکورد مرجعی است که canonical name مربوط به mail server مورد نظر را در خود دارد. سرور DNS می‌تواند نگاشت نام - آدرس این‌host (که یک رکورد نوع A است) را نیز در قسمت اطلاعات اضافی به client برگرداند.

3 قسمت آخر یعنی Answer و Authority و Additional دارای فرمی به صورت زیر میباشند:



https://memoryleaks.ir/how-dns-protocol-works/

https://www.ietf.org/rfc/rfc1035.txt

(6) دلیل توصیه RFC برای استفاده از پروتکل UDP در Query ها نسبت به TCP چیست؟

DNS یک پروتکل لایه کاربرد است. تمام پروتکل های لایه‌ی کاربرد از یکی از دو پروتکل لایه انتقال یعنی UDP و TCP استفاده می کنند. TCP قابل اعتماد است و UDP قابل اعتماد نیست. قرار است DNS قابل اعتماد باشد ، اما از UDP استفاده می کند .

حقایق جالب زیر در مورد TCP و UDP در مورد لایه انتقال وجود دارد که موارد فوق را توجیه می کند و دلیل استفاده از UDP و عدم استفاده از TCP را به صورت پیشفرض میگوید:

1) سرعت UDP بسیار بیشتر است. TCP کند است زیرا به 3-way handshake نیاز دارد. بار(load) سرورهای DNS نیز فاکتور مهمی است. سرورهای DNS (از آنجا که از UDP استفاده می کنند) مجبور نیستند connection را حفظ کنند.

2)همچنین UDP سربار و overhead کمتری دارد.

3) درخواست های DNS به طور کلی بسیار کوچک بوده و به خوبی در سگمنت های UDP قرار می گیرند.

4) UDP قابل اطمینان نیست ، اما قابلیت اطمینان می تواند به application layer اضافه شود. یک برنامه کاربردی می تواند از UDP استفاده کند و قابل اطمینان باشد به این صورت که از وقفه زمانی (timeout) استفاده کند و در لایه کاربرد مجدداً ارسال کند.

در واقع ، DNS در درجه اول از پروتکل (UDP) در پورت شماره 53 برای ارائه درخواست ها استفاده می کند. نمایش داده شد DNS شامل یک درخواست UDP از client و به دنبال آن، یک پاسخ UDP از سرور است. وقتی طول پاسخ بیش از 512 بایت باشد و هر دو سرویس گیرنده و سرور از EDNS پشتیبانی می کنند ، از بسته های UDP بزرگتر استفاده می شود. در غیر این صورت ، دوباره query با استفاده از پروتکل (TCP) ارسال می شود. TCP همچنین برای کارهایی مانند zone transfers (انتقال رکوردهای DNS از سرور DNS اصلی به ثانویه) استفاده می شود. درواقع در این حالت از پروتکل (TCP) به جای UDP استفاده می شود تا یکپارچگی داده ها را بتواند چک کند.

https://www.geeksforgeeks.org/why-does-dns-use-udp-and-not-tcp/

(7) سوکت چیست؟

به شکل ساده، سوکت ترکیبی از پورت و IP آدرس است. به تعبیر تخصصی تر، سوکت نقطه انتهایی یک ارتباط دو طرفه بین دو برنامه در حال اجرا در شبکه است. سوکت به یک عدد پورت متصل میشود تا لایه TCP شبکه بتواند برنامه موردنظر برای ارسال اطلاعات را تشخیص دهد . به تعبیر ساده تر، کار سوکت ایجاد این کانال است. از طریق کانال ارتباطی ایجاد شده توسط سوکت، داده هایی در طول شبکه ارسال و دریافت میشوند. زبانی که دو برنامه به وسیله آن از این کانال با هم مکاتبه میکنند نیز پروتکل نام دارد.

: