



دانشکده‌ی مهندسی کامپیوتر

آذر ۱۳۹۷

CE-40695

پیاده سازی یک شبکه P2P

استاد: مهدی جعفری

در این تمرین یک DNS Server^۱ پیاده‌سازی می‌کند. این پروتکل روشی توزیع‌شده برای مربوط کردن اسم‌ها به آی‌پی‌ها است. برنامه‌ی شما باید روی پورت ۵۳۵۳ برای درخواست‌ها گوش کند و به هر درخواست پاسخ‌دهد. برای هر درخواست برنامه‌ی شما باید یک فایل متنی هم تولید بکند. جزئیات هر بخش در ادامه توضیح داده می‌شوند.

۱ مقدمه

۲ مقدمه

۱.۲ Stream

در تابع constructor این کلاس ابتدا با متد `is_valid` فرمت های IP و Port را چک میکنیم تا مطمئن شویم به صورت همان فرمت مورد نظر هستند. `server_in_buf` همان بافری هست که روی سرور نوشته میشود و هر چند وقت یک بار بایستی چک شود. `Callback` Function (cb) نیز پیام های جدید را به ته `server_in_buf` میچسباند (`append`) و در نهایت Ack برمیگرداند. این باعث میشود هر جا که سوکتی وسط کار قطع شود بفهمد قطع شده است. سپس `tcpserver` را مسازیم، در یک `thread` قرار میدهم و آن را اجرا میکنیم. `self.nodes` تمام نود هایی هستند که درون ما هستند. `get_server_address` آدرس سرور را با آن فرمتی که میخواهیم به ما میدهد. `clear_in_buf` بافر سرور را پاک میکند. `add_node` نود اضافه میکند. `remove_node` نود مشخص شده را از آرایه پاک میکند و سپس متد `close` نود را اجاره میکند. `get_node_by_server` آی پی و پورت سرور یک نود را میگیرد و نود را برمیگرداند. سپس با `parse` کردن آن را به فرمت مد نظر تبدیل میکند.

`add_message_to_out_buffer` یک آدرس میگیرد و با `read_in_buf` وظیفه دارد `read_in_buf` را برگرداند.
`send_message_to_node`
`send_out_buf_message`

۲.۲ Node

۳ مقدمه

برنامه‌ی شما باید درخواست‌های معمولی را که شامل یک آدرس هستند بپذیرد و بدون توجه به بیت RD^۲ ttl^۳ ۳۶۰۰ هستند. قسمت RDATA هم همان آی‌پی است.

در این نوع درخواست‌ها باید همیشه از آی‌پی برای وصل شدن به سرورها استفاده کنید و استفاده از آدرس متنی سرورها نمره‌ای ندارد. نوع دیگری از درخواست‌ها، درخواست‌های برعکس هستند. یعنی در این درخواست‌ها آی‌پی به شما داده می‌شود و باید آدرس مربوط به آن

^۱ Domain Name Server
^۲ Recursion Desired

آی پی را پیدا کنید.

نوع معمول این درخواست‌ها به این شکل است که اگر آی پی به شکل a.b.c.d باشد باید به شکل d.c.b.a.in-addr.arpa در پکت باشد و شما هم در هنگام فرستاده کوثری به سرورها باید از همین فرم استفاده کنید اما در هنگام بررسی کد شما آی پی‌ها به طور معمولی (همان a.b.c.d) به شما داده می‌شوند و باید آن‌ها را تغییر دهید به سرور بفرستید.

فیلد Opcode باز هم صفر است (هم در پکتی که به کد شما فرستاده می‌شود و هم در پکتی که شما برای سرور می‌فرستید) ^۳ بقیه فیلدها را مانند قسمت قبل مشخص کنید. در این قسمت باز هم اگر جوابی از نوع A بود از آن و اگر چند تا بودند از کم‌ترین مقدار استفاده کنید ولی اگر هیچ رکوردی از نوع A نبود در بین رکوردهای NS کم‌ترین را (با در نظر گرفتن ترتیب دیکشنری) انتخاب کنید. البته این رکوردها متنی هستند و آی پی نیستند ولی استثناً در این قسمت وصل شدن به اسم مانعی ندارد. در نهایت هنگامی که رکوردی از نوع PTR پیدا شد که فیلد name در آن برابر با آی پی آدرس درخواست شده (بعد از تبدیل فرمت) بود آن را به آدرسی که درخواست زده بود برگردانید. فیلدهای name و rdata در پاسخ را برابر با همانی که DNS Server به شما داد، قرار دهید سایر فیلدها را هم مانند بالا تعیین کنید. فقط دقت شود که این دفعه پاسخ از نوع PTR است.

۱.۳ لاگ

برای هر درخواستی که به کد شما زده می‌شود باید یک فایل متنی با فرمت txt. بسازید و اسم آن را هم برابر با Message ID درخواست قرار دهید. فرمت این فایل به شکل زیر است: در هر زمانی که به یک آی پی وصل می‌شوید باید خط زیر را چاپ کنید:

```
connecting to [IP]
=====
```

بعد از دریافت هر پکت از طرف DNS Server ها باید آن پکت را هم در این فایل چاپ کنید که فرمت آن توضیح داده می‌شود. (این کار برای پکت که به عنوان درخواست به کد شما زده می‌شود نباید انجام شود). هر پکت دارای ۵ بخش است Header ، Question ، Answer ، Authority و Additional . در شروع هر کدام از این قسمت‌ها باید اسم به علاوه ۱۵ علامت مساوی در فایل بنویسید. در پایان هر قسمت هم باید ۱۵ تا مساوی چاپ کنید. که به ترتیب در پایین آمده‌اند.

```
HEADER
=====
QUESTION
=====
ANSWER
=====
AUTHORITY
=====
```

توجه شود پکت‌های هر قسمت باید بعد از اسم آن قسمت و علامت‌های مساوی نوشته شوند. فرمت نوشتن پکت‌ها در ادامه توضیح داده می‌شود. برای هر قسمت باید در بین در آکولاد اسم فیلدها و مقدارشان را به ترتیب الفبایی اسم فیلدها و در هر خط یک فیلد، در فایل بنویسید. قسمت Header باید به شکل زیر باشد: (مشخصاً مقادیر فرضی هستند و شما باید بر اساس پکت دریافتی آن‌ها را بنویسید.)

```
{
additional count : 14
answer count : 0
authority count : 13
id : 38145
is authoritative : False
is response : True
is truncated : False
opcode : 0
question count : 1
recursion available : False
recursion desired : False
```

^۳ درست‌تر این بود که هر دو یک باشند اما این حالت کار می‌کند و مد نظر است.

```
reserved : 0
response code : No Error
}
```

اسم فیلدها و فرمت خروجی باید دقیقا مانند بالا باشد. در ادامه سایر قسمت‌ها هم بررسی می‌شود و خروجی شما باید از لحاظ ساختار و ترتیب دقیقا مانند آن‌ها باشد.
برای فیلد response code مقادیر عددی را با توجه به قالب پایین به متن تبدیل کنید.

```
0: No Error
1: Format Error
2: Server Failure
3: Name Error
4: Not Implemented
5: Refused
```

برای قسمت Question به شکل زیر عمل کنید.

```
{
Domain Name : bing.com
Query Class : 1
Query Type : 255
}
```

ساختار قسمت‌های باقی‌مانده مانند هم است پس فقط یکی از آن‌ها مثلا Authority را بررسی می‌کنیم. در هر کدام از این قسمت‌ها تعدادی پاسخ وجود دارد که هر کدام از آن‌ها می‌توانند از انواع زیر باشند:

```
A
NS
CNAME
SOA
PTR
MX
AAAA
TXT
```

همه‌ی این انواع یک ساختار مشترک به شکل زیر دارند:

```
{
class : ans_class
name : name
rdata : rdata
rdlength : rdlength
ttl : ttl
type : ans_type
}
```

برای قسمت type باید اعداد را به اسامی بالا تبدیل کنید مگر این که عدد جزو تایپ‌های بالا نباشد که در آن صورت عدد را در فایل بنویسید. اما قسمت rdata برای هر کدام فرق می‌کند که در ادامه توضیح داده می‌شوند.
در نوع A این فیلد یک آدرس آی‌پی است که باید به شکل آی‌پی چاپ شود.

```
rdata : 30.83.55.192
```

در نوع AAAA این فیلد یک آدرس آی پی ۶ است که باید به شکل آی پی ۶ چاپ شود.

```
rdata : 2001:0503:a83e:0000:0000:0000:0002:0030
```

توجه شود جواب را مخفف نکنید و به شکل ۸ عدد ۴ رقمی در مبنای ۱۶ که با «:» جدا شده اند نشان دهید
نوع MX

```
rdata :  
{  
Mail Exchanger : mta6.am0.yahoodns.net  
Preference : 1  
}
```

نوع SOA

```
rdata :  
{  
Admin MB : hostmaster.yahoo-inc.com  
Expiration Limit : 1814400  
Minimum TTL : 600  
Primary NS : ns1.yahoo.com  
Refresh interval : 3600  
Retry interval : 300  
Serial Number : 2017101218  
}
```

انواع cname و ptr و ns مانند هم هستند و فقط شامل یک رشته هستند.

```
rdata : ns4.yahoo.com
```

نوع TXT هم شامل یک رشته است اما این رشته به شکل معمولی ذخیره شده است در سایر انواع این رشته یک encoding خاص دارد که می توانید از لینک هایی که در پانویشت صفحه ی اول هستند مطالعه کنید

```
rdata : #v=spf1 redirect=_spf.mail.yahoo.com
```

انواع دیگری از پکت های جواب هم موجود هستند که برای آن ها لازم نیست rdata را پارس کنید و بسیار کم کاربرد هستند و برای آن ها کافی است در فایل خروجی rdata را بنویسید و جلوی آن را خالی بگذارید.

```
rdata :
```

و در مقابل فیلد type هم مقدار عددی نوع آن ها را چاپ کنید. به عنوان مثال اگر از نوع WKS بود به شکل زیر عمل کنید:

```
type : 11
```

در نهایت به عنوان مثال فایل اگر ورودی به کد شما آی پی 198.41.0.4 بود و پکت درخواستی برای yahoo.com و با آی دی

موفق باشید