



دانشکده‌ی مهندسی کامپیوتر

شبکه‌های کامپیوتری

مدرس: مهدی جعفری

Socket Programming (UDP)

گردآورندگان: بهراد احمد پور - عرشیا اخوان

شنبه ۲۰ آبان ۱۴۰۲

۱ DNS

تو این تمرین می‌خواهیم که یک DNS Resolver را پیاده‌سازی کنیم. باید کدی بنویسید که روی پورت ۵۳۵۳ گوش کرده و پاسخ DNS Query هایی که برای A class هستند را بدهد. برنامه باید وقتی شروع به اجرا می‌کند محتویات فایل /etc/myhosts را خوانده و از روی آن به DNS Query پاسخ بدهد. فرمت این فایل هم مثل فرمت فایل /etc/hosts است.

۱.۱ مقدمه

همونطور که می‌دونید، DNS سرویسی هست که برای پیدا کردن IP Address دامنه استفاده می‌شود. به Request هایی که به یک سرور DNS ارسال می‌شود، Query و به پاسخ‌هایی که DNS Server می‌فرستد، Answer می‌گویند. توی این پروتکل، هر Query یک class دارد که نشون می‌دهد چه سوالی را راجع به آن دامنه می‌پرسیم، چندان از این class ها عبارتند از: A, AAAA, CNAME, MX, TXT. توی این تمرین می‌خواهیم که DNS Resolver پیاده‌سازی کنیم که پاسخ Query های کلاس A را بدهد.

۲.۱ دست‌گرمی

برای این‌که ببینید یک DNS Query چطور کار می‌کند، می‌تونید از ابزارهایی مثل dig و nslookup استفاده کنید. مثلا command زیر، IP دامنه google.com را بر می‌گرداند:

```
dig google.com
```

اختیاری: در گام نخست IP چند دامنه را با استفاده از dig یا nslookup بدست بیارید. در گام بعدی، تلاش کنید dig را مجبور کرده برای پیدا کردن IP فقط از یک سرور خاص استفاده کند.

۲ پیاده‌سازی

ابتدا باید پورت ۵۳۵۳ سیستم روی پروتکل UDP گوش کنید. به صورت سنتی، DNS Resolver روی پورت ۵۳ پاسخ می‌دهند ولی این پورت Reserve سیستم عامل است و ما دنبال دردرس نیستیم!

معمولا سرورها لیستی از دامنه‌ها و IP هاشون را دارند، پس سرور ما هم به همچین چیزی نیاز دارد. وقتی برنامه شروع می‌شود، از فایل /etc/myhosts این لیست را بخوانید. می‌توانید فرض کنید که فرمت این فایل با فرمت فایل /etc/hosts یکی است ولی فقط داخلش فرمت IPv۴ داریم و خبری از IPv۶ نیست.

سپس باید پکت‌های UDP که به ما میرسد را Parse کنید، Query را از آن استخراج کنید، پاسخ Query را از لیست دامنه‌هایی که اول کار Load کردید پیدا کنید، و در نهایت Answer را بسازید و روی Connection که داشتیم بفرستید. بهتر است که قبل از دست به کد شدن آشنایی خوبی به هدرهای پکت DNS و نحوه‌ی چیدمان Query و Answer توی این Packet داشته باشید، برای این کار RFC 1035 نقطه شروع (و احتمالا پایان) خوبیه!

در آخر هم حواستون باشه که ما تو ۲۰۲۳ زندگی می‌کنیم و توی این دوره، همه‌چی باید parallel باشد! پس حواستون باشد که کدتون توانایی پاسخ دادن به چند Query به صورت همزمان را داشته باشد.

۳ تحویل دادنی ها

فرمت فایل هایی که تحویل می دهید باید به این شکل باشد:

۱. فایل README که نام، نام خانوادگی، شماره دانشجویی و توضیحات کد می باشد.
۲. یک فایل Makefile که پروژه ی شما رو بیلد می کنه. در صورتی که از dependency خاصی استفاده می کنید حتما آن را در Makefile خود نصب کنید.
۳. کدهاتون را می تونید با هر زبانی بنویسید.
۴. توی کدهاتون نباید از هیچ گونه Package یا Library خارجی ای برای Parse کردن و ساختن پکت های DNS استفاده کنید. کدهاتون موقع نمره دهی خونده می شود، برای همین اگه در استفاده از Package شک داشتید، حتما از TA بپرسید، (البته بهتره کلا از Package غیر std استفاده نکنید.)

۴ تست و ارزیابی

۱. تست صحت: در این تست با ابزارهای DNS مثل dig و nslookup به سرور شما Query می زنیم و پاسخی که ابزار دریافت می کند را بررسی می کنیم که هم مطمئن شویم پروتکل DNS را درست پیاده سازی کردید، هم مطمئن شویم که IP درست را از روی /etc/myhosts برمی گردونید.
۲. تست همروندی: یک بار، ۱۰۰ Query رو به صورت Sequential به سرور شما می فرستیم و میانگین Response Time را بدست می آوریم. سپس ۱۰۰ Query به صورت Parallel به سرورتون می فرستیم و انتظار داریم که زمان اجراش از قسمت Sequential کمتر باشه!

موفق باشید