

گزارش پروژه پایانی

دانیال علی عظیمی

حسنا بشیریان

در این پروژه هدف پیاده سازی شبکه اشتراک گذاری فایل مانند bittorrent است. به طور کل در این پروژه دو نوع موجودیت در نظر گرفته شده است:

- Tracker
- Peer

در اینجا tracker همانند نامی که دارد وظیفه نگهداری از اطلاعات درباره فایل ها و peer هایی که فایل ها را در اختیار دارند. در این پیاده سازی tracker فقط اطلاعات درباره Port های peer هایی که یک فایل را میتوانند آپلود کنند را نگهداری میکند. یعنی اگر یک peer بخواهد اطلاعاتی درباره chunk های در دسترس بدست آورد باید به خود node ها پیام دهد تا این اطلاعات را بدست آورد. نمونه ای از transaction های یک tracker:

```
Do you wish to continue master ?uploader 1223 added for file FinalProject.pdf
{"ports": [1223], "size": 9}
send message to receiver:{"ports": [1223], 'size': 9} to 1225
port 1223 has been remove from FinalProject.pdf list
uploader 1239 added for file FinalProject.pdf
```

برای سادگی در پیاده سازی فرض شده است که فقط یک فایل در هر لحظه توسط peer آپلود میشود(میتواند موازی آپلود کند به چند نفر)

روند دریافت فایل:

در شروع اول به tracker پیام مناسب را بفرستیم و درخواست اطلاعات درباره فایل میکنیم(peer ها) پس از این tracker اطلاعات port های شامل و اندازه فایل را به ما میفرستد. سپس به هر کدام از port ها پیام داده میشود و chunk های مورد نیاز با chunk های موجود مقایسه میشود و اگر chunky باشد که ما بخواهیم آن را از او دانلود میکنیم اگر هم نه از port دیگری می پرسیم. این روند را تا تکمیل شدن تمامی فایل ادامه داده می دهیم. اگر port به ما جواب ندهد یک پیام به tracker برای update کردن وضعیت میدهیم.

```
{"name": "FinalProject.pdf", "type": "receive"}
1223
{"name": "FinalProject.pdf", "type": "list_chunks"}
message {'chunks': [0, 1, 2, 3, 4, 5, 6, 7, 8]}
{"name": "FinalProject.pdf", "type": "get_chunk", "chunk_number": 0}
{"name": "FinalProject.pdf", "type": "get_chunk", "chunk_number": 1}
{"name": "FinalProject.pdf", "type": "get_chunk", "chunk_number": 2}
{"name": "FinalProject.pdf", "type": "get_chunk", "chunk_number": 3}
{"name": "FinalProject.pdf", "type": "get_chunk", "chunk_number": 4}
```

روند آپلود فایل :

با فرض موجود بودن فایل پس از وارد شدن دستور مناسب به tracker وضعیت خود را اعلام میکنیم و سپس منتظر دریافت request از peer های دیگر برای فایل میمانیم اگر request دریافت شود در داخل یک thread جدید آن را به handler مناسب میدهیم.

در داخل handler منتظر پیام از سمت peer هستیم پیام ها از دو نوع هستند :

- Get available chunks
- Get chunk number n

در اول لیستی از chunk هایی که ما در حال حاضر در دسترس داریم ارسال میشود و در حالت دیگر داده یک chunk را دریافت میکنیم.

دقت شود که تمامی ارسال ها به format json انجام می شوند و قبل از ارسال داده با استفاده از الگوریتم رمزنگاری caesar cipher رمز میشود و ارسال میشود و در سمت دریافت کننده پیام decrypt میشود. الگوریتم با استفاده از شیفت دادن حروف الفبا به یک اندازه ثابت رمزنگاری را انجام میدهد و برای بازگشت کافی است به معکوس این مقدار شیفت انجام دهیم برای مثال :

Shift amount = 10

encrypt(data,10)

decrypt(data,26 -10)

دلیل 26 بخاطر انگلیسی بودن حروف است.

برای اینکه بتوان این الگوریتم را استفاده کرد نیاز بود تغییر کوچک در پیاده سازی آن بدهیم تا بتواند اعداد utf-8 را قبول کند.

```
('127.0.0.1', 1223)
message {'name': 'FinalProject.pdf', 'type': 'get_chunk', 'chunk_number': 8}
{"type": "chunk", "chunk": ""}
torrent -setmode IDLE FinalProject.pdf
FinalProject.pdf
exiting
{"type": "EXITING", "port": 1223, "file": "FinalProject.pdf"}
torrent -setmode send FinalProject.pdf
FinalProject.pdf
```