

FINAL RAPORU

Arif Doğru 141044014

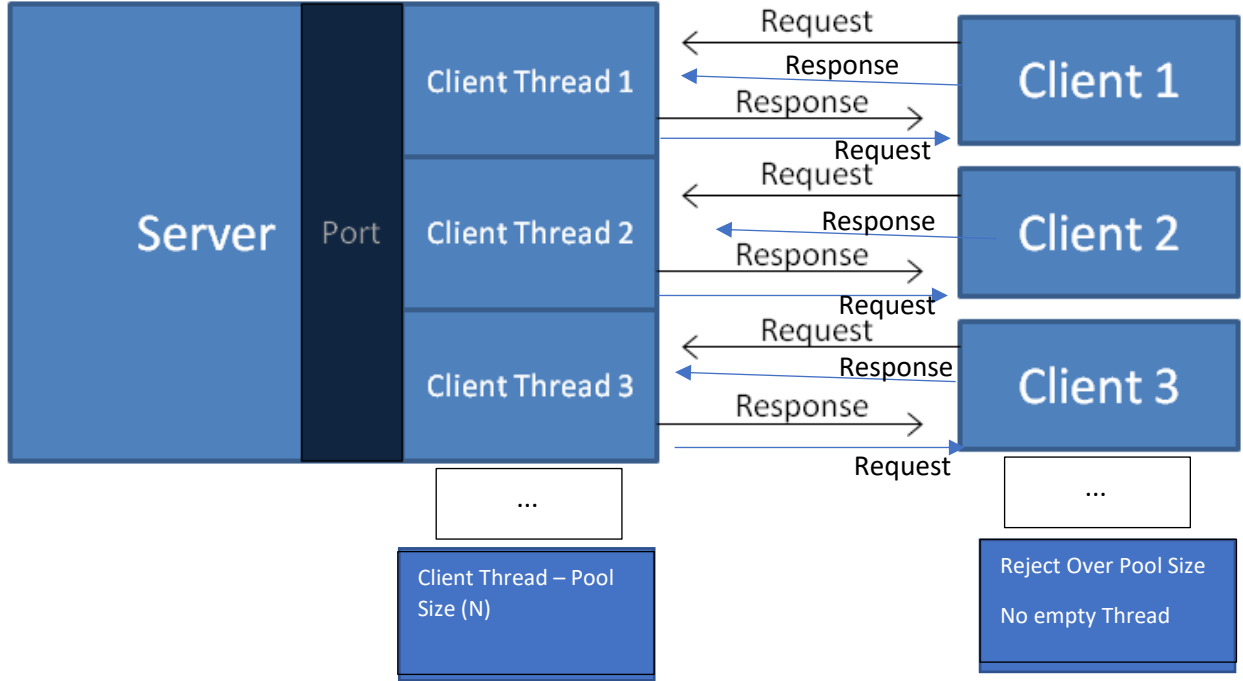
GİRİŞ

Bu proje Server.c , Client.c Makefile dosyalarından oluşmaktadır.

Kendi bilgisayarlarımda yaptığım demoyu [Youtube Profilim](#) 'den izleyebilirsiniz.

TASARIM

Multi-thread temelli projede Server-Client modeli tasarladım.



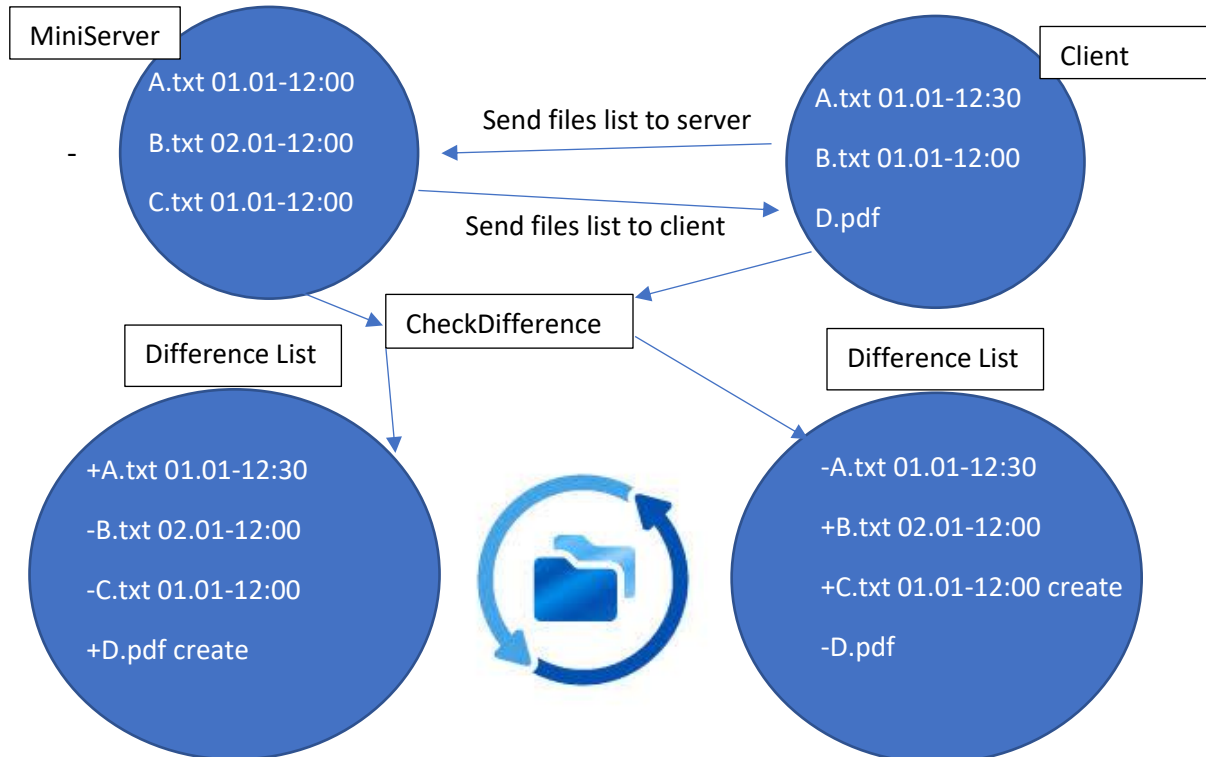
Bu modele göre Server arguman olarak verilen [thread pool size] kadar bağlantı isteğini Kabul eder. Boşta thread kalmayınca kadar gelen isteklere karşılık verilir. Ancak clientların yedeklemek istedikleri dosya pathleri aynı olmamak kaydıyla. **Bundan sonrası için bağlı clientlardan herhangi biri sonlanırsa ona hizmet veren thread boş çıkacağı için yeni bağlantı isteği kabul edilebilir hale gelir.**

SERVER:

Server çalışması için: ./Server [Directory] [Thread Pool Size] [Port]

Parametreleri girilerek süreç başlatılır.

- İlk adımda server socket oluşturur ve girilen porttan bağlantı olmasını bekler
- Bağlantı gelince client tarafından yollanan path aktif olarak senkronize edilmiyorsa ve boşta thread varsa thread e yönlendirilir.
- Main server yeni istek gelene kadar portu beklemeye devam eder.
- Miniserver thread gelen client ile ilgilenmeye başlar.
- **Miniserver ile client arasındaki haberleşme standardı öncelikle her ikisi de birbirine her şeyin yolunda olup olmadığını teyid eden flag gönderir. (\$:Okey , !:Danger)**
- İki tarafta karşılıklı bu flagleri control edip kendilerinin sonlanıp sonlanmayacağına karar verdikten sonra bir birilerine ellerindeki dosya isimlerini ve son değiştirilme tarihlerini yollarlar.
- CheckDifference adlı fonksiyonum ile her iki tarafta da bir birilerine alınıp verilecek difference listesi oluşturdum. Bunun mantığı aşağıda şematize edilmiştir.



Bu difference listesi sayesinde iki tarafta yapılan herhangi bir deęişiklik doğrudan karşıya taşınmaktadır.

- Bir tarafta dosya var dięer tarafta dosya yok ise olmayan tarafta create edilir.
 - Bir taraftaki dosya tarihi dięerinden eski ise yeni tarihli dosya dięer tarafa yazılır.
 - Dosya alışverişleri tamamlanınca tekrar karşıya flag yollanıp okunur ve aynı süreç yaşanır.
 - Server yapılan tüm bu deęişiklikleri MainLog.log adlı log dosyasına kaydeder.
 - Log dosyası server'a arguman olarak girilen directory altında oluşturulur.
 - Ayrıca her baęlı client için de baęlandığı path altında path_log.log dosyası oluşturulur.
 - Bu log dosyası ise sadece o path ile ilgili deęişiklikleri kayıt altına alır.
 - CTR-C yani SIGINT sinyali geldiğinde server en başta anlatılan haberleşme standard altında her threadine client a göndermesi için "!" flagini yollar.
 - Threadlerde signal maskelenmiştir yalnızca main server signal handler çalıştırır.
 - Threadler de yukarıda renkli olarak ifade edilmiş olan maddedekini uygular ve clientlara "!" flag yollamış olur.
 - Daha sonra mini server threadler aldıkları resource'ları geri vererek sonlanır.
 - Main server da pthread_join bloęunda threadlerin sonlanmasını bekler.
 - Threadler join olduktan sonra socketler , File descriptorlar ve file pointerlar kapatılır.
 - Server mesaj basıp sona erer.
- Server Life Cycle sonlanmış olur**

CLIENT:

Client çalışması için: ./Client [directory] [ip address] [port] olarak başlatılır.

- İlk adımda verilen ip adresindeki porta baęlantı sağlamaya çalışır.
- Eğer server ayakta deęilse başarısız olur ve hata mesajı ile sonlanır.
- Server ayaktaysa karşıya senkronize olmak istedięi path gönderir eęer bu path zaten bir client tarafından senkronize olmaya çalışılıyorsa hata mesajı ile sonlanır.
- Baęlanmak istedięi server da boş thread yoksa yine hata mesajı ile sonlanır.
- Bu aşamalardan başarısız olmamış bir client karşı tarafa pathi iletmiş ve artık haberleşme standard adımına geçmiştir.
- Yine server tarafındaki gibi öncelikle her şeyin yolunda olup olmadığını belirten flag yollayıp alır ve sorun varsa sonlanır.
- Sorun yoksa server a kendi pathindeki dosyaların isimlerini gönderir.

- Karşılığında serverdaki dosya isimlerini alır ve check difference fonksiyonuyla nelerin alınacağını ve nelerin yollanacağını belirler.
- Bu aşamadan sonra dosya alışverişi yapılır.
- Dosya alış verişi bittikten sonra tekrar karşıya flag yollanır ve aynı süreç tekrar devam eder dosya yollanmasa bile client sonlanmaz ve her hangi bir dosya değiştirilmesi ile dosya taşıma tekrar tetiklenir.

ORTAK NOKTALAR:

- Tasarıma göre client bağlandıktan sonra servera dosyasını paylaştı daha sonrasında bu dosyayı kendi pathinden silerse tekrar serverdan kendisine dosya iletilecektir.
- Bu durum server için de geçerlidir.
- Hem client hem de server dosya kopyalama işlemini blok halinde yapmaktadırlar yani her iki taraf içinde sinyal gelmesi dosya kopyalama işlemini yarıda bitirmez.
- Sinyal yakalanması yeni bir isteğin işleme alınamayacağı ve elde kalan son istek bittikten sonra sonlanacağı anlamına gelmektedir.

TEST DURUMLARI:

Durum 1: Thread pooldaki tüm threadler clientlarda senkronize oluyorken yeni istek server dolu diye reddedili.

```
es Terminal May 18 21:45
arif@arif-MSI: ~/Desktop/shared-folder/Final
gcc -c -std=gnu99 Server.c
gcc -c -std=gnu99 Client.c
gcc -o Server Server.o -lpthread
gcc -o Client Client.o -lpthread
Usage: ./Server [directory] [threadPool Size] [port]
Usage: ./Client [directory] [ip address] [port]
arif@arif-MSI:~/Desktop/shared-folder/Final$ ./Client A 127.0.0.1 8080
Baglantı başarılı
A/dogru arif.txt dosya servera yollanacak
A/dogru\arif.txt dosya servera yollanacak
A/dogru arif.txt Server'a yollandı
A/dogru\arif.txt Server'a yollandı

Main Server'a Yeni bağlantı istegi geldi
D Pathi ile Bağlantı Sağlandı
Aktif Bağlantılı Pathler
A
C
B
D
Main Server Yeni bağlantı bekliyor
Bu dosya burada yok:D/1
dosya alınacak D/1
D/1 dosyası server'a kopyalandı
Main Server'a Yeni bağlantı istegi geldi
Threadler dolu bağlantı reddedildi
Aktif Bağlantılı Pathler
A
C
B
D
Main Server Yeni bağlantı bekliyor

arif@arif-MSI:~/Desktop/shared-folder/Final$ ./Client C 127.0.0.1 8080
Baglantı başarılı
C/blaba.txt dosya servera yollanacak
C/blaba.txt Server'a yollandı

arif@arif-MSI:~/Desktop/shared-folder/Final$ ./Client D 127.0.0.1 8080
Baglantı başarılı
D/1 dosya servera yollanacak
D/1 Server'a yollandı

arif@arif-MSI:~/Desktop/shared-folder/Final$ ./Client B 127.0.0.1 8080
Baglantı başarılı
B/ahmet.c dosya servera yollanacak
B/lala.tx dosya servera yollanacak
B/ahmet.c Server'a yollandı
B/lala.tx Server'a yollandı

arif@arif-MSI:~/Desktop/shared-folder/Final$ ./Client E 127.0.0.1 8080
Baglantı başarısız: Threadlerin hepsi dolu
arif@arif-MSI:~/Desktop/shared-folder/Final$
```

Durum 2: 1. Durumdan sonra bir client bağlantısını koparırsa yani ctrl-c (SIGINT) alırsa yeni gelen istek bağlanabilir.

```
arif@arif-MSI:~/Desktop/shared-folder/Final$ make
gcc -c -std=gnu99 Server.c
gcc -c -std=gnu99 Client.c
gcc -o Server Server.o -lpthread
gcc -o Client Client.o -lpthread
Usage: ./Server [directory] [threadPool Size] [port]
Usage: ./Client [directory] [ip address] [port]
arif@arif-MSI:~/Desktop/shared-folder/Final$ ./Client A 127.0.0.1 8080
Baglantı basarılı
A/dogrur arif.txt dosya server'a yollanacak
A/dogrur arif.txt server'a yollandi
A/dogrur arif.txt Server'a yollandi
^CFlag 1 olarak set edildi
Sinyal Yakaladi Sonlandi BVE
arif@arif-MSI:~/Desktop/shared-folder/Final$

Bu dosya bende yok:D/1
dosya alınacak O/1
D/1 dosyasi server'a kopyalandi
Main Server'a Yeni baglanti istegi geldi
Threadler Dolu Baglanti Reddedildi
Aktif Baglanti Pathler
A
C
B
D
Main Server Yeni baglanti bekliyor
Main Server'a Yeni baglanti istegi geldi
E Pathi ile Baglanti saglandi
Aktif Baglanti Pathler
C
B
D
E
Main Server Yeni baglanti bekliyor

arif@arif-MSI:~/Desktop/shared-folder/Final$ ./Client C 127.0.0.1 8080
Baglantı basarılı
C/bla.txt dosya server'a yollanacak
C/bla.txt Server'a yollandi

arif@arif-MSI:~/Desktop/shared-folder/Final$ ./Client D 127.0.0.1 8080
Baglantı basarılı
D/1 dosya server'a yollanacak
D/1 Server'a yollandi

arif@arif-MSI:~/Desktop/shared-folder/Final$ ./Client B 127.0.0.1 8080
Baglantı basarılı
B/ahmet.c dosya server'a yollanacak
B/lala.tx dosya server'a yollanacak
B/ahmet.c Server'a yollandi
B/lala.tx Server'a yollandi

arif@arif-MSI:~/Desktop/shared-folder/Final$ ./Client E 127.0.0.1 8080
Baglantı basarısız: Threadlerin hepsi dolu
arif@arif-MSI:~/Desktop/shared-folder/Final$ ./Client E 127.0.0.1 8080
Baglantı basarılı
```

Durum 3: Eğer server tarafına SIGINT sinyali gelirse bağlı tüm clientlar sinyal yakaladığına dair mesaj olarak sonlanır.

```
arif@arif-MSI:~/Desktop/shared-folder/Final$ make
gcc -c -std=gnu99 Server.c
gcc -c -std=gnu99 Client.c
gcc -o Server Server.o -lpthread
gcc -o Client Client.o -lpthread
Usage: ./Server [directory] [threadPool Size] [port]
Usage: ./Client [directory] [ip address] [port]
arif@arif-MSI:~/Desktop/shared-folder/Final$ ./Client A 127.0.0.1 8080
Baglantı basarılı
A/dogrur arif.txt dosya server'a yollanacak
A/dogrur arif.txt server'a yollandi
A/dogrur arif.txt Server'a yollandi
^CFlag 1 olarak set edildi
Sinyal Yakaladi Sonlandi BVE
arif@arif-MSI:~/Desktop/shared-folder/Final$

Main Server Yeni baglanti bekliyor
Main Server'a Yeni baglanti istegi geldi
E Pathi ile Baglanti saglandi
Aktif Baglanti Pathler
C
B
D
E
Main Server Yeni baglanti bekliyor
^Cctrl-c Geldi threadler kapatiliyor..
***** Thread pool artik bos server kapaniyor *****
Toplam 5 tane client'a hizmet verildi
Asagidaki pathler icin hizmet verildi
1-> A
2-> C
3-> B
4-> D
5-> E
arif@arif-MSI:~/Desktop/shared-folder/Final$

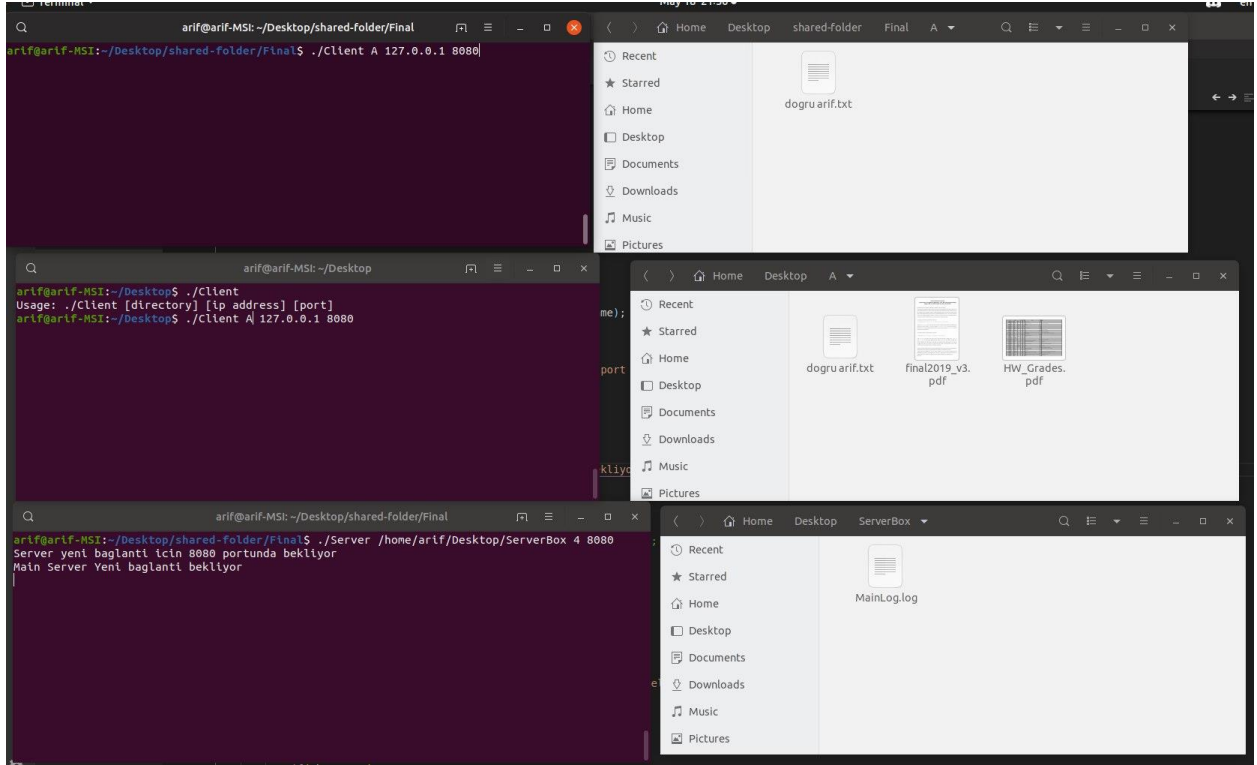
arif@arif-MSI:~/Desktop/shared-folder/Final$ ./Client C 127.0.0.1 8080
Baglantı basarılı
C/bla.txt dosya server'a yollanacak
C/bla.txt Server'a yollandi
Sinyal Yakaladi Sonlandi BVE
arif@arif-MSI:~/Desktop/shared-folder/Final$

arif@arif-MSI:~/Desktop/shared-folder/Final$ ./Client D 127.0.0.1 8080
Baglantı basarılı
D/1 dosya server'a yollanacak
D/1 Server'a yollandi
Sinyal Yakaladi Sonlandi BVE
arif@arif-MSI:~/Desktop/shared-folder/Final$

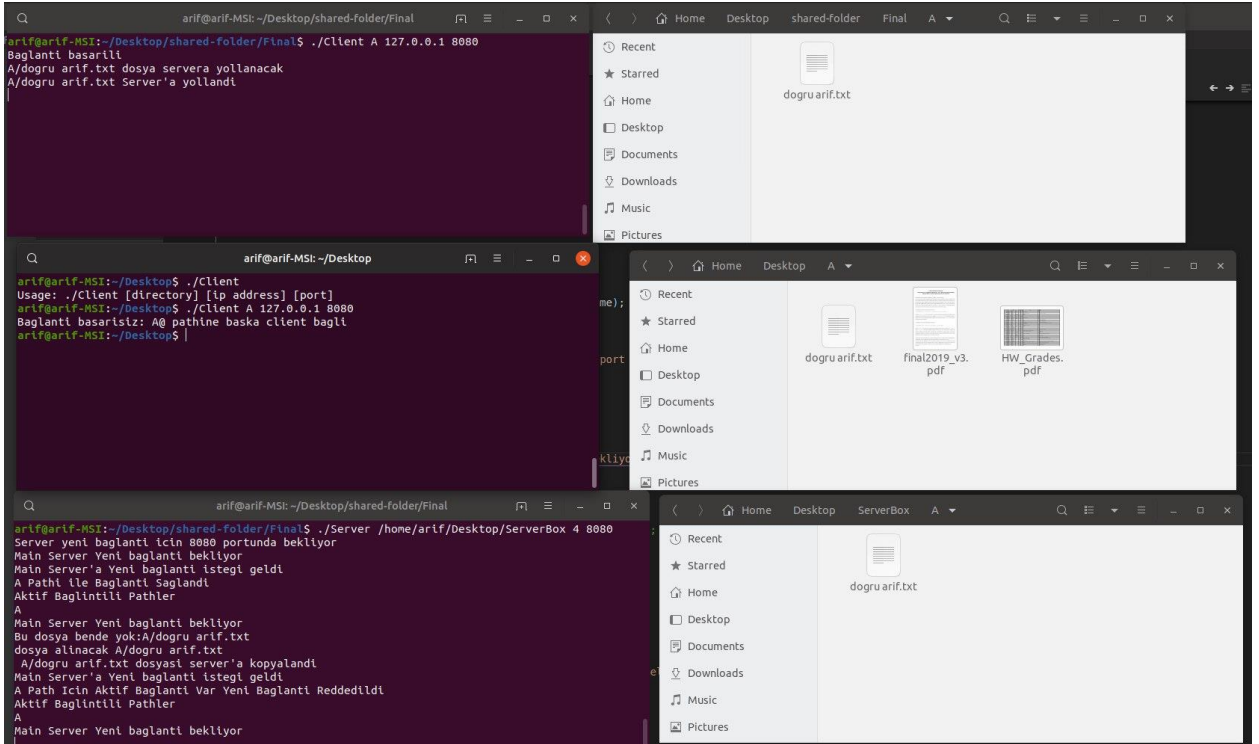
arif@arif-MSI:~/Desktop/shared-folder/Final$ ./Client B 127.0.0.1 8080
Baglantı basarılı
B/ahmet.c dosya server'a yollanacak
B/lala.tx dosya server'a yollanacak
B/ahmet.c Server'a yollandi
B/lala.tx Server'a yollandi
Sinyal Yakaladi Sonlandi BVE
arif@arif-MSI:~/Desktop/shared-folder/Final$

arif@arif-MSI:~/Desktop/shared-folder/Final$ ./Client E 127.0.0.1 8080
Baglantı basarısız: Threadlerin hepsi dolu
arif@arif-MSI:~/Desktop/shared-folder/Final$ ./Client E 127.0.0.1 8080
Baglantı basarılı
Sinyal Yakaladi Sonlandi BVE
arif@arif-MSI:~/Desktop/shared-folder/Final$
```

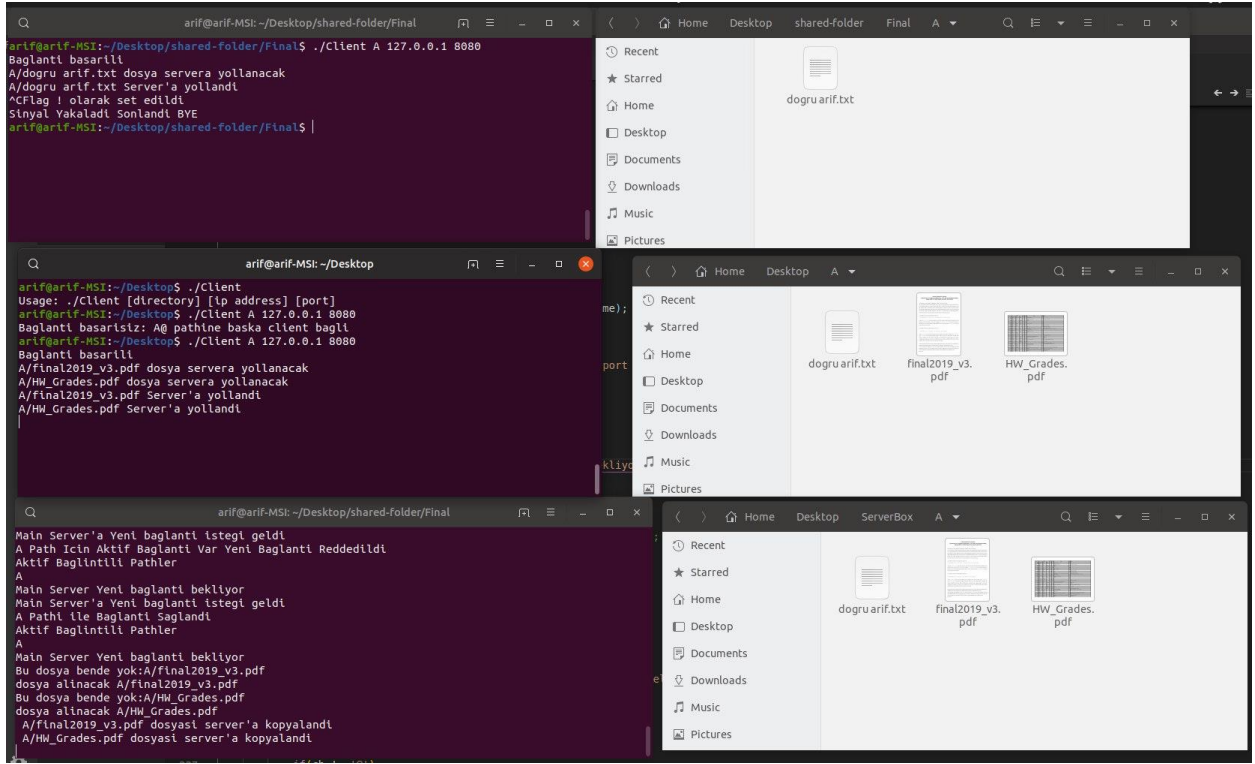
Durum 4.1: A pathine bir client bağlanır serverda dosyaları yedeklenir



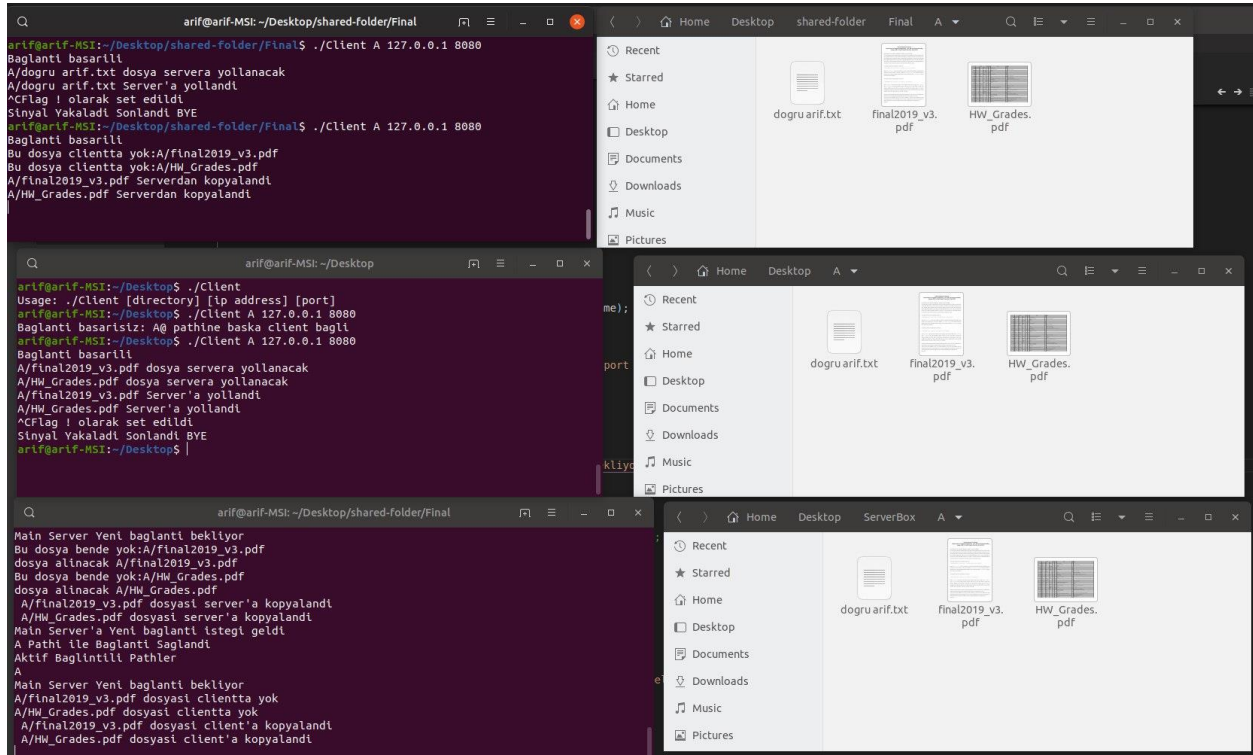
Durum 4.2: A pathine bağlı bu client online olduğu için aynı pathe bağlanmak isteyen başka client a izin verilmez.



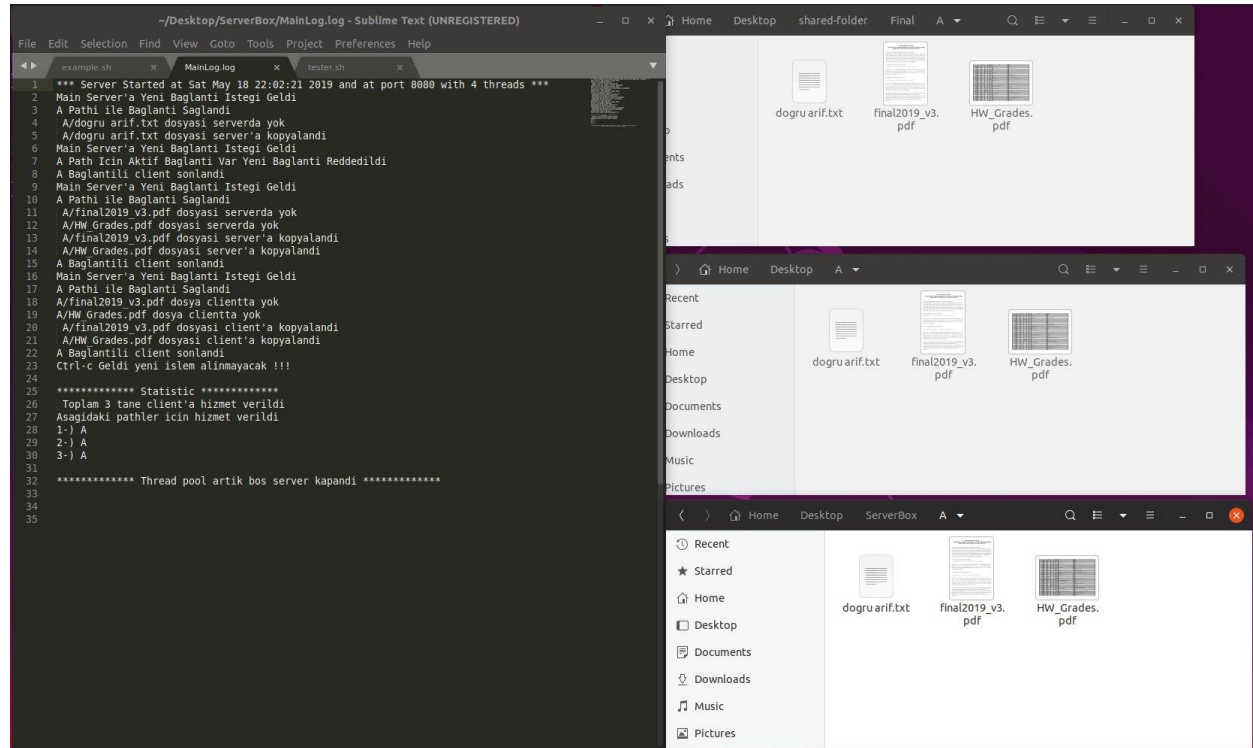
Durum 4.3: A pathine bağlı bu client sinyal yoluyla sonlandırılır. Ardından başka bir client A path ile bağlanmak ister ve bağlanır dosyaları servera aktarılır. Serverda olup o clientta olmayanlarda kendisine yollanır.



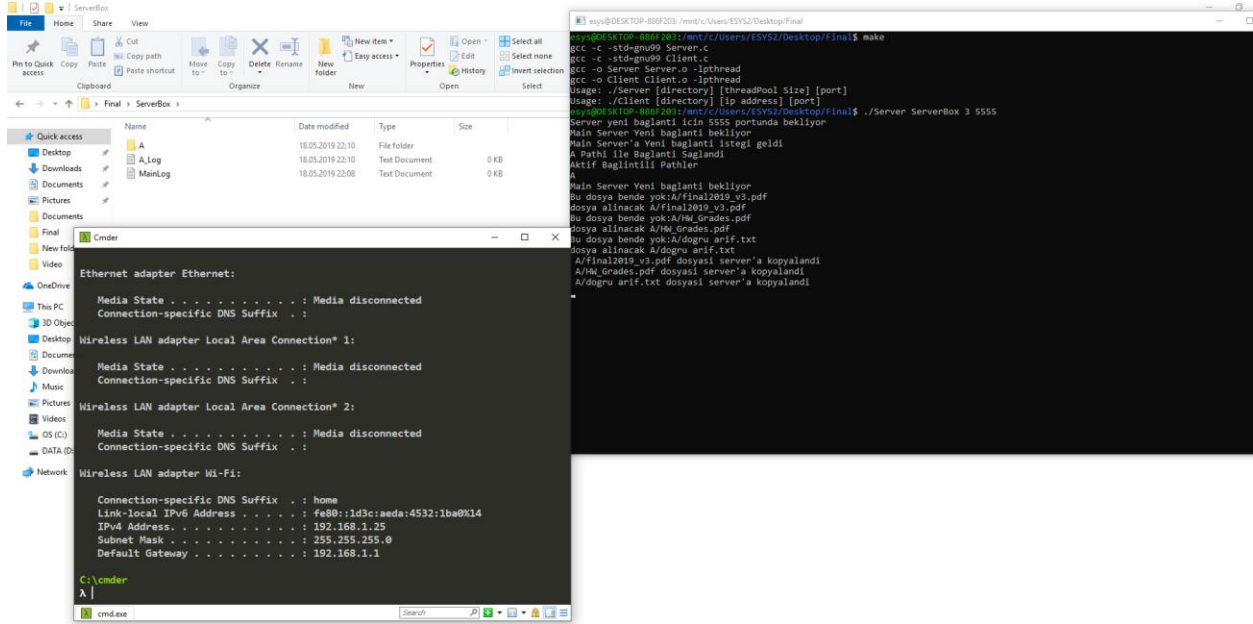
Durum 4.3: Ardından bu client offline olur bir önceki aynı client A pathi ile tekrar bağlandığında servera atılan tüm dosyalar A pathinde olduğu için ona da yüklenir.



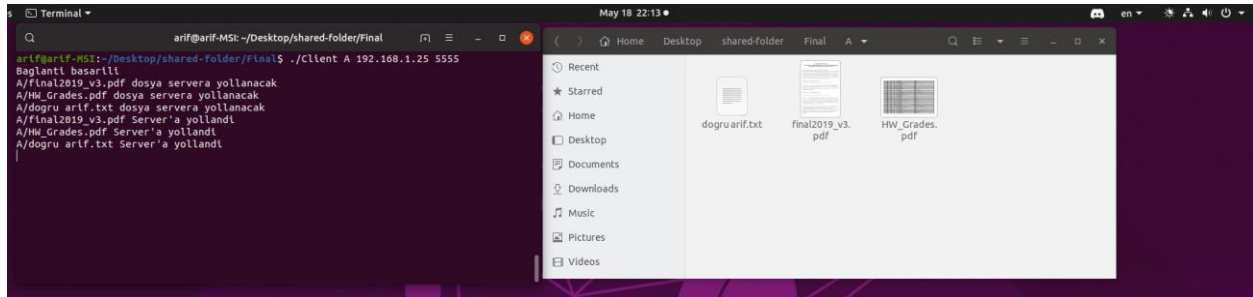
Durum 4.3: Bu durumu gösteren log dosyası ekran görüntüsü:



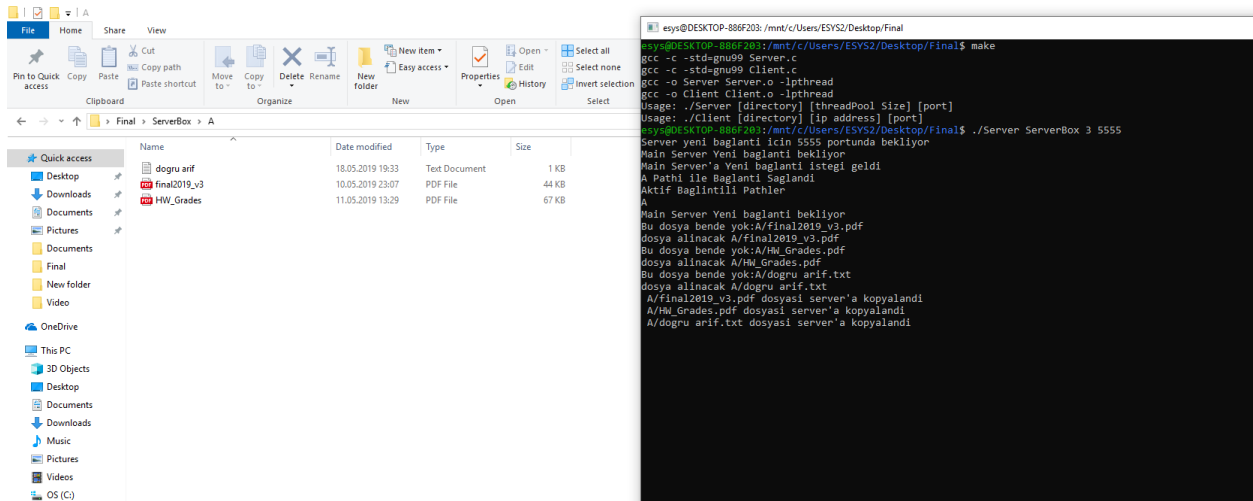
Durum 5: Farklı bilgisayarlar arası dosya aktarım görselleri adım adım aşağıda paylaşılmıştır. İp adresleri görülmektedir.



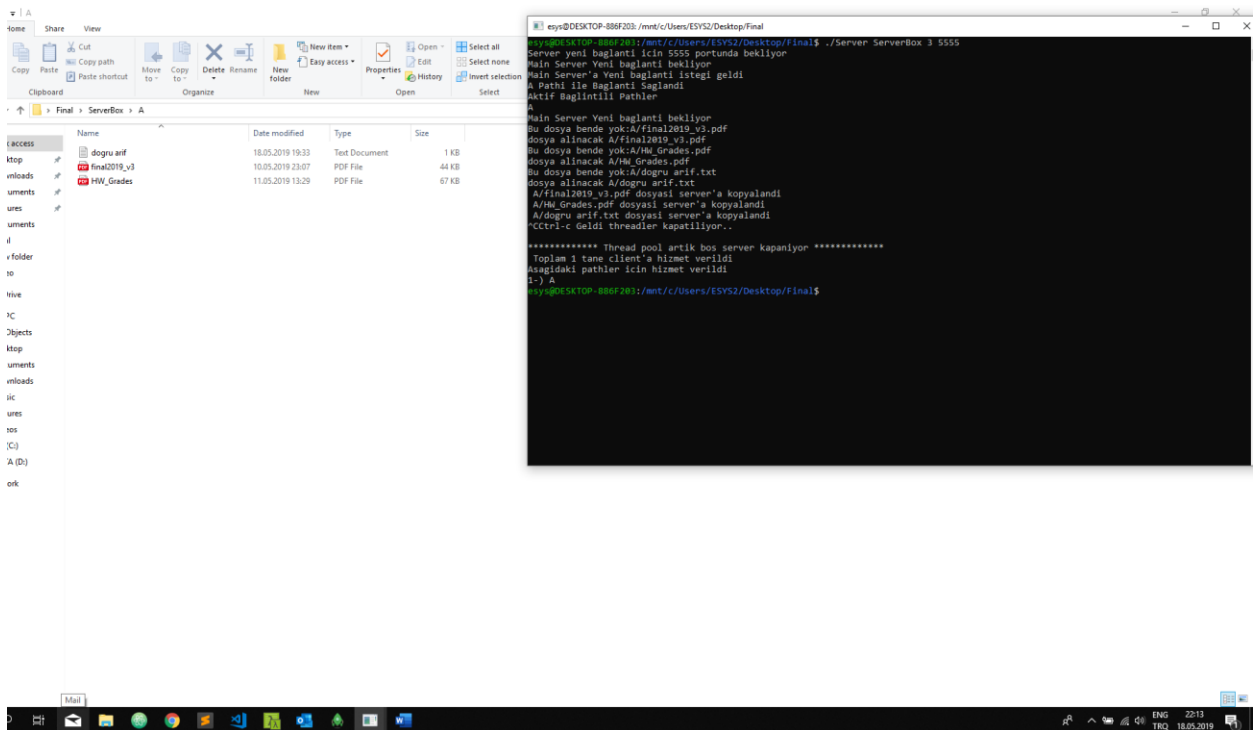
From Linux Bash



From Ubuntu

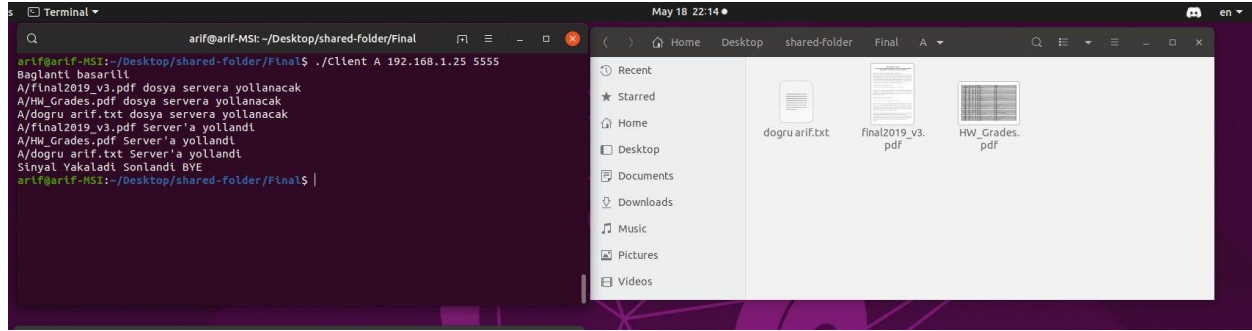


From Linux Bash



Linux Bash SIGINT geldi

Resimdeki ipden ISGINT gelince ubuntudaki client sona erdi.



Bu durumun Log Dosyası:

