

## TUGAS 5 PROGRAM JARINGAN

### 1. PENDAHULUAN

Protokol merupakan serangkaian aturan dan prosedur yang mengatur pertukaran informasi antara perangkat atau sistem. Protokol menentukan format, urutan, dan tindakan yang harus dilakukan dalam pertukaran data. Dalam tugas ini, yang menjadi fokus utama adalah membuat protokol chat yang memungkinkan pengguna untuk berkomunikasi secara real-time melalui jaringan.

### 2. TUGAS

Tugas kali ini, akan dibuat sebuah protokol chat dengan menerapkan operasi autentikasi, inbox, group chat, group chat antar realm, private chat, dan private chat antar realm.

### 3. ARSITEKTUR

#### a) Spesifikasi Komputer

Hardware Model: HP 240 G8 Notebook PC

Memory: 16,0 GiB

Processor: Intel Core i7-1065G7 CPU @ 1.30GHz × 8

Graphics: ICELAND(iceland, LLVM 15.0.6, DRM 3.47, 5.19.0-38-generic)

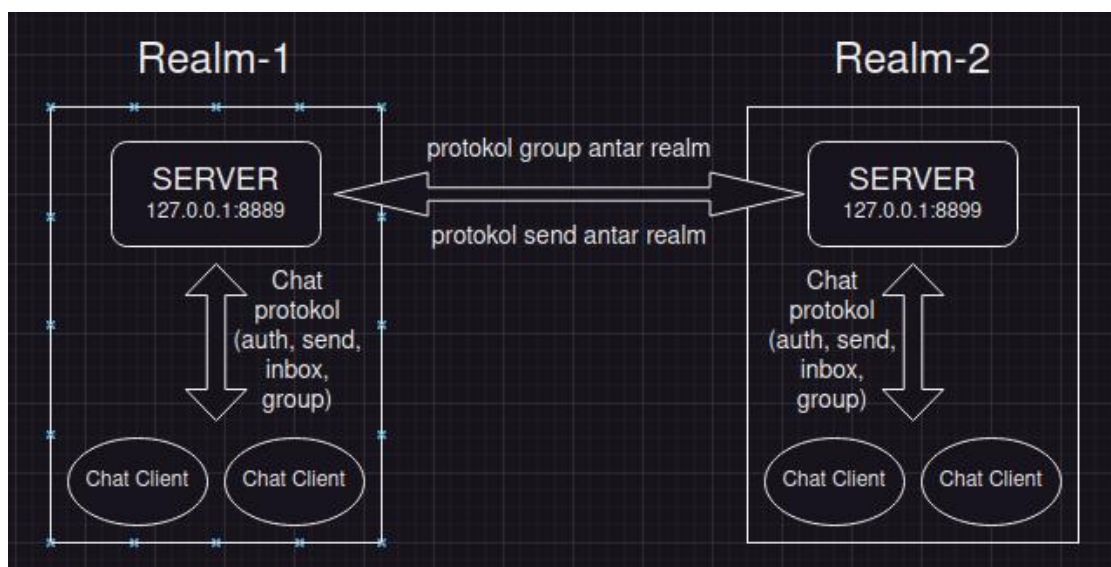
Disk Capacity: 512,1 GB

OS Name: Ubuntu 22.04.2 LTS

OS Type: 64-bit

GNOME Version: 42.5

#### b) Arsitektur Percobaan



Percobaan dilakukan dengan membuat dua server yang masing masing server beralamatkan 127.0.0.1:8889 dan 127.0.0.1:8899. Client setiap realm hanya dapat mengirim request ke server di masing-masing realm. Untuk request yang membutuhkan komunikasi antarrealm (seperti group dan send antarrealm), komunikasi dilakukan oleh server tiap realm, kemudian data yang dikembalikan kepada client dilakukan oleh server pada realm yang sama.

Pada setiap server, telah memiliki user yang terdaftar. Server realm 1 memiliki 3 username yang terdaftar, yaitu messi, henderson, dan lineker. Server realm 2 memiliki 3 username yang terdaftar, yaitu james, jantuar, dan taeno.

#### **4. DOKUMENTASI TUGAS**

##### **a) Authentikasi**

Server:

```
WARNING:root:data dari client: auth messi surabaya
```

```
WARNING:root:AUTH: auth messi surabaya
```

```
WARNING:root:balas ke client: {"status": "OK", "tokenid": "15299a77-ce5b-458e-b3ba-9b565c09c135"}
```

Client:

```
Command :auth messi surabaya
```

```
diterima dari server b'{"status": "OK", "tokenid": "15299a77-ce5b-458e-b3ba-9b565c09c135"}
```

```
diterima dari server b'5"}\r\n\r\n'
```

```
end of string
```

```
username messi logged in, token 15299a77-ce5b-458e-b3ba-9b565c09c135
```

```
Command 15299a77-ce5b-458e-b3ba-9b565c09c135:█
```

Ketika client mengirimkan request auth dengan parameter username dan password, server akan menerima request. Jika username terdaftar di server, server akan mengembalikan status serta tokenid kepada client

##### **b) Private Chat**

Server:

```
WARNING:root:data dari client: send 15299a77-ce5b-458e-b3ba-9b565c09c135 henderson apa kabar disana?
```

```
WARNING:root:SEND: session 15299a77-ce5b-458e-b3ba-9b565c09c135 send message from messi to henderson
```

```
WARNING:root:balas ke client: {"status": "OK", "message": "Message Sent"}
```

Client:

```
Command 15299a77-ce5b-458e-b3ba-9b565c09c135:send henderson apa kabar disana?
```

```
diterima dari server b'{"status": "OK", "message": "Message Sent"}\r\n\r\n'
```

```
end of string
```

```
message sent to henderson
```

Ketika client mengirimkan request send dengan parameter username tujuan dan pesan, server akan menerima request. Jika username tujuan terdaftar di server, maka server mengembalikan status dan message.

##### **c) Group Chat**

### Server:

```
WARNING:root:data dari client: group 15299a77-ce5b-458e-b3ba-9b565c09c135 batak origin  
  
WARNING:root:GROUP: batak  
WARNING:root:connecting to ('0.0.0.0', 8899)  
WARNING:root:Isi data sebelum dikirim ke server sebelah = check group messi batak other  
  
WARNING:root:Tidak ada grup batak di server sebelah, lakukan pembuatan grup baru  
WARNING:root:GROUP: batak
```

### Client (username = messi):

```
Command 15299a77-ce5b-458e-b3ba-9b565c09c135:group batak  
messi telah bergabung  
henderson telah bergabung  
halo gaes  
messi: halo gaes  
henderson: halo messi!!!  
█
```

Ketika client mengirimkan request group dengan parameter nama group, maka server akan lakukan pencarian nama group terlebih dahulu. Jika tidak ada nama group yang dimaksud pada server, maka server akan melanjutkan request ke server sebelah untuk mengecek apakah nama group yang dimaksud tersedia di server sebelah (proses ini akan dijelaskan pada definisi protokol antarserver). Jika server sebelah tidak memiliki nama group tersebut, request dikirimkan kembali ke server semula, kemudian server semula akan membuat grup baru dengan nama group tersebut.

### d) Private Chat antarrealm (messi -> james)

#### Server semula:

```
WARNING:root:data dari client: send 2ca4a83c-c193-4b67-a88b-c78cf10bc352 james Halo, apa kabar, James?  
  
WARNING:root:SEND: session 2ca4a83c-c193-4b67-a88b-c78cf10bc352 send message from messi to james  
WARNING:root:connecting to ('0.0.0.0', 8899)  
WARNING:root:Isi data sebelum dikirim ke server sebelah = server send messi james Halo, apa kabar, James?  
  
WARNING:root:balas ke client: {"status": "OK", "message": "Message Sent"}
```

#### Server beda realm:

```
WARNING:root:connection from ('127.0.0.1', 48552)  
WARNING:root:balas ke client: {"status": "OK", "message": "Message Sent", "sendback": {"msg_from": "messi", "msg_to": "James Maranata", "msg": " Halo, apa kabar, James? \r\n\r\n"}}
```

### Client:

```
Command 2ca4a83c-c193-4b67-a88b-c78cf10bc352:send james Halo, apa kabar, James?  
diterima dari server b'{"status": "OK", "message": "Message Sent"}\r\n\r\n'  
end of string  
message sent to james  
█
```

Client akan mengirimkan request send kepada server dengan parameter username tujuan. Ketika username tujuan tidak ada di server, maka server akan meneruskan request ke server sebelah. Jika terdapat username tujuan pada server sebelah, maka pesan diterima dan dituliskan pada inbox user tujuan.

Kemudian server sebelah mengembalikan status, message, serta msg, dimana msg berguna untuk dituliskan pada outgoing user pengirim.

### **e) Group Chat antarrealm**

Server semula:

```
WARNING:root:data dari client: group 270ada14-a9d4-46d8-a652-c526e8223845 batak origin  
  
WARNING:root:GROUP: batak  
WARNING:root:connecting to ('0.0.0.0', 8889)  
WARNING:root:Isi data sebelum dikirim ke server sebelah = check group james batak other
```

□

Server beda realm:

```
WARNING:root:connection from ('127.0.0.1', 59818)  
menerima pesan dari client = halo gaes
```

Client (username = messi):

```
Command 270ada14-a9d4-46d8-a652-c526e8223845:group batak  
james telah bergabung  
halo gaes  
james: halo gaes  
messi: halo james, selamat datang!!!
```

User akan mengirim request ke server dengan parameter nama grup. Jika nama grup yang dimaksud tidak terdapat di server, maka request diteruskan pada server sebelah. Server sebelah akan menerima koneksi yang berasal dari server semula, dan akan jika terdapat nama grup pada server sebelah, maka koneksi akan tetap terjaga hingga akhirnya client keluar dari grup dengan pesan 'exit'.

### **f) Inbox**

Server:

```
WARNING:root:data dari client: inbox 872cc6f7-d793-4e7a-8eff-658d528d4922
```

```
WARNING:root:INBOX: 872cc6f7-d793-4e7a-8eff-658d528d4922
```

```
WARNING:root:balas ke client: {"status": "OK", "messages": {"messi": [{"msg_from": "messi", "msg_to": "James Maranata", "msg": " Halo, apa kabar, James? \r\n"}]}}
```

Client (username = james):

```
Command 872cc6f7-d793-4e7a-8eff-658d528d4922:inbox  
diterima dari server b'{"status": "OK", "messages": {"messi": [{"msg_from": "messi", "m'  
diterima dari server b'sg_to": "James Maranata", "msg": " Halo, apa kabar, James? \r\n\r\n'  
diterima dari server b'"}]}}\r\n\r\n'  
end of string  
{"messi": [{"msg_from": "messi", "msg_to": "James Maranata", "msg": " Halo, apa kabar, James? \r\n"}]}
```

Client mengirimkan request ke server tanpa parameter. Kemudian server akan menerima request tersebut dan membuka inbox pada user. Server akan mengembalikan status dan messages kepada client.

## 5. DEFINISI PROTOKOL ANTARSERVER

Request yang menggunakan protokol antatserver adalah send serta group. Untuk mempermudah penjelasan, server semula / awal akan disebutkan sebagai server 1, dan server sebelah atau server beda realm akan disebutkan sebagai server 2.

### a) **Request Send**

Pada request send, protokol anterserver dilakukan jika username tujuan tidak ada pada server 1. Request yang dikirim oleh client akan diteruskan kepada server 2, dengan penambahan string 'server' di depan. Penambahan string ini bermaksud agar server 2 mengetahui bahwa itu request yang dikirim oleh server.

Ketika server sebelah menerima request (setelah ditambahkan string 'server') tersebut, maka server 2 akan menjalankan proses pengecekan username. Jika username tujuan terdapat pada server 2, server 2 akan menyimpan pesan tersebut di dalam inbox user tujuan kemudian mengembalikan data dengan status 'OK', keterangan, serta pesan kepada server 1 dan server 1 akan menuliskan pesan tersebut pada bagian outgoing client. Ketika username tujuan tidak ditemukan pada server 2, server 2 akan mengembalikan data dengan status 'ERROR' beserta keterangannya. Data yang dikembalikan oleh server 2 selanjutnya akan dikirimkan kembali kepada client.

### b) **Request Group**

Mula-mula, Client akan mengirimkan request kepada server 1. Command request grup yang diterima oleh server terdiri dari empat kata, yaitu group, sessionid, nama grup, serta state. State terdiri dari tiga jenis, yaitu 'origin', 'other', serta 'comeback'. Setiap state melambangkan asal dari command tersebut. State 'origin' memiliki arti bahwa command tersebut berasal dari client. State 'other' memiliki arti state tersebut berasal dari server yang diteruskan kepada server 2. State 'comeback' memiliki arti state tersebut berasal dari server 2 yang merupakan nilai kembalian.

Pada request group, protokol antarsserver dilakukan jika nama grup yang dimaksud tidak terdapat pada server 1. Jika nama grup yang dimaksud tidak terdapat pada server, maka server akan meneruskan request tersebut ke server 2 dengan penambahan string 'check' di depan request dan mengganti state request menjadi 'other'. Penambahan string ini bermaksud agar server 2 mengetahui bahwa itu merupakan request dari server 1 untuk memeriksa nama grup yang dimaksud.



Server 2 akan menerima request tersebut dan menjalankan proses pengecekan nama grup. Jika nama grup terdapat pada server 2, server 2 akan menambahkan koneksi dengan server 1 sebagai keanggotaan chat group dan koneksi antara server 1 dan server 2 akan tetap dipertahankan untuk melakukan pertukaran chat. Server 1 akan membuat 2 thread baru untuk melakukan pertukaran chat, yaitu thread untuk komunikasi dari server 2 kepada client dan thread untuk komunikasi dari client kepada server 2. Koneksi akan ditutup ketika client mengirimkan chat 'exit', dan server 2 mengirimkan nilai kembalian berupa status 'OK' serta keterangan bahwa telah selesai melakukan pengiriman chat kepada server 1, yang akan diteruskan kepada client.

Akan tetapi, jika nama group tidak ada pada server 2, maka server 2 akan mengembalikan nilai berupa status 'ERROR' serta keterangannya. Server 1 yang menerima data kembalian berupa status 'ERROR' akan mengubah state request menjadi 'comeback' yang kemudian akan diproses. Proses pada request yang memiliki state 'comeback' dilakukan dengan pembuatan sebuah grup baru dengan nama yang berasal dari request serta menambahkan koneksi dengan client sebagai keanggotaan chat group tersebut. Kemudian akan dilakukan proses pertukaran chat oleh server 1 dengan client. Koneksi akan ditutup ketika client mengirimkan chat 'exit' yang akan diakhiri dengan pengembalian nilai berupa status 'OK' serta keterangannya dari server 1 kepada client.

## **6. KESIMPULAN**

Melalui tugas protokol chat dalam kelas program jaringan, mahasiswa mendapatkan pemahaman yang lebih baik tentang konsep dan implementasi protokol dalam konteks komunikasi melalui jaringan. Juga, melalui tugas ini, mahasiswa telah meningkatkan kemampuan dalam merancang protokol yang responsif.

## **7. CODE**

Link github: <https://github.com/silabanjames/tugas-program-jaringan/tree/main/tugas6>