

Studi Kasus – Naming System

membuat program sistem penamaan dan penemuan/penerjemahan nama ke alamat

entitas object RPC



Sistem Tedistribusi – 2019/202
Henry Saptono, M.Kom

Flat naming - Broadcasting



- Dalam studi kasus ini kita akan **berlatih** untuk menerapkan **mekanisme penamaan entitas (dalam hal ini sebuah object/RPC)** dengan model **Flat Naming** dan menerapkan mekanisme **penemuan (resolusi nama)** suatu entitas dengan **metode broadcasting**
- Untuk mekanisme **penamaan entitas yang terstruktur** dalam studi kasus ini tidak akan dicoba



Maksud dan Tujuan



- Adapun maksud dari studi kasus ini adalah agar mahasiswa mengerti dan memahami seperti apakah sistem penamaan datar (*flat naming*) dan mekanisme penamaannya serta mekanisme penemuan (resolusi) nama ke suatu entitas yang dimaksud (*name resolution*) dengan metode broadcast
- Sedangkan tujuan dari studi kasus ini adalah membuat sebuah sistem (program) yang berfungsi sebagai sistem penamaan datar (*flat naming*) dan penemuan/penerjemahan nama (*name resolution*)

Skenario



- Sistem penamaan:
 - Entitas → object RPC
 - Format nama entitas → string (ex. rpcobj)
 - Format alamat entitas (access point) → string (ex. tcp://10.0.0.1:7777)
 - Memetakan nama ke alamat (ex. rpcobj → tcp://10.0.0.1:7777)
 - Registrasi dan update nama → entitas memanggil prosedur pada server sistem penamaan secara unicast untuk operasi:
 - Registrasi
 - Update
 - Remove

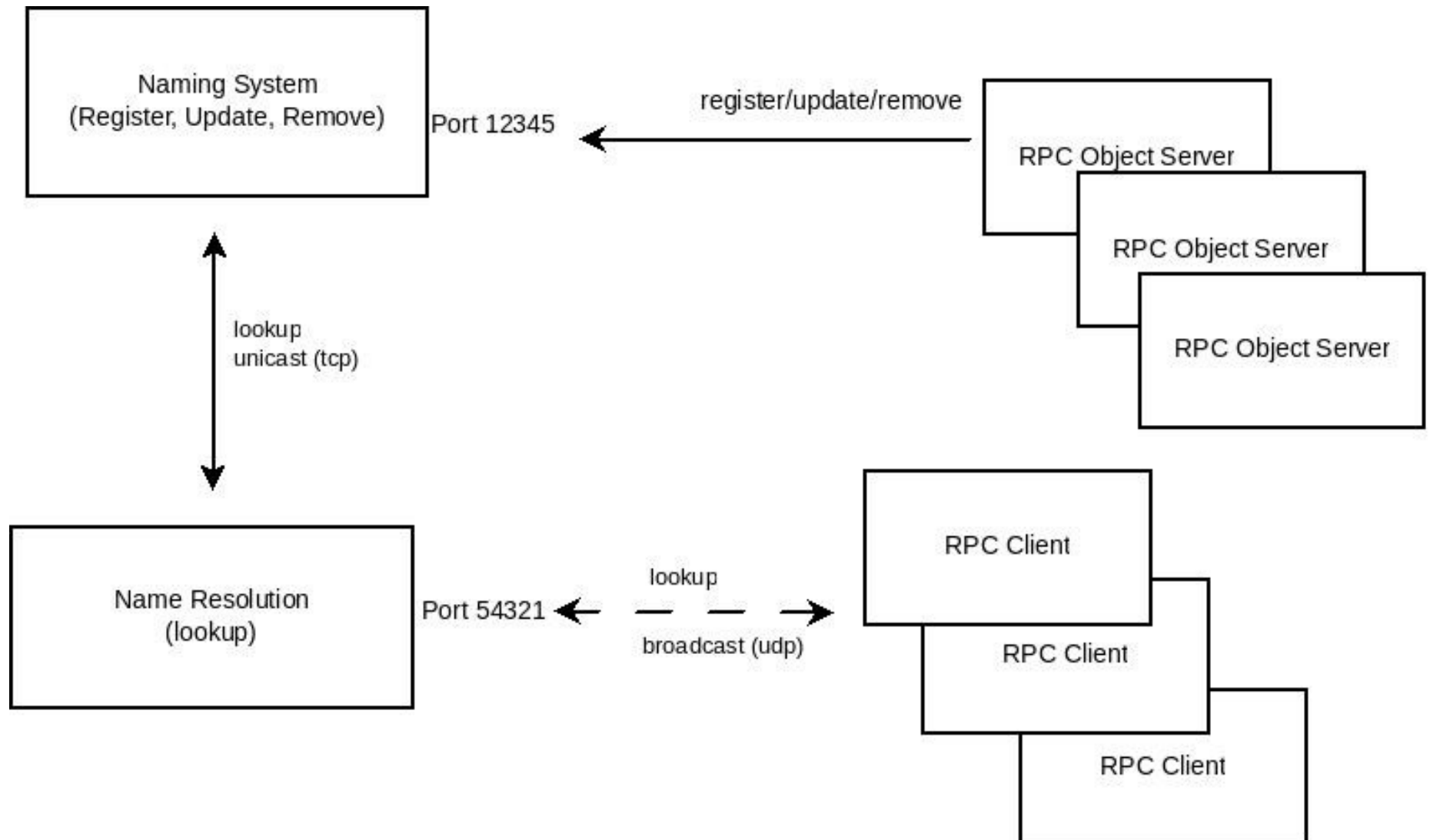
Skenario



- Sistem penemuan/penerjemahan nama (name resolution):
 - RPC Client akan menghubungi sistem penemuan/penerjemahan nama ini ketika RPC Client membutuhkan menemukan alamat dari sebuah entitas (object RPC) dengan melakukan resolusi nama
 - Resolusi nama dilakukan secara broadcast (Protokol UDP) dengan nomor port tujuan 54321. Dengan mengirimkan pesan → <nama entitas> (ex. **Rpcobject**)



Arsitektur sistem



Langkah-Langkah



- Langkah 1, kita akan membuat kode program yang akan bertanggungjawab dalam menangani proses registrasi, update dan remove nama
- Langkah 2, uji coba proses registrasi, update dan remove nama
- Langkah 3, kita akan membuat kode program yang akan bertanggungjawab dalam menangani proses penemuan/penerjemahan nama (name resolution atau looking up)
- Langkah 4, uji coba pemanfaatan sistem penamaan dan penerjemahan nama, dalam kasus pemanggilan remote object RPC

Langkah 1



- Sebelum kita membuat kode program yang bertugas untuk melayani proses penamaan beberapa persiapan perlu diperhatikan terkait software yang dibutuhkan:
 - Python 3.x
 - Zerorpc



Langkah 1



- Kode program sistem penamaan diberi nama **“nameservice.py”**
- Kode program ini didalamnya terdapat definisi class NameService yang memiliki 4 methode (register, update, remove dan lookup)
- Dalam kode program ini clas NameService disediakan ke jaringan sebagai Remote Object yang dapat dipanggil melalui alamat dengan alamat tcp://<ip address>:12345
- Kode program ini dapat Anda peroleh di <https://elen.nurulfikri.ac.id> pada **topic 10** , dengan nama file **study-case.zip**

Langkah 2



- Uji coba sistem penamaan
- Buka console/terminal, kemudian eksekusi program nameservice.py seperti berikut:

```
# python nameservice.py
```

- Kemudian buka tab/windows console/terminal baru, dan uji registrasi nama entitas dengan perintah berikut:

```
# zerorpc tcp://127.0.0.1:12345 register xyz  
tcp://192.168.1.9:8888
```

- Perhatikan output di console yang menjalankan nameservice.py !
(apa hasilnya ?)
- Uji lookup:

```
# zerorpc tcp://127.0.0.1:12345 lookup xyz
```
- Lalu perhatikan output di console ! (apa hasilnya ?)

Langkah 3



- Pada langkah 3 kita akan membuat kode program dengan nama **“nameresolution.py”**
- Kode program ini akan melakukan respon atas permintaan dari RPC Client untuk menemukan alamat dari suatu object RPC dengan berdasarkan nama entitas
- Dalam kode program ini, ada definisi prosedur lookup yang akan melakukan permintaan pemanggilan prosedur lookup nama ke **NAMESERVER**
- Dalam kode program ini juga ada pembuatan socket untuk menerima pesan broadcast dari RPC Client untuk penerjemahan nama ke alamat entitas, melalui port 54321
- Kode program ini dapat diperoleh di <https://elen.nurulfikri.ac.id> pada **topic 10** , dengan nama file **study-case.zip**

Langkah 4



- Pada langkah ke 4 ini kita akan menguji coba pemanfaatan sistem penamaan dan penerjemahan nama, dalam kasus pemanggilan remote object RPC
- Untuk itu silahkan diunduh di <https://elen.nurulfikri.ac.id> pada **topic 10** , dengan nama file **study-case.zip** kode program **rpc-server.py** (entitas Remote Object nya) dan **rpc-client.py** (entitas RPC Client)
- Kemudian buka 4 jendela/tab console/terminal, lalu jalankan masing masing kode program secara berurutan , pertama **nameservice.py**, kedua **nameresolution.py**, ketiga **rpc-server.py**, dan keempat **rpc-client.py**

Pertanyaan



- Ketika Anda jalankan ke empat kode program tersebut, maka Anda akan melihat sebuah hasil di console/terminal kode program **rpc-client.py** hasil berupa angka **300** dan **70** , jelaskan angka angka apakah itu ? Dan bagaimanakah angka itu diperoleh/dihasilkan ?
 - Apa yang terjadi jika pada kode program **rpc-server.py** saat Anda ubah parameter atau variable **address = "tcp://0.0.0.0:6000"** , menjadi **address = tcp://0.0.0.0:7000** ? Apakah kode program **rpc-client.py** , ketika dijalankan kembali masih menghasilkan angka 300 dan 70 ? jelaskan mengapa !
 - Derajat/tingkatan **transparansi distribusi** apa sajakah yang tersedia atau dicapai oleh sistem penamaan dan penemuan nama yang kita coba/buat ini , dalam konteks **Pemanggilan remote object RPC** oleh RPC Client ?
-



SELAMAT MENCOBA DAN MENGERJAKAN

