

**PEMROGRAMAN JARINGAN**  
**Tugas Praktikum Concurrency**



**Class E**

**05111840000127 - Salsabila Harlen**

**Lecturer :**

**Royyana M. Ijtihadie**

**Informatics Department**  
**Faculty of ELECTICS**  
**Institut Teknologi Sepuluh Nopember (ITS) Surabaya**  
**2021**

Buatlah program yang mengimplementasikan :

1. multi process
2. multi thread
3. multi process asynchronous
4. multi thread asynchronous

Dengan menggunakan protokol transport UDP, kasus dapat didefinisikan sendiri. dan Buatlah arsitektur jaringan anda sendiri di simulator GNS3

Buatlah laporan dalam bentuk PDF yang berisikan screenshot dari

1. deskripsi kasus yang dibuat
2. gambar arsitektur jaringan (dalam simulator GNS3)
3. program yang dibuat (1-4)
4. hasil outputnya

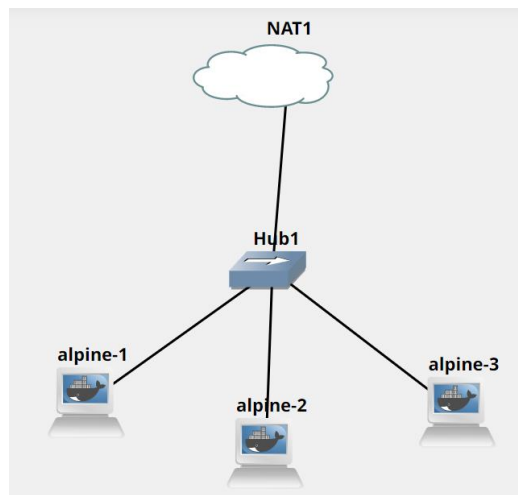
**Jawab :**

Berikut merupakan deskripsi untuk tugas praktikum implementasi kasus pada concurrency yaitu mengunduh sebuah file bertipe image menggunakan project GNS3 yang terdiri dari 3 alpine, dimana :

- Alpine 1 : 192.168.122.130 (server)
- Alpine 2 : 192.168.122.8 (client)
- Alpine 3 : 192.168.122.45 (server)

Selain itu, protocol transport udp akan mengirimkan file pdf kepada server. Pengiriman tersebut akan dengan menggunakan program multi process, multi thread, multi process asynchronous, multi thread asynchronous.

Berikut merupakan topologinya :



- server1.py disesuaikan dengan ip alpine 1

- server2.py disesuaikan dengan ip alpine 3

```

GNU nano 4.6 server2.py
import socket

UDP_IP_ADDRESS = '192.168.122.45'
UDP_PORT = 5758

serverSock = socket.socket(socket.AF_INET,socket.SOCK_DGRAM)
serverSock.bind((UDP_IP_ADDRESS,UDP_PORT))
filename='server2.jpg'
fp = open(filename,'wb+')
ditulis=0
count=0
while True:
    data, addr = serverSock.recvfrom(1024)
    count=count+len(data)
    print(addr, count,len(data), data)
    fp.write(data)

```

4. Modifikasi file – file yang terdapat pada folder progjar3 :

- Library.py

```

1 import logging
2 import requests
3 import socket
4 import os
5 import time
6 import datetime
7
8 def get_url_list():
9     urls = dict()
10    urls['olivia'] = 'https://m.media-amazon.com/images/M/MV5BYTYxODRmNzItMGFGkNS00Mzk4LTk0NTMtYjdiYjgwYWNhODQ2XkEyXkFqcGdeQXQ
11    urls['taylor'] = 'http://jadiberita.com/wp-content/uploads/2014/06/taylor-swift-presenting-jpg.jpg'
12    return urls
13
14 def download_gambar(url=None, tuliskefile='image'):
15     waktu_awal = datetime.datetime.now()
16     if url is None:
17         return False
18     ff = requests.get(url)
19     tipe = dict()
20     tipe['image/png'] = 'png'
21     tipe['image/jpg'] = 'jpg'
22     tipe['image/gif'] = 'gif'
23     tipe['image/jpeg'] = 'jpg'
24     tipe['application/zip'] = 'jpg'
25     tipe['video/quicktime'] = 'mov'
26     # time.sleep(2) #untuk simulasi, diberi tambahan delay 2 detik
27
28     content_type = ff.headers['Content-Type']
29     logging.warning(content_type)
30     if content_type in list(tipe.keys()):
31         namafile = os.path.basename(url)
32         ekstensi = tipe[content_type]
33         if (tuliskefile):
34             fp = open(f"{tuliskefile}.{ekstensi}", "wb")
35             fp.write(ff.content)
36             fp.close()
37         waktu_process = datetime.datetime.now() - waktu_awal
38         waktu_akhir = datetime.datetime.now()
39         logging.warning(f"writing {tuliskefile}.{ekstensi} dalam waktu {waktu_process} {waktu_awal} s/d {waktu_akhir}")
40         return waktu_process

```

```

41     else:
42         return False
43
44 def kirim_gambar(IP_ADDRESS, PORT, filename):
45     print(IP_ADDRESS, PORT, filename)
46     ukuran=os.stat(filename).st_size
47     clientSock = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
48
49     fp=open(filename,'rb')
50     k=fp.read()
51     ter kirim=0
52     for x in k:
53         k_bytes=bytes([x])
54         clientSock.sendto(k_bytes,(IP_ADDRESS,PORT))
55         ter kirim=ter kirim+1
56
57 if __name__=='__main__':
58
59     k = download_gambar('https://m.media-amazon.com/images/M/MV5BYTYxODRmNzItMGFkNS00Mzk4LTk0NTM0YjdiYjgwYWNhODQ2XkEyXkFqc')
60     print(k)

```

## - Multi\_process.py

```

1  from library import download_gambar, get_url_list, kirim_gambar
2  import time
3  import datetime
4  from multiprocessing import Process
5
6  def kirim_server():
7      texec = dict()
8      urls = get_url_list()
9      temp = 0
10     catat_awal = datetime.datetime.now()
11     for k in urls:
12         print(f"mendownload {urls[k]}")
13         waktu = time.time()
14         UDP_IP_ADDRESS = "192.168.122.130"
15         UDP_IP_ADDRESS2 = "192.168.122.45"
16         PORT = 5758
17         #bagian ini merupakan bagian yang menginstruksikan eksekusi fungsi download gambar secara multiprocessing
18         if temp == 0:
19             texec[k] = Process(target=kirim_gambar, args=(UDP_IP_ADDRESS,PORT,f"{k}.jpg"))
20             print('Masuk server 1')
21             temp = temp+1
22         elif temp == 1:
23             print('Masuk server 2')
24             texec[k] = Process(target=kirim_gambar, args=(UDP_IP_ADDRESS2,PORT,f"{k}.jpg"))
25             texec[k].start()
26         #setelah menyelesaikan tugasnya, dikembalikan ke main process dengan join
27         for k in urls:
28             texec[k].join()
29         catat_akhir = datetime.datetime.now()
30         selesai = catat_akhir - catat_awal
31         print(f"Waktu TOTAL yang dibutuhkan {selesai} detik {catat_awal} s/d {catat_akhir}")
32     #fungsi download_gambar akan dijalankan secara multi process
33     if __name__=='__main__':
34         kirim_server()

```

## - Multi\_process\_async.py

```
1 from library import download_gambar, get_url_list, kirim_gambar
2 import time
3 import datetime
4 from multiprocessing import Process, Pool
5
6 def kirim_server():
7     texec = dict()
8     urls = get_url_list()
9     status_task = dict()
10    temp = 0
11    task_pool = Pool(processes=20) #2 task yang dapat dikerjakan secara simultan, dapat diset sesuai jumlah core
12    catat_awal = datetime.datetime.now()
13    for k in urls:
14        download_gambar(urls[k],k)
15        print(f"mendownload {urls[k]}")
16        #bagian ini merupakan bagian yang menginstruksikan eksekusi fungsi download gambar secara multiprocess
17        UDP_IP_ADDRESS = "192.168.122.130"
18        UDP_IP_ADDRESS2 = "192.168.122.45"
19        PORT = 5758
20        if temp == 0:
21            texec[k] = task_pool.apply_async(func=kirim_gambar, args=(UDP_IP_ADDRESS,PORT,f"{k}.jpg"))
22            print('Masuk server 1')
23            temp = temp+1
24        elif temp == 1:
25            print('Masuk server 2')
26            texec[k] = task_pool.apply_async(func=kirim_gambar, args=(UDP_IP_ADDRESS2,PORT,f"{k}.jpg"))
27    #setelah menyelesaikan tugasnya, dikembalikan ke main process dengan mengambil hasilnya dengan get
28    for k in urls:
29        status_task[k]=texec[k].get(timeout=10)
30
31    catat_akhir = datetime.datetime.now()
32    selesai = catat_akhir - catat_awal
33    print(f"Waktu TOTAL yang dibutuhkan {selesai} detik {catat_awal} s/d {catat_akhir}")
34    print("status TASK")
35    print(status_task)
36    #fungsi download_gambar akan dijalankan secara multi process
37    if __name__ == '__main__':
38        kirim_server()
```

## - Multi\_thread.py

```
1 from library import download_gambar,get_url_list, kirim_gambar
2 import time
3 import datetime
4 import threading
5
6 def kirim_server():
7     texec = dict()
8     urls = get_url_list()
9     temp = 0
10    catat_awal = datetime.datetime.now()
11    for k in urls:
12        download_gambar(urls[k], k)
13        print(f"mendownload {urls[k]}")
14        waktu = time.time()
15        UDP_IP_ADDRESS = "192.168.122.130"
16        UDP_IP_ADDRESS2 = "192.168.122.45"
17        PORT = 5758
18        #bagian ini merupakan bagian yang menginstruksikan eksekusi fungsi download gambar secara multithread
19        if temp == 0:
20            texec[k] = threading.Thread(target=kirim_gambar, args=(UDP_IP_ADDRESS,PORT,f"{k}.jpg"))
21            print('Masuk server 1')
22            temp = temp+1
23        elif temp == 1:
24            print('Masuk server 2')
25            texec[k] = threading.Thread(target=kirim_gambar, args=(UDP_IP_ADDRESS2,PORT,f"{k}.jpg"))
26        texec[k].start()
27
28    #setelah menyelesaikan tugasnya, dikembalikan ke main thread dengan join
29    for k in urls:
30        texec[k].join()
31
32    catat_akhir = datetime.datetime.now()
33    selesai = catat_akhir - catat_awal
34    print(f"Waktu TOTAL yang dibutuhkan {selesai} detik {catat_awal} s/d {catat_akhir}")
35    #fungsi download_gambar akan dijalankan secara multithreading
36
37    if __name__ == '__main__':
38        kirim_server()
```

## - Muti\_thread\_async.py

```

1 from library import download_gambar, get_url_list, kirim_gambar
2 import time
3 import datetime
4 import concurrent.futures
5
6 def kirim_server():
7     texec = dict()
8     urls = get_url_list()
9     status_task = dict()
10    temp = 0
11    task = concurrent.futures.ThreadPoolExecutor(max_workers=4)
12    catat_awal = datetime.datetime.now()
13    for k in urls:
14        download_gambar(urls[k], k)
15        print(f"mendownload {urls[k]}")
16        waktu = time.time()
17        UDP_IP_ADDRESS = "192.168.122.130"
18        UDP_IP_ADDRESS2 = "192.168.122.45"
19        PORT = 5758
20        #bagian ini merupakan bagian yang menginstruksikan eksekusi fungsi download gambar secara multithread
21        if temp == 0:
22            texec[k] = task.submit(kirim_gambar, UDP_IP_ADDRESS, PORT, f"{k}.jpg")
23            print('Masuk server 1')
24            temp = temp+1
25        elif temp == 1:
26            print('Masuk server 2')
27            texec[k] = task.submit(kirim_gambar, UDP_IP_ADDRESS2, PORT, f"{k}.jpg")
28    #setelah menyelesaikan tugasnya, dikembalikan ke main thread dengan memanggil result
29    for k in urls:
30        status_task[k] = texec[k].result()
31    catat_akhir = datetime.datetime.now()
32    selesai = catat_akhir - catat_awal
33    print(f"Waktu TOTAL yang dibutuhkan {selesai} detik {catat_awal} s/d {catat_akhir}")
34    print("hasil task yang dijalankan")
35    print(status_task)
36    #fungsi download gambar akan dijalankan secara multithreading
37    if __name__ == '__main__':
38        kirim_server()

```

## 5. Masukkan file – file yang sudah dimodifikasi tadi kedalam alpine 2 (client)

```

< |ole alpine-1 x alpine-2 x alpine-3 > -
inet addr:127.0.0.1 Mask:255.0.0.0
inet6 addr: ::1/128 Scope:Host
UP LOOPBACK RUNNING MTU:65536 Metric:1
RX packets:0 errors:0 dropped:0 overruns:0 frame:0
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)

/ # git clone https://github.com/SalsabilaH12/Pemrograman_Jaringan_E.git
Cloning into 'Pemrograman_Jaringan_E'...
remote: Enumerating objects: 326, done.
remote: Counting objects: 100% (10/10), done.
remote: Compressing objects: 100% (10/10), done.
remote: Total 326 (delta 0), reused 9 (delta 0), pack-reused 316
Receiving objects: 100% (326/326), 767.23 KiB | 1.58 MiB/s, done.
Resolving deltas: 100% (153/153), done.
/ # cd Pemrograman_Jaringan_E
/Pemrograman_Jaringan_E # cd progjar3
/Pemrograman_Jaringan_E/progjar3 # nano library.py
/Pemrograman_Jaringan_E/progjar3 # nano multi_process.py
/Pemrograman_Jaringan_E/progjar3 # nano multi_process_async.py
/Pemrograman_Jaringan_E/progjar3 # nano multi_thread.py
/Pemrograman_Jaringan_E/progjar3 # nano multi_thread_async.py
/Pemrograman_Jaringan_E/progjar3 #

```

6. Jalankan server1.py pada alpine 1 dan server2.py pada alpine 3, serta install module request pada client (alpine 2)

```
< role alpine-1 x alpine-2 x alpine-3 > - x
pip' command.
/Pemrograman_Jaringan_E/progjar3 # python3 -m pip install requests
Collecting requests
  Downloading requests-2.26.0-py2.py3-none-any.whl (62 kB)
    |████████████████████| 62 kB 307 kB/s
Collecting idna<4,>=2.5
  Downloading idna-3.2-py3-none-any.whl (59 kB)
    |████████████████████| 59 kB 448 kB/s
Collecting urllib3<1.27,>=1.21.1
  Downloading urllib3-1.26.6-py2.py3-none-any.whl (138 kB)
    |████████████████████| 138 kB 1.1 MB/s
Collecting certifi>=2017.4.17
  Downloading certifi-2021.5.30-py2.py3-none-any.whl (145 kB)
    |████████████████████| 145 kB 145 kB/s
Collecting charset-normalizer~=2.0.0
  Downloading charset-normalizer-2.0.2-py3-none-any.whl (35 kB)
Installing collected packages: urllib3, idna, charset-normalizer, certifi, requests
Successfully installed certifi-2021.5.30 charset-normalizer-2.0.2 idna-3.2 requests-2.26.0 urllib3-1.26.6
WARNING: You are using pip version 21.0.1; however, version 21.1.3 is available.
You should consider upgrading via the '/usr/bin/python3 -m pip install --upgrade pip' command.
/Pemrograman_Jaringan_E/progjar3 #
```

7. Jalankan multi\_process\_async.py pada client (alpine 2)

```
< role alpine-1 x alpine-2 x alpine-3 > - x
WARNING: You are using pip version 21.0.1; however, version 21.1.3 is available.
You should consider upgrading via the '/usr/bin/python3 -m pip install --upgrade pip' command.
/Pemrograman_Jaringan_E/progjar3 # python3 multi_process_async.py
WARNING:root:image/jpeg
WARNING:root:writing olivia.jpg dalam waktu 0:00:00.087538 2021-07-15 14:11:39.814894 s/d 2021-07-15 14:11:39.902436
download https://m.media-amazon.com/images/M/MV5BYTYxODRmNzItMGFkNS00Mzk4LTk0VTMtYjdiYjgwYWNhODQ2XkEyXkFqcGdeQXVyNTg1MDY4NjQ@._V1_UY1200_CR285,0,630,1200_AL_.jpg
92.168.122.130 5758 olivia.jpg
Masuk server 1
WARNING:root:image/jpeg
WARNING:root:writing taylor.jpg dalam waktu 0:00:12.560337 2021-07-15 14:11:39.910889 s/d 2021-07-15 14:11:52.471239
download http://jadiberita.com/wp-content/uploads/2014/06/taylor-swift-presenting-jpg.jpg
Masuk server 2
92.168.122.45 5758 taylor.jpg
Waktu TOTAL yang dibutuhkan 0:00:13.939941 detik 2021-07-15 14:11:39.814877 s/d 2021-07-15 14:11:53.754818
status TASK
{'olivia': None, 'taylor': None}
/Pemrograman_Jaringan_E/progjar3 #
```



## 8. Jalankan multi\_process.py pada client (alpine 2)

```
< role      alpine-1  X      alpine-2  X      alj >  -  X
WARNING:root:image/jpeg
WARNING:root:writing taylor.jpg dalam waktu 0:00:12.560337 2021-07-15 14:11:39.9
10889 s/d 2021-07-15 14:11:52.471239
mendownload http://jadiberita.com/wp-content/uploads/2014/06/taylor-swift-presen
ting-jpg.jpg
Masuk server 2
192.168.122.45 5758 taylor.jpg
Waktu TOTAL yang dibutuhkan 0:00:13.939941 detik 2021-07-15 14:11:39.814877 s/d
2021-07-15 14:11:53.754818
status TASK
{'olivia': None, 'taylor': None}
/Pemrograman Jaringan E/progjar3 # python3 multi_process.py
mendownload https://m.media-amazon.com/images/M/MV5BYTYxODRmNzItMGFkNS00Mzk4LTk0
NTMtYjdiYjgwYWVhODQ2XkEyXkFqcGdeQXVyNTg1MDY4NjQ@._V1_UY1200_CR285,0,630,1200_AL_
.jpg
Masuk server 1
mendownload http://jadiberita.com/wp-content/uploads/2014/06/taylor-swift-presen
ting-jpg.jpg
Masuk server 2
192.168.122.130 5758 olivia.jpg
192.168.122.45 5758 taylor.jpg
Waktu TOTAL yang dibutuhkan 0:00:02.465217 detik 2021-07-15 14:12:26.322074 s/d
2021-07-15 14:12:28.787291
/Pemrograman Jaringan E/progjar3 #
```

## 9. Jalankan multi\_thread\_async.py pada client (alpine 2)

```
< role      alpine-1  X      alpine-2  X      alj >  -  X
status TASK
{'olivia': None, 'taylor': None}
/Pemrograman Jaringan E/progjar3 # python3 multi_thread_async.py
mendownload https://m.media-amazon.com/images/M/MV5BYTYxODRmNzItMGFkNS00Mzk4LTk0
NTMtYjdiYjgwYWVhODQ2XkEyXkFqcGdeQXVyNTg1MDY4NjQ@._V1_UY1200_CR285,0,630,1200_AL_
.jpg
Masuk server 1
mendownload http://jadiberita.com/wp-content/uploads/2014/06/taylor-swift-presen
ting-jpg.jpg
Masuk server 2
192.168.122.130 5758 olivia.jpg
192.168.122.45 5758 taylor.jpg
Waktu TOTAL yang dibutuhkan 0:00:02.465217 detik 2021-07-15 14:12:26.322074 s/d
2021-07-15 14:12:28.787291
/Pemrograman Jaringan E/progjar3 # python3 multi_thread_async.py
WARNING:root:image/jpeg
WARNING:root:writing olivia.jpg dalam waktu 0:00:00.653534 2021-07-15 14:14:55.7
24247 s/d 2021-07-15 14:14:56.377787
mendownload https://m.media-amazon.com/images/M/MV5BYTYxODRmNzItMGFkNS00Mzk4LTk0
NTMtYjdiYjgwYWVhODQ2XkEyXkFqcGdeQXVyNTg1MDY4NjQ@._V1_UY1200_CR285,0,630,1200_AL_
.jpg
192.168.122.130 5758 olivia.jpg
Masuk server 1
```

## 10. Jalankan multi\_thread.py pada client (alpine 2)

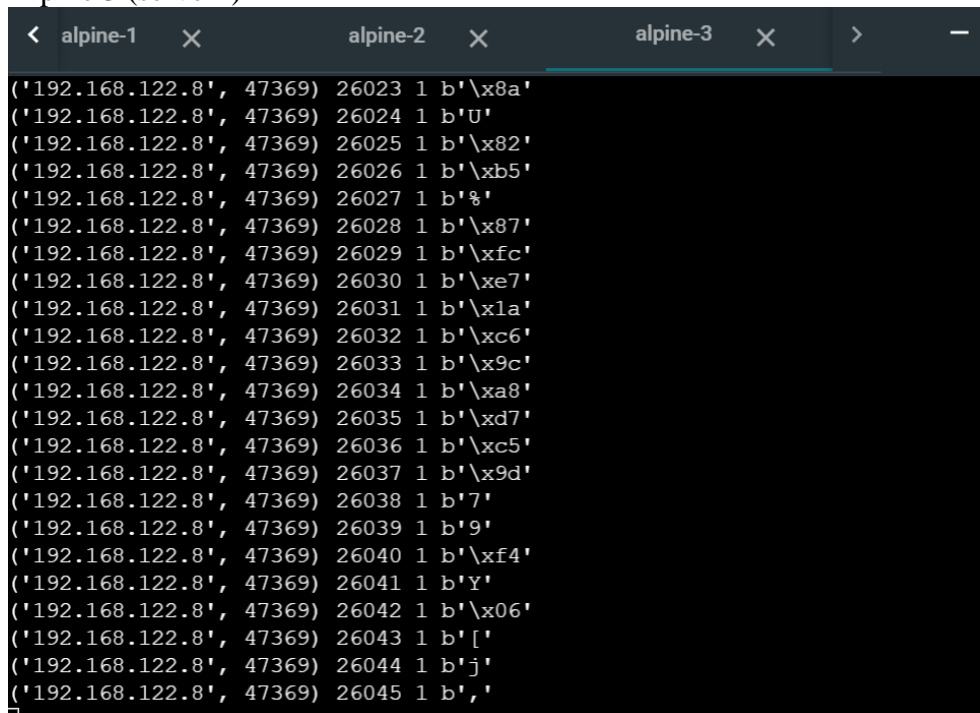
```
< |ole      alpine-1  x      alpine-2  x      al| >  -  x
192.168.122.45 5758 taylor.jpg
Waktu TOTAL yang dibutuhkan 0:00:10.436810 detik 2021-07-15 14:14:55.724241 s/d
2021-07-15 14:15:06.161051
hasil task yang dijalankan
{'olivia': None, 'taylor': None}
/Pemrograman_Jaringan_E/progjar3 # python3 multi_thread.py
WARNING:root:image/jpeg
WARNING:root:writing olivia.jpg dalam waktu 0:00:00.079220 2021-07-15 14:15:25.5
13482 s/d 2021-07-15 14:15:25.592708
mendownload https://m.media-amazon.com/images/M/MV5BYTYxODRmNzItMGFkNS00Mzk4ITk0
NTMtYjdiYjgwYWNhODQ2XkEyXkFqcGdeQXVyNTg1MDY4NjQ@._V1_UY1200_CR285,0,630,1200_AL
.jpg
Masuk server 1
192.168.122.130 5758 olivia.jpg
WARNING:root:image/jpeg
WARNING:root:writing taylor.jpg dalam waktu 0:00:08.364735 2021-07-15 14:15:25.5
93217 s/d 2021-07-15 14:15:33.957959
mendownload http://jadiberita.com/wp-content/uploads/2014/06/taylor-swift-presen
ting-jpg.jpg
Masuk server 2
192.168.122.45 5758 taylor.jpg
Waktu TOTAL yang dibutuhkan 0:00:10.335730 detik 2021-07-15 14:15:25.513474 s/d
2021-07-15 14:15:35.849204
```

## 11. Maka, pada alpine 1 dan 3 sebagai server1 dan server2 akan mengeluarkan hasil berikut.

### - Alpine 1 (server1)

```
< |ole      alpine-1  x      alpine-2  x      al| >  -
('192.168.122.8', 35580) 24499 1 b'w'
('192.168.122.8', 35580) 24500 1 b'F'
('192.168.122.8', 35580) 24501 1 b'\xd7'
('192.168.122.8', 35580) 24502 1 b'A'
('192.168.122.8', 35580) 24503 1 b'\xb7'
('192.168.122.8', 35580) 24504 1 b'\xdd'
('192.168.122.8', 35580) 24505 1 b'\xf8'
('192.168.122.8', 35580) 24506 1 b'h'
('192.168.122.8', 35580) 24507 1 b'Q'
('192.168.122.8', 35580) 24508 1 b'\x86'
('192.168.122.8', 35580) 24509 1 b'\x15'
('192.168.122.8', 35580) 24510 1 b'K'
('192.168.122.8', 35580) 24511 1 b'm'
('192.168.122.8', 35580) 24512 1 b'\xaa'
('192.168.122.8', 35580) 24513 1 b'\x10'
('192.168.122.8', 35580) 24514 1 b'x'
('192.168.122.8', 35580) 24515 1 b'\x88'
('192.168.122.8', 35580) 24516 1 b'x'
('192.168.122.8', 35580) 24517 1 b'\xa2'
('192.168.122.8', 35580) 24518 1 b'\xe0'
('192.168.122.8', 35580) 24519 1 b'\x80'
('192.168.122.8', 35580) 24520 1 b'k'
('192.168.122.8', 35580) 24521 1 b'\x9c'
```

- Alpine 3 (server2)



```
< alpine-1 x alpine-2 x alpine-3 x > -
('192.168.122.8', 47369) 26023 1 b'\x8a'
('192.168.122.8', 47369) 26024 1 b'U'
('192.168.122.8', 47369) 26025 1 b'\x82'
('192.168.122.8', 47369) 26026 1 b'\xb5'
('192.168.122.8', 47369) 26027 1 b'%'
('192.168.122.8', 47369) 26028 1 b'\x87'
('192.168.122.8', 47369) 26029 1 b'\xfc'
('192.168.122.8', 47369) 26030 1 b'\xe7'
('192.168.122.8', 47369) 26031 1 b'\xa1'
('192.168.122.8', 47369) 26032 1 b'\xc6'
('192.168.122.8', 47369) 26033 1 b'\x9c'
('192.168.122.8', 47369) 26034 1 b'\xa8'
('192.168.122.8', 47369) 26035 1 b'\xd7'
('192.168.122.8', 47369) 26036 1 b'\xc5'
('192.168.122.8', 47369) 26037 1 b'\x9d'
('192.168.122.8', 47369) 26038 1 b'7'
('192.168.122.8', 47369) 26039 1 b'9'
('192.168.122.8', 47369) 26040 1 b'\xf4'
('192.168.122.8', 47369) 26041 1 b'Y'
('192.168.122.8', 47369) 26042 1 b'\x06'
('192.168.122.8', 47369) 26043 1 b '['
('192.168.122.8', 47369) 26044 1 b'j'
('192.168.122.8', 47369) 26045 1 b','
```