Load Balancing at the Frontend

Piotr Lewandowski

Load Balancing at the Frontend

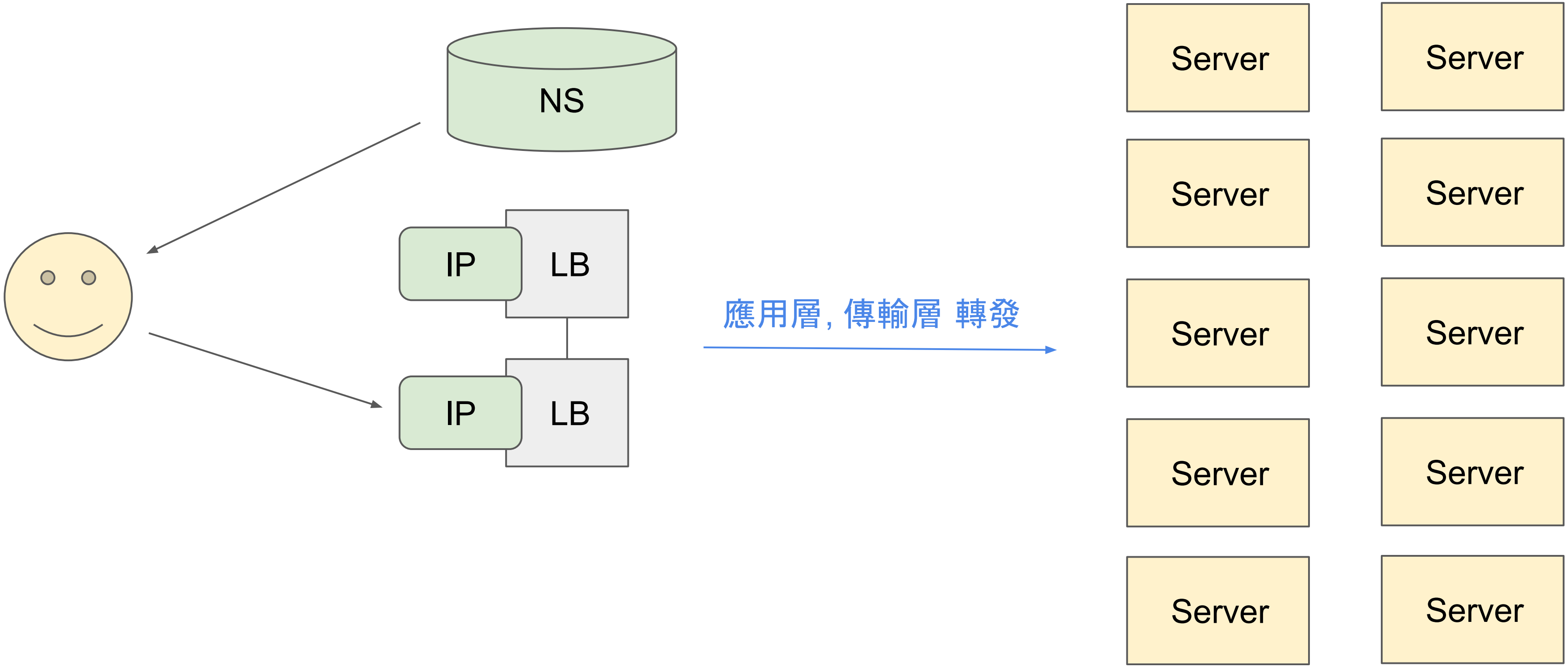
Piotr Lewandowski

負載平衡的一些考量

- 再強大的硬體都有極限
- 負載平衡系統 用來決定 哪些機器 來處理 某個請求
  - 架構考量
  - 技術考量
  - 使用者流量的屬性
- 搜尋請求 vs 影片上傳

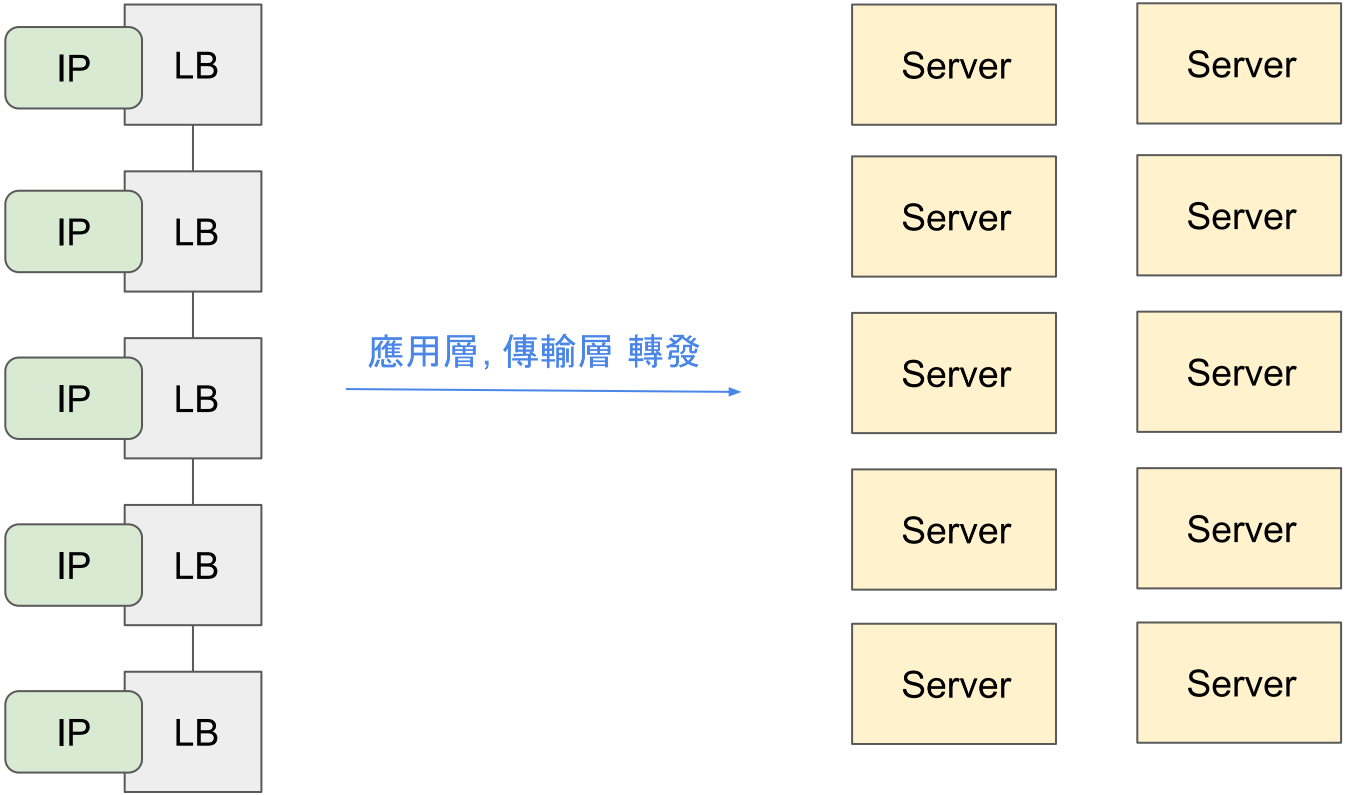
使用 DNS 進行負載平衡

DNS 是最簡單、最有效的負載平衡機制



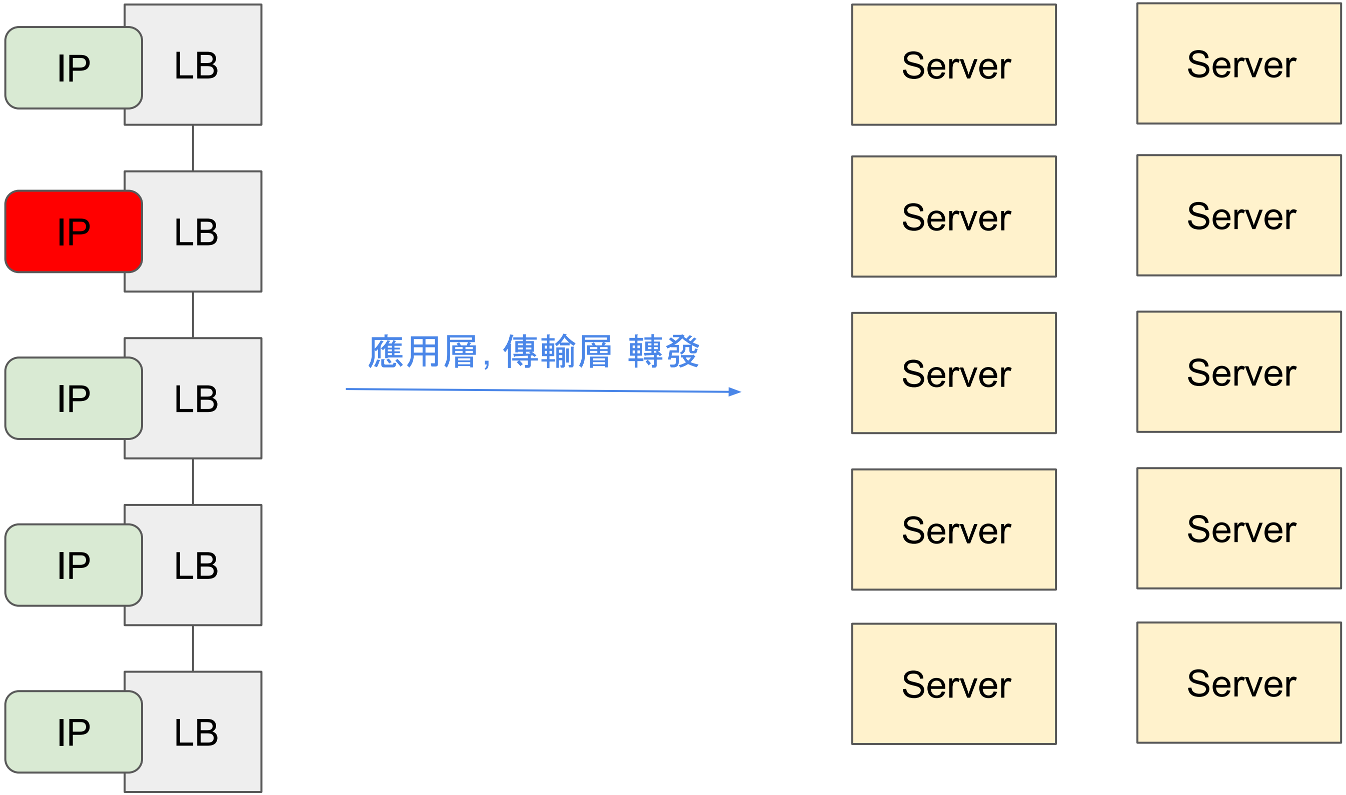
使用 DNS 進行負載平衡

DNS 是最簡單、最有效的負載平衡機制

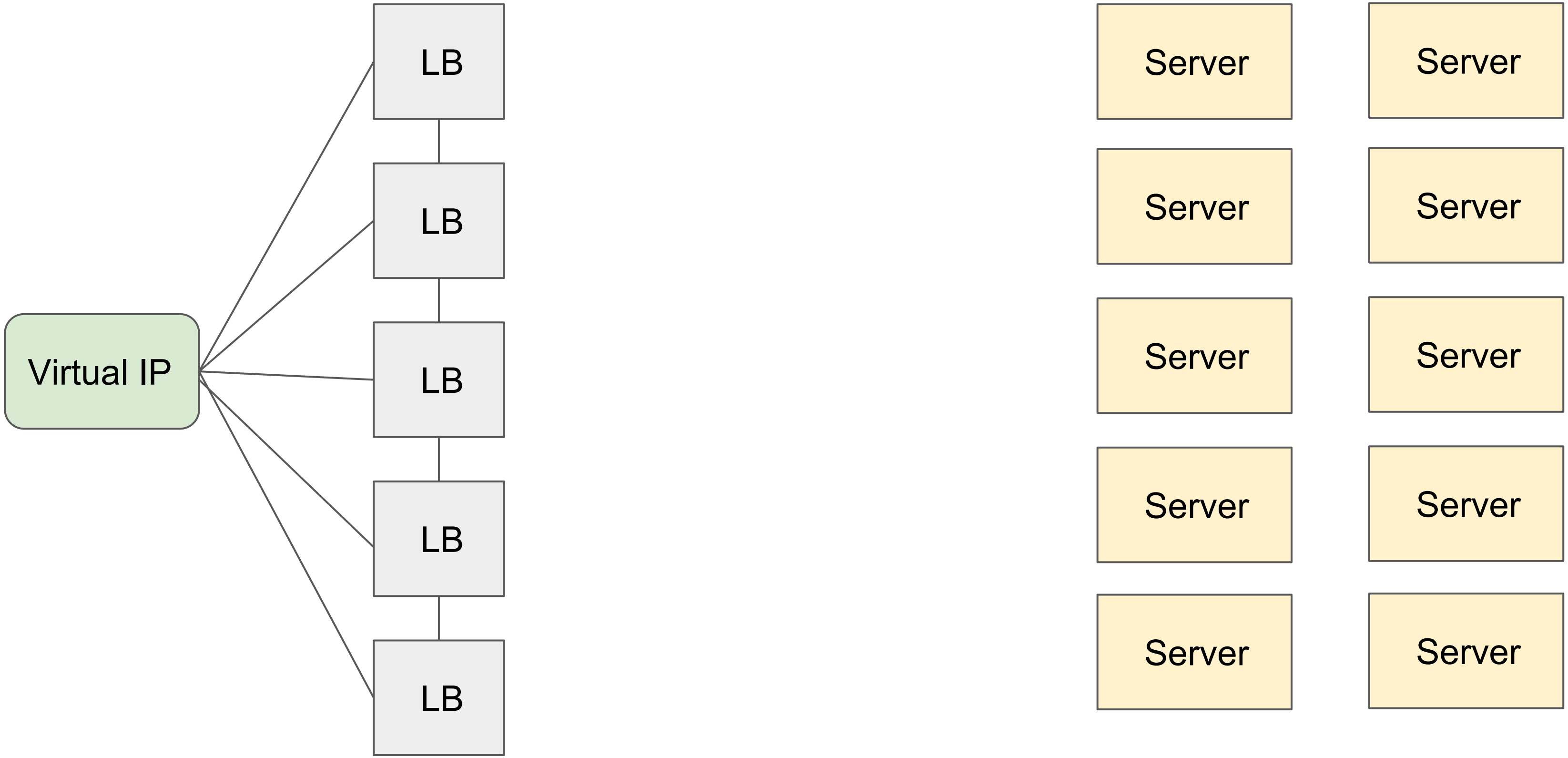


使用 DNS 進行負載平衡

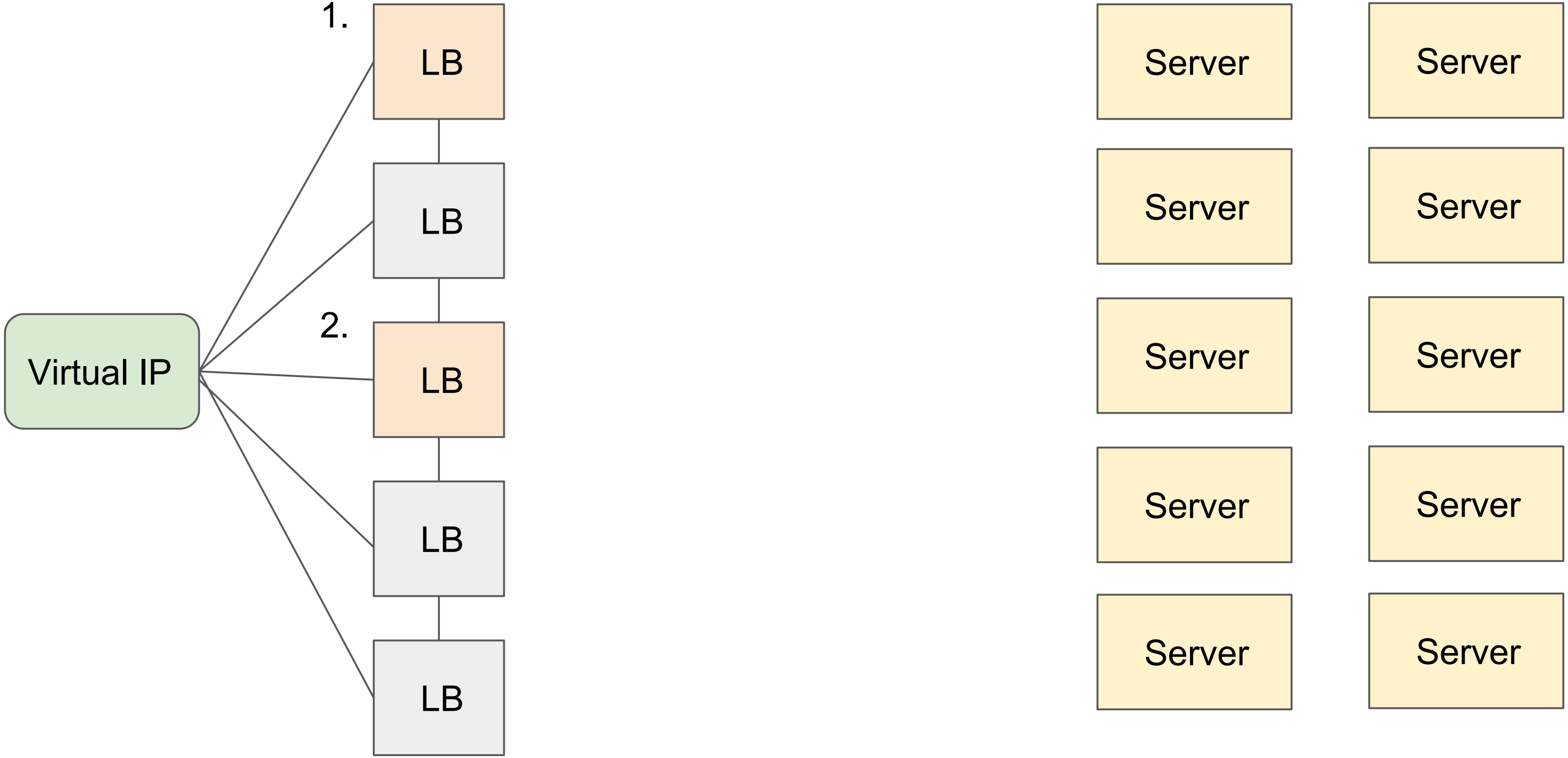
DNS 是最簡單、最有效的負載平衡機制



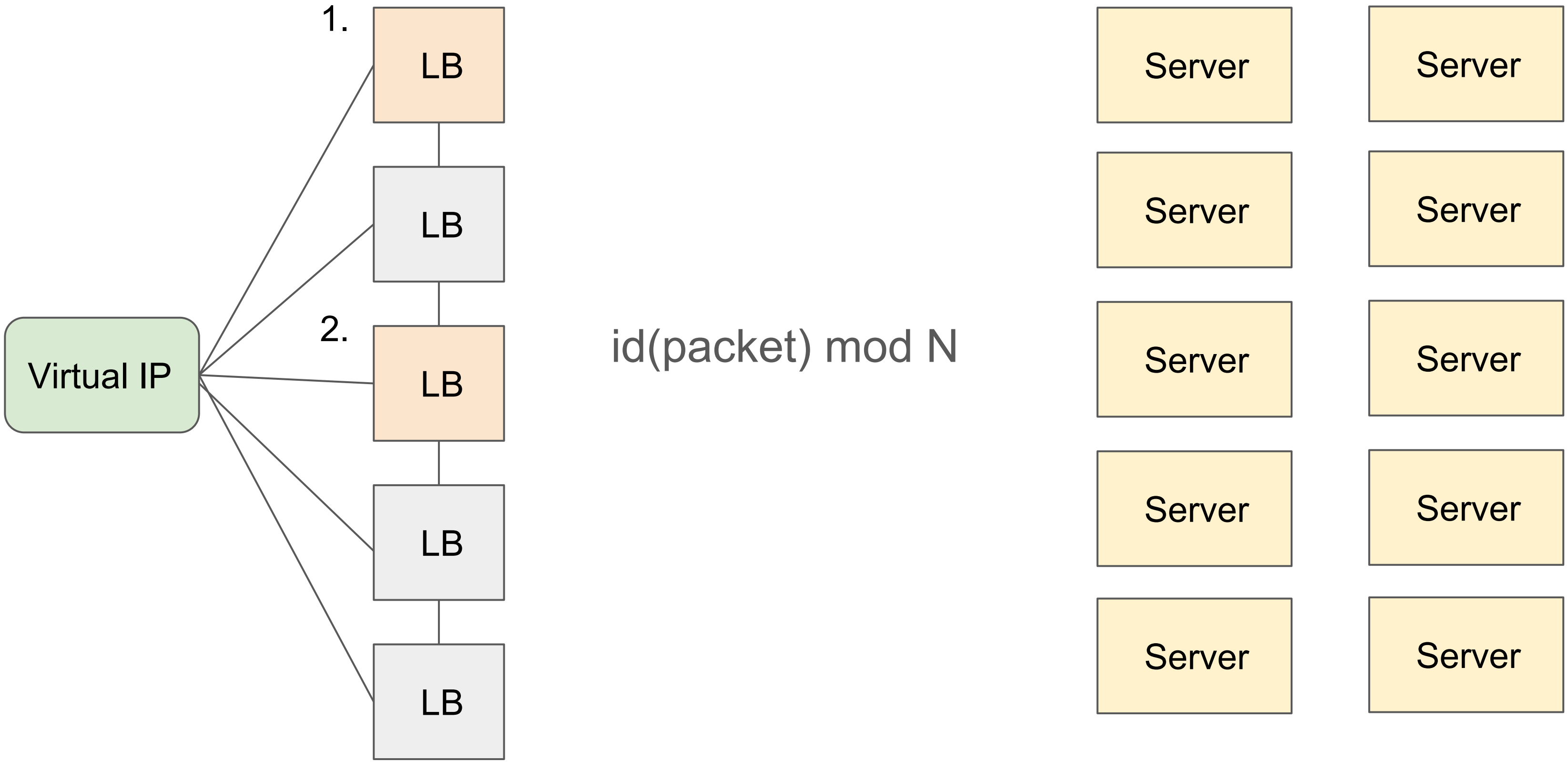
使用虛擬 IP 處理負載平衡 (NLB)



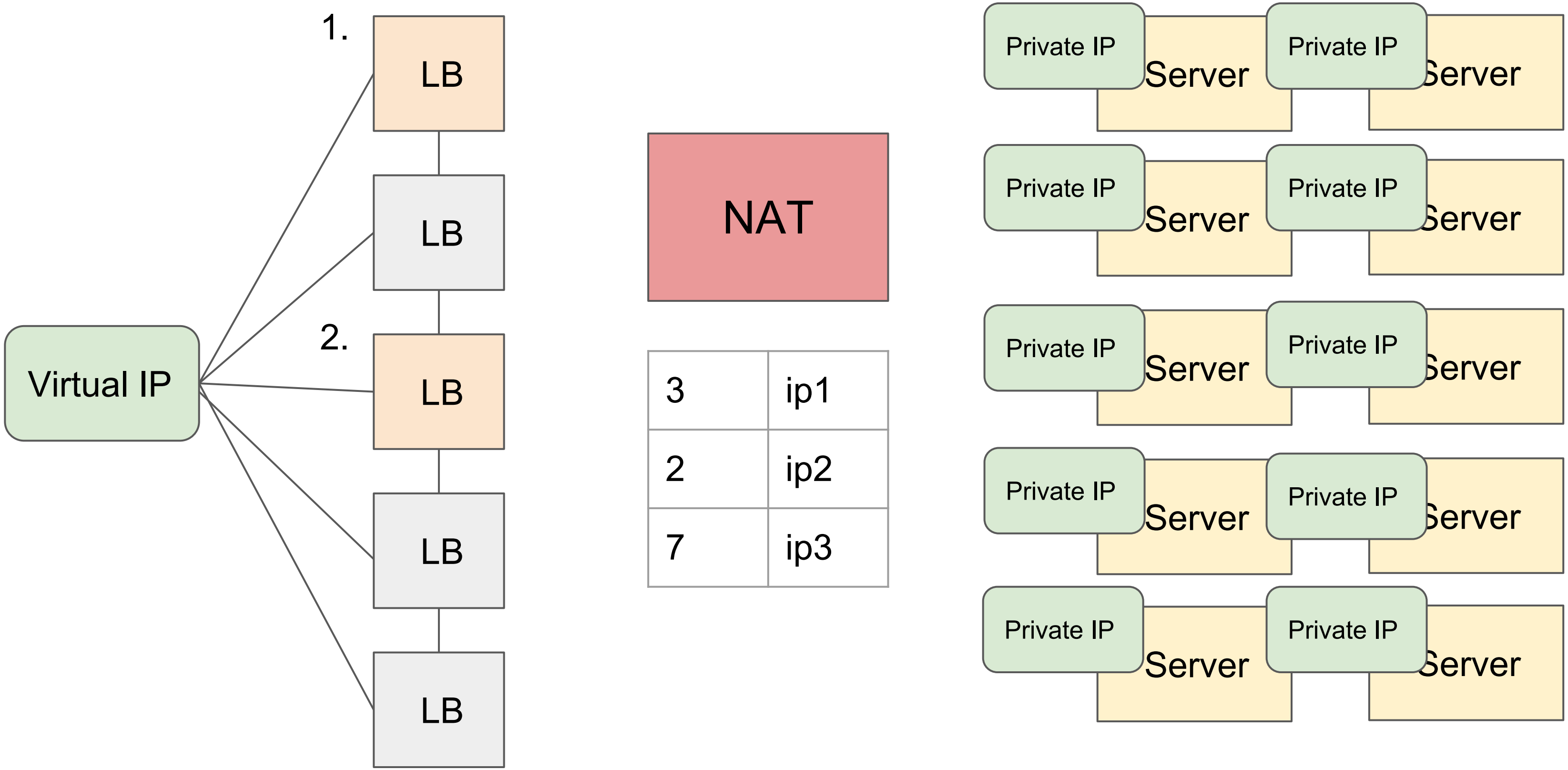
使用虛擬 IP 處理負載平衡 (NLB)



使用虛擬 IP 處理負載平衡 (NLB)

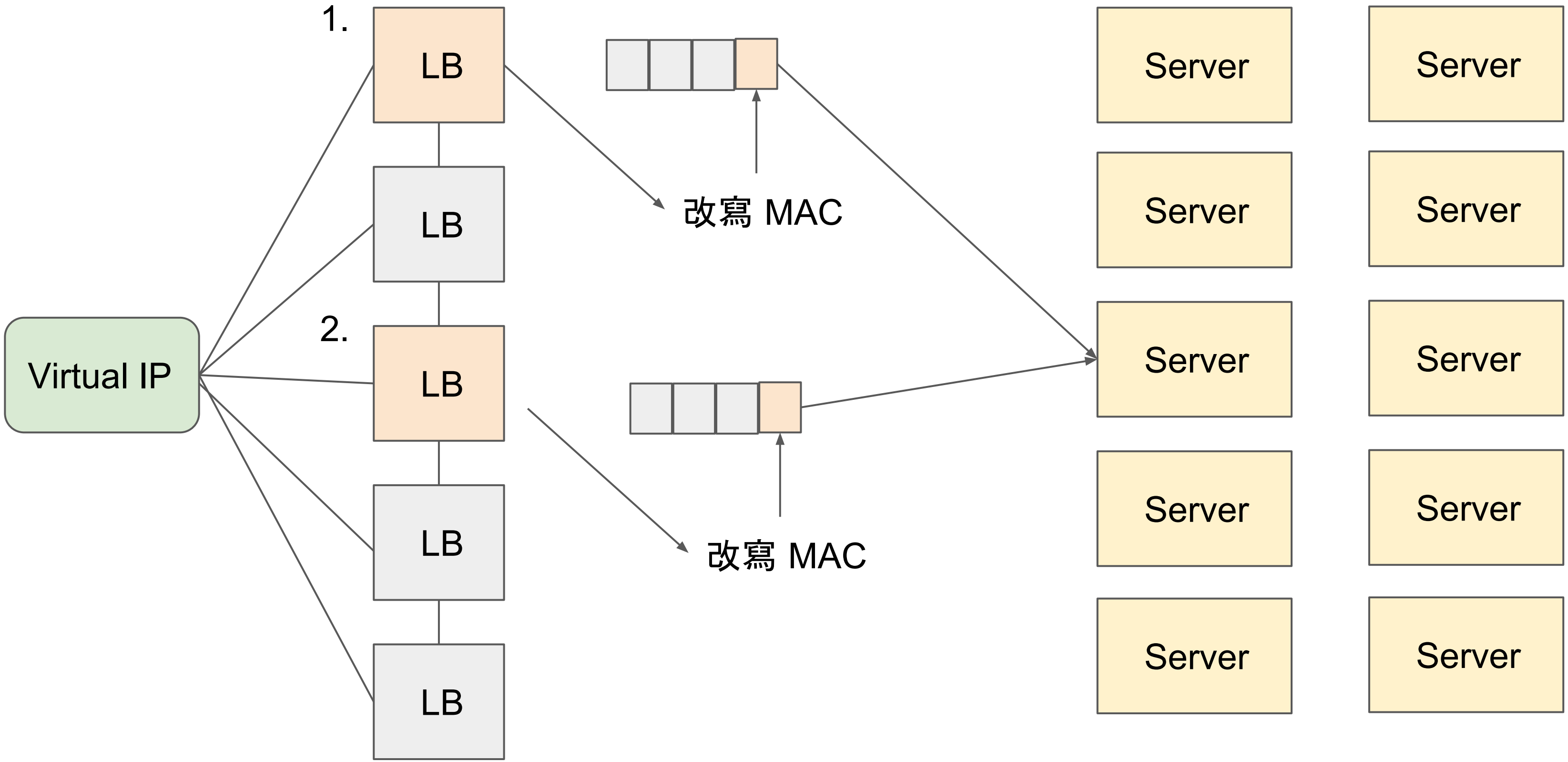


使用虛擬 IP 處理負載平衡 (NLB)

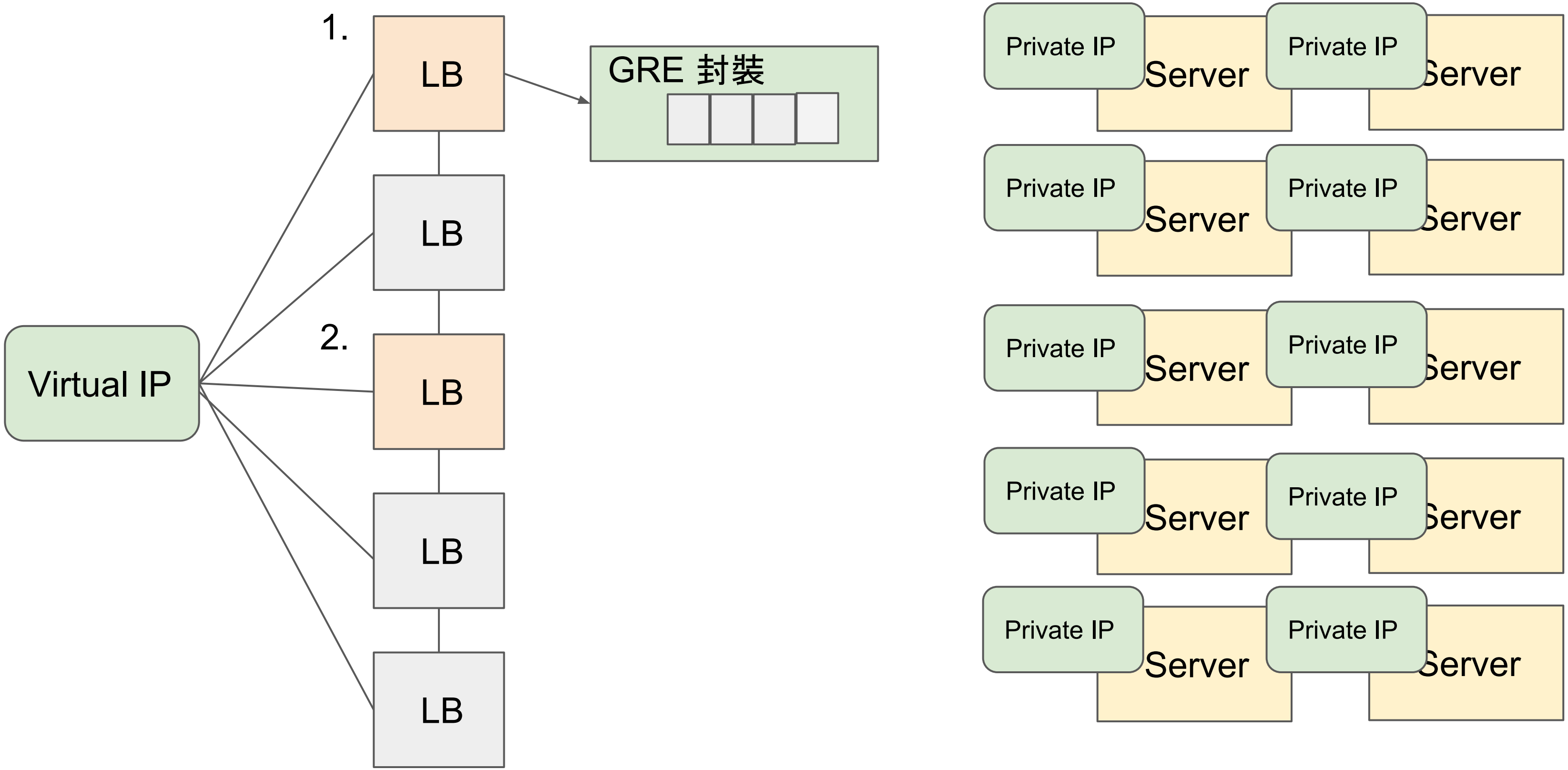


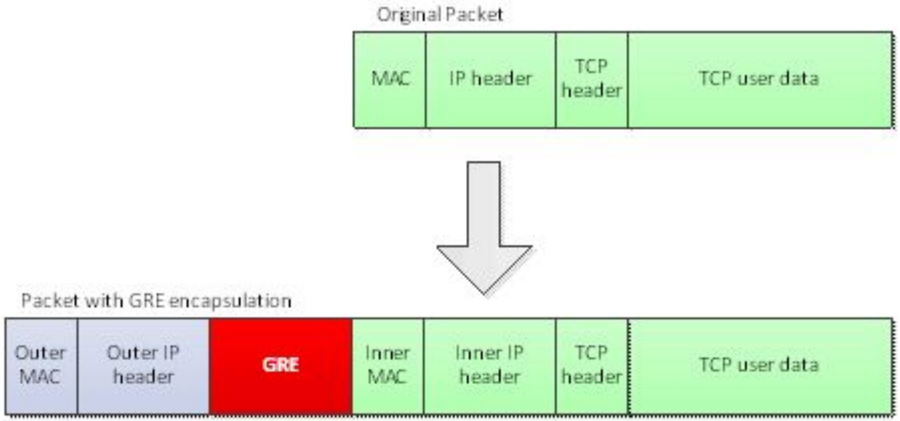
3	ip1
2	ip2
7	ip3

使用虛擬 IP 處理負載平衡 (NLB)



使用虛擬 IP 處理負載平衡 (NLB)





Original Packet

MAC	IP header	TCP header	TCP user data

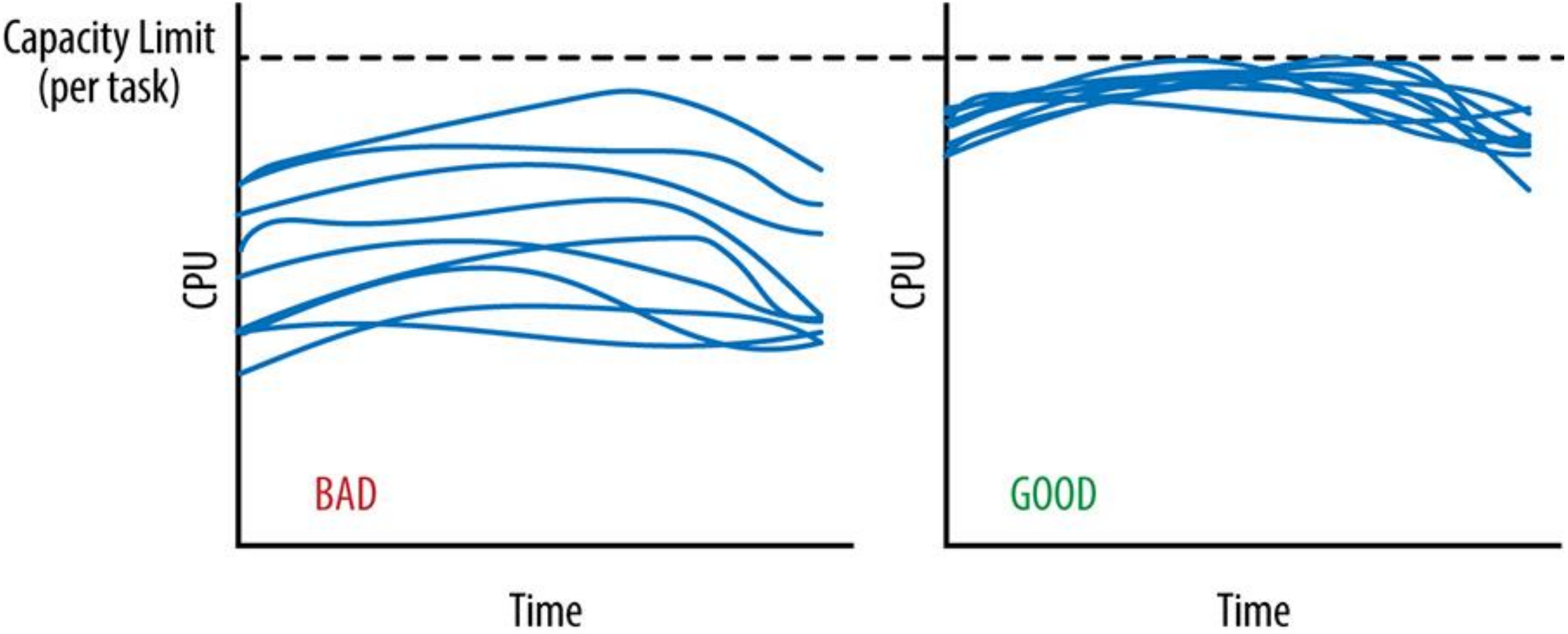
Packet with GRE encapsulation

Outer MAC	Outer IP header	GRE	Inner MAC	Inner IP header	TCP header	TCP user data

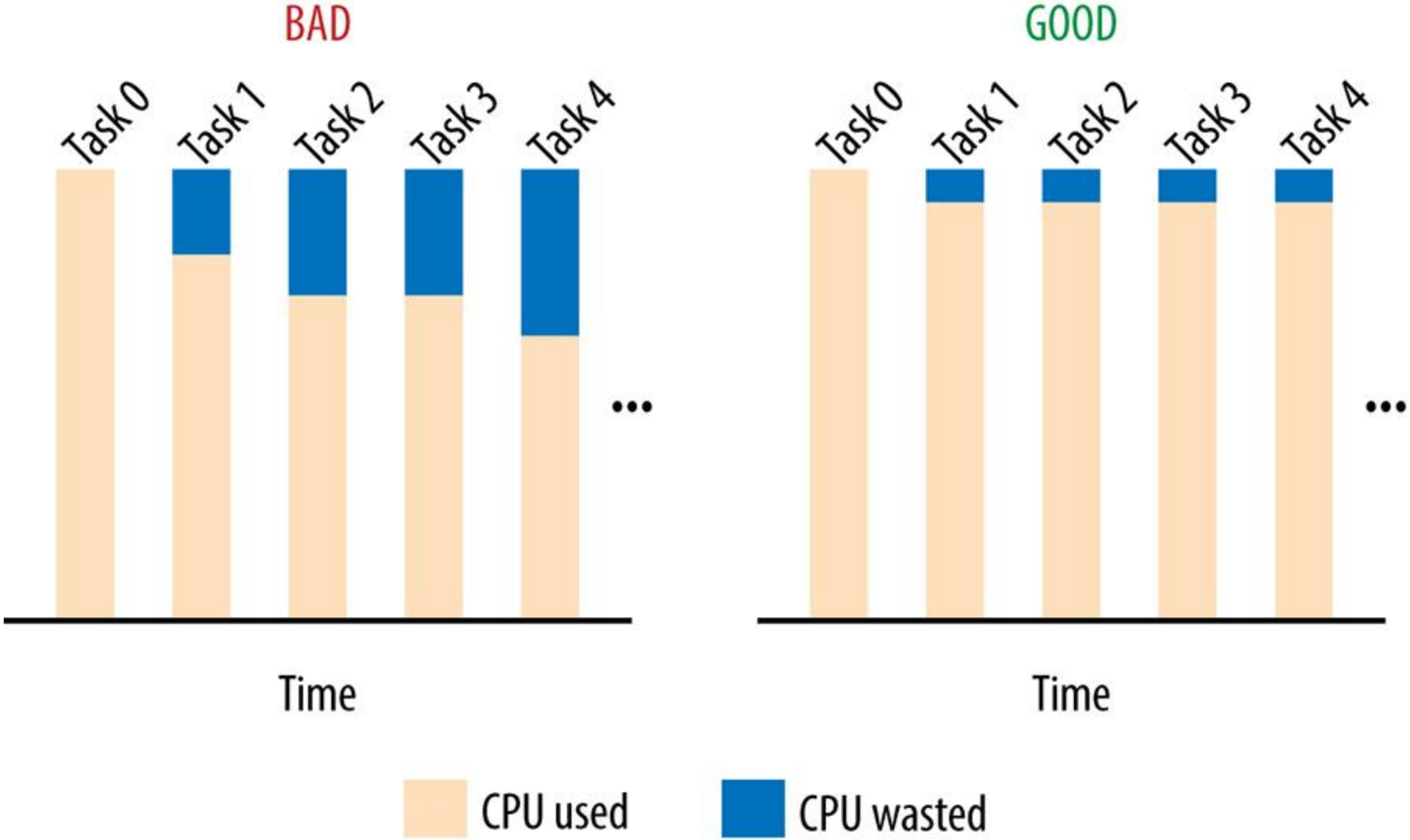
Load Balancing in the Datacenter

Alejandro Forero Cuervo

Per-task Load Distribution



CPU Usage by Task at a Given Time



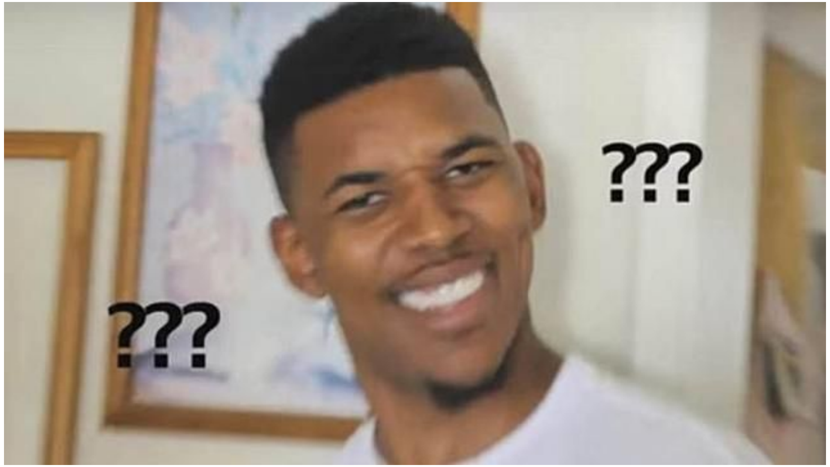
流量控制

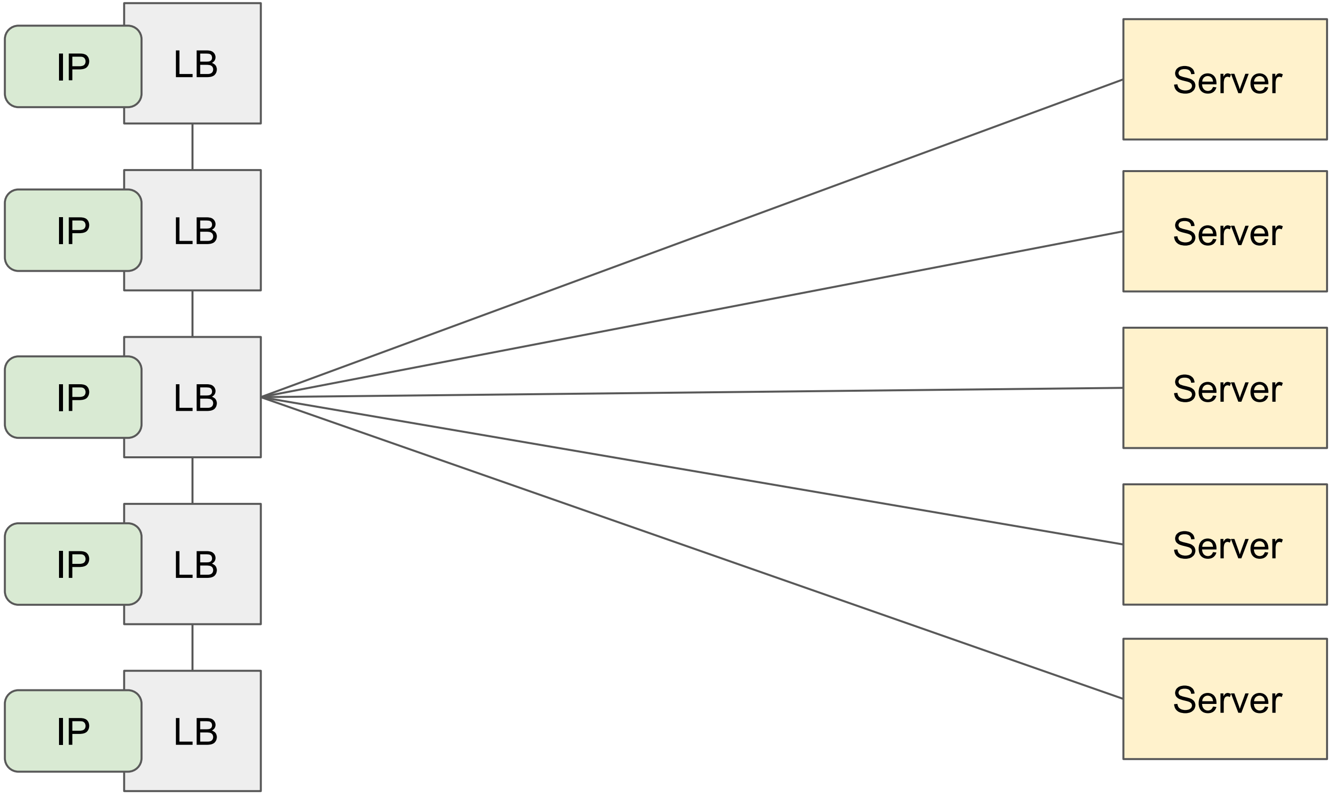
跛腳鴨狀態 Unhealthy Tasks: Lame Duck State

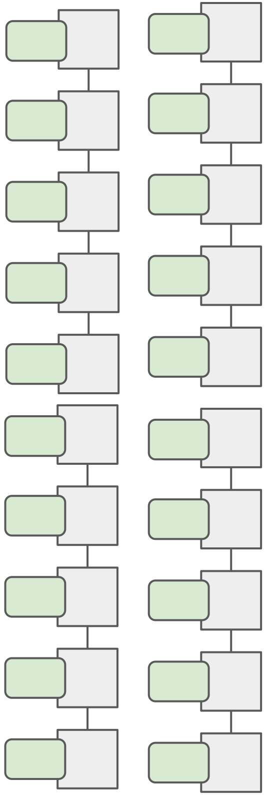
健康、拒絕連接、跛腳鴨狀態(半正常狀態)

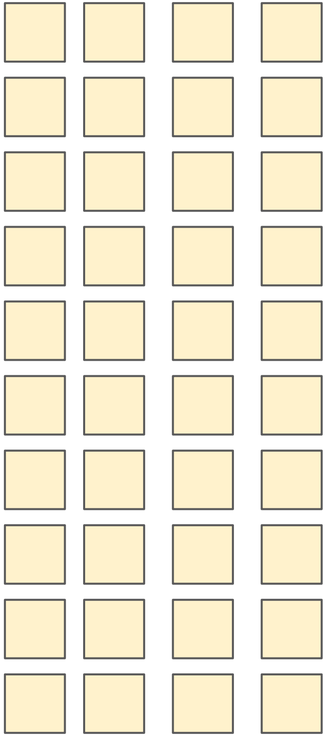
除了健康管理...

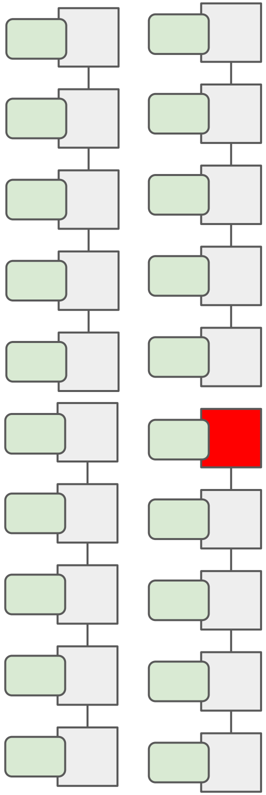
還要做　子集劃分

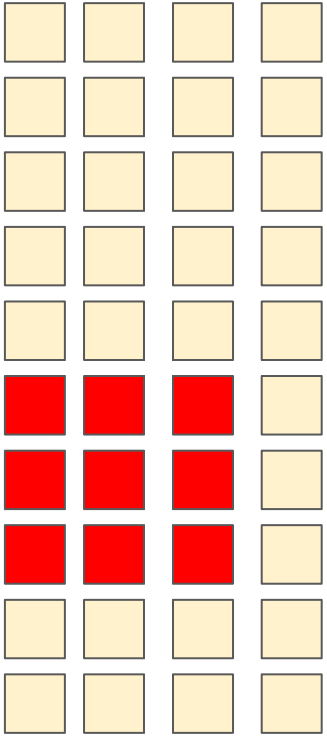



IP
LB
IP
LB
IP
LB
IP
LB
IP
LB
Server
Server
Server
Server
Server

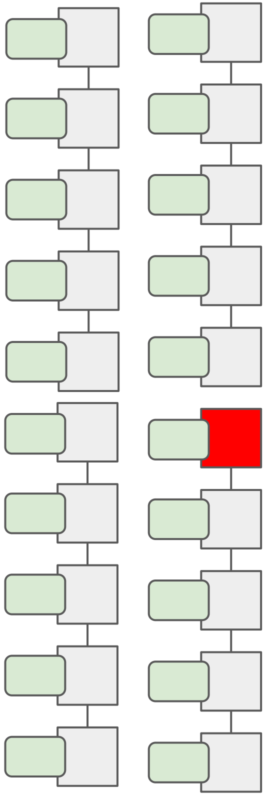




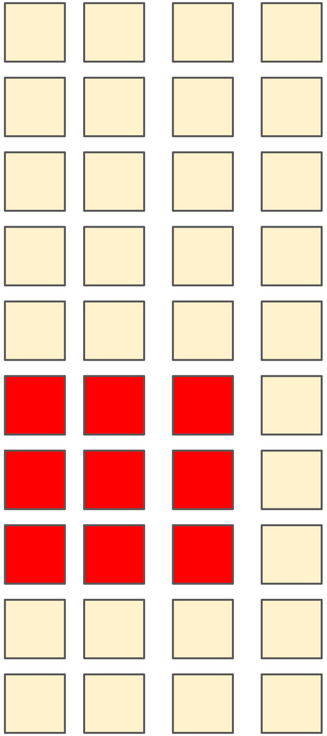




300

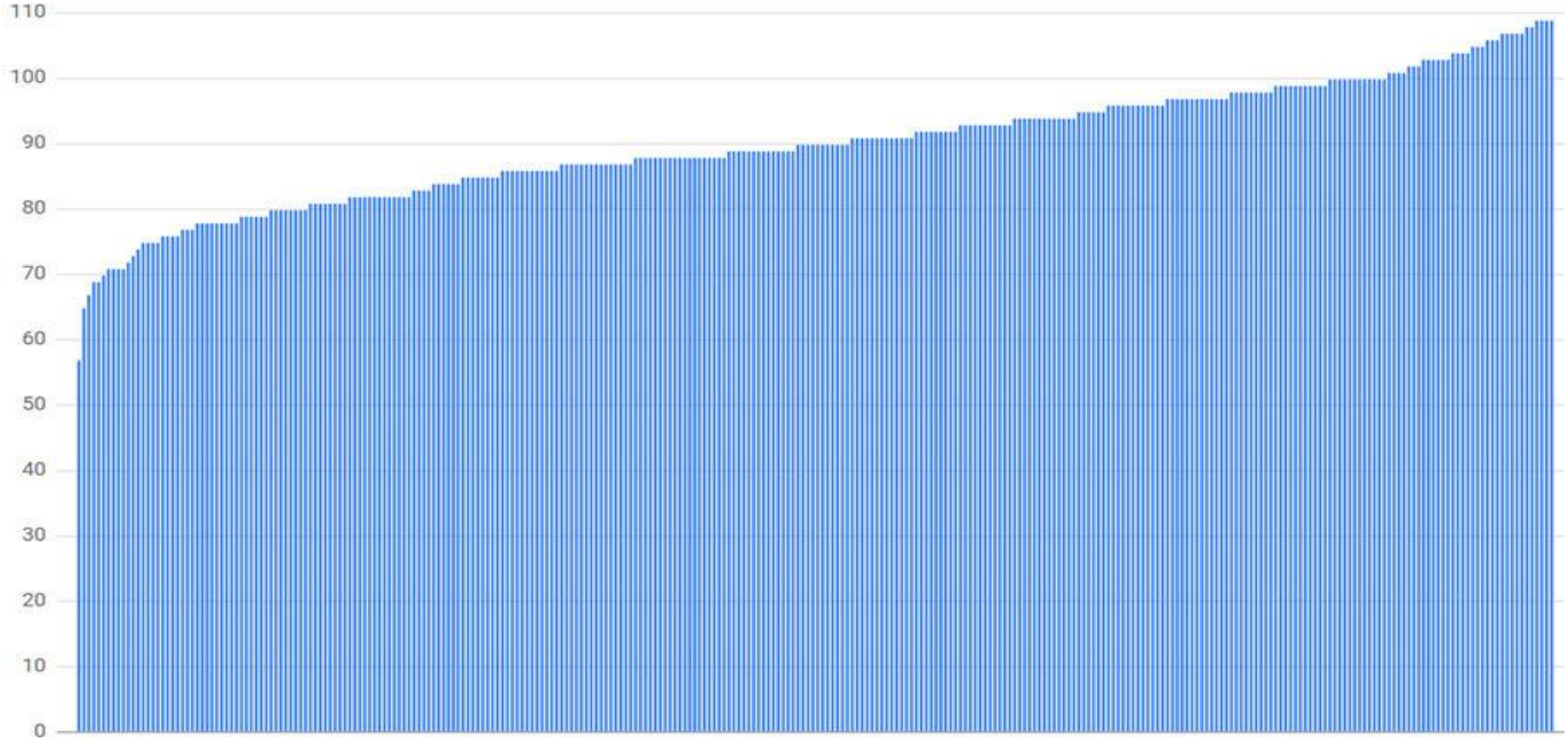


300

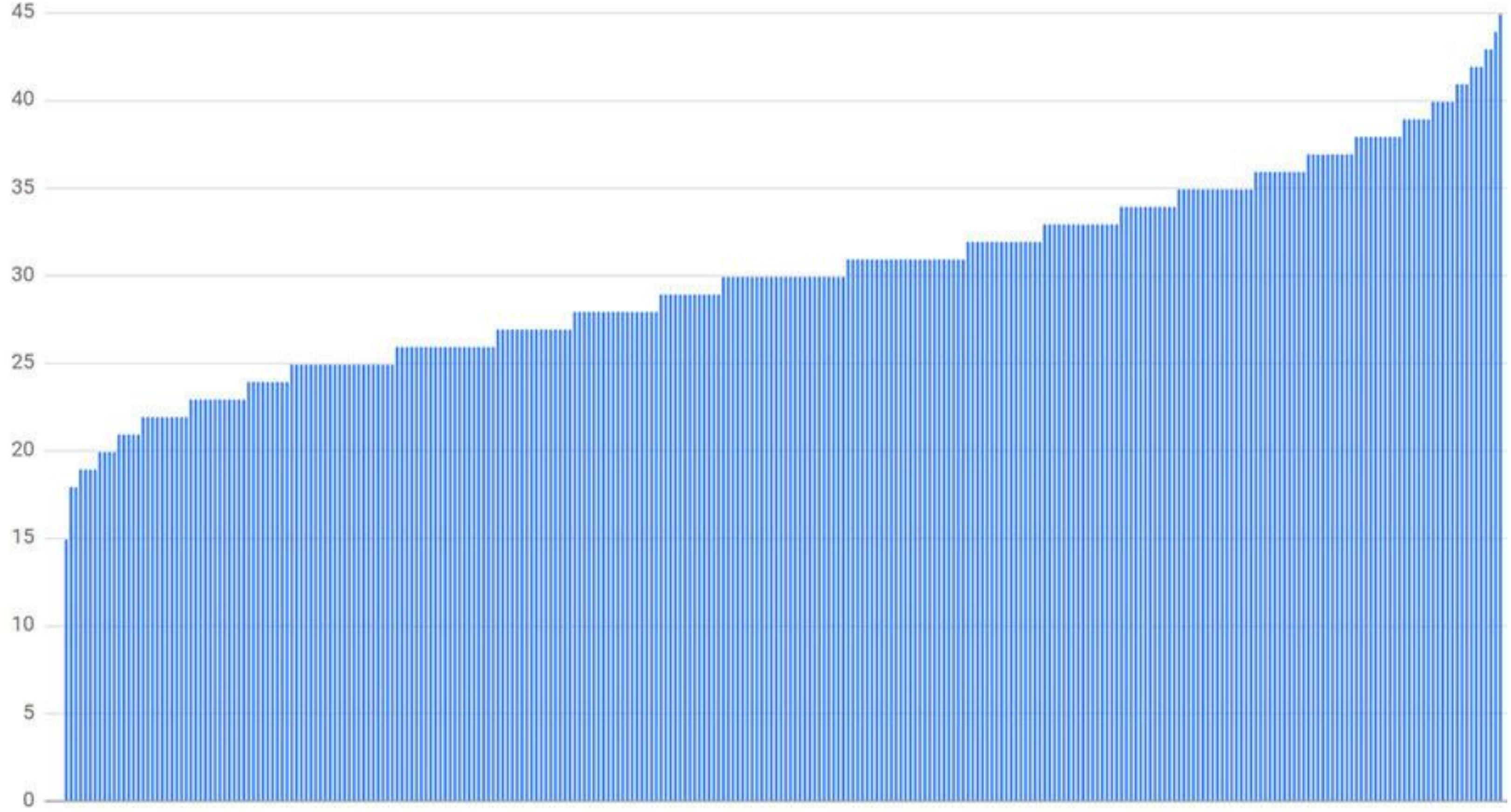


30%

Connection distribution with 300 clients, 300 backends, and a subset size of 30%

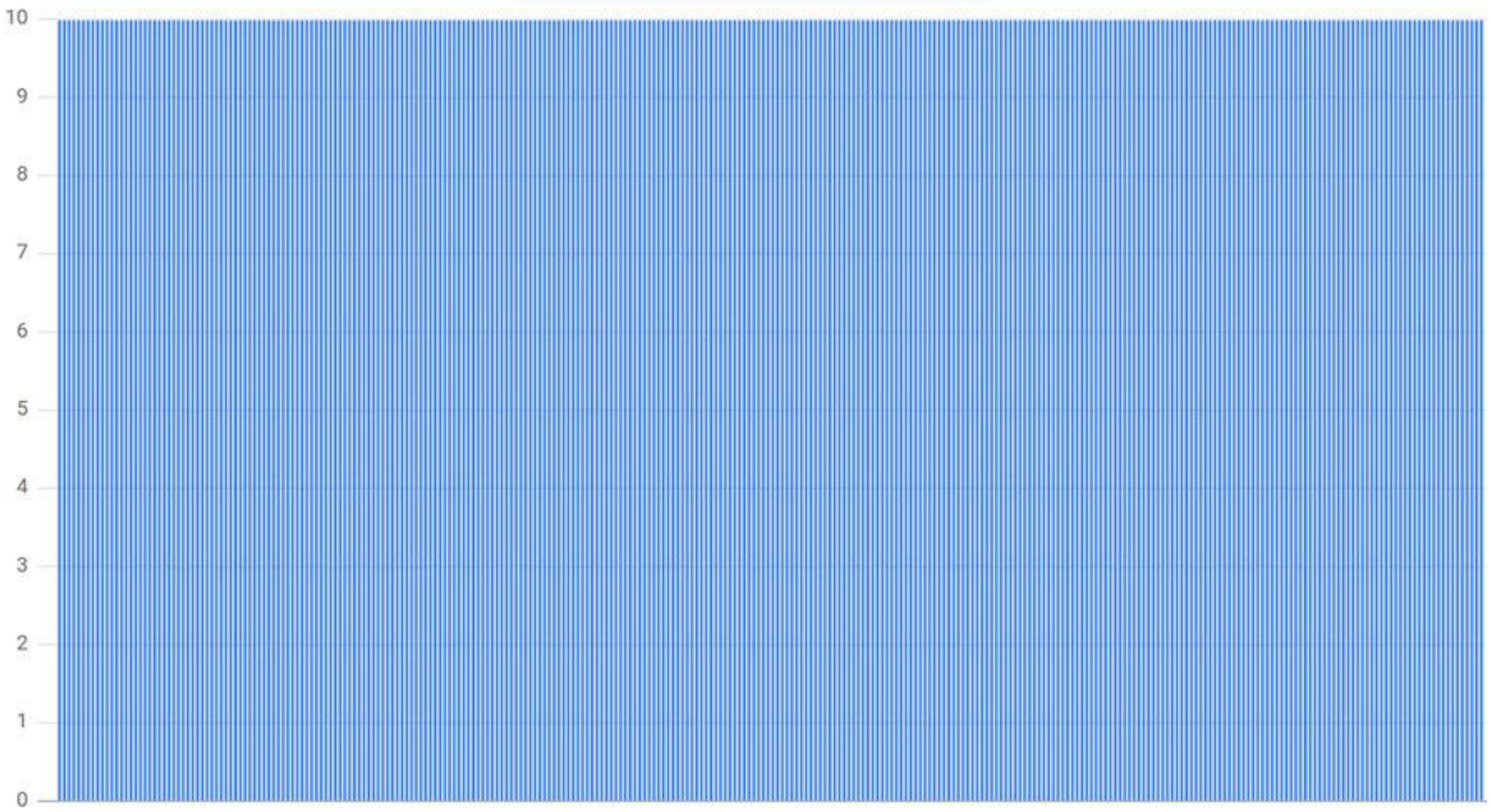


Connection distribution with 300 clients, 300 backends, and a subset size of 10%



```
def Subset(backends, client_id, subset_size):
  subset_count = len(backends) / subset_size

  # Group clients into rounds; each round uses the same shuffled list:
  round = client_id / subset_count
  random.seed(round)
  random.shuffle(backends)

  # The subset id corresponding to the current client:
  subset_id = client_id % subset_count

  start = subset_id * subset_size
  return backends[start:start + subset_size]
```

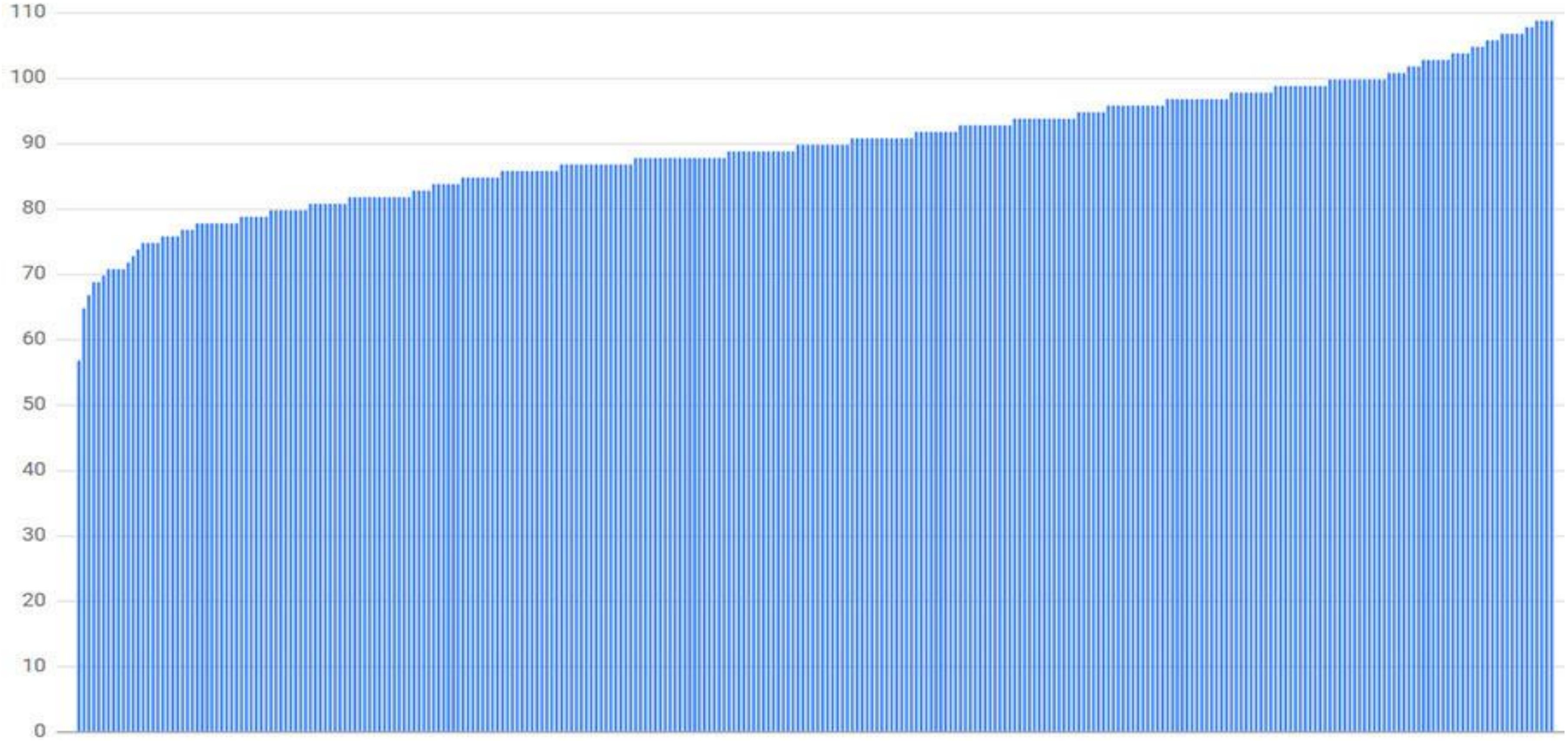
Connection distribution with 300 clients and deterministic subsetting to 10 of 300 backends



```
backend_ids # int array 1 到 300。 代表 backend server 的 id
stat        # dict key: 1 到 300。 value: 被選到的次數，一開始為 0

for client_id in range(0,300):
    random.shuffle(backend_ids)
    for i in range(0,90):   # 30% 子集大小，也就是 90
        stat[backend_ids[i]] = stat[backend_ids[i]] + 1

[ v for v in sorted(stat.values())]
```

```
[66, 67, 71, 71, 73, 73, 74, 74, 75, 75, 75, 75, 75, 76, 76, 77, 77, 77, 77, 78, 78,
78, 78, 78, 78, 78, 78, 78, 79, 79, 79, 79, 79, 79, 80, 80, 80, 81, 81, 81, 81, 81, 81,
81, 82, 82, 82, 82, 82, 82, 82, 83, 83, 83, 83, 83, 83, 83, 83, 83, 83, 83, 84, 84, 84,
84, 84, 84, 84, 84, 84, 84, 84, 84, 85, 85, 85, 85, 85, 85, 85, 85, 85, 85, 85, 85, 85,
85, 86, 86, 86, 86, 86, 86, 86, 86, 86, 87, 87, 87, 87, 87, 87, 87, 87, 87, 87, 87, 88,
88, 88, 88, 88, 88, 88, 88, 88, 88, 88, 88, 88, 88, 89, 89, 89, 89, 89, 89, 89, 89, 89,
89, 89, 89, 89, 89, 90, 90, 90, 90, 90, 90, 90, 90, 90, 90, 90, 90, 90, 90, 91, 91, 91,
91, 91, 91, 91, 91, 91, 91, 91, 91, 91, 91, 91, 91, 91, 91, 91, 91, 91, 91, 92, 92, 92,
92, 92, 92, 92, 92, 92, 92, 92, 92, 92, 92, 92, 92, 92, 92, 93, 93, 93, 93, 93, 93, 93,
93, 93, 93, 93, 93, 93, 93, 93, 94, 94, 94, 94, 94, 94, 94, 95, 95, 95, 95, 95, 95, 95,
95, 95, 95, 95, 96, 96, 96, 96, 96, 96, 96, 96, 96, 96, 96, 96, 97, 97, 97, 97, 97, 97,
97, 97, 97, 97, 97, 98, 98, 98, 98, 98, 98, 98, 98, 98, 98, 98, 98, 99, 99, 99, 99, 99,
99, 99, 99, 100, 100, 100, 100, 100, 101, 101, 101, 101, 101, 101, 101, 101, 101, 102,
103, 103, 103, 103, 103, 103, 103, 104, 104, 105, 105, 106, 106, 107, 108, 108, 109,
111, 111]
```

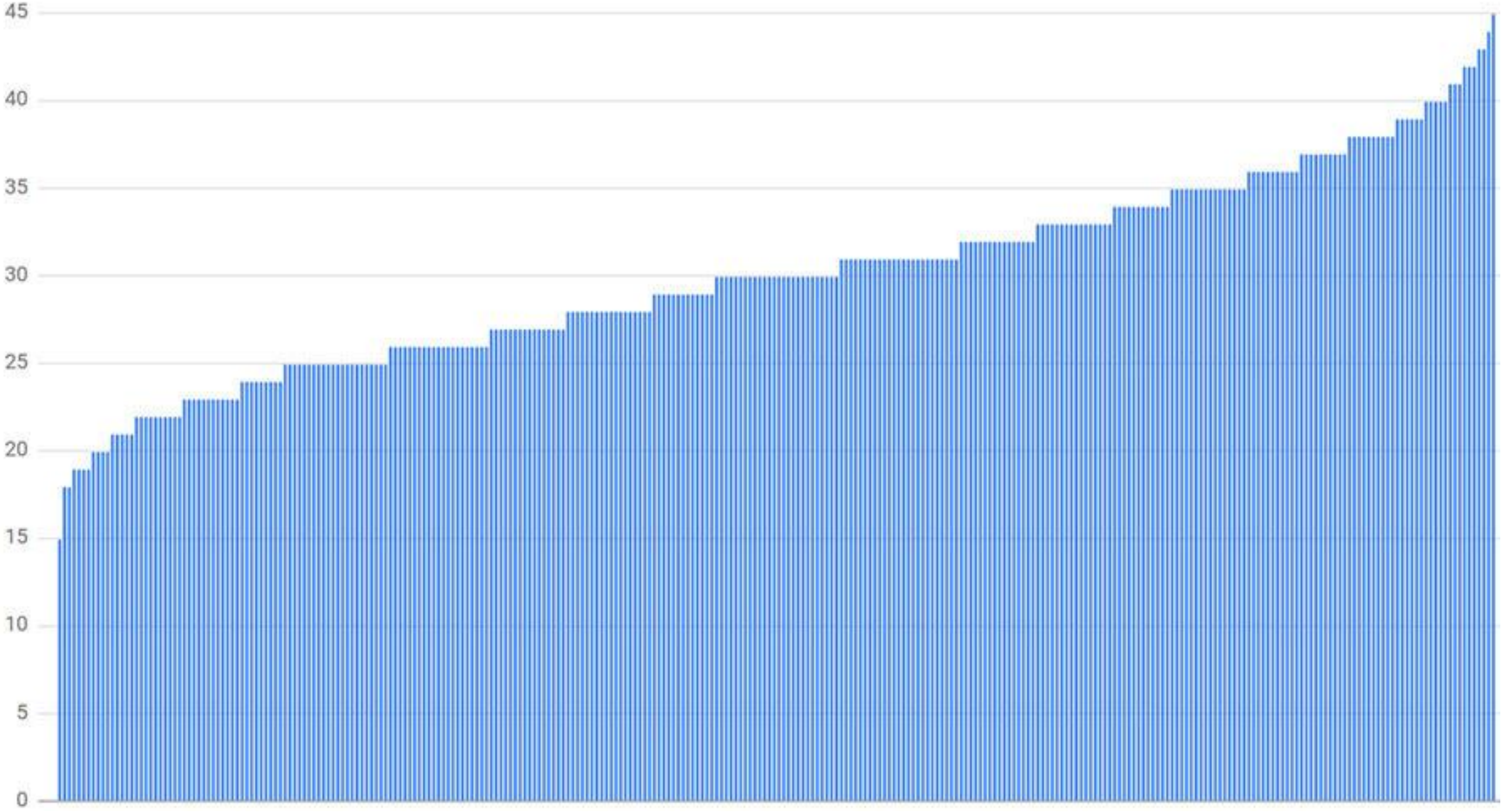
Connection distribution with 300 clients, 300 backends, and a subset size of 30%

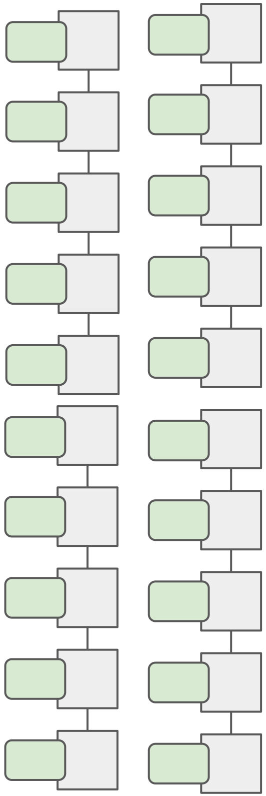


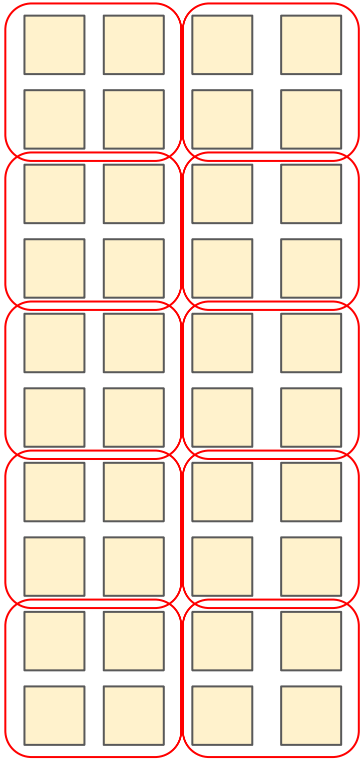
```
backend_ids # int array 1 到 300。 代表 backend server 的 id
stat        # dict key: 1 到 300。 value: 被選到的次數，一開始為 0

for client_id in range(0,300):
    random.shuffle(backend_ids)
    for i in range(0,30):   # 10% 子集大小，也就是 30
        stat[backend_ids[i]] = stat[backend_ids[i]] + 1

[ v for v in sorted(stat.values())]
```

```
[16, 16, 17, 18, 19, 19, 19, 19, 20, 20, 20, 20, 21, 21, 21, 21, 22, 22, 22, 22, 22,
22, 23, 23, 23, 23, 23, 23, 23, 23, 23, 23, 23, 24, 24, 24, 24, 24, 24, 24, 24, 24, 24,
24, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 26, 26, 26, 26,
26, 26, 26, 26, 26, 26, 26, 26, 26, 26, 26, 26, 26, 26, 27, 27, 27, 27, 27, 27, 27, 27,
27, 27, 27, 27, 27, 27, 27, 28, 28, 28, 28, 28, 28, 28, 28, 28, 28, 28, 28, 28, 28, 28,
28, 28, 28, 28, 28, 28, 28, 28, 28, 28, 29, 29, 29, 29, 29, 29, 29, 29, 29, 29, 29, 29,
29, 29, 29, 29, 29, 29, 29, 29, 29, 29, 29, 29, 29, 29, 29, 29, 29, 29, 30, 30, 30, 30,
30, 30, 30, 30, 30, 30, 30, 30, 30, 30, 30, 30, 30, 30, 30, 30, 30, 30, 30, 30, 30, 30,
31, 31, 31, 31, 31, 31, 31, 31, 31, 31, 31, 31, 31, 31, 31, 31, 31, 31, 31, 31, 31, 32,
32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 33, 33, 33, 33, 33, 33, 33, 33, 33,
33, 33, 33, 34, 34, 34, 34, 34, 34, 34, 34, 34, 35, 35, 35, 35, 35, 35, 35, 35, 35, 35,
35, 36, 36, 36, 36, 36, 36, 36, 36, 36, 36, 36, 36, 36, 36, 37, 37, 37, 37, 37, 37, 37,
37, 37, 37, 37, 37, 37, 37, 37, 37, 38, 38, 38, 38, 38, 38, 38, 38, 38, 39, 39, 39, 39,
39, 39, 40, 40, 40, 40, 40, 40, 41, 41, 42, 43, 43, 44, 44]
```

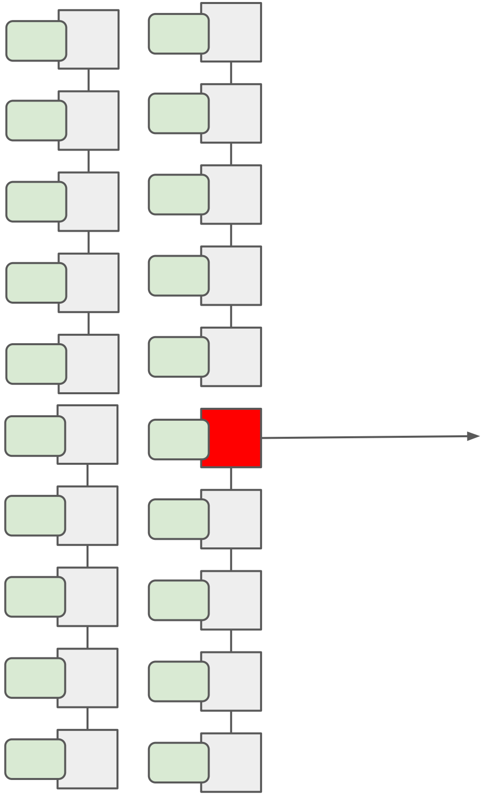
Connection distribution with 300 clients, 300 backends, and a subset size of 10%







如果連線數設成 10 的話....
backend: 300
frontend: 300
總共可以分成 group1~30

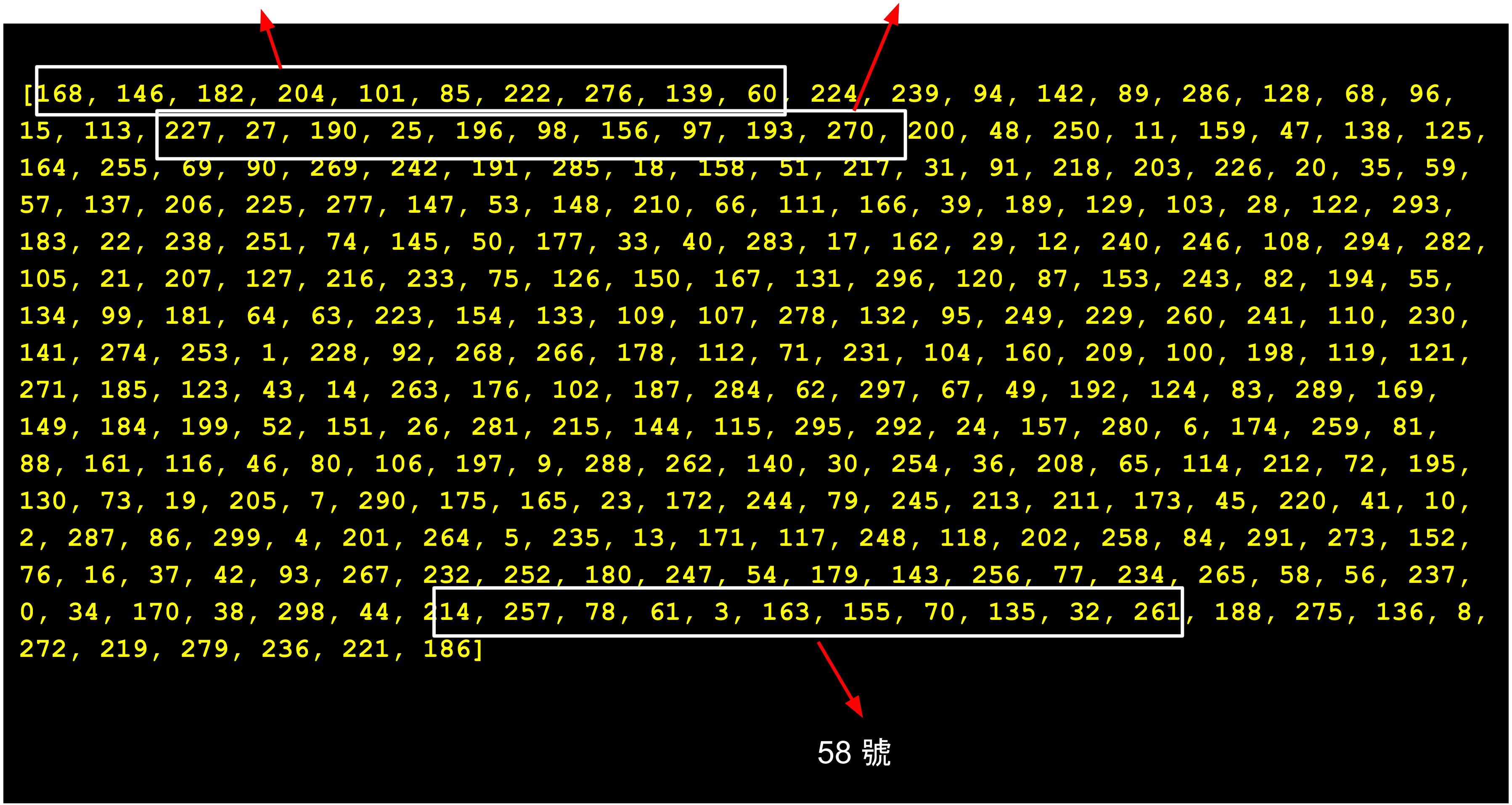


53 號, 50~60 號 都在 group 5

```
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23,
24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45,
46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67,
68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89,
90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108,
109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125,
126, 127, 128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142,
143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 155, 156, 157, 158, 159,
160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171, 172, 173, 174, 175, 176,
177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193,
194, 195, 196, 197, 198, 199, 200, 201, 202, 203, 204, 205, 206, 207, 208, 209, 210,
211, 212, 213, 214, 215, 216, 217, 218, 219, 220, 221, 222, 223, 224, 225, 226, 227,
228, 229, 230, 231, 232, 233, 234, 235, 236, 237, 238, 239, 240, 241, 242, 243, 244,
245, 246, 247, 248, 249, 250, 251, 252, 253, 254, 255, 256, 257, 258, 259, 260, 261,
262, 263, 264, 265, 266, 267, 268, 269, 270, 271, 272, 273, 274, 275, 276, 277, 278,
279, 280, 281, 282, 283, 284, 285, 286, 287, 288, 289, 290, 291, 292, 293, 294, 295,
296, 297, 298, 299]
```

random.seed(5) and shuffle

```
[168, 146, 182, 204, 101, 85, 222, 276, 139, 60, 224, 239, 94, 142, 89, 286, 128, 68, 96,
15, 113, 227, 27, 190, 25, 196, 98, 156, 97, 193, 270, 200, 48, 250, 11, 159, 47, 138, 125,
164, 255, 69, 90, 269, 242, 191, 285, 18, 158, 51, 217, 31, 91, 218, 203, 226, 20, 35, 59,
57, 137, 206, 225, 277, 147, 53, 148, 210, 66, 111, 166, 39, 189, 129, 103, 28, 122, 293,
183, 22, 238, 251, 74, 145, 50, 177, 33, 40, 283, 17, 162, 29, 12, 240, 246, 108, 294, 282,
105, 21, 207, 127, 216, 233, 75, 126, 150, 167, 131, 296, 120, 87, 153, 243, 82, 194, 55,
134, 99, 181, 64, 63, 223, 154, 133, 109, 107, 278, 132, 95, 249, 229, 260, 241, 110, 230,
141, 274, 253, 1, 228, 92, 268, 266, 178, 112, 71, 231, 104, 160, 209, 100, 198, 119, 121,
271, 185, 123, 43, 14, 263, 176, 102, 187, 284, 62, 297, 67, 49, 192, 124, 83, 289, 169,
149, 184, 199, 52, 151, 26, 281, 215, 144, 115, 295, 292, 24, 157, 280, 6, 174, 259, 81,
88, 161, 116, 46, 80, 106, 197, 9, 288, 262, 140, 30, 254, 36, 208, 65, 114, 212, 72, 195,
130, 73, 19, 205, 7, 290, 175, 165, 23, 172, 244, 79, 245, 213, 211, 173, 45, 220, 41, 10,
2, 287, 86, 299, 4, 201, 264, 5, 235, 13, 171, 117, 248, 118, 202, 258, 84, 291, 273, 152,
76, 16, 37, 42, 93, 267, 232, 252, 180, 247, 54, 179, 143, 256, 77, 234, 265, 58, 56, 237,
0, 34, 170, 38, 298, 44, 214, 257, 78, 61, 3, 163, 155, 70, 135, 32, 261, 188, 275, 136, 8,
272, 219, 279, 236, 221, 186]
```



51 號

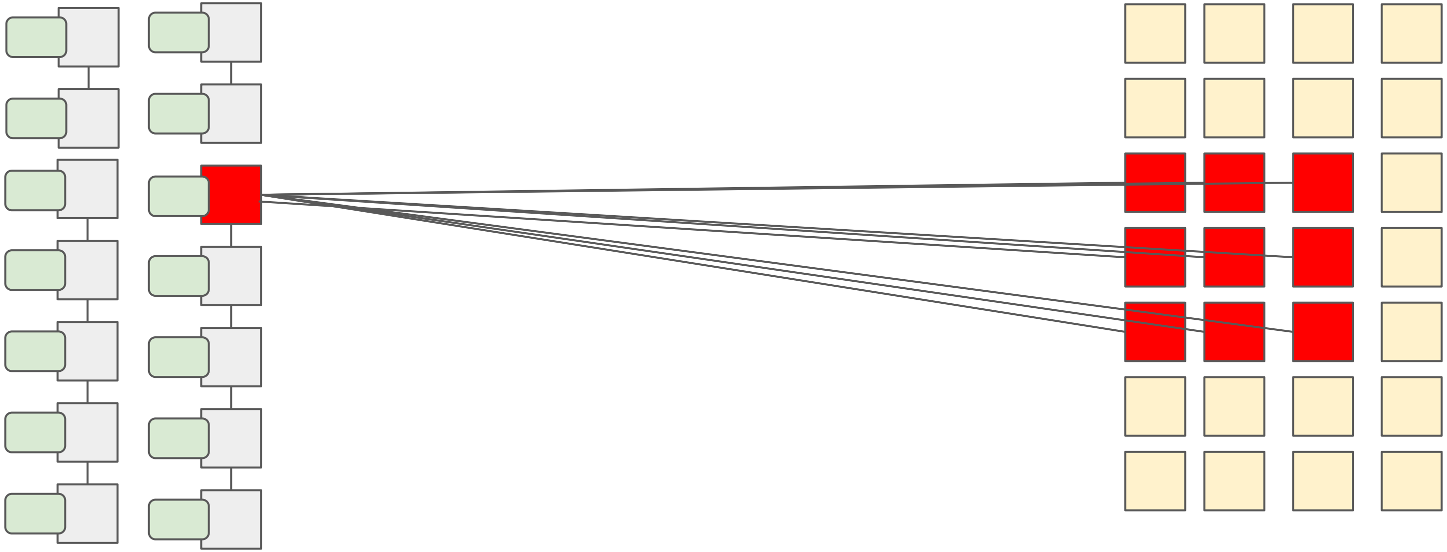
53 號

```
[168, 146, 182, 204, 101, 85, 222, 276, 139, 60, 224, 239, 94, 142, 89, 286, 128, 68, 96,
15, 113, 227, 27, 190, 25, 196, 98, 156, 97, 193, 270, 200, 48, 250, 11, 159, 47, 138, 125,
164, 255, 69, 90, 269, 242, 191, 285, 18, 158, 51, 217, 31, 91, 218, 203, 226, 20, 35, 59,
57, 137, 206, 225, 277, 147, 53, 148, 210, 66, 111, 166, 39, 189, 129, 103, 28, 122, 293,
183, 22, 238, 251, 74, 145, 50, 177, 33, 40, 283, 17, 162, 29, 12, 240, 246, 108, 294, 282,
105, 21, 207, 127, 216, 233, 75, 126, 150, 167, 131, 296, 120, 87, 153, 243, 82, 194, 55,
134, 99, 181, 64, 63, 223, 154, 133, 109, 107, 278, 132, 95, 249, 229, 260, 241, 110, 230,
141, 274, 253, 1, 228, 92, 268, 266, 178, 112, 71, 231, 104, 160, 209, 100, 198, 119, 121,
271, 185, 123, 43, 14, 263, 176, 102, 187, 284, 62, 297, 67, 49, 192, 124, 83, 289, 169,
149, 184, 199, 52, 151, 26, 281, 215, 144, 115, 295, 292, 24, 157, 280, 6, 174, 259, 81,
88, 161, 116, 46, 80, 106, 197, 9, 288, 262, 140, 30, 254, 36, 208, 65, 114, 212, 72, 195,
130, 73, 19, 205, 7, 290, 175, 165, 23, 172, 244, 79, 245, 213, 211, 173, 45, 220, 41, 10,
2, 287, 86, 299, 4, 201, 264, 5, 235, 13, 171, 117, 248, 118, 202, 258, 84, 291, 273, 152,
76, 16, 37, 42, 93, 267, 232, 252, 180, 247, 54, 179, 143, 256, 77, 234, 265, 58, 56, 237,
0, 34, 170, 38, 298, 44, 214, 257, 78, 61, 3, 163, 155, 70, 135, 32, 261, 188, 275, 136, 8,
272, 219, 279, 236, 221, 186]
```

58 號

Load Balancing Policies

我們已經知道了怎麼維持和管理健康的連線...



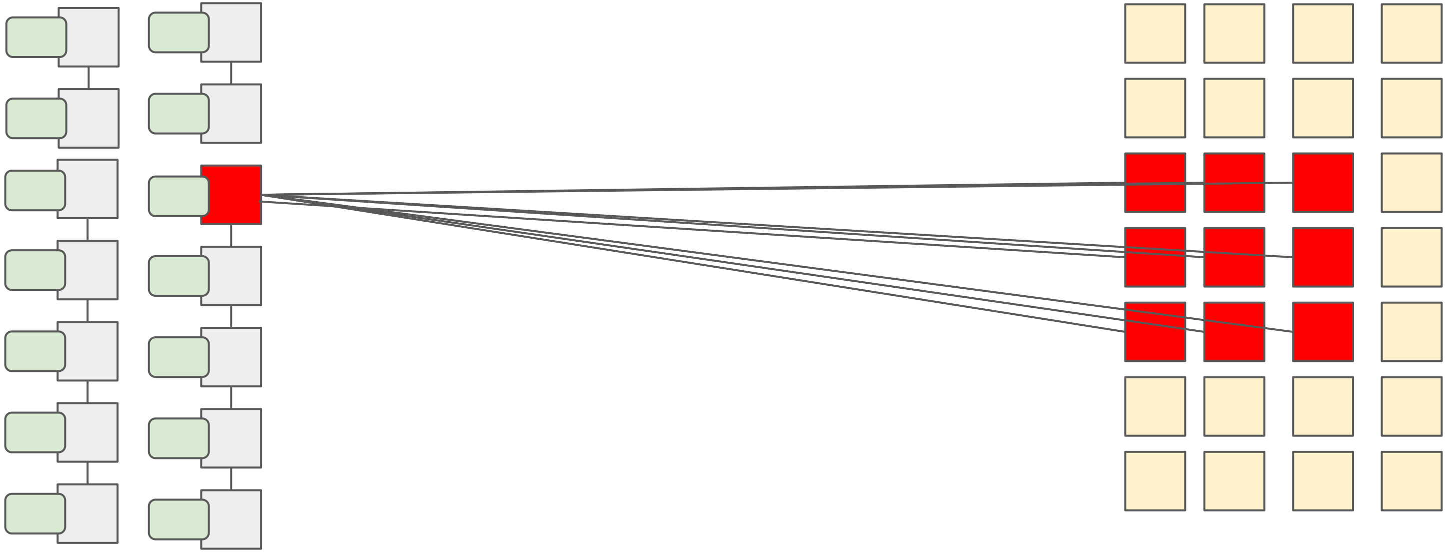
Simple Round Robin

可能會造成的問題

所謂的問題是...

Load Balancing Policies

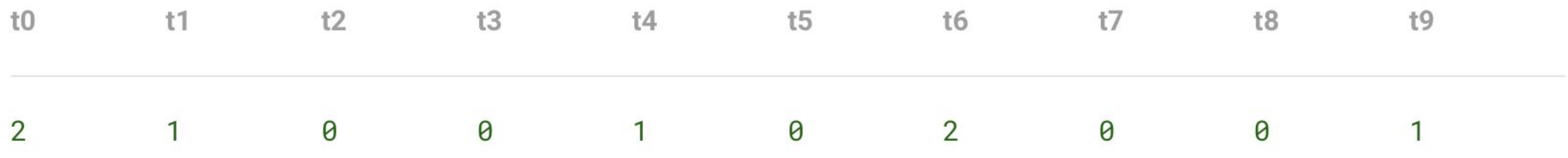
右邊九個 backend 平均的資源利用 (e.g. CPU)



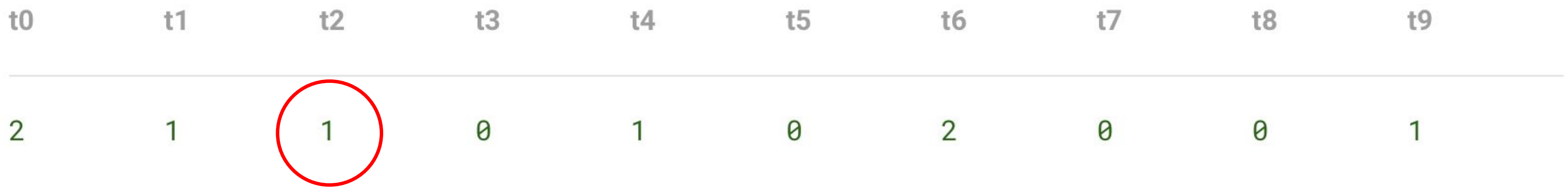
Simple Round Robin

- 子集太小
- 每個 Request 成本不同
- 不同等級的後端 Server
- 其他因素:
    - Run 在 Server 上程序的效能特性不同

Least-Loaded Round Robin

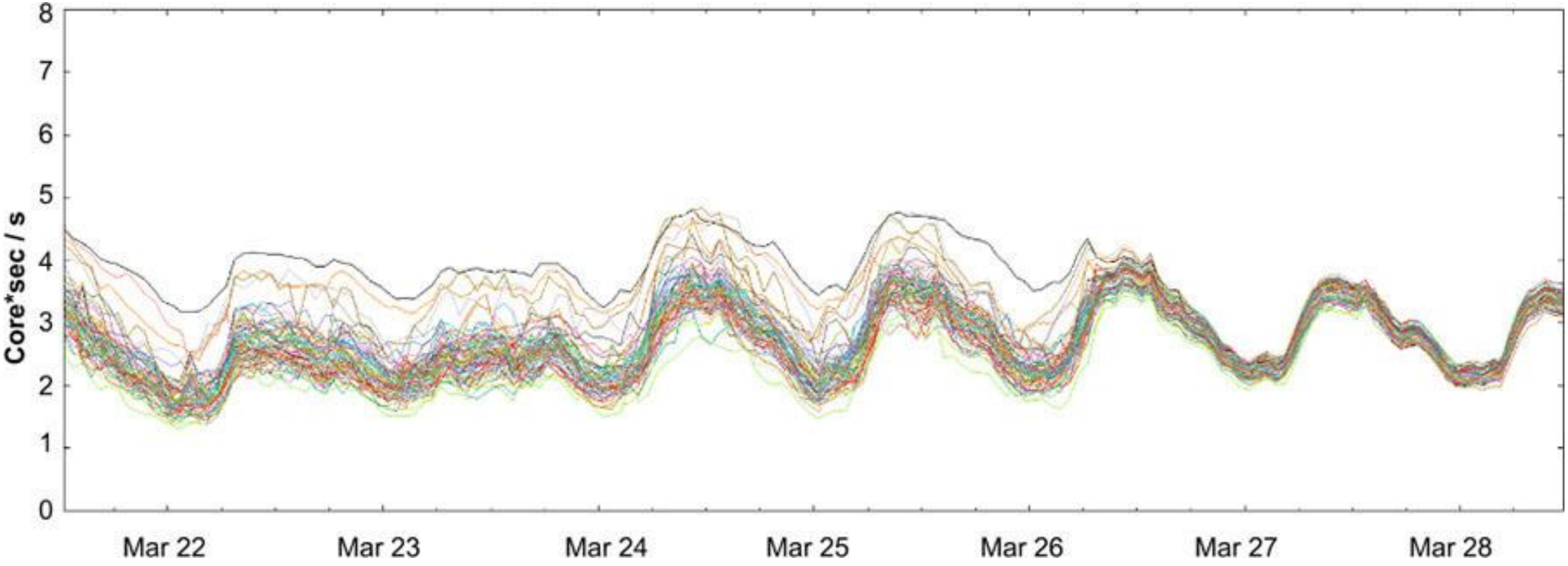


t0	t1	t2	t3	t4	t5	t6	t7	t8	t9
2	1	0	0	1	0	2	0	0	1



t0	t1	t2	t3	t4	t5	t6	t7	t8	t9
2	1	1	0	1	0	2	0	0	1

Weighted Round Robin


Core*sec / s
Mar 22
Mar 23
Mar 24
Mar 25
Mar 26
Mar 27
Mar 28