LLM-CCA

This document is intended for IoT lab researchers. For details, please refer to the LLM-CCA documentation.

This document is used to quickly build the experimental structure, not to quickly complete all experiments.

Preparation

There are 3 VMs in total:

VM_ BBR2: Contains BBR2 congestion control algorithm

VM_ BBR3: Contains BB3 congestion control algorithm

VM_L4S: Contains everything about the L4S architecture, but only for the VMs for Client and Server (**TCP Prague congestion control algorithm and ECN enabled**). Routers need to manually install the L4S architecture and enable **DualPI2 AQM**.

The VM backup is only accessible to the Deakin account, and I don't have permission to change it to anyone else.

VM_BBR2 OneDrive link: VM bbr2

VM_BBR3 OneDrive link: VM bbr3

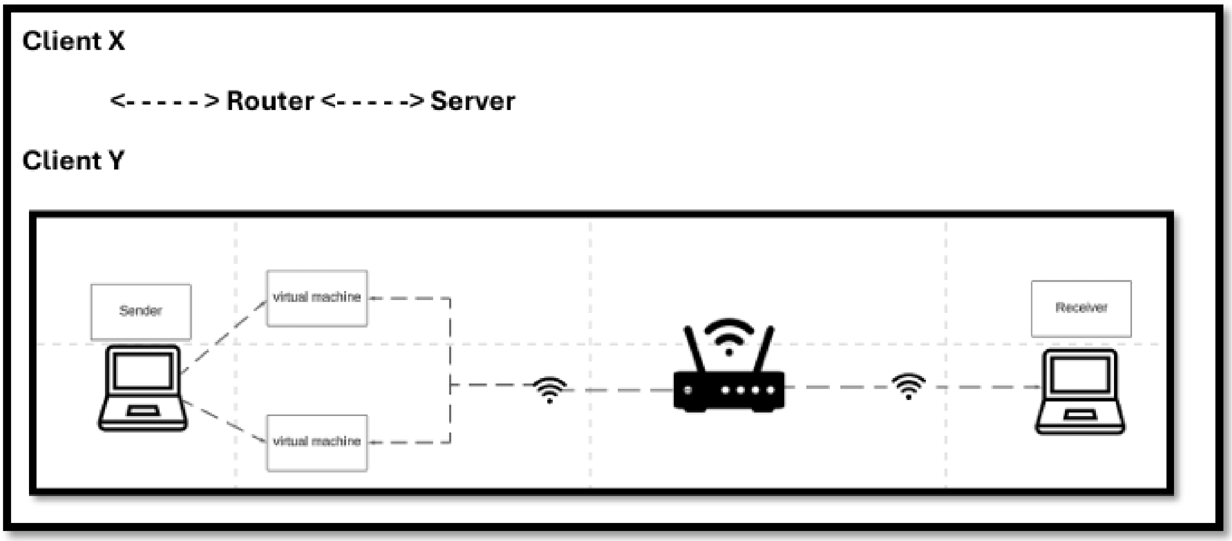
VM_L4S OneDrive link: Ubuntu 64 prague

The su password and login password for the VM are both **SIT7232024**

Process

Hardware: 2 Windows computers (PC1 and PC2), 1 Ubuntu computer or Linux-supported router.

Software: VMware workstation Pro (17th generation is a personal free version, I use 16th generation, any differences need to be solved by yourself.)



Client X

<-----> Router <-----> Server

Client Y

Roles: Client, Router, Server.

Client X is a client that supports L4S architecture, and Client Y is a client that does not support L4S architecture.

Client X uses VM_L4S, and Client Y uses VM_BBR2, or VM_BBR3 according to your experimental requirements.

Router requires you to install L4S architecture on MikroTik router or Ubuntu system computer.

Server uses VM_L4S.

Open VMware workstation Pro on PC1 and PC2 and start VM_L4S on both VMware.

Enter the following command in the VMs terminal

Server side：

iperf3 -s -p 3000

iperf3 -s -p 3001

Client X side：

iperf3 -c [ip address] -p 3000 -C prague -tinf

Client Y side：

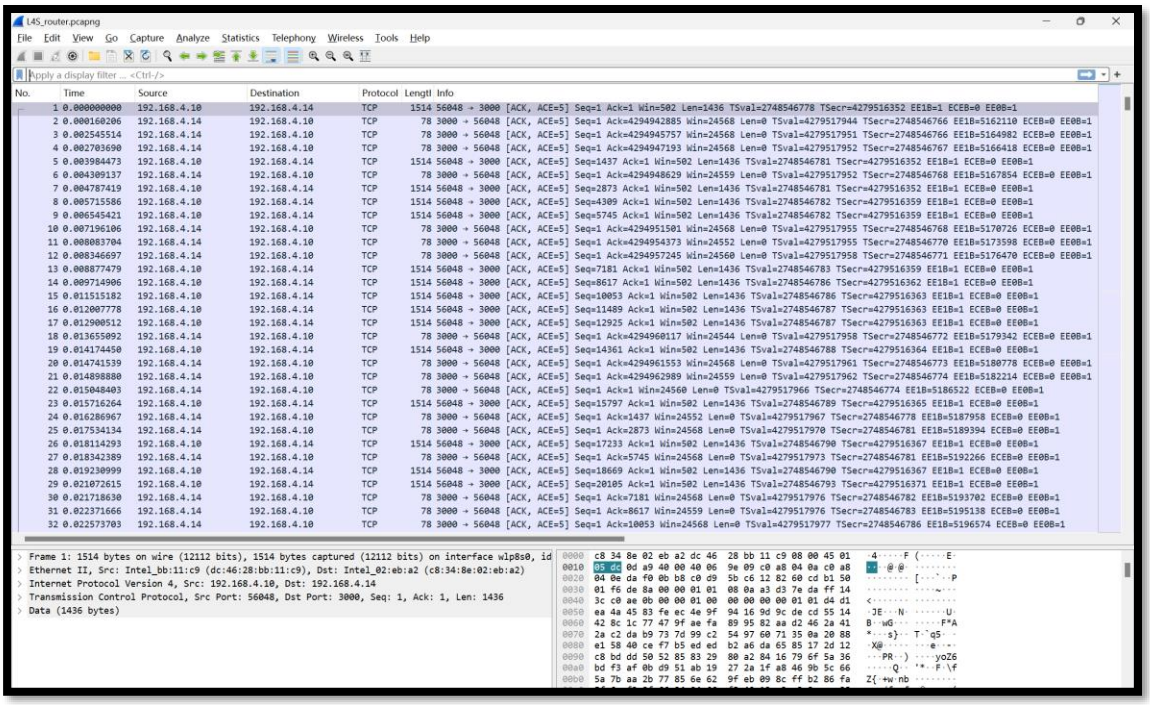
iperf3 -c [ip address] -p 3001 -C cubic -tinf

[ip address] is the IP address of your Server side.

Launch Wireshark on Router or Server side, to check L4S is enable or not.

After iperf3 connects properly, please keep Clients connected to Server. Then start Wireshark on Server side or Router side, after opening Wireshark enter

ip.dsfield.ecn == 3

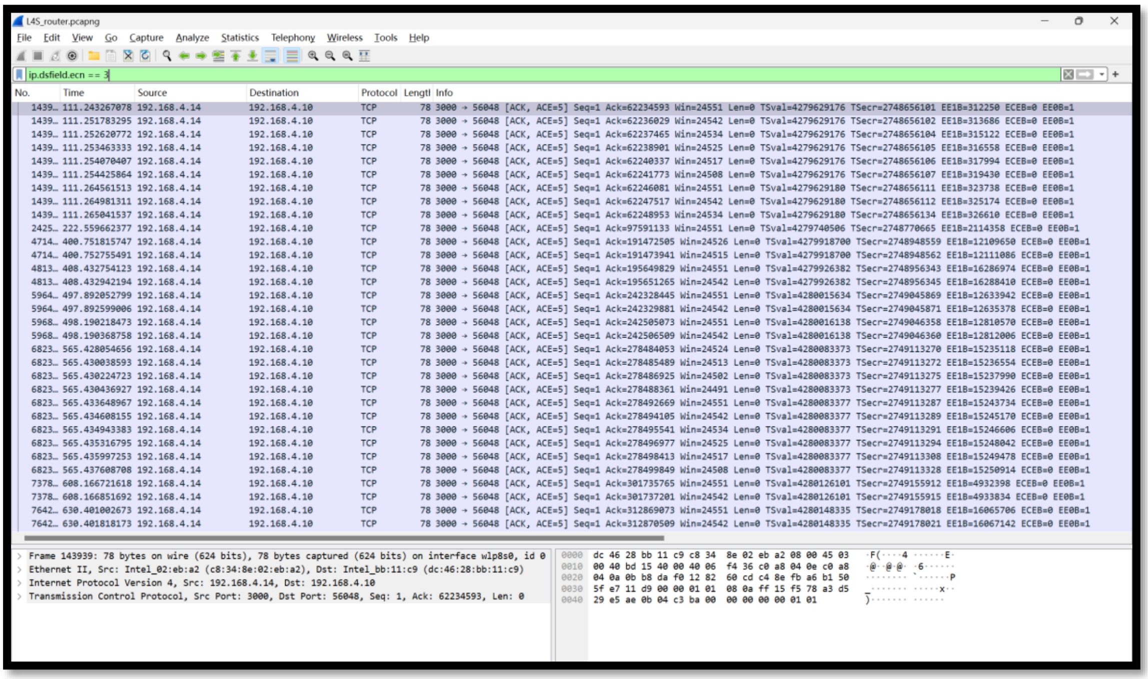


```
No.     Time         Source          Destination     Protocol Lengtl Info
      1 0.000000000  192.168.4.10    192.168.4.14    TCP      1514 56048 → 3000 [ACK, ACE=5] Seq=1 Ack=1 Win=502 Len=1436 TSval=2748546778 TSecr=4279516352 EE1B=1 ECEB=0 EE0B=1
      2 0.000160206  192.168.4.14    192.168.4.10    TCP        78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=4294942885 Win=24568 Len=0 TSval=4279517944 TSecr=2748546766 EE1B=5162110 ECEB=0 EE0B=1
      3 0.002545514  192.168.4.14    192.168.4.10    TCP        78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=4294945757 Win=24568 Len=0 TSval=4279517951 TSecr=2748546766 EE1B=5164982 ECEB=0 EE0B=1
      4 0.002703690  192.168.4.14    192.168.4.10    TCP        78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=4294947193 Win=24568 Len=0 TSval=4279517952 TSecr=2748546767 EE1B=5166418 ECEB=0 EE0B=1
      5 0.003984473  192.168.4.10    192.168.4.14    TCP      1514 56048 → 3000 [ACK, ACE=5] Seq=1437 Ack=1 Win=502 Len=1436 TSval=2748546781 TSecr=4279516352 EE1B=1 ECEB=0 EE0B=1
      6 0.004309137  192.168.4.14    192.168.4.10    TCP        78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=4294948629 Win=24559 Len=0 TSval=4279517952 TSecr=2748546768 EE1B=5167854 ECEB=0 EE0B=1
      7 0.004787419  192.168.4.10    192.168.4.14    TCP      1514 56048 → 3000 [ACK, ACE=5] Seq=2873 Ack=1 Win=502 Len=1436 TSval=2748546781 TSecr=4279516352 EE1B=1 ECEB=0 EE0B=1
      8 0.005715586  192.168.4.10    192.168.4.14    TCP      1514 56048 → 3000 [ACK, ACE=5] Seq=4309 Ack=1 Win=502 Len=1436 TSval=2748546782 TSecr=4279516359 EE1B=1 ECEB=0 EE0B=1
      9 0.006545421  192.168.4.10    192.168.4.14    TCP      1514 56048 → 3000 [ACK, ACE=5] Seq=5745 Ack=1 Win=502 Len=1436 TSval=2748546782 TSecr=4279516359 EE1B=1 ECEB=0 EE0B=1
     10 0.007196106  192.168.4.14    192.168.4.10    TCP        78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=4294951501 Win=24568 Len=0 TSval=4279517955 TSecr=2748546768 EE1B=5170726 ECEB=0 EE0B=1
     11 0.008083704  192.168.4.14    192.168.4.10    TCP        78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=4294954373 Win=24552 Len=0 TSval=4279517955 TSecr=2748546770 EE1B=5173598 ECEB=0 EE0B=1
     12 0.008346697  192.168.4.14    192.168.4.10    TCP        78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=4294957245 Win=24560 Len=0 TSval=4279517958 TSecr=2748546771 EE1B=5176470 ECEB=0 EE0B=1
     13 0.008877479  192.168.4.10    192.168.4.14    TCP      1514 56048 → 3000 [ACK, ACE=5] Seq=7181 Ack=1 Win=502 Len=1436 TSval=2748546783 TSecr=4279516359 EE1B=1 ECEB=0 EE0B=1
     14 0.009714906  192.168.4.10    192.168.4.14    TCP      1514 56048 → 3000 [ACK, ACE=5] Seq=8617 Ack=1 Win=502 Len=1436 TSval=2748546786 TSecr=4279516362 EE1B=1 ECEB=0 EE0B=1
     15 0.011515182  192.168.4.10    192.168.4.14    TCP      1514 56048 → 3000 [ACK, ACE=5] Seq=10053 Ack=1 Win=502 Len=1436 TSval=2748546786 TSecr=4279516363 EE1B=1 ECEB=0 EE0B=1
     16 0.012007778  192.168.4.10    192.168.4.14    TCP      1514 56048 → 3000 [ACK, ACE=5] Seq=11489 Ack=1 Win=502 Len=1436 TSval=2748546787 TSecr=4279516363 EE1B=1 ECEB=0 EE0B=1
     17 0.012900512  192.168.4.10    192.168.4.14    TCP      1514 56048 → 3000 [ACK, ACE=5] Seq=12925 Ack=1 Win=502 Len=1436 TSval=2748546787 TSecr=4279516363 EE1B=1 ECEB=0 EE0B=1
     18 0.013655092  192.168.4.14    192.168.4.10    TCP        78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=4294960117 Win=24544 Len=0 TSval=4279517958 TSecr=2748546772 EE1B=5179342 ECEB=0 EE0B=1
     19 0.014174450  192.168.4.10    192.168.4.14    TCP      1514 56048 → 3000 [ACK, ACE=5] Seq=14361 Ack=1 Win=502 Len=1436 TSval=2748546788 TSecr=4279516364 EE1B=1 ECEB=0 EE0B=1
     20 0.014741539  192.168.4.14    192.168.4.10    TCP        78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=4294961553 Win=24568 Len=0 TSval=4279517961 TSecr=2748546773 EE1B=5180778 ECEB=0 EE0B=1
     21 0.014898880  192.168.4.14    192.168.4.10    TCP        78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=4294962989 Win=24559 Len=0 TSval=4279517962 TSecr=2748546774 EE1B=5182214 ECEB=0 EE0B=1
     22 0.015048403  192.168.4.14    192.168.4.10    TCP        78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=1 Win=24560 Len=0 TSval=4279517966 TSecr=2748546774 EE1B=5186522 ECEB=0 EE0B=1
     23 0.015716264  192.168.4.10    192.168.4.14    TCP      1514 56048 → 3000 [ACK, ACE=5] Seq=15797 Ack=1 Win=502 Len=1436 TSval=2748546789 TSecr=4279516365 EE1B=1 ECEB=0 EE0B=1
     24 0.016286967  192.168.4.14    192.168.4.10    TCP        78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=1437 Win=24552 Len=0 TSval=4279517967 TSecr=2748546778 EE1B=5187958 ECEB=0 EE0B=1
     25 0.017534134  192.168.4.14    192.168.4.10    TCP        78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=2873 Win=24568 Len=0 TSval=4279517970 TSecr=2748546781 EE1B=5189394 ECEB=0 EE0B=1
     26 0.018114293  192.168.4.10    192.168.4.14    TCP      1514 56048 → 3000 [ACK, ACE=5] Seq=17233 Ack=1 Win=502 Len=1436 TSval=2748546790 TSecr=4279516367 EE1B=1 ECEB=0 EE0B=1
     27 0.018342389  192.168.4.14    192.168.4.10    TCP        78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=5745 Win=24568 Len=0 TSval=4279517973 TSecr=2748546781 EE1B=5192266 ECEB=0 EE0B=1
     28 0.019230999  192.168.4.10    192.168.4.14    TCP      1514 56048 → 3000 [ACK, ACE=5] Seq=18669 Ack=1 Win=502 Len=1436 TSval=2748546790 TSecr=4279516367 EE1B=1 ECEB=0 EE0B=1
     29 0.021072615  192.168.4.10    192.168.4.14    TCP      1514 56048 → 3000 [ACK, ACE=5] Seq=20105 Ack=1 Win=502 Len=1436 TSval=2748546793 TSecr=4279516371 EE1B=1 ECEB=0 EE0B=1
     30 0.021718630  192.168.4.14    192.168.4.10    TCP        78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=7181 Win=24568 Len=0 TSval=4279517976 TSecr=2748546782 EE1B=5193702 ECEB=0 EE0B=1
     31 0.022371666  192.168.4.14    192.168.4.10    TCP        78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=8617 Win=24559 Len=0 TSval=4279517976 TSecr=2748546783 EE1B=5195138 ECEB=0 EE0B=1
     32 0.022573703  192.168.4.14    192.168.4.10    TCP        78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=10053 Win=24568 Len=0 TSval=4279517977 TSecr=2748546786 EE1B=5196574 ECEB=0 EE0B=1
```

```
> Frame 1: 1514 bytes on wire (12112 bits), 1514 bytes captured (12112 bits) on interface wlp8s0, id
> Ethernet II, Src: Intel_bb:11:c9 (dc:46:28:bb:11:c9), Dst: Intel_02:eb:a2 (c8:34:8e:02:eb:a2)
> Internet Protocol Version 4, Src: 192.168.4.10, Dst: 192.168.4.14
> Transmission Control Protocol, Src Port: 56048, Dst Port: 3000, Seq: 1, Ack: 1, Len: 1436
> Data (1436 bytes)
```

```
0000  c8 34 8e 02 eb a2 dc 46  28 bb 11 c9 08 00 45 01   ·4·····F (·····E·
0010  05 dc 0d a9 40 00 40 06  9e 09 c0 a8 04 0a c0 a8   ····@·@· ········
0020  04 0e da f0 0b b8 c0 d9  5b c6 12 82 60 cd b1 50   ········ [···`··P
0030  01 f6 de 8a 00 00 01 01  08 0a a3 d3 7e da ff 14   ········ ····~···
0040  3c c0 ae 0b 00 00 01 00  00 00 00 00 01 01 d4 d1   <······· ········
0050  ea 4a 45 83 fe ec 4e 9f  94 16 9d 9c de cd 55 14   ·JE···N· ······U·
0060  42 8c 1c 77 47 9f ae fa  89 95 82 aa d2 46 2a 41   B··wG··· ·····F*A
0070  2a c2 da b9 73 7d 99 c2  54 97 60 71 35 0a 20 88   *···s}·· T·`q5· ·
0080  e1 58 40 ce f7 b5 ed ed  b2 a6 da 65 85 17 2d 12   ·X@····· ···e··-·
0090  c8 bd dd 50 52 85 83 29  80 a2 84 16 79 6f 5a 36   ···PR··) ····yoZ6
00a0  bd f3 af 0b d9 51 ab 19  27 2a 1f a8 46 9b 5c 66   ·····Q·· '*··F·\f
00b0  5a 7b aa 2b 77 85 6e 62  9f eb 09 8c ff b2 86 fa   Z{·+w·nb ········
```

As long as you can see the result after typing it, it means that ECN=3 marked packets appeared, it also means that DualPI2 is marking the congested packets with ECN. please let iperf3 connect for a longer period of time, in my previous tests congestion appeared at a ratio of about 1:6000 or 20,000th packets before the first congestion marking appeared.

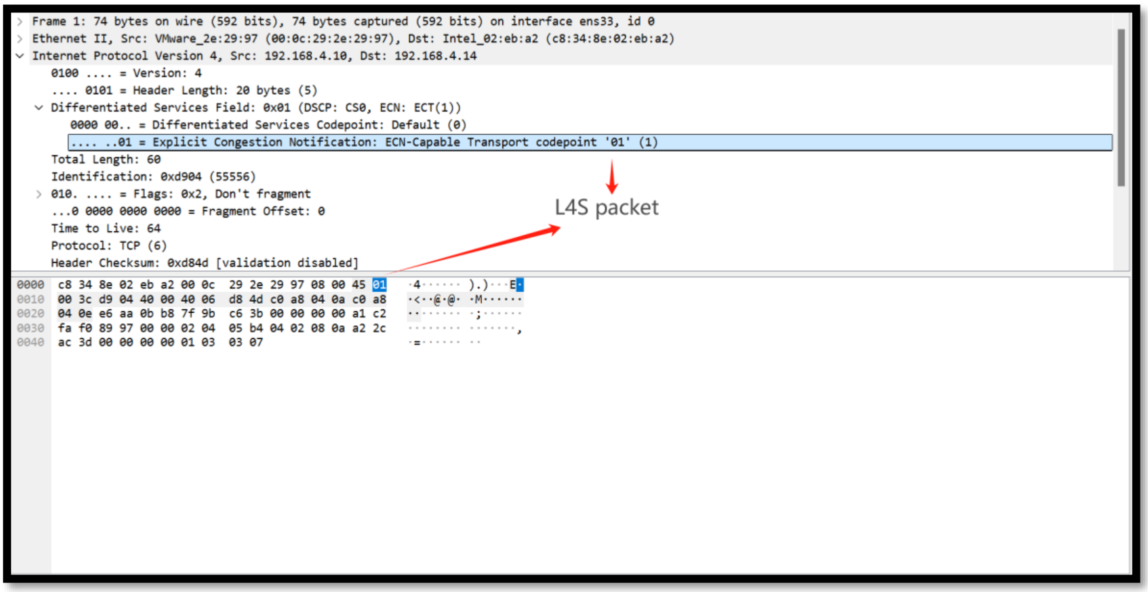


```
ip.dsfield.ecn == 3
```

```
No.       Time         Source          Destination     Protocol Lengtl Info
1439… 111.243267078 192.168.4.14    192.168.4.10    TCP      78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=62234593 Win=24551 Len=0 TSval=4279629176 TSecr=2748656101 EE1B=312250 ECEB=0 EE0B=1
1439… 111.251783295 192.168.4.14    192.168.4.10    TCP      78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=62236029 Win=24542 Len=0 TSval=4279629176 TSecr=2748656102 EE1B=313686 ECEB=0 EE0B=1
1439… 111.252620772 192.168.4.14    192.168.4.10    TCP      78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=62237465 Win=24534 Len=0 TSval=4279629176 TSecr=2748656104 EE1B=315122 ECEB=0 EE0B=1
1439… 111.253463333 192.168.4.14    192.168.4.10    TCP      78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=62238901 Win=24525 Len=0 TSval=4279629176 TSecr=2748656105 EE1B=316558 ECEB=0 EE0B=1
1439… 111.254070407 192.168.4.14    192.168.4.10    TCP      78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=62240337 Win=24517 Len=0 TSval=4279629176 TSecr=2748656106 EE1B=317994 ECEB=0 EE0B=1
1439… 111.254425864 192.168.4.14    192.168.4.10    TCP      78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=62241773 Win=24508 Len=0 TSval=4279629176 TSecr=2748656107 EE1B=319430 ECEB=0 EE0B=1
1439… 111.264561513 192.168.4.14    192.168.4.10    TCP      78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=62246081 Win=24551 Len=0 TSval=4279629180 TSecr=2748656111 EE1B=323738 ECEB=0 EE0B=1
1439… 111.264981311 192.168.4.14    192.168.4.10    TCP      78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=62247517 Win=24542 Len=0 TSval=4279629180 TSecr=2748656112 EE1B=325174 ECEB=0 EE0B=1
1439… 111.265041537 192.168.4.14    192.168.4.10    TCP      78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=62248953 Win=24534 Len=0 TSval=4279629180 TSecr=2748656134 EE1B=326610 ECEB=0 EE0B=1
2425… 222.559662377 192.168.4.14    192.168.4.10    TCP      78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=97591133 Win=24551 Len=0 TSval=4279740506 TSecr=2748770665 EE1B=2114358 ECEB=0 EE0B=1
4714… 400.751815747 192.168.4.14    192.168.4.10    TCP      78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=191472505 Win=24526 Len=0 TSval=4279918700 TSecr=2748948559 EE1B=12109650 ECEB=0 EE0B=1
4714… 400.752755491 192.168.4.14    192.168.4.10    TCP      78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=191473941 Win=24515 Len=0 TSval=4279918700 TSecr=2748948562 EE1B=12111086 ECEB=0 EE0B=1
4813… 408.432754123 192.168.4.14    192.168.4.10    TCP      78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=195649829 Win=24551 Len=0 TSval=4279926382 TSecr=2748956343 EE1B=16286974 ECEB=0 EE0B=1
4813… 408.432942194 192.168.4.14    192.168.4.10    TCP      78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=195651265 Win=24542 Len=0 TSval=4279926382 TSecr=2748956345 EE1B=16288410 ECEB=0 EE0B=1
5964… 497.892052799 192.168.4.14    192.168.4.10    TCP      78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=242328445 Win=24551 Len=0 TSval=4280015634 TSecr=2749045869 EE1B=12633942 ECEB=0 EE0B=1
5964… 497.892599006 192.168.4.14    192.168.4.10    TCP      78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=242329881 Win=24542 Len=0 TSval=4280015634 TSecr=2749045871 EE1B=12635378 ECEB=0 EE0B=1
5968… 498.190218473 192.168.4.14    192.168.4.10    TCP      78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=242505073 Win=24551 Len=0 TSval=4280016138 TSecr=2749046358 EE1B=12810570 ECEB=0 EE0B=1
5968… 498.190368758 192.168.4.14    192.168.4.10    TCP      78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=242506509 Win=24542 Len=0 TSval=4280016138 TSecr=2749046360 EE1B=12812006 ECEB=0 EE0B=1
6823… 565.428054656 192.168.4.14    192.168.4.10    TCP      78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=278484053 Win=24524 Len=0 TSval=4280083373 TSecr=2749113270 EE1B=15235118 ECEB=0 EE0B=1
6823… 565.430038593 192.168.4.14    192.168.4.10    TCP      78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=278485489 Win=24513 Len=0 TSval=4280083373 TSecr=2749113272 EE1B=15236554 ECEB=0 EE0B=1
6823… 565.430224723 192.168.4.14    192.168.4.10    TCP      78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=278486925 Win=24502 Len=0 TSval=4280083373 TSecr=2749113275 EE1B=15237990 ECEB=0 EE0B=1
6823… 565.430436927 192.168.4.14    192.168.4.10    TCP      78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=278488361 Win=24491 Len=0 TSval=4280083373 TSecr=2749113277 EE1B=15239426 ECEB=0 EE0B=1
6823… 565.433648967 192.168.4.14    192.168.4.10    TCP      78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=278492669 Win=24551 Len=0 TSval=4280083377 TSecr=2749113287 EE1B=15243734 ECEB=0 EE0B=1
6823… 565.434608155 192.168.4.14    192.168.4.10    TCP      78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=278494105 Win=24542 Len=0 TSval=4280083377 TSecr=2749113289 EE1B=15245170 ECEB=0 EE0B=1
6823… 565.434943383 192.168.4.14    192.168.4.10    TCP      78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=278495541 Win=24534 Len=0 TSval=4280083377 TSecr=2749113291 EE1B=15246606 ECEB=0 EE0B=1
6823… 565.435316795 192.168.4.14    192.168.4.10    TCP      78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=278496977 Win=24525 Len=0 TSval=4280083377 TSecr=2749113294 EE1B=15248042 ECEB=0 EE0B=1
6823… 565.435997253 192.168.4.14    192.168.4.10    TCP      78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=278498413 Win=24517 Len=0 TSval=4280083377 TSecr=2749113308 EE1B=15249478 ECEB=0 EE0B=1
6823… 565.437608708 192.168.4.14    192.168.4.10    TCP      78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=278499849 Win=24508 Len=0 TSval=4280083377 TSecr=2749113328 EE1B=15250914 ECEB=0 EE0B=1
7378… 608.166721618 192.168.4.14    192.168.4.10    TCP      78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=301735765 Win=24551 Len=0 TSval=4280126101 TSecr=2749155912 EE1B=4932398 ECEB=0 EE0B=1
7378… 608.166851692 192.168.4.14    192.168.4.10    TCP      78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=301737201 Win=24542 Len=0 TSval=4280126101 TSecr=2749155915 EE1B=4933834 ECEB=0 EE0B=1
7642… 630.401002673 192.168.4.14    192.168.4.10    TCP      78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=312869073 Win=24551 Len=0 TSval=4280148335 TSecr=2749178018 EE1B=16065706 ECEB=0 EE0B=1
7642… 630.401818173 192.168.4.14    192.168.4.10    TCP      78 3000 → 56048 [ACK, ACE=5] Seq=1 Ack=312870509 Win=24542 Len=0 TSval=4280148335 TSecr=2749178021 EE1B=16067142 ECEB=0 EE0B=1
```

```
> Frame 143939: 78 bytes on wire (624 bits), 78 bytes captured (624 bits) on interface wlp8s0, id 0
> Ethernet II, Src: Intel_02:eb:a2 (c8:34:8e:02:eb:a2), Dst: Intel_bb:11:c9 (dc:46:28:bb:11:c9)
> Internet Protocol Version 4, Src: 192.168.4.14, Dst: 192.168.4.10
> Transmission Control Protocol, Src Port: 3000, Dst Port: 56048, Seq: 1, Ack: 62234593, Len: 0
```

```
0000  dc 46 28 bb 11 c9 c8 34  8e 02 eb a2 08 00 45 03   ·F(····4 ······E·
0010  00 40 bd 15 40 00 40 06  f4 36 c0 a8 04 0e c0 a8   ·@··@·@· ·6······
0020  04 0a 0b b8 da f0 12 82  60 cd c4 8e fb a6 b1 50   ········ `······P
0030  5f e7 11 d9 00 00 01 01  08 0a ff 15 f5 78 a3 d5   _······· ·····x··
0040  29 e5 ae 0b 04 c3 ba 00  00 00 00 00 01 01         )······· ······
```



```
> Frame 1: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface ens33, id 0
> Ethernet II, Src: VMware_2e:29:97 (00:0c:29:2e:29:97), Dst: Intel_02:eb:a2 (c8:34:8e:02:eb:a2)
v Internet Protocol Version 4, Src: 192.168.4.10, Dst: 192.168.4.14
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  v Differentiated Services Field: 0x01 (DSCP: CS0, ECN: ECT(1))
      0000 00.. = Differentiated Services Codepoint: Default (0)
      .... ..01 = Explicit Congestion Notification: ECN-Capable Transport codepoint '01' (1)
    Total Length: 60
    Identification: 0xd904 (55556)
  > 010. .... = Flags: 0x2, Don't fragment
    ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 64
    Protocol: TCP (6)
    Header Checksum: 0xd84d [validation disabled]
```

```
0000  c8 34 8e 02 eb a2 00 0c  29 2e 29 97 08 00 45 01   ·4······ ).)···E·
0010  00 3c d9 04 40 00 40 06  d8 4d c0 a8 04 0a c0 a8   ·<··@·@· ·M······
0020  04 0e e6 aa 0b b8 7f 9b  c6 3b 00 00 00 00 a1 c2   ········ ·;······
0030  fa f0 89 97 00 00 02 04  05 b4 04 02 08 0a a2 2c   ········ ·······,
0040  ac 3d 00 00 00 00 01 03  03 07                     ·=······ ··
```

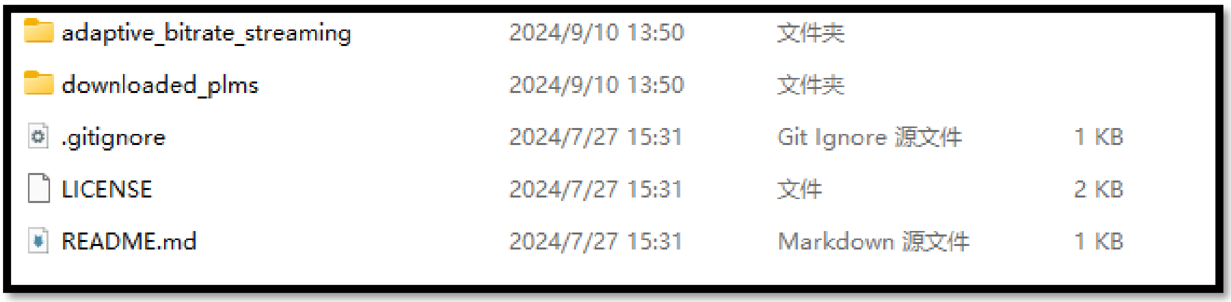
Checkpoint: If you can see packets with ECN=3, it means that the L4S architecture has been enabled. You have completed the setup part of the experiment. The next step is to collect L4S data, convert the collected data into PKL experience pool files that LLM can understand, and finally train in RunPod.

Next, please follow the AQM-LLM Documentation to download the files required for the LLM model and NetLLM architecture. (If you are in a hurry, please seek help from Deol in the IoT lab. The workstation computer theoretically has the complete NetLLM + LLM files.)

After you download the Llama2-7b-hf and NetLLM files, the structure of the entire trained model should look like this:

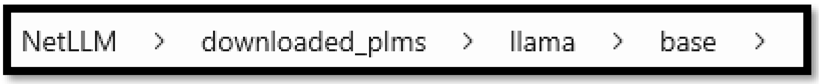
NetLLM

Open the NetLLM folder:

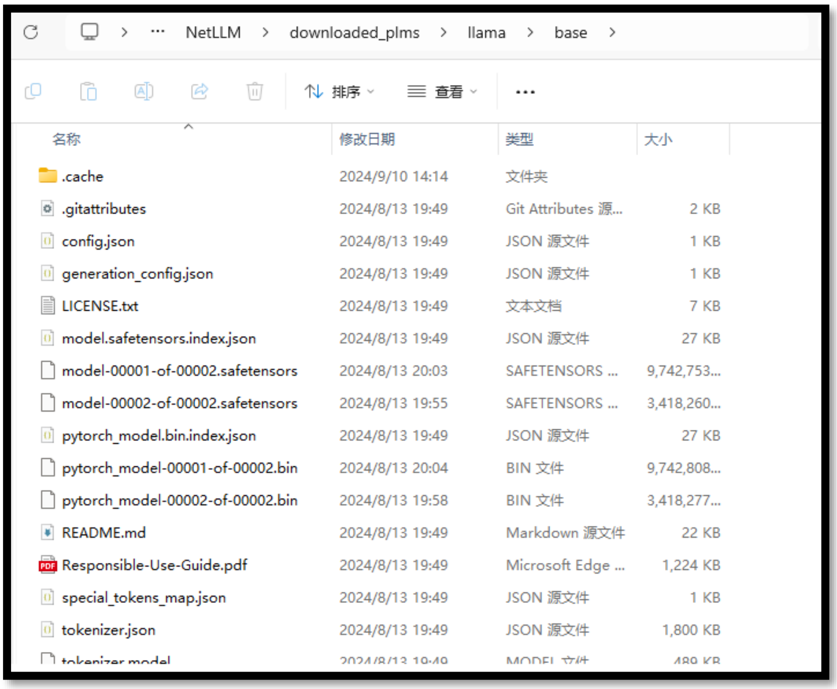


adaptive_bitrate_streaming	2024/9/10 13:50	文件夹	
downloaded_plms	2024/9/10 13:50	文件夹	
.gitignore	2024/7/27 15:31	Git Ignore 源文件	1 KB
LICENSE	2024/7/27 15:31	文件	2 KB
README.md	2024/7/27 15:31	Markdown 源文件	1 KB

Adaptive_bitrate_steaming is all the files needed to train LLM and downloaded_plms is the location where the LLM (Llama2-7b-hf) model is stored. The directory where the LLM is saved is:



NetLLM > downloaded_plms > llama > base >



··· > NetLLM > downloaded_plms > llama > base >

名称	修改日期	类型	大小
.cache	2024/9/10 14:14	文件夹	
.gitattributes	2024/8/13 19:49	Git Attributes 源...	2 KB
config.json	2024/8/13 19:49	JSON 源文件	1 KB
generation_config.json	2024/8/13 19:49	JSON 源文件	1 KB
LICENSE.txt	2024/8/13 19:49	文本文档	7 KB
model.safetensors.index.json	2024/8/13 19:49	JSON 源文件	27 KB
model-00001-of-00002.safetensors	2024/8/13 20:03	SAFETENSORS ...	9,742,753...
model-00002-of-00002.safetensors	2024/8/13 19:55	SAFETENSORS ...	3,418,260...
pytorch_model.bin.index.json	2024/8/13 19:49	JSON 源文件	27 KB
pytorch_model-00001-of-00002.bin	2024/8/13 20:04	BIN 文件	9,742,808...
pytorch_model-00002-of-00002.bin	2024/8/13 19:58	BIN 文件	3,418,277...
README.md	2024/8/13 19:49	Markdown 源文件	22 KB
Responsible-Use-Guide.pdf	2024/8/13 19:49	Microsoft Edge ...	1,224 KB
special_tokens_map.json	2024/8/13 19:49	JSON 源文件	1 KB
tokenizer.json	2024/8/13 19:49	JSON 源文件	1,800 KB
tokenizer.model	2024/8/13 19:49	MODEL 文件	489 KB

The process of renting RunPod is not explained here. If you have a local GPU that can train NetLLM, you don’t need to rent RunPod. It is recommended to read the process of AQM-LLM Documentation before quick operation.

Use the scp command to upload the NetLLM folder to RunPod. Be sure to upload it to cd /workspace, otherwise all storage contents will be erased when you close RunPod. cd to the adaptive_bitrate_streaming folder and use the following command:

```
python -m pip install --upgrade pip && pip install openprompt==1.0.1 && pip install numpy==1.24.4 && pip install peft==0.6.2 && pip install transformers==4.34.1 && pip install --upgrade huggingface_hub && pip install scikit-learn && pip install munch
```

```
python run_plm.py --adapt --grad-accum-steps 32 --plm-type llama --plm-size base --rank 128 --device cuda:0 --lr 0.0001 --warmup-steps 2000 --num-epochs 80 --eval-per-epoch 2
```

This process is trained entirely using the official NetLLM file, just to familiarize you with how the entire NetLLM and LLM work. You don't have to run 80 iterations.

Next, you need to modify the PKL file yourself, modify other codes to remove the restrictions on the model's PKL content, and then train. The specific process AQM-LLM Documentation has explained it.