Лабораторийн ажил №5

IP Protocol

МТЭС, МКУТ, Компьютерийн ухаан

Б.Барсболд, 22B1NUM4397

Ажлын зорилго

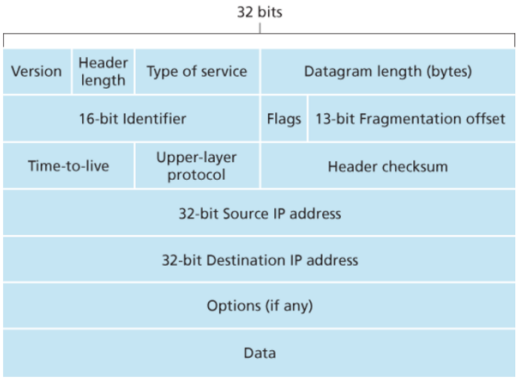
Энэхүү лабораторийн ажлаар бид **IP датаграм** буюу сүлжээний түвшинд байгаа пакет дээр тулгуурлан IP протоколын бүтцийг судална. Мөн түүнчлэн, traceroute програмын үр дүнд үүсэх илгээгч, хүлээн авагч IP datagram-ийн дээр анализ хийнэ. Лабораторийн ажлын төгсгөлд IP datagram–ийн талбарууд, IP фрагментийн талаар мэдлэгтэй болно.

Онолын судалгаа

Бидний өнөөгийн ашигладаг IP 2 үндсэн хувилбар байдаг. IPv4 [RFC 791] болон IPv6 [RFC 2460; RFC 4291].

1. IPv4-ийн датаграм бүтэц

Network layer-д дамжуулагдах хамгийн бага нэгжийг датаграм гэдэг.



Зураг 1 IPv4 датаграмын бүтэц

- Version number. Энэ 4 бит тоо нь датаграмыг аль IP хувилбарын ашиглаж байгааг илэрхийлдэг бөгөөд үүнийг шалгаад router үлдсэн хэсгийг хэрхэн хөрвүүлэхээ мэдэж авдаг. IP-ийн өөр хувилбарууд өөр өөр датаграм бүтэц ашигладаг.
- Header length. IP датаграм олон төрлийн сонголтууд хадгалж чаддагийн улмаас толгой хэсэг нь харилцан адилгүй байдаг. Тиймээс толгой хэсгийн уртын зааж өгөх зорилготой энэ 4 бит байдаг.

- Type of Service (TOS). Энэ талбар нь өөр өөр төрлийн датаграмуудыг хооронд нь ялгахад хэрэглэгддэг. Жишээ нь бодит хугацаанд дамжуулагдах ёстой датаграмыг, зүгээр л дамжуулах ёстой датаграмаас ялгах нь чухал юм.
- Datagram length. Энэ нь датаграмын толгой болон өгөгдлийн талбаруудын нийт урт бөгөөд байтаар илэрхийлэгдэнэ. Энэ талбар нь 16 байт бөгөөд онолын хувьд нэг датаграмын урт нь 65,535 байт байж болно. Гэхдээ практикт 1500 байтаас уртгүй байдаг нь нэг ethernet фрейм-ийн агуулж чадах өгөгдлийн хэмжээ юм.
- Identifier, flags, fragmentation offset. Энэ гурван талбар нь датаграмын хуваалтад хэрэглэггдэг. Сонирхолтой нь IPv6 датаграмыг хуваахыг зөвшөөрдөггүй.
- Time-to-live (TTL). Энэ талбар нь датаграм сүлжээнд үүрд нэг тойрогоор эргэлдэх хөдөлгөөн хийхээс сэргийлдэг. Энэ талбар сүлжээнд router-р дамжин өнгөрөх тоолонд нэгээр хорогддог. 0 болох үед router энэ датаграмыг устгадаг.
- Protocol. Энэ талбар нь ерөнхийдөө IP датаграм нь эцсийн цэгтээ хүрсэн үед ашиглагдах бөгөөд энэ талбарт хадгалагдсан утга нь тодорхой transport-layer-ийн протоколыг илэрхийлж байдаг бөгөөд үүнийг эцсийн цэгт тухайн өгөгдөл хаашаа дамжуулагдхыг илэрхийлдэг. Жишээ нь 6 байвал TCP, 17 байвал UDP гэх мэт.
- Header checksum. Энэ талбар нь router-ийн хүлээж авсан датаграмд битийн алдаа байгаа эсэхийг шалгадаг.
- Source and destination IP addresses. Илгээгч датаграм үүсгэх үед өөрийн IP хаягийг source IP address талбарт тавьж хамгийн төгсгөлийн хүлээн авагчийн IP хаягийг destination IP address талбарт тавьдаг. Ихэнх тохиолдолт илгээгч хүлээн авагчийн IP хаягийг DNS-ийн тусламжтай олдог.
- Options. Энэ талбар IP-ийн толгой хэсгийг өргөтгөхөд хэрэглэгддэг. Мөн энэ талбар нь IP хаягийн толгой хэсэг хувьсах урттай байдагтай нягт холбоотой. Зарим датаграм option хэсэгтэй байхийг шаарддаг бол зарим датаграм огт option талбаргүй байна.
- Data (payload). Хамгийн сүүлийн энэ хэсэг хамгийн чухал бөгөөд ихэнх тохиолдолд энэ талбарт хүлээн авагчид хүргэгдэх TCP, UDP гэх мэт transport-layer-ийн сегментүүд хадгалдаг. Гэхдээ бусад төрлийн өгөгдлийг ч бас тээвэрлэж болно. Жишээ нь ICMP мессежүүд гэх мэт.

2. IPv4 датаграм хуваалт

Link-layer протоколууд янз янзын хэмжээтэй фреймүүд дамжуулдаг. Зарим протоколууд том хэмжээтэй датаграм дамжуулж чаддаг байхад зарим нь харьцангуй жижиг хэмжээтэй датаграм дамжуулж чаддаг. Жишээ нь ethernet фрейм 1500 байт хүртлэх өгөгдөл дамжуулж чаддаг байхад зарим өргөн хүрээний холболтууд 576 байтаас илүү өгөгдөл дамжуулж чаддаггүй. Link-layer-р

дамжуулж чадах хамгийн их фреймийн хэмжээ maximum transmission unit (MTU) гэж нэрлэгддэг. IP датаграм нь link-layer-ийн фрейм дотор битүүмжлэгддэг учир link-layer-ийн MTU нь IP датаграм-д маш чанга хязгаарлалт болдог. IP датаграм нь link-layer-ийн MTU-ээр хязгаарлагдах нь асуудал биш ч гэсэн илгээгчээс хүлээн авагчийн хооронд төрөл бүрийн холбоос байдаг бөгөөд энэ холбоос бүр өөр өөр MTU-тэй байдаг асуудалтай.

Олон холбоосуудын уулзвар болдог router хүлээн авсан фреймээ forwarding table-ээ шалган өөр нэг холбоос руу дамжуулдаг. Энэ үед хоёр холбоосын MTU өөр байж болох ба хүлээн авч буй холбоосын MTU нь гарж буй холбоосын MTU-ээс их байж болно. Энэ үед IP датаграмын өгөгдлийн талбарт байгаа өгөгдлийг жижиг хэсгүүд болгон хэсэг бүрийг тусд нь link-layer фрейм-д битүүмжлэн гаргадаг. Хуваагдсан жижиг датаграм бүрийг fragment гэж нэрлэдэг. Энэ fragment-үүд эцсийн хүлээн авагчийн transport layer-д очхоос өмнө буцаж угсрагдах ёстой ба TCP болон UDP нар бүрэн бүтэн хуваагдаагүй сегментүүд хүлээн авдаг. Харин энэ буцааж угсрах үйлдлийг router дээр хийх нь router-ийн хурдийг бууруулах учир энэ үйлдлийг төгсгөлийн систем дээр хийхээр загварчлагдсан байдаг.

3. IPv4 Addressing

Хост төхөөрөмж бүр ерөнхий ганцхан сүлжээнд гарах гарцтай байх ба IP-аар датаграм илгээх үед энэ холбоосоор дамждаг. Хост болон физик холбоос хоорондын хилийг интерфэйс гэнэ. Router-ийн үүрэг нь нэг холбоосоор орж ирсэн датаграмыг өөр нэг холбоосоор гаргах учир хоёр болон түүнээс олон холбоосуудтай холбогдсон байна. Router болон түүний холбоосуудын хоорондох хилийг ч мөн адил интерфейс гэнэ. Router-ийн интерфейс бүр IP датаграмуудыг илгээх болон хүлээн авч чаддаг. Тийм учраас IP-д интерфэйс бүр өөрийн гэсэн IP хаягтай байх хэрэгтэй. IP хаяг тухайн хостых гэхээсээ илүү хостын интерфэйсийн гэж ойлгож болно.

IP хаяг болгон 32 битийн урттай байдаг бөгөөд энэ нь нийтдээ 2^{32} боломж байдаг. Мөн энэ хаяг байт бүр цэгээр (.) тусгаарлагдан 10-ын тоололд илэрхийлэгддэг. Бүх интернетэд байрласан router болон хостууд дээр байрлаж буй интерфэйсүүд дахин давтагдашгүй хаягтай байх ёстой. Тийм учраас энэ хаягийг хааш яаш байдлаар сонгож болохгүй юм.

Туршилт

- WireShark программаа ажлуулж дамжуулж байгаа пакетуудыг бичиж авах үйлдлйиг идэвхжүүл.
- Командын мөр (cmd) дээр ping команд ашиглаж дурын сервер болон өөрийн холбогдсон байгаа сүлжээны гарц руу мэдээлэл дамжуулаарай.
- Командын мөр (cmd) дээр tracert комманд ашиглаж дурын сервер рүү мэдээлэл дамжуул.

```
❯ ping fb.com
PING fb.com (157.240.211.35): 56 data bytes
64 bytes from 157.240.211.35: icmp_seq=0 ttl=52 time=58.110 ms
64 bytes from 157.240.211.35: icmp_seq=1 ttl=52 time=58.357 ms
64 bytes from 157.240.211.35: icmp_seq=2 ttl=52 time=57.725 ms
64 bytes from 157.240.211.35: icmp_seq=3 ttl=52 time=60.048 ms
64 bytes from 157.240.211.35: icmp_seq=4 ttl=52 time=58.429 ms
```

Зураг 2 Дурын сервер рүү хүсэлт [1]байгаа байдал

```
❯ ping -s 128 fb.com
PING fb.com (157.240.211.35): 128 data bytes
136 bytes from 157.240.211.35: icmp_seq=0 ttl=52 time=60.063 ms
136 bytes from 157.240.211.35: icmp_seq=1 ttl=52 time=59.544 ms
136 bytes from 157.240.211.35: icmp_seq=2 ttl=52 time=59.070 ms
136 bytes from 157.240.211.35: icmp_seq=3 ttl=52 time=59.407 ms
136 bytes from 157.240.211.35: icmp_seq=4 ttl=52 time=59.646 ms
```

Зураг 3 Дурын сервер рүү 128 байт өгөгдөл илгээж буй байдал

```
❯ ping -s 256 fb.com
PING fb.com (157.240.211.35): 256 data bytes
264 bytes from 157.240.211.35: icmp_seq=0 ttl=52 time=58.617 ms
264 bytes from 157.240.211.35: icmp_seq=1 ttl=52 time=57.857 ms
264 bytes from 157.240.211.35: icmp_seq=2 ttl=52 time=58.612 ms
^C
--- fb.com ping statistics ---
3 packets transmitted, 3 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 57.857/58.362/58.617/0.357 ms
```

Зураг 4 Дурын сервер рүү 256 байт өгөгдөл илгээж буй байдал

```
❯ ping -s 512 fb.com
PING fb.com (157.240.211.35): 512 data bytes
520 bytes from 157.240.211.35: icmp_seq=0 ttl=52 time=58.116 ms
520 bytes from 157.240.211.35: icmp_seq=1 ttl=52 time=57.803 ms
520 bytes from 157.240.211.35: icmp_seq=2 ttl=52 time=58.259 ms
520 bytes from 157.240.211.35: icmp_seq=3 ttl=52 time=57.791 ms
^C
--- fb.com ping statistics ---
4 packets transmitted, 4 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 57.791/57.992/58.259/0.202 ms
```

Зураг 5 Дурын сервер рүү 512 байт өгөгдөл илгээж буй байдал

```
❯ ping -s 1024 fb.com
PING fb.com (157.240.211.35): 1024 data bytes
1032 bytes from 157.240.211.35: icmp_seq=0 ttl=52 time=58.348 ms
1032 bytes from 157.240.211.35: icmp_seq=1 ttl=52 time=58.731 ms
1032 bytes from 157.240.211.35: icmp_seq=2 ttl=52 time=58.293 ms
1032 bytes from 157.240.211.35: icmp_seq=3 ttl=52 time=58.075 ms
^C
--- fb.com ping statistics ---
4 packets transmitted, 4 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 58.075/58.362/58.731/0.236 ms
```

Зураг 6 Дурын сервер рүү 1024 байт өгөгдөл илгээж буй байдал

```
❯ ping -s 1536 fb.com
PING fb.com (157.240.211.35): 1536 data bytes
Request timeout for icmp_seq 0
Request timeout for icmp_seq 1
Request timeout for icmp_seq 2
Request timeout for icmp_seq 3
^C
--- fb.com ping statistics ---
5 packets transmitted, 0 packets received, 100.0% packet loss
```

Зураг 7 Дурын сервер рүү 1536 байт өгөгдөл илгээж буй байдал

[1] ping коммандыг unix-like үйлдлийн систем дээр ашиглаж байгаа учир -l option-ны оронд -s option ашиглав.

fb.com гэсэн домайн хаягтай сервер рүү ping комманд ашиглан илгээж үзэв. 1536 байт өгөгдөл илгээх үед timeout болж байна.

```
❯ ping 192.168.1.1 -s 2000
PING 192.168.1.1 (192.168.1.1): 2000 data bytes
Request timeout for icmp_seq 0
Request timeout for icmp_seq 1
2008 bytes from 192.168.1.1: icmp_seq=2 ttl=64 time=2.745 ms
2008 bytes from 192.168.1.1: icmp_seq=3 ttl=64 time=3.194 ms
2008 bytes from 192.168.1.1: icmp_seq=4 ttl=64 time=3.104 ms
2008 bytes from 192.168.1.1: icmp_seq=5 ttl=64 time=2.345 ms
^C
--- 192.168.1.1 ping statistics ---
6 packets transmitted, 4 packets received, 33.3% packet loss
round-trip min/avg/max/stddev = 2.345/2.847/3.194/0.335 ms
```

Зураг 8 Өөрийн gateway рүү 2000 байт өгөгдөл илгээж буй байдал

```
❯ ping 192.168.1.1 -s 3800
PING 192.168.1.1 (192.168.1.1): 3800 data bytes
Request timeout for icmp_seq 0
Request timeout for icmp_seq 1
Request timeout for icmp_seq 2
Request timeout for icmp_seq 3
3808 bytes from 192.168.1.1: icmp_seq=4 ttl=64 time=3.481 ms
3808 bytes from 192.168.1.1: icmp_seq=5 ttl=64 time=3.652 ms
3808 bytes from 192.168.1.1: icmp_seq=6 ttl=64 time=6.213 ms
3808 bytes from 192.168.1.1: icmp_seq=7 ttl=64 time=3.719 ms
3808 bytes from 192.168.1.1: icmp_seq=8 ttl=64 time=4.498 ms
3808 bytes from 192.168.1.1: icmp_seq=9 ttl=64 time=3.458 ms
3808 bytes from 192.168.1.1: icmp_seq=10 ttl=64 time=3.919 ms
3808 bytes from 192.168.1.1: icmp_seq=11 ttl=64 time=4.032 ms
3808 bytes from 192.168.1.1: icmp_seq=12 ttl=64 time=3.599 ms
```

Зураг 9 Өөрийн gateway рүү 3800 байт өгөгдөл илгээж буй байдал

Даалгавар

1. Илгээгчийн IP хаяг талбарт байгаа хаягийг бич. Server_name-ийн талбарт ямар хаяг байна вэ?
    - Source Address: 192.168.1.1
    - Destination Address: 192.168.1.20
2. Layer protocol field –ийн хэмжээ хэд вэ? Ping командад -l сонголт өгөхөд өөрчлөгдөж байна уу? Үр дүнг тайлбарла.
3. IP header –д IP datagram –ийн payload-д хэдэн byte байна? Payload bytes –ийн тоо?

    IP header нь 20 байт payload хэсэг нь 64 байт байна.
4. IP datagram fragmented хийгдсэн үү? Яагаад?

    Хийгдээгүй байна. Учир нь MTU буюу maximum transmission unit хэмжээнээс хэтрээгүй учир хуваалт хийх шаардлагагүй байна.
5. Нэг удаа ping команд ажиллуулахад хэдэн ICMP датаграм илгээж байна вэ? Тэдгээр нь хоорондоо ялгаатай юу?

    Миний туршилтын хувьд гараас зогсоох хүртэл тасралтгүй илгээж байсан.
6. Ping командуудын эхний датаграмуудыг хооронд нь харьцуул. Ялгаа байна уу? Хоёр дах датаграмууд ялгаатай байна уу?

Эхний датаграмын нийт урт 84 байт урттай байхад хоёр дахь датаграмуудын урт 156 байт урттай байна.

7. Бүх датаграмуудыг ажигла. Толгой хэсгийн аль талбар тогтмол, мөн өөрчлөгдөж байна вэ? Яагаад?

Датаграмуудын flags талбараас бусад нь тогтмол бөгөөд length талбар датаграмынхаа уртыг зааж байна.

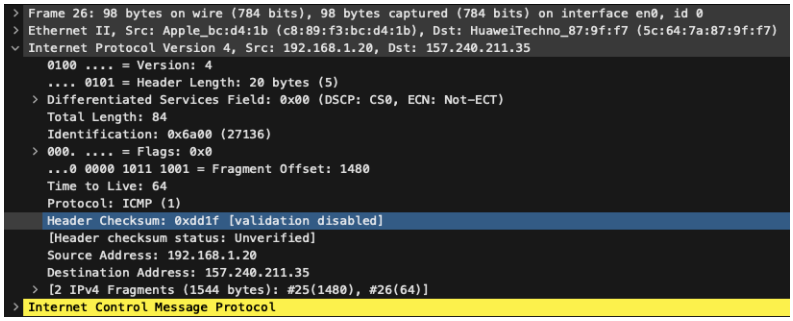
8. Identification ба Time to live талбарын утгууд хэд байна вэ? Ping командууд хооронд эдгээр утгууд ялгаатай байна уу, тайлбарла.
   - Identification: 0x3999 (14745) Хэрэв IP датагамыг fragmentation хийсэн бол энэ ID буцааж нийлүүлхэд ашиглагддаг.
   - Time to Live: 64 Энэ талбар нь хүсэлт хэдэн hop буюу хэдэн route дамжсаны дараа deleted болхыг заадаг. Hop бүр дээр нэгээр багасдаг.
9. Tracert командын үр дүнгээс эхний 2 рүүтэртэй солилцсон датаграмуудыг ажиглая. TTL-ийн утга ямар байна вэ? Өөрчлөгдөж байна уу? TTL exceeded ямар утгыг илэрхийлж байна вэ?

Ирсэн хүсэлтийн TTL утга 248 байна. TTL exceeded in transit нь TTL талбарын утга очих цэгтээ хүрэхийн өмнө 0 болсон бол гардаг. Traceroute комманд нь ICMP хүсэлт илгээх болгондоо TTL-г нэгээр багасгаж илгээж ажилдаг учир энэ зүйл болж байна.

10. Аль Ping командыг гүйцэтгэхэд Fragment хийгдсэн байна бэ? Яагаад?

ping -s 1536 fb.com коммандыг ажилуулах үед fragment хийгдэж байна. Учир нь сүлжээгээр ийм том хэмжээны өгөгдөл нэг frame-д дамжуулах боломжгүй болсон учир fragment хийж байна.

11. Хуваагдсан датаграмуудаас эхний fragment-ийг харуул. IP header-ийн аль талбар Fragment хийгдсэнийг илэрхийлэж байна вэ?



Зураг 10 Fragment хийгдсэн датаграмын эхнийх

Fragment offset хэсэгт утга хадгалагдсан байна.

12. Бүх хуваагдсан хэсгүүдийг ажигла. Хэрхэн датаграмын хуваагдлыг эхний ба сүүлийнх гэдгийг тодорхойлж байна вэ?
13. Эхний болон 2 дах хуваагдсан датаграмын толгой хэсгийн аль талбарууд өөрчлөгдсөн байна вэ? Тайлбарла.

14. Датаграмыг хуваахдаа хэмжээг хэрхэн тогтоох вэ? Хуваагдсан датаграмууд дээр тайлбарла.

Сорих асуулт

1. TTL-ийн утгыг тайлбарлана уу?

Энэ талбар нь хүсэлт хэдэн hop буюу хэдэн route дамжсаны дараа deleted болхыг заадаг. Hop бүр дээр нэгээр багасдаг.

2. Яагаад заавал TTL-ийн утга тодорхойлдог вэ?

Энэ талбар нь тухайн датаграм тасралтгүй цикл-д орон хүрэх газраа хүрэхгүйгээр эргэлдхээс сэргийлдэг.

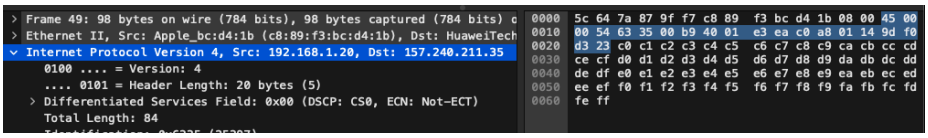
3. Хэрэв TTL-ийн утга 1 болвол яах вэ?

TTL-ийн утга 1 болвол тухайн router тус датаграмыг цааш нь дамжуулна. Дараагын router TTL-г нэгээр багасган 0 болвол тус датаграмыг устгана.

4. IP header-ийн TOS (Type of Service) талбар ямар утгыг илэрхийлдэг вэ?

Энэ талбар нь transport-layer-д ямар протокол ашиглаж буйг илэрхийлдэг.

5. IP datagram-ийн checksum хэрхэн тооцоолох вэ? Жишээгээр тайлбарла.
    1. IP header-г 16 битээр хуваагаад бүгдийг нь нэмнэ.
    2. Хэрэв нийлбэр нь 16 бит-д багтахгүй байвал нэмэлт нэг битийг дараагийн нэмэх үйлдэлд зориулан хадгална.
    3. Дахин нэмэх үйлдлээ хийнэ. Бүх 16 бит хуваалтуудаа нэмсэн үед бид 0-г 1-ээр 1-г 0-ээр солин гарсан хариуг checksum талбарт тавина.



Зураг 11 Жишээ болгон туршилтын үр дүнг авав

Дээрх Зураг 11-д авсан датаграмын checksum-г тооцоолж үзье. Эхлээд 16 битээр салгаж хоёртын тоололд илэрхийлье. Жишээ учир эхний 8 ширхэг 16 битийг авч үзье. Гэхдээ checksum-г өөрийг нь 0000 буюу хоосон 16 бит гэж үзнэ гэдгийг анхаарах хэрэгтэй.

4500: 0100 0101 0000 0000

0054: 0000 0000 0101 0100

Эхний нийлбэр: 0100 0101 0101 0100

6335: 0110 0011 0011 0101

Хоёр дахь нийлбэр: 1010 1000 1000 1001

00B9: 0000 0000 1011 1001

Гурав дахь: 1010 1001 0100 0010

4001: 0100 0000 0000 0001

Дөрөв дэх: 1110 1001 0100 0011

E3EA: 1110 0011 1110 1010

Тав дахь: (1)1100 1101 0010 1101 Урд санасан нэг битээ бид ард талд нь нэмэх хэрэгтэй.

1100 1101 0010 1110

C0A8: 1100 0000 1010 1000

Зургаа дахь: (1)1000 1101 1101 0110

1000 1101 1101 0111

0114: 0000 0001 0001 0100

Сүүлийх: 1000 1110 1110 1011

Checksum: 7114: 0111 0001 0001 0100

Дүгнэлт

Энэ лабораторoор бид IP протокол болон түүгээр өгөгдөл дамжих хамгийн бага нэгж болох датаграм түүний header хэсгүүдийн талаар судалж WireShark програм ашиглаж ping болон traceroute коммандуудын тусламжтай үүсгэсэн пакетуудыг барьж аван үр дүнгүүд дээр анализ хийлээ.