4 TCP, UDP ПРОТОКОЛ

МТЭС Програм хангамж
О.Энэрэл 22B1NUM0506

4.1 Ажлын зорилго

Энэхүү лабораторийн ажлаар TCP/IP -ын тээвэрлэлтийн давхарга (*Transport layer*) -ын Transmission Control Protocol (TCP), User Datagram Protocol (UDP) протоколуудынажиллах зарчмыг судлах болно.

4.2 Туршилт

Wireshark программын тусламжтайгаар UDP болон TCP протокол, түүнтэй холбоотой мессежүүд хэрхэн дамжиж байгааг судлах болно. Wireshark программаа нээж, сүлжээний интерфейсээр дамжиж өнгөрөх урсгалуудыг барьж авна. Шүүлтүүр хэсэг TCP, UDP протоколуудыг бичсэнээр уг протоколоор дамжсан өгөгдлийн урсгалыг шүүж харах боломжтой юм.

4.3 Даалгавар

4.3.1 User Datagram Protocol

1. Өөрийн цуглуулсан өгөгдлийн урсгалаас дурынс нэг UDP пакетыг сонгоно. Энэ пакетын толгой мэдээлэл (header) хэсэгт ямар талбарууд байгааг тайлбарлана уу.

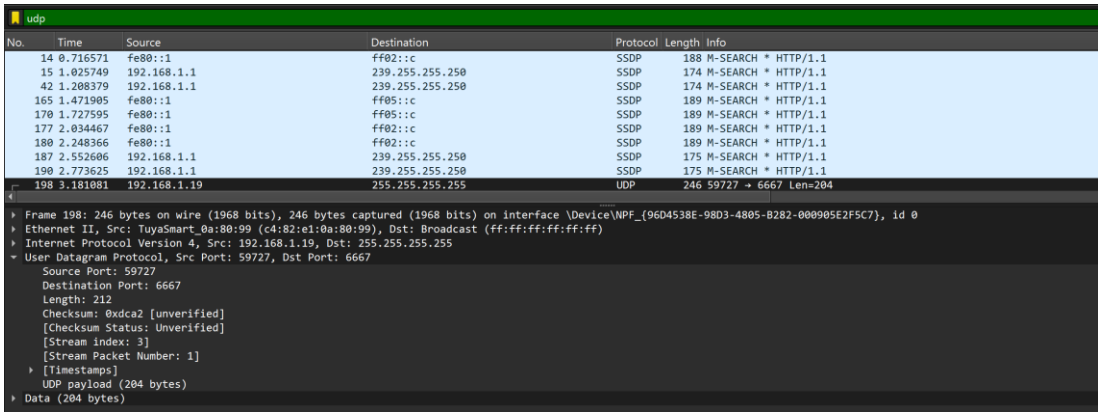


Figure 1 UDP header

UDP пакетын толгой мэдээлэлийн талбарууд :

source port: Илгээж буй хостын портын дугаар.

destination port: Хүлээн авч буй хостын портын дугаар

length: UDP толгой болон өгөгдлийн урт.

checksum : Пакетийн агуулгыг шалгах зориулалттай алдааны шалгалтын утга.

2. Wireshark дээр уг пакетын packet content хэсгийг ажиглаж, UDP пакетын толгойн

мэдээллийн урт хэдэн байт болохыг тэмдэглэнэ.

UDP толгой нь 4 талбараасс бүрддэг бөгөөд талбар бүр нь 2 байт тул UDP пакетын толгойн мэдээллийн урт нь 8 байт.

3. UDP пакетын толгойн хэсэгт Length талбар юуны уртыг илэрхийлж байгаа вэ? Өөрийн цуглуулсан UDP пакет дээрээ баталж, тайлбарлан тайланд оруулна.

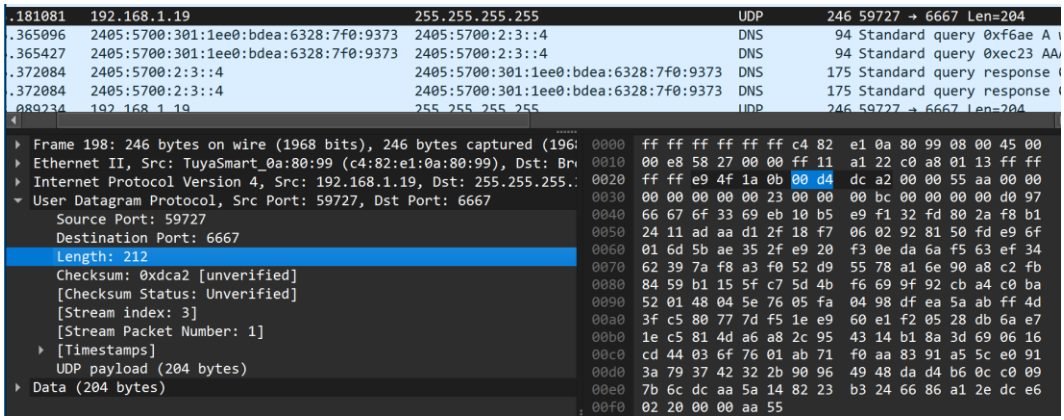


Figure 2 Length

Толгойн мэдээллийн урт Length талбар нь UDP пакетын нийт уртыг илэрхийлдэг. Энэ нь толгойн мэдээллийн урт ба өгөгдлийн хэсгийг (payload) багтаана.

Length талбарын утга 00 d4 байна, 10-т руу хөрвүүлбэл 212.

UDP толгойн мэдээлэл: 8 байт

UDP өгөгдлийн хэсэг (payload): 204 байт

4. UDP payload хэсэгт хамгийн ихдээ хэд байтын өгөгдлийг дамжуулах вэ?

UDP пакетын нийт уртыг тодорхойлдог Length талбар нь 16-битийн урттай тул хамгийн ихдээ 2^{16} = 65535 байт утгыг агуулж чадна. Гэхдээ нийт урт нь UDP толгой болон өгөгдлийн хэсгийг багтах тул 65535 – 8 = 65527 байт өгөгдлийг дамжуулна.

5. Илгээгчийн портын дугаар хамгийн ихдээ хэд байх боломжтой вэ?

UDP портын дугаар нь 16-битийн урттай байдаг. Энэ нь илгээгч болон хүлээн авагч талын портын дугаарыг 0-65535 хооронд үүсгэх боломжтой гэсэн үг. Тиймээс илгээгчийн портын дугаарын хамгийн их утга нь 65535 байж болно.

6. Сүлжээний давхаргын толгой мэдээлэлд агуулагдах UDP протоколын дугаар (type) хэд байна вэ? Сүлжээний давхаргын толгой мэдээллийг ажиглаж хариулна уу.

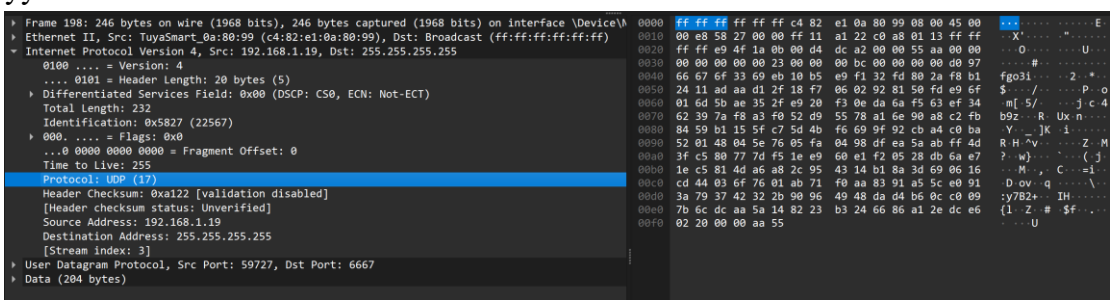


Figure 3 UPD type

UDP протоколын дугаар 17 байна.

7. Өөрийн PC-ээс илгээгдэж байгаа хэд хэдэн UDP пакетууд дээр дараахажиглалтыг

хийгээрэй. Илгээгч эхний UDP пакетыг илгээх бөгөөд хоёр дахь UDP пакет нь эхний UDP пакетын хариу гэвэл (Энд, эхний пакетын хариуд хоёр дахь пакет илгээгдсэн бол эхний пакет илгээгч нь хоёр дахь пакет нь хүлээн авагч байхёстой), уг хоёр пакетын порт дугааруудын хоорондын хамаарал ямар байх вэ, тайлбарлаарай.

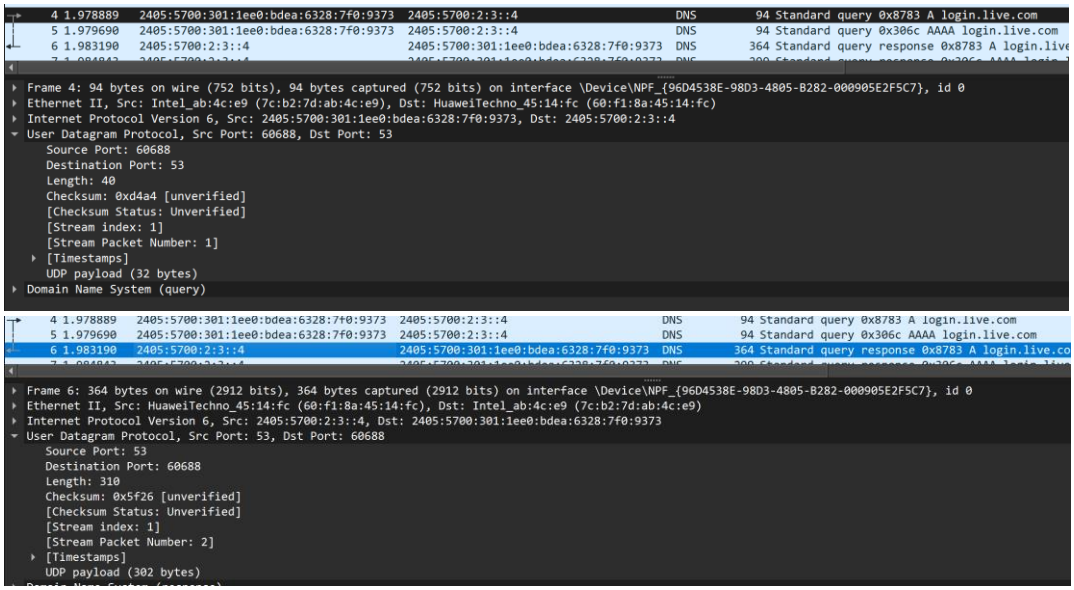


Figure 4 ip addr and port number

Эхний пакет дээр илгээгчийн Source Port хаяг хариу пакетын Destination Port болж, хүлээн авагчийн Destination Port хаяг хариу пакетын Source Port болж солигдсон.

4.3.2 Transmission Control Protocol

Туршилт 1

TCP холболтын 3-way handshake процесс болон HTTP протоколын POST мессежийн талаар судална. Wireshark программыг эхлүүлж, сүлжээний интерфейсээр дамжиж өнгөрөхурсгалуудыг барьж авна. Веб хөтчөөс дараах хаяг руу хандаж http://netconf.num.edu.mn/wiresharkfiles/alice.txt файлыг татаж авна. http://netconf.num.edu.mn/wiresharkfiles/TCP-wireshark-file1.html хуудас руу хандаж Browsetовчийг ашиглан дээр татсан файлыг сонгож, “Upload alice.txt file” дарж gaia.cs.umass.edu сервер рүү илгээнэ. Wireshark программыг зогсоож, дараах асуултад хариулна уу.

1. Илгээгч, хүлээн авагч төхөөрөмжүүдийн IP хаяг, TCP портын дугаар ямар байна вэ? Аль пакетаас уг мэдээллийг авч байгааг тайланд тусгаарай.

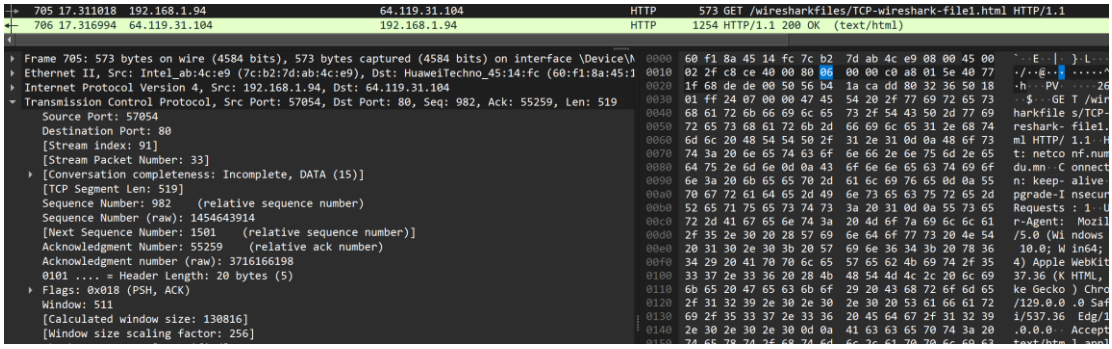


Figure 5 ip address, port number

Илгээгчийн IP хаяг: 192.168.1.94, портын дугаар 57054

Хүлээн авагчийн IP хаяг: 64.119.31.104, портын дугаар 80

HTTP GET пакетаас уг мэдээллийг авсан.

2. Илгээгч компьютер болон серверийн хооронд TCP холболт үүсгэж эхлэхэд ашиглагдаж байгаа TCP SYN сегментийн дарааллын дугаар (sequence number) ямар байна вэ? Уг портын дугаарыг дараа дараагийн дамжуулалд хэрхэн ашиглаж байгааг ажиглан тайланд тусгана уу.

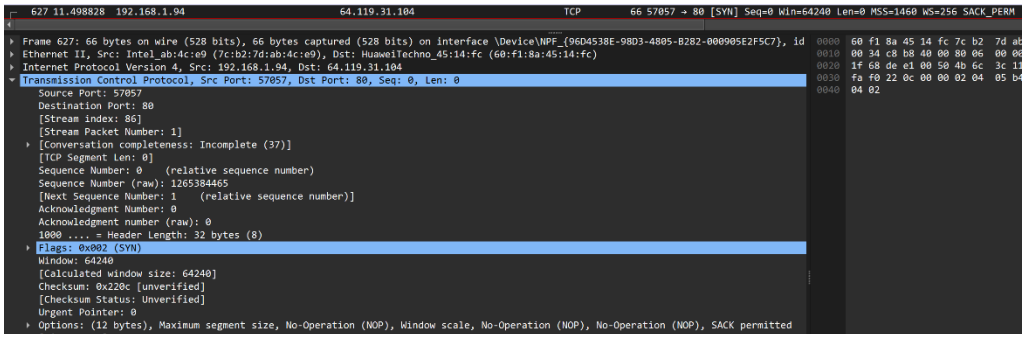


Figure 6 TCP SYN

TCP SYN сегментийн дарааллын дугаар нь 0.

3. Серверээс клиент рүү илгээгдсэн SYN ACK сегментийн дарааллын дугаар ямар байна? SYN ACK сегментийн acknowledgement талбар ямар утгатай байна вэ? Сегментийн аль утга дээр үндэслэн SYN ACK сегмент таньж байгаа вэ?

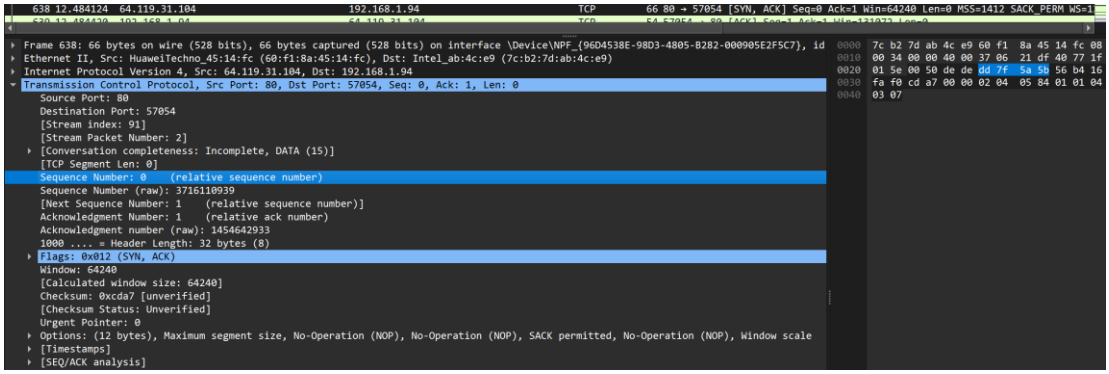


Figure 7 SYN ACK

SYN ACK сегментийн дарааллын дугаар нь 0, acknowledgement талбар нь 1 утгатай байж Flags талбараас SYN ACK сегмент таньж байна.

4. HTTP POST мессэжийг агуулж байгаа TCP сегментийн дарааллын дугаар ямар байна вэ?

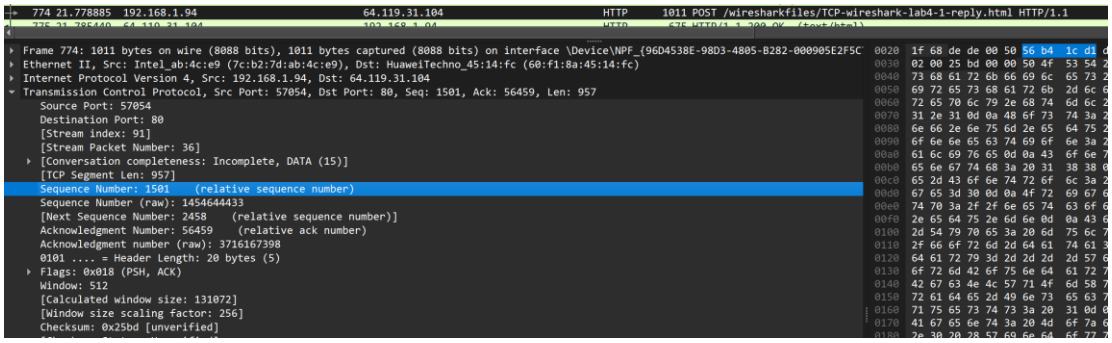


Figure 8 http post

TCP сегментийн дарааллын дугаар нь 1501.

5. TCP холболтын эхний сегментээр HTTP POST агуулж байгаа TCP сегментийг гэж үзвэл эхний 6 сегментүүдийн дарааллын дугаар ямар байна вэ? Дээрх 6 сегмент тус бүрийн илгээсэн болон acknowledgement хүлээн авсан хоорондын хугацаа болох RTT (Round Trip Time) нь сегмент тус бүр ямар байна вэ? ACK хүлээж авсны дараах `EstimatedRTT` утга ямар байх вэ?

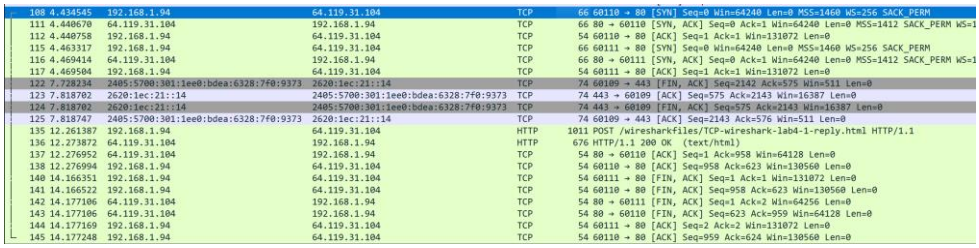


Figure 9 RTT

Packet	Segment	Sequence Number	Илгээсэн цаг(s)	Хүлээн авсан(s)	RTT(ms)	EstimatedRTT(ms)
108	SYN	0	4.434545	4.440678	133	104.13
111	SYN-ACK	0	4.440678	4.440758	80	101.64
112	ACK	1	4.440758	12.261387	7820	1066.43
135	HTTP POST	1	12.261387	12.276994	15.607	935.10
138	ACK	958	12.276994	14.166522	188.9528	1053.39
141	FIN, ACK	958	14.166522			

Туршилт 2

TCP холболтын бөгнөрөлөөс зайлсхийх арга болох TCP Reno -ийн талаар судална. Доорх зургийг ажиглан асуултад хариулаарай.

1. *TCP Slow start* эхэлж байгаа хугацааны интервалыг тодорхойл.

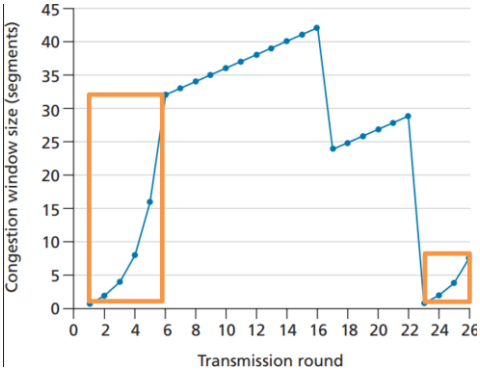


Figure 10 Slow start

Slow start нь [1,6], [23, 26] хугацааны интервалд үүсэж байна.

2. *TCP congestion avoidance* эхэлж байгаа хугацааны интервалыг тодорхойл.

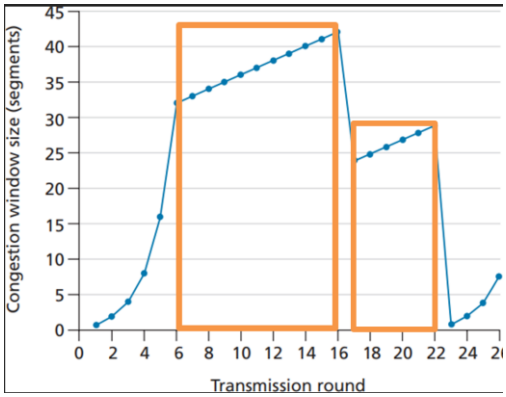


Figure 11 congestion avoidence

Congestion avoidance [6,16], [хүртэл үргэлжилж байна.

3. 16-р дамжууллын дараа, сегментийг *triple duplicate ACK* эсвэл *timeout* -ийн аль нь болсон эсэхийг тодорхойл.

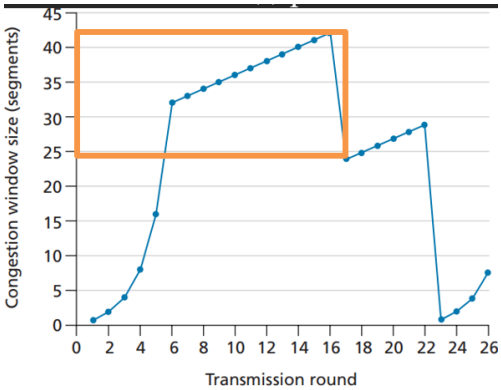


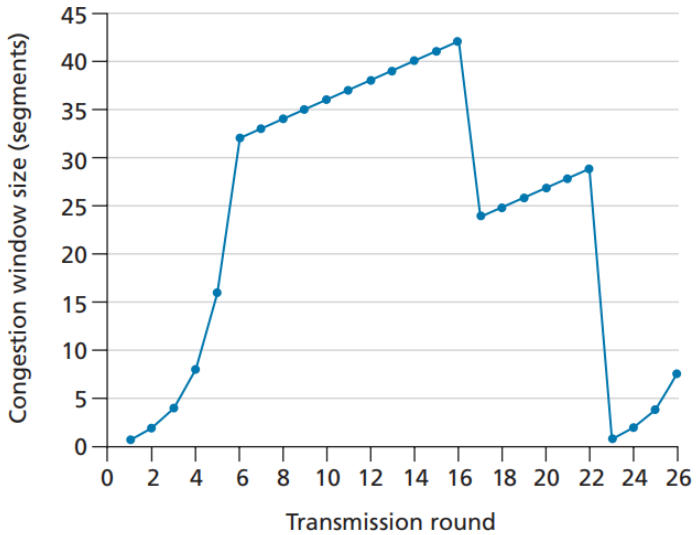
Figure 12 triple duplicate

16-р дамжууллын дараа графикт огцом уналт гарч, congestion window нь ойролцоогоор 42 сегментээс 24 орчим сегмент хүртэл буураад тэндээс дахин аажмаар нэмэгдэж байгаа нь triple duplicate ACK болж, TCP нь fast recovery механизмаар ажиллаж эхэлснийг илэрхийлж байна. Timeout болсон бол уналт илүү их байх байсан.

4. 22-р дамжууллын дараа, сегментийн алдагдлыг *triple duplicate ACK* илрүүлсэн эсвэл *timeout* болсон эсэхийг тодорхойл.

22-р дамжууллын дараа 0 сэгмент хүртэл буурсан тус timeout болсон.

5. Эхний дамжууллын үед *ssthresh* утга хэд байх вэ?
   Эхний дамжууллын үед *ssthresh* утга 32 байна.
6. 18 дахь дамжууллын үед *ssthresh* утга хэд байх вэ?
   16 дахь дамжууллын үеэр алдагдал илэрсэн үед congestion window-ийн хэмжээ 42 байсан тул 18 дахь дамжууллын үед *ssthresh* утга 42/2 = 21 байна.
7. 24 дахь дамжууллын ssthresh утга хэд болсон байна вэ?
   22 дахь transmission үеэр алдагдал илэрсэн үед congestion window-ийн хэмжээ 26 байсан иймд 24 дахь дамжууллын ssthresh утга 26/2 = 13 байна.
8. 70 дахь сегментийг дамжуулж байна гэж үзвэл, хэд дэх дамжууллын үед илгээгдэх вэ?
   70 дахь сегментийг 30-40 дамжуулалтын хооронд илгээгдэх магадлалтай.
9. 26 дахь дамжууллын дараа *triple duplicate ACK* хүлээн авч пакетын алдагдлыг илрүүлсэн бол, *congestion window size* утга болон *ssthresh*-ийн утга хэд болох вэ?
   Triple duplicate ACK хүлээн авах үед TCP нь congestion window-ийг бууруулдаг. Хэрэв 26 дахь дамжууллын үед congestion window-ийн утга 8 сегмент байна, шинэ congestion window ойроцоогоор 4 сегмент болж ssthresh нь 4 болно.
10. Дээрх тохиолдолд TCP tahoe аргыг ашиглаж байна гэж үзвэл, 16 дахь дамжууллын үед *triple duplicate ACK* хүлээн авсан гэе. Тэгвэл 19 дахь дамжууллын үед *ssthresh* болон *congestion window size* хэд байх вэ?
    16 дахь дамжууллын үед congestion window нь 42 сегмент байсан, Tahoe нь triple duplicate ACK хүлээн авснаар ssthresh-ийн утга нь cwnd-ийн хагас болж шинэчилнэ, иймд 21 сегмент болно. Үүний дараа, 17, 18, болон 19 дэх дамжууллуудад slow start эхэлж, congestion window дахин өсөж эхэлнэ.
    17 дахь дамжуулалтын дараа cwnd = 2 сегмент
    18 дахь дамжуулалтын дараа cwnd = 4 сегмент
    19 дахь дамжуулалтын дараа cwnd = 8 сегмент



Зураг 4.2 TCP Reno алгоритм

Дүгнэлт

Тээвэрлэлтийн давхаргын протоколууд нь хостуудын хооронд логик холболт үүсгэх, өгөгдлийг сегментээр дамжуулахад чухал үүрэгтэй. TCP (Transmission

Control Protocol) нь холболтод суурилсан протокол бөгөөд өгөгдлийг найдвартай, дараалалтайгаар дамжуулахад зориулагдсан. TCP-ийн three-way handshaking механизмаар логик холболтыг үүсгэж, сегментүүдийн дараалал, алдаагийн хяналтыг хангадаг. UDP (User Datagram Protocol) нь холболтод суурилаагүй протокол бөгөөд энгийн дамжуулалт хийхэд ашиглагддаг. UDP нь найдвартай байдал, дарааллыг хангадаггүй учраас хурдтай дамжуулалт шаарддаг стрийминг үйлчилгээ, дуу, видео зэрэгт хэрэглэгддэг. UDP протокол нь сегмент бүрийг хүлээн авагчид хүргэх эсэхийг нягтладаггүй бөгөөд энэ нь алдагдсан пакетуудыг анхаардаггүй.