

Šablona pro odevzdávání výstupů z distančních cvičení předmětu MPC-PKT určená k editaci a odevzdání po vytvoření PDF verze

Vaše jméno	Alex Sporni
VUT ID	204633
Vypracovaný lab (označení)	Lab3 (Srovnání unicastového a multicastového provozu v NS-3)

---

### 1. číslovaný úkol z návodu

Zadání úkolu: Na základě výše popsaného kódu pro nastavení unicastové aplikace na klienty n1 až n3, samostatně doplňte kód s nastavením pro klienta n4. Pomocí printscreenu zdokumentujte a vložte do protokolu vámi doplněné nastavení unicastové aplikace pro klienta 4.

```
onOffHelper onOffUnicast4("ns3::UdpSocketFactory", InetSocketAddress(interfaces13_4.GetAddress(1), 9));
onOffUnicast4.SetAttribute("PacketSize", UIntegerValue(1024));
onOffUnicast4.SetAttribute("DataRate", StringValue(appBitRate));
onOffUnicast4.SetAttribute("OffTime", StringValue ("ns3::ConstantRandomVariable[Constant=0]"));

ApplicationContainer unicastAppCont4 = onOffUnicast4.Install(nodes.Get(0));
unicastAppCont4.Start(Seconds(startTime));
unicastAppCont4.Stop(Seconds(endTime));
unicastAppCont4 = sink.Install(nodes.Get(4));
```

Řešení:

---

### 2. číslovaný úkol z návodu

Zadání úkolu: Pomocí printscreenu zdokumentujte a vložte do protokolu výstup z příkazového řádku po spuštění simulace unicastového vysílání. Výsledný výpis by měl odpovídat výstupu konzole výše.

Řešení:

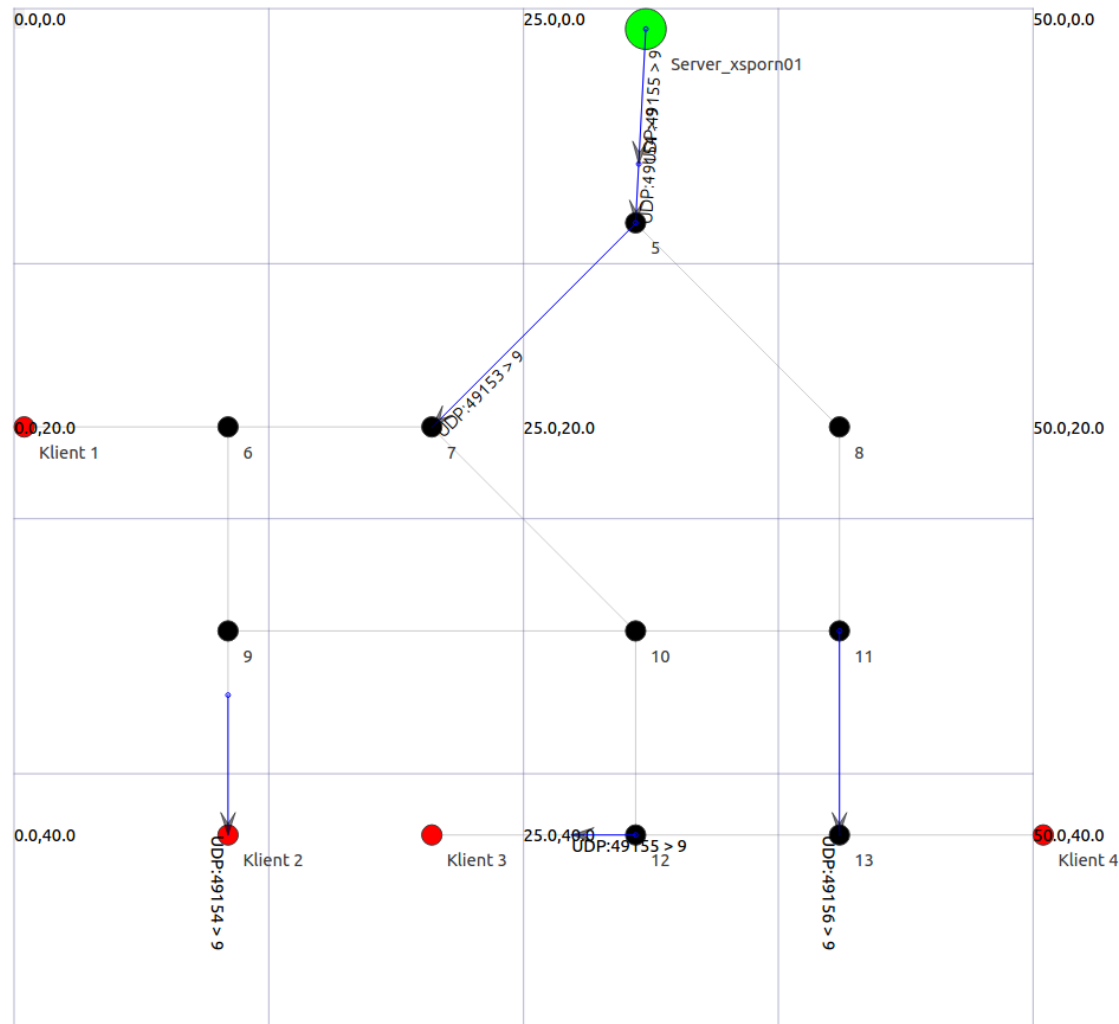
```
root@fekt: /home/student/ns-allinone-3.21/ns-3.21
At time 9.88397s packet sink received 1024 bytes from 10.1.1.1 port 49154 total Rx 195584 bytes
At time 9.8924s packet sink received 1024 bytes from 10.1.1.1 port 49155 total Rx 195584 bytes
At time 9.90084s packet sink received 1024 bytes from 10.1.1.1 port 49156 total Rx 195584 bytes
At time 9.9053s on-off application sent 1024 bytes to 10.1.11.2 port 9 total Tx 197632 bytes
At time 9.9053s on-off application sent 1024 bytes to 10.1.14.2 port 9 total Tx 197632 bytes
At time 9.9053s on-off application sent 1024 bytes to 10.1.7.2 port 9 total Tx 197632 bytes
At time 9.9053s on-off application sent 1024 bytes to 10.1.5.2 port 9 total Tx 197632 bytes
At time 9.90607s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 196608 bytes
At time 9.92493s packet sink received 1024 bytes from 10.1.1.1 port 49154 total Rx 196608 bytes
At time 9.93336s packet sink received 1024 bytes from 10.1.1.1 port 49155 total Rx 196608 bytes
At time 9.9418s packet sink received 1024 bytes from 10.1.1.1 port 49156 total Rx 196608 bytes
At time 9.94626s on-off application sent 1024 bytes to 10.1.11.2 port 9 total Tx 198656 bytes
At time 9.94626s on-off application sent 1024 bytes to 10.1.14.2 port 9 total Tx 198656 bytes
At time 9.94626s on-off application sent 1024 bytes to 10.1.7.2 port 9 total Tx 198656 bytes
At time 9.94626s on-off application sent 1024 bytes to 10.1.5.2 port 9 total Tx 198656 bytes
At time 9.94703s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 197632 bytes
At time 9.96589s packet sink received 1024 bytes from 10.1.1.1 port 49154 total Rx 197632 bytes
At time 9.97432s packet sink received 1024 bytes from 10.1.1.1 port 49155 total Rx 197632 bytes
At time 9.98276s packet sink received 1024 bytes from 10.1.1.1 port 49156 total Rx 197632 bytes
At time 9.98722s on-off application sent 1024 bytes to 10.1.11.2 port 9 total Tx 199680 bytes
At time 9.98722s on-off application sent 1024 bytes to 10.1.14.2 port 9 total Tx 199680 bytes
At time 9.98722s on-off application sent 1024 bytes to 10.1.7.2 port 9 total Tx 199680 bytes
At time 9.98722s on-off application sent 1024 bytes to 10.1.5.2 port 9 total Tx 199680 bytes
At time 9.98799s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 198656 bytes
At time 10.0069s packet sink received 1024 bytes from 10.1.1.1 port 49154 total Rx 198656 bytes
At time 10.0153s packet sink received 1024 bytes from 10.1.1.1 port 49155 total Rx 198656 bytes
At time 10.0237s packet sink received 1024 bytes from 10.1.1.1 port 49156 total Rx 198656 bytes
At time 10.0289s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 199680 bytes
At time 10.0478s packet sink received 1024 bytes from 10.1.1.1 port 49154 total Rx 199680 bytes
At time 10.0562s packet sink received 1024 bytes from 10.1.1.1 port 49155 total Rx 199680 bytes
At time 10.0647s packet sink received 1024 bytes from 10.1.1.1 port 49156 total Rx 199680 bytes
root@fekt: /home/student/ns-allinone-3.21/ns-3.21#
```

### 3. číslovaný úkol z návodu

Zadání úkolu: Pomocí printscreenu zdokumentujte a vložte do protokolu grafický výstup z programu NetAmin pro unicastový přenos. Pokuste se zachytit moment, ve kterém bude vidět, že pakety jdou od serveru ke všem klientům. Než vložíte obrázek do protokolu, přejmenujte uzel n0 ze „Server“ na „Server\_vášVUTlogin“.

Přejmenování proveďte vhodnou změnou ve zdrojovém kódu.

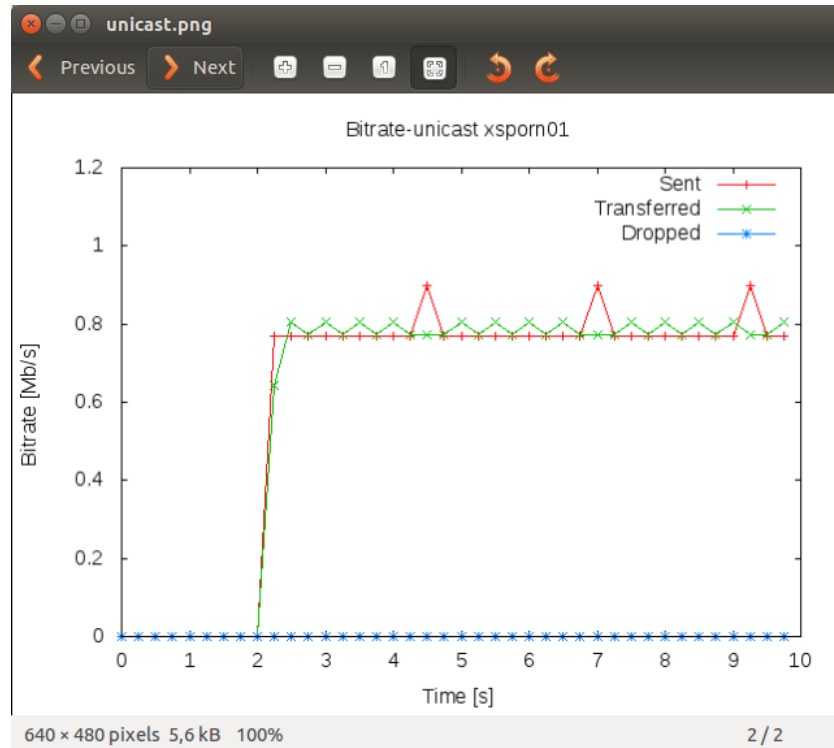
Řešení:



#### 4. číslovaný úkol z návodu

Zadání úkolu: Pomocí printscreenu zdokumentujte a vložte do protokolu výsledný graf vygenerovaný příkazem gnuplot. Výsledný graf by se měl být podobný jako obr. 4-1. Než graf vložíte do protokolu, přejmenujte název grafu z „Bitrate-unicast“ na „Bitrate-unicast\_vášVUTlogin“. Přejmenování provedte vhodnou změnou ve zdrojovém kódu.

Řešení:



#### 5. číslovaný úkol z návodu

Zadání úkolu: Pomocí printscreenu zdokumentujte a vložte do protokolu výstup z příkazového řádku po spuštění simulace multicastového vysílání. Výsledný výpis by měl odpovídat výstupu konzole výše.

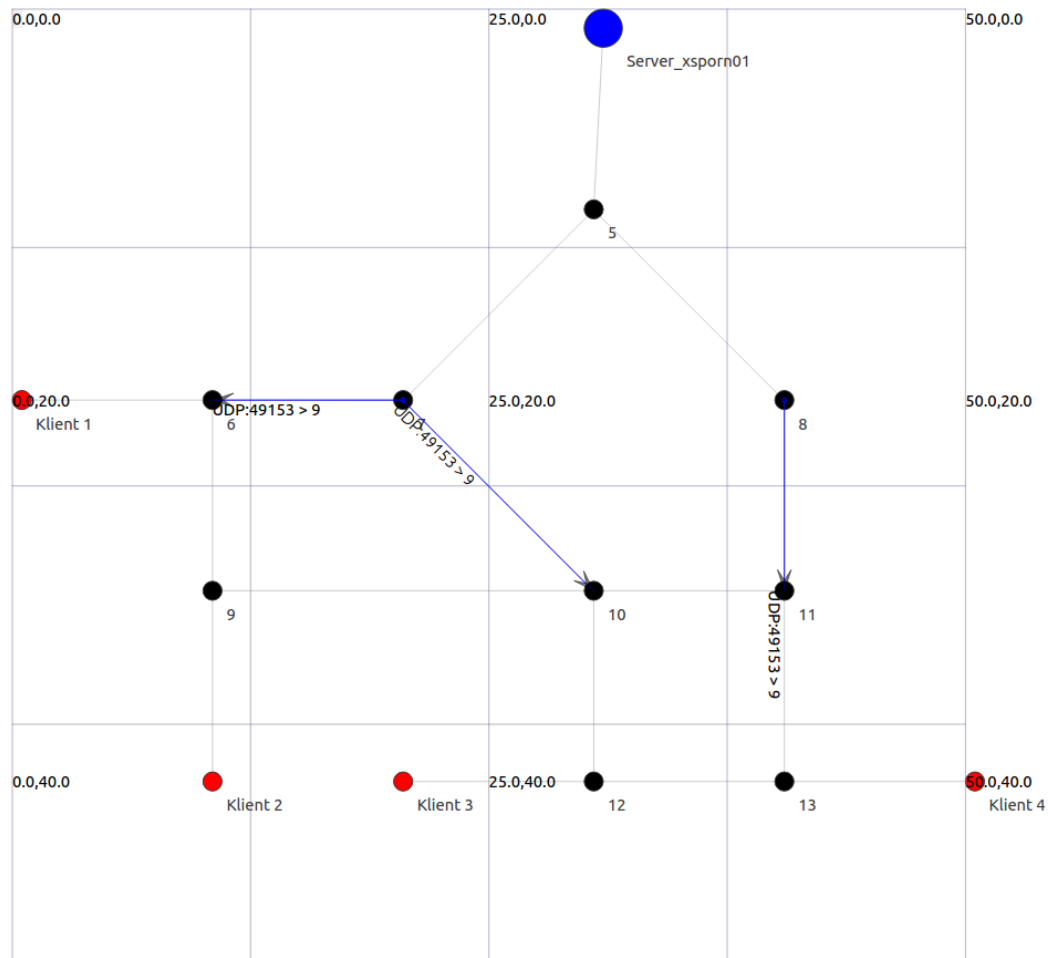
Řešení:

```
root@fekt: /home/student/ns-allinone-3.21/ns-3.21
At time 9.79362s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 193536 bytes
At time 9.79362s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 193536 bytes
At time 9.82338s on-off application sent 1024 bytes to 239.1.2.4 port 9 total Tx 195584 bytes
At time 9.82415s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 194560 bytes
At time 9.83458s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 194560 bytes
At time 9.83458s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 194560 bytes
At time 9.83458s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 194560 bytes
At time 9.86434s on-off application sent 1024 bytes to 239.1.2.4 port 9 total Tx 196608 bytes
At time 9.86511s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 195584 bytes
At time 9.87554s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 195584 bytes
At time 9.87554s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 195584 bytes
At time 9.87554s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 195584 bytes
At time 9.9053s on-off application sent 1024 bytes to 239.1.2.4 port 9 total Tx 197632 bytes
At time 9.90607s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 196608 bytes
At time 9.9165s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 196608 bytes
At time 9.9165s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 196608 bytes
At time 9.9165s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 196608 bytes
At time 9.94626s on-off application sent 1024 bytes to 239.1.2.4 port 9 total Tx 198656 bytes
At time 9.94703s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 197632 bytes
At time 9.95746s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 197632 bytes
At time 9.95746s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 197632 bytes
At time 9.95746s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 197632 bytes
At time 9.98722s on-off application sent 1024 bytes to 239.1.2.4 port 9 total Tx 199680 bytes
At time 9.98799s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 198656 bytes
At time 9.99842s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 198656 bytes
At time 9.99842s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 198656 bytes
At time 9.99842s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 198656 bytes
At time 10.0289s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 199680 bytes
At time 10.0394s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 199680 bytes
At time 10.0394s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 199680 bytes
At time 10.0394s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 199680 bytes
root@fekt: /home/student/ns-allinone-3.21/ns-3.21#
```

## 6. číslovaný úkol z návodu

Zadání úkolu: Pomocí printscreenu zdokumentujte a vložte do protokolu grafický výstup z programu NetAmin pro multicastový přenos. Pokuste se zachytit moment, ve kterém bude vidět, že se pakety na hraničních směrovačích duplikují, tak jak je tomu na obrázku níže (Obr. 5).

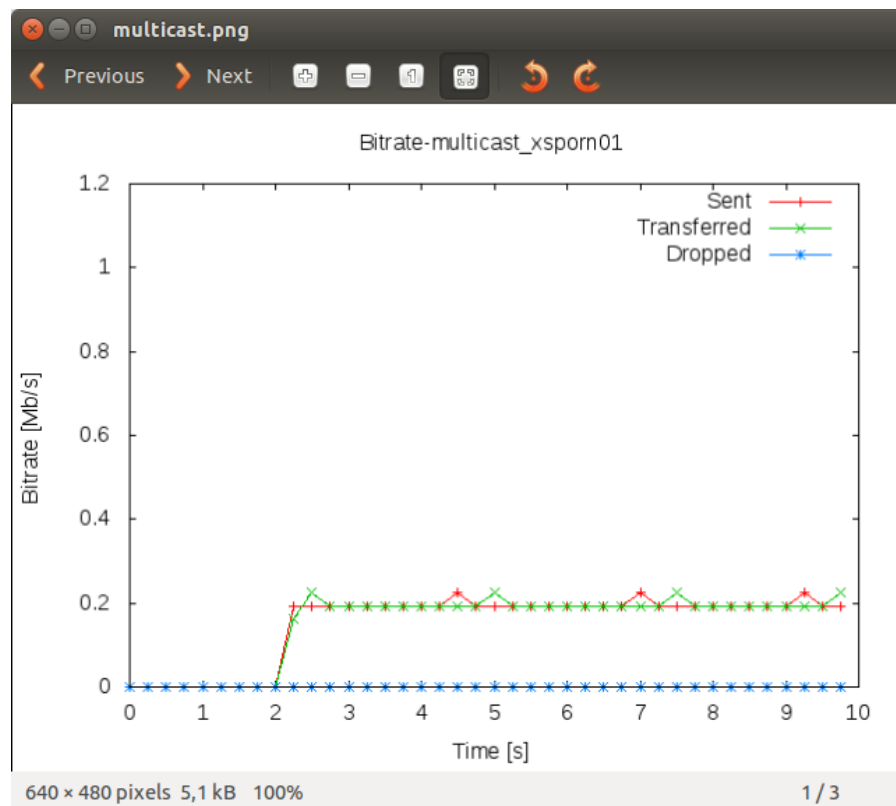
Řešení:



## 7. číslovaný úkol z návodu

Zadání úkolu: Pomocí printscreenu zdokumentujte a vložte do protokolu výsledný graf vygenerovaný příkazem gnuplot pro multicastový přenos. Výsledný graf by měl vypadat podobně, jako Obr. 6. Název grafu upravte na „Bitrate-multicast\_vášVUTlogin“.

Řešení:



## 8. číslovaný úkol z návodu

Zadání úkolu: **Vysvětlete princip unicastového a multicastového přenosu. Porovnejte animace v programu NetAnim pro unicastový a multicastový přenos, měli byste spatřit dva zásadní rozdíly. Za jakých podmínek a kde v síti je multicastový přenos výrazně efektivnější než unicastový?**

Řešení:

**Unicast:** Podstatou unicastového přenosu je zasílání paketů iba jednému cíli (uzol, stanica). Využíva sa v prípade priamej komunikácie medzi dvoma uzlami v sieti.

**Multicast:** V prípade multicastového přenosu prebieha zasílání paketů z jednoho zdroja skupine viacerým koncovým staniciam. Metóda funguje na princípe posielania informácií (s IP adresou zdroja a adresou cieľovej skupiny) zo zdrojového uzla cez spojenia v sieti (väčšinou routery) len jedným dátovým tokom a ak je informácia v lokálnej sieti vyžadovaná, tak sa informácia do nej replikuje. Cieľom technológie multicast je znížiť zahľtenie siete unicastovým přenosom.

**Porovnanie:** Z výstupov grafov v úlohách 4 a 7 vyplýva, že přenosová rýchlosť v prípade multicastového vysielania má 4x nižší bitrate (0.2 Mb/s) oproti unicastovému



prenosu (0.8 Mb/s). Multicastové vysielanie je efektívnejšie v sieťach nižšou prenosovou rýchlosťou linky, umožňuje participovať výrazne väčšiemu počtu zariadení na prenose a nezaťažovať tak prenosovú linku.

---

## 9. číslovaný úkol z návodu

Zadání úkolu: **Který protokol je využíván na transportní vrstvě pro multicastový přenos a proč?**

Řešení: Je to protokol UDP, ktorý na rozdiel od protokolu TCP nevyžaduje potvrdenie o príjme. Neprebehne 3-way handshake (connectionless). UDP sa taktiež využíva napr. pri DNS, streamovaní médií, prenose hlasu alebo online hier. Využívajú ho najmä serveri, ktoré odpovedajú veľkému množstvu požiadavok od klientov.

---

## 10. číslovaný úkol z návodu

Zadání úkolu: **Co jsou to distribuční stromy v případě multicastového přenosu a jaké typy distribučních stromů znáte? Vysvětlete jejich princip, výhody a nevýhody. Pomocí programu NetAnim zjistěte, jaký distribuční strom byl vytvořen manuálně v této úloze.**

Řešení: Distribuční stromy riešia tok dát po sieti, konkrétne od zdroja vysielania až ku koncovým adresátom. Podľa [zdroja](#) sa distribučné stromy delia na 2 typy. Prvý typ predstavuje Shortest-Path Tree (SPT) a druhý Shared Tree. V prípade SPT sa dáta posielajú najkratšou cestou v topológii, avšak každý zdroj vysielania musí poznať celú topológiu, aby mohol poslať cez najkratšiu možnú cestu. V prípade Shared tree sa v sieti nachádza tzv. rendezvous point (RP), ktorému routre forwardujú multicastové pakety a následne z tohto RP sú ďalej pakety forwardované ďalej po distribučnom strome. Výhodu oproti SPT predstavuje práve znalosť len jedného distribučného stromu, avšak nevýhoda môže byť dlhšie posielanie dát do tohto RP. V prípade nášho zadania bol použitý SPT, čiže dáta sa posielajú najkratšou cestou od vysielateľa k príjemcom. Podrobnejšie informácie sa nachádzajú na zdroji: <https://bit.ly/3kp9OSQ>

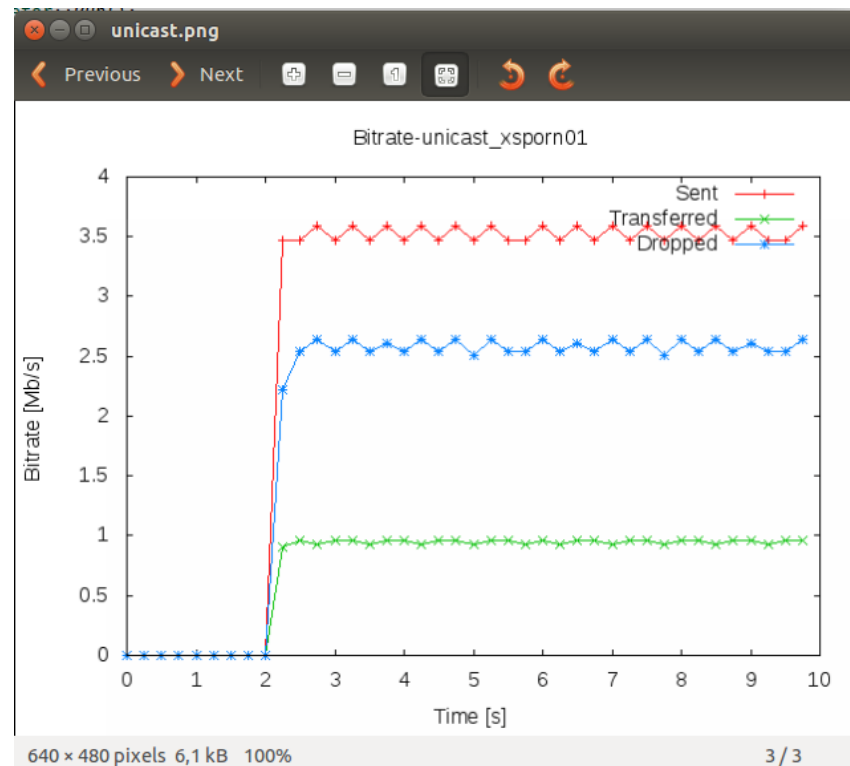
---

## 11. číslovaný úkol z návodu

Zadání úkolu: **Tento úkol se bude zabývat případem, kdy je bitová rychlost dat generována aplikacemi větší než přenosová rychlost linky. Nastavte simulaci pro posílání unicastových paketů a dále nastavte prenosovou rychlost aplikací (appBitRate) na hodnotu 0.9 Mb/s. Dále zakomentujte řádek pro nastavení rozsahu osy y v grafu. Po spuštění simulace byste měli dostat graf podobný tomu na Obr. 7. Je vidět, že dochází k zahazování paketů prakticky ihned po zahájení přenosu. To je dáno tím, že čtyři generované datové toky svými požadavky na prenosovou kapacitu linky n0 -> n5 výrazně překračují rychlost linky, která má pouze 1 Mb/s. Dalším faktorem pak je skutečnost, že délka fronty (queueLength) pro posílání je nastavena na relativně malou hodnotu, což vede k jejímu rychlému zaplnění. Výsledný graf zdokumentujte.**



Řešení:



## 12. číslovaný úkol z návodu

Zadání úkolu: **Jaká je souvislost mezi bitovým rychlostmi poslaných, úspěšně přenesených a zahozených dat v předchozím úkolu? V konzoli dále zjistěte, kolik bylo celkem v tomto scénáři zahozeno paketů.**

Řešení:

Na základe konzolového výpisu je možné vidieť, že súčet zahodených paketov predstavuje celkovo **2555 paketov**.

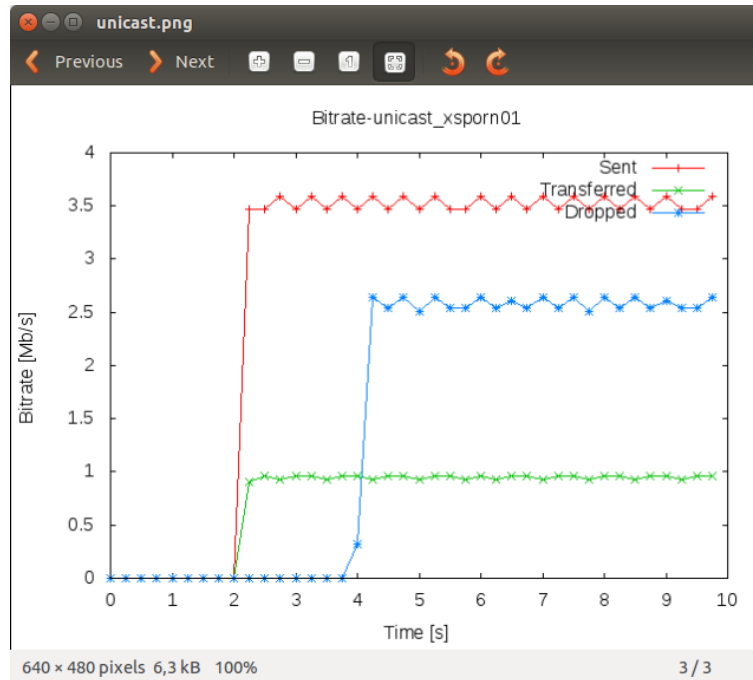
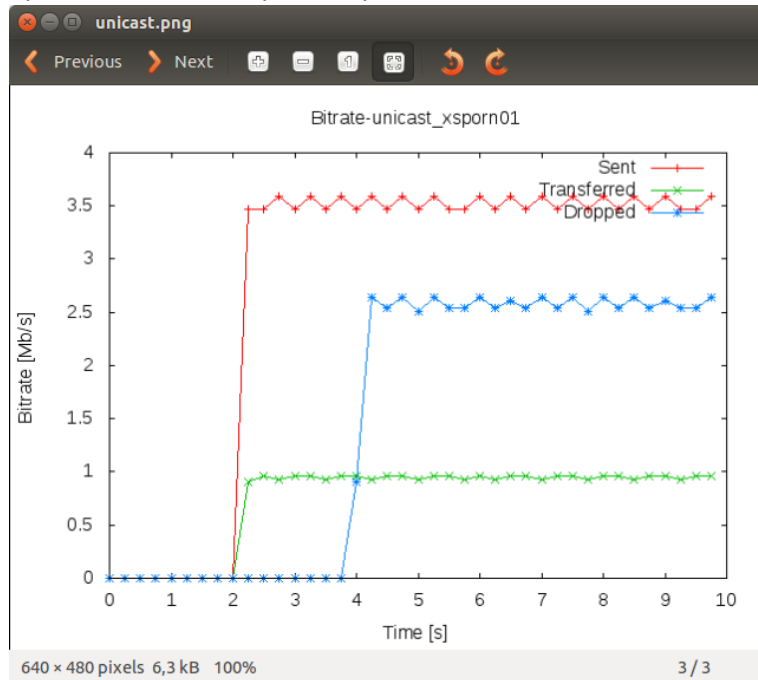
```
root@fekt: /home/student/ns-allinone-3.21/ns-3.21
Packet Drop Event. Total number of dropped packets in simulation (up to now):
2547
At time 9.97355s on-off application sent 1024 bytes to 10.1.14.2 port 9 total Tx 897024 bytes
Packet Drop Event. Total number of dropped packets in simulation (up to now):
2548
At time 9.97355s on-off application sent 1024 bytes to 10.1.7.2 port 9 total Tx 897024 bytes
Packet Drop Event. Total number of dropped packets in simulation (up to now):
2549
At time 9.97355s on-off application sent 1024 bytes to 10.1.5.2 port 9 total Tx 897024 bytes
At time 9.97691s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 883712 bytes
At time 9.98265s on-off application sent 1024 bytes to 10.1.11.2 port 9 total Tx 898048 bytes
Packet Drop Event. Total number of dropped packets in simulation (up to now):
2550
At time 9.98265s on-off application sent 1024 bytes to 10.1.14.2 port 9 total Tx 898048 bytes
Packet Drop Event. Total number of dropped packets in simulation (up to now):
2551
At time 9.98265s on-off application sent 1024 bytes to 10.1.7.2 port 9 total Tx 898048 bytes
Packet Drop Event. Total number of dropped packets in simulation (up to now):
2552
At time 9.98265s on-off application sent 1024 bytes to 10.1.5.2 port 9 total Tx 898048 bytes
At time 9.98534s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 884736 bytes
At time 9.99175s on-off application sent 1024 bytes to 10.1.11.2 port 9 total Tx 899072 bytes
Packet Drop Event. Total number of dropped packets in simulation (up to now):
2553
At time 9.99175s on-off application sent 1024 bytes to 10.1.14.2 port 9 total Tx 899072 bytes
Packet Drop Event. Total number of dropped packets in simulation (up to now):
2554
At time 9.99175s on-off application sent 1024 bytes to 10.1.7.2 port 9 total Tx 899072 bytes
Packet Drop Event. Total number of dropped packets in simulation (up to now):
2555
At time 9.99175s on-off application sent 1024 bytes to 10.1.5.2 port 9 total Tx 899072 bytes
At time 9.99377s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 885760 bytes
At time 10.0022s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 886784 bytes
At time 10.0106s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 887808 bytes
At time 10.0191s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 888832 bytes
```

### 13. číslovaný úkol z návodu

Zadání úkolu: **Vypočtete počet posílaných paketů (předpokládejte celkovou velikost 1052 B) za sekundu a počet paketů, který je schopen kanál přenést za sekundu. Podle těchto údajů nastavte délku fronty (queueLength) tak, aby k prvnímu zahazování došlo až v době 4 s (2 s po odstartování aplikací). Poznámka: do proměnné queueLength je nutné uložit počet paketů, nikoliv velikost v bajtech. Po spuštění simulace byste měli dostat graf podobný jako je na Obr. 8. Výsledný graf zdokumentujte, vhodně popište a vložte do protokolu.**

Řešení: Prenosová rýchlosť predstavuje 0.9 Mbps = 900 kbps, máme celkovo 4 streamy. Vieme, že prenosová rýchlosť linky je 1024 kbps. Rozdiel paketov, ktorú sú zahadzované je  $900 \cdot 4 - 1024 = 3600 - 1024 = 2576$  kbps. Vieme, že veľkosť posielaného paketu je 1052 B čo sa rovná  $1052 \cdot 8 = 8,416$  kbps. Zahadzovaných paketov potom bude  $2576 / 8,416 = 306$  paketov za sekundu. Keďže chceme aby sa začali zahadzovať až v dobe  $t = 4$  s, tak danú hodnotu 306 vynásobíme 2  $\rightarrow 306 \cdot 2 = 612$  paketov.

Je potrebné nastaviť queLength na hodnotu **612**. Avšak táto hodnota nie je úplne presná, postupnými experimentmi s navyšovaním queLength som došiel k záveru, že optimálna hodnota by mala byť na úrovni cca **630+3**.



#### 14. číslovaný úkol z návodu

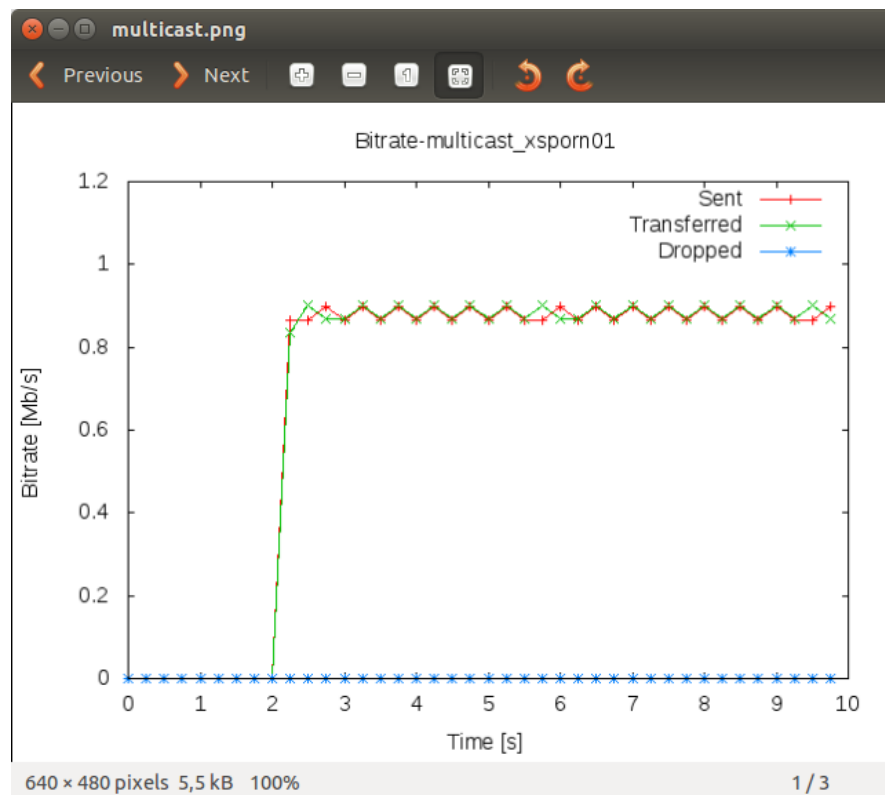
Zadání úkolu: **Jaké dopady má na přenášené pakety takovéto zvětšení délky fronty pro pakety k odeslání (viz předchozí úkol)? Představuje toto vhodné řešení daného problému?**

Řešení: Hlavný dopad představuje čas v ktorom sa začnú zahadzovať pakety. Vďaka zvýšeniu queLength na hodnotu **630** sa začali zahadzovať až v čase  $t = 4$  s od začiatku simulácie. Nie je to optimálne riešenie, lebo k zahadzovaniu paketov stále dochádza len v rozdielnom čase...

#### 15. číslovaný úkol z návodu

Zadání úkolu: **Upravte scénář pro multicastový přenos tak, aby aplikace využívala 0.9 Mb/s jako v úkolu č. 10 pro unicast. Délku fronty ponechte na výchozí hodnotě (10). Je kapacita přenosových kanálů v tomto případě dostatečná pro přenos bez zahazování paketů? Dokumentujte fungování přenosu grafem. Ověřte, že nedošlo k zahazení žádného paketu také ve výpisu do konzole.**

Řešení:



```
root@fekt: /home/student/ns-allinone-3.21/ns-3.21
At time 9.97355s on-off application sent 1024 bytes to 239.1.2.4 port 9 total Tx 897024 bytes
At time 9.97887s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 892928 bytes
At time 9.9802s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 891904 bytes
At time 9.9802s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 891904 bytes
At time 9.9802s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 891904 bytes
At time 9.98265s on-off application sent 1024 bytes to 239.1.2.4 port 9 total Tx 898048 bytes
At time 9.98797s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 893952 bytes
At time 9.9893s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 892928 bytes
At time 9.9893s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 892928 bytes
At time 9.9893s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 892928 bytes
At time 9.99175s on-off application sent 1024 bytes to 239.1.2.4 port 9 total Tx 899072 bytes
At time 9.99707s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 894976 bytes
At time 9.9984s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 893952 bytes
At time 9.9984s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 893952 bytes
At time 9.9984s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 893952 bytes
At time 10.0062s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 896000 bytes
At time 10.0075s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 894976 bytes
At time 10.0075s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 894976 bytes
At time 10.0075s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 894976 bytes
At time 10.0153s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 897024 bytes
At time 10.0166s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 896000 bytes
At time 10.0166s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 896000 bytes
At time 10.0166s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 896000 bytes
At time 10.0244s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 898048 bytes
At time 10.0257s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 897024 bytes
At time 10.0257s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 897024 bytes
At time 10.0257s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 897024 bytes
At time 10.0335s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 899072 bytes
At time 10.0348s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 898048 bytes
At time 10.0348s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 898048 bytes
At time 10.0348s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 898048 bytes
At time 10.0439s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 899072 bytes
At time 10.0439s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 899072 bytes
At time 10.0439s packet sink received 1024 bytes from 10.1.1.1 port 49153 total Rx 899072 bytes
root@fekt: /home/student/ns-allinone-3.21/ns-3.21#
```

Na základe nameraných hodnôt, ktoré je možné vidieť v konzolovom výpise alebo na grafe je možné prehlásiť, že kapacita prenosových kanálov je v tomto prípade **dostatočná** a **nedochádza** k zahadzovaniu paketov. Multicastové vysielanie sa javí ako ideálne riešenie na rozdiel od navýšenia hodnoty `queLength`.

## 16. Číslovaný úkol z návodu

Zadání úkolu: Vytvořený distribuční strom multicastu v zadané topologii se stará o to, aby se tok paketů dostal ke všem klientům. Upravte zdrojový kód v simulaci tak, aby byl distribuční strom více efektivní z pohledu celé sítě. Tzn., aby bylo tímto provozem zatíženo co nejmenší množství linek a uzlů a přesto se provoz dostal ke všem čtyřem příjemcům. Možné řešení znázorňuje Obr. 9. Úpravu kódu vložte do protokolu, fungování upraveného stromu dokumentujte vhodným obrázkem či obrázky z animace, grafem a situaci okomentujte. V konzoli zjistěte, kolik bylo celkem v tomto scénáři zahazeno paketů.

Řešení:

