Analiza RTP i RTCP protokola pomoću mrežnog analizatora

Vračić, Josip

SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU

FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA

Diplomski studij Procesnog računarstva

ANALIZA RTP I RTCP PROTOKOLA POMOĆU MREŽNOG ANALIZATORA

Diplomski rad

Josip Vračić

Osijek, 2022.

SADRŽAJ

1. UVOD

Rad se bavi analizom RTP protokola(engl. *Real Time Transport Protocol*) i RTCP protokola (engl: *Real Time Control Protocol*).

Nekada su mreže mogle prenositi samo jednu vrstu informacija, no današnjom tehnologijom istovremeno se prenosi više raznih vrsta podataka i informacija. Neophodno je bilo sve takve različite informacije umrežiti, sa svrhom međusobne komunikacije, stoga su sukladno tome definirani razni standardi i protokoli koji su i omogućili navedeno.

U suvremenim IP (engl. *Internet Protocol*) mrežama sve je veća potreba za isporukom audio-vizualnih sadržaja u stvarnom vremenu (npr. telefonija, video-konferencijske usluge, televizijske usluge i sl.). RTP protokol razvijen je kao standardizirana podrška prijenosa ovakve vrste sadržaja, te se uvijek koristi u kombinaciji sa kontrolnim protokolom RTCP.

Cilj je rada analizirati RTP i RTCP protokole, uključujući i različite scenarije njihove praktične primjene. Nadalje, također je potrebno uspostaviti multimedijski prijenos kao stvarnovremensku uslugu unutar više udaljenih točaka unutar IP mreže (primjerice telefonija ili video-konferencija). Pomoću mrežnog analizatora (Wireshark) cilj je također analizirati podatkovne tokove s ciljem analize RTP i RTCP protokola s naglaskom na kvalitetu i performanse navedenih.

Rad je podijeljen na više teorijskih cjelina. Prvo poglavlje rada je uvod, zatim se drugo poglavlje pod nazivom „Open Systems Interconnection“ odnosi na analizu slojeva navedenog modela, te su u sklopu iste cjeline obrađeni svi slojevi. Nadalje, treće se poglavlje odnosi na VoIP protokol (engl. *Voice Over Internet Protocol*), pri čemu je obrađena arhitektura istog protokola, njegove temeljne značajke, kao i kvaliteta usluge (engl. *Quality of Service*).

U četvrtom se poglavlju rada govori o UDP protokolu (engl. *User Datagram Protocol*), analizirane su njegove prednosti, mane i datagrami. Nadalje, analiziran je i obrađen TCP protokol (engl. *Transmission Control Protocol*). Šesta i sedma cjelina rada odnose se na analizu RTP I RTCP protokola, obrađene su njihove temeljne značajke i arhitektura, usluge koje pružaju, pojašnjena su zaglavlja i sl. Osmo poglavlje rada odnosi se na alat Wireshark, koji je korišten pri analizi samih protokola, dakle, obrađene su mogućnosti koje alat pruža i način na koji radi.

Zatim se u radu analizira audio sjednica, video sjednica i video konferencija, uz pomoć aplikacije Wireshark. Zaključno, kraj rada rezerviran je za zaključak u kojem će biti iznesen završni komentar autora.

2. OPEN SYSTEMS INTERCONNECTION (OSI)

Mrežni promet prvenstveno podrazumijeva cjelokupnu količinu transakcija koje se odvijaju putem mreže, te je iz same količine i vrste prometa koji se preko mreže odvija, moguće sznati mnoštvo podataka koji se odnose na svrhu i način upotrebljavanja računalne mreže.

Ukoliko se mrežni promet promatra sa stajališta računalne sigurnosti, sam proces analize prometa koji se odvija putem mreže onda može otkriti mnoštvo informacija kako o samoj računalnoj mreži, tako i o kontekstima korištenja iste, kao i o podacima koji se na taj način prenose. Kako bi bilo uopće moguće analizirati mrežni promet, važno je biti dobro upućen u samu teoriju, odnosno znati informacije o mrežnim protokolima, kao i određene standarde na kojima je zasnovana određena mreža koja se analizira [1].

Podaci koji se prenose putem računalne mreže, prenose se u obliku paketa, a svaki pojedini paket sadržava svoje zaglavlje, u kome su sadržane ključne informacije koje se odnose na sam paket. Također, navedene se informacije sastoje od izvorišne i odredišne adrese mreže, odredišnog i izvorišnog mrežnog *porta*, kao i mnoge druge podatke. Kako bi računalni sustavi uopće mogli imati mogućnost da obavljaju proces komunikacije, potrebno je da isti budu standardizirani [2]. Dakle, za standardizaciju računalnih komunikacija uspostavljen Open Systems Interconnection (u daljnjem tekstu: OSI) referentni model. Kao takav, on određuje i identificira recentne standarde koje računalni sustavi moraju slijediti, kako bi uopće bili u mogućnosti obavljati komunikaciju s ostalim računalnim sustavima. Ovaj model sadrži 7 slojeva, što će prikazati Slika 1., dok će objašnjenje navedenih slojeva biti dano u nastavku.

Slika 1. Slojevi OSI referentnog modela

2.1. Slojevi OSI modela

Kako je ranije navedeno, OSI sustav sastoji se od sedam različitih slojeva, dok će u nastavku biti detaljno obrađen svaki od navedenih slojeva OSI modela, koje je prikazala Slika 1.

2.1.1. Fizički i podatkovni sloj

Fizički sloj (engl. *physical*) najniži je sloj, koji prvenstveno služi kako bi se mogao uspostaviti fizički kanal među računalnim sustavima, koji dakle međusobno komuniciraju. Ovaj se sloj odnosi na osiguravanje postupka prenošenja jedinica informacija te se uz navedeno bavi i sklopovljem i električnim značajkama signala, a u sebi sadrži medije, konektore, mehanička svojstva, proceduru uspostave, odnosno prekida veze te električka svojstva [3]. Podatkovni sloj (engl. *data link*) pak s druge strane osigurava prenošenje podataka među računalnim sustavima, koji obavljaju komunikaciju, zajedno s kontroliranjem prijenosa. Također, on uobličava pakete i podatkovne okvire na takav način da dodaje i uklanja zaglavlja, obavlja provjeru ispravnosti podataka te upravlja protokom navedenih.

Unutar podatkovnog sloja postoji *Ethernet* protokol, a u *Ethernet* mreži sva računala imaju vlastitu 48 bitnu MAC (engl. *Media Access Control*) adresu, temeljem koje se podaci šalju prema određenim računalima, kod takvih mreža je specifičan ARP (engl. *Address Resolution Protocol*) protokol, koji obavlja pretvorbu IP adrese u MAC adresu [1].

2.1.2. Mrežni sloj (Network)

Unatoč činjenici da je mrežnom sloju prije svega svrha prenošenje paketa i njihovo usmjeravanje, njegova je zadaća ipak znatno opsežnija. Temeljni je protokol kod ovog sloja IP (engl. *Internet Protocol*), a koji obavlja spajanje računalnih mreža raznih vrsta tehnologija, primjene te topologija. U protokole navedenog sloja ubrajaju se IP i ICMP protokoli, dok se u protokole transportnog sloja ubrajaju TCP i UDP protokoli o kojima će biti više rečeno u nastavku rada. Dakle, temeljna je funkcija ovog sloja prvenstveno usmjeravanje i adresiranje mrežnih paketa. Unutar sloja upotrebljava se i ICMP (engl. *Internet Control Message Protocol*)

protokol, koji se primarno koristi kako bi se otkrilo koja su aktivna računala u mreži i dijagnosticiralo različite pogreške unutar mreže [1].

2.1.3. Transportni sloj (Transport)

U ovom su sloju sadržani TCP (engl. *Transport Control Protocol*) kao i UDP (engl. *User Datagram Protocol*) protokoli. Kada dođe do prenošenja određenih podataka, TCP protokol radi određenu sjednicu između poslužitelja i klijenta, a s druge strane, UDP protokol se fokusira na prenošenje paketa, no kontrola postupka prenošenja na razini tog protokola tu izostaje. Ključna obilježja TCP protokola tako su: pouzdanost (budući da svaki poslani paket sadrži vlastiti potvrdni broj), kompletna dvosmjerna komunikacija (odnosi se na to da je moguće istovremeno i primati i slati podatke) te sigurnost (zato što neovlašteni korisnici nisu u mogućnosti ubacivati lažne podatke u sjednicu koja je otvorena, osim ako on sam nema mogućnost nadgledanja te sjednice) [1]. Transportni sloj dijeli informacije na pakete, osigurava točan, odnosno pravilan redoslijed paketa, dijeli razgovor na više veza i obrnuto te upravlja prijenosom s kraja na kraj.

2.1.4. Sjednički, prezentacijski i aplikacijski sloj (Session, Presentation, Application)

Sjednički sloj je sloj koji je prije svega namijenjen ostvarivanju kao i održavanju sjednice između određenih mrežnih aplikacija. Dakle, prvenstveno se bavi uspostavom veze među krajnjim korisnicima, te sinkronizacijom, stoga na taj način omogućuje komuniciranje putem mreže i podjelu funkcija koje korisnici imaju nad postavljenim zadacima, no najjednostavnije ga se može pojasniti kod videa, u situaciji kada se ne želi imati ton bez slike, odnosno, slika bez tona.

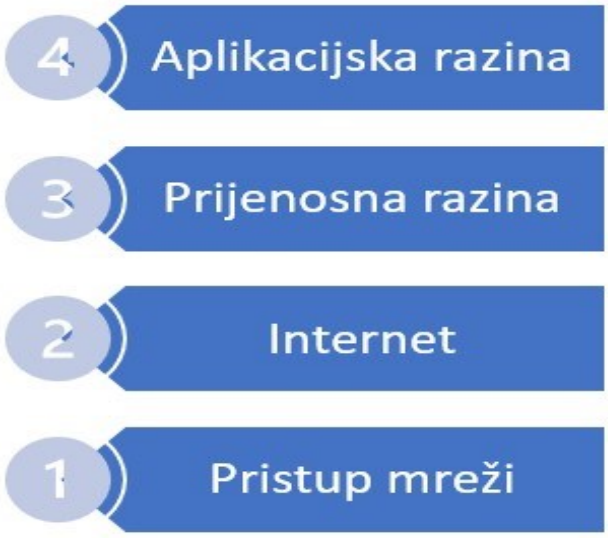
S druge strane, prezentacijski sloj aplikaciji podatke daje u obliku koji joj je shvatljiv, dok se aplikacijski sloj odnosi na samu mrežnu aplikaciju i obavlja određene zadatke gdje se misli na primjerice e-poštu, prenošenje datoteka i sl. Svaki OSI sloj dakle, kao što je iz prethodnih dijelova vidljivo, posjeduje svoju svrhu i namjenu, a svi slojevi skupa osiguravaju standardizirane komunikacije putem mreže. Dakako, za proučavanje su mrežnih transakcija najkvalitetniji od nabrojanih OSI slojeva sljedeći slojevi: podatkovni, mrežni i transportni sloj.

2.2. TCP/IP arhitektura (Transmission Control Protocol/Internet Protocol)

Ovaj je protokol, odnosno arhitektura najzastupljeniji standard na internetu, prvenstveno zbog prednosti koje je navedeni protokol u određenom periodu jedini mogao ponuditi, a neke su od njih primjerice:

- Samostalnost, odnosno, neovisnost o vrsti sklopovlja i operacijskim sustavima koje korisnici upotrebljavaju, kao i o pojedinom proizvođaču,
- TCP/IP nije ovisan o vrsti mrežne opreme, što dakle vodi ka integraciji raznih tipova mreža,
- Unikatan način adresiranja koji omogućuje spajanje u komunikacijski proces onih uređaja koji mogu podržavati navedeni protokol i sl. [4].

TCP/IP protokol nastao je od implementacije OSI modela, za razliku od OSI modela koji provodi vertikalni pristup, dok TCP/IP protokol provodi horizontalni pristup. Valja spomenuti kako TCP/IP zapravo kombinira dva sloja: sloj sesije i sloj prezentacije, sve u jednom sloju, odnosno sloju aplikacije, dok OSI ima drugačiji pristup prezentaciji, s različitim sesijama i slojevima [5]. Slika 2. daje prikaz razina TCP/IP arhitekture, dok će u nastavku biti objašnjene funkcije svake od tih razina.



Slika 2. Razine TCP/IP arhitekture

Razina pristupa mreži odnosi se na osiguravanje pristupa zajedničkom mediju od strane uređaja, te kao takva sadrži značajke fizičke razine, odnosno, razine podatkovne veze OSI modela [4].

Nadalje, internetska razina odnosi se na značajke i obilježja mrežne razine kod OSI modela, a temeljni je protokol internet razine IP protokol, čemu će se posvetiti pažnja kroz nastavak rada.

Prijenosna razina u ekvivalentnom je odnosu prema prijenosnoj razini OSI modela, dok su jedni od primarnih protokola u prijenosnoj razini tako TCP (engl. *Transmission Control Protocol*) i UDP (engl. *User Datagram Protocol*), čemu će pažnja također biti posvećena kroz sljedeća poglavlja rada.

Aplikacijska razina sastoji se od sjedničke, prezentacijske i aplikacijske razine OSI modela. Aplikacija sama po sebi zapravo svaki pojedini proces koji se odvija između prijenosne razine TCP/IP arhitekture. Slika 3. daje prikaz odnosa OSI referentnog modela i TCP/IP arhitekture.

Slika 3. Odnos OSI referentnog modela i TCP/IP arhitekture

Dakle, iz slike je vidljivo kako TCP/IP model ima nešto manje slojeva od OSI referentnog modela, odnosno, svega 4, a u njima su obuhvaćene sve funkcionalnosti koje ima i

OSI model. Aplikacijska razina kod TCP/IP modela ima podjednaka obilježja gornja tri sloja OSI modela, dok pristup mreži TCP/IP modela sadržava podjednaka obilježja donja dva sloja OSI modela, kako je na Slici 3. i prikazano. I jedan i drugi model upotrebljavaju slojeve za prikazivanje komunikacije, te navedeni slojevi zato imaju i podjednake uloge, a također koriste tehnologiju koja se odnosi na prespajanje paketa (engl. *packet-switched*) [6].

Takva tehnologija zapravo opisuje odašiljanje podataka koji su upakirani u male jedinice takvih podataka, a one se zovu paket, a potom se oni preusmjeravaju preko mreže pritom dakle koristeći odredišnu adresu, koju sadržava paket. Kada se podaci za slanje dijele u pakete, tako je moguće da se identične komunikacijske veze, odnosno linije, dijele među većim brojem korisnika u mreži, a takav se oblik komuniciranja, odnosno komunikacije, naziva *connectionless* [7].

Dakle, iako je evidentno da postoje sličnosti između ova dva modela, postoje i mnogobrojne razlike, tako primjerice, TCP ima kako je već navedeno, 4 sloja, dok OSI ima 7 slojeva. OSI model ima nešto strože granice, dok TCP/IP nema. Nadalje, TCP/IP u svom aplikacijskom sloju koristi sloj sesije i prezentacijski sloj, dok OSI upotrebljava različite sesije i prezentacijske slojeve. TCP/IP ovisi o protokolu, dok je OSI neovisan od protokola.

3. VOICE OVER INTERNET PROTOCOL (VoIP)

Voice Over Internet Protocol (u daljnjem tekstu: VoIP) tehnologija osigurava proces korištenja računalne mreže koja se koristi kod glasovnih aplikacija, a tu se misli na primjerice poruke u audio obliku, telefoniju ili pak telekonferencije. Također, navedena tehnologija određuje i način prenošenja glasovnih poziva preko IP mreže, što sadrži i digitalizaciju, odnosno, paketizaciju glasovnih tokova. Navedeni sustav glas pretvara u signal digitalnog oblika, a isti se pritom može prenositi putem IP mreže. Ukoliko se poziva klasični telefonski broj, u toj se situaciji onda signal pretvara u klasični telefonski signal, i to prije nego što uopće dođe do odredišta [8].

3.1. Arhitektura VoIP mreže

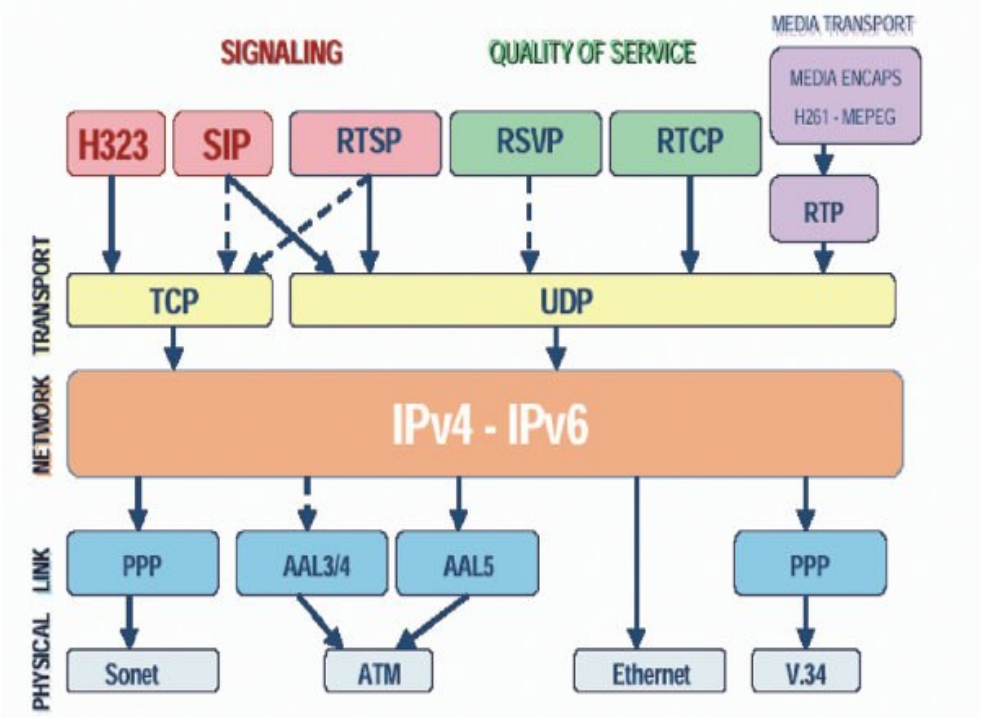
Arhitektura i temeljne komponente koje su potrebne za realnu implementaciju VoIP-a prikazane su na Slici 4., a one su:

- Mrežna infrastruktura: ona podržava VoIP tehnologiju, te se uz navedeno, može smatrati jednom logičkom glasovnom mrežom, koja je distribuirana putem IP okosnice, a ona pritom osigurava mogućnost spajanja kao i prenošenja audio paketa, dakle, svrha je navedene IP infrastrukture usmjerena ka omogućavanju prenošenja glasovnih paketa bez ikakvih poteškoća.
- Procesori (odnosno kontrolori) poziva: ovdje se misli na module koji su prijeko potrebni kako bi se poziv mogao uspostaviti i nadzirati, autoriziranje korisnika, pružanje bazičnih telefonskih usluga te za nadziranje, odnosno, kontroliranje brzine prijenosa.
- Prevoditelji: oni su potrebni kako bi do poziva uopće moglo doći, zapažanje poziva, postupak pretvorbe glasa iz analognog oblika u digitalni oblik.
- Korisnički VoIP terminali
- Osobna računala

Protokoli koji se koriste u radu današnjih VoIP sustava dijele se na dvije skupine a to su:

1) protokoli namijenjeni prijenosu multimedijskog sadržaja

2) protokoli namijenjeni prijenosu signalizacijskog sadržaja.



Slika 4. Arhitektura VoIP mreže [9]

3.1.1. Protokoli za prijenos audio signala

Navedenim se protokolima uloga prvenstveno odnosi na isporuku audiosadržaja, videosadržaja, te na kontrolu *media* servera, a isti obavlja kontrolu i uspostavu medijske sesije, nadzire prijenos podataka te rezervaciju resursa određenim aplikacijama. Dakle, navedeni su protokoli sljedeći: RTP (engl. *Real Time Transport Protocol*), RTSP (engl. *Real Time Streaming* Protocol), RTCP (engl. *Real Time Control Protocol*) i RSVP (engl. *Resource Reservation Protocol*).

RTP protokol zapravo radi na osiguravanju funkcije potrebne kako bi se mogli mrežno prenositi podaci u stvarnom vremenu, a takvi su podaci primjerice video podaci, a oni se upotrebljavaju za određene simulacije putem *unicast* ili *multicast* usluga. . Također, navedeni protokol ima mogućnost podržavanja određenih usluga kao primjerice identificiranje vrste

prenešenih podataka te nadziranje dostavljanja podataka [10]. O ovom će protokolu biti detaljnije rečeno u šestom poglavlju ovog rada.

RTSP protokol je protokol aplikacijskog nivoa, koji je prije svega namijenjen kontroliranju procesa dostavljanja podataka, sa svojstvima koja se odnose na isporuku u stvarnom vremenu, te se tu prvenstveno misli na audio i video. Ovaj protokol ne traži konekciju, dakle, server sam održava sesiju koja ne ovisi o vezi prijenosnih, odnosno transportnih slojeva [10]. Također, nije zadužen niti za prenošenje samih podataka, a RTSP serveri uobičajeno za prenošenje podataka koriste RTP i RTCP protokole. Ključne su značajke, odnosno funkcije ovog protokola dohvaćanje multimedijskog sadržaja sa medijskog poslužitelja, poziv medijskog poslužitelja na konferenciju, te dodavanje medija na postojeću sjednicu.

RTCP protokol jest protokol koji obavlja nadziranje, odnosno kontroliranje prijenosa podataka na velikim *multicast* mrežama, čime direktno potpomaže kod rada RTP protokola. Korišten je uglavnom u VoIP mrežama te u videokonferencijama. Ovaj će protokol biti detaljnije obrazložen u sedmom poglavlju ovog rada. RSVP protokol se koristi u situaciji kada je potrebno zatražiti određenu kvalitetu odabrane usluge od mreže, za pojedini *data stream*. Također, koristi se i od strane *routera* kako bi dostavio zahtjev za kvalitetom usluge (u daljnjem tekstu: *QoS-Quality of Service*) svim pojedinim čvorovima kroz koje prolazi *data stream*. Također, uz pomoć se navedenog protokola ne dolazi do postupka prenošenja podataka, nego se on upotrebljava samo kao kontrolni protokol. Nadalje, on ne služi kao *routing protokol*, nego se zapravo oslanja na *unicast i multicast routing* protokole [11].

Ovaj protokol se odnosi na skup pravila koja identificiraju i određuju kanale komunikacije, koji se koriste kod *multicast* prijenosa videozapisa te ostalih poruka. Dio je *Internet Integrated Services* modela (u daljnjem tekstu: IIS), a isti pritom ostvaruje najkvalitetniju uslugu u realnom vremenu te radi na kontroliranom dijeljenju veza. [12].

3.1.2. Signalizacijski protokoli

Signalizacija je po svojoj definiciji zapravo sposobnost razmjenjivanja te generiranja kontrolnih informacija korištenih za praćenje, upravljanje te otpuštanje veza između krajnjih točki. Glasovna signalizacija traži mogućnost pružanja funkcija nadziranja, adresiranja te

upozorenja među čvorovima [13]. Tri najučestalije korištena signalizacijska protokola danas jesu H.323, SIP (engl. *Session Initiation Protocol*) te MGCP (engl. *Media Gateway Control Protocol*).

H.323 je standard, odnosno signalizacijski protokol, koji određuje na koji to način određeni uređaji vrše komunikaciju preko mreže, iako ista sama po sebi ne može jamčiti kvalitetu pojedine usluge (kao što je primjerice Internet). H.323 terminali i njihova oprema imaju mogućnost prenošenja videa u realnom vremenu, jednako kao i glasovno razgovora, podataka, odnosno kombinira te elemente. Proizvodi koji koriste navedeni standard za zvuk i sliku omogućuju komunikaciju i spajanje sa drugim korisnicima preko Interneta, isto kao što i korisnici koji imaju različite telefone mogu međusobno komunicirati [13]. Temeljna obilježja H.323 standarda su sljedeća:

- Uobičajena kompresija, odnosno, dekompresija,
- Povezivanje i spajanje različite opreme,
- Ovaj protokol nije ovisan o mreži,
- Pruža podršku za uspostavu konferencijske veze,
- Obavlja nadziranje mreže te
- Može podržavati postupak komuniciranja s nekoliko različitih krajnjih točaka [14].

Ovaj je protokol tako trenutno najsloženiji, odnosno najkompliciraniji, no i najpotpuniji standard koji vrši obradu više kategorija, kao što su to primjerice: kompresiranje i prenošenje podataka kao što su govor i slika u stvarnom vremenu, kontroliranje parametara koji se odnose na kvalitetu veze i uspostavljanje iste, autorizaciju, registraciju korisnika i sl. Temeljne su mu komponente terminali (kao što je to primjerice *Microsoft NetMeeting*), mrežni prolazi (engl. *gateway*) koji povezuju H.323 zone i druge mreže i sl. [15]. Također, ovaj protokol sadrži određene različite protokole a to su primjerice: audio/video CODEC, H.225 signal poziva, H.245 signal kontrole i drugi.

Sesijski inicijacijski protokol, odnosno SIP, jest signalni protokol kojeg se upotrebljava kada se uspostavlja VoIP veza. Odnosno, napravljen je zapravo kao alternativa H.323 signalizaciji, a primarno se koristi onda kada se želi uspostaviti ili prekinuti glasovne ili videopozive, no istovremeno ga se može upotrebljavati i kod modificiranja postojećih poziva, primjerice, kako bi se moglo dodati ili ukloniti određenog sudionika u pozivu. Ovaj protokol

koristi model koji funkcionira na principu upit-odgovor, tako da se određena SIP transakcija sastoji od upita upućenog od strane klijenta, a ista mora biti praćena s najmanje jednim odgovorom koji upućuje poslužitelj. Ovaj protokol dakle, ne može prenositi zvuk od izvora do odredišta. Uglavnom koristi *User Datagram Protocol* (u daljnjem tekstu: UDP) kao protokol transportnog sloja, iz razloga što UDP ne traži slanje potvrde o prijemu paketa te na taj način omogućuje razgovor u realnom vremenu [15].

Temeljni su dijelovi SIP arhitekture sljedeći: korisnički agenti, a to može svaki klijent, odnosno uređaj koji traži SIP spajanje (primjerice IP telefon), zatim posredni poslužitelj (kojemu je funkcija pronalaženje korisnika te prevođenje adresa), identifikacijski poslužitelj (koji prihvaća identifikacijske zahtjeve), preusmjerivački poslužitelj (koji prima zahtjeve i onda daje odgovor s O ili nekim drugim adresama za uspostavu veze) i locirajući poslužitelj (koji služi za to da se može pronaći trenutna lokacija, odnosno IP adresa korisnika).

3.2. Temeljne značajke VoIP sustava

VoIP odražava mogućnost održavanja telefonskih poziva i slanja fakseva putem mreža za prenošenje podataka te se odnosi i na zadovoljavajuću razinu kvalitete usluge te adekvatnog međuodnosa između cijene i troškova. Često se u praksi VoIP naziva i IP telefonijom. Važno je razumjeti to da se VoIP uvelike razlikuje od PSTN (engl. *Public Switched Telephone Network*).

U situaciji gdje se upućuje jedan, laički rečeno, "obični" PSTN poziv, uspostavljen je virtualno spojeni krug među izvorom i odredištem, za svo vrijeme dok poziv traje, dok kod VoIP-a navedeni krug zapravo uopće ne postoji. Analogni signal u komu je sadržan glas,koji je kodiran u digitalnom obliku, dijeli se na više različitih paketa, a svaki se taj paket neovisno prenosi bilo putem privatnih mreža, ili preko Interneta koristeći pritom IP protokol. Sukladno tome, VoIP se zasigurno s pravom može shvaćati kao direktan konkuren u PSTN mreži kod industrije koja pokriva usluge telefonije.

Što se tiče povijesnog razvoja, podrijetlo VoIP-a datira još od 1994. godine kada je došlo do pojave jednostavnog *shareware* programa za stolna računala koji je imao mogućnost takozvanog glasovng *chata* koji se odvijao putem interneta. U današnje vrijeme, sadržaj VoIP-a je širok, te on zapravo podržava video u stvarnom vremenu, višestranačku konferenciju, faks

prijenose, i sl. Unatoč činjenici da danas postoji najviše IP mreža, ugradnja VoIP-a na njima i dalje nužno traži izuzetno velika ulaganja i napore [16].

Kao i kod ostalih usluga, tako ovdje, korisnici traže da kvaliteta usluge bude na visokom nivou, a neki su od takvih uvjeta koji klijenti očekuju primjerice: besprekidna usluga, koja se odvija tako da gubitci i kašnjenja praktički ne postoje, važna je i sigurnost samih korisnika i njihovih vlastitih podataka i sl. U nastavku će biti analizirana kvaliteta usluga VoIP mreže, kao i potencijalne sigurnosne prijetnje i metode zaštite od mogućih prijetnji.

3.3. Kvaliteta usluge (Qualiy of Service)

Quality of Service (u daljnjem tekstu: QoS) odnosi se na kvalitetu usluge, odnosno, diferencijaciju vrste prometa i vrsta usluga. Više je tražen na privatnim mrežama, nego na Internetu i ISP mrežama. Važan je alat za VoIP uspjeh, dakle, tu se kvaliteta odnosi na biti u stanju slušati i govoriti jasnim i neprekinutim glasom, bez neželjenih smetnji. Kvaliteta ovisi o sljedećim faktorima: gubitak podataka, dosljedne karakteristike kašnjenja i latencije [17]. QoS je zapravo temeljni problem u implementaciji VoIP-a.

Kako bi bilo moguće ugraditi VoIP na način gdje bi korisnici dobili zadovoljavajući nivo kvalitete glasa, u VoIP prometu mora se osigurati određena pojasna širina, latencija te *jitter*. Općenito gledajući, QoS omogućava kvalitetniju uslugu na mreži obilježjima koja slijede u nastavku:

- Kod pružanja podrške namjenskoj širini pojasa,
- Pruža poboljšanje određenih karakteristika gubitaka,
- Usmjeren je na izbjegavanje i upravljanje zagušenjima mreže,
- Oblikuje mrežni promet te
- Postavlja prioritete prometa putem mreže [18].

Također, uvećanjem kompleksnosti VoIP sustava, došlo je i do pojave mnogobrojnih prijetnji, koje potencijalno mogu ugroziti sigurnu komunikaciju putem VoIP-a. Takvi se napadi na sigurnost mogu podijeliti na dvije temeljne skupine: pasivne i aktivne napade.

Pasivnim je napadima moguće doći do određenih podataka čiji se proces prenošenja odvija na mreži, no time se ipak ne mijenja sadržaj tih podataka, a u tom se slučaju mogu presretati poruke, koje sadržavaju osjetljive informacije, odnosno moguće je analizirati algoritam komunikacije. Jedan od ključnih problema u VoIP sustavima jest taj da se u određenim situacijama ipak ne upotrebljava nikakva zaštita, stoga se neke informacije i podaci, koji mogu biti primjerice korisničko ime ili lozinka mogu lako saznati, takozvanom metodom gdje se razgovor prisluškuje.

S druge strane, napadi koji su aktivne prirode, vrše izmjene sadržaja poslane poruke, odnosno, rade lažne tokove podataka. U ovakve se napade ubraja primjerice *DoS Attack* (otkaz servisa), neovlašteni pristup, napad na aplikacije, napad na protokole itd. Ovakvi napadi uglavnom se mogu riješiti detekcijom i oporavkom [19].

4. USER DATAGRAM PROTOCOL (UDP)

User Datagram Protocol (u daljnjem tekstu: UDP) jednostavan je protokol, i on se nalazi u prijenosnom, odnosno transportnom sloju OSI referentnog modela, te je uz navedeno jedan od ključnih internetskih protokola zajedno s TCP uslugom, uz činjenicu da je TCP nešto pouzdaniji od njega. Dakle, UDP je nespojni protokol, stoga on prije nego li dođe do prijenosa podataka, ne zahtijeva nužno uspostavu veze između dvaju točaka interneta. UDP se tako koristi uglavnom u aplikacijama gdje se prvenstveno traži velika brzina prijenosa, a ne nužno i pouzdanost, odnosno, u onim aplikacijama koje ne zahtijevaju potvrdu prijema.

UDP služi tome da aplikacije na računalu mogu slati poruke, odnosno, *datagrame*, drugim *hostovima* na mreži. Također, ovaj protokol pretpostavlja da nema potrebe za provjeravanjem ima li grešaka, stoga tako izbjegava procesiranje na mrežnom sučelju. UDP protokol najčešće se koristi u video sastancima, konferencijama, računalnim igricama i sl. [20]. Ovaj se protokol također nalazi i u jednom dijelu u transportnoj razini kod OSI modela, i uz TCP se može smatrati jednim od ključnih Internet protokola. Kao takav, ovaj protokol daje mogućnost slanja kratkih poruka (engl. *datagram*) iz određenih aplikacija s računala koja pritom moraju biti umrežena. Kada se uspoređuje s mrežnom razinom OSI modela, može se reći kako UDP protokol dodaje samo funkcije multipleksiranja i provjere grešaka za vrijeme prijenosa podataka, no ipak nema mogućnost koja se odnosi na provjeru je li poruka zaprimljena, zato što on ne sprema, odnosno ne pohranjuje informacije koje se odnose na stanje veze (odnosno, može se reći da radi na principu da nešto pošalje i onda zaboravi). Upravo zbog prethodno navedene činjenice, on se koristi onda kada je korisniku od veće važnosti učinkovitost te brzina, nego sama pouzdanost, to je primjerice kod situacije kada treba poslati istu poruku na nekoliko različitih odredišta (engl. *multicast*) [21]. Dakle, ovaj protokol zapravo daje nesigurnu uslugu prijenosa paketa, a tu je moguće i ostvariti međusobnu komunikaciju a da se ne uspostavi stalna veza. Sukladno činjenici da UDP nije zadužen za brigu o potencijalnom gubljenju paketa, oni protokoli koji se nalaze višem sloju moraju voditi računa o navedenom. UDP je tako najniža nadogradnja IP protokola, te mu je iz tog razloga zaglavlje nešto pojednostavljenije nego što je to zaglavlje kod TCP protokola. UDP protokolom se prenose određeni višemedijski sadržaji, te su protokoli za prijenos video i audio sadržaja dizajnirani na način da više toleriraju povremeno izgubljene pakete, nego velika kašnjenja u slučaju transmisije paketa, tako da se javlja samo zanemariva degradacija u kvaliteti prijenosa. UDP protokol ne podržava kontrolu toka.

Nadograđeni dio se sastoji od izvorišnih i odredišnih vrata, a ona onda determiniraju postupke pošiljatelja, odnosno, primatelja paketa, duljinu koja se odnosi na broj okteta u okviru cijelog paketa, tu uključuje također i zaglavlje, kao i kontrolnu sumu istog. [21].

4.1. Prednosti i mane UDP protokola

Neke su od temeljnih prednosti UDP protokola tako:

- Nema kašnjenja s ponovnim prijenosom - UDP je prikladan za vremenski osjetljive aplikacije, koje si ne mogu dozvoliti kašnjenja za ponovno slanje eventualno odbačenih paketa. Primjerice, neke su od takvih aplikacija mrežne igrice ili *streaming* mediji.
- Brzina - brzina je UDP protokola izuzetno korisna za protokole koji rade s odgovorima na upit (primjerice DNS protokol).
- Prikladan za *broadcast* - UDP *broadcast* ima mogućnost primiti veliki broj klijenata, bez poslužitelja [21].

S druge strane, mane kod UDP protokola su:

- Ne garantira uručivanje paketa,
- Nema provjere spremnosti računala koje prima poruku,
- Nema zaštite od dupliciranih paketa,
- Nema garancije da će odredište doista zaprimiti sve pakete [21].

4.2. UDP datagrami

UDP se promet odvija putem takozvanih *datagrama*. Slika 5. prikazuje izgled *datagrama*, odnosno zaglavlja, dok će u nastavku biti obrazložena struktura UDP paketa. UDP *portovi* neovisni su od TCP *portova*, te se UDP zaglavlje sastoji od četiri polja. Također, UDP pruža aplikacijsko multipleksiranje, preko kompjuterskog *porta*, te verifikaciju integriteta, preko kontrolne sume zaglavlja i sadržaja koji se njome prenosi.

Ukoliko je poželjna pouzdanost transmisije, onda se navedeno posebno mora implementirati u korisničkoj aplikaciji.

Slika 5. Format UDP datagrama [22]

Izvorišni *port*, odnosno, *source port*, odnosi se na izvorišnu priključnu točku usluge, koja je sama po sebi polje koje je zapravo opcionalno. U situaciji kada se koristi, odnosi se na priključnu točku procesa, koji odašilje podatke. Na nju slijedi odgovor tek u situaciji gdje nema neke druge određene informacije. Ako se ovo polje ne koristi, onda se popunjava oznakom nula.

Odredišni *port*, (engl. *destination port)*, odnosi se na odredišnu priključnu točku usluge.

Duljina paketa, (engl. *length)*, odnosi se na duljinu UDP paketa u oktetima, što uključuje a minimalna duljina UDP *datagrama* iznosi 9 okteta.

Kontrola sume, (engl. *checksum)*, odnosi se na kontrolnu sumu, odnosno zbroj zaglavlja, i može se izračunati temeljem pseudo zaglavlja iz IP i UDP zaglavlja.

Podaci, odnosno *data*, odnosi se na potrebne podatke.

U nekim je situacijama za prenošenje svih poruka preporučljivo koristiti UDP protokol, a neke su od takih situacija sljedeće:

- Prenošenje podataka kod onih aplikacija koje same osiguravaju prijenos koji je pouzdan, odnosno, u situaciji kada aplikacija ipak dozvoljava gubitke u prihvatljivom obujmu,
- Odašiljanje upita od strane jednog računala prema drugom, s mogućnošću ponovnog slanja ako odgovor ipak ne dođe nakon što istekne određeni vremenski interval,
- U situaciji kad treba poslati manji blok podataka, izraženoj u veličini 1 paketa, stoga je lakše i efikasnije prenesti samo podatke, ali bez ostalih kontrola, dok u mogućem slučaju pogrešnog prihvata, podatke treba poslati opet [22].

5. TRANSMISSION CONTROL PROTOCOL (TCP)

Transmission Control Protocol (u daljnjem tekstu: TCP) vrši podjelu podataka u pakete, a njih mreža može efikasno, odnosno učinkovito nositi, na taj način potvrđuje da su oni ondje gdje su trebali stići, te ponovno sklapa podatke. Temelji se na takozvanoj *point-to-point* komunikaciji između 2 mrežna domaćina, te tako prima podatke iz programa i obrađuje iste putem nadolazećih bajtova. Oni su grupirani u segmente, odnosno dijelove, koje TCP onda brojčano označava te sekvencionira za isporuku [23].

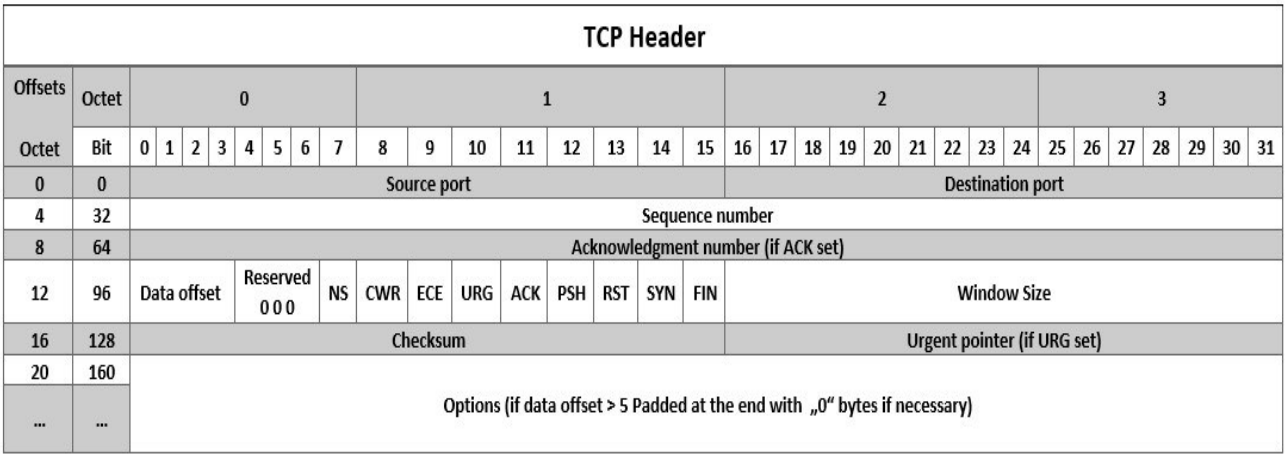
TCP je dominantni prijenosni protokol interneta, i kao takav jamči sigurno isporučivanje podataka od njihovog izvora do mjesta gdje trebaju stići, u redoslijedu koji je kontroliran. Temeljna svojstva usluga koje pruža TCP jesu sljedeće: pouzdan je, pruža sigurnu vezu, dvosmjerno prenosi podatke, a ti su podaci tretirani kao niz okteta [24].

Za vrijeme korištenje TCP usluge njezini entiteti prolaze kroz nekoliko faza, od kojih su sljedeće:

1. Uspostava veze: ova faza odvija se na jednom računalu, a isto nastoji uspostaviti vezu s nekim drugim računalom. Ono računalo što traži uspostavljanje veze se inače naziva klijent, a drugo se naziva poslužitelj.
2. Razmjena podataka: TCP entiteti, kao što je već navedeno, podatke razmjenjuju u formi segmenata, a jedan se segment sastoji od zaglavlja, a ono sadrži 20 okteta, uz opcionalni dio, iza kojeg ide ili nula, ili više okteta podataka, a on nastaje sakupljanjem podataka od nekoliko upisivanja. Veličina segmenta može se mijenjati, ali ima dva ograničenja (svaki segment, što se odnosi i na zaglavlje, mora stati u 65 535 okteta IP paketa, te svaka mreža posjeduje MTU (najveću dopuštenu brzinu prijenosa).
3. Prekid veze: u situaciji kada se klijentska aplikacija ipak odluči na prekid veze sa poslužiteljem, ista šalje TCP segment sa FIN bitom postavljenim u 1 i uđe u FIN_WAIT_1 stanje. Dok je u tom stanju, klijentski TCP čeka TCP segment potvrde od poslužitelja. Kada primi taj segment, klijentski TCP ulazi u FIN_WAIT_2 stanje. Dok je u tom stanju, klijent čeka sljedeći segment od strane poslužitelja s FIN bitom postavljenim u 1. Nakon što primi taj segment klijentski TCP ulazi u TIME_WAIT stanje. Vrijeme provedeno u TIME-WAIT stanju ovisi o implementaciji, ali tipične

vrijednosti su trideset sekundi, jedna minuta i dvije minute. Nakon čekanja veza se formalno zatvori [24].

Procesi podatke šalju pozivom TCP protokola prosljeđujući mu pritom *buffere* podataka kao argumente, te zatim TCP obavlja pakiranje podataka iz tih *buffera* u segmente i onda poziva internet modul (primjerice IP) za prenošenje svih segmenata na odredišni TCP. Slika 6. prikazuje zaglavlje TCP paketa, a u nastavku će biti pojašnjeni najvažniji dijelovi zaglavlja.



Slika 6. Zaglavlje TCP paketa [25]

Najvažniji su dijelovi zaglavlja TCP paketa tako:

- *Source port*- predstavlja identifikatora *porta* pošiljatelja,
- *Destination port*- odnosno, identifikator *porta* primatelja,
- *Sequence number*- broj 1. bajta podataka
- *Acknowledment number*- broj trenutno primljenih bajtova, plus 1,
- *Data offset*- detaljizira veličinu TCP zaglavlja,
- *Window size*- odnosi se na veličinu primljenog paketa
- *Checksum*- odnosno, kontrolno polje.

6. REAL TIME TRANSPORT PROTOCOL (RTP)

Real Time Transport Protocol (u daljnjem tekstu: RTP) po svojoj definiciji jest protokol koji radi na osiguravanju funkcija za mrežno prenošenje podataka, u stvarnom vremenu, a takvi podaci su primjerice video i audio podaci koji se upotrebljavaju sa svrhom simulacije putem *unicast* i *multicast* mrežnih usluga. Također, ovaj protokol određuje i strukturu paketa koji su zaduženi za prijenos audio i zvučnih zapisa, te tako pruža identifikaciju tipa sadržaja, numeraciju sekvenci paketa i vremenske oznake. Iako dizajniran uz visok stupanj neovisnosti od ostalih protokola, RTP se ipak vrlo često upotrebljava u kombinaciji s *User Datagram Protocol-om* (u daljnjem tekstu: UDP) kako bi mogao iskoristiti njegove prednosti i potencijale prilikom multipleksiranja i *checksum* provjere. Nadalje, ovaj protokol ne sadrži mehanizme dostave paketa, odnosno, *multicasting* ni brojeve portova, stoga mu je iz navedenog razloga prijeko potreban UDP kako bi uspješno funkcionirao. Dakle, RTP se javlja u ulozi posrednika između UDP-a i aplikacije.

6.1. Temeljne značajke i arhitektura RTP protokola

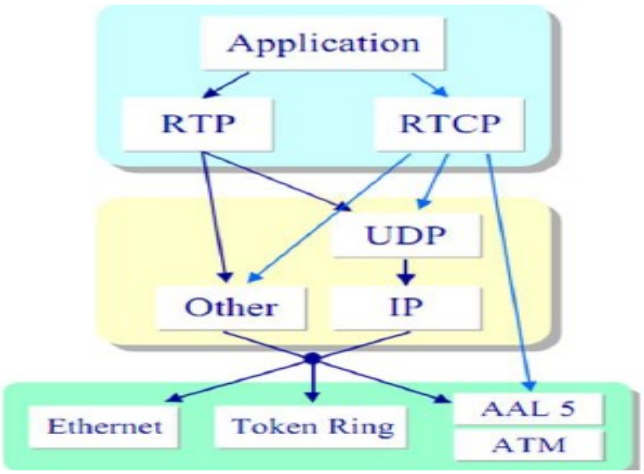
RTP je protokol koji se nalazi unutar mrežnog sloja OSI modela, a ključna mu je zadaća isporučivanje audio sadržaja i video sadržaja preko mreže. Uglavnom se koristi u onim sustavima koji su prvenstveno namijenjeni za komunikaciju i zabavu, a ti se sustavi pritom odnose i na *streaming*-medije (a ti su mediji primjerice telefonija, TV usluge, videokonferencija i sl.). Ovaj protokol određuje i način prema kojem programi upravljaju kod prenošenja multimedijskog sadržaja u stvarnom vremenu putem *unicast* i *multicast* usluga. Također, ne može jamčiti trenutnu dostavu podataka budući da je ovisan o određenim karakteristikama koje mreža ima [26], i uz navedeno, jedan je od temelja VoIP-a.

Nadgledavanje prijenosa osigurava primatelju detekciju gubitaka podataka te nadoknadu nedostataka koji su uzrokovani kašnjenjem. Informacije koje su sadržane u zaglavlju RTP-a omogućavaju primatelju rekonstruiranje istih a uz navedeno sadržavaju i informacije koje navode kako se bitovi kodeka raspadaju, odnosno dijele u pakete. Ako se ovaj protokol promatra prema hijerarhijskom odnosu, može se reći kako se on izvršava iznad UDP protokola [27].

Temeljne su osobine RTP protokola:

- *Timestamp* (svojstvo koje pamti datume i vrijeme, bez informacije o vremenskoj zoni; dozvoljava specifikaciju vremena u milijarditom dijelu sekunde),
- Numeracija sekvenci (broj sekvenci se u ovom slučaju koristi kako bi se moglo otkriti koji su paketi izgubljeni),
- Miješanje *stream-ova* (RTP se dakle primarno oslanja na protokole koji se nalaze ispod njega) [28].

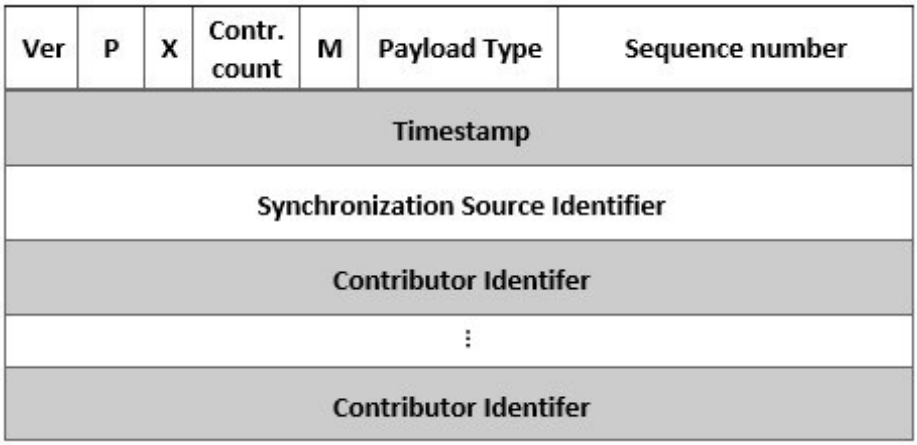
Kao i *Real Time Control Protocol* (u daljnjem tekstu: RTCP), RTP pripada aplikacijskom sloju, samo s razlikom što se RTP spaja na UDP. Pri *online stream-u* važna je brzina samog prijenosa paketa, pri čemu zbog velikih količina paketa nerijetko dolazi do grešaka. Slika 7. prikazuje arhitekturu RTP protokola, a RTP je zapravo vrsta transportnog protokola. Oslanja se na UDP, odnosno na određeni transportni protokol koji provodi multipleksiranje i zaštitnu sumu, stoga ipak nije u potpunosti pouzdan, budući da ne može jamčiti isporuku u realnom vremenu. Može se prilagoditi određenoj aplikaciji pomoću profila te specifikacije formata tipa tereta. Slika 7. prikazuje arhitekturu RTP protokola.



Slika 7. Arhitektura RTP protokola [28]

6.2. Format RTP protokola

Format RTP *datagrama* jednostavan je i dostatan kako bi pokrio sve *real-time* aplikacije koje danas postoje. Ona aplikacija koja zahtijeva dodatne informacije, dodaje ih na početku korisnih podataka u paketu, odnosno, *payload-a*. Slika 8. daje prikaz formata RTP protokola.



Slika 8. Format RTP protokola

Kao što prethodna slika prikazuje, RTP protokol sastoji se od nekoliko podataka, a u radu će biti dan prikaz svakog od njih. Dakle,

- Ver se odnosi na bitove koji definiraju verziju RTP-a,
- P se odnosi na jedan bit, koji daje podatak o tome je li upotrijebljen *padding*, odnosno dopunjavanje paketa,
- X, odnosno *extension* odnosi se na jedan bit, ukazuje na dodatno zaglavlje između standardnog i *data* polja.
- *Contributor count* jest 4-bitno polje, koje definira broj izvora, dok je njihov maksimalan broj 15,
- M, odnosno *marker* je jedan bit, kojega koriste neke aplikacije za primjerice, oznaku kraja *stream-a*,
- *Payload type* odnosi se na 7-bitnu informaciju koja definira tip podataka,
- *Seq.Number*- 16 bita; ovo se polje koristi za numeraciju RTP paketa, budući da UDP ne numerira svoje pakete, a prijemnik navedeno polje koristi kako bi identificirao broj izgubljenih paketa,
- *Timestamp*- 32-bitno polje, koje definira vremenski iznos između paketa,
- *Synchronization Source Identifier* (SSRC), 32 bita- ukoliko postoji samo jedan izvor, ovaj ga broj određuje,

- *Contributor Identifer* (CSRC)- 32 bita; svaki od navedenih identifikatora definira jedan izvor, a maksimalno ih može biti 15 [28].

6.3. Usluge RTP-a

RTP protokol nudi više usluga, kao što su to:

- Identificiranje vrste tereta (*payload type*) – ovu vrstu usluge RTP protokola karakterizira činjenica kako se navedenim protokolom može obavljati prijenos različitih vrsta podataka, te različitih formata,
- Numeracija paketa unutar niza – moguće je otkriti gubitak, odnosno potencijalno pogrešan redoslijed paketa,
- Vremenska oznaka (*timestamp*) – pruža vremensku informaciju, te takva informacija omogućava sinkroniziranje, kao i prikaz realnih vremenskih podataka, te,
- Nadzor isporučivanja paketa korištenjem RTCP-a [29].

Ključnih značajki ovog protokola je nekoliko, a one su primjerice identifikacija opterećenja, budući da je u sklopu svakog RTP paketa sadržan i jedinstveni identifikacijski broj opterećenja, kako bi se moglo opisati kodiranje medija. Nadalje, RTP ima i indikator okvira, a tu se misli na podatak da se audio i video ovdje šalje u logičkim jedinicama koje se zovu okviri, a njihov se početak i kraj označava s bitom *frame marker*. Također, u RTP se paketu broj sekvenci koristi kako bi se moglo otkriti koji su paketi izgubljeni, a također je potrebna i identifikacija izvora, kao bi se moglo odrediti koji su pokretači okvira. Konačno, kako bi se mogla nadoknaditi razna odgađanja zagušenja kod paketa, u jednom ili istom prijenosu, ovaj protokol nudi vremenske oznake [30].

7. REAL TIME CONTROL PROTOCOL (RTCP)

Real Time Control Protocol (u daljnjem tekstu: RTCP) protokol je namijenjen kontroliranju prenošenja podataka. Svoje funkcije obavlja surađujući sa RTP protokolom, a temeljna mu je uloga zapravo pružanje povratne informacije o kvaliteti usluga RTP protokola. Ovaj protokol ima funkcije koje vrše sinkronizaciju među različitim vrstama medija, a aplikacijama zaduženima za prijenos, daje potrebne informacije o nivou kvalitete mreže, broju gledatelja i sl. [31].

RTCP protokol se bazira na periodičnoj transmisiji kontrolnih paketa svim sudionicima sesije, a koristi identične mehanizme distribuiranja, koji su korišteni i kod prenošenja podataka. Protokol koji se nalazi u sloju ispod, mora osigurati multipleksiranje kontrolnih i podatkovnih paketa [10]. Surađuje sa RTP protokolom u isporuci i pakiranju multimedijskih podataka, no sam ne prenosi nikakve medijske podatke.

7.1. Funkcije RTCP protokola

RTP se protokol uglavnom šalje na parni UDP priključak, sa parnim brojem, dok se RTCP poruke šalju preko sljedećeg višeg, ali neparnog *porta*. RCTP protokol sam po sebi ne pruža sve metode šifriranja, odnosno provjere autentičnosti. Pruža temeljne funkcije za koje se očekuje da će biti implementirane u svim RTP sesijama, a to su:

- Prikupljanje statistike o aspektima kvalitete distribucije medija za vrijeme sesije, te prijenos tih istih podataka izvoru medija i ostalim sudionicima sesije. Takve informacije zatim izvor može koristiti za kodiranje medija te otkrivanje nastalih grešaka u prijenosu.
- RTCP također pruža izvorne identifikatore krajnje točke (CNAME) svim sudionicima sesije. Unatoč tome što se očekuje da će identifikator izvora RTP toka biti jedinstven, trenutno vezanje identifikatora izvora na krajnje točke se može promijeniti tijekom sesije.
- RTCP protokol je prikladno sredstvo za dosezanje svih sudionika sesije, dok sam RTP protokol nije, budući da on prenosi samo medijski izvor.

- Kako bi se izbjegla mrežna zagušenja, protokol mora uključivati upravljanje propusnošću sesije, što se postiže dinamičkom kontrolom učestalosti prijenosa izvještaja, te korištenje RTCP propusnosti ne smije prijeći 5% ukupne propusnosti sesije [32].

7.2. Usluge RTCP protokola

Kako je navedeno, RTCP je protokol koji vrši nadzor prenošenja podataka na velikim *multicast* mrežama. Korišten je prvenstveno u VoIP-u, videokonferencijama i u *streaming* medijima.

Neki od podataka koje RTCP može prenositi jesu količine bajtova koje se odnose na prijenos, količina izgubljenih te poslanih paketa te kružno kašnjenje (engl. *round trip delay)* među krajnjim točkama. RTCP protokol koristi 5 različitih tipova paketa, odnosno poruka, za potrebe prenošenja informacija, a oni su sljedeći:

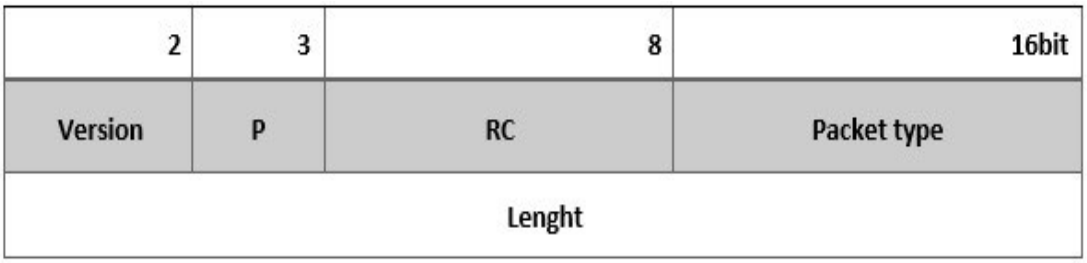
- RR (*Receive report*)- koristi se za svojevrsnu statistiku prijema od onih sudionika su neaktivni pošiljatelji
- SR (*sender report*)- koristi se za dobivanje statistike prijenosa,
- SDES- odnosi se na podatke koji opisuju izvor,
- BYE- odnosi se na završetak sudjelovanja u prijenosu,
- APP- funkcije koje su specifične za aplikaciju [33].

Neke od dodatnih usluga koje RTCP pruža sudionicima jesu:

- Identifikacija: neke od informacija kao što su primjerice adresa e-pošte, telefonskog broja ili pak imena, uključene su u pakete RTCP protokola, stoga su svi korisnici u relativno lakoj mogućnosti saznati identitet ostalih korisnika u istoj sesiji.
- QoS povratna informacija: kako je navedeno u prethodnom poglavlju, RTCP se upotrebljava za prijavu kvalitete usluge, stoga takve informacije sadržavaju broj koji se odnosi na izgubljene pakete i sl., dok izvori takve informacije upotrebljavaju kako bi mogli prilagoditi brzinu prijenosa podataka.
- Međumedijska sinkronizacija: unatoč činjenici da se audio i video uglavnom šalju putem raznih *streamova*, svakako je potrebno sinkronizirati ih na prijamniku, kako bi se mogli skupa reproducirati.
- Kontrola sesije: pomoć paketa BYE, koji označava kraj sesije za sudionike [33].

7.3. Zaglavlje RTCP protokola

Slika 9. prikazuje strukturu RTCP protokola.



Slika 9. Struktura RTCP protokola

U nastavku će biti dan prikaz svakog dijela strukure, stoga:

- *Version*- odnosi se na identifikaciju RTP protokola, koja je ista u RTCP paketima kao i u RTP paketima podataka. U ovom su prikazu dane specifikacije 2 verzije,
- P- odnosi se na punjenje. Kada postoji P, RTCP paket sadržava neke dodatne oktete s punjenjem na kraju, a koji nisu dio kontrolnih informacija. Posljednji oktet punjenja broji koliko okteta s punjenjem treba zanemariti. Može biti potreban pojedinim algoritmima za šifriranje sa fiksnim veličinama paketa.
- RC- odnosi se na broj izvješća o prijemu, odnosno broj blokova izvješća o prijemu koji su sadržani u paketu.
- *Lenght* - odnosno duljina, odnosi se na duljinu RTCP paketa, izraženu u 32-bitnim riječima, minus jedan, što uključuje zaglavlje i sve razmake [34].

8. WIRESHARK

Wireshark je jedan od vodećih svjetskih mrežnih analizatora protokola, koji ima vrlo široku primjenu u praksi. Navedeni program omogućava da se i na najdetaljnijoj razini može uočiti što se događa na mreži, te je kao takav postao standard u mnogim različitim ustanovama, kao što su to primjerice, komercijalna ili neprofitna poduzeća, razne vlade agencije, pa čak i u obrazovnim institucijama diljem svijeta. Ima bogat set značajki, koji se sastoji od sljedećeg:

- detaljan uvid u više od stotinu protokola, a učestalo ih se dodaje sve više
- izvanmrežna analiza kao i snimanje uživo
- više platformi, pa tako radi na gotovo svim platformama koje postoje (npr. Windows, Linux, macOS i dr.)
- VoIP analiza
- Podaci se mogu čitati uživo putem mreže, Wi-Fi-ja, PPP / HDLC, ATM, USB, *Bluetooth*, FDDI, *Token Ring, Frame Relay* i drugih
- Pruža podršku koja je namijenjena dekodiranju većine protokola, uključujući primjerice WEP i WPA / WPA2, IPsec, ISAKMP, Kerberos i dr.
- Snimljeni podaci mogu se spremiti u gotovo svim popularnim oblicima [35].

Wireshark se primarno koristi kako bi se moglo analizirati mrežne pakete. Ovdje se konkretno radi o alatu koji obuhvaća podatke koji se mrežom prenose u paketima te ih iskazuje na najprecizniji i najopsežniji mogući način. Primjerice, ovaj se alat koristi kako bi bilo moguće ukloniti potencijalne probleme mreže (ukoliko se pojave), kako bi se moglo analizirati sigurnosne propuste i ranjivosti, kako bi se moglo ugraditi ili razvijati neke nove protokole i sl. [36].

8.1. Način rada i osnovne mogućnosti alata Wireshark

Navedeni se alat koristi uglavnom za hvatanje, filtriranje, kao i uvoz i izvoz paketa. Ima grafičko sučelje, što omogućuje jednostavno rukovanje alatom. Također, Wireshark alat

podržava različite protokole, kao što su primjerice TCP/IP protokoli, te može očitati pakete s više vrsta mreža.

Ovaj program je zapravo programski alat a on, jednostavno rečeno, „razumije“ način funkcioniranja kao i strukturu raznih mrežnih protokola. Stoga, ovaj alat može ukazati na podatke iz različitih paketa, koji su individualni za različite protokole. Wireshark rezultate mrežne analize sprema u datoteku koda .pcap (engl. *Packet capture*) za hvatanje paketa, što znači da on može hvatati samo one pakete s mreža koje .pcap prepoznaje. Takvi se podaci mogu direktno uhvatiti s aktivne mrežne veze, odnosno, mogu se učitati iz datoteke gdje su spremljeni paketi koji su već uhvaćeni.

Snimljene je podatke moguće analizirati putem grafičkog korisničkog sučelja ili preko terminala kod upotrebe Tsharka, a isti se mogu programski uređivati bilo preko terminala, bilo uz pomoć potprograma naziva „*editcap*“. Također, Wireshark sadržava filter za prikaz podataka a pomoću njega se može prikazati i samo određeni dio podataka, što ovisi o načinu filtriranja. Prema činjenici da je Wireshark alat otvorenog koda, može se reći kako je relativno lako ugraditi programske dodatke za neke nove protokole [36].

8.2. Wireshark i sigurnost mreže

Wireshark alatom moguće je identificirati pojedine sigurnosne propuste i nepravilnosti. Sigurnosni propusti očituju se u malicioznim radnjama koje mogu uzrokovati štetu mreži, kao i njezinim korisnicima. Kao takav, ovaj alat radi na sprječavanju sigurnosnih propusta, izvršavajući analizu potencijalnih problema i radnji koje potencijalno mogu napraviti probleme. Zadaci analize u sklopu Wiresharka mogu se podijeliti na one prije nego li se greške uoče (preventivne) i na one koje nakon uočavanja greške (reaktivne). Zadaci prije uočavanja greške pritom sadržavaju „*baselining*“ mrežne metode (a to je najjednostavniji način za analizu mrežnih performansi) za očitavanje trenutnog statusa mreže i aplikacije. Dakle, preventivne metode omogućavaju da se uoči gubitak paketa prije nego li taj gubitak počne ikako utjecati na mrežnu komunikaciju i na taj se način izbjegava problem prije nego je uočen od strane samog korisnika. Reaktivne metode koriste se nakon što su greške uočene. Primjerice, ukoliko postoji problem s nekim poslužiteljem, Wireshark će problem prijaviti tek nakon što pokuša prvo sam uhvatiti pakete s mreže. No ipak, u Wiresharku su još uvijek reaktivne analize zastupljenije od

preventivnih što je relativno loše, budući da reaktivne analize uočavaju problem prije nego on može utjecati na mrežu i korisnika, dok kod preventivnih to ipak nije slučaj [36].

Wireshark nudi svojim korisnicima nekoliko vrsta analiza koje im pomažu pri očuvanju sigurnosti i administraciji mreže, a one su sljedeće:

- pronalaženje najaktivnijih korisnika mreže,
- identificiranje aplikacija koje se trenutno upotrebljavaju,
- popis svih korisnika na promatranoj mreži,
- vrste i duljine paketa korištenih od strane aplikacija za prijenos podataka na mreži,
- uočavanje najučestalijih problema (opterećena/loša mreža, nemogućnost identifikacije korisnika i sl.),
- uočavanje kašnjenja kod zahtjeva za prijenos paketa i samog prijenosa paketa,
- uočavanje korisnika s nepravilnim postavkama računa i nedopuštenim pristupom (npr. duplicirana IP adresa),
- uočavanje sumnjivog prometa na mreži,
- efikasna identifikacija grešaka,
- grafički prikaz prometa koji se odvija na mreži,
- grafički prikaz prometa u aplikaciji i komparacija s cjelokupnim prometom mreže,
- uočavanje onih protokola koji nisu uobičajeni i dr. [36].

9. JITSI PROGRAM

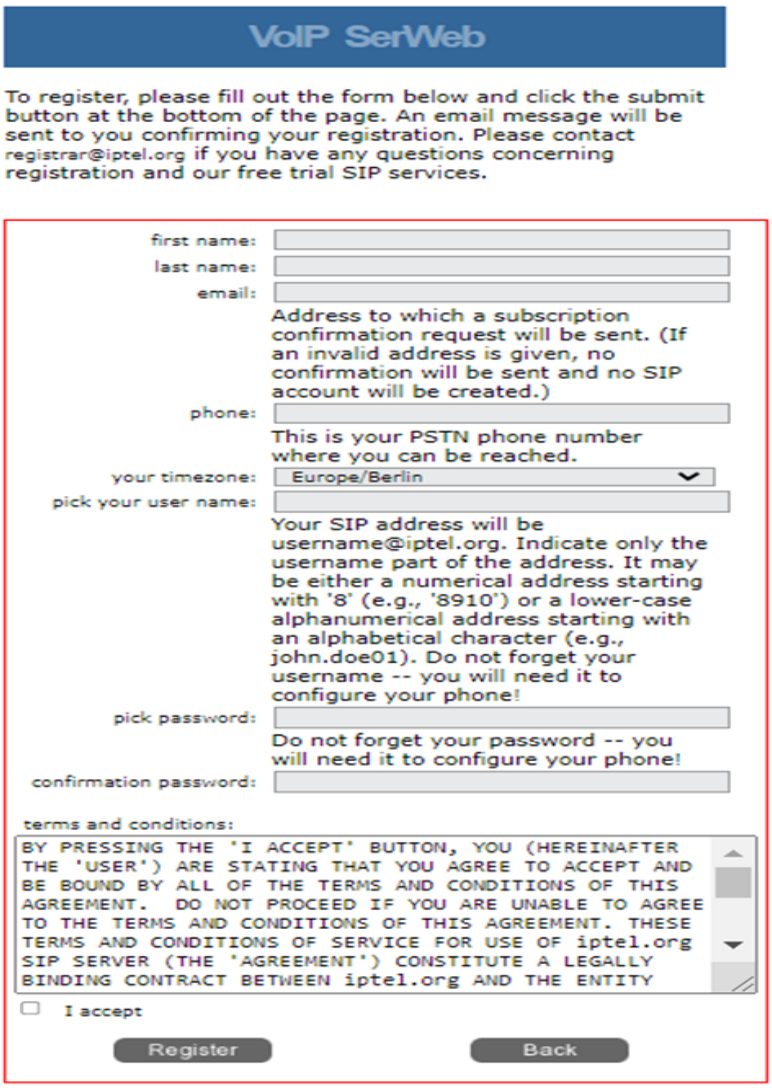
Jitsi program je zapravo zbirka besplatnih aplikacija otvorenog koda na više platformi (VoIP), nudi videokonferencije kao i mogućnost razmjene trenutnih poruka na web platformama kao što su to primjerice Windows, Linux, Android i dr. Razvoj Jitsija započinje nastankom Jitsi Desktopa, koji je prvenstveno bio poznat kao SIP *Communicator*. Sve ubrzanijim razvojem poziva i video-poziva preko *weba*, fokus projektnog tima Jitsija usmjerio se na Jitsi *Videobridge*, za omogućavanje video-poziva s više strana na *webu*. Nešto kasnije, dolazi do osnutka Jitsi Meet, aplikacije za video konferencije, kojoj su mogli pristupiti klijenti sa platformi kao što su Android i iOS. Jitsi je dobio podršku od različitih institucija, kao što su primjerice Sveučilište u Strassbourgu ili Europska komisija.

Prvi projekt Jitsi tima nastao je 2003. godine, a zvao se Jitsi Desktop, poznat kao SIP *Communicator*, odnosno samo kao Jitsi. Ova aplikacija se odnosi na razmjenu trenutnih poruka, podržava više operativnih sustava kao što su Linux, Windows i dr. Mobilna aplikacija može se preuzeti na *App Storeu* i na *Google Play Storeu* [38]. Neke od usluga koje Jitsi Desktop nudi su tako:

- prijenos poziva,
- automatsko ponovno povezivanje,
- automatski odgovor i automatsko prosljeđivanje,
- snimanje poziva,
- šifriranje poziva,
- konferencijski pozivi,
- *streaming* na radnoj površini,
- šifrirana pohrana lozinke pomoću glavne lozinke,
- prikrivanje gubitka paketa kodecima i dr. [38].

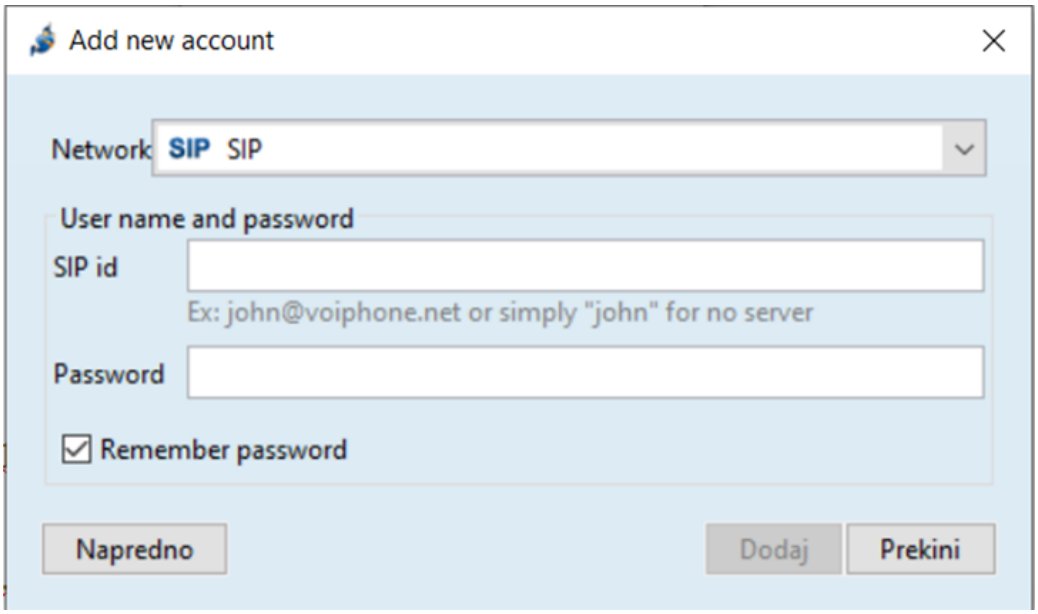
S druge strane, Jitsi Videobridge rješenje je za video konferencije, koje podržava Web RTC, koji omoućuje video komunikaciju između više korisnika. To je jedinica za selektivno prosljeđivanje i prosljeđuje samo odabrane *streamove* drugim korisnicima koji sudjeluju u video konferencijskom pozivu, stoga snaga procesora nije toliko ključna za dobre performanse.

Kako bi bilo moguće koristiti Jitsi, treba prvo podesiti SIP račun, a navedeni račun moguće je napraviti na SIP registru http://serweb.iptel.org/user/reg/index.php, te on mora biti u obliku xyz@iptel. org. Slika 10. prikazuje obrazac za SIP registraciju.



Slika 10. Obrazac za SIP registraciju [39]

Nakon kreiranja SIP računa, potrebno je otvoriti aplikaciju Jitsi. U postavkama Jitsija *File* → *Add new account* potrebno je odabrati protokol SIP. Potom na sljedećem dijalogu treba unijeti korisničko ime (SIP) i lozinku, te odabrati opciju Dodaj. Slika 11. prikazuje kako izgleda proces dodavanja računa u Jitsi aplikaciju.



Slika 11. Dodavanje računa u Jitsi aplikaciju

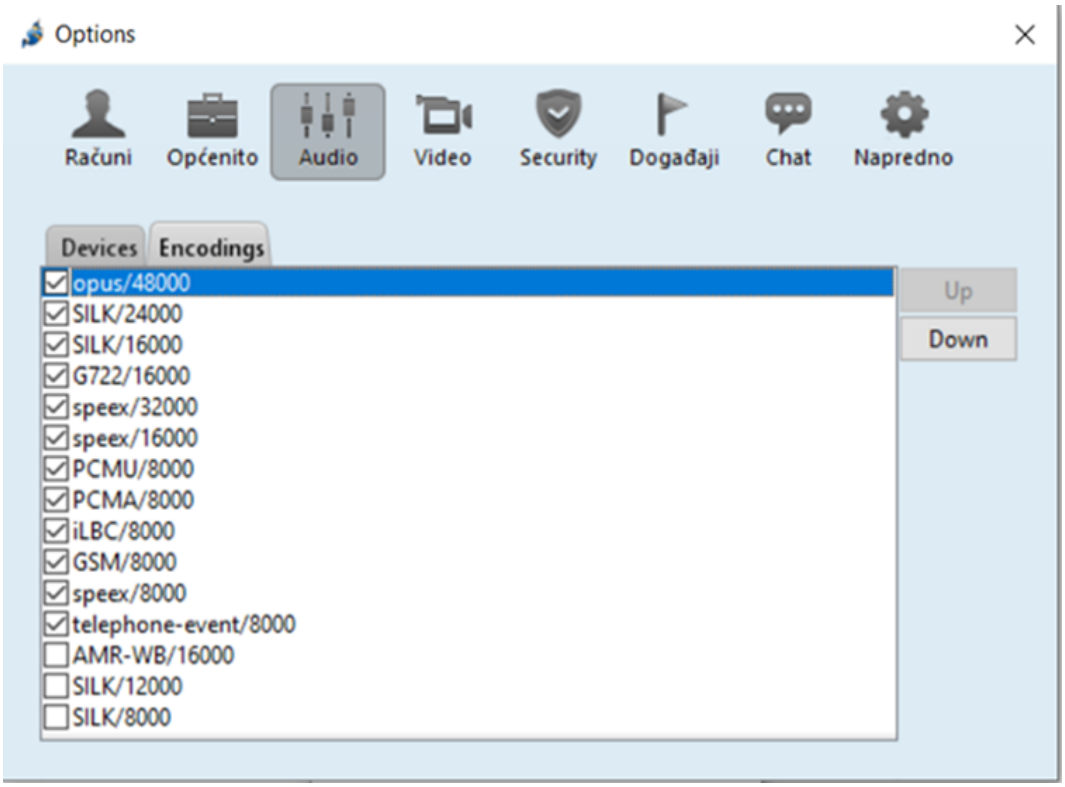
10. ANALIZA SJEDNICE IZMEĐU DVA KORISNIKA

Analizu prometa vrši se pomoću programa Wireshark. U ovom će se radu analizirati:

1. audio-sjednica između dva korisnika i
2. video-sjednica između dva korisnika i
3. video-sjednica između tri korisnika

10.1. Analiza audio sjednice između dva korisnika

Cilj je ovog poglavlja uspostava sjednice između dva korisnika i potom izvršiti analizu iste. Pretpostavit će se da korisnik A poziva korisnika B. U opcijama programa Jitsi za audio i video (*Tools → Options → Video → Encoding i Tools → Options →Audio → Encoding*) može se vidjeti koji su kodeci uključeni za vrijeme audio poziva, što prikazuje Slika 12.

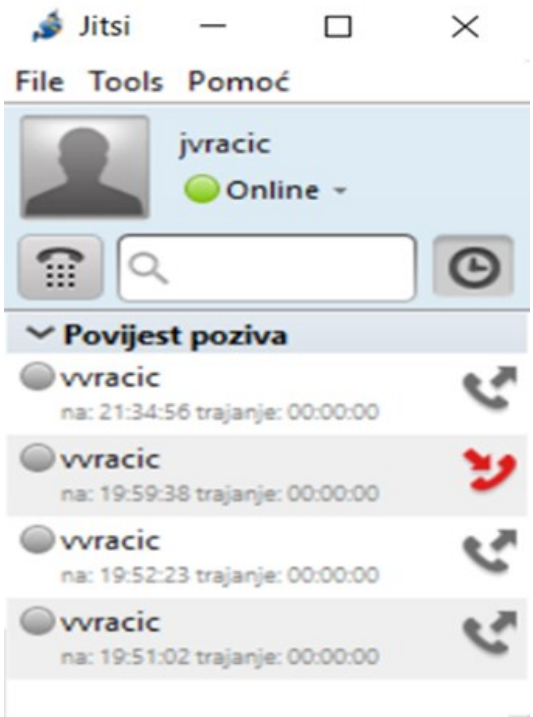


Slika 12. Uključeni kodeci tijekom audio poziva

Nakon toga pokreće se mrežni analizator Wireshark i počinje se sa snimjem prometa koji se odvija na mrežnom sučelju (*Capture →Options → Input → Start*). Zatim treba pokrenuti Jitsi i uspostaviti audio sjednicu s drugim korisnikom, upotrebljavajući pritom njegovu SIP adresu.

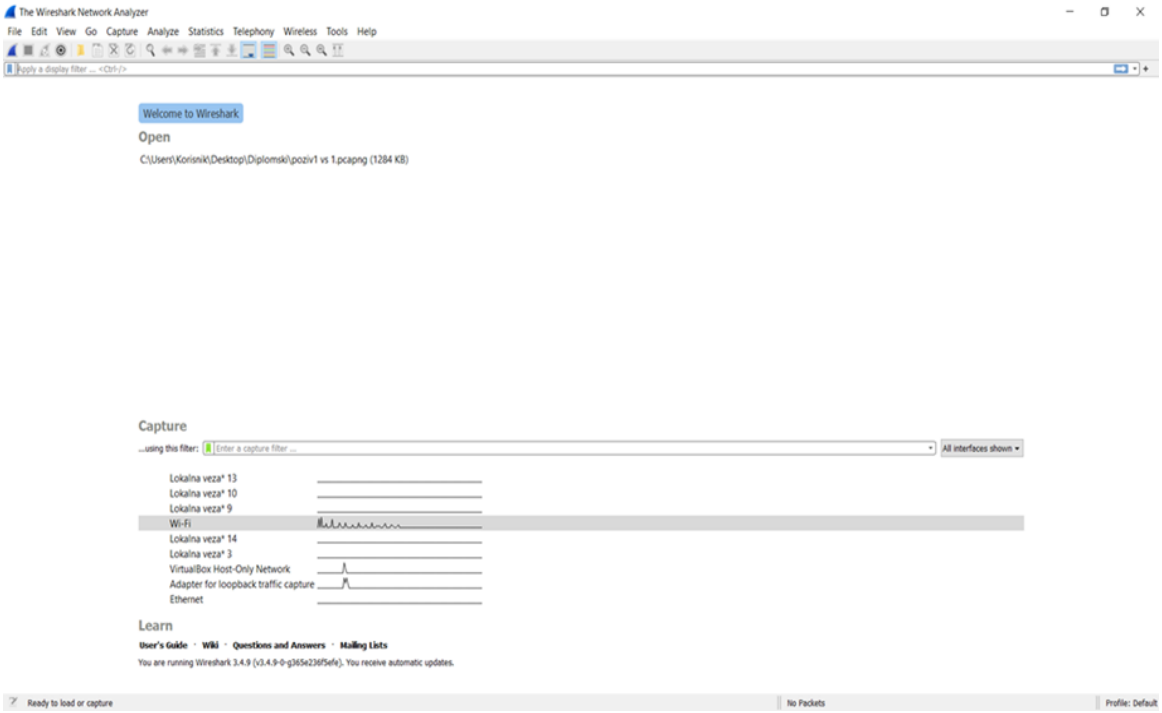
Slika 13. prikazuje glavni izbornik aplikacije Jitsi. Prije svega, treba napomenuti kako su paketi uhvaćeni Wiresharkom umjesto oznake RTP označeni sa „*Opus*“. Dakle, *Opus* je potpuno otvoreni i visoko svestrani audio kodek, koji nema nikakve licence. Standardizirao ga je *Internet Engineering Task Force* (IETF) kao RFC 6716. Značajke koje podržava *Opus* jesu:

- brzina prijenosa od 6 kb/s do 510 kb/s
- brzina uzorkovanja od 8 kHz do 48 kHz
- veličine okvira od 2,5 ms do 60 ms
- podrška za konstantnu brzinu bita (CBR) te promjenjiva brzina prijenosa (VBR)
- podrška za govor i glazbu i dr.



Slika 13. Glavni izbornik aplikacije Jitsi

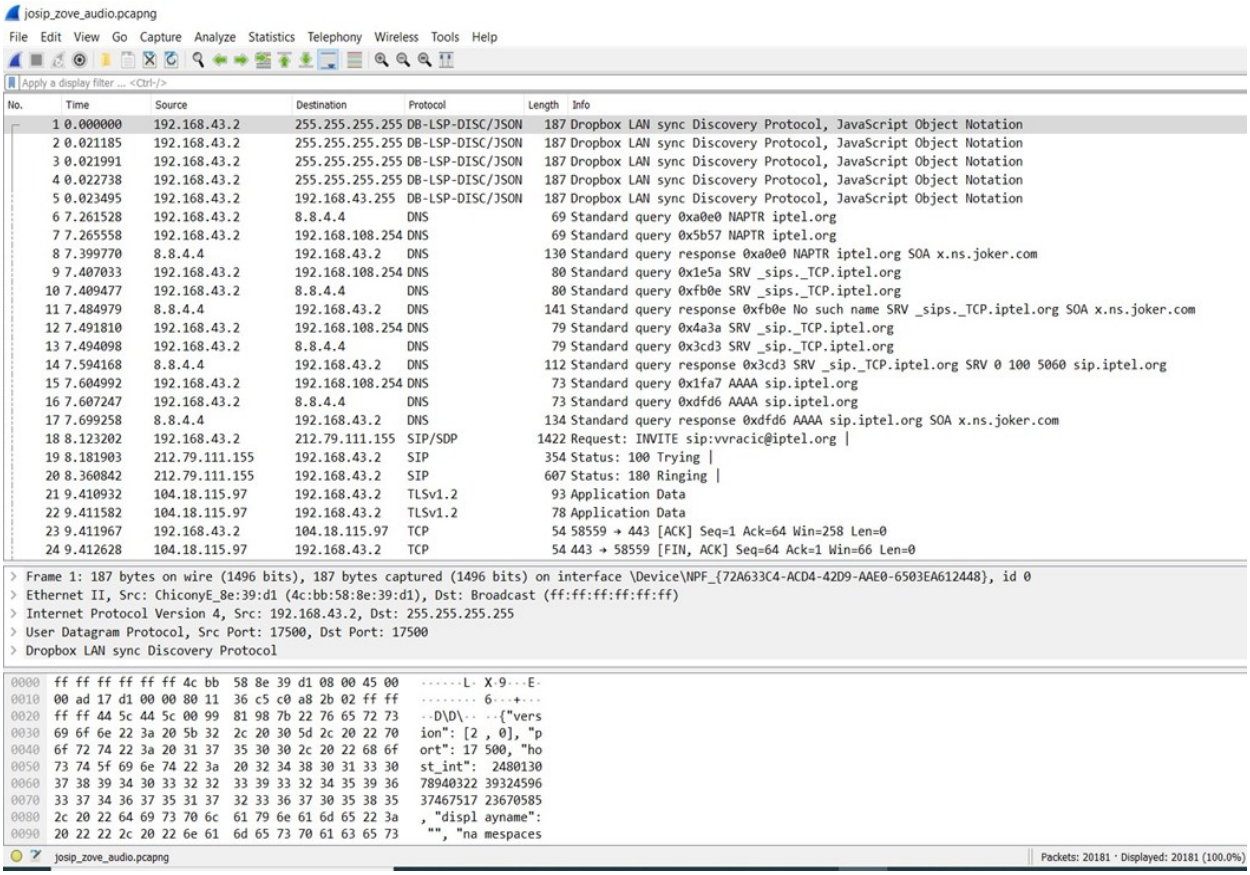
Slika 14. prikazuje kako izgleda, odnosno od čega se sastoji prikaz mrežnih sučelja u Wiresharku.



Slika 14. Wireshark prikaz mrežnih sučelja

Dakle, pokrene se Wireshark, i počne se snimati promet na mrežnom sučelju (*Capture* → *Options* → *Input* → *Start* ili *Ctrl+E*). Nakon toga potrebno je pokrenuti aplikaciju Jitsi, te pozvati drugog korisnika. Po završetku razgovora vrati se u program Wireshark te odabere Capture → Stop (Ctrl+K). Snimljeni promet spremljen je u datoteku sa nastavkom .pcapng.

Nastavak pcapng je skraćenica od PCAP *Next Generation Dump File Format*. Temeljna je značajka ovakvog formata što promet koji je snimljen sa više sučelja, može biti pohranjen u jednoj datoteci. Ovakav je format standardni format za pohranu snimljenih podataka, te svaka pcapng. datoteka sadrži nekoliko blokova podataka koji sadrže različite informacije o snimljenom prometu. Slika 15. daje prikaz snimljenih paketa u programu Wireshark.



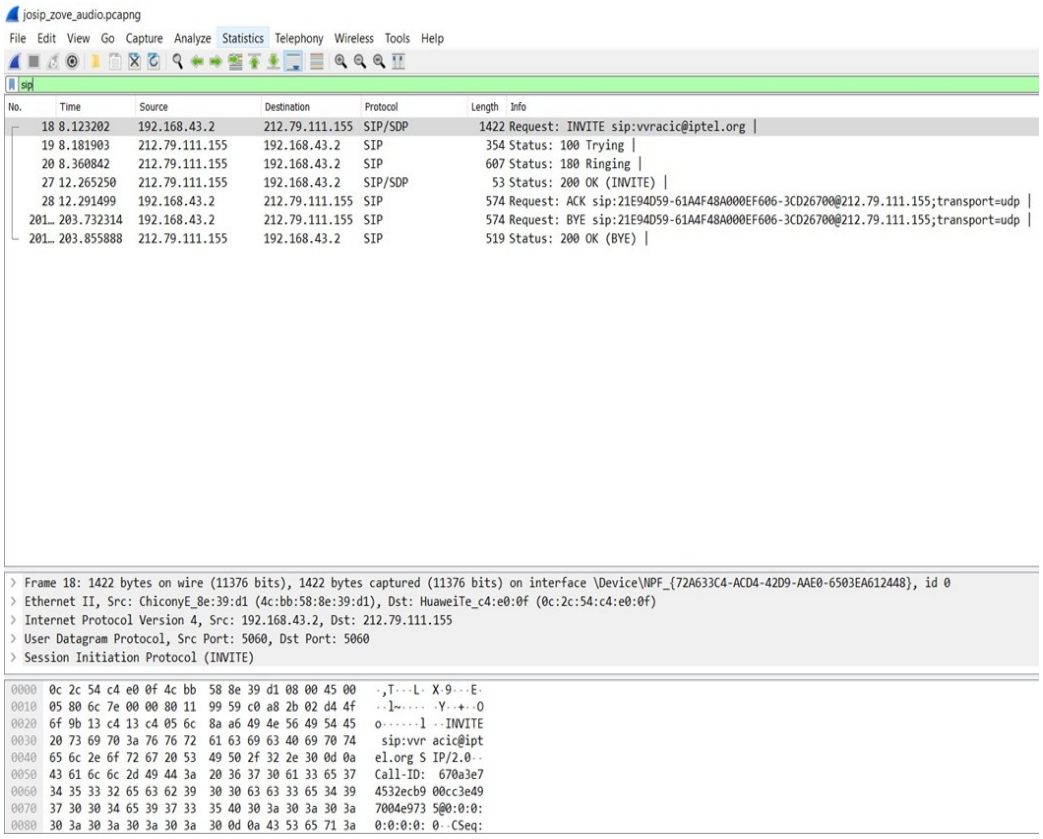
Slika 15. Prikaz snimljenih paketa u Wireshark progamu

Iz slike je vidljivo kako snimljeni podaci nisu nužno dio obavljenog razgovora. Pokretanjem Wiresharka, on automatski snima sav promet na mreži, a ne samo onaj koji se želi analizirati, stoga je potrebno filtrirati rezultate jednostavnim unošenjem imena protokola koji će se analizirati. Slika 16. prikazuje rezultat filtriranja SIP protokola u Wireshark programu.

Potrebno je analizirati od čega se sastoji prikaz snimljenih paketa Wireshark programa. Dakle, na vrhu zaslona, u prvom redu prikazano je okno popisa paketa. Svaki se pojedini komad razgrađuje na broj s vremenom, izvorom, odredištem, kao i protokolom te informacijama o podršci. Nadalje, pojedinosti o odabranom paketu se nalaze u sredini prikaza, pri čemu pokazuju protokole izabranog paketa. Svaki se odjeljak može dodatno proširiti klikom na strelicu kraj izabranog retka. Mogu se koristiti i dodatni filtri klikom desne tipke miša na izabranu stavku.

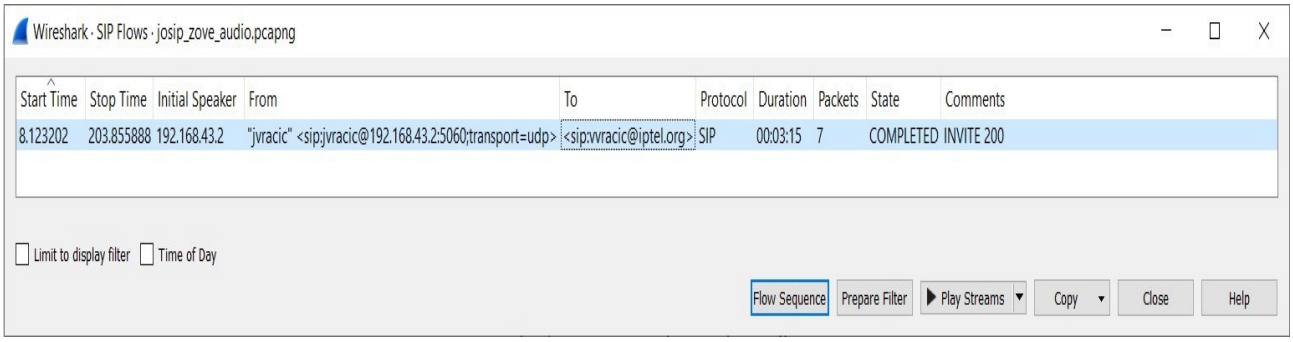
Zatim, na dnu stranice nalazi se prikaz okna bajtova paketa. Ono prikazuje interne podatke izabranog paketa. Ukoliko bi se u ovom odjeljku istaknulo samo dio podataka, oni bi se također isticali u oknu s pojedinostima o paketu. Prema podacima, svi se prikazuju u šesnaestom

formatu. Ukoliko bi se htjelo promjeniti u bitni format, desnim klikom miša trebalo bi kliknuti na okno i izabrati spomenutu opciju.



Slika 16. Rezultat filtriranja SIP protokola

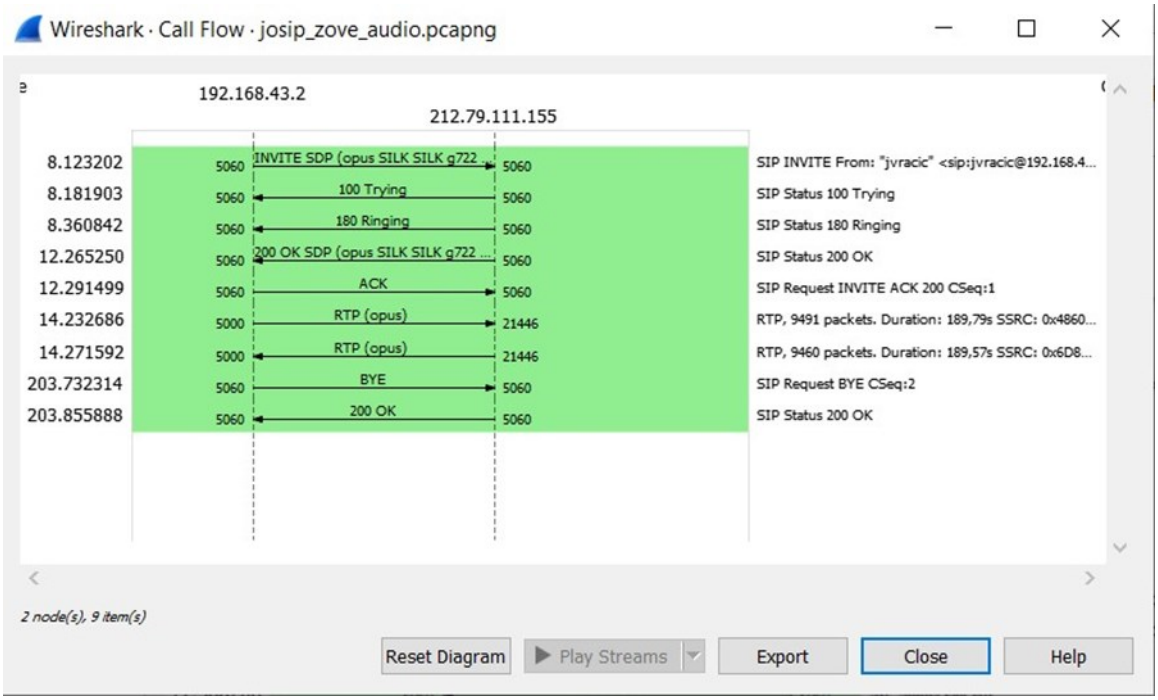
Slika 17. prikazuje odabir željenog SIP streama.



Slika 17. Odabir željenog SIP streama

U nastavku će biti dan prikaz kako izgleda SIP *Flow Graph*, odnosno, analizirat će se upućeni poziv prema korisniku. Dakle, kao što Slika 18. prikazuje, poziv započinje sa porukom

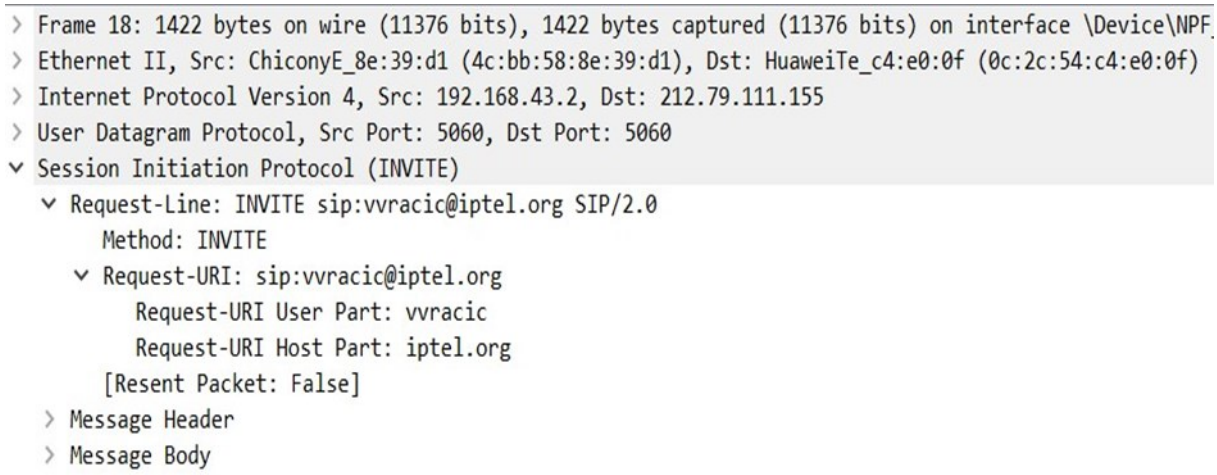
INVITE, skupa sa SDP-om (engl. *Session Description Protocol*). Navedeno se odnosi na zahtjev koji upućuje pozivatelj pozvaniku, a pozvanik zatim odgovara porukom označenom sa 180 *Ringing*, dok se prije odgovora uglavnom odašilje 100 *Trying*, s ciljem naznake da je poruka zaprimljena, te s ciljem sprječavana pozivatelja da opet šalje istu poruku. Pozvanik zatim daje odgovor 200 OK (što znači da će prihvatiti poziv), a može ga i odbiti zbog greške. Zatim pošiljatelj šalje ACK (a to je metoda koja daje potvrdu da je zaprimljen konačni odgovor na *INVITE* zahtjev). Kad sip:vvracic@iptel.org prihvati zahtjev za uspostavom sesije, pozivatelj (sip: jvracic@iptel.org) šalje ACK te se nakon toga uspostavlja se RTP sesija. Ista sesija je gotova onda kada jedna od ove dvije strane pošalje zahtjev *BYE*, a taj zahtjev ukazuje ne prekid dijaloga i poziv je onda završen. Poslije navedenog zahtjeva druga strana šalje odgovor 200 OK, te takva, zadnja poruka zapravo daje potvrdu primanja *BYE* zahtjeva i označava da je poziv završen.



Slika 18. SIP Flow Graph

Samo INVITE i 200 OK poruke sadržavaju SDP opis sjednice, gdje je opisana sjednica, uz medije koji će se prenositi te kodeci koji će se koristiti. Tako INVITE poruka sugerira, odnosno predlaže video ili audio kodeke, a s 200 OK potvrđuju se formati koje pozvani korisnik podržava. Poruka sadrži zahtjev/odziv, zaglavlje poruke te tijelo poruke. U zahtjevu ili odzivu

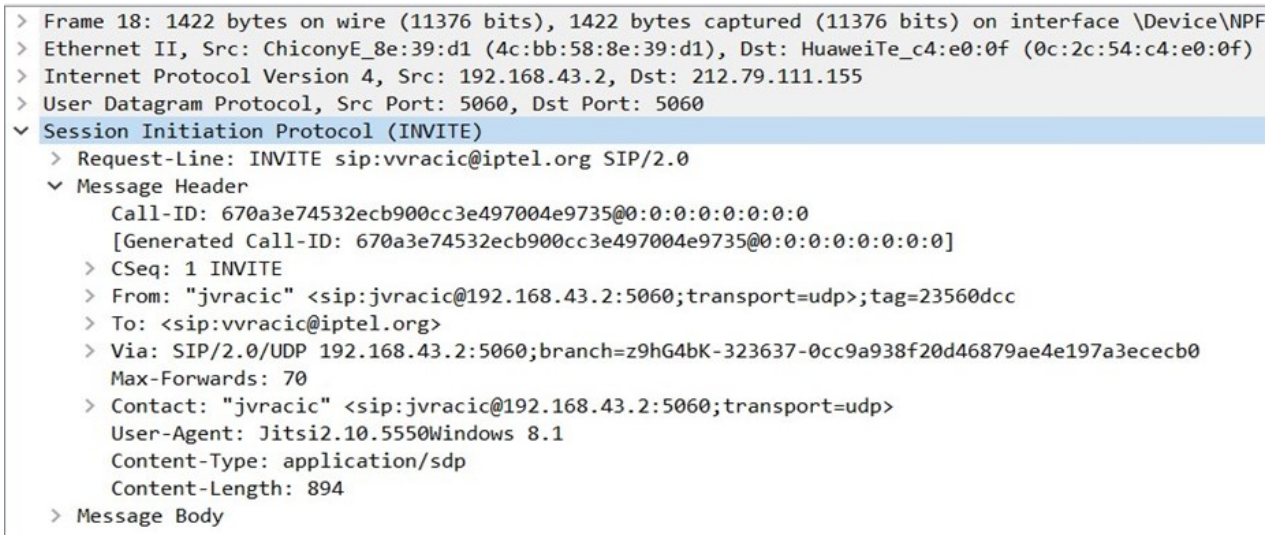
definirana je vrsta SIP poruke, a te su poruke primjerice INVITE, BYE, NOTIFY, OPTIONS i sl. Također, na taj je način označeno ime i IP adresa korisnika i port, što prikazuje Slika 19.



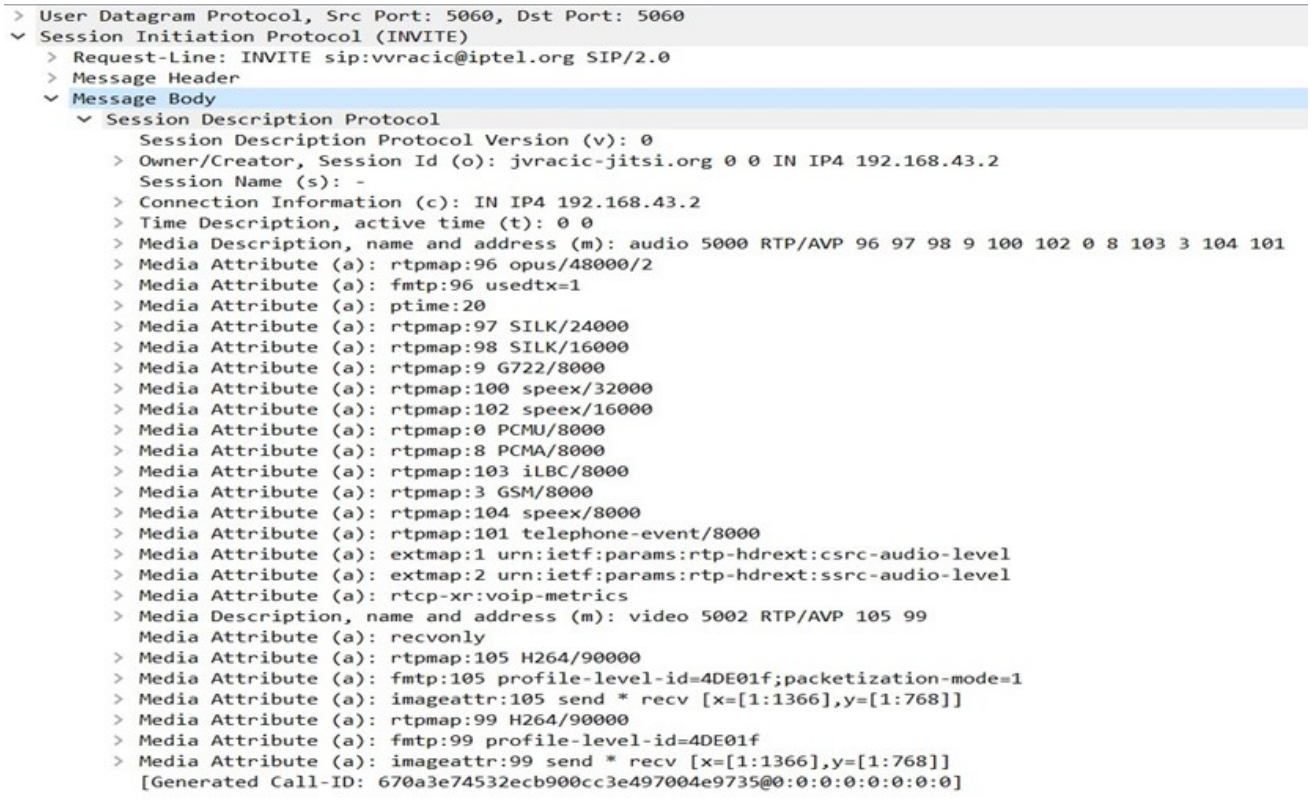
Slika 19. Zahtjev/odziv SIP poruke

Dakle, u zaglavlju se poruke nalaze određeni podaci koje prikazuje Slika 20., a ti se podaci odnose na rute koje poruka mora proći da bi uopće mogla dospjeti do svog odredišta (to je zaglavlje *Via*), zatim su to podaci o izvoru i odredištu poruka (odnosno, to su zaglavlja *From* i *To*); zatim ima unikatni identifikator pozivatelja. Nadalje, sadržani su i podaci o tome koje su vrste poruka dozvoljene unutar sjednice (to prikazuje zaglavlje *Allows*) te veličina zaglavlja kao i ostali podaci koji se odnose na samu sjednicu.

Slika 21. prikazuje tijelo SIP poruke, koje sadrži SDP opis sjednice, koje sadrži niz podataka koji su zapisani pomoću atributa kojima se dodjeljuje određena vrijednost, odnosno format SDP opisa je: atribut = vrijednost. Svaki pojedini SDP opis sjedice nužno sadrži informacije o tome koja se verzija protokola SDP koristi, o inicijatoru i identifikatoru te o tome kako se sjednica zove. Poslije toga, slijede opcionalni podaci o samoj sjednici, a onda opis medija koji su preneseni u sjednici. Tip medija se definira uz pomoć atributa „m“, a ostali podaci uz pomoć atributa „a“. Dakle, u ovom se primjeru prvo vide podaci o audiu, odnosno, informacije o transportnoj adresi i vrsti mogućih kodeka. Nakon toga, slijede precizniji podaci o svakom kodeku, kao što su primjerice frekvencija uzrokovanja signala i sl.

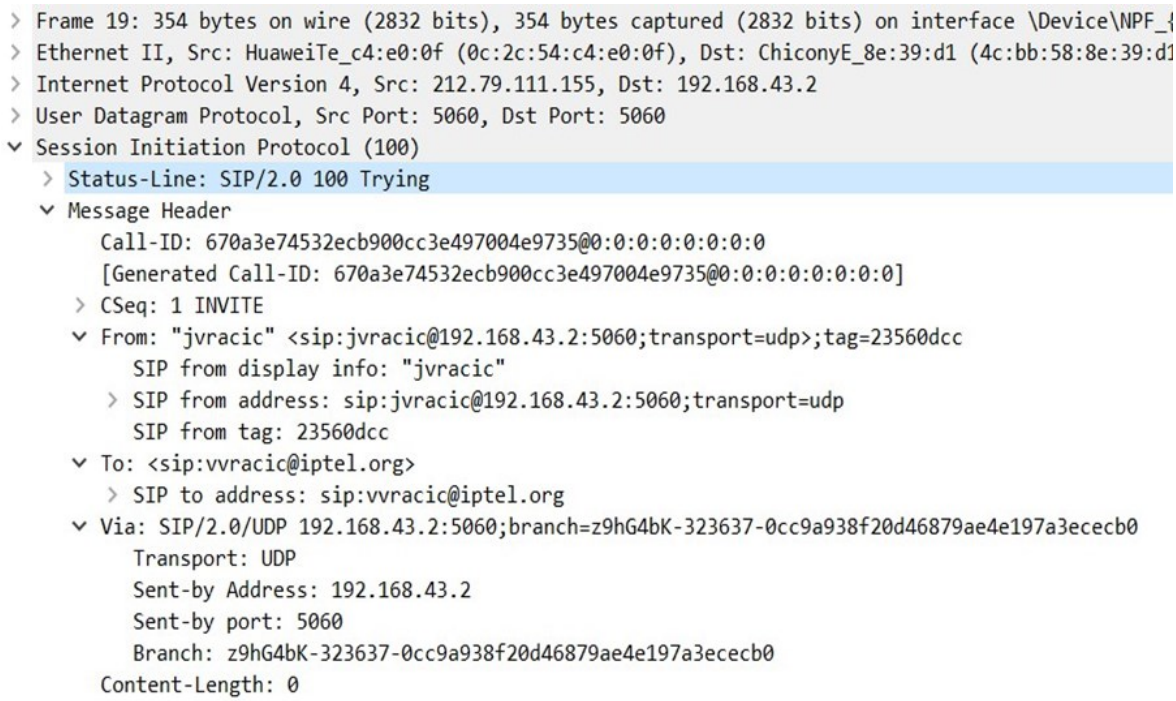


Slika 20. Zaglavlje SIP poruke



Slika 21. Tijelo poruke sa SDP opisom sjednice

Slika 22. detaljno prikazuje Trying metodu, koja daje pošiljatelju do znanja kako je poruka zaprimljena, te koja za cilj ima spriječiti pošiljatelja da opet pošalje poruku.

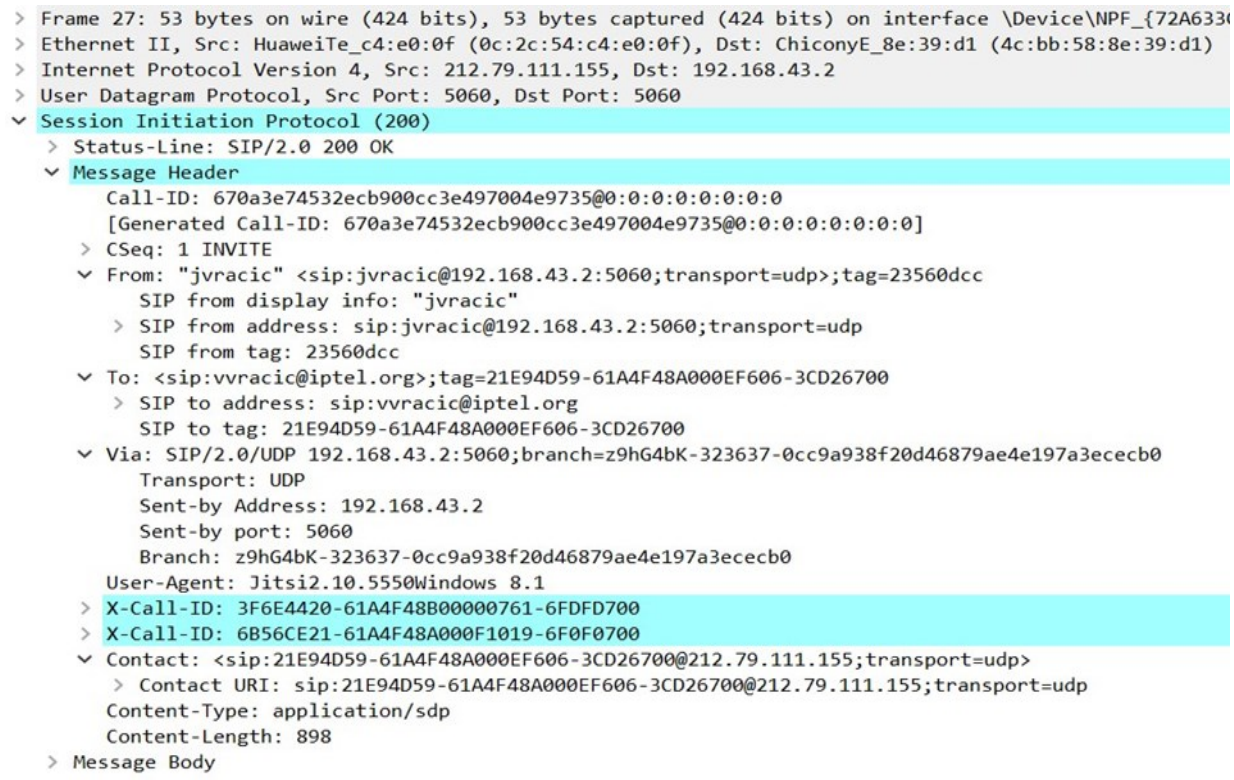


Slika 22. Metoda Trying

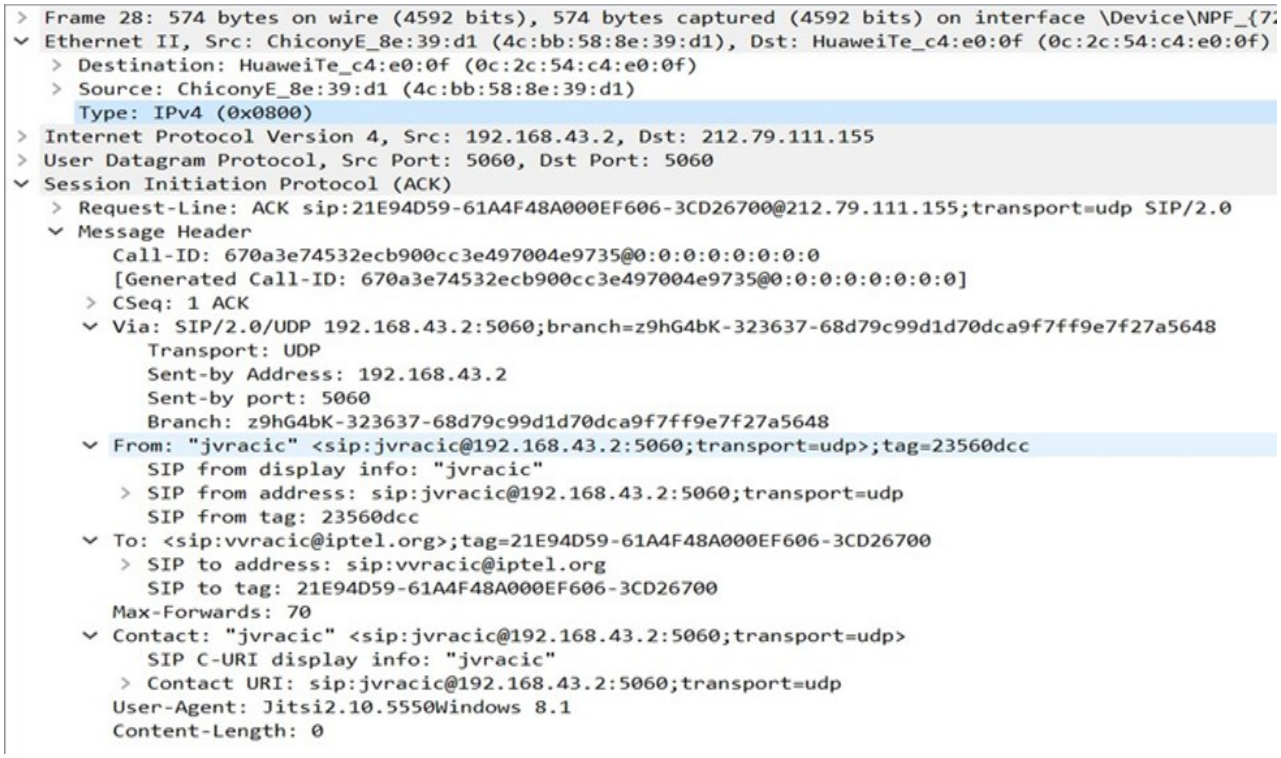
Proxy server nakon toga prima *INVITE* zahtjev od sip adrese: jvracic@iptel.org i vrši pretragu SIP URI zahtjeva (sip: vvracic@iptel.org) kako bi mogao locirati primatelja poziva. DNS po bazi podataka traži lokaciju i onda *INVITE* zahtjev šalje na primateljevu IP adresu. 180 *Ringing* onda pristiže na *proxy* server, a ondje se na temelju identifikatora transakcije (branch=z9hG4bK-323637-0cc9a938f20d46879ae4e197a3ececb0), šalje prema identifikatoru transakcije, odnosno pozivatelju (sip: jvracic@iptel.org), Slika 23.

Slika 23. Metoda Ringing

Kada sip: vvracic@iptel.org poziv prihvati, onda šalje odgovor 200 OK. Odgovor se prvo šalje na *proxy* server, koji zatim poruku prosljeđuje prema sip: jvracic@iptel.org. Zato što se u sklopu poruke 200 OK u *Contact* zaglavlju nalazi SIP URI adresa sip: vvracic@iptel.org, nastavak komuniciranja između pošiljatelja i primatelja poziva odvija se izravno među njima, a prisustvo *proxy* servera izostaje. Slika 24. prikazuje navedeno. SIP: jvracic@iptel.org šalje ACK na adresu sip: vvracic@iptel.org. Poslije ACK odgovora, dolazi do uspostave medijske sesije među članovima sjednice. Slika 25. prikazuje ACK metodu.

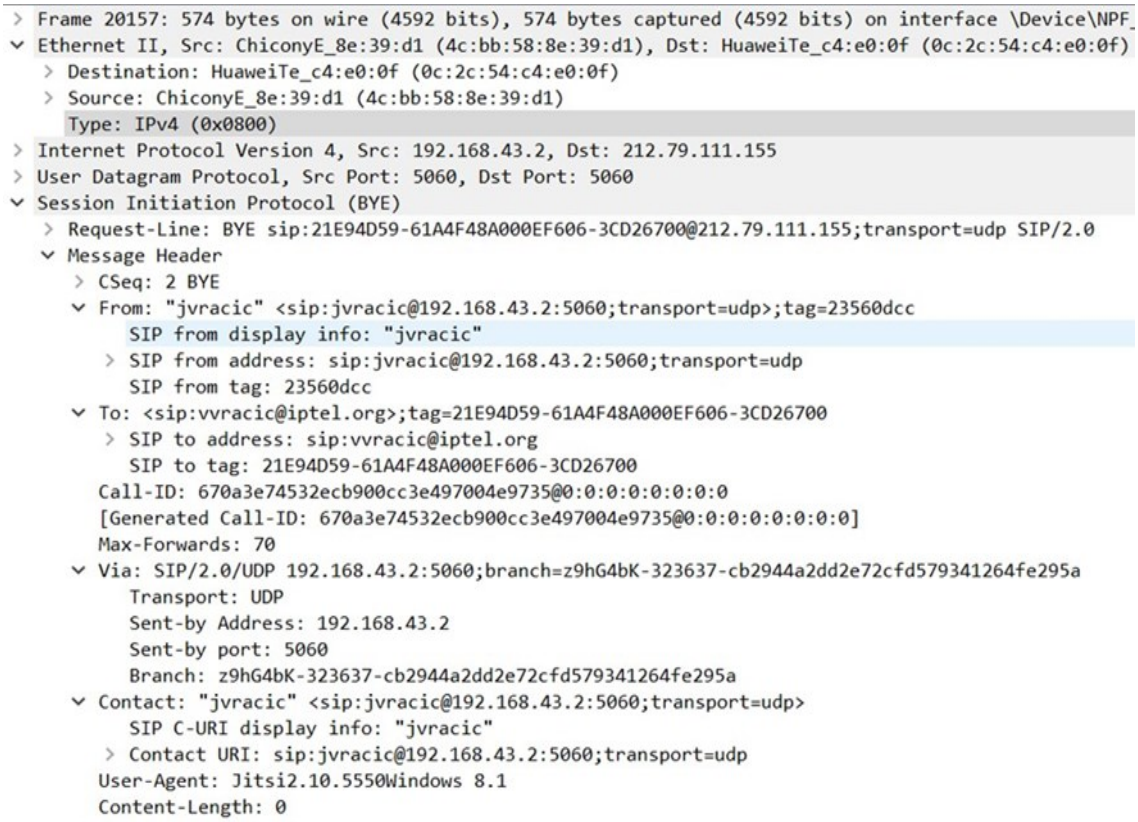


Slika 24. Metoda OK



Slika 25. Metoda ACK

Medijska sesija završava onda kad jedna strana komunikacije pošalje drugoj *BYE* poruku. U ovom slučaju, na *BYE* poruku odgovor šalje jvracic na vvracic *Contact URI* (sip: vvracic@iptel.org). Vvracic onda odgovara s 200 OK, i tako dolazi do prestanka komunikacije. Slika 26. prikazuje metodu *BYE*.



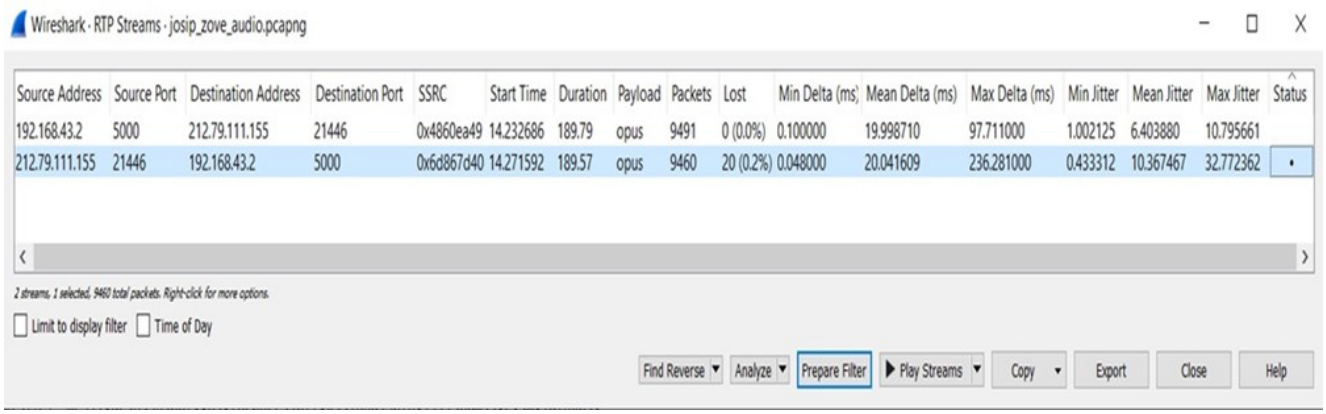
Slika 26. Metoda BYE

```
> Frame 20171: 519 bytes on wire (4152 bits), 519 bytes captured (4152 bits) on interface \Device\NPF_
v Ethernet II, Src: HuaweiTe_c4:e0:0f (0c:2c:54:c4:e0:0f), Dst: ChiconyE_8e:39:d1 (4c:bb:58:8e:39:d1)
  > Destination: ChiconyE_8e:39:d1 (4c:bb:58:8e:39:d1)
  > Source: HuaweiTe_c4:e0:0f (0c:2c:54:c4:e0:0f)
    Type: IPv4 (0x0800)
> Internet Protocol Version 4, Src: 212.79.111.155, Dst: 192.168.43.2
> User Datagram Protocol, Src Port: 5060, Dst Port: 5060
v Session Initiation Protocol (200)
  > Status-Line: SIP/2.0 200 OK
  v Message Header
    > CSeq: 2 BYE
    v From: "jvracic" <sip:jvracic@192.168.43.2:5060;transport=udp>;tag=23560dcc
        SIP from display info: "jvracic"
      > SIP from address: sip:jvracic@192.168.43.2:5060;transport=udp
        SIP from tag: 23560dcc
    v To: <sip:vvracic@iptel.org>;tag=21E94D59-61A4F48A000EF606-3CD26700
      > SIP to address: sip:vvracic@iptel.org
        SIP to tag: 21E94D59-61A4F48A000EF606-3CD26700
      Call-ID: 670a3e74532ecb900cc3e497004e9735@0:0:0:0:0:0:0:0
      [Generated Call-ID: 670a3e74532ecb900cc3e497004e9735@0:0:0:0:0:0:0:0]
    v Via: SIP/2.0/UDP 192.168.43.2:5060;branch=z9hG4bK-323637-cb2944a2dd2e72cfd579341264fe295a
        Transport: UDP
        Sent-by Address: 192.168.43.2
        Sent-by port: 5060
        Branch: z9hG4bK-323637-cb2944a2dd2e72cfd579341264fe295a
      User-Agent: Jitsi2.10.5550Windows 8.1
    > X-Call-ID: 3F6E4420-61A4F48B00000761-6FDFD700
    > X-Call-ID: 6B56CE21-61A4F48A000F1019-6F0F0700
      Content-Length: 0
```

Slika 27. Metoda OK

10.2. Analiza RTP paketa u audio sjednici

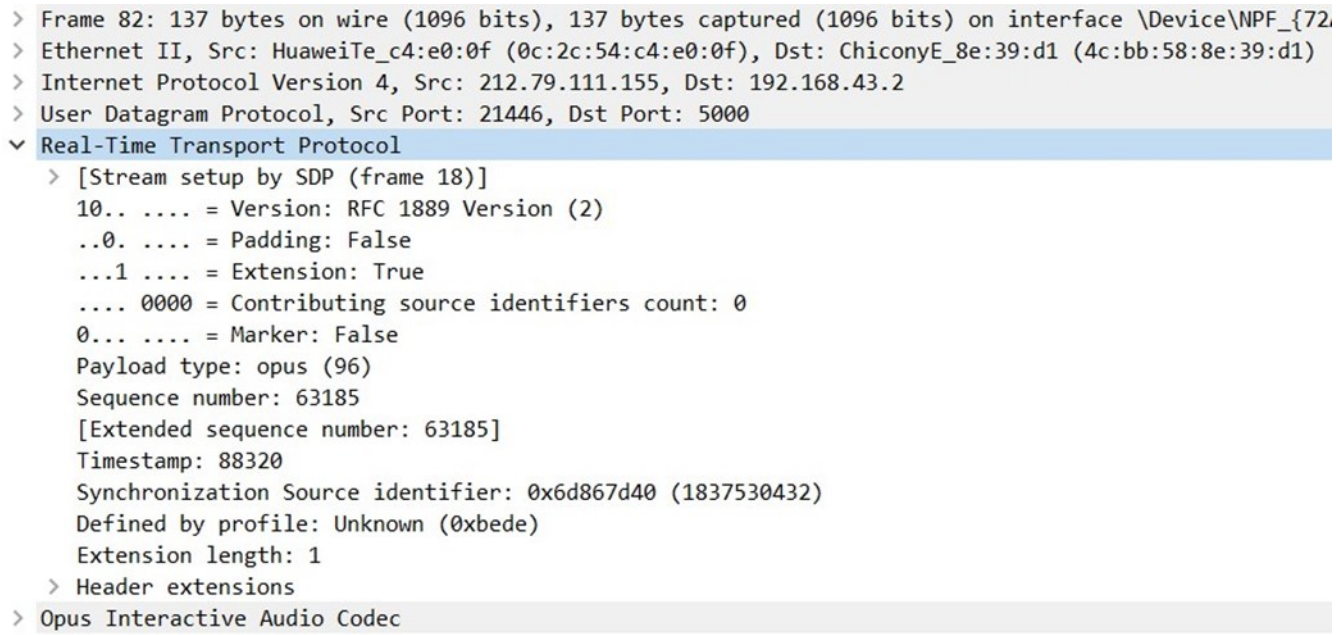
Odabirom *Telephony* → RTP → RTP streams u programskom alatu Wireshark dobiti će se svi RTP tokove koji su se pojavili u ovoj sjednici. U ovoj je audio sjednici bilo ukupno dva RTP toka, što i prikazuje Slika 28. Jedan tok ide prema SIP posredniku, a drugi prema SIP klijentu.



Slika 28. RTP streamovi u audio pozivu

Slika 29. prikazuje jedan RTP paket, a svi se RTP paketi prenose pomoću transportnog protokola UDP. RTP paket se sastoji od sljedećih zaglavlja:

1. *Version* – odnosi se na verziju RTP protokola koja je korištena
2. *Padding* – označava ima li dodatnih bitova u RTP paketu
3. *Extension* – označava ima li dodatno Extension zaglavlje
4. *Contributing Source Identifiers Count* – označava broj CSRC identifikatora
5. *Marker* – označava važnost postojećih informacija za aplikaciju
6. *Payload Type* – označava vrstu kodeka
7. *Sequence Number* –postojeći broj RTP paketa, koji se može koristiti za dešifriranje audia
8. *Timestamp* – vremenska oznaka za trenutne video/audio podatke u sklopu RTP paketa
9. *Synchronization Source Identifier* – identifikator izvora podataka
10. *Payload* – podaci



Slika 29. RTP paket

U VoIP tehnologijama podrhtavanje (engl. *jitter*) se odnosi na varijaciju kašnjenja u primanju paketa. Ovo kašnjenje utječe na prijenos kvalitete glasa i glasovnih podataka. Slika 30. prikazuje

sažete rezultate za jedan *stream* u smjerovima pošiljatelj (engl. *forward*) i primatelj (engl. *reverse*) za jedan VoIP poziv. Uređaj sa IP adresom 192.168.43.2. je računalo sa kojega je iniciran poziv, a uređaj sa IP adresom 212.79.111.155. je odabrani SIP poslužitelj (iptel. org). Prema Cisco-u, prosječni *jitter* za jednu stranu komunikacije trebao bi biti manji od 30 ms. Sažeti razultati dobiveni Wiresharkom sadrže i druge podatke, kao što su:

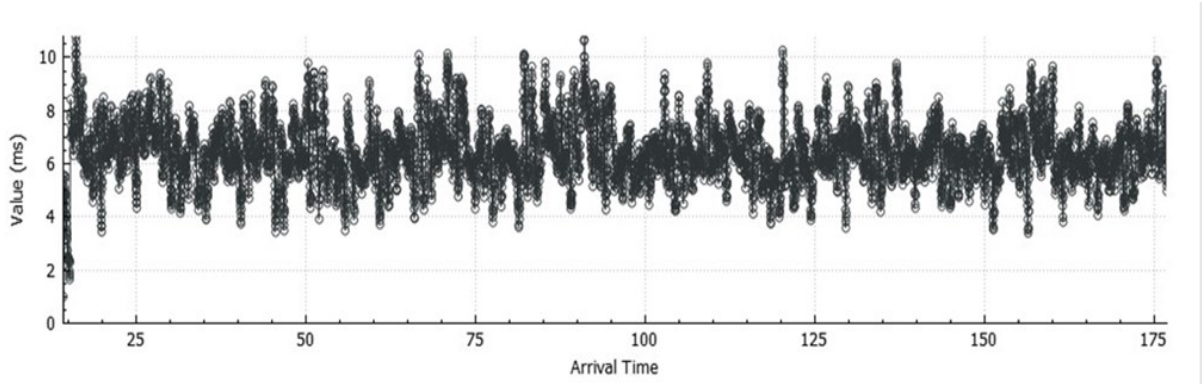
- SSRC- identifikator izvora podataka
- "*Max Delta*" - razlika između vremena primitka trenutnog paketa i vremena primitka paketa prije njega
- "*Max Skew*" - koliko dugo je trenutni paket ispred ili iza cijelog poziva u odnosu na nominalnu brzinu pakiranja
- "*RTP Packets*" pokazuje ukupan broj RTP paketa
- *"Lost "*– broj I postotak izgubljenih paketa
- *"Duration*" trajanje poziva

Stream		Stream	
192.168.43.2:5000 → 212.79.111.155:21446		212.79.111.155:21446 → 192.168.43.2:5000	
SSRC	0x4860ea49	**SSRC**	0x6d867d40
Max Delta	97.711000 ms @ 163	**Max Delta**	236.281000 ms @ 3584
Max Jitter	10.795661 ms	**Max Jitter**	32.772362 ms
Mean Jitter	6.419156 ms	**Mean Jitter**	10.617388 ms
Max Skew	91.132000 ms	**Max Skew**	-222.162000 ms
RTP Packets	8133	**RTP Packets**	8115
Expected	8133	**Expected**	8132
Lost	0 (0.00 %)	**Lost**	17 (0.21 %)
Seq Errs	0	**Seq Errs**	267
Start at	14.232686 s @ 45	**Start at**	14.271592 s @ 52
Duration	162.64 s	**Duration**	162.62 s
Clock Drift	0 ms	**Clock Drift**	0 ms
Freq Drift	0 Hz (0.00 %)	**Freq Drift**	0 Hz (0.00 %)

Slika 30. Sažeti rezultati iz Wiresharka

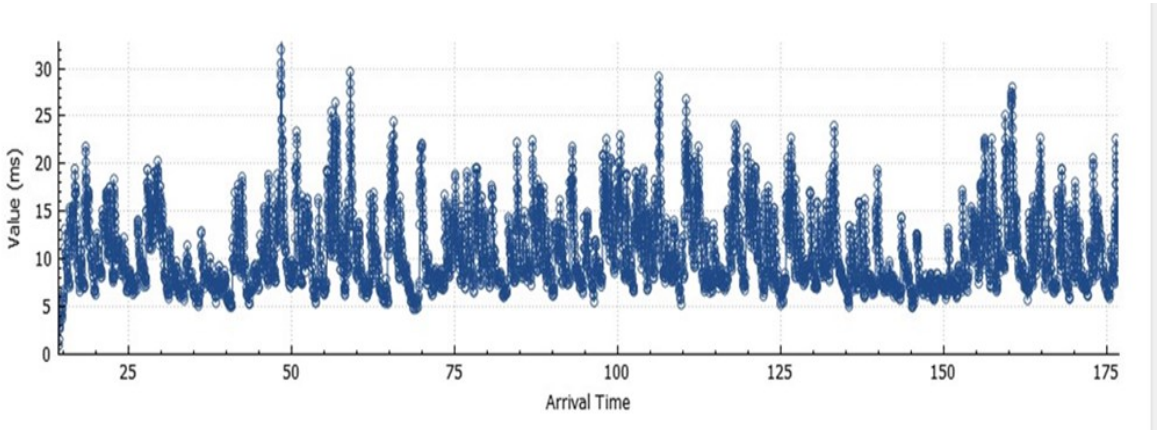
Slika 31. prikazuje *„jitter"* tijekom komunikacije pozivatelj-poslužitelj. U ovom slučaju sa strane pozivatelja nema znatnih odstupanja od nominalnih vrijednosti , jer maksimalna

vrijednost iznosi 10.795661 ms. Komunikacija u kojoj se nalazi ovakav *jitter* odvija se sa minimalnim ili nikakvim gubitkom paketa kao što je i u ovom slučaju.



Slika 31. "Stream jitter" (pozivatelj-SIP poslužitelj)

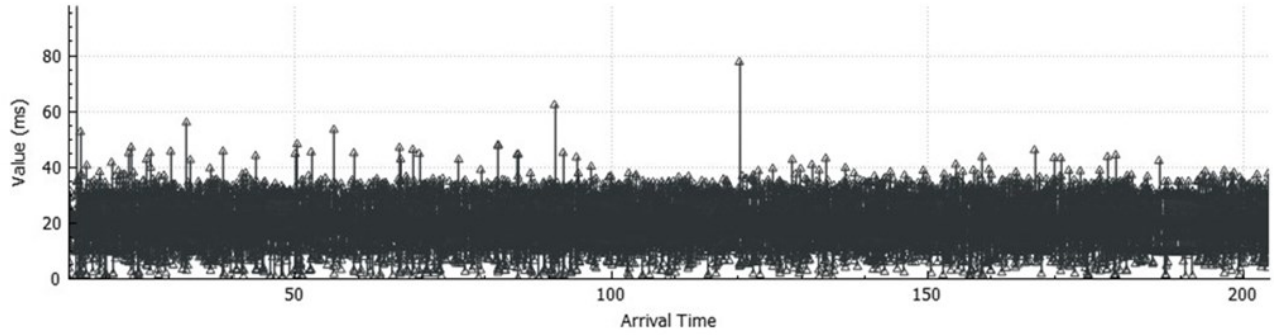
Slika 32. prikazuje *jitter* tijekom komunikacije poslužitelj-pozivatelj, te se može vidjeti vidjeti kako je ovdje vrijednost *jittera* izraženija u odnosu na komunikaciju pozivatelj-poslužitelj, a posljedica prelaska *jittera* preko dozvoljenih vrijednosti dovodi do gubitka paketa i smanjenja kvalitete komunikacije.



Slika 32. "Stream jitter" (SIP poslužitelj-pozivatelj)

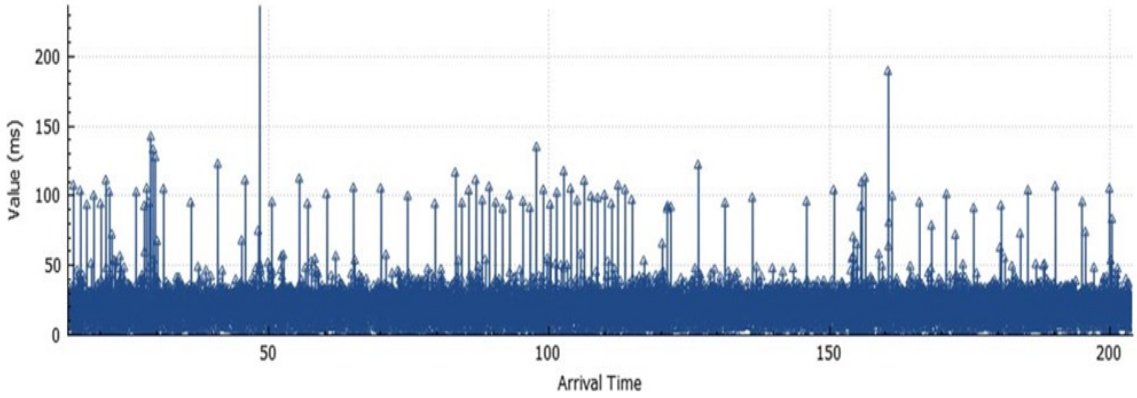
„Stream Delta" ukazuje na vrijeme dolaska između dva paketa ili poznato kao vremensko kašnjenje. Prema zadanim postavkama, gotovo se uvijek podudara sa vremenom paketizacije - 20ms. Zbog prisutnosti *„jitter-a*"na mreži, njezina se vrijednost može povećati i dovesti do gubitka paketa. Jednosmjerno kašnjenje u prijenosu ne bi trebalo premašiti 150 ms. (preporuka G.114), dok maksimalno kašnjenje povratnog signala ne smije prelaziti 300 ms [39].

Slika 33. prikazuje kašnjenja za cijeli razgovor, te je iz iste vidljivo kako nije bilo kašnjenja preko dozvoljenih granica, odnosno kašnjenja preko 150 ms.



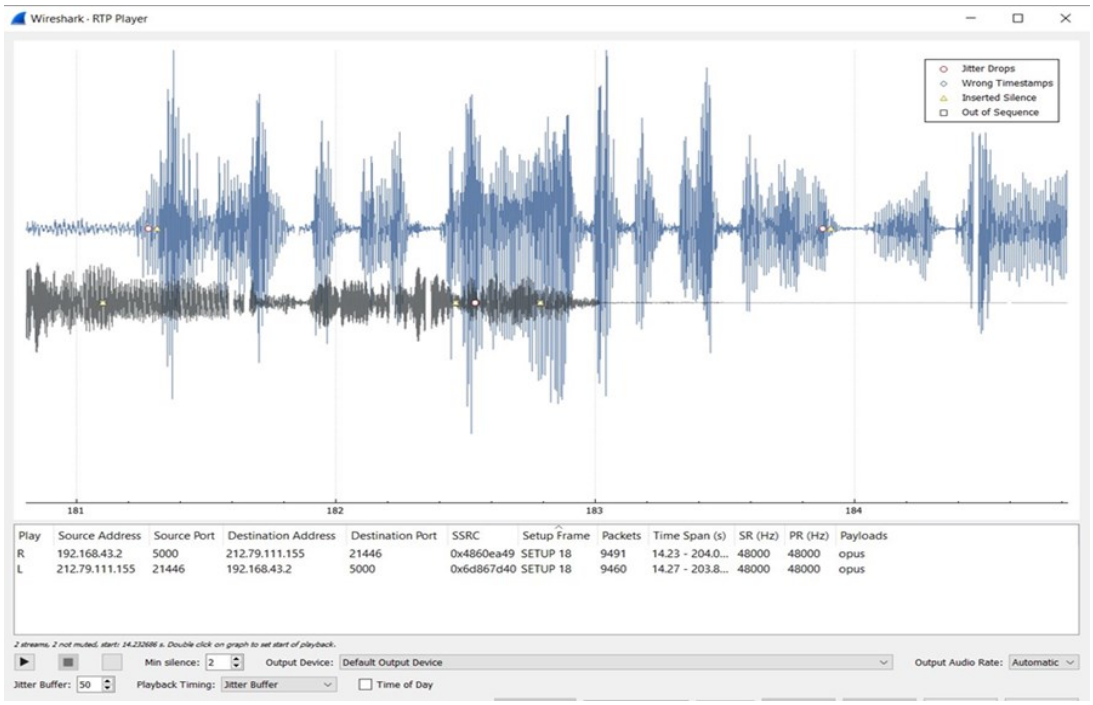
Slika 33. "Stream delta" (pozivatelj-SIP poslužitelj)

Slika 34. prikazuje kašnjenje u suprotnom smjeru komunikacije, te kao što je vidljivo, ni ovdje nije bilo kašnjenja preko dozvoljenih granica, odnosno kašnjenja preko 300 ms.



Slika 34. "Stream delta" (SIP poslužitelj-pozivatelj)

Odabirom *Telephony → RTP → RTP streams →Play streams* otvara se RTP player, Slika 35., posebna aplikacija unutar programskog alata Wireshark. Ona omogućava preslušavanje snimljenog sadržaja, kao i grafički prikaz snimljenog signala. Također, pruža brzi uvid u svaki medijski tok i njegove vrijednosti.

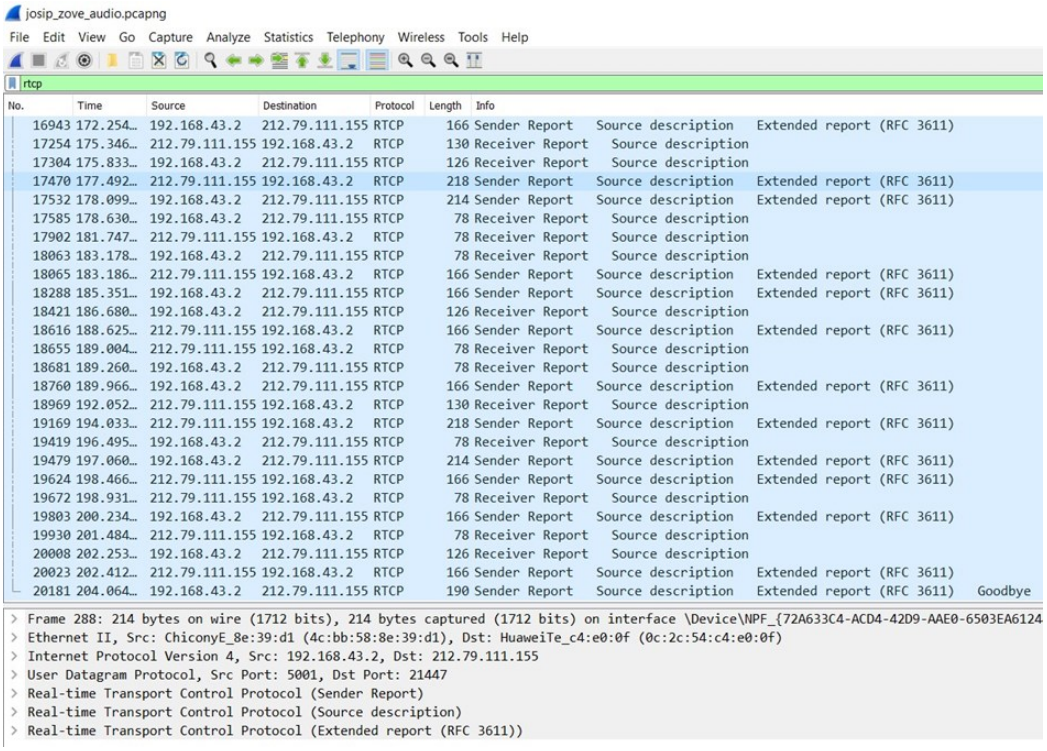


Slika 35. RTP player

10.3. Analiza RTCP paketa u audio sjednici

U ovom se dijelu rada analiziraju dijelovi RTCP paketa, točnije izvještaji pošiljatelja (engl. *Sender Report*) i izvještaji prijamnika (engl. *Receiver Report*). Vidjeti će se što točno predstavljaju bitne stavke ovoga protokola te koja im je funkcija prilikom prijenosa podataka.

Slika 36. prikazuje vrste izvještaja RTCP protokola, a u nastavku će se pojasniti svaki od njih.



Slika 36. Rezultat filtriranja RTCP paketa promatranog streama

10.3.1. Sender report

Izvještaj pošiljatelja (*Sender report*), sastoji se od tri dijela:

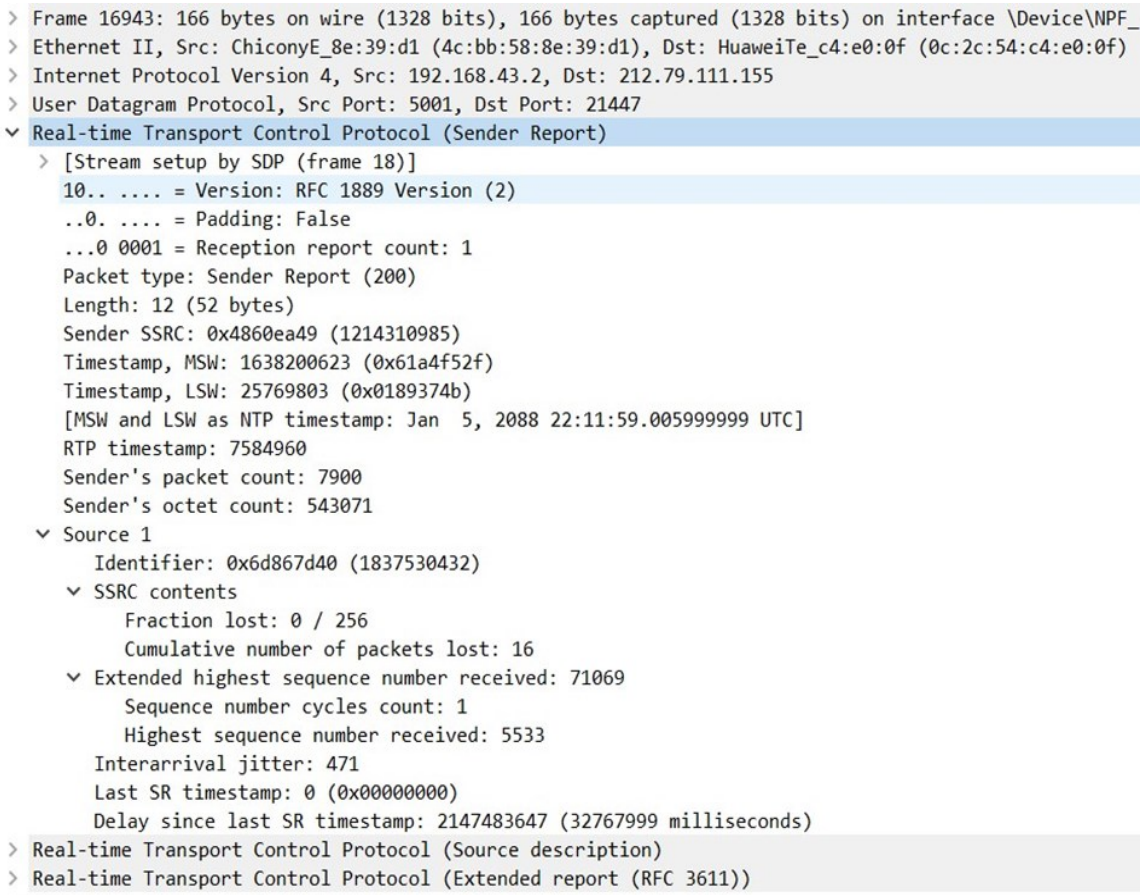
1. *Sender Report*
2. *Source description*
3. *Extended report*

Aktivni pošiljatelji povremeno šalju izvješće o pošiljatelju u sjednici radi prijenosa izvještaja i statistika prijema za sve RTP pakete. Izvješće pošiljatelja uključuje dvije različite vremenske oznake:

1. Apsolutna vremenska oznaka (*Network Time Protocol*) NTP – koja je u sekundama u odnosu na ponoć UTC 1.1.1900.

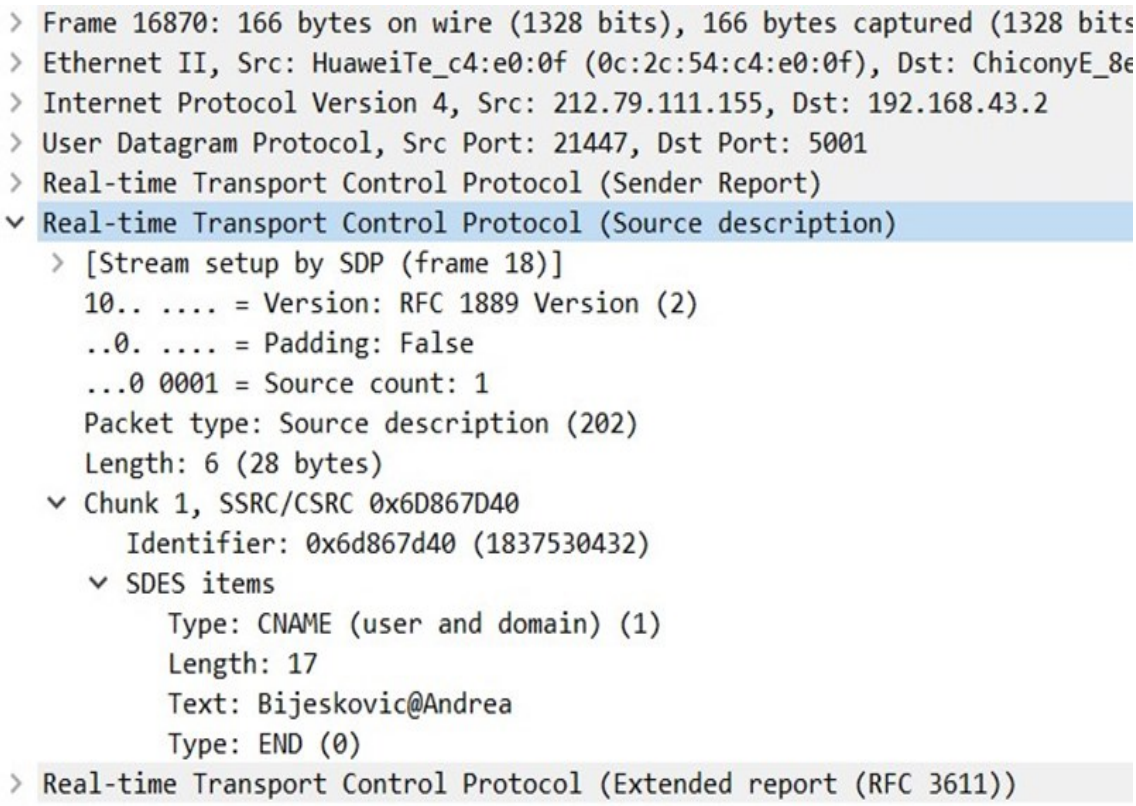
2. Vremensku oznaku RTP-a koja odgovara istom vremenu kao i NTP oznaka u istim jedinicama i s istim slučajnim pomakom kao RTP vremenske oznake u podatkovnim paketima opisanim u ovom izvještaju pošiljatelja.

Apsolutna vremenska oznaka omogućuje sinkronizaciju RTP poruka na prijemnoj strani, što je naročito važno kod prijenosa audio i video signala. Slika 37. prikazuje sadržaj *Sender Report* paketa.



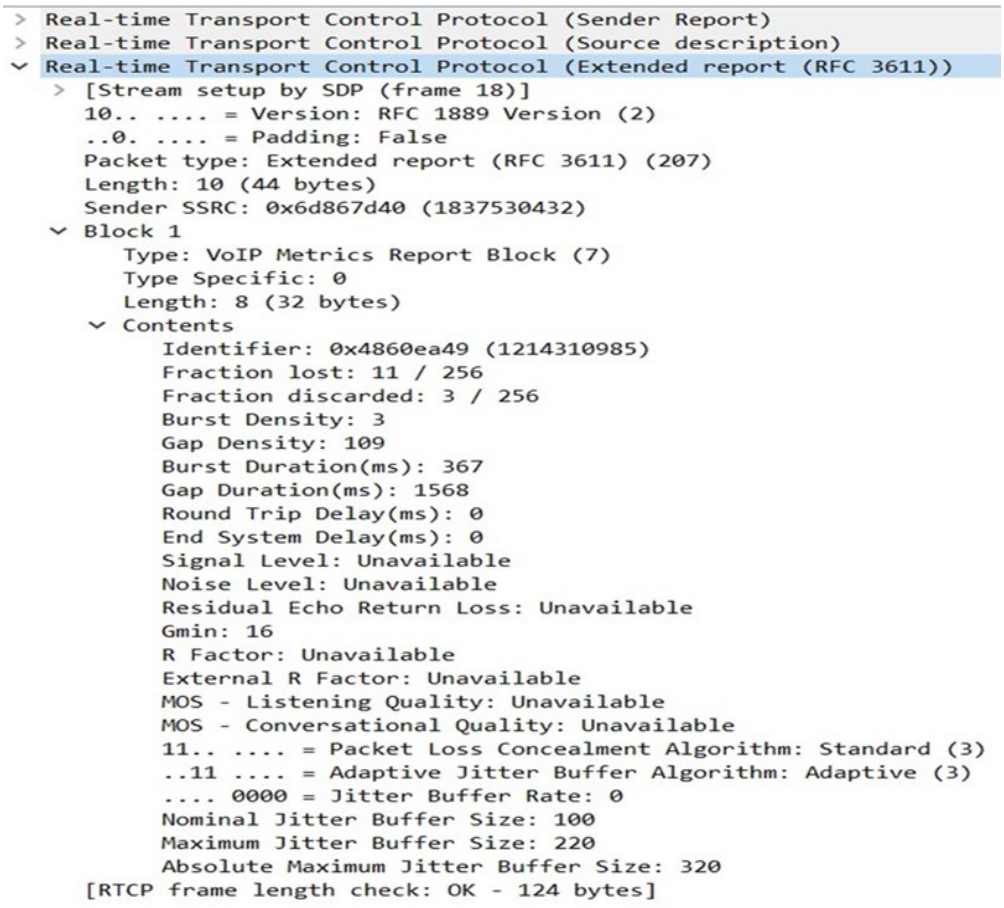
Slika 37. RTCP Sender Report (1)

Source description, Slika 38. se koristi za slanje CNAME-a (engl. *Canonical Name Record*) svim sudionicima sjednice.



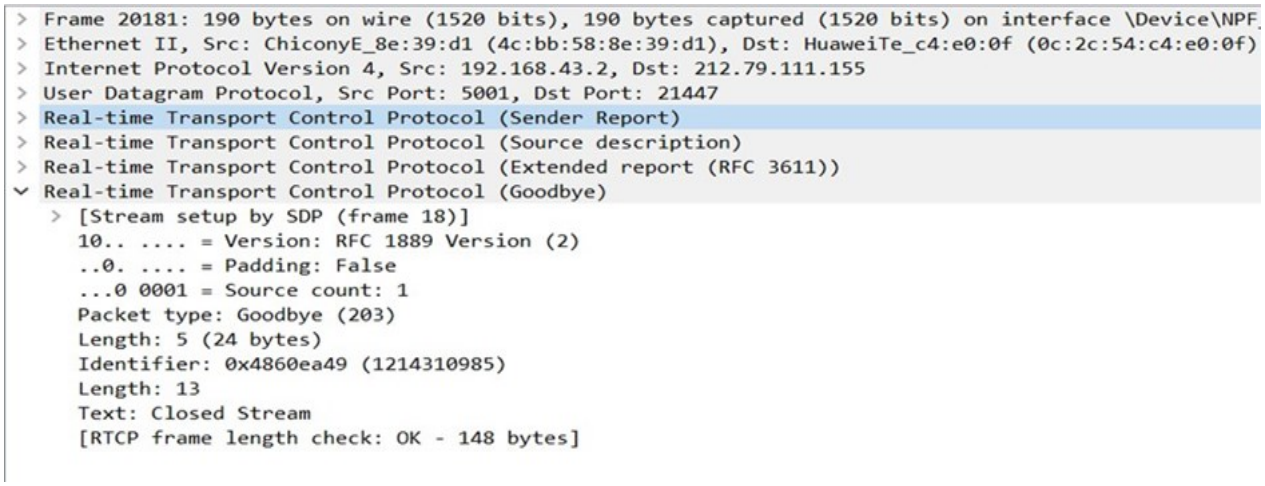
Slika 38. RTCP Source description (2)

Na Slici 39. prikazan je izgled i sadržaj proširenog izvještaja RTCP paketa (*RTCP Extended report*) koji pruža detaljnije informacije i statističke podatke koji pružaju mrežnim operatorima određene dodatne podatke iz kojih je moguće zaključiti, odnosno vidjeti podatke o mrežnim performansama i zadovoljenju kvalitete usluge krajnjeg korisnika.



Slika 39. RTCP Extended report (3)

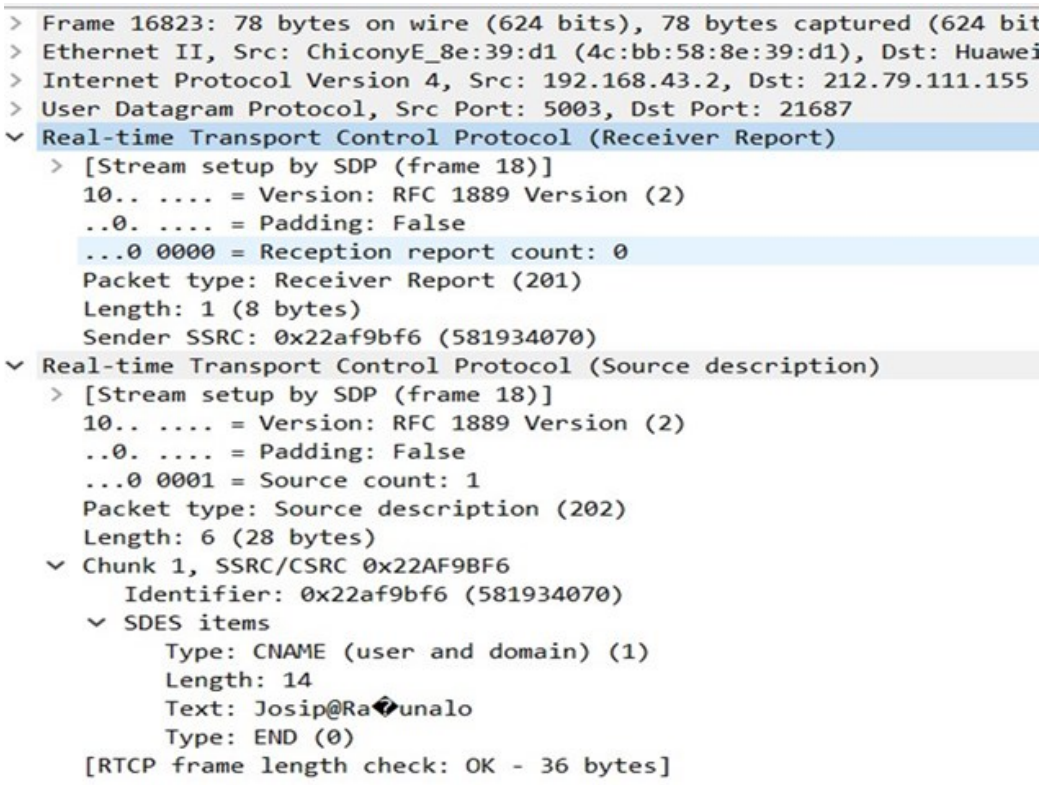
Izvor podataka šalje poruku *Goodbye*, da bi se isključio tok i time obavještava krajnju točku da napušta sjednicu. Sadržaj ove poruke vidljiv je na Slici 40. Ovaj paket je također dio RTCP *sender reporta*.



Slika 40. RTCP Goodbye

10.3.2. Receiver report

Ova vrsta izvještaja namijenjena je pasivnim sudionicima, odnosno onima koji ne šalju RTP pakete. Na ovaj način se obavještava pošiljatelja i druge primatelje o kvaliteti usluge. Opis izvora (SDES) - koristi se za slanje CNAME stavke svim sudionicima sesije te na taj način uspostavlja jedinstvenu identifikaciju krajnjih korisnika. Primjer ovakvog izvještaja prikazan je na Slici 41.



Slika 41. Receiver report

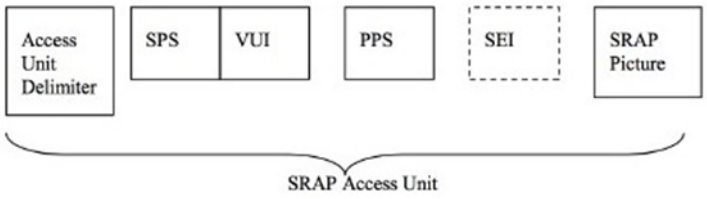
11. ANALIZA VIDEO SJEDNICE IZMEĐU DVA KORISNIKA

11.1. H.264 kodek

H.264 komprimira video podatke u dvije široke kategorije, koje obuhvaćaju: podatke o slici i dijelovima slike, koji se odnose izravno na jednu komprimiranu sliku, ili pak na dio slike. Druga se kategorija odnosi na meta podatke, odnosno one podatke koji bi mogli biti korisni za više slika ili isječaka, kao što su primjerice, podaci o rezoluciji slike, kompresijski parametri vremena prezentacije i dekodiranja i sl. [40]. Meta podaci nalaze su u nekoliko primarnih jedinica podataka: SPS (sadrži informacije potrebne za rekonstrukciju slijeda slika), u Informacijama o upotrebljivosti videa (VUI), koje sadrže potrebne informacije za rekonstruiranje videa iz rekonstruirane slike, odnosno dijelova slike, Skup parametara slike (PPS), koji sadrži informacije potrebne za rekonstrukciju pojedinačne slike ili reza slike i Dodatne informacije o poboljšanju (SEI), koje sadrže dodatne informacije korisne za poboljšanje videozapisa, primjerice titlovi.

Općenito govoreći, prilikom prijenosa ili pakiranja H.264 komprimiranog videa, podaci o slici su uključeni. Međutim, Meta podaci (PPS, SPS i SEI) mogu i ne moraju biti uključeni. H.264 standard je napisan kako bi se omogućilo slanje meta podataka odvojeno od podataka o slici, pod uvjetom da stignu u dekoder dovoljno unaprijed. Neki od njih (poput PPS podataka) mogu se čak i unaprijed konfigurirati i ostaju statični tijekom cijelog procesa kodiranja i stoga se uopće ne šalju. Primarni dio standarda H.264 ne uključuje meta podatke u sintaksi toka. Pruža samo sintaksu za sliku/slajd toka podataka. Dodatak B standardu H.264 pruža mehanizam za prijenos kompletnog mrežnog toka sa svim informacijama potrebnim dekoderu za ponovno stvaranje cijelog videa, uključujući SPS, PPS, VUI, SEI i druge podatkovne jedinice [40].

Dodatak B je sintaksa koju koristi MPEG2-TS za prijenos H.264. Blu-ray diskovi, DVB i ATSC koriste ovo mapiranje za prenošenje H.264 u MPEG2-TS. SCTE koristi ograničeni oblik dodatka B za prijenos na američkim kabelskim sustavima. Ograničava vrijeme između I- ili IDR-okvira (sadržanih u SCTE paketu sa slučajnim pristupom (SRAP)) na manje od 1,0 sekunde, kako bi se osiguralo razumno vrijeme promjene kanala.

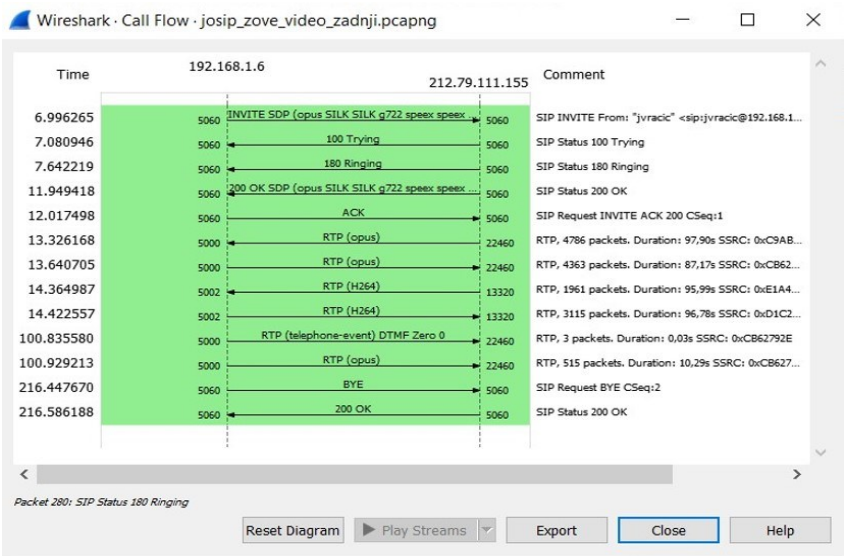


Slika 42. Struktura NAL pristupne jedinice

11.2. Analiza video-poziva između dva korisnika

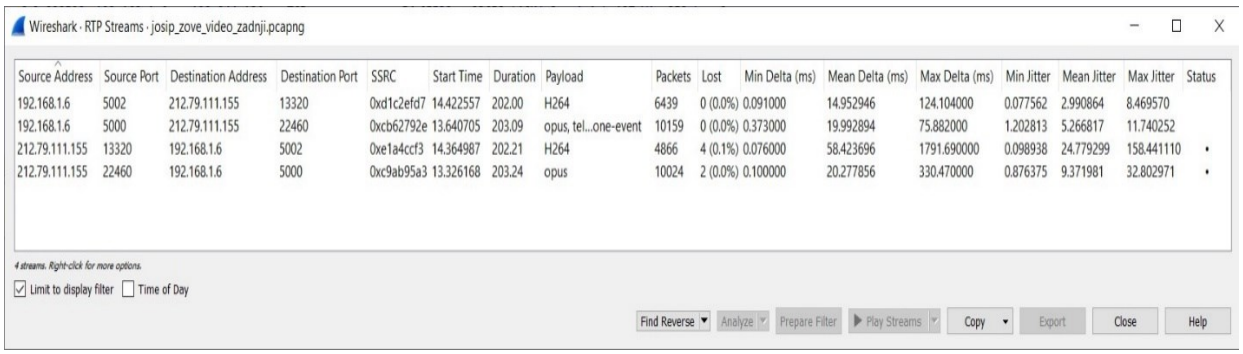
Video-poziv se bitno razlikuje od audio-poziva po tome što postoje mnogi slučajevi upotrebe u kojima su poželjni asimetrični medijski tokovi pa tako primjerice, širokopojasne usluge gdje su različite brzine prijenosa i preuzimanja te zbog toga što je kodiranje videozapisa zahtjevnije za procesor nego dekodiranje. Krajnje točke videozapisa obično se mogu dekodirati u višoj rezoluciji nego kodirati. Zbog potrebe za podrškom asimetričnim video *stream-ovima* mogućnosti videokodeka nalaze se u SDP opisu paketa, gdje ponudu i odgovor treba promatrati kao mogućnost primanja sadržaja poziva krajnjih točaka, više nego pregovaračkim sposobnostima zajedničkima za oba uređaja.

Analogno načinu analize audio-poziva, analizirati će se i video-poziv. Slika 43. prikazuje *„flow graph“* video poziva. Kao što je vidljivo iz slike za razliku od audio sjednice, ovdje se prikazuju i dijelovi RTP paketa koji se do sada nisu pojavljivali, a biti će objašnjeni u nastavku rada.



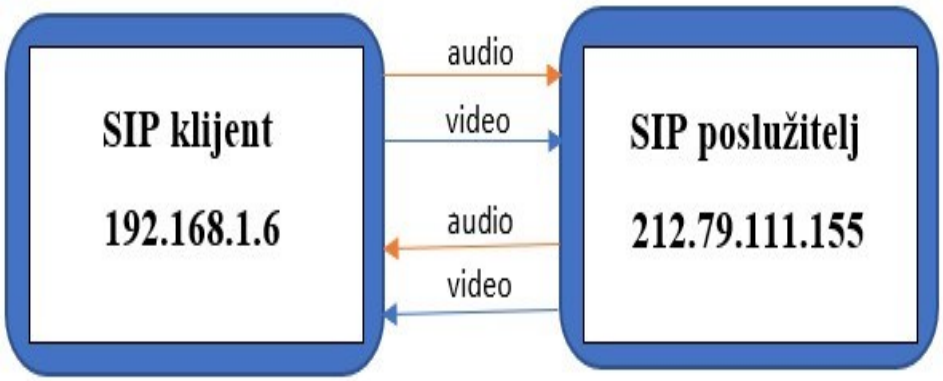
Slika 43. "Flow graph" video poziva

Slika 44. prikazuje postojeće RTP tokove unutar sjednice, a za razliku od audio sjednice, video sjednica sadrži četiri RTP toka. Dva su za prijenos audio signala, a preostala dva za video signal. Razmjena RTP paketa vrši se od pozivatelja prema SIP poslužitelju i obratno.



Slika 44. RTP tokovi unutar video sjednice

Slika 45. prikazuje pojednostavljeni prikaz razmjene RTP paketa između SIP klijenta i SIP poslužitelja.



Slika 45. Pojednostavljeni prikaz razmjene RTP paketa

Slika 46. prikazuje jedan RTP paket, koji prenosi audio signal, te u njemu nema značajnijih razlika u odnosu na RTP paket iz audio sjednice.

```
> Frame 3134: 63 bytes on wire (504 bits), 63 bytes captured (504 bits) on interface \Device\NPF_{72A6
> Ethernet II, Src: UbeeInte_82:f3:e3 (6c:38:a1:82:f3:e3), Dst: ChiconyE_8e:39:d1 (4c:bb:58:8e:39:d1)
> Internet Protocol Version 4, Src: 212.79.111.155, Dst: 192.168.1.6
> User Datagram Protocol, Src Port: 22460, Dst Port: 5000
v Real-Time Transport Protocol
  > [Stream setup by SDP (frame 258)]
    10.. .... = Version: RFC 1889 Version (2)
    ..0. .... = Padding: False
    ...1 .... = Extension: True
    .... 0000 = Contributing source identifiers count: 0
    0... .... = Marker: False
    Payload type: opus (96)
    Sequence number: 46163
    [Extended sequence number: 46163]
    Timestamp: 679680
    Synchronization Source identifier: 0xc9ab95a3 (3383465379)
    Defined by profile: Unknown (0xbede)
    Extension length: 1
  > Header extensions
> Opus Interactive Audio Codec
```

Slika 46. Prikaz RTP paketa za prijenos audio signala

Kao što Slika 42. prikazuje, uz RTP pakete za prenošenje audio signala, postoje i oni namijenjeni prenošenju video signala, a isti su označeni sa kodekom koji ih obrađuje, a u ovom slučaju to je H.264. Izgled jednog RTP paketa koji u sebi sadrži H.264 kodek prikazan je na Slici 47.

Wireshark · Packet 632 · josip_zove_video_zadnji.pcapng

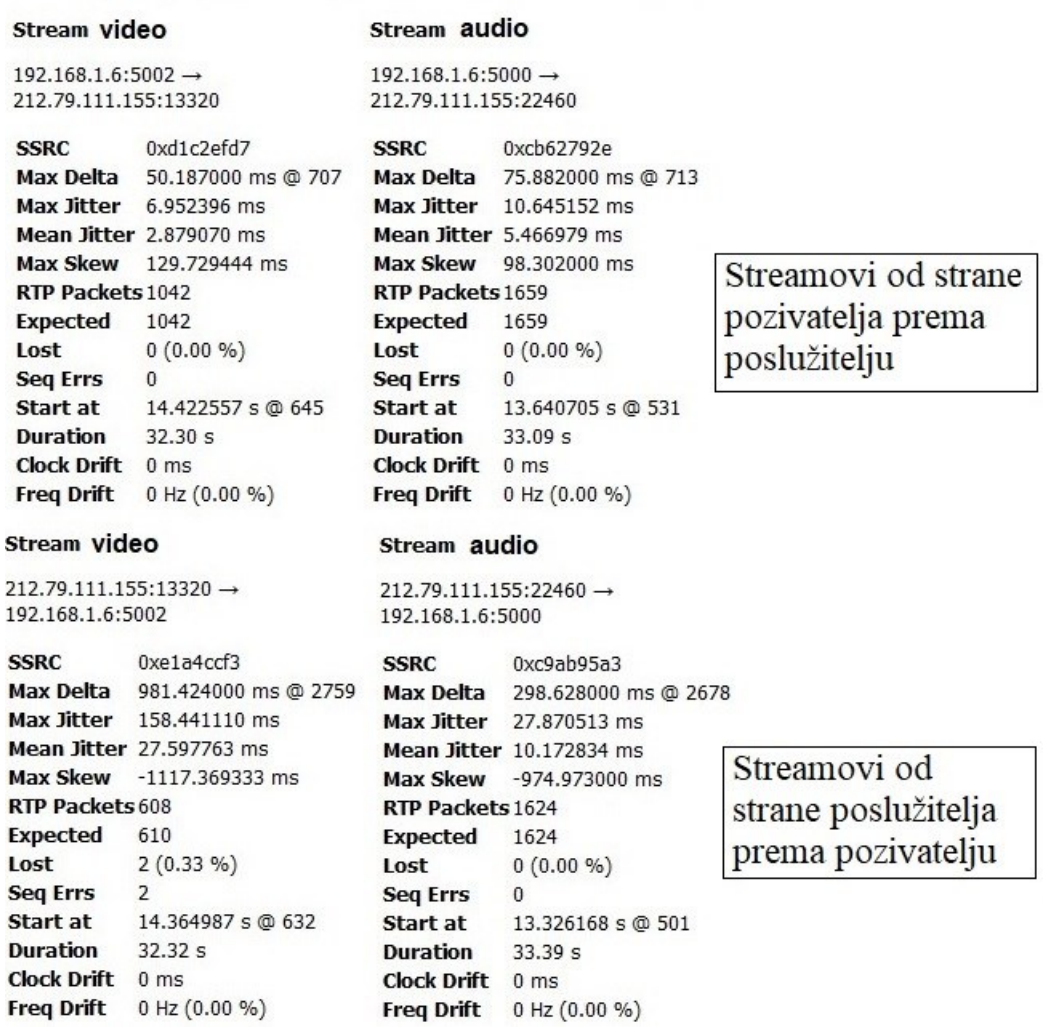
```
> Frame 632: 77 bytes on wire (616 bits), 77 bytes captured (616 bits
> Ethernet II, Src: UbeeInte_82:f3:e3 (6c:38:a1:82:f3:e3), Dst: Chico
> Internet Protocol Version 4, Src: 212.79.111.155, Dst: 192.168.1.6
> User Datagram Protocol, Src Port: 13320, Dst Port: 5002
v Real-Time Transport Protocol
  > [Stream setup by SDP (frame 258)]
    10.. .... = Version: RFC 1889 Version (2)
    ..0. .... = Padding: False
    ...0 .... = Extension: False
    .... 0000 = Contributing source identifiers count: 0
    0... .... = Marker: False
    Payload type: H264 (105)
    Sequence number: 28926
    [Extended sequence number: 94462]
    Timestamp: 3711763476
    Synchronization Source identifier: 0xe1a4ccf3 (3785673971)
v H.264
  > NAL unit header or first byte of the payload
  > H264 NAL Unit Payload
```

Slika 47. RTP paket koji sadrži H.264 kodek

Zahtjeve za videokonferenciju moguće je koristiti kao sposobnost „jedan na jedan“, odnosno, kao konferenciju, ali sa više točaka, a ti su zahtjevi sljedeći:

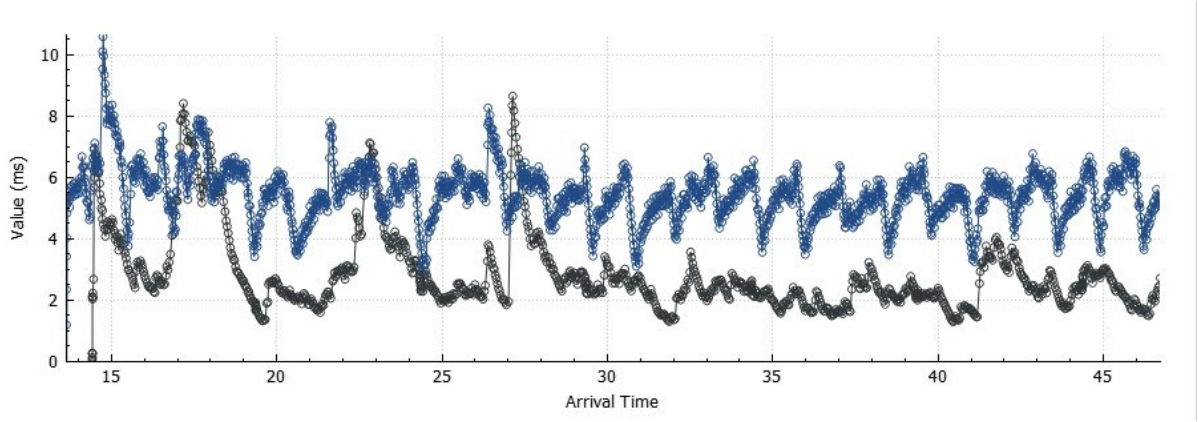
- ≤ 150 ms jednosmjernog kašnjenja (prema ITU G.114 standardu).
- Podrhtavanje od 30 ms.
- ≤ 1 posto gubitka paketa.
- garancija minimalne širine pojasa -sjednica videokonferencija plus 20 %.
  Primjerice, sesija videokonferencije od 384 kbps traži 460 kbps zajamčenu prioritetnu širinu pojasa [41].

Slika 48. prikazuje sažete rezultate Wiresharka za obje strane komunikacije.



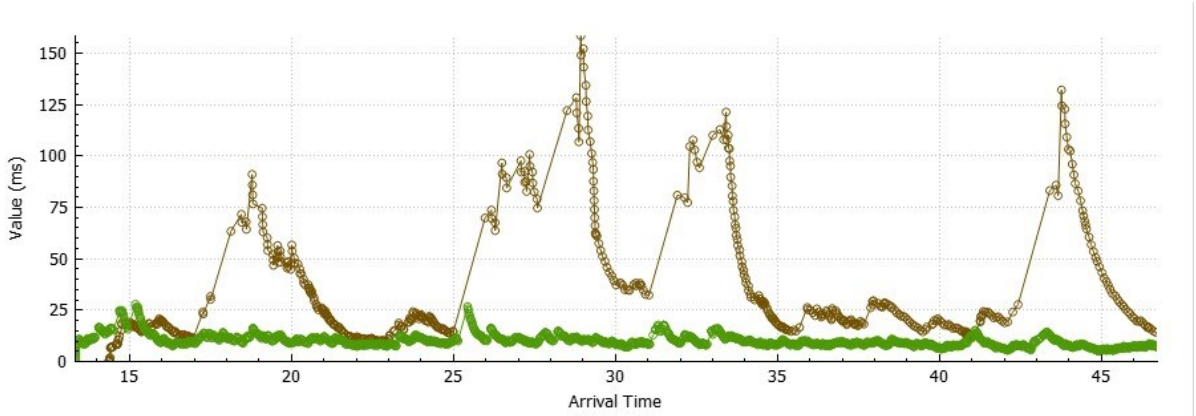
Slika 48. Sažeti rezultati Wiresharka za obje strane komunikacije

Slika 49. prikazuje varijacije u kašnjenju paketa u smjeru pozivatelja prema SIP poslužitelju. Vidljivo je da maksimalni iznos *„jittera“* iznosi 10.64 ms, što je manje od dozvoljenih 30 ms propisanim prema ITU G.114 standardu.



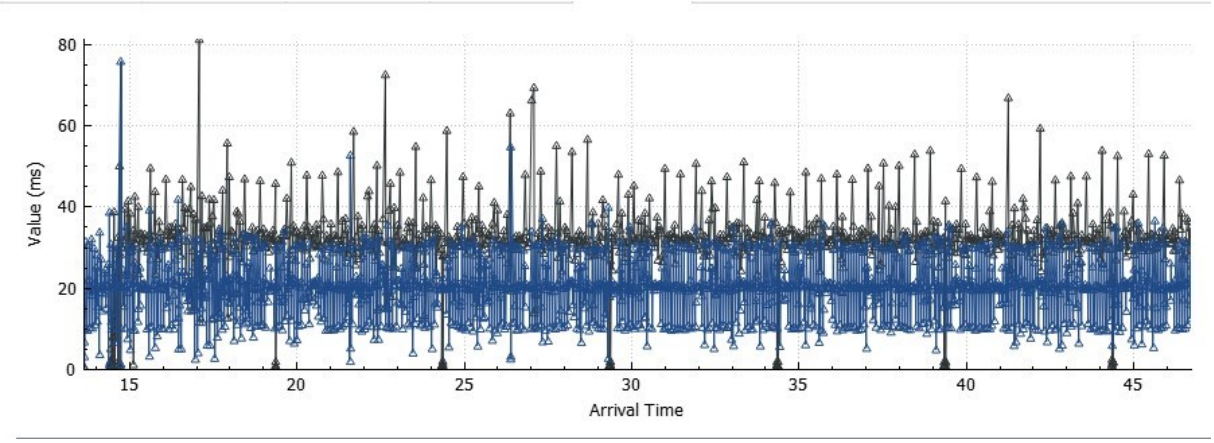
Slika 49. "Stream Jitter" audio i video signala u smjeru pozivatelj-SIP poslužitelj

Slika 50. prikazuje varijacije u kašnjenju u smjeru SIP poslužitelja prema pozivatelju, te je vidljivo kako je u jednom trenutku došlo do velikog odstupanja, u kašnjenju od 158.44 ms, i to prilikom prijenosa videosignala. Upravo zbog toga je došlo i do gubitka nekoliko paketa upisanih u sažetim rezultatima. Nakon nekoliko trenutaka veza se stabilizirala, te više nije bilo takvih odstupanja.



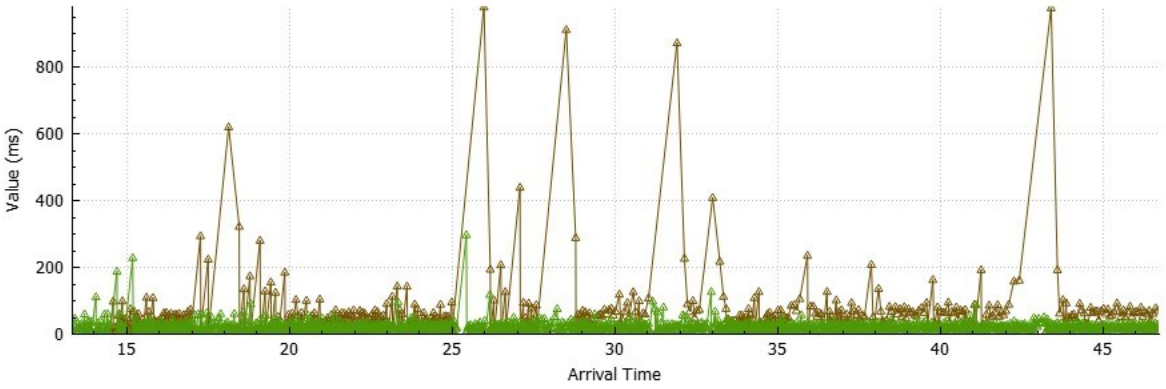
Slika 50. "Stream Jitter" audio i video signala u smjeru SIP poslužitelj-pozivatelj

Slika 51. prikazuje kašnjenja za cijeli razgovor, promatrano od strane pozivatelja prema SIP poslužitelju. Iz analize grafa vidljivo je da nije bilo vidljivih odstupanja od nominalnih vrijednosti, odnosno, kašnjenje nije prelazilo 150 ms.



Slika 51. "Stream Delta" audio i video signala: pozivatelj-SIP poslužitelj

Slika 52. prikazuje kašnjenja za cijeli razgovor promatrano od strane SIP poslužitelja prema pozivatelju. Iz analize grafa vidljivo je kako su vrijednosti kašnjenja prelazile dozvoljene granice iznad 150 ms, te su u nekoliko trenutaka vrijednosti kašnjenja bile i do nekoliko puta veće. Odstupanja iznad dozvoljenih vrijednosti dovode do gubitka paketa i loše kvalitete komunikacije.



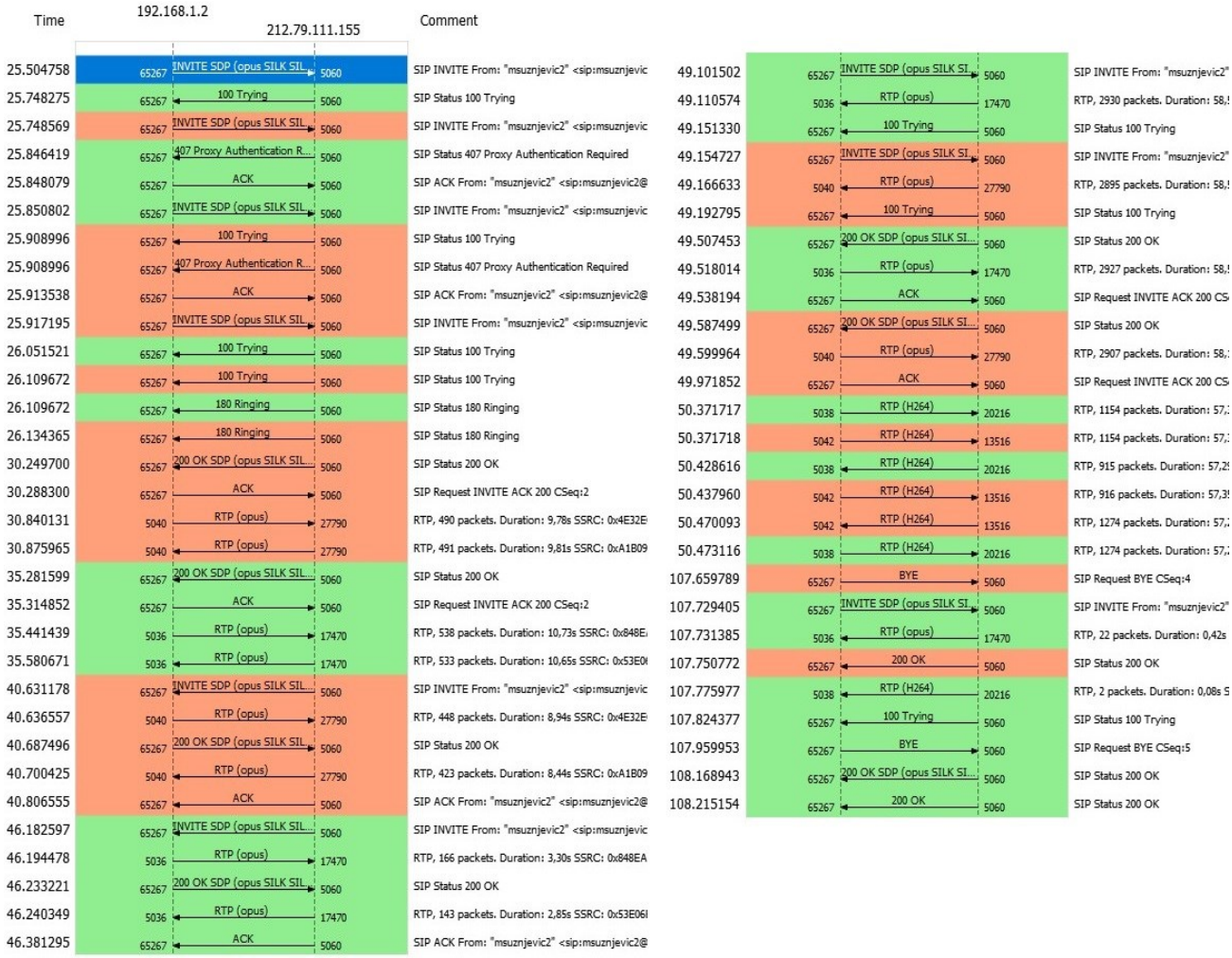
Slika 52. "Stream Delta" audio i video signala: SIP poslužitelj-pozivatelj

Slika 53. prikazuje RTP *Player* pomoću kojega se može preslušati audio zapis razgovora, te se također može vidjeti i koji signal ima najviše odstupanja od nominalnih vrijednosti.

Slika 53. RTP Player

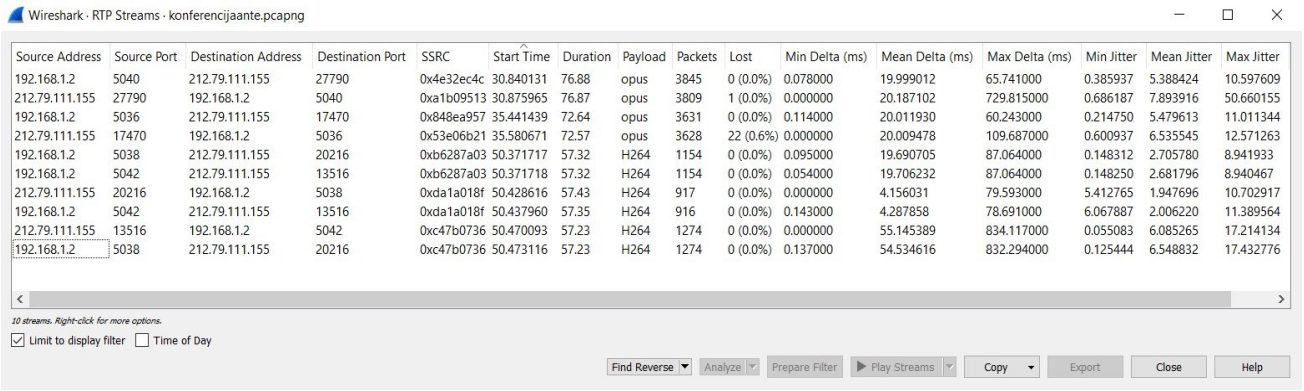
12. ANALIZA SJEDNICE IZMEĐU VIŠE OSOBA – KONFERENCIJSKI POZIV

Slika 54. prikazuje „flow graph“ konferencijskog video poziva. Korisnik msuznjevic2@iptel.org je kreirano sjednicu, prvo je poslan INVITE zahtjev korisniku jvracic@iptel.org. Zatim on prihvaća poziv sa 200 OK, kreator sjednice odgovara sa ACK te započinje prijenos audio i video paketa. Nakon toga, korisniku vvracic@iptel.org također se šalje INVITE poziv, a korisnik zatim odgovara na takav poziv, te je na taj način konferencija uspostavljena. Nakon uspostavljanja audio sjednice, svi sudionici poziva su uključili kameru.



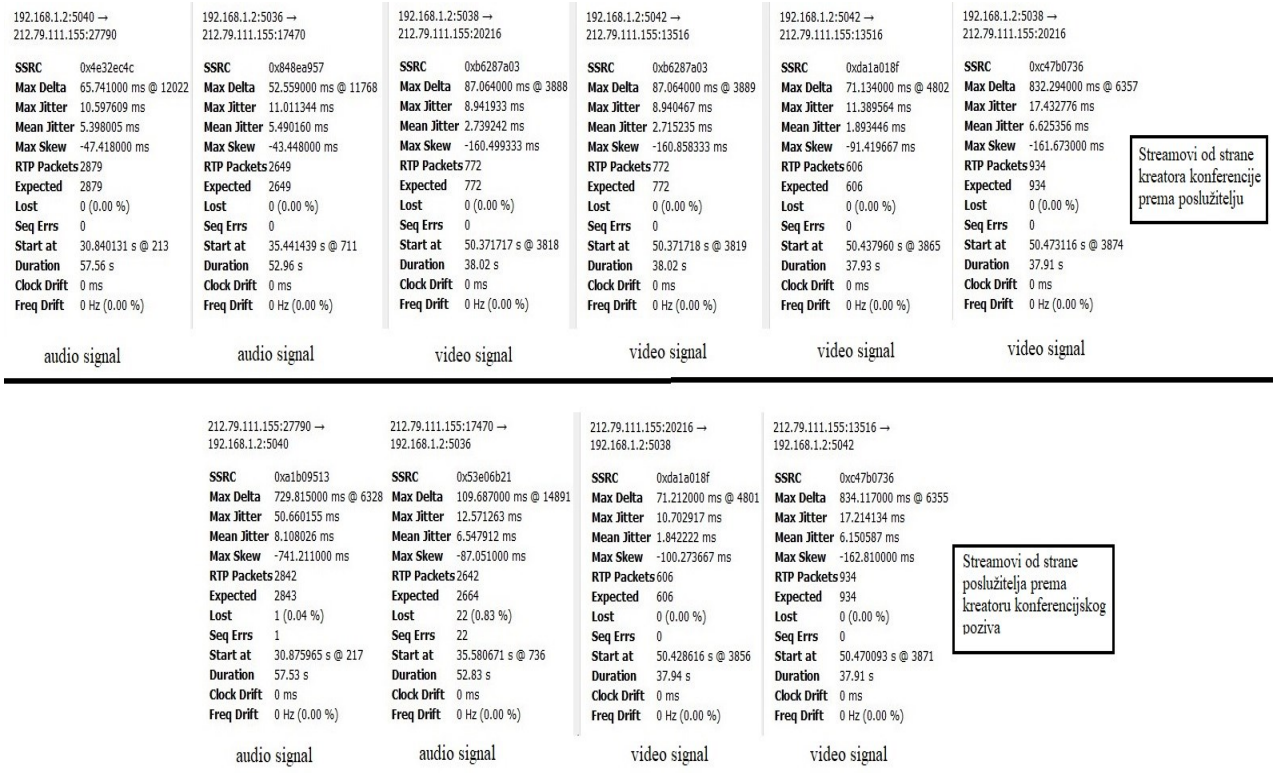
Slika 54. "Flow graph" videokonferencije

Slika 55. prikazuje RTP tokove unutar konferencije. Prema broju portova, vidljivo je da ne postoji direktna veza između korisnika jvracic@iptel.org i vvracic@iptel.org, nego SIP račun kreatora sjednice msuznjevic2@iptel.org zapravo služi kao poveznica između njih.



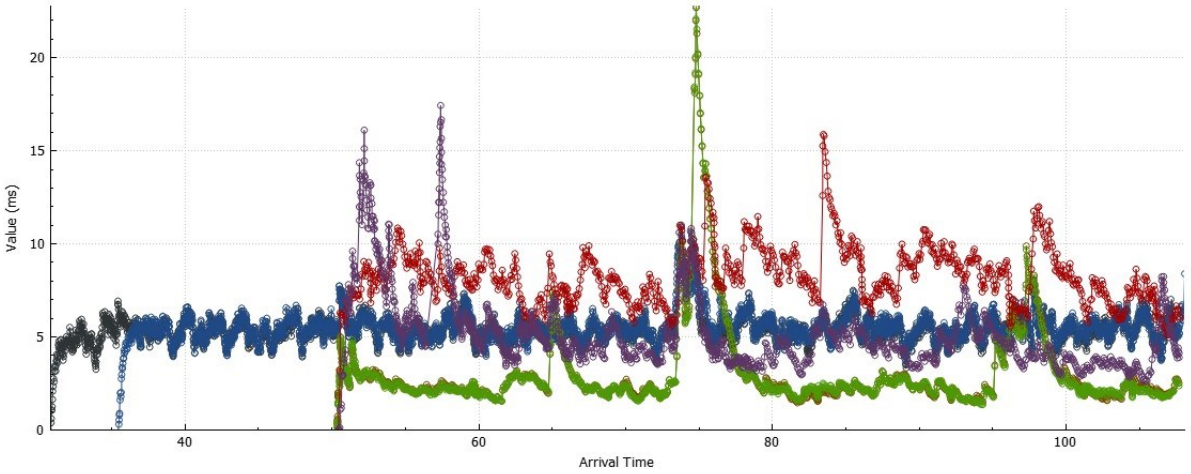
Slika 55. RTP tokovi unutar konferencijskog poziva

Slika 56. prikazuje sažete rezultate Wiresharka za obje strane komunikacije.



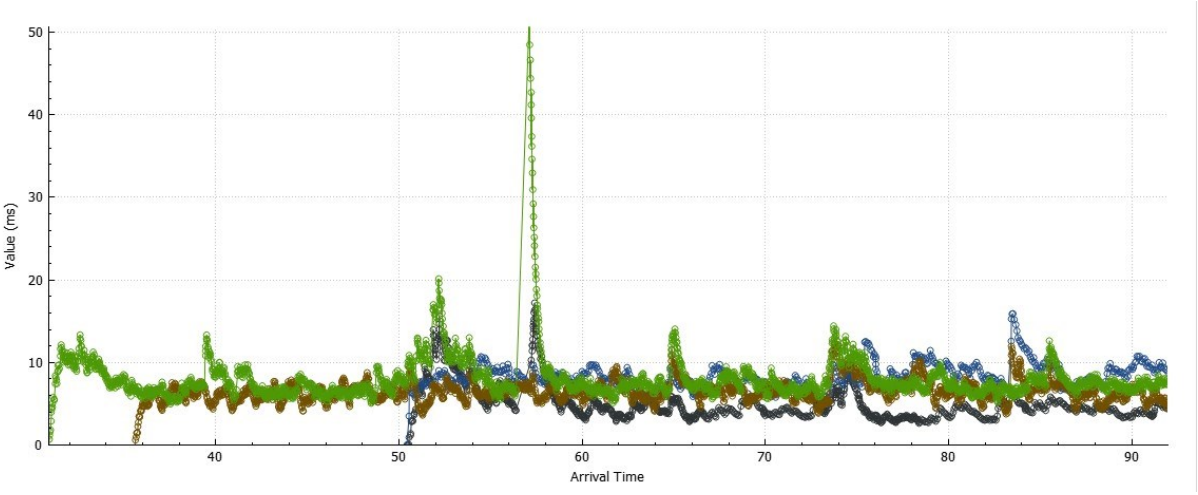
Slika 56. Sažeti rezultati Wiresharka za obje strane komunikacije

Nadalje, Slika 57. prikazuje varijacije kašnjenja RTP tokova u smjeru kreatora sjednice prema SIP poslužitelju. Vidljivo je da je maksimalni iznos kašnjenja nešto iznad 20 ms, što je i dalje unutar dozvoljenih granica od 20 ms.



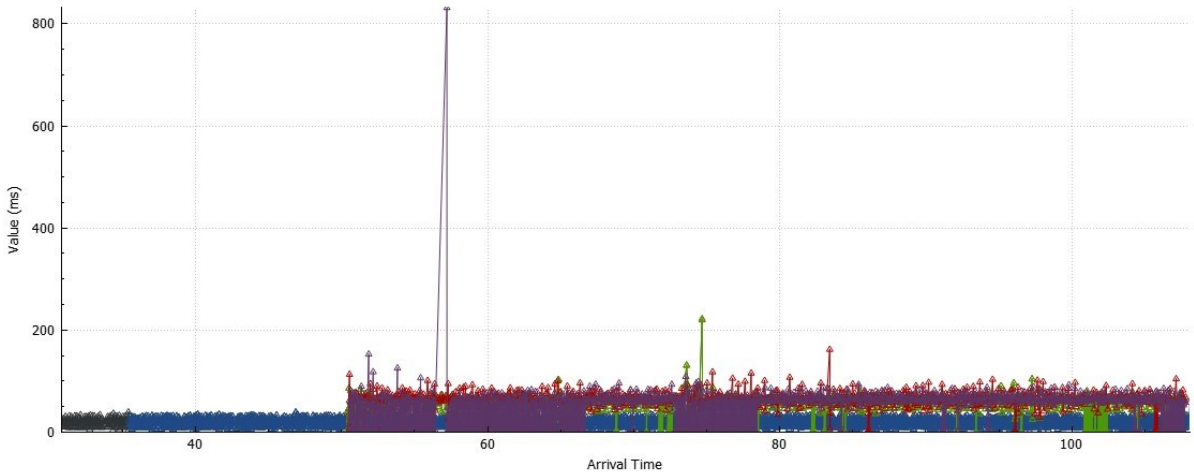
Slika 57. Stream "Jitter" audio i video signala: kreator sjednice-SIP poslužitelj

Slika 58. prikazuje varijacije u kašnjenju u smjeru poslužitelj-kreator sjednice. Vidljivo je da je u određenom trenutku iznos kašnjenja prešao dozvoljenu granicu od 30 ms, a nakon toga se *stream* ipak stabilizirao te nije više bilo tolikih odstupanja.



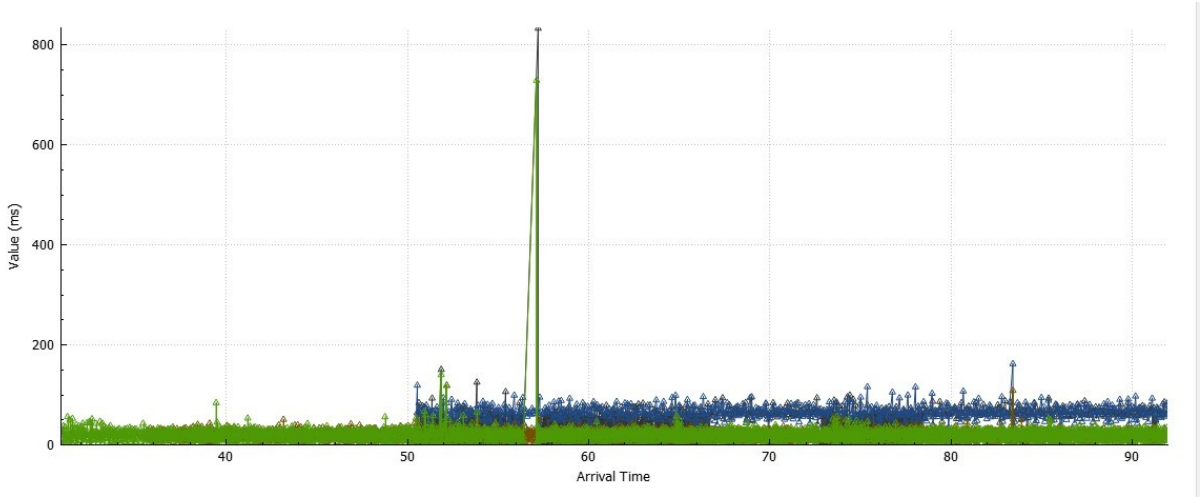
Slika 58. Stream "Jitter" audio i video signala: SIP poslužitelj-kreator sjednice

Slika 59. prikazuje kašnjenje za cijeli razgovor, promatrano od strane pozivatelja prema SIP poslužitelju. Iz analize grafa je vidljivo da je u određenom trenutku maksimalna vrijednost kašnjenja nadvisila dozvoljenu vrijednost, no nakon toga *stream* se stabilizirao.



Slika 59. "Stream Delta" audio i video signala: pozivatelj-SIP poslužitelj

Slika 60. prikazuje kašnjenje za cijeli razgovor promatrano od strane SIP poslužitelja prema pozivatelju. Iz analize grafa vidljivo je da je vrijednost kašnjenja prelazila dozvoljene granice iznad 150 ms, a u određenom su trenutku vrijednosti kašnjenja bile i do nekoliko puta veće od dozvoljenih. Odstupanja iznad dozvoljenih vrijednosti dovode do gubitka paketa i loše kvalitete komunikacije.



Slika 60. "Stream Delta" audio i video signala: SIP poslužitelj-pozivatelj

Slika 61. prikazuje RTP *Player* konferencijskog videopoziva, gdje je moguće poslušati pojedinačno svaki *stream*, kao i cijeli razgovor. Preduvjet za to jest da podaci koji se prenose ne budu kriptirani.

Slika 61. RTP Player - konferencijski poziv

13. ZAKLJUČAK

Cilj je ovog rada bio analizirati RTP i RTCP protokole, uključujući i različite scenarije njihove primjene u praksi. Uz pomoć mrežnog analizatora (Wireshark) analizirani su podatkovni tokovi radi analize RTP i RTCP protokola s naglaskom na kvalitetu i performanse navedenih.

Zaključno, budući da je ključni dio ovog rada bila sama analiza prometa, odnosno prijenosa podataka u audio i video sjednici među dva korisnika, te u sjednici između tri korisnika, u nastavku se ukratko iznose zaključci do kojih se došlo za vrijeme analize. Dakle, podaci koje snimi program Wireshark nisu nužno dio obavljenog razgovora, budući da navedeni program automatski snima i sav promet na mreži, a ne samo onaj koji se želi analizirati. Uspješnu pretragu i analizu sjednice omogućava upotreba filtera, te su zahtjevi i odgovori standardni za svaku SIP sjednicu. Prilikom svakog prijenosa podataka postoji određeno kašnjenje, no zbog razvoja internet tehnologije i mrežne infrastrukture, brzine interneta su poprilično velike, stoga nema puno gubitaka paketa. Zbog toga što RTP protokol ne garantira trenutnu dostavu paketa, usporedno s njim analiziran je RTCP protokol koji pruža povratne informacije o kvaliteti usluge RTP protokola. Kod audio sjednice između dva korisnika postoje dva RTP toka za prijenos audio signala, kod video sjednice između dva korisnika postoje četiri RTP toka; dva za audio, dva za video u oba smjera, dok kod konferencijskog poziva RTP tokova u ovom slučaju ima deset, što znači da više korisnika postavlja veće zahtjeve i opterećenje za mrežu, te je samim time gubitak paketa veći, a kvaliteta komunikacije lošija.

Sažetak

Kontinuirani razvoj tehnologije imao je velik utjecaj na način današnje komunikacije i razmjene informacija, stoga se kroz rad nastojalo analizirati RTP (engl. *Real Time Transport Protocol*) i RTCP protokole (engl: *Real Time Control Protocol*). U prošlosti su prve mreže omogućavale prenošenje samo jedne vrste informacija, dok je danas situacija znatno drugačija, stoga je današnjom tehnologijom moguće istovremeno obaviti prijenos više različitih vrsta informacija. RTP protokol je tako razvijen kako bi bio standardizirana podrška prilikom prijenosa isporuke audio-vizualnog sadržaja, u stvarnom vremenu i sa što manje kašnjenja, a uvijek se koristi u korelaciji sa RTCP protokolom. Za potrebe izrade ovog rada korišten je mrežni analizator Wireshark, pomoću kojega su analizirani podatkovni tokovi radi analize RTP i RTCP protokola. Dakle, pomoću istog programa analizirana je audio sjednica, video sjednica te video konferencija, a promet navedenih sjednica odvijao se putem aplikacije Jitsi.

Ključne riječi: RTP, RTCP, VoIP, Wireshark, Jitsi

Summary

Continuous development of technology has had a great impact on the way of today's communication and information exchange, so the paper sought to analyze RTP (Real-Time Transmission Protocol) and RTCP Protocol (Real-Time Control Protocol). In the past, the first networks allowed the transmission of only one type of information, while today the situation is much different, so with today's technology it is possible to transmit several different types of information at the same time. The RTP protocol has been developed to be a standardized support for the transmission of audio-visual content delivery, in real time and with as few delays as possible, and is always used in correlation with the RTCP protocol. For the purposes of this paper, the Wireshark network analyzer was used, which was used to analyze data flows for the analysis of RTP and RTCP protocols. Accordingly, with the help of the same program, the audio session, video session and video conference were analyzed, and the traffic of these sessions took place via Jitsi applications.

Keywords: RTP, RTCP, VoIP, Wireshark, Jitsi

POPIS LITERATURE

1. Carnet – Hrvatska akademska i istraživačka mreža: Analiza mrežnog prometa, dostupno na: https://www.cis.hr/www.edicija/LinkedDocuments/CCERT-PUBDOC-2004-09-90.pdf?fbclid=IwAR15pqfDY1YM%20H-WedNl01H_I-KqVr9ccnCBpVO9i4UUOmE1Z9egF93zbDpw, pristupljeno 09. 08. 2021.
2. How and why automation can improve network – device securitiy, dostupno na: https://www.networkworld.com/a rticle/3239677/the-osi-model-explained-and-how-to-easily-remember-its-7-layers.html, pristupljeno 09. 08. 2021.
3. Uvod mrežne protokole, dostupno na: https:// forum.bug.hr/ forum/ topic/internet/uvod-mrezne-protokole/96549.aspx, pristupljeno 10. 08. 2021.
4. TCP/IP arhitektura, dostupno na: http://mreze.layer-x.com/s010200-0.html, pristupljeno 11. 08. 2021.
5. OSI i TCP IP model – 2021 – INTERNET, dostupno na: https://hr.weblogographic.com/difference-between-osi-and-tcp-ip-model-7779, pristupljeno 11. 08. 2021.
6. https://sysportal.carnet.hr/system/files/tcp_ip_model.png, pristupljeno 11. 08. 2021.
7. Računalne mreže – OSI referentni model, dostupno na: https://sysportal. carnet.hr/node/352, pristupljeno 12. 08. 2021.
8. Top Ten Security Issues Voice over IP (VoIP), dostupno na: https://www.scribd.com/document/246746404/Top-Ten-Voip-Security-Issue, pristupljeno 12. 08. 2021.
9. https://www.ericsson.hr/etk/revija/Br_1_2001/govor_slike/3.jpg, pristupljeno 12. 08. 2021.
10. Malo o protokolima, dostupno na: http://pvprm.zesoi.fer.hr/2001-2002-web/studenti-rad/seminar-radio/Stranice/protokoli.htm, pristupljeno 13. 08. 2021.
11. RSVP protokol, dostupno na: http://pvprm.zesoi.fer.hr/2003-2004-web/studenti-rad/nkozul/seminar.html, pristupljeno 13. 08. 2021.
12. The Use of RSVP with IETF Integrated Services, dostupno na: https://datatracker.ietf.org/doc/html/rfc2210, pristupljeno 13. 08. 2021.

13.VoIP Fundamentals (Introducing Voice over IP Networks) Part 1, dostupno na: http://what-when-how.com/cisco-voice-over-ip-cvoice/voip-fundamentals-introducing-voice-over-ip-networks-part-1/, pristupljeno 13. 08. 2021.

14. The H. 323 Standard, dostupno na: https://msdn.microsoft.com/en-us/library/ms709083(v=vs.85).aspx, pristupljeno 13. 08. 2021.

15. Integracija telefonske i internet komunikacije, VOIP standardi i protokoli, dostupno na: https://sites.google.com/site/boskovicsrdjan123/home/voip-standardi-i-protokoli, pristupljeno 13. 08. 2021.

16. Sigurni VoIP, dostupno na: https: // security. foi. Hr / wiki / index . php / Sigurni_ VoIP.html#.C5.A0to_je_VoIP.3F, pristupljeno 13. 08. 2021.

17.Kvaliteta usluge – QoS i VoIP, dostupno na: https: // hr. Eye wated . com / kvaliteta-usluge-qos-i-voip/, pristupljeno 14. 08. 2021.

18. Upravljanje mrežom, dostupno na: http://estudent.fpz.h r/Predmeti/A/Arhitektura_telekomunikacijske_mreze/Materijali/6_Arhitektura_multimedijskih_mreza_-_10112 016.pdf, pristupljeno 14. 08. 2021.

19. 29. Telekomunikacioni forum Telfor 2021, dostupno na: https: // www. Telfor.rs/telfor2006/Radovi/02_TM_07.pdf, pristupljeno 14. 08. 2021.

20. UDP protokol, dostupno na: http://mihael-kasica.from.hr/udp-protokol/, pristupljeno 15. 08. 2021.

21. UDP, dostupno na: https://hr.wikipedia.org/wiki/UDP, pristupljeno 15. 08. 2021.

22. UDP protokol, dostupno na: http://mreze.layer-x.com/s040200-0.html, pristupljeno 15. 08. 2021.

23. Obitelj Internet protokola, skupina TCP/IP protokola, dostupno na: https://pcchip.hr/helpdesk/obitelj-internet-protokola-skupina-tcpip-protokola, pristupljeno: 16. 08. 2021.

24. TCP protokol, dostupno na: http://mreze.layer-x.com/s040100-0.html, pristupljeno 16. 08. 2021.

25.Transmission Control Protocol, dostupno na: https ://en . wikipedia. org/ wiki /Transmission_Control_Protocol#Network_function, pristupljeno 17. 08. 2021.

26. Implementing Quality of Service Over Cisco MPLS VPNs, dostupno na: https://www.ciscopress.com/articles/article.asp?p=471096&seqNum=6, pristupljeno 08. 08. 2021.

27.Voice over IP: Protocols and Standards, dostupno na: https: // www. cse. wustl .edu/~jain/cis788-99/ftp/voip_protocols/, pristupljeno 08. 08. 2021.

28. RTP-RTCP-RTSP, Fakultet elektrotehnike u Tuzli, dostupno na: file: ///C :/ Skinuto %20Chrome/toaz.info-rtp-rtcp-rtsp-pr_cf7212 28e31922 c6339b07b9 677 1846a.pdf, pristupljeno 08. 08. 2021.

29. Umrežavanje sadržaja – Mreže za potporu strujanju višemedijskog sadržaja, dostupno na: https://www.fer.unizg.hr/_download/repository/US-2017_03.pdf, pristupljeno 09. 08. 2021.

30. RTP: A Transport Protocol for Real-Time Applications, dostupno na: https://tools.ietf.org/html/rfc3550, pristupljeno: 09. 08. 2021.

31. Simpson W. Video over IP, second edition. Oxford, UK: Elsevier Inc.; 2008.

32.RTP kontrolni protokol – RP – Eisenbahn, dostupno na: https://hr2.wiki/wiki/RTP_Control_Protocol#Protocol_functions, pristupljeno 23. 08. 2021.

33. RTP: A Transport Protocol for Real – Time Applications, dostupno na: https://datatracker.ietf.org/doc/html/rfc3550, pristupljeno 23. 08. 2021.

34.Network Protocols Handbook, 2nd Edition, (2004),str. 145., dostupno na: https://www.academia.edu/9960204/Second_Edition_Network_Protocols, pristupljeno 23. 08. 2021.

35. About Wireshark, dostupno na: https://www.wireshark.org/, pristupljeno 24. 08. 2021.

36. Analiza alata Wireshark, NCERT – PUBDOC – 2010 – 09 – 312, Carnet, Hrvatska akademska i istraživačka mreža, dostupno na: https://www.cis.hr/www.edicija/LinkedDocuments/NCERT-PUBDOC-2010-09-312.pdf?fbclid=IwA R0W62 nJgsNnU0140VOi-DbYvr05bJyYRVTjcCyCYTzk98BfjQeHSSB HW_s, pristupljeno 24. 08. 2021.

37. How do I ensure that BYE messages bypass a SIP proxy?, dostupno na: https://stackoverflow.com/questions/57630610/how-do-i-ensure-that-bye-messages-bypass-a-sip-proxy, pristupljeno 02. 10. 2021.

38."Jitsi.org – develop and deploy full-featured video conferencing", dostupno na: jitsi.org, pristupljeno 28. 11. 2021.

39. VoIP SerWeb, dostupno na: https://serweb.iptel.org/user/reg/index.php, pristupljeno 11. 09. 2021.

40. Notes on H.264 mapping, dostupno na: https://grouper.ieee.org/groups/1722/contributions/2015/h264Notes-bechtel-2015-05-11.pdf, pristupljeno 17. 01. 2022.

41. Implementing Quality of Service Over Cisco MPLS VPNs, dostupno na: https://www.ciscopress.com/articles/article.asp?p=471096&seqNum=6, pristupljeno 09. 03. 2022.

POPIS SLIKA