

PENANGGULANGAN PROTOKOL SIP YANG TIDAK MENJANGKAU CLIENT DI BELAKANG NAT MENGGUNAKAN SERVER ELASTIX

SIP protocol solving that doesn't reach client behind NAT using Elastix server

Nurshihab Idris¹, Asep Mulyana, S.T., MT.², Tengku Ahmad Riza, ST., MT.³

^{1,2,3}Prodi D3 Teknik Telekomunikasi, Fakultas Ilmu Terapan, Telkom University

¹skyartofgod@student.telkomuniversity.ac.id, ²asepmulyana@tass.telkomuniversity.ac.id,

³tengkuriza@tass.telkomuniversity.ac.id

Abstrak

SIP (*Session Initiated Protocol*) merupakan protokol aplikasi yang digunakan untuk *signaling* dalam pembentukan/pembubaran koneksi komunikasi data secara *real-time* (media) pada jaringan internet seperti VoIP (*Voice over IP*). SIP ini menggunakan protokol IP (*Internet Protocol*) untuk menentukan jalur-jalur yang akan digunakan pada jaringan. Namun, terdapat permasalahan pada protokol SIP yang menyertakan alamat IP pada *payloads* paket *signaling* yang dikirim, sehingga tidak dapat menjangkau *client* pada sebuah *Local Area Network* (LAN) yang berada di belakang *Network Address Translation* (NAT) saat dirutekan pada jaringan publik. Hal ini menyebabkan komunikasi VoIP tidak dapat dilaksanakan.

Untuk mengatasi permasalahan tersebut, pada proyek akhir ini dilakukan perancangan dan realisasi server Elastix dan softphone Linphone yang mendukung protokol STUN (*Session Traversal Utilities for NAT*) dimana pada server Elastix ditambahkan informasi TRUNK. STUN untuk mengikat pengguna dengan IP publik yang digunakan serta memetakannya dengan alamat IP privat. Disisi lain TRUNK berfungsi untuk mengetahui informasi mengenai IP publik server tetangga dimana pengguna lain berada.. Dengan cara ini, tiap pengguna dapat saling berkomunikasi melalui jaringan publik meskipun berada di belakang NAT.

Dari hasil pengujian *voice call* maupun *video call* didapatkan bahwa hasil rancangan dapat berfungsi sebagaimana mestinya dengan hasil pengukuran parameter QoS (*Quality of Service*) masing-masing untuk *voice call* : *delay* 19,97 ms, *jitter* 2,2 ms, *packet loss* 0% dengan nilai MOS 4,4. Sedangkan untuk *video call* didapat *delay* 19,97 ms, *jitter* 3,7 ms, dan *packet loss* 0% dengan nilai MOS 4,1.

Kata Kunci: VoIP, SIP, NAT, STUN, TRUNK.

Abstract

SIP (*Session Initiated Protocol*) is an application protocol used for *signaling* (call setup and tear down) of *real-time* media communication over internet such as VoIP (*Voice over IP*). This SIP uses an IP protocol (*Internet Protocol*) to determine the paths on the network. However, there are problems with the SIP protocol that includes the IP address in the *payloads* of the *signaling* package in which it cannot reach the client on a *Local Area Network* (LAN) behind of *Network Address Translation* (NAT) server. This causes VoIP communication cannot be realization.

To overcome these problems, in this final project designed and implemented the Elastix server with Linphone softphone as client that supported by the STUN (*Session Traversal Utilities for NAT*) protocol where TRUNK information is added to the Elastix server. STUN is used to map and bind the private with the public IP address. On the other hand TRUNK functions to find out information about the public IP of the neighboring server where other users are located. In this way, each user can communicate with each other through the public network even though it is behind NAT of the LAN.

From the results of *voice call* and *video call* testing, it was found that the system could function properly with the QoS (*Quality of Service*) parameter measurement for the *voice call*: 19.97 ms delay, 2.2 ms jitter, and 0% packet loss with MOS value 4.4. Whereas for *video calls* obtained 19.97 ms delay, 3.7 ms jitter, and 0% packet loss with MOS value 4.1.

Keywords : VoIP, SIP, NAT, STUN, TRUNK.

1. Pendahuluan

Protokol SIP (*Session Initiated Protocol*) merupakan protokol signalling yang digunakan untuk media pertukaran data secara *real-time* pada jaringan berbasis internet, seperti VoIP (*Voice over Internet Protocol*). Protokol ini menggunakan IP (*Internet Protocol*) untuk menentukan jalur-jalur yang akan digunakan oleh tiap pengguna agar dapat saling berkomunikasi. Data yang berisi jalur media tersebut terdapat di dalam SDP (*Session Initiated Protocol*) bagian dari *payload* untuk mendeskripsikan sesi multimedia [1]. IP yang terdapat pada

payload ini kemudian digunakan oleh *server* untuk merutekan kembali media seperti *audio* atau *video*. Hal ini menjadi masalah jika seorang pengguna berada di belakang NAT.

Ketika pengguna mengirim permintaan ke *server* dari jaringan lokal ke publik, NAT akan membiarkan paket tersebut keluar ke jaringan publik. NAT sebelumnya mengganti alamat dari header paket tersebut dengan alamat IP publik yang dapat dirutekan. Server kemudian menerima paket tersebut. Akan tetapi, ketika server mencoba menggunakan informasi dalam *payload* untuk mengirim paket kembali kepada pengguna, alamat jaringan lokal tersebut tidak dapat dirutekan.

Pembuatan protokol SIP oleh pembuat standar awalnya ditujukan untuk IPv6 (Internet Protocol version 6) yang berkapasitas alamat IP cukup besar dengan harapan setiap perangkat khususnya *user terminal* seperti komputer pribadi, IP Phone dsb memperoleh alamat IP *public* sehingga dapat berkomunikasi melalui jaringan publik (internet) tanpa bantuan NAT. Namun, pada prakteknya dalam masih banyak yang menggunakan alamat IP *privat* di lingkungan LAN karena mahalnya harga IP *public* lebih-lebih bagi yang masih menggunakan IPv4.

Berdasarkan permasalahan diatas, maka dibutuhkan suatu mekanisme pengikatan, pemetaan dan pendeskripsian pada IP yang digunakan oleh pengguna. Mekanisme ini berjalan pada sebuah IP-PBX (IP Private Branch Exchange) dan juga didukung oleh *softphone* yang dapat melakukan mekanisme tersebut. Mekanisme pengikatan dan pemetaan ini dinamakan protokol STUN (*Session Traversal Utilities for NAT*). Dan untuk pendeskripsian IP publik pada *server* pengguna dinamakan TRUNK. Dengan kedua hal tersebut, komunikasi antara pengguna yang berada dibelakang NAT dapat berlangsung.

Pada proyek akhir ini direalisasikan konfigurasi aringan seperti di atas yang dapat menganalisis permasalahan dan memberikan solusi pada jaringan VoIP yang masing-masing pengguna dan *server* berada di belakang NAT. Perancangan ini pun diharapkan mampu diimplementasikan pada jaringan sebenarnya di lapangan.

2. Dasar Teori

2.1 SIP (Session Initiated Protocol)

SIP atau session initiation protokol adalah sebuah layer aplikasi protokol yang digunakan untuk membuat, memodifikasi dan memutuskan sesi multimedia seperti panggilan telpon yang menggunakan internet. SIP juga dapat mengundang peserta ke sesi yang telah dibuat, seperti konferensi.

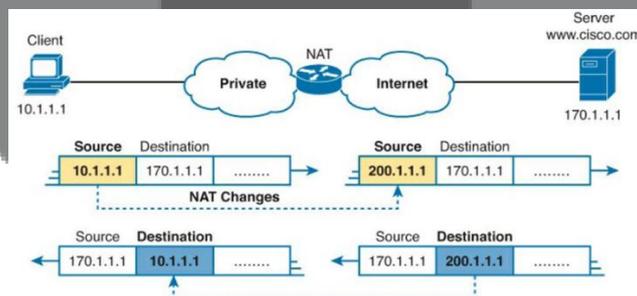
SIP mendukung lima aspek untuk membangun dan mengakhiri komunikasi:

- User Location: penentuan sistem akhir yang akan digunakan untuk komunikasi
- User Availability: penentuan kesediaan pihak yang dipanggil untuk terlibat dalam komunikasi
- User Capabilities: penentuan parameter media yang akan digunakan
- Session Setup: “ringing”, penetapan parameter sesi pada tiap pemanggil dan yang dipanggil
- Session Management: termasuk transfer dan penghentian sesi, memodifikasi parameter sesi dan layanan pemanggilan

2.2 NAT (Network Address Translation)

Network Address Translation adalah sebuah metode ketika alamat IP dipetakan dari suatu bidang ke alamat bidang yang lain. Ada banyak variasi dari address translation yang dibagi ke aplikasi yang berbeda.

2.2.1 Cara kerja NAT



Gambar 2. 2 Kinerja NAT [3]

NAT melakukan tugasnya dengan mengganti alamat source(asal) dari sebuah IP private menjadi alamat IP publik yang dapat dirutekan pada jaringan internet. Sebagai contoh gambar di atas menunjukkan seorang client dengan IP asal 10.1.1.1 yang akan mengakses server “www.cisco.com” yang menggunakan IP 170.1.1.1 pada jaringan public. Pada saat paket akan keluar, router NAT akan mengganti alamat private pada header paket dengan IP publik yaitu 200.1.1.1, selanjutnya server akan menerima bahwa IP 200.1.1.1 telah mengakses jaringannya dan akan mengembalikan paket tersebut ke alamat 200.1.1.1. Router yang menerima paket ini akan membaca tabel route dan meneruskan paket dari server ke client dengan membaca alamat private yang meminta data tersebut [3].

2.3 STUN(*Session Utilities for NAT*)

Session Traversal Utilities for NAT (STUN) adalah protokol yang berfungsi sebagai alat untuk protokol lain dalam menangani traversal Network Address Translation (NAT). STUN dapat digunakan oleh end-point untuk menentukan alamat IP dan port yang dialokasikan oleh NAT kepadanya. STUN juga dapat digunakan untuk memeriksa konektivitas antara dua titik akhir, dan mempertahankan binding NAT (pengikatan IP private ke publik) sebagai protokol “keep-alive”. STUN berfungsi dengan banyak tipe NAT yang ada dan tidak memerlukan perilaku khusus [5].

2.3.1 *Binding pada STUN*

STUN adalah protokol client-server yang mendukung dua jenis transaksi yaitu transaksi permintaan/tanggapan dimana client mengirim permintaan ke server dan server mengembalikan respon. Yang kedua adalah indikasi dimana client atau server tidak menghasilkan respon.

Permintaan Binding (pengikatan) dikirim dari client ke server STUN, yang mungkin telah melewati satu atau lebih NAT. Ketika pesan permintaan binding melewati NAT, NAT akan memodifikasi alamat sumber (yaitu, alamat IP dan port sumber) dari paket. Akibatnya, alamat transport sumber dari permintaan yang diterima oleh server menjadi alamat publik dan port alamat yang dibuat oleh NAT terdekat. Alamat ini disebut reflexive transport address (alamat transport refleksi). Server STUN menyalin alamat transport sumber kedalam atribut XOR-MAPPED-ADDRESS dalam respon STUN binding dan mengirim respon binding kembali ke client. Ketika paket ini kembali melalui NAT, NAT akan memodifikasi alamat transport tujuan dalam header IP, tetapi alamat transport dalam atribut XOR-MAPPED-ADDRESS dalam payload respon STUN tidak akan tersentuh. Dengan ini, client dapat mempelajari alamat transport refleksinya yang dialokasikan oleh NAT terluar terhadap server STUN [5].

2.4 Linphone

Linphone adalah open source SIP phone untuk panggilan suara dan video, tersedia di lingkungan ponsel dan desktop. Linphone di dalamnya memiliki pemisahan antara antarmuka (interface) pengguna dan mesin inti, memungkinkan untuk menciptakan berbagai jenis antarmuka pengguna di atas fungsi yang sama banyak lainnya.

2.5 Elastix

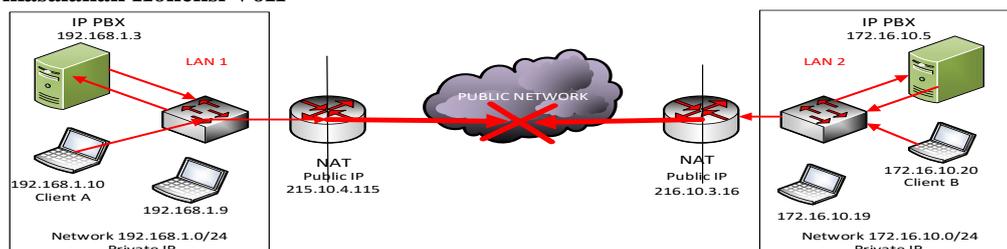
Elastix adalah opensource software yang membuat sebuah platform media komunikasi terpadu yang terdiri dari komponen atau modul berbagai teknologi media komunikasi yang lazim digunakan dewasa ini seperti: VoIP, video conference dll. Hampir semua modul dapat dikelola dan dikonfigurasi melalui antarmuka grafis, dimana Elastix mendukung fitur-fitur terdepan seperti voicemail, softphones, dan banyak lainnya [8].

2.6 Quality of Service

Quality of service (QoS) adalah deskripsi atau pengukuran kinerja keseluruhan layanan, seperti jaringan telepon, komputer atau layanan cloud computing, terutama kinerja yang dilihat oleh pengguna jaringan. Agar VoIP menjadi pengganti yang realistis untuk layanan telepon jaringan publik standar (PSTN), pelanggan harus menerima kualitas transmisi suara yang sama yang mereka terima dengan layanan telepon umum. Ini berarti transmisi suara berkualitas tinggi secara konsisten. Seperti aplikasi real-time lainnya, VoIP sangat sensitif terhadap delay. Agar transmisi VoIP dapat dimengerti oleh penerima, paket suara tidak boleh dibuang, terlalu lama tertunda, atau mengalami penundaan yang bervariasi (dikenal sebagai jitter). Dan juga menggunakan pendekatan matematis untuk menentukan Nilai MOS dan R-faktor dari jaringan yang dimodelkan dengan E-Model yang distandarkan kepada ITU-T G.107.

3. Analisis Perancangan dan Solusi Permasalahan

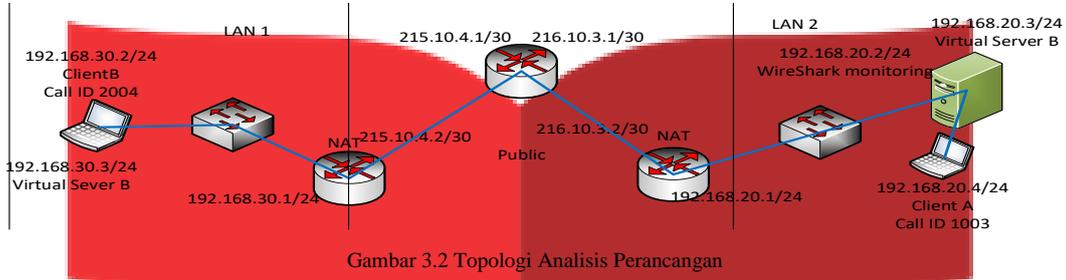
3.1 Permasalahan Koneksi VoIP



Gambar 3.1 Gambaran Permasalahan

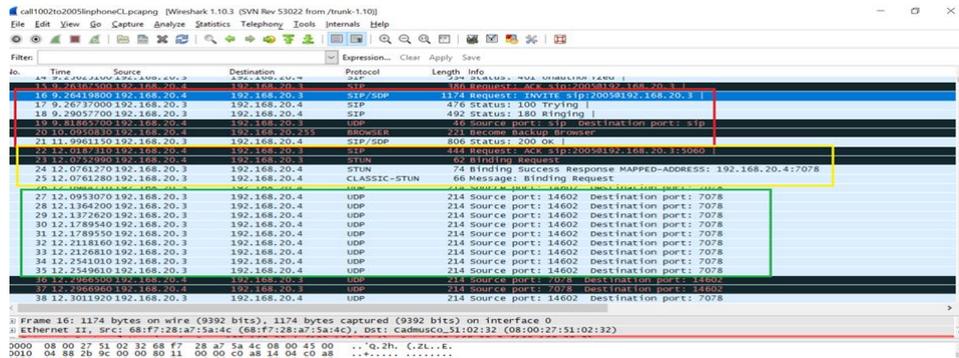
Pada gambar, terdapat dua buah LAN yang dipisahkan oleh sebuah publik network. Tiap-tiap LAN memiliki IP-PBX untuk menelpon rekan yang berada pada LAN yang sama. LAN1 memiliki network 192.168.1.0/24 dengan publik 214.10.4.115, sedangkan LAN2 memiliki network 172.16.10.0/24 dengan publik 216.10.3.16. Saat client dari salah satu LAN ingin berkomunikasi dengan client pada LAN lain, hal ini tidak dapat dilakukan karena tiap LAN berada di belakang NAT.

3.2 Analisis Perancangan dan Solusi Permasalahan



Gambar 3.2 Topologi Analisis Perancangan

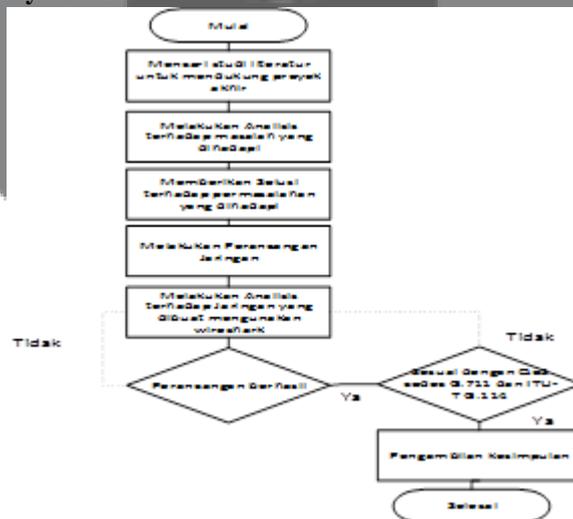
Solusi pada perancangan ini yaitu menggunakan softphone Linphone dan juga mekanisme TRUNK untuk dapat merutekan proses signalling keluar ke jaringan publik. Perancangan ini dimulai dari 1002 sebagai pemanggil dan 2005 sebagai penerima. Panggilan dapat dilakukan dan terjadi komunikasi antara client A dan B.



Gambar 3.3 Topologi Analisis Perancangan

Pada gambar, kotak merah berisikan pesan invite yang ditujukan ke client B dengan nomor panggilan 2005. Status yang diberikan server trying dan ringing, hingga status OK dan panggilan akan berlangsung. Terdapat mekanisme STUN(kotak kuning) yang dijalankan oleh client A untuk melakukan Binding request, server pun menerima dan melakukan MAPPED-ADDRESS untuk client A. Hal ini ditujukan untuk mengikat client A dengan IP publik miliknya dan memetakan lokasi dari client A. Sehingga seakan-akan client A memiliki IP yang sama dengan IP publiknya. Server mengirimkan paket-paket media menggunakan protokol UDP.

3.3 Proses Pengerjaan Proyek Akhir



Gambar 3.4 Proses Pengerjaan Proyek Akhir

Tahapan pengerjaan yaitu mengumpulkan data berupa referensi-referensi yang berkaitan dengan permasalahan jaringan SIP dan NAT. melakukan analisis permasalahan yang menyebabkan komunikasi pada client yang berada dibelakang NAT tidak dapat dilakukan. Memberikan solusi terhadap permasalahan yang terjadi berdasarkan analisis yang dilakukan. Membuat perancangan jaringan yang akan mendeskripsikan solusi yang dibuat. Melakukan analisis terhadap jaringan yang dibuat menggunakan software Wireshark dengan mengambil capture dari paket-paket yang terdapat pada jaringan, mencakup pula penggantian softphone pada tiap client yang mendukung solusi yang diberikan.

3.4 Skenario Quality of Service

Berikut adalah scenario pengukuran QoS menggunakan wireshark pada jaringan:

1. Mengukur delay sesuai dengan spesifikasi ITU G.114 yang merekomendasikan delay untuk satu kali end-to-end kurang dari 150 miliseconds(ms). Untuk panggilan internasional, penundaan satu arah hingga 300 ms dapat diterima [6].
2. Mengukur jitter, lebih jauh merupakan keterlambatan end-to-end, dan biasanya hanya efektif jika delay kurang dari 100 ms [6].
3. Mengukur paket loss yang harus sesuai dengan standar codec G.729. Dimana paket loss harus kurang dari 1% untuk menghindari adanya kesalahan yang dapat didengar. Idealnya, tidak ada paket yang hilang untuk VoIP [6]

4.

5. Pengujian

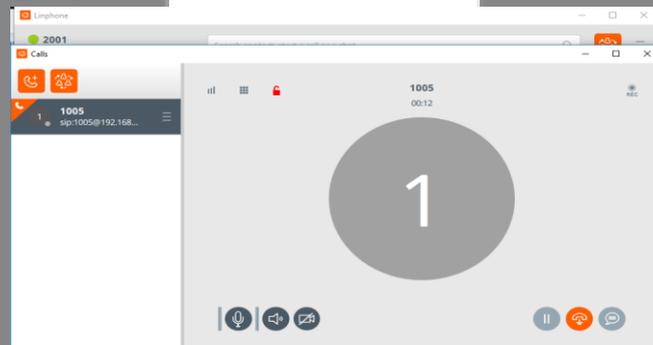
4.1 Hasil dari Perancangan yang dibuat

Perancangan ini menggunakan 2 buah server Elastix yang diinstal pada virtualbox. Tiap-tiap virtualbox terdapat pada PC dan Laptop. Terdapat 3 client yang menggunakan softphone Liphone untuk menjalin komunikasi yaitu smartphone, laptop dan PC. Masing-masing nomor panggilan 1005, 1002 dan 2001. Untuk angka 1XXX berada pada LAN2 dan 2XXX berada pada LAN1.

4.2 Pengujian Layanan

Layanan yang digunakan sebagai parameter pengujian yaitu suara atau (voice), video call, ring group dan keberhasilan melakukan panggilan terhadap client lain yang berada di belakang NAT. Berikut gambar dari pengujian yang dilakukan

4.2.1 Pengujian Layanan Voice



Gambar 4.2 Pengujian layanan Voice pada Laptop

4.2.2 Pengujian Layanan Video Call

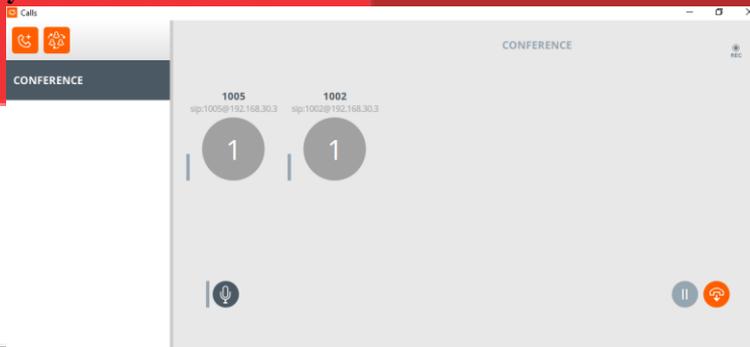


Gambar 4.4 Pengujian video Call pada Laptop



Gambar 4.5 Pengujian video Call pada Smartphone

4.2.3 Pengujian Layanan Conference Call



Gambar 4.5 Pengujian Layanan Conference Call

4.3 Pengujian QoS pada layanan VoIP

4.3.1 Analisis Delay, Packet Loss dan Jitter

Pengujian layanan untuk *packet loss* dan *jitter* dilihat menggunakan software wireshark sehingga tidak dilakukan perhitungan matematis. Untuk total *delay* tiap panggilan dilakukan perhitungan matematis dengan rumus sebagai berikut:

$$Delay\ rata - rata = \frac{total\ delay}{total\ paket\ yang\ diterima}$$

Berdasarkan rumus didapatkan hasil *delay* dari tiap-tiap panggilan. Dari hasil analisa, berikut tabel panggilan *voice* yang didapatkan:

Tabel 4. 1 Tabel panggilan voice

NO	Status	Total Paket	Total delay(s)	Paket Loss(%)	Delay(s)	Jitter(s)
1	Panggilan 1	783	15,64	0	0,01997	0,0038
2	Panggilan 2	863	17,24	0	0,01998	0,0039
3	Panggilan 3	791	15,79	0	0,01996	0,0018
4	Panggilan 4	817	16,31	0	0,01996	0,0038
5	Panggilan 5	775	15,48	0	0,01997	0,00066
6	Panggilan 6	782	15,62	0	0,01997	0,003
7	Panggilan 7	795	15,88	0	0,01997	0,003
8	Panggilan 8	815	16,28	0	0,01998	0,00038
9	Panggilan 9	781	15,6	0	0,01997	0,00074
10	Panggilan 10	798	15,94	0	0,01997	0,0013
Rata-rata				0	0,01997	0,0022

Berdasarkan table panggilan *voice* diatas maka total rata-rata *delay* yaitu 0,01997 atau sama dengan 19,97ms, total *paket loss* sama dengan 0%, dan total *jitter* sama dengan 0,0022 detik atau 2,2ms. Maka dapat dikatakan bahwa layanan *voice* yang diberikan bersifat baik dan layak. Berikut tabel panggilan video call

Tabel 4. 2 Tabel panggilan Video call

NO	Status	Total Paket	Total delay(s)	Paket Loss(%)	Delay(s)	Jitter(s)
1	Panggilan video 1	792	15,81	0	0,01996	0,00418
2	Panggilan video 2	789	15,76	0	0,01997	0,00407
3	Panggilan video 3	824	16,46	0	0,01998	0,00119
4	Panggilan video 4	781	15,59	0	0,01996	0,00418
5	Panggilan video 5	817	16,33	0	0,01999	0,00463
Rata-rata				0	0,01997	0,0037

Berdasarkan tabel panggilan *video call* diatas maka total rata-rata *delay* yaitu 0,01997 atau sama dengan 19,97ms, total *paket loss* sama dengan 0%, dan total *jitter* sama dengan 0,0037 detik atau 3,7ms.

4.3.2 Analisis Faktor R dan MOS (Mean Opinion Score)

Berikut hasil dari penilaian R-faktor dan MOS

$$\text{Untuk voice call} = 94,3 - [0,024(19,97) + 0,11(19,97 - 177,3)0(19,97 - 177,3)] - [0 + 30 \ln(1 + 15 (0))] = 93,83$$

$$\text{Untuk video call} = 94,3 - [0,024(19,97) + 0,11(19,97 - 177,3)0(19,97 - 177,3)] - [10 + 25,21 \ln(1 + 15 (0))] = 83,83$$

Perhitungan MOS dari hasil R factor yang didapatkan

$$\text{MOS voice call} = 1 + 0,035 (93,83) + 7 \times 10^{-6}(93,83)(93,83 - 60)(100 - 93,83) = 4,4$$

$$\text{MOS video call} = 1 + 0,035 (83,83) + 7 \times 10^{-6}(83,83)(83,83 - 60)(100 - 83,83) = 4,1$$

6. Penutup

5.1 Kesimpulan

Proyek akhir ini dapat disimpulkan sebagai berikut

1. Permasalahan yang muncul akibat dari client yang menggunakan layanan protokol SIP di belakang NAT yaitu, tidak bisanya melakukan panggilan ke client lain yang juga berada di belakang NAT. Hal ini diakibatkan oleh protokol SIP yang menyisipkan IP private dari client pada payload paket signalling. IP ini dimaksudkan untuk mengetahui identitas dari perangkat pengguna dan juga menegosiasikan jalur untuk mengirimkan media berupa suara atau video.
2. Solusi yang diberikan yaitu menggunakan mekanisme STUN(Session Traversal Utilities for NAT) dan TRUNK. Mekanisme ini terdapat pada IP-PBX dan didukung oleh softphone client. Mekanisme STUN berfungsi untuk mengikat IP private dari tiap client dan memetakan jalur yang akan digunakan oleh tiap client agar dapat saling berkomunikasi. Hal ini harus dipadukan dengan TRUNK yang berisi informasi mengenai server IP-PBX tetangga(client yang dituju) agar mekanisme pemetaan STUN dapat berjalan.
3. Hasil pengujian QoS menunjukkan bahwa jaringan yang dibuat sesuai dengan standar QoS dan MOS. Hal ini terlihat dari pengujian voice call dimana rata-rata delay dari pengujian yaitu sebesar 0.01997 detik atau sama dengan 19,97 ms, untuk jitter sebesar 0.0022 s atau 2,2 ms, packet loss sebesar 0% dengan nilai MOS sebesar 4,4. Untuk video call rata-rata delay dari pengujian yaitu sebesar 0.01997 detik atau sama dengan 19,97 ms, untuk jitter sebesar 0.0037 s atau 3,7 ms, packet loss sebesar 0% dengan nilai MOS sebesar 4,1.

5.2 Saran

Untuk penelitian lebi lanjut disarankan untuk dilakukan uji lapangan secara real menggunakan aamat IP privat dan publik secara konkrit

Daftar Pustaka

1. J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley dan E. Schooler, SIP: Session Initiation Protocol, The Internet Society, 2002.
2. B. Fedat, "The NAT and SIP Problem," N2Net, Inc, 30 Juni 2014. [Online]. Available: <https://www.n2net.net/agent-blog/sip-nat>. [Diakses 27 July 2018].

3. CCNA Exam, "Network Address Translation," 28 February 2014. [Online]. Available: <https://4anm.wordpress.com/2014/02/28/network-address-translation/>. [Diakses 27 July 2018].
4. M. Handley, V. Jacobson dan C. Perkins, SIP: Session Initiation Protocol, The Internet Society, 2006.
5. J. Rosenberg, R. Mahy, P. Matthews dan D. Wing, Session Traversal Utilities for NAT (STUN), Internet Enggining Task Force , 2008.
6. Cisco, "Quality of Service for Voice over IP," 13 April 2001. [Online]. Available: https://www.cisco.com/c/en/us/td/docs/ios/solutions_docs/qos_solutions/QoSVoIP/QoSVoIP.html. [Diakses 27 July 2018].
7. Mandalaya.com, "Pengertian Mikrotik," Mandalaya.com, 26 November 2014. [Online]. Available: <http://www.mandalamaya.com/pengertian-mikrotik/>. [Diakses 26 July 2018].
8. E. Zaida, "ELASTIX," 9 December 2012. [Online]. Available: <https://efrizalzaida.wordpress.com/2012/12/09/elastic/>. [Diakses 28 July 2018].
9. Y. Yuniati, H. Fitriawan dan D. F. Jaya Patih, "ANALISA PERANCANGAN SERVER VOIP (VOICE INTERNET PROTOCOL) DENGAN OPENSOURCE ASTERISK DAN VPN (VIRTUAL PRIVATE NETWORK) SEBAGAI PENGAMAN JARINGAN ANTAR CLIENT," *Teknologi dan Industri*, vol. 12, p. 113, 2014.
10. S. Avakian, "Linphone," 23 February 2017. [Online]. Available: <http://www.linphone.org/user-guide.html>. [Diakses 31 July 2018].
11. M. D. Falih, Belajar VOIP dengan Elastix, Bandung, 2018
12. Y. Rekhter, R. G. Moskowitz, D. Karrenberg, G. Jan de Groot dan E. Lear, Address Allocation for Private Internets, Internet Community, 1996.
13. A. S. H. TIMORYANSYAH, IMPLEMENTASI VOIP SERVER DENGAN MENGGUNAKAN MINI PC, Bandung: Telkom University, 2015.

