

Jurnal ENERGI ELEKTRIK

Volume 2 Nomor 2

Oktober 2013

Pengaruh Jumlah Sudu terhadap Prediksi Performansi Turbin Angin Sumbu Vertikal Jenis Darrieus dengan NACA-0015

Zulfikar

Filter Aktif Menggunakan Deteksi Tegangan dan Deteksi Arus Untuk Mengkompensasi Harmonisa Pada Sistem Distribusi Radial

Asran

Understanding TCP Performance Using Drop -Tail Queuing Mechanism

Muhammad Ikhwanus

Studi Sistem Logika Fuzzy Sederhana Dengan Software Matlab

Fajriana

Pengujian Kekuatan Nilai Hash Password Yang Dienkripsi Dengan Menggunakan Algoritama MD5

Hendrawaty

Azhar

Mulyadi

Optimasi Waktu Pada Replikasi Dan Distribusi Database Dengan Metode Round Robin

Rizalul Akram

Replikasi Standby Database Menggunakan Metode Incremental Backup

Defry Hamdhana

Sayed Fachrurrazi

Penerbit :

Jurusan Teknik Elektro
Fakultas Teknik
Universitas Malikussaleh

Jl. UNIMAL Cot Tgk Nie Releut
Kecamatan Muara Dua
Aceh Utara Po Box 141



Jurusan Teknik Elektro
Fakultas Teknik
Universitas Malikussaleh

JURNAL
ENERGI ELEKTRIK

ISSN : 2303 - 1360

Pengantar

Jurnal Energi Elektrik mempublikasikan hasil penelitian ilmiah dibidang energi, baik penelitian dasar, perancangan, pengembangan dan studi mengenai pengembangan bidang energi. Jurnal energi ini akan terbit secara berkala 2 kali dalam satu tahun (Oktober dan April)

DEWAN REDAKSI

Penanggung Jawab:
Ketua Jurusan Teknik Elektro

Pemimpin Redaksi
Andik Bintoro, S.T., M.Eng.

Anggota Redaksi
Munirul Ula, S.T., M.Eng.
Maryana, S.Si., M.Si.

Penyunting Ahli
Prof. Dr. Ir. T. Haryono, M.Sc.
Ir. Tumiran, M.Eng, Ph.D
Dr. Ir. Rizal Munadi, M.M., M.T.
Dr. Azhari, S.T., M.Sc.
Muhammad Ikhwanus, S.T., M.Eng.
Ezwarsyah, S.T., M.T.

Redaksi Pelaksana
Dahlia, A. Ma
Sufriani, S.P.
Abdul Hadi, S.T.

Penerbit
Jurusan Teknik Elektro
Fakultas Teknik
Universitas Malikussaleh
Jl. UNIMAL Cot Tgk Nie Releut
Kecamatan Muara Dua
Aceh Utara Po Box 141

DAFTAR ISI

Pengaruh Jumlah Sudu terhadap Prediksi Performansi Turbin Angin Sumbu Vertikal Jenis Darrieus dengan NACA-0015 <i>Zulfikar</i>	1
Filter Aktif Menggunakan Deteksi Tegangan dan Deteksi Arus Untuk Mengkompensasi Harmonisa Pada Sistem Distribusi Radial <i>Asran</i>	7
Karakteristik TCP Performance Using Drop -Tail Queuing Mechanism <i>Muhammad Ikhwanus</i>	13
Studi Sistem Logika Fuzzy Sederhana Dengan Software Matlab <i>Fajriana</i>	17
Pengujian Kekuatan Nilai Hash Password Yang Dienkripsi Dengan Menggunakan Algoritama MD5 <i>Hendrawaty</i> <i>Azhar</i> <i>Mulyadi</i>	21
Optimasi Waktu Pada Replikasi Dan Distribusi Database Dengan Metode Round Robin <i>Rizalul Akram</i>	28
Replikasi Standby Database Menggunakan Metode Incremental Backup <i>Defry Hamdhana</i> <i>Sayed Fachrurrazi</i>	36

Pengaruh Jumlah Sudu terhadap Prediksi Performansi Turbin Angin Sumbu Vertikal Jenis Darrieus dengan NACA-0015

Zulfikar

*Jurusan Teknik Mesin UNIMAL
Fakultas Teknik*

Abstrak— Pemanfaatan energi angin merupakan solusi yang tepat dalam pemenuhan energi di Indonesia. Turbin angin merupakan komponen utama dalam suatu sisten konversi energi angin. Turbin angin sumbu vertikal memiliki keunggulan yaitu dapat beroperasi pada kecepatan angin rendah, menangkap angin dari segala arah, konstruksi yang sederhana dan ekonomis. Penelitian ini bertujuan mengkaji pengaruh jumlah sudu terhadap kinerja turbin angin Darrieus dengan pendekatan Computational Fluid Dynamics 2-D. Model turbin telah dibangun dengan diameter 1030 mm dan airfoil Naca 0015 dengan chord 85,8 mm. Hasil menunjukkan bahwa torsi yang dibangkitkan dipengaruhi oleh posisi rotor dan berfruktusi dalam satu rotasi. Torsi juga dipengaruhi oleh jumlah sudu dalam rotor. Kestabilan dan torsi terbaik terjadi pada rotor dengan 5 sudu.

Keywords— Turbin Angin Sumbu Vertikal, Computational Fluid Dynamics, Torsi.

I. PENDAHULUAN

Kebutuhan energi di dunia sampai saat ini masih bergantung dengan sumber-sumber energi yang tidak dapat diperbaharui yaitu sumber energi berupa fosil-fosil alam yang akan habis. Energi-energi tersebut berupa minyak bumi, batu bara, gas alam dan lainnya. Penggunaan energi tidak dapat diperbaharui dapat menimbulkan terjadinya kenaikan jumlah karbon dioksida pada lapisan atmosfer yang menyebabkan pemanasan global. Alam menyediakan begitu banyak sumber energi diantaranya adalah sumber energi terbarukan yang dapat dimanfaatkan secara berkala dan ramah lingkungan, seperti energi air, energi matahari, energi panas bumi dan energi angin. Oleh sebab itu pemanfaatan energi terbarukan kini banyak diteliti dan dikembangkan.

Pemanfaatan energi angin merupakan salah satu solusi yang dapat diterapkan di Indonesia. Penggunaan turbin angin yang dapat berfungsi pada kecepatan angin rendah adalah solusi yang dapat diterapkan. Turbin angin sumbu vertikal merupakan turbin yang berfungsi pada kecepatan angin rendah dan mampu menangkap angin dari segala arah. Ada beberapa faktor yang mempengaruhi kinerja turbin angin sumbu vertikal tersebut, diantaranya jumlah sudu yang digunakan pada konstruksi turbin angin tersebut. Penelitian ini bertujuan mengkaji pengaruh jumlah sudu

terhadap prediksi performansi turbin angina Darrieus-H airfoili Naca 0015 dengan pendekatan CFD.

II. TINJAUAN PUSTAKA

Castelli, M.R [1], telah mengkaji pengaruh jumlah sudu lurus turbin angin sumbu vertical Darrieus menggunakan CFD-2D. Airfoil sudu Naca 0025 dengan chord 85,8 mm, diameter rotor 1030 mm. Jumlah sudu divariasikan 3, 4 dan 5 pada kecepatan angin 9 m/s. Hasil menunjukkan bahwa turbin dengan rotor 3 sudu memiliki kinerja lebih baik dibandingkan dengan rotor 4 sudu dan rotor 5 sudu.

Irigoien, J.G. [2], telah melakukan desain dan analisis airfoil sudu turbin angin sumbu vertical dengan CFD. Airfoil yang digunakan adalah Naca 0012, Naca 0015 dan Naca 0020 dengan jumlah sudu 3 yang diuji dengan model 2 dimensi dan 3 dimensi. Hasil analisis menunjukkan bahwa airfoil Naca 0015 memiliki nilai torsi lebih baik dibandingkan dengan airfoil Naca 0012 dan Naca 0020. Perbandingan pengujian model 2 dimensi dan 3 dimensi tidak terdapat perbedaan yang signifikan sehingga pengujian secara 2 dimensi sudah dapat digunakan sebagai data awal penelitian selanjutnya.

Deshpande, dkk, [3] telah memprediksi dan analisis aliran pada turbin angin sumbu vertical jenis giromill 3 sudu dengan pendekatan numerik pada kecepatan 3 m/s dan 5 m/s. Airfoil sudu yang dimodelkan berbeda soliditas (0,2 dan 0,4) yang simetris dan tidak simetris. Airfoil sudu yang digunakan adalah Naca 0018, Naca 0015 dan S 1210. Rotor dengan sudu Naca 0018 dan S1210 menunjukkan hasil menyerupai analisis teoritikal pada kecepatan angin 3 m/s dan 5 m/s. Efisiensi maksimum dicapai pada *tip speed ratio* 3. Koefisien power tertinggi 0,4 untuk Naca 0018 pada kecepatan 5 m/s. Untuk kasus soliditas 0,4, airfoil Naca 0015 menunjukkan kinerja yang baik pada kecepatan angin 5 m/s tetapi gagal membangkitkan tenaga pada kecepatan 3 m/s. Koefisien tenaga tertinggi dengan Naca 0015 adalah 0,42 pada *tip speed ratio* 2. Airfoil S1210 menghasilkan daya yang relative rendah tetapi kecenderungan koefisien tenaga dan *tip speed ratio* menyerupai kasus sebelumnya.

Seng Q dkk., [4] telah melakukan analisis numerik terhadap hidrodinamik turbin pasang surut sumbu vertical. Pemodelan dua dimensi dan simulasi aliran tidak tunak melalui sudu turbin dengan menggunakan ANSYS CFX dengan Reynolds-Averaged Navier-Stokes (RANS).

Simulasi telah dilakukan dengan pitch tetap dan berubah menggunakan Shear Stress Transport Turbulence (SST dengan parameter hidrodinamik yang ditinjau adalah torsi T, Momoen, koefisien performansi dan koefisien torsi. Pemodelan dan meshing rotor turbin menggunakan ICEM-CFD. Mesh bergerak telah digunakan untuk turbin dengan pitch variable. Hasil yang diperoleh divalidasi dengan hasil analitik dari Edinburgh Design Ltd. Hasil menunjukkan bahwa turbin dengan pitch variable mencapai koefisien performansi maksimum dan tenaga puncak pada putaran dan tsr rendah.

Cheng [5] telah mengkaji aerodinamika rotor H dengan dua pendekatan yang berbeda yaitu metode direct numerical simulation (DNS) dan model Reynolds averaged Navier-Stoke (RANS). Kajian DNS diselesaikan dengan persamaan Navier-Stokes inkompresibel paket CFD OVERTURE dengan teknik overlapping grid bergerak untuk gerakan rotasi turbin angin. Simulasi RANS menggunakan paket CFD ANSYS-Fluent dengan sliding mesh digunakan untuk menganalisis interaksi antara komponen stasioner dan yang berotasi. Hasil menunjukkan bahwa DNS tidak tepat memprediksi koefisien tenaga. Simulasi RANS menunjukkan hasil yang mirip dengan data eksperimental dan berdaya guna serta dapat diandalkan untuk mempelajari aerodinamika turbin angin. Turbin dengan NACA 0015 koefisien tenaga mencapai 0,16 pada tip speed ratio 2,5 dengan kecepatan angin 3,9 m/s. Sedangkan untuk turbin dengan NACA 002, koefisien tenaga maksimum 0,21 pada tsr 2,5 dengan kecepatan angin yang sama.

Hameed M.s., dkk [6] telah mengkaji desain aerodinamika sudu secara analitik dan CFD pada unit turbin angin vertikal skala kecil. Parameter desain dan dimensi sudu diperlukan untuk memperoleh daya keluaran dan analisis model dikembangkan untuk mengevaluasi gaya aerodinamika seperti lift dan drag pada permukaan sudu. Gaya sangat berguna untuk mengevaluasi integritas structural sudu turbin angin sumbu vertikal yang diperoleh dengan simulasi CFD menggunakan perangkat lunak komersial ANSYS 13,0. Model CFD dikembangkan pada sudut pitch berbeda hingga penuh 360° dimana gaya aerodinamika dievaluasi, dibandingkan dengan nilai analitik pada lokasi yang sama. Persentase error antara hasil numeric dan analitik untuk gaya lift, drag dan tangensial secara berturut-turut adalah 1,07; 9,46; dan 4,2.

Gupta R dan Biswas A [7] telah mengkaji performansi turbin angin Darrieus-H dengan tiga sudu lurus dan dipelintir 30° pada ujung sudu dengan simulasi CFD-2D pada kondisi tunak menggunakan perangkat lunak Fluent 6.0. Simulasi numeric menyelesaikan persamaan konservasi massa dan momentum dengan menggunakan unstructured-grid finite volume dan moving meshing dengan model turbulensi standard k-ε. Hasil menunjukkan nilai koefisien tenaga meningkat dengan penambahan tip speed ratio hingga mencapai nilai maksimum pada tip speed ratio tertentu dan kemudian menurun dengan penambahan tip speed ratio. Tekanan dan kecepatan menurun dari sisi dari upstream hingga sisi downstream turbin. Tekanan dinamik pada sisi upstream menjadi sangat tinggi sebelum sudu pada sudut posisi 355°. Kondisi ini menjadikan performansi

turbin dengan sudu dipelintir meningkat karena gradien kecepatan yang besar antara sisi upstream dan downstream turbin.

Roy S dan Biswas A [8] telah menganalisis CFD terhadap bentuk airfoil rotor Darrieus-H dengan tiga sudu menggunakan Fluent 6.2. Model dua dimensi telah dibangun dengan menggunakan Gambit. Kecepatan masuk dan kecepatan putar diperoleh dari hasil pengujian di dalam terowongan angin sub-sonik. Koefisien power dihitung pada perbandingan tinggi terhadap diameter (H/D) berbeda, yaitu : 0,85; 1,0; 1,33; 1,72 dan 2,0. Koefisien power tertinggi diperoleh pada rasio H/D 1 dengan deviasi antara hasil CFD dan eksperimental $\pm 1,5\%$.

Sabaiefard P dkk, [9] telah melakukan kajian komputasional dan eksperimental pengaruh beberapa parameter desain terhadap aerodinamika dan performansi turbin angin sumbu vertikal, yaitu jumlah sudu, jenis airfoil dan soliditas turbin. Simulasi transien dengan model turbulensi K-ε dan CFD dengan kapabilitas model multiple reference frame (MRF). Hasil optimum pada TSR dengan koefisien power maksimum 0,36 untuk hasil simulasi CFD dan 0,32 hasil pengujian eksperimental.

Gosselin R dkk, [10] telah mempelajari pengaruh soliditas, jumlah sudu, bilangan Reynolds, sudut pitch sudu (tetap dan berubah) dan ketebalan sudu terhadap efisiensi aerodinamik turbin sumbu vertikal dengan model k-ω Shear Stress Transport Turbulence (SST) Reynolds-Averaged Navier-Stokes (RANS). Hasil diperoleh soliditas optimal berdasarkan radius adalah 0,2 tetapi soliditas lebih tinggi dapat meningkatkan performansi dengan pengaturan pitch. Variabel pitch memungkinkan peningkatan 27% efisiensi mendekati batasan Betz untuk prediksi 2 dimensi. Untuk 3 dimensi, rasio aspek sudu kecil (AR=7) efisiensi relative berkurang 60% dibandingkan prediksi 2 dimensi. Sudu yang panjang dalam simulasi 3 dimensi menunjukkan peningkatan efisiensi sangat besar.

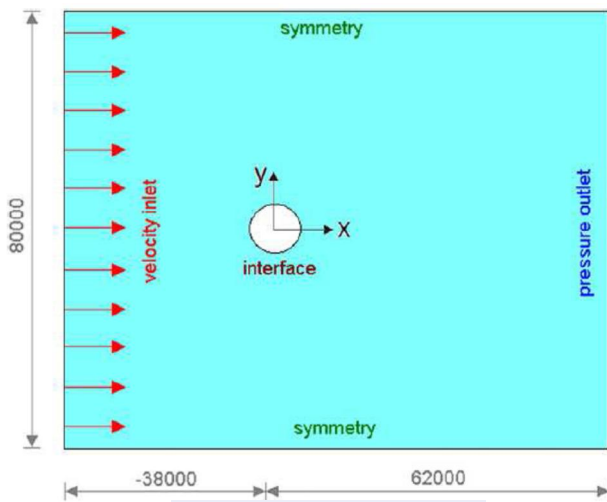
III. METODOLOGI PENELITIAN

A. Desain Model CFD

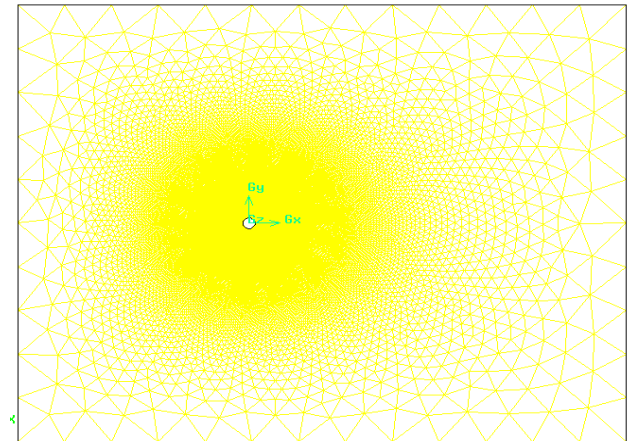
Medan alir CFD yang diuji memiliki lebar 80000 mm dan panjang 100000 mm seperti diperlihatkan pada gambar 1. Medan alir rotor dibuat dengan sub-grid tersendiri dan ditempatkan pada medan alir bidang uji dengan jarak 38000 mm dari sisi masuk aliran. Geometri rotor memiliki diameter 1030 mm dengan jumlah sudu divariasikan yaitu: 3, 4 dan 5. Sudu menggunakan Naca 0015 dengan *chord* 85,8 mm. Rotor dimodelkan dengan domain sub-grid tersendiri dengan spesifikasi diameter interface (domain rotor) 2000 mm, diameter interior (domain sudu) 400 mm seperti diperlihatkan pada gambar 2.

Deskripsi medan alir dilakukan baik pada bidang uji maupun bidang rotor seperti diperlihatkan pada gambar 3 dan 4.

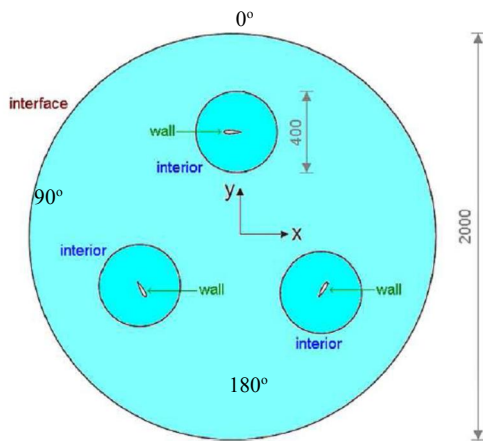
Model viskositas yang digunakan adalah k-ε (2 eqn) standard dan kondisi tunak. Tahapan



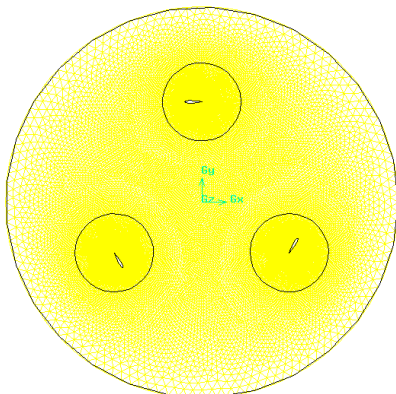
Gambar 1 Medan alir bidang uji



Gambar 4. Deskritasi bidang alir bidang uji



Gambar. 2 Medan alir rotor



Gambar 3. Deskritasi bidang alir rotor

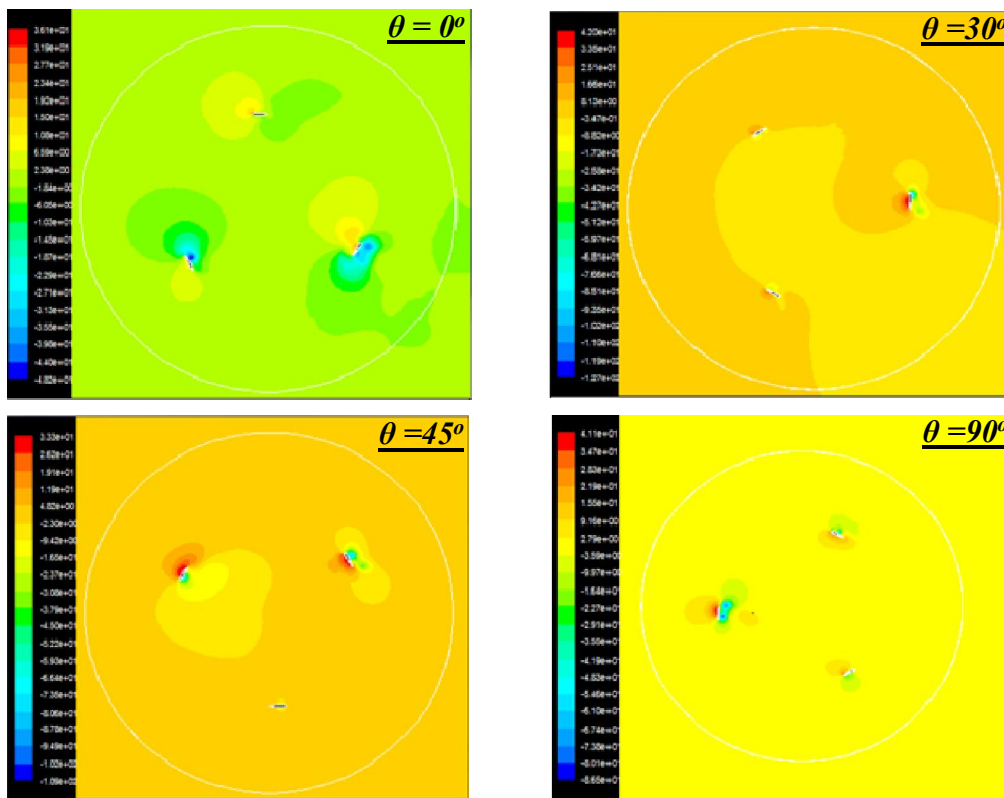
IV. HASIL DAN PEMBAHASAN

Simulasi aliran fluida dengan pendekatan CFD telah dilakukan untuk mempelajari pengaruh jumlah sudu terhadap prestasi dan pola aliran yang melalui sudu turbin angin Darrius jenis H dengan profil airfoil Naca 0015 dengan kondisi : kecepatan angin $U = 5 \text{ m/s}$, densitas $\rho = 1.1774 \text{ kg/m}^3$, viskositas dinamik $\mu = 1.983 \times 10^{-5} \text{ kg/m.s}$, dengan diameter rotor 1030 mm, chord 85,8 mm dan simulasi 2 dimensi.

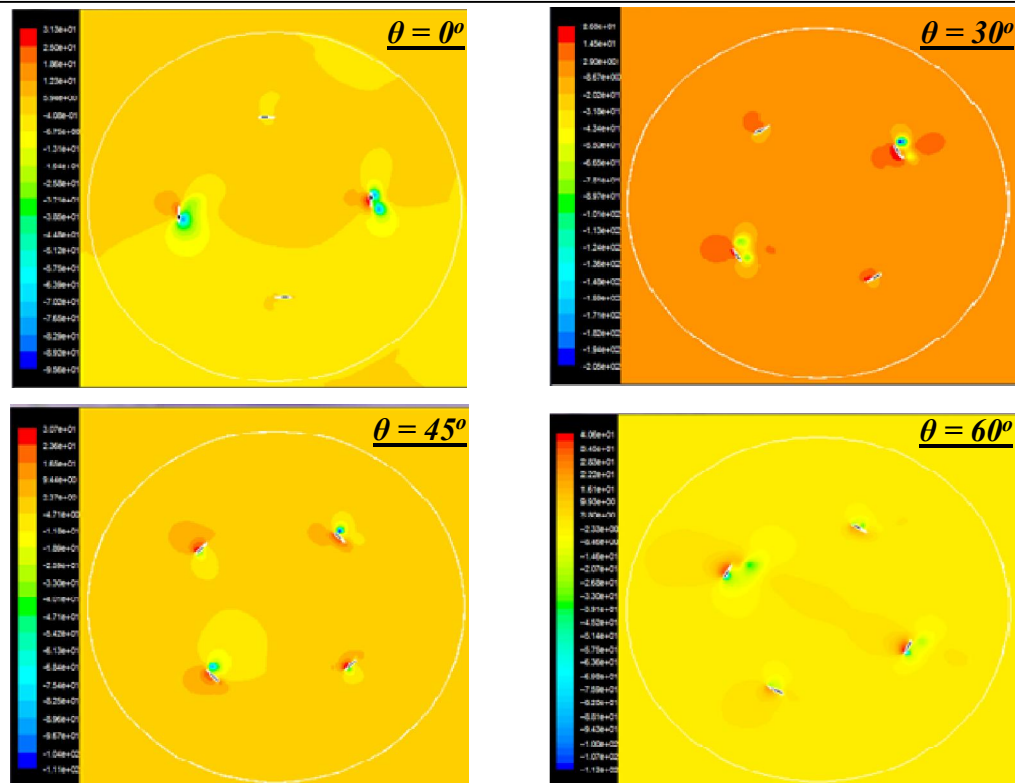
A. Analisis Bidang Kontur Tekanan

Hasil medan tekanan dalam rotor untuk sudu 3, 4 dan 5 ditunjukkan pada gambar 5, 6 dan 7. Medan tekanan dalam rotor menunjukkan tidak ada keseragaman.

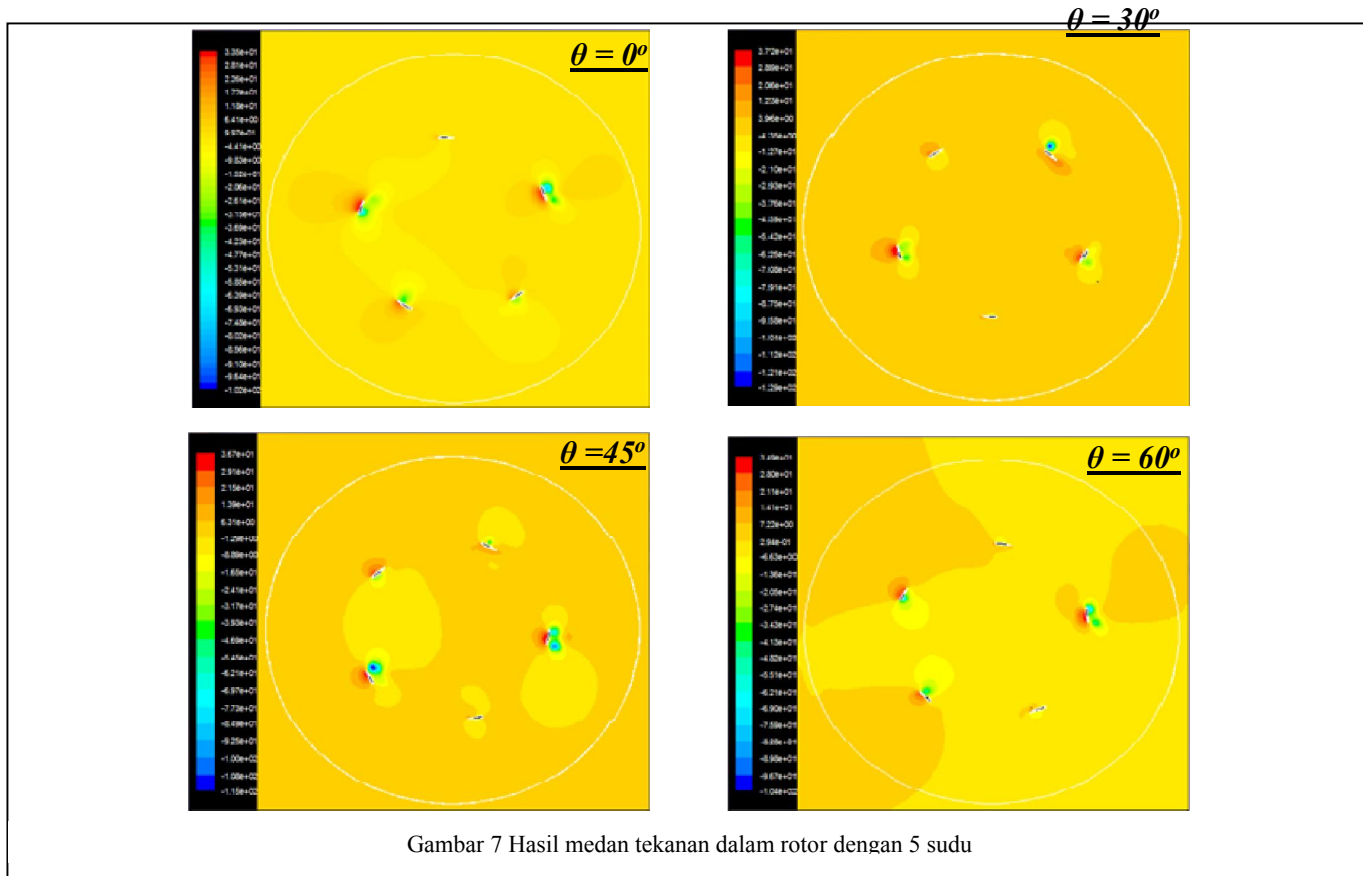
Simulasi rotor dengan jumlah sudu 3, 4 dan 5 telah dilakukan pada beberapa posisi sudut. Rotor dengan jumlah sudu tiga telah disimulasikan pada sudut 0° , 30° , 60° dan 90° seperti diperlihatkan pada gambar 5. Rotor dengan jumlah sudu empat dan lima telah disimulasikan pada sudut 0° , 30° , 45° dan 60° seperti diperlihatkan pada gambar 6 dan 7.



Gambar 5 Hasil medan tekanan dalam rotor dengan tiga sudu

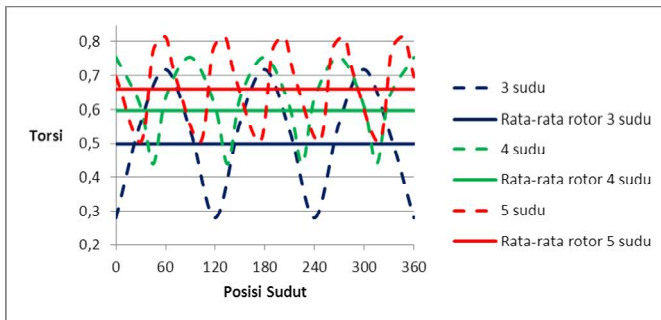


Gambar. 6 Hasil medan tekanan dalam rotor dengan empat sudu

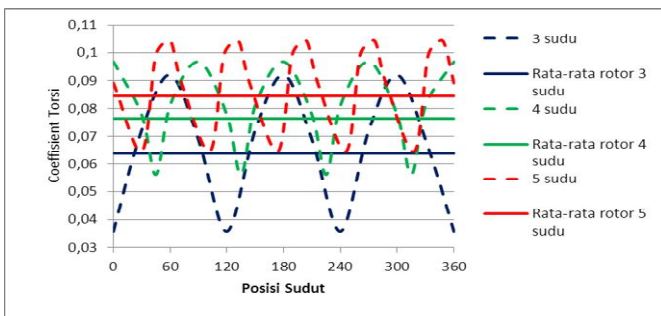


B. Torsi dan Koefisien Torsi

Torsi dan koefisien torsi yang dihasilkan dari simulasi rotor 3 sudu, 4 sudu dan 5 sudu diperlihatkan pada gambar 8 dan 9.



Gambar 8. Hasil torsi untuk rotor dengan 3, 4 dan 5 sudu.



Gambar 9. Hasil koefisien torsi untuk rotor dengan 3, 4 dan 5 sudu.

Gambar 8 dan 9 memperlihatkan torsi dan koefisien torsi yang dihasilkan berfluktuasi pada saat rotor berotasi pada porosnya. Rentang fluktuasi yang besar akan sangat berdampak pada ketidakseimbangan rotor. Rentang fluktuasi yang kecil terjadi pada rotor dengan empat dan lima sudu dibandingkan dengan rotor dengan tiga sudu. Rotor dengan 4 sudu dan 5 sudu lebih stabil dibandingkan dengan rotor dengan tiga sudu. Rotor dengan empat sudu, torsi dan koefisien torsi tertinggi terjadi posisi rotor 0°, yaitu sebesar 0,755295 Nm dan 0,096747, sedangkan nilai terendah pada posisi rotor 45° yaitu : 0,43838552 dan 0,05615. Rotor dengan lima sudu, torsi dan koefisien torsi tertinggi terjadi posisi rotor 0°, yaitu sebesar 0,81336807Nm dan 0,104186, sedangkan nilai terendah pada posisi rotor 45° yaitu : 0,50518697 dan 0,06471. sedangkan rotor dengan tiga sudu memiliki rentang fluktuasi yang besar. Torsi dan koefisien torsi tertinggi dicapai pada posisi rotor telah berputar 60° ke arah angin datang yaitu sebesar 0,718 Nm dan 0,09199. Sedangkan torsi dan koefisien torsi terendah pada saat posisi rotor 0°, yaitu 0,2797 Nm dan 0,035835.

Gambar 8 dan 9 juga memperlihatkan bahwa rotor dengan 5 sudu memiliki nilai torsi rata-rata yang lebih tinggi dibandingkan rotor dengan 4 sudu dan 3 sudu.

V. PENUTUP

Simulasi CFD untuk mempelajari pengaruh sudu terhadap prestasi turbin Darrius jenis H telah berhasil dilakukan dengan kesimpulan sebagai berikut :

1. Perubahan posisi sudut putar turbin berpengaruh terhadap nilai torsi yang dihasilkan.
2. Tekanan aliran bagian upstream rotor lebih tinggi dibandingkan bagian downstream.
3. Jumlah sudu pada turbin vertikal sangat mempengaruhi prestasi dari turbin Darrius jenis H.
4. Rotor turbin dengan jumlah 5 sudu menggunakan airfoil Naca 0015 menunjukkan prestasi yang lebih baik dibandingkan rotor dengan 4 sudu dan 3 sudu.

UCAPAN TERIMA KASIH

Terima kasih disampaikan kepada Chaidir yang telah meluangkan waktu membantu pelaksanaan penelitian ini.

VI. REFERENSI

- [1] Castelli M.R., S. De Betta dan Benini E., "Effect of Blade Number on a Straight-Bladed Vertical-Axis Darrius Wind Turbine", *Int. Jour. Of Mechanical and Aerospace Engineering*, Vol. 6, 2012, pp. 69-75.
- [2] Iriso, J.G dan Ursua, A. "Analysis and design of a vertical axis wind turbine". *Journal Escuela técnica superior de ingenieros Industriales de telecomunicación*. 2011
- [3] P. Deshpande dan X. Li, " Numerical Study of Giromill-Type Wind Turbines with Symmetrical and Non-symmetrical Airfoils", *European International Journal of Science and technology*, vol. 2 No.8 October 2013.
- [4] Sheng Q, Khalid S.S, Xiong Z., Sahib G dan Zhang L., "CFD Simulation of Fixed and Variable Pitch Vertical Axis Tidal Turbine", *J. Marine Sci. Appl.* (2013) vol. 12 pp. 185-192.
- [5] Chen Y, " Numerical Simulation of the Aerodynamic Performance of An H-Rotor", Thesis Department of Mechanical Engineering, University of Louisville, Kentucky. August 2011.
- [6] Hameed M.S., dan Shahid F., "Evaluation of Aerodynamic Forces over a Vertical Axis Wind Turbine Blade through CFD analysis", *Journal Applied Mechanical Engineering*, 2012, vol. 1, Issue 5, pp.1-6.
- [7] Gupta R., dan Biswas A., "CFD Analysis of a Three-bladed Straight Chord H-Darrius Turbine with 30 Degree Twist at Blade Tip" *Proceeding of ASE 2011* 5th International Conference on Energy Sustainability , August 7-10, Washington, DC, USA.
- [8] Roy S, dan Biswas A., "CFD Analysis of an Airfoil Shaped Three Bladed H-Darrius Rotor Made From Fibreglass Reinforced Plastic (FRP)", *Proceeding of the 37th National & 4th International Conference on Fluid Mechanics and Fluid Power*, December 16-18, 2010, IIT Madras, Chennai, India.
- [9] Sabaeifard P, Razzaghi H, dan Forouzandeh A., " Determination of Vertical Axis Wind Turbines Optimal Configuration through CFD Simulation", *International Conference on Future Environment and Energy*, 2012, vol 28, IACSIT Press, Singapore.
- [10] Gosselin R, Dumas G dan Boudreau M, " Parametric Study of H-Darrius Vertical-Axis Turbines using uRANS Simulations", *21st Annual Conference of the CFD Society of Canada*, May 6-9, 2013, Sherbrooke,

Filter Aktif Menggunakan Deteksi Tegangan dan Deteksi Arus Untuk Mengkompensasi Harmonisa Pada Sistem Distribusi Radial

Asran

Jurusan Teknik Elektro
Universitas Malikussaleh Lhokseumawe
Email : Asran_lsm@yahoo.com

Abstrak— Pada penelitian ini didesain suatu filter aktif yang menggunakan deteksi tegangan dan dibandingkan dengan deteksi arus untuk mengkompensasi harmonisa pada sisi akhir saluran distribusi radial sistem tenaga listrik. Filter aktif ini akan mengurangi penyebaran harmonisa yang dihasilkan oleh beban non linier di saluran distribusi. Pada penelitian ini filter aktif yang diajukan menggunakan transformasi tegangan dari sistem abc ke dq yang dikombinasikan dengan filter, kemudian digunakan metode sinusoidal PWM untuk mengendalikan inverter. Filter aktif disimulasikan untuk diterapkan di Penyulang Pandanwangi, sistem distribusi Kota Malang Jawa Timur. Dari hasil simulasi didapatkan kesimpulan bahwa harmonisa pada bus dapat diturunkan dengan baik, dan hasil perbandingan deteksi tegangan ITHD 5,03 % sedangkan menggunakan deteksi arus ITHD 6,11 %.

Keywords— Deteksi tegangan, filter aktif, PWM inverter, saluran distribusi radial.

I. LATAR BELAKANG

Harmonisa timbul karena pemakaian beban non linier dan menjadi masalah yang serius dalam sistem tenaga listrik akhir-akhir ini, penggunaan beban non linier dapat menimbulkan perubahan gelombang arus dan tegangan yang tidak sinusoidal. Pemakaian converter sebagai sumber daya searah membawa kerugian pada jaring listrik, yaitu dapat menghasilkan arus harmonisa. Peralatan sumber arus searah dapat menyuntik sejumlah harmonisa kedalam sistem distribusi daya. Harmonisa ini mengakibatkan penurunan kualitas daya di sistem transmisi distribusi [1, 2].

Untuk meningkatkan kualitas sistem tenaga listrik maka distorsi harmonisa harus ditekan seminimal mungkin dengan menggunakan filter harmonisa. Filter harmonisa selain dapat mereduksi distorsi harmonisa juga dapat berfungsi sebagai kompensator reaktif pada frekuensi fundamental [3, 4, 5].

Pada penelitian ini didesain suatu filter aktif yang menggunakan deteksi tegangan untuk mengkompensasi harmonisa pada sisi akhir saluran distribusi radial sistem tenaga listrik. Filter aktif akan mengurangi penyebaran harmonisa yang dihasilkan akibat oleh beban non linier di saluran distribusi [6].

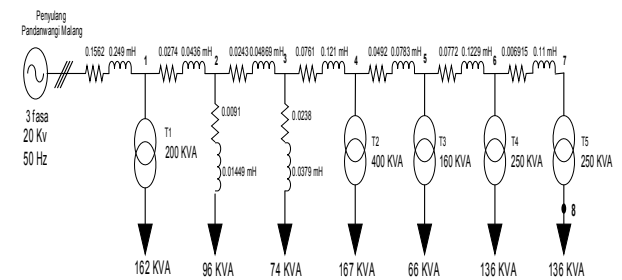
II. SISTEM KONFIGURASI FILTER YANG DIAJUKAN

Pada Gambar 1 menunjukkan suatu simulator saluran distribusi daya tiga fasa digunakan untuk percobaan. Saluran

simulator pada rating 20 kV, 50 Hz yang merupakan jaring distribusi di penyulang Pandanwangi Kabupaten Malang.

Saluran antara suatu bus dan yang ber-sebelahan bus dapat diwakili oleh suatu rangkaian RL, dimana parameter tergantung pada ketebalan dan panjangnya saluran.

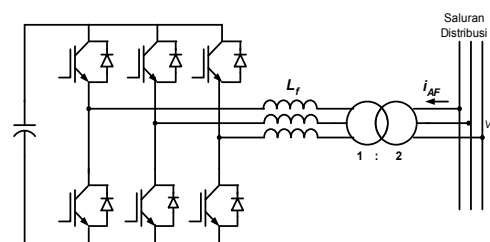
Filter aktif akan mengurangi penyebaran harmonisa yang dihasilkan oleh beban non linier di saluran distribusi. Analisa yang akan dilakukan terhadap kinerja sistem distribusi penyulang Pandanwangi Malang. Filter Aktif dihubungkan pada akhir bus pada tegangan 380 Volt, 50 Hz, 136 KVA.



Gambar 1. Permodelan sistem distribusi penyulang Pandanwangi Malang

2.1. Prinsip Kerja Filter Aktif

Filter aktif yang diajukan menggunakan transformasi tegangan dari sistem abc ke dq yang dikombinasikan dengan filter, kemudian digunakan metode untuk memperoleh arus referensi berdasarkan daya aktif dan reaktif sesaat serta menggunakan PWM inverter sebagai sumber arus harmonisa kompensasi. PWM (pulse width modulation) inverter yang merupakan penggabungan 6 buah IGBT (Insulated gate bipolar transistor). Suatu kapasitor elektronik 3300 μ F dihubungkan pada sisi dc, dan tegangan dc adalah 260 Volt, sedangkan induktor $L_f = 10,5$ mH dihubungkan pada sisi ac, arus yang keluar dari inverter diinjeksi ke sistem melalui transformator tiga fasa dengan perbandingan putaran 2 : 1.



Gambar 2. Rangkaian daya filter aktif

Filter aktif tiga fasa bekerja didasarkan atas topologi VSAF (Voltage Source Aktive Filter) yang terdiri dari enam switch inverter bridge filter induktansi dan kapasitor dc. Arus yang harus dihasilkan oleh masing-masing switch adalah arus maksimum filter induktor (IF). Dengan menerapkan hukum arus Kirchoof pada node bus akan didapat persamaan sebagai berikut :

$$I_F = i_S + i_{NL} \quad (1)$$

Dimana arus line harus support terhadap daya rata-rata yang digunakan oleh beban non linier dan filter aktif, dan tegangan yang dihasilkan oleh masing-masing switch adalah tegangan maksimum yang muncul pada kapasitor dc. Arus kapasitor diswitch sedemikian sehingga kinerjanya dapat dikontrol untuk memaksakan arus IF memiliki bentuk gelombang yang benar. Tegangan nominal pada bus dc (Vdc) harus lebih besar atau sama dengan tegangan puncak line-line untuk mendapatkan arus IF yang terkontrol.

$$V_{dc} \geq \frac{3}{2} \cdot V_{L-L} \quad (2)$$

Komponen induktor (LF) pada filter aktif merupakan komponen yang menghubungkan filter aktif dengan jaringan sebagai kontrol terhadap filter aktif yaitu menjaga agar ripple arus yang terjadi agar tetap kecil atau berfungsi sebagai first order pasive filter dengan menghilangkan frekuensi tinggi berupa ripple yang dibangkitkan oleh PWM inverter. Dalam pemilihan besar nilai komponen induktor harus memperhatikan bahwa LF harus menjamin filter aktif dapat membangkitkan arus dengan slop sama dengan slop maksimum arus beban non linier. Batasan maksimum untuk komponen induktor pada filter aktif adalah :

$$L_F \leq \frac{\frac{2}{3} V_{dc} - V_{x'n}}{\max \left| \frac{di_{NLx}}{dt} \right|} \quad (3)$$

Dimana $V_{x'n}$ dan i_{NLx} adalah tegangan puncak fasa-netral dan arus beban fasa x. Batas bahwa nilai induktor adalah level frekuensi switching arus ripple yang diterima, sedangkan untuk nilai C didapatkan dari tegangan ripple maksimum yang dapat diberikan seperti pada persamaan berikut :

$$C \geq \frac{\max \left| \int_0^t ip dt \right|}{\Delta Vc \max} \quad (4)$$

Dimana $\Delta Vc \max$ adalah tegangan ripple maksimum. Nilai C dan nilai L_F dapat diatur berdasarkan simulasi yang akan dilakukan.

2.2. Rangkaian Kontrol

2.2.1. Rangkaian Kontrol Filter Aktif

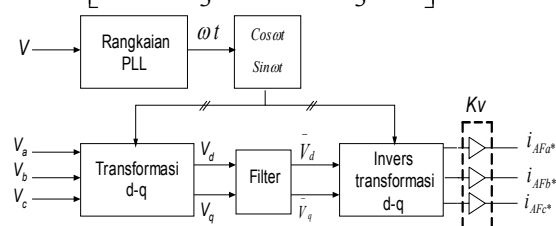
Pada Gambar 3. menunjukkan rangkaian kontrol filter aktif. Untuk mendapatkan pola pengontrolan gate filter aktif, maka terlebih dahulu harus mengukur tegangan bus

dan arus sumber tiga fasa. Dan tegangan sumber harmonisa pada tiap fasa dihitung dengan menggunakan teori d-q :

$$\begin{bmatrix} v_d \\ v_q \\ v_0 \end{bmatrix} = \frac{2}{3} \begin{bmatrix} \cos \theta & \cos(\theta - \frac{2\pi}{3}) & \cos(\theta + \frac{2\pi}{3}) \\ \sin \theta & \sin(\theta - \frac{2\pi}{3}) & \sin(\theta + \frac{2\pi}{3}) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \begin{bmatrix} v_a \\ v_b \\ v_c \end{bmatrix} \quad (5)$$

Pada persamaan (5) adalah komponen tegangan sumber dari V_a, V_b dan V_c yang masih mengandung komponen dc dan komponen harmonisa ke dalam bentuk V_d dan V_q . Kemudian dua buah filter dengan frekuensi cut-off 5 Hz ketika satu sama lain menyaring komponen AC \tilde{V}_d dan \tilde{V}_q dari V_d dan V_q . Komponen AC diaplikasikan sebagai rangkaian transformasi invers, dan digunakan sebagai arus kompensasi tiga fasa pada filter aktif dengan menggunakan persamaan berikut :

$$\begin{bmatrix} v_a \\ v_b \\ v_c \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta & 1 \\ \cos(\theta - \frac{2\pi}{3}) & \sin(\theta - \frac{2\pi}{3}) & 1 \\ \cos(\theta + \frac{2\pi}{3}) & \sin(\theta + \frac{2\pi}{3}) & 1 \end{bmatrix} \begin{bmatrix} v_d \\ v_q \\ v_0 \end{bmatrix} \quad (6)$$



Gambar 3. Rangkaian kontrol filter aktif

Dengan demikian rangkaian kontrol menghasilkan tegangan harmonisa tiga fasa. Membesarnya masing-masing tegangan harmonisa oleh suatu penguat KV menghasilkan referensi arus sebagai berikut :

$$i_{AF}^* = Kv * v_h^* \quad (7)$$

Persamaan (7) mengasumsikan bahwa filter aktif bertindak seperti sebuah resistor senilai $1/KV \Omega$ untuk frekuensi harmonisa, digunakan untuk menentukan besarnya i_{AF}^* . Arus nyata tiga fasa i_{AFa}, i_{AFb} dan i_{AFc} dikontrol sedemikian untuk mengikuti referensi arus.

III. SIMULASI DAN ANALISIS

Dalam melakukan simulasi penggunaan filter aktif terhubung seri untuk mengkompensasi harmonisa dalam sistem tenaga listrik. Simulasi yang dilakukan terbagi dalam beberapa bagian, yaitu simulasi sistem tanpa filter aktif dan simulasi sistem setelah dipasang filter aktif.

3.1. Nilai Parameter yang Digunakan Untuk Simulasi.

Nilai-nilai simulasi yang digunakan adalah sebagai berikut :

- Tegangan sistem tiga fasa (V_s) = 20 kV (V_{L-L}).
- Frekuensi = 50 Hz.
- Daya = 20 MVA.
- Rugi-rugi transmisi, Resistor ; $R_1=0.156 \Omega$, $R_2=0.027 \Omega$, $R_3=0.024 \Omega$, $R_4=0.076 \Omega$, $R_5=0.049 \Omega$, $R_6=0.077\Omega$, $R_7=0.0069 \Omega$.
- Induktansi saluran $L_1=0.249$ mH, $L_2=0.0436$ mH, $L_3=0.04869$ mH, $L_4=0.121$ mH, $L_5=0.0783$ mH, $L_6=0.1229$ mH, $L_7=0.11$ mH.
- Beban non linier berupa penyearah.

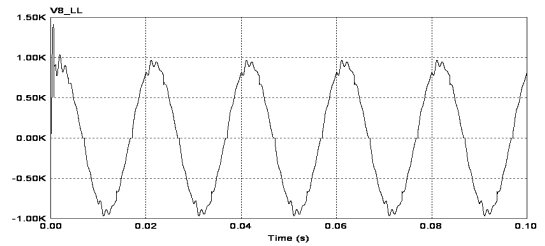
Konfigurasi yang dibentuk untuk percobaan ini berupa sistem distribusi penyalang Pandanwangi Malang. Filter aktif terdiri dari PWM inverter yang merupakan penggabungan 6 buah IGBT. Filter aktif dihubungkan pada suatu bus pada tegangan 380 Volt, 50 Hz, 136 KVA. Saluran simulator melalui suatu transformator tiga fasa dengan perbandingan putaran 2 : 1. Suatu kapasitor elektrik 3300 μ F dihubungkan pada sisi dc, dengan tegangan dc adalah 260 Volt, sedangkan induktor $L_f = 10,5$ mH dihubungkan pada sisi ac

3.2. Simulasi Sistem Tanpa Filter Aktif.

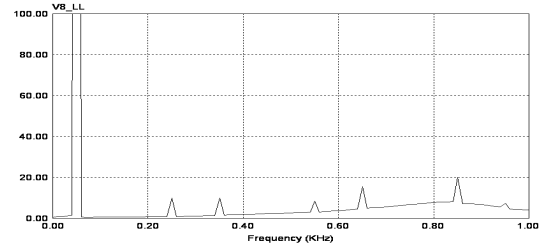
Untuk simulasi awal, mensimulasikan sistem tanpa filter aktif yang terdiri atas sumber tegangan, impedansi saluran dan beban non linier.

Dengan mensimulasikan sistem tanpa filter, maka akan didapat besar arus, spektrum frekuensinya dan *total harmonic distortion* (THD). Data-data simulasi ini digunakan sebagai dasar untuk melakukan simulasi selanjutnya, baik saat simulasi dengan menggunakan filter aktif yang terhubung seri.

Adapun hasil simulasi sistem ideal tanpa filter aktif adalah sebagaimana Gambar 5 berikut :

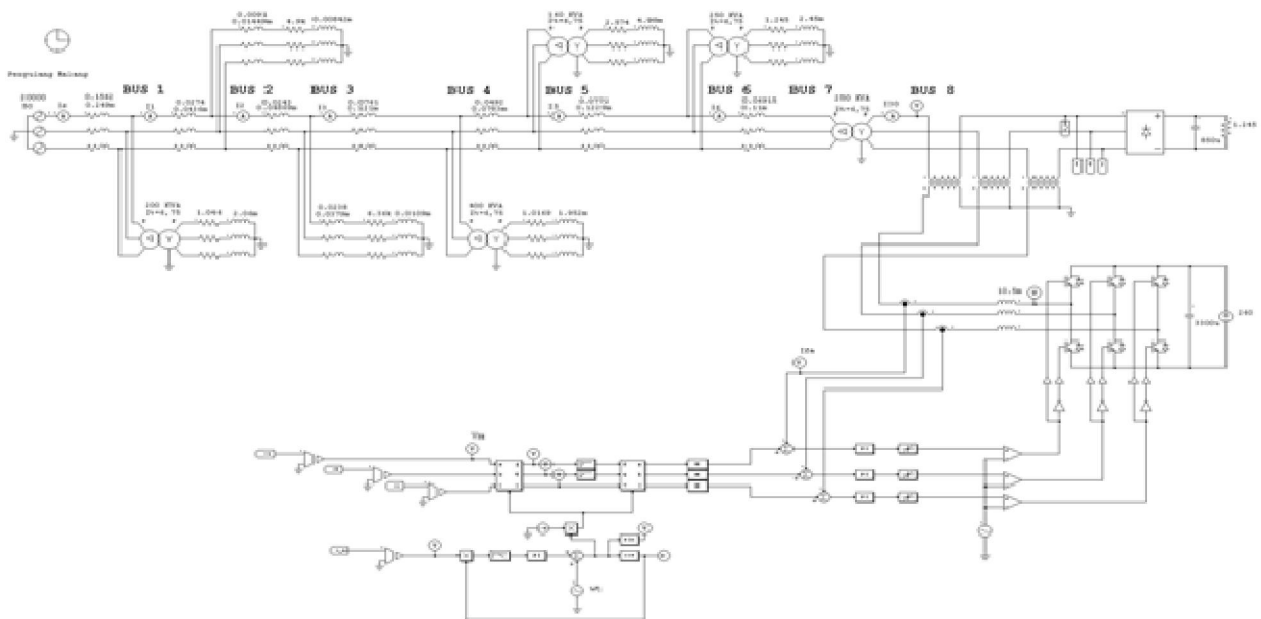


Gambar 5. Tegangan pada bus 8 sebelum dipasang filter aktif. (THD = 3,49 %)

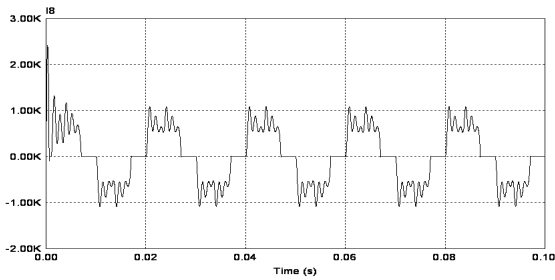


Gambar 6. Spektrum tegangan pada bus 8 sebelum dipasang filter aktif

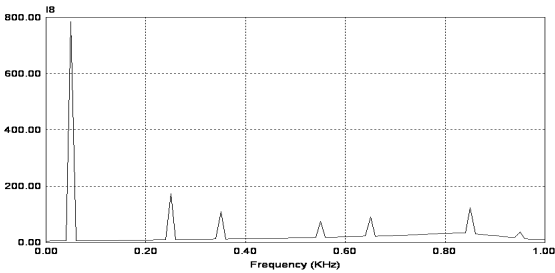
Gambar 5 merupakan gelombang yang dihasilkan transformator pada bus 8 pada tegangan 380 Volt. Dari hasil pengukuran tampak adanya harmonisa tegangan dengan THD tegangan sebesar 3,49 %. Hal ini terjadi diakibatkan pengaruh impedansi pada saluran distribusi dan beban non linier yang dipasang. Gambar 6 tampak besar magnitudo tegangan harmonisa ke-5 (250 Hz) = 5,68 V, tegangan harmonisa ke-7 (350 Hz) = 4,85 V, $V_{Fundamental}$ (50 Hz) = 534,07 V.



Gambar 4. Rangkaian simulasi Filter Aktif Menggunakan Deteksi Tegangan



Gambar 7. Arus pada bus 8 sebelum dipasang filter aktif (THD = 33,97 %)



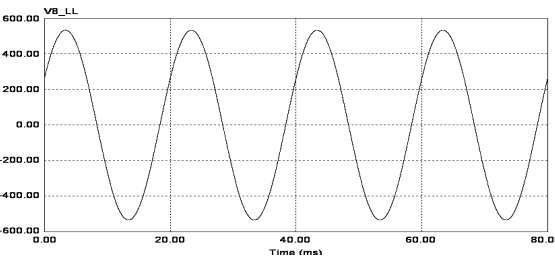
Gambar 8. Spektrum arus pada bus 8 sebelum dipasang filter aktif

Gambar 7 merupakan hasil pengukuran arus bus 8 pada tegangan 380 V sebelum dipasang filter aktif shunt dengan nilai THD arus sebesar 33.97 %, Gambar 8 merupakan spektrum gelombang arus bus 8 sebelum dipasang filter aktif dengan besar magnitudo arus harmonisa ke-5 (250 Hz) = 174,2 A, arus harmonisa ke-7 (350 Hz) = 104 A, $I_{Fundamental}$ (50 Hz) = 783,7 A.

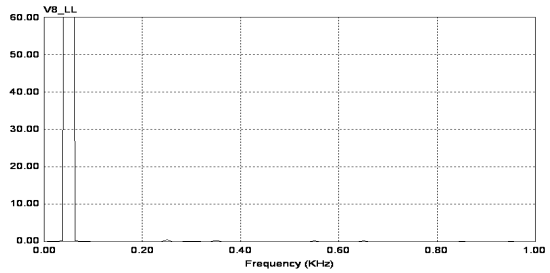
3.3 Simulasi Sistem Setelah Dipasang Filter Aktif Menggunakan Deteksi Tegangan

Dari Spektrum frekuensi arus pada sisi bus 8 tegangan 380 Volt dapat dilihat bahwa harmonisa yang dominan adalah harmonisa ke-5 (250 Hz) dan harmonisa ke-7 (350 Hz). Dengan melihat harmonisa yang dikandung, maka filter aktif yang dipilih untuk mengkompensasi harmonisa yang dihubungkan ke sistem.

Adapun hasil simulasi setelah dipasang filter aktif adalah sebagaimana Gambar 9 :

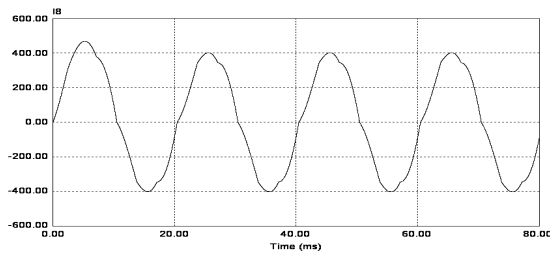


Gambar 9. Tegangan pada bus 8 setelah dipasang Filter Aktif (THD = 0,1 %)

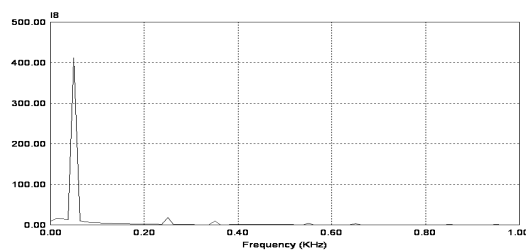


Gambar 10. Spektrum Tegangan pada bus 8 setelah dipasang Filter Aktif

Gambar 9 merupakan gelombang yang dihasilkan transformator pada bus 8 pada tegangan 380 Volt. Dari hasil pengukuran tampak bahwa terjadi perubahan nilai THD sebagai akibat pemasangan filter aktif. Pemasangan filter aktif menyebabkan sebagian besar harmonisa yang ditimbulkan oleh beban non linier mengalir menuju filter aktif yang dipasang. Terjadi penurunan THD tegangan dari 3,49 % menjadi 0,1 %. Gambar 10 merupakan spektrum gelombang tegangan pada bus 8 pada tegangan 380 Volt setelah dipasang filter aktif dengan besar magnitudo tegangan harmonisa ke-5 (250 Hz) = 0,58 V, tegangan harmonisa ke-7 (350 Hz) = 0,39 V, $V_{Fundamental}$ (50 Hz) = 534,3 V.



Gambar 11. Arus pada bus 8 setelah dipasang Filter Aktif (THD = 5,03 %)



Gambar 12. Spektrum Arus pada bus 8 setelah dipasang Filter Aktif

Gambar 11 menunjukkan gelombang arus bus 8 tegangan 380 Volt, tampak terjadi perubahan nilai THD arus sebagai akibat pemasangan filter aktif. Pemasangan filter aktif menyebabkan sebagian besar harmonisa yang ditimbulkan oleh beban non linier mengalir menuju filter aktif yang dipasang. Terjadi penurunan THD arus dari 33,97 % menjadi 5,03 %. Gambar 12 merupakan spektrum gelombang arus bus 8 setelah dipasang filter aktif dengan besar magnitudo arus harmonisa ke-5 (250 Hz) = 18,26 A, arus harmonisa ke-7 (350 Hz) = 8,96 A, $I_{Fundamental}$ (50 Hz) = 412 A.

Tabel 1. Nilai THD sebelum dan setelah dipasang filter aktif menggunakan deteksi tegangan

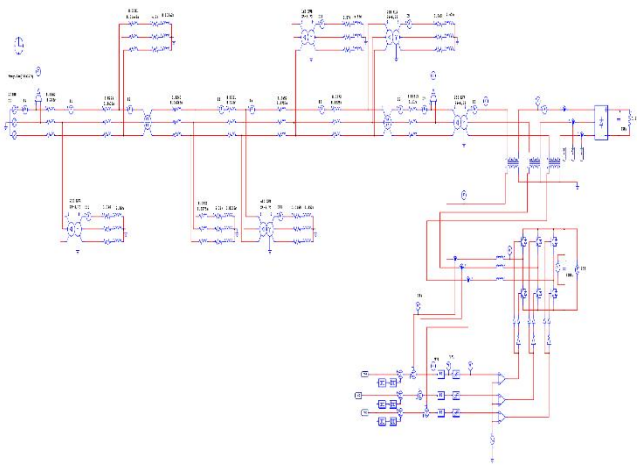
Bus	Rated kV	Sebelum dipasang filter aktif		Setelah dipasang filter aktif deteksi tegangan	
		%V THD	% I THD	%V THD	% I THD
Sumber	20	0,01	12,94	0	1,73
1	20	0,04	14,92	0	1,99
2	20	0,05	16,17	0	2,11
3	20	0,06	16,39	0	2,18
4	20	0,08	21,85	0	2,72
5	20	0,10	25,02	0	3,16
6	20	0,12	35,59	0	5,50
7	20	0,14	35,59	0	5,50
8	0,38	3,49	33,97	0,1	5,03

Arus harmonisa yang mengalir ternyata tidak seluruhnya dapat dihilangkan karena arus yang diinjeksikan oleh filter aktif tidak bisa tepat dengan arus harmonisa yang dikandung, ini disebabkan karena beberapa hal :

- Sumber tegangan dc murni inverter tidak bisa mengikuti perubahan arus harmonisa secara tepat, sehingga arus injeksinya kadang lebih besar atau lebih kecil dari seharusnya.
- Induktor perata arus L dari filter aktif mempengaruhi terjadi pergeseran fasa pada arus filter aktif walaupun sedikit.

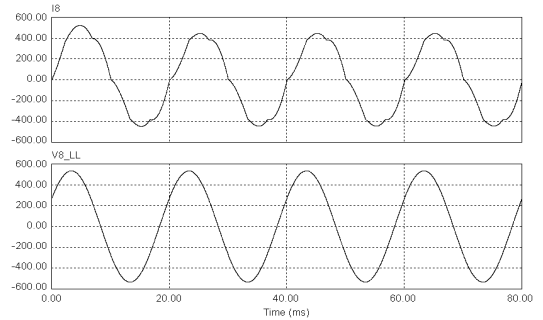
3.4 Simulasi Sistem Dengan Filter Aktif Menggunakan Deteksi Arus.

Pada sub bab ini dilakukan simulasi khusus untuk bus 8 menggunakan deteksi arus. Hasil simulasi ini akan dibandingkan dengan hasil simulasi filter aktif menggunakan deteksi tegangan. Pada Gambar 13 menunjukkan sistem saat dipasang filter aktif menggunakan deteksi arus.



Gambar 13. Rangkaian simulasi Filter Aktif Menggunakan Deteksi Arus

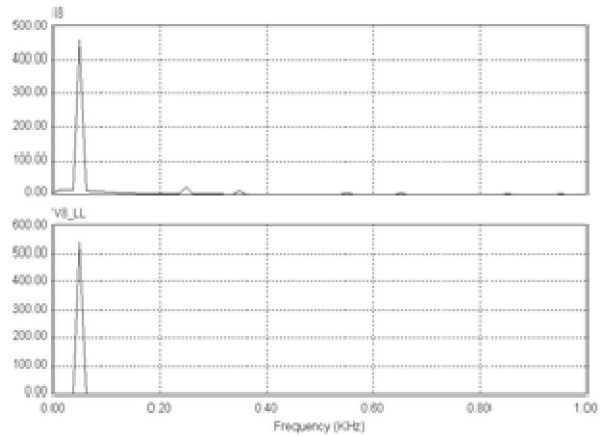
Hasil simulasi filter aktif menggunakan deteksi arus pada bus 8 dapat dilihat pada Gambar 14 berikut ini :



Gambar 14 Bentuk gelombang arus dan tegangan pada bus 8 dengan filter aktif menggunakan deteksi arus

Gambar 14 menunjukkan bentuk gelombang arus dan tegangan setelah pemasangan filter aktif menggunakan deteksi arus.

Untuk perbandingan hasil simulasi THD arus dan THD tegangan dapat dilihat pada Tabel 2, dari hasil simulasi terlihat hampir sama. Besar THD arus menggunakan deteksi arus adalah 6,11 % sedangkan menggunakan deteksi tegangan 5,03 % dan besar THD tegangan menggunakan deteksi arus 0,2 % sedangkan menggunakan deteksi tegangan 0,1 %.



Gambar 15. Spektrum arus dan tegangan pada bus 8 dengan filter aktif menggunakan deteksi arus.

Gambar 15 merupakan spektrum arus dan tegangan bus 8 setelah dipasang filter aktif menggunakan deteksi arus dengan besar magnitudo arus harmonisa ke-5 (250 Hz) = 24,39 A, arus harmonisa ke-7 (350 Hz) = 12,13 A, $I_{Fundamental}$ (50 Hz) = 456,6 A. Gambar 15 yang kedua merupakan spektrum gelombang tegangan pada bus 8 pada tegangan 380 Volt setelah dipasang filter aktif menggunakan deteksi arus dengan besar magnitudo tegangan harmonisa ke-5 (250 Hz) = 0,822 V, tegangan harmonisa ke-7 (350 Hz) = 0,573 V, $V_{Fundamental}$ (50 Hz) = 534,13 V

3.5 Perbandingan Sistem Dengan Filter Aktif Menggunakan Deteksi arus dan Filter Aktif Menggunakan Deteksi Tegangan.

Tabel 2 menunjukkan perbandingan nilai THD arus dan THD tegangan untuk keseluruhan sistem saat pemasangan filter aktif deteksi arus dan deteksi tegangan.

THD arus terbesar terdapat di bus 8. Pada pemasangan filter aktif deteksi arus diperoleh THD arus sebesar 6,11 % sedangkan dengan deteksi tegangan dapat diperbaiki menjadi 5,03 %. Secara umum filter aktif deteksi tegangan dapat menghasilkan THD yang lebih kecil dibandingkan deteksi arus.

Tabel 2 Perbandingan nilai THD menggunakan deteksi arus dan menggunakan deteksi tegangan.

Bus	Rated kV	Filter aktif menggunakan deteksi arus		Filter aktif menggunakan deteksi tegangan	
		%V THD	% ITHD	%V THD	% ITHD
jumlah	20	0	1,73	0	1,73
1	20	0	2,02	0	1,99
2	20	0	2,13	0	2,11
3	20	0	2,24	0	2,18
4	20	0	2,93	0	2,72
5	20	0	3,36	0	3,16
6	20	0	6,00	0	5,50
7	20	0	6,00	0	5,50
8	0,38	0.2	6,11	0.1	5,03

IV. KESIMPULAN

Dari hasil simulasi yang telah dilakukan dapat diambil kesimpulan sebagai berikut :

1. Filter aktif yang telah didesain, yang menggunakan deteksi tegangan dapat bekerja dengan baik untuk menurunkan arus harmonisa.
2. Dari hasil simulasi menunjukkan kinerja filter aktif dapat menurunkan harmonisa pada bus dengan baik, dari THD tegangan 3,49 % menjadi 0,1 % dan THD arus dari 33,97 % menjadi 5,03 %.
3. THD arus sebelum dipasang filter aktif adalah 33,97 %, setelah pemasangan filter aktif menggunakan deteksi arus menjadi 6,11 % sedangkan menggunakan deteksi tegangan dapat menghasilkan THD yang lebih kecil yaitu 5,03 %.

V. REFERENSI

- [1]. A. van Zyl, J. H. R. Enslin, and R. Sp'ee, (1996), "Converter-based solution to power quality problems on radial distribution lines", IEEE Trans. Ind. Applicat., vol. 32, pp. 1323-1330.
- [2]. Margo Pujiantara, Dedet Chandra R, Agus Wibawa B. U. (2003), "Improvement Passive Filter As Harmonic Filter Using Active Techniqu", CECI.
- [3]. T.-N. L'e, M. Pereira, K. Renz, and G. Vaupel, (1994), "Active damping of resonance in power systems", IEEE Trans. Power Del., vol. 9, pp. 1001-1008.
- [4]. H. Akagi, (1997), "Control strategy and site selection of a shunt active filter for damping of harmonic propagation in

power distribution systems", IEEE Trans. Power Delivery, vol.12, pp. 354-363.

- [5]. H. Akagi, (1996), "New trends in active filters for power conditioning", IEEE Trans. Ind. Applicat., vol. 32, pp. 1313-1322.
- [6]. H. Akagi, H. Fujita, and K.Wada, (1999), "A shunt active filter based on voltage detection for harmonic termination of a radial power distribution line", IEEE Trans. Ind. Applicat., vol.35, no. 3, pp. 638-645.
- [7]. F. Z. Peng, H. Akagi, and A. Nabae, (1990), "A new approach to harmonic compensation in power systems—A combined system of shunt passive and series active filters", IEEE Trans. Ind. Applicat., vol. 26, pp. 983-990.
- [8]. Andrea Cavallini and Gian Carlo Montanari, (1994), "Compensation Strategies for Shunt Active - Filter Control", IEEE Transactions On Power Electronics, Vol.9, No.6, , pp 587-593.
- [9]. Peng F.Z., Akagi H. and Nabae A., (1993), "Compensation Characteristics of the combined system of shunt passive and series active filters", IEEE Trans. Ind. Appl., Vol. 29, pp. 144-152.

Understanding TCP Performance Using Drop -Tail Queuing Mechanism

Muhammad Ikhwanus

Electrical Department, Malikussaleh University.
 Email: ikhwan_2k5@yahoo.com

Abstrak— TCP offers error correction so there is a “guaranteed delivery.” If the ACK is not received within a certain amount of time, it will retransmit the packet. The scenarios of this research were to create a simple scenario of TCP packets with Drop-tail queuing mechanism on the gateway from node 0 to 4 through Gateway, 1 to 4 through Gateway, and 2 to 4 through Gateway and bandwidth link from 1 to 8 Mb. The results of this scenario were when a packet of data is sent, a collision may occur in the network and the receiver can re-requests the packet from the node until the whole packet is completed and is identical to its original. The Drop-tail queuing mechanism use First-In-First-Out(FIFO) queue to store packets and drop-tail (or called First-Come-First-Serve) discipline to drop incoming packets when the queue is full. The result shows dropped packet is not fair in drop-tail mechanism

Keywords— TCP, ACK, Drop Tail mechanism.

And uses the following rules:

- The sender may send all packets within the window without receiving an ACK, but must start a timeout timer for each of them.
- The receiver must acknowledge each packet received, indicating the sequence number of the last well-received packet.
- The sender slides the window on each ACK received.

In our example, the sender may transmit packets 1 to 5 without waiting for any acknowledgment:

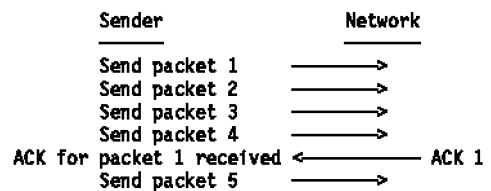


Figure3. Window Principle

I. INTRODUCTION

TCP (Transmission Control Protocol) provides a reliable end to end service that delivers packets over the Internet. It has a scenario send a packet and then wait for an acknowledgment from the receiver before sending the next packet. If the ACK is not received within a certain amount of time, it will retransmit the packet.

At the moment the sender receives the ACK 1 (acknowledgment for packet 1), it may slide its window to exclude packet 1:

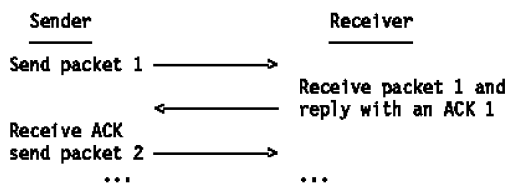


Figure 1. The Window Principle

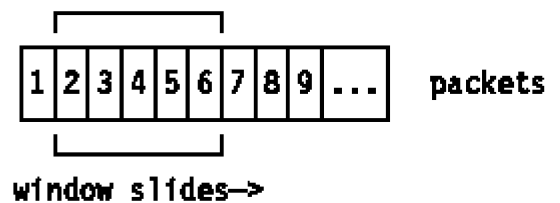


Figure 4 Message packets

While this mechanism ensures reliability, it only uses a part of the available network bandwidth.

At this point, the sender may also transmit packet 6.

Consider now a protocol where the sender groups its packets to be transmitted as below:

To make easier understanding about TCP scenario, we use Network Simulator 2 or(NS-2) to simulate how TCP make connection between different network entities (hosts/routers/gateway, data links, packets, etc).

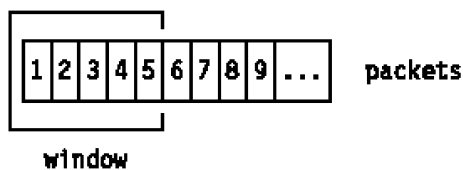


Figure 2. Message packets

The objective of this research are as follow:

1. To have a basic understanding of the network parameters, which can be modified.
2. Understand the basic components of TCP transmission.
3. Observe the behavior of TCP sessions.

II. SCOPE & METHODOLOGY

The scenarios of the research were to create a simple scenario of TCP packets with Drop-tail queuing mechanism on the gateway from node 0 to 4 through Gateway, 1 to 4 through Gateway, and 2 to 4 through Gateway and bandwidth link from 1 to 8 Mb as describe in below.

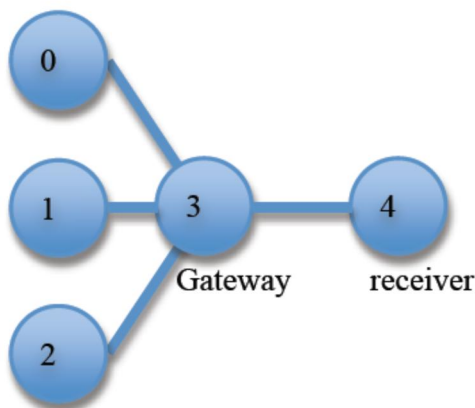


Figure 5.Scenario

Based on this above picture, we had to observe the characteristic of TCP algorithm and get understanding of the network parameters by using NS-2.

III. FINDING AND DISCUSSION

In this scenario we made same color for each node link and packet in order to get easier observation. We set the speed of animation at 0.5 ms. and 1 Mb bandwidth for each link in NS-2, there were some events of TCP behavior in the network as bellow:

1. Figure 6 show that When a TCP connection started, the n1, n2 and n3 starts to sent the ACK at the same time because of each node had no knowledge of the state of the routers along the end-to-end path. They might be dealing with rather large queues, or they might be totally underutilized at the time. Consequently, the gateway would not begin blasting out data at a high rate, but instead chooses a more conservative approach known as slow-start.

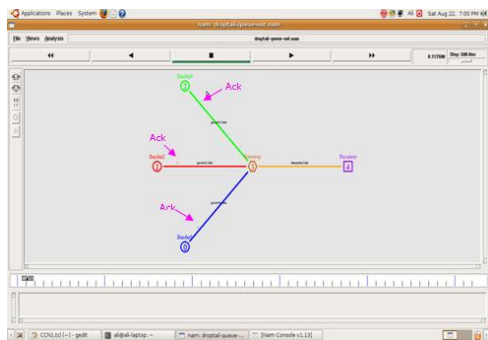


Figure 6. Each node sent ACK

Then the ACK would sent back from receiver to each node to ensure the node can transmitted the packet as shown in figure 7.

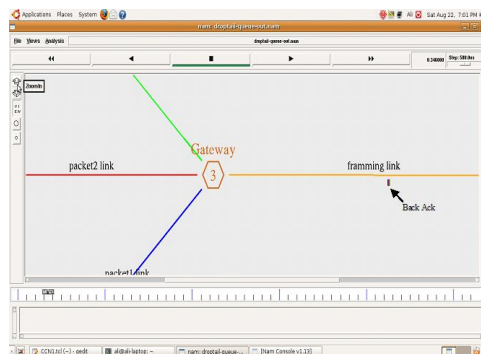


Figure 7. ACK send back to each node

2. In slow start animation (at 500 ms), the congestion window (the amount of data packets that can be sent out at once) was initially set to one, so each node began by transmitting only one data packet. When an ACK was received by node, the node then doubles the congestion window and sent out two packets as shown in figure 8a to 8b and the ACK would be sent back to each node after receiving packet as shown in figure 9. The process repeats, and the congestion window increases exponentially until it reached some threshold.

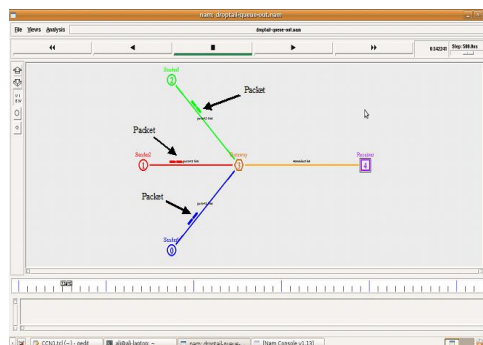


Figure 8a. Each node sending two packet after receive ACK

In this manner, congestion due to an overzealous server initiating a data transfer can be avoided, but without significant loss in throughput even for highly underutilized links thanks to the exponential increase of the congestion window.

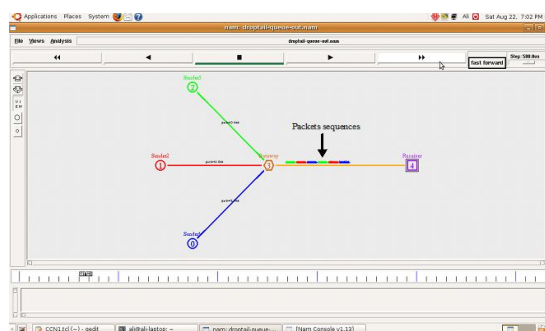


Figure 8b. the sequence of packet in transmission link.

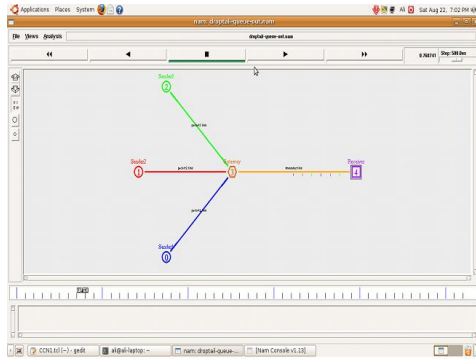


Figure 9. ACK sent back to each node

3. After all sending packet received by receiver without loss and all ACK received by each node, the each node tried to double packet as shown in figure 10.

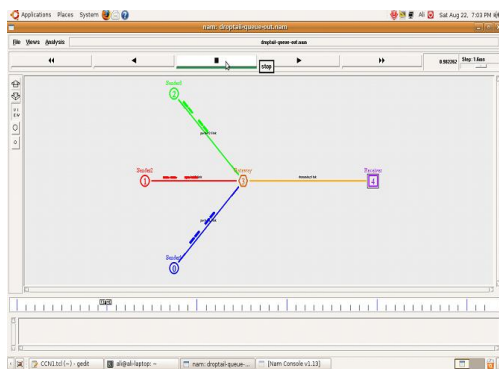


Figure 10. Each node tried to send double packet

4. Figure 11 shows one packet from sender 1 (the blue ones) and one packet from sender 2 (the red ones) were dropped in the transmission link because of using the drop tail queuing buffer mechanism (first in first out). The capacity of link can not adopt all of the packet sent by each node, so two packets would be dropped in the link transmission.

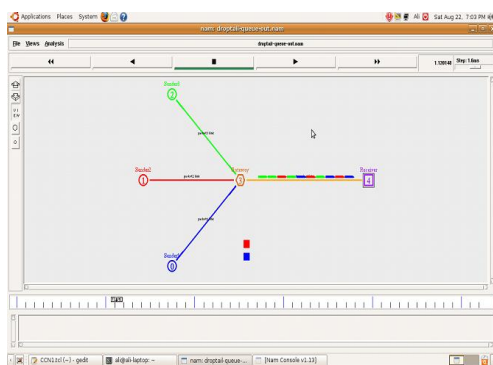


Figure 11. The sequence and dropping packet

Only the packet received by receiver would sent ACK to each node. Totally, There were twelve packets sent by three node, but only ten packets received by receiver. That means two packets were dropped as shown in figure 12.

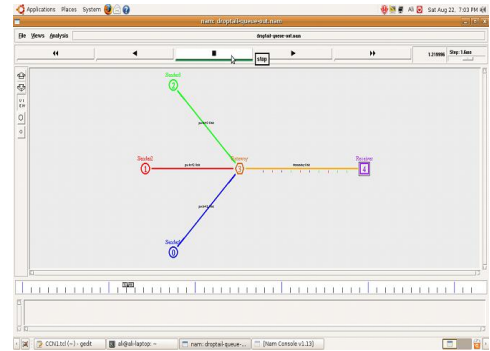


Figure 12. The ACK sent back to each node

5. Each node sent packet by adding two packet but only the sender 3 sent more two packets than sender 1 and sender 2. Sender three sent more packet because there was no dropping packet from the previous transmission as shown in figure 13. So it means the link still enough capacity to receive the packet. But in figure 14 two packets from sender 3 would dropped because the transmission link can adopted limited bandwidth capacity.

- 6.

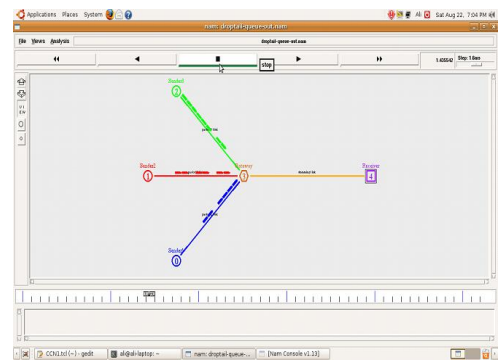


Figure 13. adding packets among sender

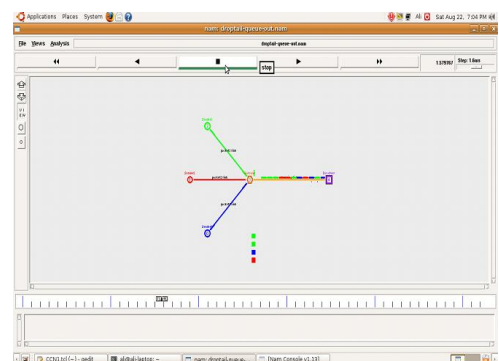


Figure 14. Dropping packet happen in sender 3.

It was also important to note that how these TCP algorithms respond to a loss. For a lost packet in slow-start, the congestion window is set to 1 and slow-start begins a new, while a loss during congestion avoidance will cause the congestion window to decrease by a factor of 2. If the congestion window is less than the slow-start threshold, slow start will be used to re-increase the congestion window

exponentially until the threshold is reached. Otherwise, the congestion avoidance method increases the congestion window at a lower rate.

IV. CONCLUSION

TCP window mechanism ensures:

1. TCP offers error correction so there is a “guaranteed delivery.”
2. TCP has a method called “flow control” that can be need to be re-sent data, and stops the flow of data until previous packets are successfully transferred.
3. When a packet of data is sent, a collision may occur in the network and the receiver can re-requests the packet from the node until the whole packet is complete and is identical to its original.
4. Drop-tail queuing mechanism use First-In-First-Out(FIFO) queue to store packets and drop-tail (or called First-Come-First-Serve) discipline to drop incoming packets when the queue is full.

5. The dropped packet is not fair in drop-tail mechanism.

V. RECOMMENDATION

Flow-control, as the receiver may delay replying to a packet with an acknowledgment, knowing its free buffers available and the window-size of the communication. To get more fairness queuing, replace drop-tail mechanism with Random Early Detection (RED) or SFQ (stochastic fair queuing) queue.

VI. REFERENCES

- [1]. Forouzan, Behrouza A., and Fegan, Sophia Chung. “TCP/IP Protocol Suite, Third Edition, Mc Graw Hill, 2006.
- [2]. Stalling, William. “Data and computer communication, 7th ed, Prentice-hall. 2004.

Studi Sistem Logika Fuzzy Sederhana Dengan Software Matlab

Fajriana

Jurusan Teknik Informatika UNIMAL
Fakultas Teknik

Abstrak— Penelitian ini membahas tentang system logika fuzzy dengan akan mengambil contoh penerapan pada permasalahan hubungan berat badan dan tinggi badan. Struktur system fuzzy yang akan digunakan adalah struktur metode Mamdani dan metode Sugeno.

Keywords— fuzzy, mamdani, Sugeno.

I. PENDAHULUAN

Logika fuzzy atau system fuzzy adalah suatu cara yang tepat untuk memetakan suatu ruang input ke dalam ruang output.

Dalam menentukan berat badan dan tinggi badan yang ideal dapat dilakukan dengan rumusan IMT (Indeks Masa Tubuh).

Variabel berat badan dibagi menjadi tiga kategori, yaitu ringan, sedang (normal), dan berat. Variabel tinggi badan dibagi menjadi tiga kategori, yaitu rendah, sedang (normal), dan tinggi.

Metode yang digunakan adalah metode Sogeno dan metode Mamdani. Tujuan dari penulisan ini adalah ingin mengetahui metode mana yang signifikan yang dapat digunakan pada berat badan dan tinggi badan.

II. TINJAUAN PUSTAKA

A. Logika Fuzzy

1. Pengertian Logika Fuzzy

Fuzzy secara bahasa diartikan sebagai kabur atau samar-samar. Suatu nilai dapat bernilai benar atau salah dalam waktu bersamaan. Dalam fuzzy dikenal derajat keanggotaan yang memiliki rentang nilai nol sampai satu. Awal konsep dari ketidakpastian adalah teori yang dipapar kan oleh Lotfi A Zadeh (1965), Dimana Zadeh mengatakan bahwa himpunan fuzzy merupakan suatu kelas objek dengan kontinu nilai keanggotaan. Himpunan seperti ini dicirikan dengan keanggotaan (karakteristik) fungsi yang memberikan ke setiap objek kelas keanggotaan berkisar antara nol dan satu.

2. Fungsi Keanggotaan

Definisi:

Fungsi keanggotaan untuk himpunan fuzzy pada semesta pembicaraan X di definisikan sebagai:

$$\mu_A: X \rightarrow [0, 1]$$

Dimana setiap elemen X dipetakan ke nilai antara 0 dan 1. Nilai ini yang disebut nilai keanggotaan atau derajat keanggotaan, tingkat kualifikasi keanggotaan dari elemen di X untuk himpunan fuzzy A .

Fungsi keanggotaan untuk grafis mewakili suatu himpunan fuzzy. Sumbu x merupakan semesta pembicaraan, sedangkan sumbu y merupakan derajat keanggotaan dalam interval $[0,1]$.

Operasi dasar

Seperti dalam logika klasik, dalam logika fuzzy ada tiga operasi dasar yang digunakan yaitu: gabungan, irisan dan komplemen.

Aturan adalah sebagai berikut:

1. Gabungan:

$$\mu_{A \cup B}(x) = \max(\mu_A(x), \mu_B(x))$$

2. Irisan:

$$\mu_{A \cap B}(x) = \min(\mu_A(x), \mu_B(x))$$

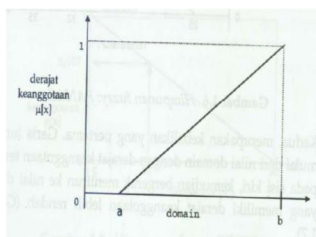
3. Komplemen:

$$\mu_{A^c}(x) = 1 - \mu_A(x)$$

Salah satu cara untuk mendapatkan nilai keanggotaan adalah dengan melalui pendekatan fungsi. Ada beberapa fungsi yang dapat digunakan yaitu:

1. Fungsi Linear.

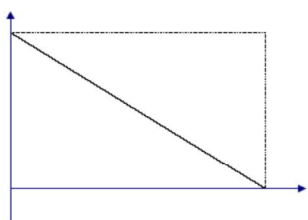
- Kenaikan himpunan dimulai pada nilai domain yang memiliki derajat keanggotaan nol bergerak ke kanan menuju nilai domain yang memiliki derajat keanggotaan lebih tinggi.



Fungsi Keanggotaan:

$$\mu(x) = \begin{cases} 0, & x \leq a \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ 1, & x \geq b \end{cases}$$

- b. Garis lurus dimulai dari nilai domain dengan derajat keanggotaan tertinggi pada sisi kiri, kemudian bergerak menurun ke nilai domain yang memiliki derajat keanggotaan lebih rendah

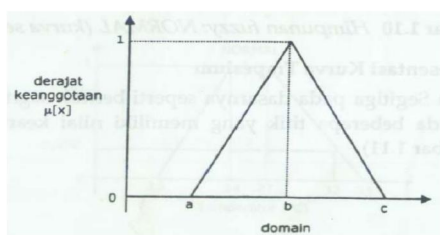


Fungsi Keanggotaan:

$$\mu(x) = \begin{cases} 1, & x \leq a \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ 0, & x \geq b \end{cases}$$

2. Kurva Segitiga

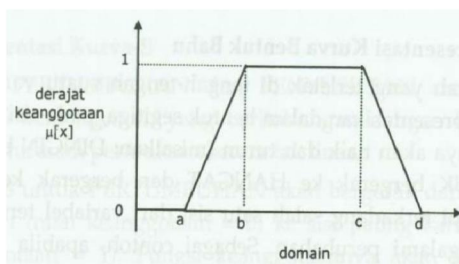
Kurva Segitiga ini didasarkan pada penggabungan antara dua garis (Fungsi Linear).



Fungsi Keanggotaan:

$$\mu(x) = \begin{cases} 0, & x \leq a \text{ atau } x \geq c \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ \frac{c-x}{c-b}, & b \leq x \leq c \end{cases}$$

3. Kurva Trapezium

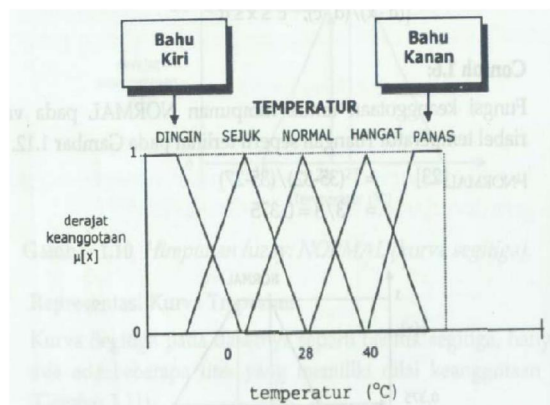


Fungsi Keanggotaan:

$$\mu(x) = \begin{cases} 0; & x \leq a \text{ atau } x \geq d \\ (x-a)/(b-a); & a \leq x \leq b \\ 1; & b \leq x \leq c \\ (d-x)/(d-c); & c \leq x \leq d \end{cases}$$

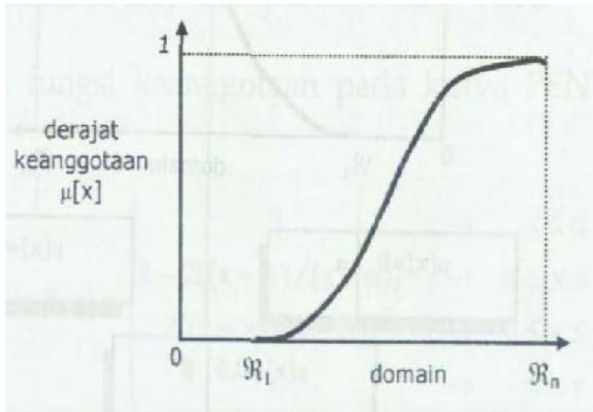
4. Kurva Bentuk Bahu

Daerah yang terletak di tengah-tengah suatu variabel yang direpresentasikan dalam bentuk segitiga, pada sisi kanan dan kirinya akan naik dan turun (misalkan: DINGIN bergerak ke SEJUK bergerak ke HANGAT dan bergerak ke PANAS). Tetapi terkadang salah satu sisi dari variable tersebut tidak mengalami perubahan. Sebagai contoh, apabila telah mencapai kondisi PANAS, kenaikan temperatur akan tetap berada pada kondisi PANAS. Himpunan fuzzy 'bahu', bukan segitiga, digunakan untuk mengakhiri variabel suatu daerah fuzzy. Bahu kiri bergerak dari benar ke salah, demikian juga bahu kanan bergerak dari salah ke benar.

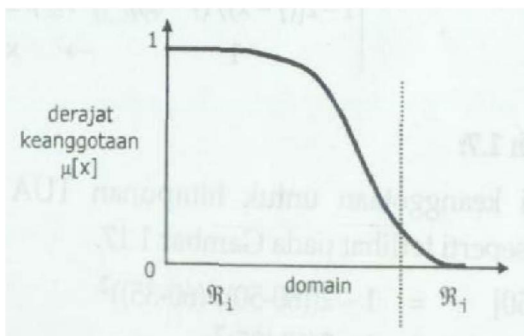


5. Kurva S

Kurva-S untuk PERTUMBUHAN akan bergerak dari sisi paling kiri (nilai keanggotaan = 0) ke sisi paling kanan (nilai keanggotaan = 1). Fungsi keanggotaannya akan tertumpu pada 50% nilai keanggotaannya yang sering disebut dengan titik infleksi. Kurva S hampir sama dengan kurva linear tetapi nilai yang tidak pasti berurut naik atau turun melainkan fleksibel.

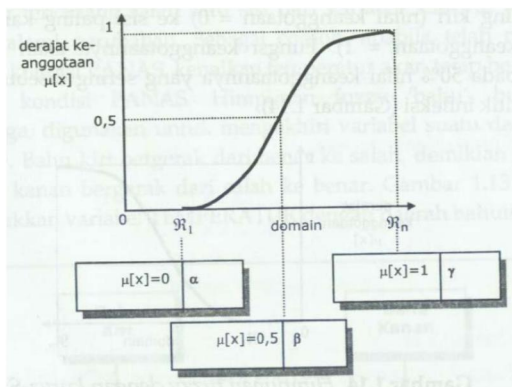


Kurva-S untuk PENYUSUTAN akan bergerak dari sisi paling kanan (nilai keanggotaan = 1) ke sisi paling kiri (nilai keanggotaan = 0)



Kurva-S didefinisikan dengan menggunakan 3 parameter, yaitu:

- nilai keanggotaan nol (a)
- nilai keanggotaan lengkap(y)
- titik infleksi atau crossover (p) yaitu titik yang memiliki domain 50% benar.



Fungsi Keanggotaan Pertumbuhan:

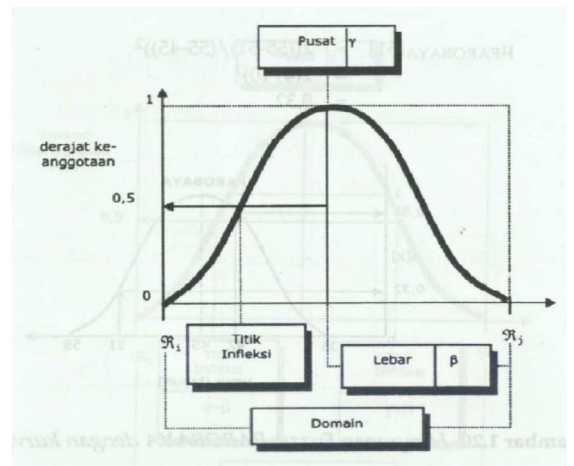
$$S(x; \alpha, \beta, \gamma) = \begin{cases} 0 & \rightarrow x \leq \alpha \\ 2((x - \alpha) / (\gamma - \alpha))^2 & \rightarrow \alpha \leq x \leq \beta \\ 1 - 2((\gamma - x) / (\gamma - \alpha))^2 & \rightarrow \beta \leq x \leq \gamma \\ 1 & \rightarrow x \geq \gamma \end{cases}$$

Fungsi Keanggotaan Kurva Penyusutan:

$$S(x; \alpha, \beta, \gamma) = \begin{cases} 1 & \rightarrow x \leq \alpha \\ 1 - 2((x - \alpha) / (\gamma - \alpha))^2 & \rightarrow \alpha \leq x \leq \beta \\ 2((\gamma - x) / (\gamma - \alpha))^2 & \rightarrow \beta \leq x \leq \gamma \\ 0 & \rightarrow x \geq \gamma \end{cases}$$

6. Kurva bentuk Lonceng, terdiri atas 3 bagian: yaitu Kurva Pi, Kurva Beta dan Kurva Gauss

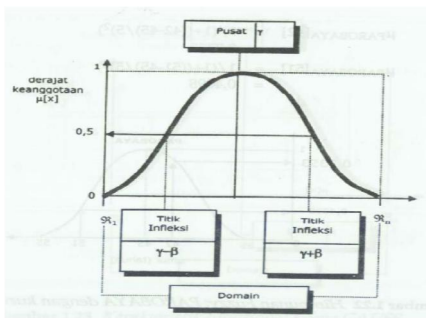
- Kurva π berbentuk lonceng dengan derajat keanggotaan 1 terletak pada pusat dengan domain (γ), dan lebar kurva (β)



Fungsi Keanggotaan:

$$\Pi(x, \beta, \gamma) = \begin{cases} S(x; \gamma - \beta, \gamma - \frac{\beta}{2}, \gamma) & \rightarrow x \leq \gamma \\ 1 - S(x; \gamma, \gamma + \frac{\beta}{2}, \gamma + \beta) & \rightarrow x > \gamma \end{cases}$$

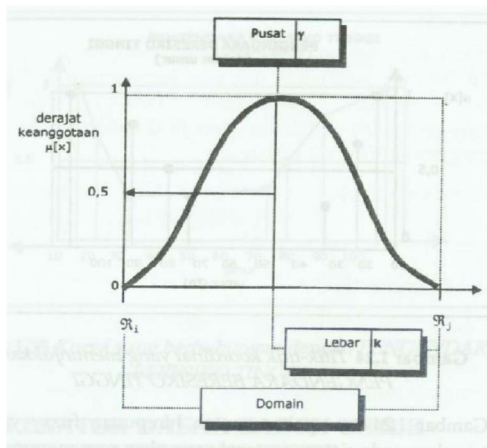
- Seperti halnya kurva PI, kurva BETA juga berbentuk lonceng namun lebih rapat. Kurva ini juga didefinisikan dengan 2 parameter, yaitu nilai pada domain yang menunjukkan pusat kurva (γ), dan setengah lebar kurva (β)



Fungsi Keanggotaan:

$$B(x; \gamma, \beta) = \frac{1}{1 + \left(\frac{x - \gamma}{\beta}\right)^2}$$

- c. Kurva GAUSS juga menggunakan (γ) untuk menunjukkan nilai domain pada pusat kurva, dan (k) yang menunjukkan lebar kurva



Fungsi Keanggotaan:

$$G(x; k, \gamma) = e^{-k(\gamma - x)^2}$$

A. Metode Mamdani

Metode inferensi fuzzy mamdani adalah metodologi yang paling sering umum digunakan dalam aplikasi, karena struktur operasi sederhana “min-max”. Metode mamdani adalah system control pertama dibangun dengan menggunakan teori fuzzy. Hal itu diusulkan pada tahun 1975 oleh Ebrahim Mamdani sebagai upaya untuk mengendalikan mesin uap dan kombinasi boiler dengan sintesis seperangkat aturan control linguistic yang diperoleh dari operator manusia yang berpengalaman. Upaya Mamdani itu didasarkan pada paper Lotfi A Zadeh 1973 tentang fuzzy algoritma untuk system yang kompleks dan proses pengambilan keputusan.

Langkah-langkah:

1. Evaluasi anteseden untuk setiap aturan.

2. Mendapatkan kesimpulan setiap aturan.
3. Kesimpulan agregat
4. Defuzzifikasi.

B. Metode Sugeno

Model fuzzy Sugeno juga di kenal sebagai model fuzzy TSK yang diusulkan Takagi, Sugeno dan Kang dalam upaya untuk mengembangkan pendekatan sistematis untuk menghasilkan aturan fuzzy dari data set input-output yang diberikan. Sebuah tipe rule fuzzy dalam sebuah model fuzzy sugeno memiliki bentuk:

Model Fuzzy Sugeno:

**Jika x adalah A dan y adalah B maka
 $z = f(x, y)$**

Dimana A dan B adalah himpunan fuzzy enteseden, sedangkan $z = f(x, y)$ adalah fungsi skrip. Biasanya $f(x, y)$ adalah polynomial dalam variable input x dan y , tetapi fungsi apapun asalkan tepat bisa menggambarkan output dari model dalam wilayah fuzzy ditentukan oleh aturan enteseden.

III. DAFTAR PUSTAKA

- [1] Dubois D, Prade H, 1980, Fuzzy Sets and Systems: Theory and Applications, Mathematics In Science And Engineering, Volume 144. Academic Press, Inc. (Diunduh tanggal: 22 Desember 2013).
- [2] Fajriana, 2001, Studi Sistem Logika Kabur Sederhana, Skripsi, FMIPA UNSYIAH, Banda Aceh.
- [3] Wulandari, Y., 2011 Aplikasi Metode Mamdani Dalam Penentuan Status Gizi Dengan Indeks Massa Tubuh (Imt) Menggunakan Logika Fuzzy, Skripsi, FMIPA, Universitas Negeri Yogyakarta. (Diunduh tanggal: 22 Desember 2013).
- [4] Net Gully, 1999, Fuzzy Logic Toolbox For Use with MATLAB. (Diunduh tanggal: 22 Desember 2013).
- [5] Sivanandam. S. N., Sumathi. S., and Deepa. S. N., 2007, Introduction to Fuzzy Logic using MATLAB, Springer. (Diunduh tanggal: 22 Desember 2013).
- [6] Zadeh LA., 1965, Fuzzy sets, Departement of Electrical Engineering and Electronics Reseach Laboratory, University of California. (Diunduh tanggal: 22 Desember 2013).
- [7] Zimmermann, HJ, 1996, Fuzzy sets and Its Applications, Third Edition, Kluwer Academic Publisher. (Diunduh tanggal: 22 Desember 2013).

Pengujian Kekuatan Nilai Hash Password Yang Dienkripsi Dengan Menggunakan Algoritma MD5

Hendrawaty¹, Azhar², Mulyadi³

¹Staff Pengajar Jurusan Teknik Elektro Politeknik Negeri Lhokseumawe

²Staff Pengajar Jurusan Teknik Elektro Politeknik Negeri Lhokseumawe

³Staff Pengajar Jurusan Teknik Elektro Politeknik Negeri Lhokseumawe

Abstrak— Baris pertahanan paling depan mengatasi para penyusup (*intruders*) adalah system password (*password system*). Hampir semua sistem yang multiuser mengharuskan pengguna (*user*) memberikan tidak hanya sebuah nama atau pengenal (ID) akan tetapi *password* juga. *Password* berfungsi untuk mengotentikasi ID dari individu yang sedang *login* ke sistem. sedangkan ID berfungsi antara lain untuk menentukan apakah pengguna berwenang untuk mendapatkan akses ke dalam sistem dan menentukan seperti apa hak akses yang diberikan kepada pengguna (hak akses penuh atau terbatas). Setiap *password* pengguna biasanya dienkrpsi (diacak dan diubah ke bentuk kode-kode yang tidak dimengerti maksudnya) untuk keperluan faktor keamanan, sebelum disimpan ke dalam suatu file atau sistem database. Fungsi hash dengan algoritma MD5 (*Message Digest Algorithm 5*) banyak digunakan untuk melakukan enkripsi password. Algoritma MD5 menerima masukan berupa pesan dengan ukuran sembarang dan menghasilkan sebuah keluaran yaitu *message digest* dengan panjang 128 bit. berapapun panjang *string* yang dienkrpsi hasilnya tetap 128 bit. Pada Makalah ini akan dibahas pengujian kekuatan password yang telah dienkrpsi dengan menggunakan algoritma MD5, pengujian dimadsukkan untuk mengetahui sejauh mana kekuatan password-pasword yang telah terenkrpsi tersebut. Pengujian dilakukan dengan cara mengenkrpsi setiap password yang akan diuji, kemudian dilakukan proses dekripsi (memecahkan hasil enkripsi ke bentuk semula) untuk mendapatkan password yang sebenarnya (*plaintext*).

Keywords— *kekuatan password, MD5, pengujian password, enkripsi*

I. PENDAHULUAN

Hampir semua sistem yang *multiuser* mengharuskan pengguna (*user*) memberikan nama pengguna (*username*) atau pengenal (*userID*) dan *password* ketika akan login ke dalam sistem. *UserID* dan *Password* digunakan sebagai baris pertahanan paling luar atau paling depan dalam mencegah para penyusup (*intruders*) masuk ke sistem. *Password* berfungsi untuk meng-otentikasi ID dari individu yang

sedang *login* ke sistem. sedangkan ID berfungsi antara lain untuk menentukan apakah pengguna berwenang untuk mendapatkan akses ke dalam sistem dan menentukan seperti apa hak akses yang diberikan kepada pengguna (hak akses penuh atau terbatas). Untuk meningkatkan faktor keamanan, password-password pengguna harus dienkrpsi.

Enkripsi adalah teknik penyandian pesan, yaitu mengubah bentuk pesan asli (*plaintext*) menjadi kode-kode tertentu yang teracak dan tidak dapat dimengerti lagi maksudnya (*chiphertext*). biasanya password-password pengguna disimpan di dalam suatu file atau sistem database. Apabila Password tidak dienkrpsi maka dapat dengan mudah dibaca, dan digunakan oleh pihak-pihak tertentu yang tidak berhak (*unauthorized user*) untuk masuk ke dalam sistem atau aplikasi. Untuk meningkatkan faktor keamanan, password-password pengguna dienkrpsi terlebih dahulu sebelum disimpan ke dalam file atau sistem database. Admin atau pihak-pihak tertentu hanya dapat melihat *password* yang terenkrpsi bukan password yang sebenarnya. *password* yang terenkrpsi tersebut tentu saja tidak dapat digunakan untuk proses *login* ke sistem. Proses enkripsi itu sendiri membutuhkan suatu algoritma agar dapat mengubah password asli atau password yang sebenarnya (*plaintext*) menjadi password yang tersandikan (*chiphertext*).

Fungsi hash dengan algoritma MD5 masih banyak digunakan untuk mengenkrpsi password. Algoritma ini akan mengubah *password* yang panjangnya sembarang menjadi *password* dengan ukuran tetap yaitu 128 bit atau 32 karakter yang terdiri dari bilangan hexadesimal. berapapun panjang input yang diberikan, outpunya akan selalu tetap.

Password terenkrpsi dengan panjang 32 karakter yang dihasilkan oleh proses enkripsi menggunakan algoritma MD5 apakah sudah cukup aman, sehingga tidak dapat dimanfaatkan oleh pihak-pihak tertentu yang tidak berhak ?. Pembahasan pada makalah ini ditujukan untuk menjawab pertanyaan tersebut, dengan cara melakukan pengujian kekuatan password-password yang dihasilkan dari proses enkripsi menggunakan fungsi hash dengan algoritma MD5.

II. TINJAUAN PUSTAKA

Pada bagian ini akan dibahas tentang Kriptographi, Aspek-aspek Keamanan informasi, Fungsi Hash, Algoritma MD5, Pengamanan Password, dan Kekuatan Password. Untuk lebih jelasnya dapat dilihat pada uraian berikut ini.

Kriptografi

Kriptografi, secara umum adalah ilmu dan seni untuk menjaga kerahasiaan berita. Selain pengertian tersebut terdapat pula pengertian ilmu yang mempelajari teknik-teknik matematika yang berhubungan dengan aspek keamanan informasi seperti kerahasiaan data, keabsahan data, integritas data dan autentikasi. Akan tetapi tidak semua aspek keamanan informasi bisa ditangani oleh kriptografi.

Aspek-Aspek Keamanan Informasi

Beberapa aspek keamanan informasi [1] yang juga merupakan tujuan dari kriptografi antara lain adalah:

- Confidentiality- yaitu kerahasiaan- agar pesan tidak dapat dibaca oleh pihak-pihak yang tidak berhak.
- Data Integrity, menjamin keaslian atau keutuhan pesan/data (belum pernah dimanipulasi selama pengiriman)
- Authentication, untuk mengidentifikasi kebenaran pihak-pihak yang berkomunikasi
- Non-Repudiation, untuk mencegah entitas yang berkomunikasi melakukan penyangkalan

Serangan Brute Force

Brute force merupakan salah satu teknik yang digunakan untuk membobol suatu sistem keamanan contohnya untuk membobol/memecahkan suatu password yang digunakan untuk login kesuatu sistem, kunci maupun kode. Teknik serangan ini menggunakan percobaan terhadap semua kunci yang mungkin.

Pengamanan Password

Pada penjelasan sebelumnya telah disebutkan bahwa hampir semua system yang multiuser menggunakan *userID* dan *password* sebagai baris pengamanan paling depan untuk mengamankan sistemnya dari pihak yang tidak berhak login. Dan untuk meningkatkan faktor keamanan, password-pasword pengguna perlu dienkripsi terlebih dahulu sebelum disimpan dalam suatu database atau file khusus. Apabila password tidak dienkripsi maka akan dengan mudah dapat dibaca, dan digunakan oleh orang lain yang tidak berhak.

Enkripsi adalah teknik penyandian pesan asli (dapat dimengerti) yang disebut dengan *plaintext* menjadi pesan yang teracak (tidak dimengerti) yang disebut dengan *chiphertext* [1]. Pengenkripsian password umumnya dilakukan dengan menggunakan fungsi hash atau yang dikenal dengan istilah one-way fuction. Algoritma yang digunakan pada fungsi hash dapat berupa MD5 dan SHA. Algoritma inilah yang digunakan untuk mengenkripsi password sebelum di simpan dalam suatu sistem database atau dalam suatu file khusus.

Password yang dimasukkan pengguna pada saat login, akan dihitung nilai hashnya. Kemudian nilai hash tersebut dibandingkan dengan nilai hash password si pengguna yang ada di dalam database. Apabila kedua nilai hash tersebut cocok/sama, maka pengguna diijinkan login ke sistem, demikikan sebaliknya apabila nilai hash dari password yang dimasukkan tidak cocok/ sama dengan nilai hash password si pengguna yang telah disimpan (ketika user mendaftar) maka akses ditolak (*acces denied*), dengan demikian pengguna tidak diijinkan masuk ke dalam sistem.

Fungsi Hash (Hash Function)

Fungsi hash adalah fungsi yang secara efisien mengubah *string* input dengan panjang berhingga menjadi string output dengan panjang tetap yang disebut nilai hash, hal ini dapat dilihat dari persamaan berikut:

$$h = H(m) \dots\dots\dots(1)$$

dimana :

- h = nilai hash
- H =fungsi hash
- m = string input

Fungsi hash H, merupakan fungsi satu arah (*one way function*) sebab sulit untuk dibalikkan yang berarti untuk nilai fungsi hash h, sulit ditemukan nilai input x yang memenuhi persamaan:

$$H(x) = m \dots\dots\dots(2)$$

Perubahan satu bit saja dari input yang akan di hash akan menghasilkan keluaran yang sangat berbeda. Hash function umumnya digunakan untuk menjamin integritas data dan digital signature.

Fungsi Hash Kriptographis

Fungsi Hash kriptographis adalah fungsi hash yang memiliki beberapa sifat keamanan tambahan sehingga dapat dipakai untuk tujuan keamanan data. Umumnya digunakan untuk keperluan autentikasi dan integritas data. beberapa contoh algoritma fungsi hash kriptographis antara lain seperti MD4, MD5, SHA-0, SHA-1, SHA-256, SHA-512.

Algoritma MD5 dan SHA masing-masing merupakan pengembangan dari algoritma MD4. Algoritma MD5 menerima masukan berupa pesan dengan ukuran sembarang dan menghasilkan nilai hash atau *message diigest* yang panjangnya 128 bit, sedangkan algoritma SHA menghasilkan nilai hash yang panjangnya 160 bit.

Algoritma MD5 (Message Digest Algorithm 5)

Algoritma MD5 adalah fungsi hash kriptografis yang dibuat oleh Ron Rivest yaitu seorang professor dari MIT. Algoritma MD5 merupakan pengembangan dari algoritma MD4. Algoritma MD5 menerima masukan berupa pesan dengan ukuran sembarang dan menghasilkan sebuah nilai hash atau *message digest* dengan panjang 128 bit. berapapun *string* yang dienkripsi hasilnya tetap 128 bit. untuk lebih jelasnya dapat dilihat pada contoh berikut [5]:

$$\begin{aligned} MD5("") &= d41d8cd98f00b204e9800998ecf8427e \\ MD5("The quick brown fox jumps over the lazy dog") &= 9e107d9d372bb6826bd81d3542a419d6 \end{aligned}$$

Plaintext pertama yang diinputkan kosong (""), berarti panjang karakternya 0. hasil enkripsi atau nilai hash yang dihasilkan dari *plaintext* pertama adalah: d41d8cd98f00b204e9800998ecf8427e (32 karakter atau 128 bit).

Plaintext kedua yang diinputkan adalah "The quick brown fox jumps over the lazy dog". Nilai hash yang

dihasilkan dari *plaintext* kedua tersebut adalah: 9e107d9d372bb6826bd81d3542a419d6 (32 karakter atau 128 bit) . Dari kedua contoh tersebut dapat dilihat bahwa panjang keluarannya tetap sama yaitu 32 karakter atau 128 bit, walaupun panjang pesan yang diinputkan berbeda.

Perubahan yang kecil pada pesan asli yang akan di hash akan menghasilkan keluaran yang sangat berbeda. Sebagai contoh, kata *dog* pada pesan “*The quick brown fox jumps over the lazy dog*” diganti dengan *cog*, maka nilai hash yang dihasilkan adalah:

MD5(“*The quick brown fox jumps over the lazy cog*”) = 1055d3e698d289f2af8663725127bd4b

Penggunaan Algoritma MD5

Pada penjelasan sebelumnya ada disebutkan bahwa penggunaan algoritma MD5 salah satunya adalah untuk mengenkripsi *password* yang diinputkan pengguna ketika *login* ke dalam sistem. selain itu, penggunaan MD5 antara lain untuk menguji integritas suatu file atau yang dikenal dengan istilah *MD5 checksum*. Pengujian ditujukan untuk memastikan suatu file tidak disisipi atau corrupt (hilang sebagian) atau mungkin terinveksi, baik itu karena virus, *malware*, atau injeksi *software* berbahaya lainnya. dan MD5 juga digunakan untuk aplikasi tanda tangan digital (*digital signature*)

Sistem Password

Umumnya *user* membuat password dari tanggal lahir, nama anak, nama artis favorit, kata yang berasal dari kamus, urutan angka seperti 1234 dan huruf yang dieja dari belakang (kesemuanya menghasilkan password yang lemah). Ada beberapa hal/faktor yang perlu dipertimbangkan dalam hal pemilihan password, agar password kuat tidak mudah dibobol atau dipecahkan [6].

Panjang karakter password merupakan salah satu faktor yang berpengaruh kepada kekuatan password itu. password yang panjangnya 4 karakter lebih lemah dibanding dengan password yang panjangnya diatas 8 karakter. Logika sederhananya, semakin panjang password maka waktu dan usaha yang diperlukan untuk membobolnya akan lebih lama dibandingkan password yang pendek. selain panjang password, kombinasi karakter yang digunakan untuk membuat password juga berpengaruh terhadap kekuatannya.

Password yang hanya terdiri dari huruf saja, atau angka saja lebih mudah dipecahkan (*password* lemah) dibanding dengan *password* yang memiliki kombinasi huruf, angka dan special karakter. Beberapa contoh password yang kuat seperti t3wahsetyeT4 (tidak terdapat di kamus, dan terdiri dari kombinasi alpha dan numerik), 4pRte!ai@3 (tidak berasal dari kamus, terdiri dari alpha dan numeric serta tanda baca), *Convert 100£ to Euros!* (tidak terdapat di kamus, terdiri dari alpha, numeric, tanda baca dan lambing). *Password* paling tidak sebaiknya

- paling tidak terdiri dari 8 karakter atau lebih
- paling tidak terdiri dari huruf besar, huruf kecil, nomor atau numerik dan symbol seperti ! * () : |
- tidak terdapat dalam kamus kata
- tidak terdiri dari NetID anda

- tidak terdiri dari karakter yang berurutan seperti AAA, abc, 123, 321, dll

III. METODE PENELITIAN

Pada bagian ini akan dibahas tentang pemilihan password-password yang akan diuji, software yang digunakan untuk menghasilkan nilai hash MD5, dan *software* yang digunakan untuk melakukan proses dekripsi atau pemecahan nilai hash menjadi *password* asli, dan bagaimana proses pengujian password dilakukan. Penjelasan lebih detail dapat dilihat pada uraian-uraian berikut ini.

Hardware dan Software yang digunakan

Pada penelitian ini digunakan *hardware* maupun *software* untuk melakukan pengujian. *Hardware* yang digunakan pada penelitian ini terdiri dari seperangkat komputer *dual core*, dan modem yang digunakan untuk koneksi ke jaringan internet. *Software* yang digunakan terdiri dari: satu buah *software* aplikasi untuk melakukan enkripsi *plaintext* menjadi *chipertext*, dan 2 buah *software* aplikasi untuk melakukan dekripsi pada *chipertext* yang dihasilkan. Ketiga *software* aplikasi yang digunakan merupakan *software* aplikasi *online* pada suatu website yang umumnya digunakan untuk melakukan proses dekripsi, dengan URL :

1. <http://www.md5hashgenerator.com/index.php>
2. <http://www.md5decrypter.co.uk/>
3. <http://www.md5crack.com>

Pemilihan Password

Perancangan *password* yang akan diuji dibuat sedemikian rupa sehingga dapat mewakili hal-hal yang akan diteliti dan dianalisa. jumlah *password* yang diuji keseluruhannya sebanyak 100 buah. Tabel 1. memperlihatkan 32 password (dari 100) yang digunakan sebagai *plaintext*.

Tabel 1. Pemilihan plaintext untuk password

No.	Password (Plaintext)
1	AKU
2	123
3	ANDI
4	IDNA
5	IKAZ
6	ALFAROMEIO
7	LEONIDAS
8	CODE
9	CODING
10	HONEY
11	KEYBOARD
12	MESIN
13	MOUSE
14	WALKING
15	CRAK
16	CRACKING
17	PERUMAHAN
18	MERCUSUAR
19	BINTANGSAMUDRA
20	123456789

21	26021970
22	ANDI123
23	ALFAROME0123
24	alfaromeo
25	AlfaROmeo
26	AN&DI
27	ANDI!
28	AnDi
29	an&DI
30	AAAAAAAAAAAA
31	AAAAAAAAAAAAAAAAAAAA
32	AAAA&AAAA
33	arnold
34	jamesbond

Dari Tabel 1 dapat dilihat bahwa pemilihan *plaintext* disesuaikan dengan password-password yang biasanya dibuat oleh pengguna, yaitu *password* yang dibuat dari tanggal lahir, nama anak, ejaan yang dibacat terbalik, urutan angka, kata yang terdapat dalam kamus, dan password-password yang dibuat sedemikian rupa sehingga terdiri dari kombinasi huruf besar dan kecil, huruf dan angka, serta symbol tertentu atau tanda baca.

Pembangkitan Message Digest

Nilai hash atau *Message Digest* dari *password* yang akan diuji diperlihatkan pada Tabel 2.

Tabel 2. Nilai Hash *Password*

No	Plaintext	Nilai Hash
1	AKU	5c1eb349da635b47a9f98aa8da5927aa
2	123	202cb962ac59075b964b07152d234b70
3	ANDI	39298c98110ba1e8b00cb7024e1241d3
4	IDNA	39298c98110ba1e8b00cb7024e1241d3
5	IKAZ	5473cbaa46bc4ed700020f518ad5b150
6	ALFAROME0	7f2fdec245f247c68716db7fc2f9d6a
7	LEONIDAS	ba25e07dbd13fb3e49dcf60a6561e8bb
8	CODE	08054846bbc9933fd0395f8be516a9f9
9	CODING	e7383517dfeea878b01f37e2b130cb78
10	HONEY	70dbad0fb90c4fcdae13ca4fe3a56947
11	KEYBOARD	edda266cdb6345b9f6914cc2e7577475
12	MESIN	208b93856eb0ce9c2ea01ae568a934fe
13	MOUSE	2ec2be318cc86a1224c0e9dec71321e4
14	WALKING	606c114184493a665cflf6a12fbab9d3
15	CRAK	2e6d9801a7e8f255fd21bb5c34d946ae

16	CRACKING	1b02103bf91f68fb8f7b9b791f68a1e6
17	PERUMAHAN	d4c1964bd795b4a5bf5abfcea07f4a10
18	MERCUSUAR	01cfc4c043b2f98993c326a9845c4cc8
19	BINTANGSAMUDRA	a70da4f045ffe7ff1c90d4cd26c20727
20	123456789	25f9e794323b453885f5181f1b624d0b
21	26021970	89f87eb5daff0f8aa91093a6039157b4
22	ANDI123	08030677a5de6b0b72ab754d1b4aba35
23	ALFAROME0123	7ea0e8c9de8cb856c4062099ce17c70
24	alfaromeo	5458ac2651502c439bd2d11b08e23914
25	AlfaROmeo	c4a2bc8fa2ead066984ff572978dcfe2
26	AN&DI	042e2f2f727d07fd5c1488426b470cbb
27	ANDI!	8412f240cd474a3cd439843126b5e2da
28	AnDi	47594e0028ef68bb6ffaaff550e06911
29	an&DI	8b84426b14e03edce6185653f5fceff33
30	AAAAAAAAAAAA	16c52c6e8326c071da771e66dc6e9e57
31	AAAAAAAAAAAAAAAAAAAA	76d36e98f312e98ff908c8c82c8dd623
32	AAAA&AAAA	0c4e4d01ba6ea212750a628b9bd8e287
33	AAAA12AAAA	a210b72771eb3c8db6ecc339873ee9b9
34	AAAAaaAAAA	83c1b3f7f874a16766473cc2f362c904
35	ANDI (MD5 3KALI)	dfd095bae29f5a0eb62b2da02ac580b1
36	arnold	49a30d03c669a09f2c01c3655032af3e
37	jamesbond	e84c55c90d955bf1cfa2d31a1f425383

Pengujian Password Yang Terenkripsi

Pengujian dilakukan dengan cara menginputkan hasil enkripsi atau nilai hash dari setiap password asli (*plaintext*) ke Aplikasi dekripsi MD5 online, kemudian mengumpulkan /mencatat hasilnya. Masing-masing nilai hash diuji 2 kali, dengan menggunakan 2 aplikasi pendekrip md5 yang berbeda. selain untuk melihat kekuatan dari nilai hash *password*, pengujian juga dimaksudkan untuk menganalisa kinerja *software* aplikasi online yang ada dalam mendekripsi nilai hash yang dihasilkan oleh algoritma MD5.

IV. HASIL DAN PEMBAHASAN

Pada bagian ini akan disampaikan hasil dan pembahasan dari pengujian-pengujian yang telah dilakukan. baik pengujian untuk melihat kekuatan nilai hash maupun kinerja dari *software* aplikasi dekripsi online yang digunakan.

Hasil Pengujian Kekuatan Password Terenkripsi

Tabel 3., memperlihatkan nilai hash dari password yang bisa dibuka kembali menjadi plainteks dan yang tidak dapat dicrack dengan menggunakan software aplikasi online yang telah disebutkan sebelumnya.

Tabel 3. Hasil Dekripsi Nilai Hash

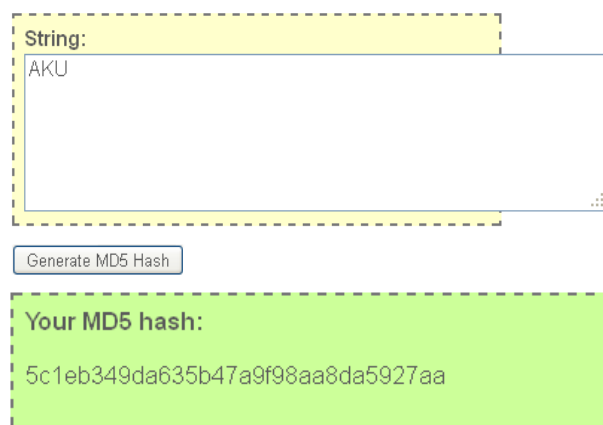
No	Password (Plaintext)	Nilai Hash	Dapat dicrack
1	AKU	5c1eb349da635b47a9f98aa8da5927aa	Ya
2	123	202cb962ac59075b964b07152d234b70	Ya
3	ANDI	39298c98110ba1e8b00cb7024e1241d3	Ya
4	IDNA	39298c98110ba1e8b00cb7024e1241d3	Ya
5	IKAZ	5473cbaa46bc4ed700020f518ad5b150	Ya
6	ALFAROME0	7f2fdec245f247c68716db7fcc2f9d6a	Ya
7	LEONIDAS	ba25e07dbd13fb3e49dcf60a6561e8bb	Ya
8	CODE	08054846bbc9933fd0395f8be516a9f9	Ya
9	CODING	e7383517dfeea878b01f37e2b130cb78	tdk
10	HONEY	70dbad0fb90c4fcdae13ca4fe3a56947	Ya
11	KEYBOARD	edda266cdb6345b9f6914cc2e7577475	Ya
12	MESIN	208b93856eb0ce9c2ea01ae568a934fe	Ya
13	MOUSE	2ec2be318cc86a1224c0e9dcc71321e4	Ya
14	WALKING	606c114184493a665cf1f6a12fbab9d3	Ya
15	CRAK	2e6d9801a7e8f255fd21bb5c34d946ae	Ya
16	CRACKING	1b02103bf91f68fb8f7b9b791f68a1e6	Tdk
17	PERUMAHAN	d4c1964bd795b4a5bf5abfcea07f4a10	Tdk
18	MERCUSUAR	01cfc4c043b2f98993c326a9845c4cc8	Tdk
19	BINTANGSAMUDRA	a70da4f045ffe7ff1c90d4cd26c20727	Tdk
20	123456789	25f9e794323b453885f5181f1b624d0b	Ya
21	26021970	89f87eb5daff0f8aa91093a6039157b4	Ya
22	ANDI123	08030677a5de6b0b72ab754d1b4aba35	Tdk
23	ALFAROME0123	7ea0e8c9de8bcb856c4062099ce17c70	Tdk
24	alfaromeo	5458ac2651502c439bd2d11b08e23914	Ya
25	AlfaRomeo	c4a2bc8fa2ead066984ff572978dfe2	Tdk
26	AN&DI	042e2f2f727d07fd5c1488426b470cbb	Tdk
27	ANDI!	8412f240cd474a3cd439843126b5e2da	Tdk

28	AnDi	47594e0028ef68bb6ffaa ff550e06911	Ya
29	an&DI	8b84426b14e03edce6185653f5fcef33	Tdk
30	AAAAAAAAAA	16c52c6e8326c071da771e66dc6e9e57	Ya
31	AAAAAAAAAAAA AAAAAAAAAA	76d36e98f312e98ff908c8c82c8dd623	Ya
32	AAAA&AAAA	0c4e4d01ba6ea212750a628b9bd8e287	Tdk
33	AAAA12AAAA	a210b72771eb3c8db6ec c339873ee9b9	Tdk
34	AAAAaaAAAA	83c1b3f7f874a16766473cc2f362c904	Tdk
35	ANDI (MD5 3 kali) dfd095bae29f5a0eb 62b2da02ac580b1	da3e1985903b8c897fdb bea9548a30f4	Tdk
36	Arnold	49a30d03c669a09f2c01c3655032af3e	Ya
37	jamesbond	e84c55c90d955bf1cfa2 d31a1f425383	Ya

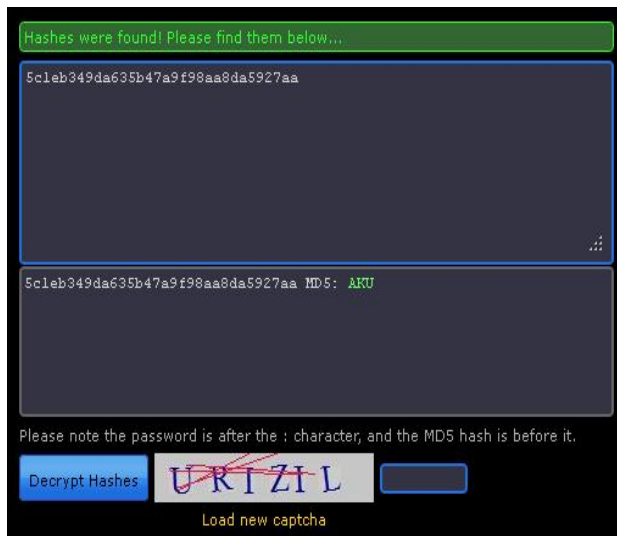
berikut ini merupakan beberapa contoh gambar tampilan yang dihasilkan software aplikasi online pada saat pengujian dilakukan.

MD5 Hash Generator

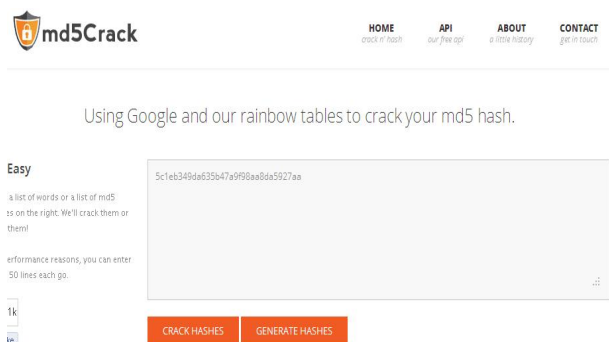
Use this generator to create an MD5 hash of a string.



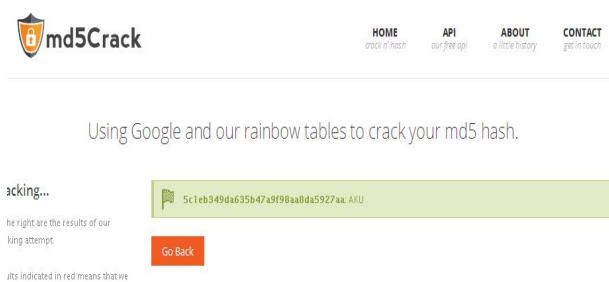
Gambar 1. nilai hash dari plaintext AKU



Gambar 2. Hasil Dekripsi dengan aplikasi md5decrypter



Gambar 3. proses dekripsi dengan aplikasi md5Crack



Gambar 4. Hasil dekripsi dengan aplikasi md5Crack

Dari hasil pengujian yang dilakukan dapat dianalisa bahwa walaupun password telah dienkripsi dengan menggunakan algoritma MD5 (panjangnya sudah 32 karakter), namun kekuatan dari password tersebut masih tetap dipengaruhi oleh bagaimana caranya password tersebut dipilih atau ditentukan. Hal ini disebabkan karena proses dekripsi yang dilakukan oleh software-software gratis yang ada di internet semuanya (berdasarkan fakta yang telah ditemukan selama penelitian dilakukan) sama yaitu

melakukan pencarian nilai hash yang cocok dengan nilai hash yang telah tersimpan dalam databasenya.

Dengan kondisi yang demikian dan dari hasil pengujian yang dilakukan dapat dikatakan bahwa password yang hanya terdiri dari huruf saja, angka saja, dan berbagai password yang dibuat dari tanggal lahir, nama anak, nama artis favorit, kata yang berasal dari kamus terutama kata-kata atau istilah dalam bahasa Inggris, urutan angka seperti 1234 dan huruf yang dieja dari belakang kesemuanya menghasilkan password yang lemah, karena usaha dan waktu yang diperlukan untuk menemukan nilai hash yang sesuai relative cepat.

Sementara password yang berasal dari hasil enkripsi menggunakan MD5 secara berulang (3 kali dienkripsi), password yang ukurannya lebih dari 8 karakter yang bukan berasal dari bahasa Inggris, serta password yang memiliki kombinasi huruf, angka dan special karakter, terbukti sebagai password yang kuat, karena usaha dan waktu yang diperlukan untuk mencoba berbagai kombinasi untuk mendapatkan nilai hash yang sesuai memakan waktu yang lama.

Pengujian Kinerja Software Aplikasi Online

Setelah melakukan pengujian semua nilai hash atau *message digest* dari password, dapat diketahui bahwa proses dekripsi yang dilakukan oleh software aplikasi online bukan merupakan proses dekripsi sebagaimana proses dekripsi pada algoritma untuk kriptografi seperti DES (Algoritma yang digunakan pada *symmetric key encryption*) maupun RSA (algoritma yang digunakan pada *Asymmetric Key Encryption*) atau dengan kata lain dapat dikatakan bahwa nilai hash dari MD5 belum benar-benar dapat dienkripsi. Hal ini ditunjukkan dengan banyaknya password yang tidak dapat di dekripsi.

Software Aplikasi untuk mendekripsi nilai hash yang tersedia secara online bekerjanya hanya sebatas melakukan pencarian pada database. Jika nilai hash yang ingin di dekripsi terdapat dalam database, maka aplikasi tersebut akan menampilkan hasil yang berupa plaintext (password yang belum di dekripsi), dan jika nilai hash yang diinputkan belum ada tersimpan dalam databasenya maka aplikasi akan menampilkan pemberitahuan bahwa software tidak berhasil melakukan dekripsi terhadap nilai hash yang diinputkan.

Nilai hash yang tersimpan di dalam database diperoleh dengan cara mencoba-coba alias melakukan *brute force* saja, terhadap berbagai macam kemungkinan password. Password tersebut kemudian kemudian dienkripsi dengan menggunakan algoritma MD5, dan hasilnya yang berupa nilai hash atau *message digest* kemudian disimpan dalam database. Jumlah plaintext dan nilai hash yang tersimpan dalam database tersebut cukup banyak jumlahnya, ada database yang sudah berisi sekitar 8,5 juta kata dan nilai hash, jadi prinsipnya mirip dengan kamus.

V. KESIMPULAN

Beberapa kesimpulan yang dapat diambil dari hasil penelitian ini adalah:

- 1) Password yang hanya terdiri dari huruf saja, angka saja, dan berbagai password yang dibuat dari tanggal lahir,

- 2) nama anak, nama artis favorit, kata yang berasal dari kamus terutama kata-kata atau istilah dalam bahasa Inggris, urutan angka seperti 1234 dan huruf yang dieja dari belakang kesemuanya menghasilkan password yang lemah, karena usaha dan waktu yang diperlukan untuk menemukan nilai hash yang sesuai relative cepat.
- 3) Password yang berasal dari hasil enkripsi menggunakan MD5 secara berulang (3 kali dienkripsi), password yang ukurannya lebih dari 8 karakter yang bukan berasal dari bahasa Inggris, serta password yang memiliki kombinasi huruf, angka dan special karakter, terbukti sebagai password yang kuat.
- 4) Nilai hash dari MD5 belum benar-benar dapat diekripsi. Kekuatan nilai hash dari password masih tetap dipengaruhi oleh bagaiman caranya password tersebut dipilih atau ditentukan.
- 5) Software Aplikasi untuk mendekripsi nilai hash yang tersedia secara online bekerjanya hanya sebatas melakukan pencarian pada database (prinsipnya mirip dengan kamus).

VI. DAFTAR PUSTAKA

- [1] Stallings, William. 1995. *Network And Internetwork Security*. New Jersey: Prentice-Hall, Inc.
- [2] Software Aplikasi Online md5 Hash Generator Internet: <http://www.md5hashgenerator.com/index.php> Tanggal akses: 1 Desember 2013 pukul: 08.00
- [3] Software Aplikasi Online md5 Decrypter Internet: <http://www.md5decrypter.co.uk/> Tanggal akses: 1 Desember 2013 pukul: 08.00
- [4] Software Aplikasi Online md5 Hash Generator Internet: <http://www.md5crack.com> Tanggal akses: 1 Desember 2013 pukul: 08.00
- [5] Saiful Bahri, Diana, Susan. Studi dan Implementasi Pengamanan Basis Data Menggunakan Metode Enkripsi MD5. Internet: Diakses 28 September 2013 pukul: 17.00
- [6] Erwin, Analisa Kombinasional Pembuatan Password dan Kemungkinan Menjebolnya. Internet: Tanggal akses 28 September 2013 pukul: 20.00
<http://en.wikipedia.org/wiki/MD5> Tanggal akses: 28 September 2013 pukul: 20.00

Optimasi Waktu Pada Replikasi Dan Distribusi Database Dengan Metode Round Robin

Rizalul Akram

Program Studi Magister Teknik Informatika, Universitas Sumatera Utara
Medan, Sumatera Utara, Sumut-Indonesia

Abstrak— Bagi lembaga pendidikan tinggi ataupun perusahaan yang memiliki banyak cabang, mereka membutuhkan sistem informasi yang sangat memadai untuk kemudahan akses data dan back-up data untuk menjamin ketersediaan data secara berkelanjutan. Merupakan hal yang penting bagi lembaga pendidikan dan perusahaan yang memiliki banyak cabang untuk dapat melakukan integrasi data, yang terkadang penyimpanan data dilakukan pada masing-masing bagian atau cabang terhubung langsung dengan bagian pusat instansi. Penggunaan teknologi distribusi database sangat tepat untuk kasus ini. Dengan teknologi ini tiap-tiap bagian dari instansi di lembaga pendidikan ataupun perusahaan dapat mengakses data secara langsung, tanpa perlu harus ke pusat data. Untuk saat ini memang teknologi ini masih mempunyai banyak kekurangan dalam kinerjanya, salah satunya adalah proses replikasi dan distribusi datanya. Proses replikasi dan distribusi data menjadi permasalahan utama dalam kasus ini. Dimana lamanya waktu yang dihabiskan untuk replikasi ini disebabkan oleh cara kerja sistem yang masih harus dikembangkan lagi dan penggunaan algoritma terbaik juga sangat diperlukan untuk kasus ini. Penggunaan tabel status/temporary merupakan teknik untuk mengoptimalkan proses replikasi dan penggunaan algoritma round robin untuk mempercepat proses load data pada distribusi data.

Keywords — Distribusi database, replikasi, tabel status/temporary, round robin

I. PENDAHULUAN

Basis data terdistribusi (distributed database) adalah suatu basis data yang berada di bawah kendali sistem manajemen basis data (DBMS) terpusat dengan peranti penyimpanan (*storage devices*) yang terpisah-pisah satu dari yang lainnya. Tempat penyimpanan ini dapat berada di satu lokasi yang secara fisik berdekatan (misal dalam satu bangunan) atau terpisah oleh jarak yang jauh dan terhubung melalui jaringan lokal maupun internet. Penggunaan basis data terdistribusi dapat dilakukan melalui jaringan local maupun jaringan internet. Untuk menjaga agar basis data yang terdistribusi tetap up to date, ada dua proses untuk menjaganya, yakni replikasi dan duplikasi. Dalam replikasi, digunakan suatu perangkat lunak untuk mencari atau lebih tepatnya melacak perubahan yang terjadi di satu basis data. Setelah perubahan dalam satu basis data teridentifikasi dan diketahui, baru kemudian dilakukan perubahan agar semua basis data sama satu dengan yang lainnya. Proses replikasi memakan waktu yang lama dan membebani komputer karena kompleksitas prosesnya. Seiring dengan makin

berkembangnya distribusi database, tentunya masih banyak terdapat kekurangan disana-sini. Kekurangan yang paling mendasar adalah terletak pada di proses refleksi dan load data. Penelitian ini terfokus pada penyelesaian masalah pada refleksi dan distribusi database. Adapun langkah yang akan dilakukan untuk menyelesaikan masalah ini adalah :

- ✓ Pada proses refleksi berfokus untuk pengupdatean data di semua server *client*. Replikasi ini menggunakan metode *teknik single master replicated*.
- ✓ Pada proses distribusi database dari beberapa client server menyebabkan terjadi *time waiting* yang sangat lama dan bahkan sampai *error*. Untuk permasalahan ini penulis akan menggunakan algoritma penjadwalan round robin.

II. DASAR TEORI

a. Optimasi

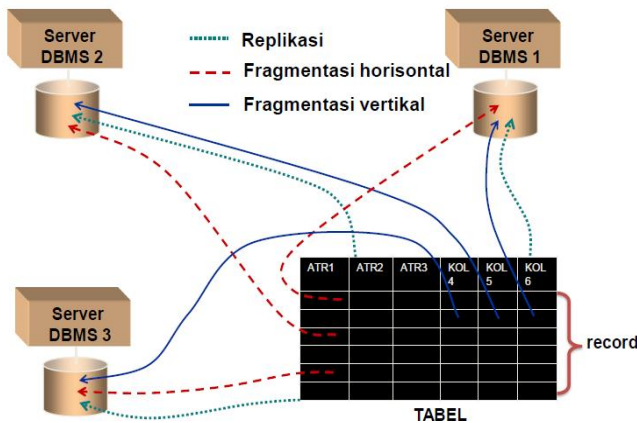
Optimasi adalah salah satu disiplin ilmu dalam Matematika yang fokus untuk mendapatkan nilai minimum atau maksimum secara sistematis dari suatu fungsi, peluang, maupun pencarian nilai lainnya dalam berbagai kasus. Optimasi sangat berguna di hampir segala bidang rangka melakukan usaha secara efektif efisien untuk mencapai target hasil yang ingin dicapai. Tentunya hal ini akan sangat sesuai dengan prinsip ekonomi yang berorientasikan untuk senantiasa menekan pengeluaran untuk menghasilkan outputan yang maksimal. Optimasi ini juga penting karena persaingan saat ini sudah benar benar sangat ketat.

b. Distribusi Database

Database terdistribusi adalah sebuah database yang diatur oleh sebuah *Database Management System* (DBMS) yang tersimpan secara fisik pada beberapa komputer yang terdapat pada beberapa lokasi dengan terkoneksi pada sebuah jaringan. Terdapat dua macam sifat dari database terdistribusi yaitu *Heterogenous* dan *Homogenous*. *Homogenous* artinya suatu database terdistribusi dimana data di distribusikan pada beberapa komputer dengan menggunakan DBMS (*database management system*) yang sama. DBMS digunakan pada database terdistribusi untuk melakukan koordinasi data pada beberapa node. Sedangkan *Heterogenous* adalah kebalikan dari *Homogenous* dimana data di sebarakan dengan menggunakan DBMS yang berbeda. Basis data terdistribusi (*distributed database*) adalah suatu basis data yang berada di bawah kendali sistem manajemen basis data (DBMS) terpusat dengan piranti penyimpanan (*storage devices*) yang

terpisah-pisah satu dari yang lainnya. Tempat penyimpanan ini dapat berada di satu lokasi yang secara fisik berdekatan (misal dalam satu bangunan) atau terpisah oleh jarak yang jauh dan terhubung melalui jaringan internet. Penggunaan basis data terdistribusi dapat dilakukan di server internet, intranet atau ekstranet kantor, atau di jaringan perusahaan. Untuk menjaga agar basis data yang terdistribusi tetap update, ada dua proses untuk menjaganya, yakni replikasi dan duplikasi. Dalam replikasi, digunakan suatu perangkat lunak untuk mencari atau lebih tepatnya melacak perubahan yang terjadi di satu basis data. Setelah perubahan dalam satu basis data teridentifikasi dan diketahui, baru kemudian dilakukan perubahan agar semua basis data sama satu dengan yang lainnya.

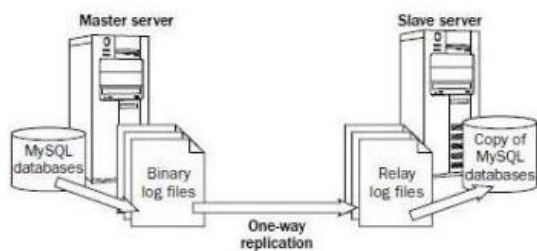
c. Gambaran Ilustrasi Replikasi, Fragmentasi Horizontal dan Vertikal



Gambar 2.1. Gambaran ilustrasi replikasi, fragmentasi horizontal dan vertikal

d. Teknik Single Master Replicated

Dengan metode ini, salah satu komputer berfungsi sebagai master dan yang lainnya berfungsi sebagai slave. Pada prosesnya, komputer yang digunakan sebagai server akan dapat *read* dan *write* ke dalam database. Sedangkan komputer yang berfungsi sebagai slave, hanya akan *read* saja kedalam database tersebut. Apabila kita melakukan perubahan data pada master, maka otomatis data pada slave akan berubah. Tetapi jika kita melakukan perubahan data pada slave, database pada master tidak akan berubah.



Gambar 2.2. Teknik Single Master Replication

e. Algoritma Round Robin

Penjadwalan Round Robin adalah penjadwalan proses yang menerapkan strategi *preemptive*, bukan di *preempt* oleh proses lain tapi terutama oleh penjadwal berdasarkan jatah waktu pemroses yang disebut kwanta. Ketentuannya adalah jika kwanta habis dan proses belum selesai maka pemroses dialihkan ke proses lain dan jika kwanta belum habis tapi proses telah selesai maka proses diakhiri dan pemroses dialihkan ke proses lain.

Algoritma *Round-robin* merupakan :

- Penjadwalan ini bukan dipreempt oleh proses lain tetapi oleh penjadwal berdasarkan lama waktu berjalannya proses (*preempt by time*).
- Penjadwalan tanpa prioritas.
- Berasumsi bahwa semua proses memiliki kepentingan yang sama, sehingga tidak ada prioritas tertentu.

Ketentuan algoritma *round-robin* adalah sebagai berikut:

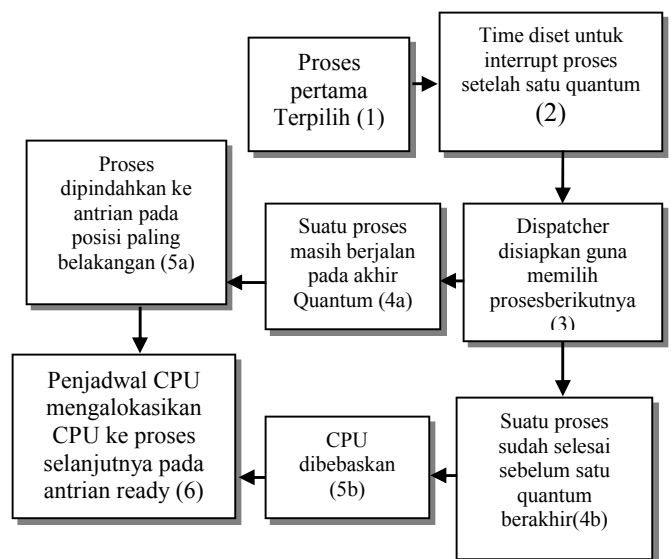
1. Jika kwanta habis dan proses belum selesai maka proses menjadi *runnable* dan pemroses dialihkan ke proses lain.
2. Jika kwanta belum habis dan proses menunggu suatu kejadian (selesainya operasi I/O), maka proses menjadi *blocked* dan pemroses dialihkan ke proses lain.

Jika kwanta belum habis tapi proses telah selesai, maka proses diakhiri dan pemroses dialihkan ke proses lain. Diimplementasikan dengan:

1. Mengelola senarai proses *ready* (*runnable*) sesuai urutan kedatangan.
2. Ambil proses yang berada di ujung depan antrian menjadi *running*.
3. Bila kwanta belum habis dan proses selesai, maka ambil proses di ujung depan antrian proses *ready*.

Jika kwanta habis dan proses belum selesai, maka tempatkan proses *running* ke ekor antrian proses *ready* dan ambil proses di ujung depan antrian proses *ready*.

Alur urutan kejadian algoritma *Round-robin*.



Gambar 2.3. Alur urutan kejadian algoritma *Round-Robin*

III. METODOLOGI PENELITIAN

Replikasi dan distribusi database yang masih lambat merupakan permasalahan yang masih belum teratasi dalam distribusi database. Pada penelitian ini penulis menggunakan teknik single master replicated dan tabel status (*temporary*) untuk memaksimalkan proses replikasi dan menggunakan algoritma penjadwalan round robin untuk mempercepat distribusi/load data saat terjadi distribusi data dalam jumlah banyak pada waktu bersamaan dari server client ke server data center. Adapun bahan yang digunakan pada penelitian ini adalah berupa data acak data mahasiswa. Rancangan jaringan untuk simulasi seperti pada gambar 3.1

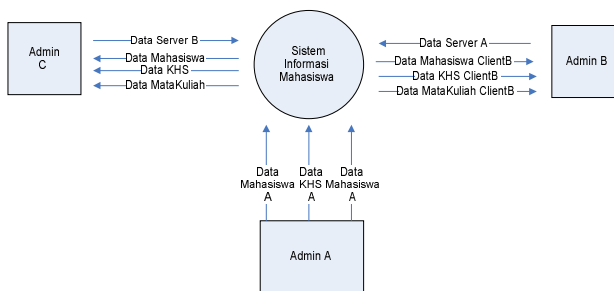
linkdown dia akan mencari jalur lain untuk mencapai tujuannya.

4. Pada proses distribusi database dari server masterA ke server clientB menggunakan algoritma round robin. Tiap data yang diinputkan dari master masing-masing diatur jatah waktu perjurusan untuk didistribusikan ke server clientB. Begitu juga proses *scedulling* dari server client B ke C.

IV. PERANCANGAN SISTEM

Data Flow Diagram (DFD) merupakan diagram yang menggambarkan aliran data pada aplikasi sistem optimasi waktu pada replikasi dan distribusi database dengan metode round robin. DFD Level Konteks sering digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut mengalir atau lingkungan fisik dimana data tersebut disimpan.

a. Diagram konteks



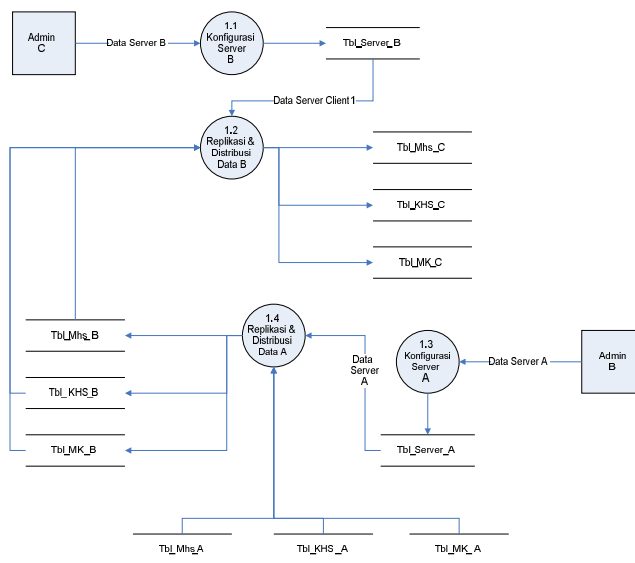
Gambar 4.1. Diagram konteks

Gambar 3.1. Rancangan topologi jaringan yang di bangun untuk simulasi

Secara garis besar alur kerja jaringan ini adalah sebagai berikut :

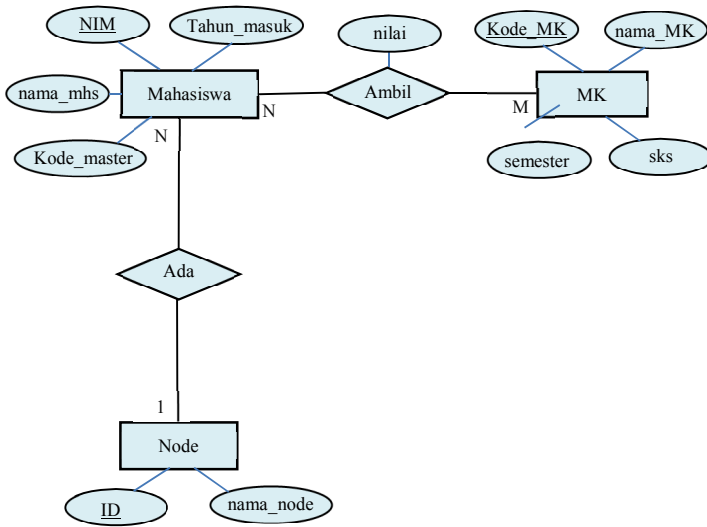
1. Jaringan ini mempunyai dua fasilitas, yaitu fasilitas untuk mahasiswa dan admin tiap server
2. Pada saat mahasiswa ingin menginput data/pengisian KRS (kartu rencana studi) maka sistem akan secara otomatis mengirim data tersebut ke server jurusan masing-masing dengan membaca NIM (nomor induk mahasiswa) dari mahasiswa tersebut yang diberikan saat registrasi manual. NIM merupakan *primary key* dari entitas mahasiswa. Setelah data tersebut masuk ke server jurusan, kemudian di distribusikan ke server clientB masing-masing dan kemudian ke server clientC. Digunakan topologi mesh supaya sistem dalam jaringan akan tetap berjalan walaupun jika terjadi gangguan pada salah satu *link* (salah satu link *down*). Dan pada topologi ini sistem akan menggunakan routing dinamis, jadi bisa dipastikan jika salah satu

b. Data Flow Diagram (DFD) level 1



Gambar 4.2. DFD level 1

c. Entity Relational Diagram (ERD)



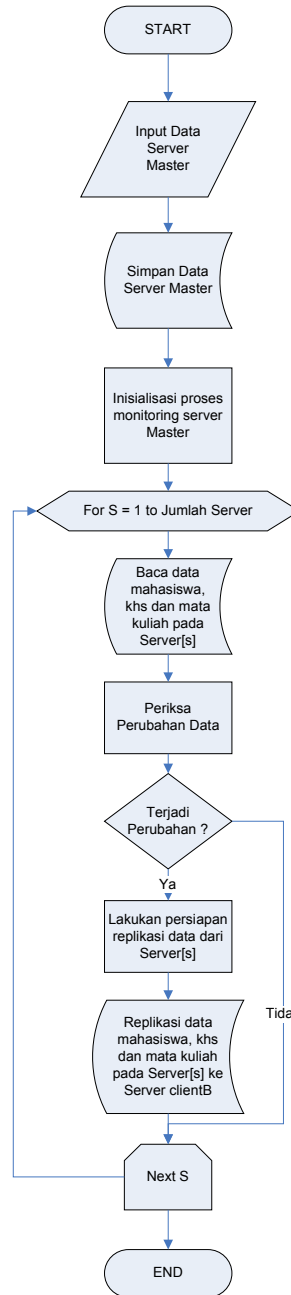
Gambar 4.3.ERD

Dari gambar diatas selanjutnya dilakukan proses mapping dan didapatkan tabel sebagai berikut :

- ✓ Mahasiswa (NIM*, nama_mhs, alamat_mhs, Idserver*)
- ✓ MK (kode_MK*, nama_MK, sks, semester)
- ✓ KHS (NIM*, kode_MK*, nilai)
- ✓ Node (ID, nama_node).

d. Proses Replikasi

Proses replikasi ini menggunakan teknik *single master replicated*. Teknik ini memberikan hak kepada master server A untuk melakukan perubahan data. Jika perubahan/*update* data dilakukan oleh master server maka akan dilakukan perubahan secara otomatis di server node lain yang linear. Contoh linear disini adalah masterA1 linear dengan clientB1 dan C. Adapun penjelasan detail dari proses replikasi ini tergambar pada gambar 6.

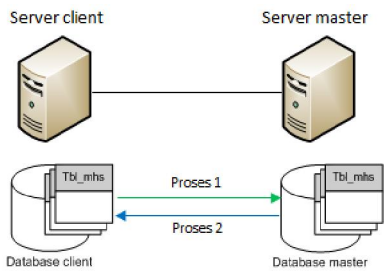


Gambar 4.4.Flowchat proses replikasi

V. HASIL DAN ANALISA

5.1. Analisa Sistem Saat Ini

Menurut hasil kajian yang dilakukan, sistem saat ini melakukan proses replikasi ini seperti pada gambar 4.1. Dimana sistem saat ini membaca *record* satu persatu beserta meng*updatenya*. Jadi setiap perubahan yang terjadi akan langsung tereplikasi. Sistem seperti ini akan memakan waktu yang lama jika datanya sudah sangat banyak.



Tabel Client

Nim	Nama	Tahun masuk
117010	Rizal	2012
117011	Defry	2012
117012	Iqbal	2012
117013	Ihsan	2012
117014	Ivan	2012

Tabel Master

Nim	Nama	Tahun masuk
117010	Rizal	2012
117012	Defry	2012
117014	Iqbal	2012
117016	Ihsan	2012
117018	Ivan	2012

Proses 1: Database client reads data from Database master.
Proses 2: Database client updates data in Database master.

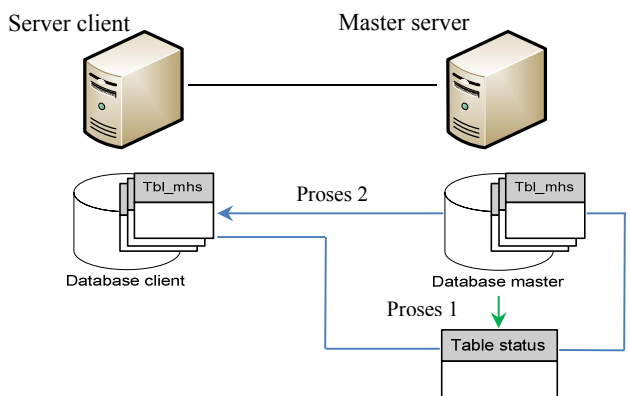
Gambar 5.1. Analisa sistem saat ini

Keterangan

1. Proses 1: database client yang *sincronised* dengan database master, membaca langsung data pada database master satu-persatu
2. Proses 2: melakukan *update* data yang telah dibaca

5.2. Analisa Sistem Pengembangan

Analisa sistem replikasi yang dikembangkan adalah seperti pada gambar 4.3. Sistem ini tidak mendeteksi perubahan baris data/*record* satu persatu. Tetapi setiap perubahan/*update* yang terjadi langsung masuk ke tabel status. Tabel status ini sama halnya seperti *temporary*, yang menampung data sementara. Fungsi tabel status ini adalah untuk menampung informasi perubahan dari database master yang selalu dibaca oleh database client dan node dibawahnya. Dan setelah proses tiap-tiap replikasi data selesai maka semua data yang tertampung disini akan dihapus otomatis dan begitu seterusnya. Tabel database client membaca perubahan/*update* database master di tabel status master. Baru kemudian database client menuju *record* perubahan tersebut di database master dan kembali untuk melakukan perubahan/*updating*.



Gambar 5.2. Skema kerja sistem yang dibangun

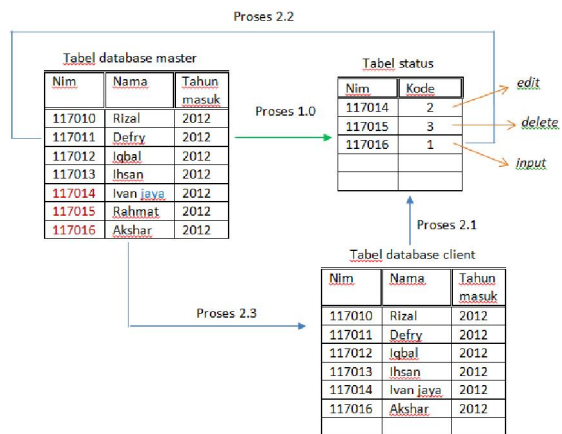
Contoh kerja dari sistem ini adalah seperti pada gambar 5.2 dan 5.3. Tabel 5.1 adalah salah satu contoh tabel yang ada di database data master dan client sebelum proses replikasi terjadi.

Tabel 5.1. Contoh data yang ada di database master dan client

Nim	Nama	Tahun masuk
117010	Rizal	2012
117011	Defry	2012
117012	Iqbal	2012
117013	Ihsan	2012
117014	Ivan	2012
117015	Rahmat	2012

Nim	Nama	Tahun masuk
117010	Rizal	2012
117011	Defry	2012
117012	Iqbal	2012
117013	Ihsan	2012
117014	Ivan	2012
117015	Rahmat	2012

Proses replikasi berjalan seperti pada gambar 5.3. Nim yang diberi warna merah menunjukkan yang *record* yang mengalami perubahan.



Gambar 5.3. Analisa sistem saat ini

Keterangan kode di tabel status

- Kode 1: input data
- Kode 2: edit data
- Kode 3: hapus data

Keterangan proses :

- Proses 1.0 : Database master memasukan Nim (*primary key*) data yang berubah ke tabel status
- Proses 2.1 : Database ClientB selalu membaca setiap data yang masuk ke tabel status
- Proses 2.2 : Database data client memeriksa perubahan data yang terjadi di database data master
- Proses 2.3 : Database data client melakukan perubahan/*updating*.

5.1. Simulasi

Simulasi pada penelitian ini dilakukan dengan jaringan yang terkoneksi secara LAN. Tujuan simulasi ini untuk pengujian sistem yang telah dibangun. Simulasi ini berfokus pada distribusi database dan replikasi database. Aplikasi yang dibangun menggunakan dua fasilitas. Fasilitas master server

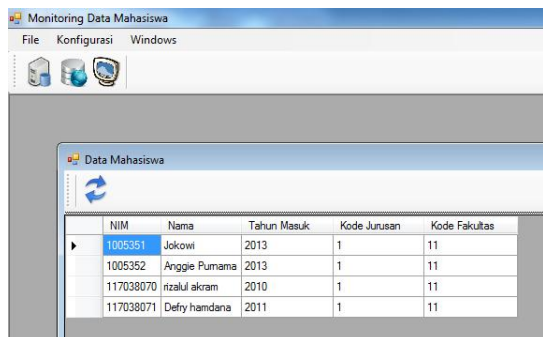
adalah server A, dimana di server ini bisa dilakukan input data, edit, dan hapus. Server/node B dan C hanya bisa view data saja. Adapun skema jaringan yang telah dilakukan untuk percobaan ini ada seperti pada gambar 5.2. Dan pengaturan IP server untuk kedua jaringan di atas seperti pada tabel 5.2.

Tabel 5.2. Pengaturan IP server

Server	Network	IP Server	IP-Range
A (Master)	192.168.1.0/24	192.168.1.5	192.168.1.1 – 192.168.1.254
B (Client)	192.168.1.0/24	192.168.1.10	192.168.1.1 – 192.168.1.254

a. Distribusi Database Dari Node A Sampai Ke C

Tabel mahasiswa structural terdiri dari kolom Nim, Nama, Tahun Masuk, Kode Master/jurusan dan KodeB/fakultas. Tabel mahasiswa akan didistribusikan ke tingkat ClientB dan clientC masing-masing dengan mendistribusikan kolom/field kode master/jurusan untuk tingkat ClientB dan field clientB/fakultas untuk Client C. Distribusi/load data ini menggunakan menggunakan algoritma round robin untuk mempercepat prosesnya. Setiap data yang didistribusikan menggunakan menggunakan limit waktu yang berbeda. Limit waktu tranfer data server master ke clientB sebesar 30 sekon dan limit waktu transfer data dari server clientB ke server clientC sebesar 120 sekon. Pengaturan demikian di karenakan data dari tiap clientB lebih besar dari data tiap master. Berikut adalah uji program distribusi data mulai dari server masterA sampai ke server clientC. Sistem distribusi ini menggunakan metode pull. Metode ini menggunakan satu database master server, dimana di server inilah data di input dan kemudian di distribusikan ke server-server/node di bawahnya. Di penelitian ini master servernya adalah server master. Server clientB dan clientC adalah server client yang hanya menerima data/view saja. Gambar 10 menunjukkan tabel mahasiswa ditingkat clientC, yang sudah dilengkapi dengan field kode fakultas.



Gambar 5.4. Tampilan data mahasiswa didatabase clientC

Proses *scedulling* round robin hanya digunakan untuk sistem yang dikembangkan. Algoritma ini digunakan untuk proses distribusi data. Sistem ini menggunakan algoritma ini

untuk load data ke berbagai *client/node*. Pengujian ini dilakukan terhadap 400 *record* data yang diload dengan waktu *time sharing* sebesar 0,25s atau 250ms dan selesai dalam waktu 455ms. Proses ini selesai dalam dua tahap dengan waktu jeda load kembali sekitar 5 ms. Hasil pengujian ini seperti pada tabel 5.3.

Tabel 5.3. Hasil pengujian *scedulling* round robin

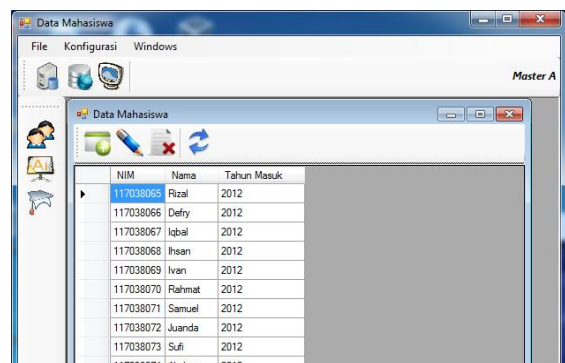
Tahap	Record	Waktu (sekon)
1	222	250ms
2	178	205ms
Total	400	455ms

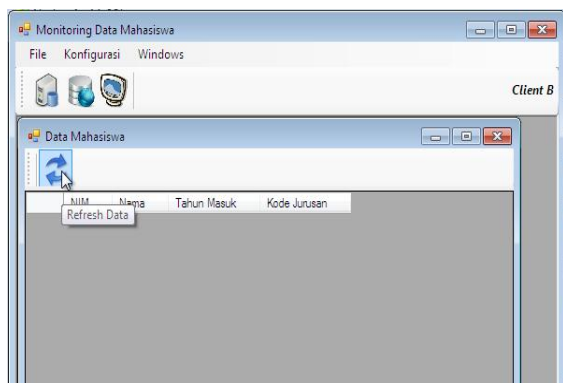
b. Replikasi Database Dari Server Master Sampai Ke Node C

Pengujian akan dilakukan untuk membandingkan kecepatan dari proses replikasi biasa dan proses replikasi yang diajukan pada penelitian ini. Pengujian menggunakan alur replikasi dari data mahasiswa dari server master ke server clientB. Pengujian data mahasiswa dari server clientB ke server clientC tidak akan dilakukan mengingat diperkirakan hasilnya akan sama dengan pengujian replikasi data mahasiswa dari server master ke server clientB karena menggunakan proses yang sama. Proses replikasi biasa akan menggunakan tabel duplikat dari tabel proses replikasi yang dikembangkan sehingga dapat terlihat proses dari replikasi yang dilakukan. Replikasi ini menggunakan teknik *single master replicated*, dimana hanya terdapat satu server master saja. Server clientB dan clientC sebagai server client/node, dimana kedua server ini hanya bisa *view* data saja. Sistem replikasi ini menggunakan fasilitas monitoring untuk mengontrol selalu setiap perubahan yang terjadi di server/node di bawahnya. Fasilitas monitoring ini aktif terus menerus supaya replikasi bekerja secara otomatis.

1. Pengujian Replikasi Input Data

Sebagai pengujian diasumsikan tidak ada data pada database masterA dan clientB. Kemudian di inputkanlah 16 (enam belas) data mahasiswa pada databse server masterA dan data pada database server clientB masih kosong.

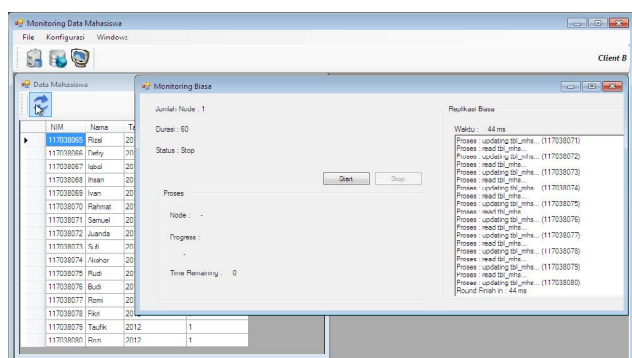




Gambar 5.5. Data setelah di input di masterA dan belum tereplikasi ke clientB

Proses pengujian dilakukan dengan dua tahap, yaitu :

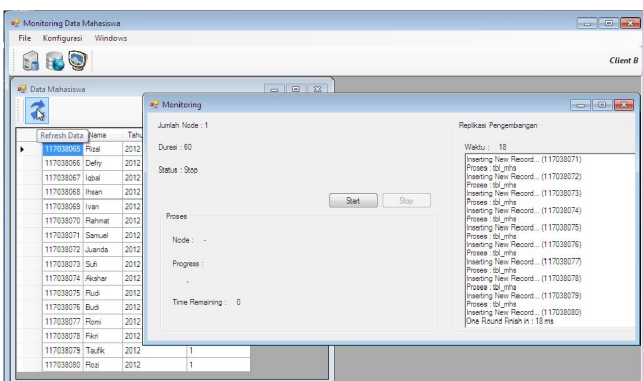
- Monitoring sistem replikasi saat ini



Gambar 5.6. Proses replikasi sistem saat ini pada data input

Pada gambar 5.6, proses replikasi membaca *record* satu persatu dan kemudian melakukan inserting data baru jika data tersebut tidak terdapat pada database ini. Setelah proses membaca *record* dari nim 117038065 dan seterusnya, proses melakukan *inserting* karena data tersebut tidak terdapat pada database server. Proses replikasi data sebanyak 16 *record* pada sistem ini membutuhkan waktu sebesar 44ms (milisekon).

- Monitoring replikasi pengembangan



Gambar 5.7. Proses replikasi pengembangan pada data input

Gambar 5.7 adalah proses replikasi pengembangan yang lebih efektif dimana proses langsung mengidentifikasi *record* data baru dan kemudian melakukan *inserting* data tersebut pada database ini. Proses replikasi inputan 16 *record* data pada sistem ini menghabiskan waktu 18ms.

Setelah hasil penelitian yang dilakukan di atas, dipaparkan kembali hasilnya dalam tabel 5.4. Tabel ini berisi hasil replikasi sistem saat ini, sistem yang di kembangkan, dan persentase penghematan waktu. Persentase penghematan waktu ini adalah perbandingan dari waktu replikasi yang dihabiskan oleh sistem saat ini dan sistem yang di kembangkan. Persentase ini tidaklah tetap, tergantung dari banyaknya data yang diupdate di sistem pengembangan. Waktu yang dihabiskan oleh sistem yang di kembangkan sangat tergantung dari jumlah data yang diupdate, tidak sama halnya dengan sistem saat ini, yang membaca dan mengupdate semua data yang ada. Tabel 5.4 adalah hasil percobaan yang dilakukan terhadap 16 *record* data.

Tabel 5.4. Hasil pengujian proses replikasi

Proses	Data yang diupdate (record)	Banyak Data (record)	Hasil (waktu)		
			Sistem saat ini	Sistem pengembangan	Persentase penghematan waktu
Input data	16	0	44ms	18ms	59%
Edit data	4	16	43ms	4ms	91%
Delete data	4	16	38ms	5ms	87%

2. KESIMPULAN

Dari hasil penelitian yang dilakukan didapatkan kesimpulan sebagai berikut :

- Untuk data yang sama sebanyak 16 *record* diperoleh waktu semakin baik/cepat pada replikasi dengan metode pengembangan. Baik pada proses *input*, *edit* dan *delete* data. Penggunaan tabel status/temporary cukup efektif untuk mempercepat waktu pada proses replikasi
- Proses distribusi data dengan metode *scedulling* round robin pada metode pengembangan dapat berjalan dengan baik, dengan hasil uji coba terhadap 400 *record* data selesai dalam dua tahap dengan pengaturan *time sharing* sebesar 0,25s atau 250ms.

VI. DAFTAR PUSTAKA

- [1] Arun Kumar Yadav, Dr. Ajay Agarwal, *A Distributed Architecture for Transactions Synchronization in Distributed Database Systems*, Vol. 02, No. 06, 2010, 1984-1991, (IJCSSE) International Journal on Computer Science and Engineering
- [2] Ms.P.R.Bhuyar, Dr.A.D.Gawande & Prof. A. B. Deshmukh, *Horizontal Fragmentation Technique in Distributed Database*, Volume 2, Issue 5, May 2012, International Journal of Scientific and Research Publications (ISSN 2250-3153)

- [3] Shahidul Islam Khan & Dr. A. S. M. Latiful Hoque, A New Technique for Database Fragmentation in Distributed Systems, Volume 5– No.9, August 2010, International Journal of Computer Applications (0975 – 8887)
- [4] Shatha J. Kadhim and Kasim M. Al-Aubidy, Design and Evaluation of a Fuzzy-Based CPU Scheduling Algorithm, Computer Eng. Dept, Philadelphia University, Amman, Jordan
- [5] Swati Gupta, Kuntal Saroha, Bhawna, Fundamental Research of Distributed Database, Vol. 11, Issue 02, Aug 2011, IJCSMS International Journal of Computer Science and Management Studies ISSN (Online): 2231-5268
- [6] Rakesh Kumar Yadav, Abhishek K Mishra, Navin Prakash and Himanshu Sharma, An Improved Round Robin Scheduling Algorithm for CPU scheduling, Vol. 02, No. 04, 2010, 1064-1066, (IJCE) International Journal on Computer Science and Engineering
- [7] Rizky Tahara Shita, Model Akses Database Terdistribusi Berdasarkan Pendekatan Services Oriented Architecture, Vol.2 No.2, September 2010, Jurnal Telematika Mkom, ISSN 2085-725X
- [8] Yin-Fu Huang And Jyh-Her Chen, Fragment Allocation in Distributed Database Design, Journal Of Information Science And Engineering 17, 491-506 (2001)
- [9] Özsu, M.T & Valduriez, Principles of Distributed Database Systems, (e-ISBN 978-1-4419-8834-8)

Replikasi Standby Database Menggunakan Metode Incremental Backup

Defry Hamdhana¹, Sayed Fachrurrazi²

¹Program Studi Teknik Informatika, Universitas Sumatera Utara

²Program Studi Teknik Informatika, Universitas Malikussaleh
Reuleut, Aceh Utara, Aceh-Indonesia

E-mail: defryhamdhana@gmail.com

Abstrak— Penelitian ini membahas tentang metode yang dapat diterapkan pada sebuah sistem server database (primary) untuk melakukan replikasi secara incremental backup dan selalu tersinkronisasi pada server database cadangan (standby database). Standby database mengaktifkan failover ketika server primary database mengalami failure (kerusakan). Sehingga standby database menjadi primary database yang baru.

Pada simulasi ini aplikasi dibangun dengan menggunakan VB.Net dan MySQL Front. Serta menggunakan data sampel acak yang penulis unduh dari internet. Aplikasi terdiri atas 2 bagian. Yang pertama berupa aplikasi pergudangan yang berada pada komputer klien (admin), dan yang kedua berupa aplikasi master yang berada pada server primary database.

Hasil dari simulasi ini adalah melakukan pembuktian terhadap replikasi yang terjadi antara primary database dengan standby database secara realtime. Dan aktifnya failover yang membuat standby database menjadi primary database yang baru saat primary database mengalami kerusakan. Sehingga transaksi data yang dilakukan oleh klien (admin) dapat berjalan dengan normal.

Keywords— Primary Database, Standby Database, Incremental Backup, Failover

I. PENDAHULUAN

Kebutuhan akan sistem database yang semakin meningkat menjadikan data menjadi aset yang bernilai tinggi. Dengan demikian penting untuk menjaga data agar tetap ada kapan saja dibutuhkan. Akan tetapi resiko terjadinya kerusakan (failure) pada database yang mengakibatkan data tidak dapat diakses atau bahkan mengakibatkan data loss dapat terjadi setiap saat. Gangguan tersebut dapat berupa maintenance, kerusakan database, kerusakan media dan data corruption. Database juga dapat rusak akibat adanya bencana alam seperti kebakaran, gempa bumi dan banjir.

Kerugian yang terjadi dari rusaknya database secara tiba-tiba akan sangat besar. Hal ini dapat diatasi apabila database yang rusak dapat langsung digantikan oleh database cadangan (secondary) sehingga sistem dapat berjalan dengan

baik. Oleh sebab itu dibutuhkan sebuah disaster recovery plan yang dapat menjamin data tetap konsisten walaupun database mengalami gangguan bahkan failure (kerusakan).

Adapun teknik penyelamatan data yang sering dilakukan, yaitu backup data. Backup data adalah teknik menyalin data ke dalam media lain. Dalam penelitian ini berarti data yang berada pada primary database (utama) disalin ke dalam secondary database (cadangan). Jika primary database mengalami kerusakan maka data tetap aman di dalam secondary database. Dan secondary database akan mengembalikan (restore) data apabila primary database sudah siap untuk digunakan kembali. Namun demikian hal ini belum dapat menjadi solusi terbaik. Karena backup tidak menggantikan kinerja primary database secara langsung yang mengakibatkan data tidak dapat diakses sampai maintenance pada sistem primary database tersebut berakhir.

Kemampuan standby database adalah untuk melakukan recovery dan backup pada primary database[9]. Standby database dapat melakukan backup data dari lingkungan geografis yang berbeda, maka apabila terjadi hal-hal yang dapat menyebabkan database pada primary database rusak karena suatu hal, standby database dapat mengambil fungsi peran dari primary database yang mengalami kerusakan. Dan oleh sebab itu penulis melakukan riset dan analisis terhadap standby database sebagai disaster recovery.

Tujuan penelitian ini adalah membuat standby database yang mampu melakukan backup data secara real time pada lingkungan DBMS MySQL dan menggantikan primary database pada saat primary database mengalami kerusakan (failure) serta menguji waktu respon dan throughput pada proses replikasi dari primary database server ke standby database server yang diharapkan mampu mengatasi kehilangan data jika terjadi gangguan seperti bencana alam, kerusakan database dan kerusakan media.

II. DASAR TEORI

a. Replikasi

Replikasi dicapai dengan memiliki sistem standby, yang merupakan duplikasi dari database produksi. Replikasi standby diperbaharui setelah database produksi memanipulasi data, sehingga membuat sistem standby sangat dekat dengan sistem utama [7]. Replikasi adalah suatu teknik untuk melakukan copy dan pendistribusian data dan objek-objek database dari satu database ke database lain

dan melakukan sinkronisasi antara *database* sehingga konsistensi data dapat terjamin [10].

Pada dasarnya sistem replikasi membutuhkan minimal dua buah server untuk digunakan sebagai *master* dan *slave*. Dengan menggunakan teknik replikasi, data dapat didistribusikan ke lokasi yang berbeda melalui koneksi jaringan lokal maupun internet.

b. Primary Database

Primary database adalah *database* utama yang digunakan untuk menyimpan data. *Database* ini diharapkan mampu diakses setiap saat. Maka dari itu jika terjadi transisi *role* dan *primary database* mati, maka secara otomatis pengaksesan terhadap *primary database* tersebut juga tidak dapat dilakukan.

c. Standby Database

Standby database merupakan replikasi *database* yang terdiri dari backup data dari sebuah *primary database*. Dengan penerapan *redo log archive* dari *primary database* untuk *standby database*, sehingga data dapat di simpan di dalam server *database* yang tersinkronisasi. Sebuah *standby database* memiliki tujuan utama sebagai *disaster recovery*, *backup*, analisis, dan *reporting* (laporan). Jika *primary database* hancur atau rusak, *admin* dapat melakukan *failover* ke *standby database*, dalam kasus ini *standby database* menjadi *primary database* yang baru. *Admin* juga dapat membuka *standby database* dengan opsi *read only*, sehingga dapat berfungsi sebagai sebuah *database reporting* independen.

d. Failover

Pada konfigurasi *standby database* semua pengguna aplikasi melakukan transaksi pada *primary database*. Namun apabila server *primary database* mengalami kegagalan secara tidak terduga, maka *admin* harus melakukan *failover*. *Failover* adalah operasi yang mengubah *standby database* menjadi *primary database* sehingga dapat berjalan secara normal. Operasi ini juga disebut aktivasi *standby database*. Penting untuk diperhatikan bahwa setelah melakukan *failover*, *admin* tidak dapat mengembalikan *standby database* yang saat ini telah menjadi *primary database* untuk menjadi *standby database* kembali.

III. Metodologi

Penelitian ini menggunakan metode penelitian eksperimen (*Experimental Research*). Pengumpulan data dan informasi dengan menggunakan metode observasi yaitu meninjau dan mengamati secara langsung pada *primary database server* dan *standby database server* untuk mendapatkan gambaran tentang replikasi yang terjadi serta aktifnya *standby database server* pada saat *primary database server* rusak.

3.1. Alat Penelitian dan Bahan

Pada penelitian ini alat penelitian yang digunakan berupa perangkat keras dan perangkat lunak sebagai berikut:

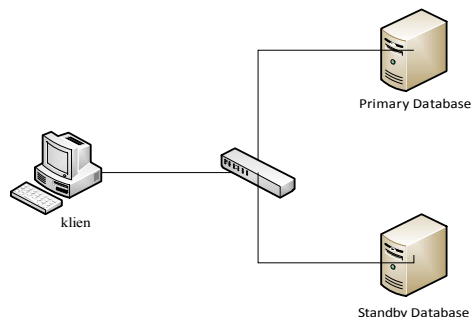
- a. Perangkat keras (hardware)
 - 1. Satu unit *switch*.

- 2. Dua unit *personal computer* Processor Intel(R) Core(TM)2Duo 1.80 GHz untuk *database server*.
- 3. Satu unit *personal computer* Processor Intel(R) Core(TM)2Duo 1.80 GHz untuk klien.

- b. Perangkat lunak (software)
 - 1. Sistem Operasi Windows 7 untuk *database server* dan klien.
 - 2. Microsoft Visual Studio 2010 Professional
 - 3. MySQL untuk *database server*.
 - 4. XAMPP V 1.3.1 Win32
 - 5. Bandwith Meter Pro V 2.6.0

Bahan yang digunakan pada penelitian ini adalah berupa data acak pada sebuah gudang supermarket yang penulis unduh dari internet.

Adapun skenario jaringan untuk pengujian sistem ini adalah seperti pada gambar 1 sebagai berikut:



Gambar 3.1. Rancangan Topologi Jaringan yang dibangun untuk Simulasi

Secara garis besar alur kerja jaringan ini adalah sebagai berikut:

- 1. Jaringan ini mempunyai dua server, yaitu server untuk *primary database* dan server untuk *standby database*. Dan satu klien yang bertindak sebagai *admin* pergudangan.
- 2. Pada saat *admin* pergudangan ingin meng-*input* data barang maka sistem akan secara otomatis mengirim data tersebut ke server *primary database*. Setiap data yang berada di dalam *primary database* akan dimasukkan ke dalam *archived log file* dan kemudian akan dikirimkan ke *standby database* dalam proses replikasi data.
- 3. Topologi jaringan yang digunakan adalah topologi star untuk semua jaringan. Dan pada setiap jaringan akan diberikan *IP Static* kelas C yang akan menghubungkan ketiga jaringan.

3.2. Proses Penelitian

Proses penelitian ini meliputi pengumpulan data, perancangan skenario simulasi, pembuatan eksperimen simulasi berdasarkan *flowchart*, dan analisa data yang terangkum sebagai berikut:

a. Pengumpulan Data

Pengumpulan data dilakukan melalui penelitian laboratorium dengan membuat jaringan yang terisolasi menggunakan 1 (satu) unit *switch* dan 2 (dua) unit *datas*

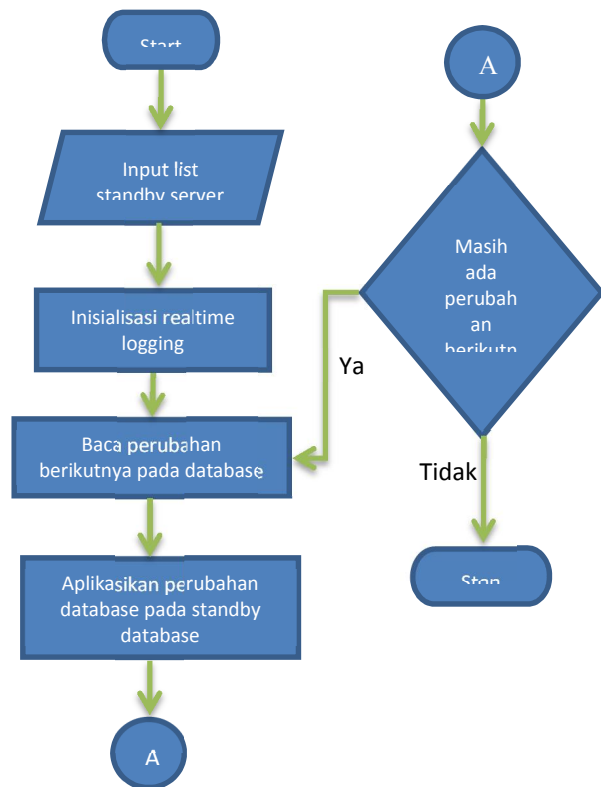
server dan 1 (satu) unit klien dengan rincian 1 (satu) switch untuk menghubungkan jaringan *primary database server*, *secondary database server* dan klien.

b. Rancangan Skenario Simulasi

Skenario simulasi pengujian secara umum terdiri atas dua bagian yaitu pengujian tentang replikasi (*logging*) dan pengujian *failover*. Sebagai langkah awal masing-masing laptop harus terhubung secara *Local Area Network (LAN)* pada kelas *Internet Protocol (IP)* yang sama.

IV. PERANCANGAN SISTEM

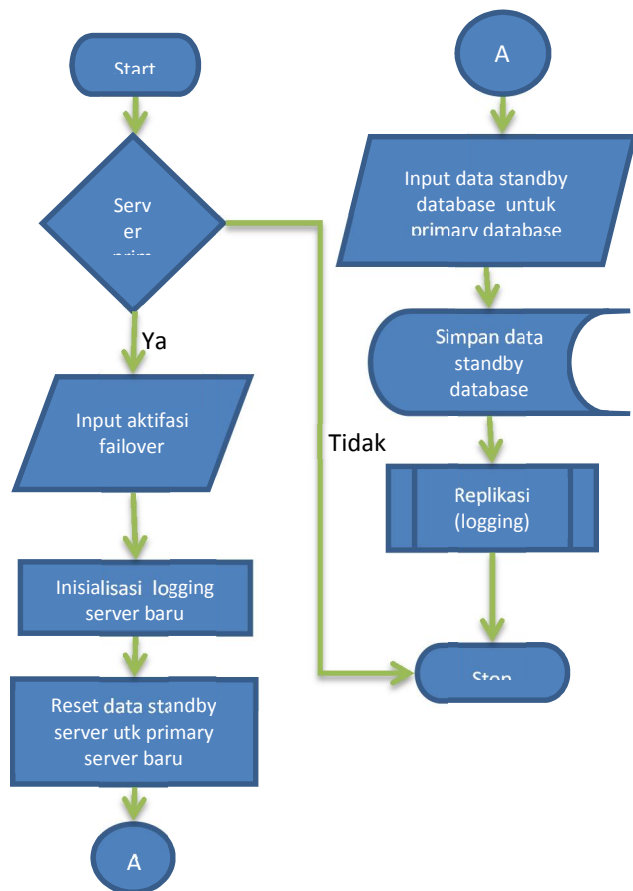
Pada proses terjadinya replikasi, awalnya *admin* menginput data yang ditujukan untuk *primary database*. Setelah itu *admin* mengaktifkan *realtime logging* untuk mereplikasi data dari *primary database* ke *standby database*. *Realtime logging* bertugas untuk membaca setiap perubahan yang terjadi pada *primary database*. Semua perubahan baik itu *input* data baru, *edit*, ataupun *delete* yang tercatat pada *realtime logging* akan diaplikasikan kepada *standby database*. Kemudian *realtime logging* akan memeriksa kembali apakah ada perubahan data lainnya di dalam server *primary database*. Jika ada maka proses pereplikasian data masih akan terus terjadi. Hal ini akan terus terjadi hingga *realtime logging* menyatakan bahwa *primary database* sudah tidak melakukan perubahan baik itu *input* data, *edit* data, ataupun *delete* data. Berikut adalah *flowchart* dari replikasi (*logging*) yang terjadi antara *primary database* dan *standby database*.



Gambar 4.1. Flowchart pada sebuah Replikasi (Logging)

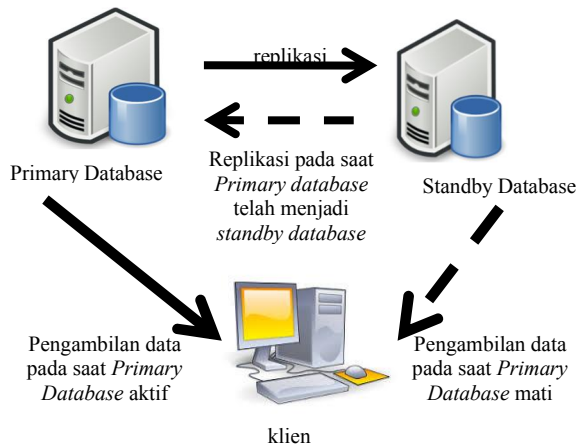
Secara *default* user akan mengakses data melalui *primary database*. Hal ini akan terus terjadi apabila *primary database* selalu dalam keadaan normal. Namun apabila

primary database mengalami kerusakan (*failure*), maka *standby database* akan menggantikan fungsi *primary database*. *Standby database* menjamin bahwa data yang dimiliki tidak akan hilang ketika *primary database* rusak. Hal ini disebabkan karena setiap transaksi yang terjadi antara klien dan *primary database* akan langsung direplikasikan ke server *standby database* dengan cara mengumpulkan *log file* di dalam *archived log*. Selama *primary database* dalam perbaikan, maka semua klien masih dapat mengakses data secara normal menggunakan *standby database*. Proses pengalihan (*switch*) *standby database* menjadi *primary database* disebut *failover*. Untuk melihat gambaran kerja *failover* melakukan teknik *switching* untuk mengalihkan fungsi *standby database* menjadi *primary database* dapat dilihat pada gambar 3.



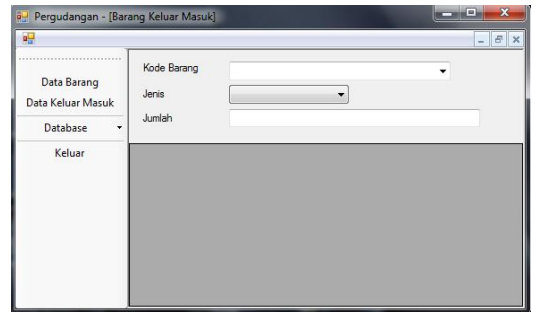
Gambar 4.2. Flowchart pada Proses Failover

Untuk dapat mengetahui gambaran umum proses *standby database* dapat dilihat pada gambar 3.4.

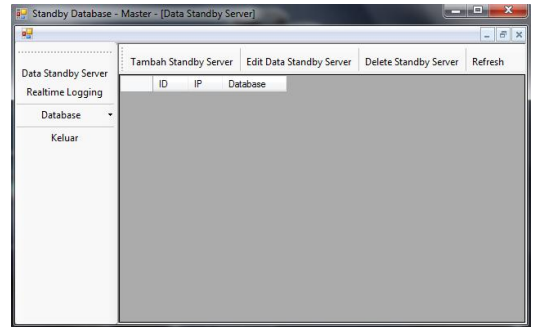


Gambar 4.3. Proses Standby Database

Klien mengambil atau menginput data melalui *primary database*. Data yang di-input ke *primary database* tereplikasi ke *standby database*. Pada saat *primary database* rusak, *standby database* langsung menggantikan kerja sistem *primary database*. Sehingga klien melakukan transaksi data ke *standby database*. Proses transaksi data kembali ke *primary database* pada saat *primary database* dapat berjalan secara normal. Serta data yang ada pada *standby database* kembali direplikasi ke *primary database*.

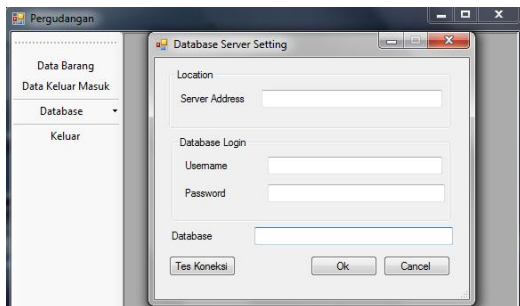


Gambar 5.3. Data Keluar Masuk Barang

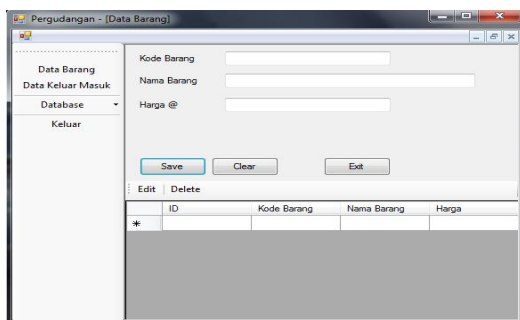


Gambar 5.4. Data Standby Server

V. TAMPILAN APLIKASI



Gambar 5.1. Input IP Address Primary Server

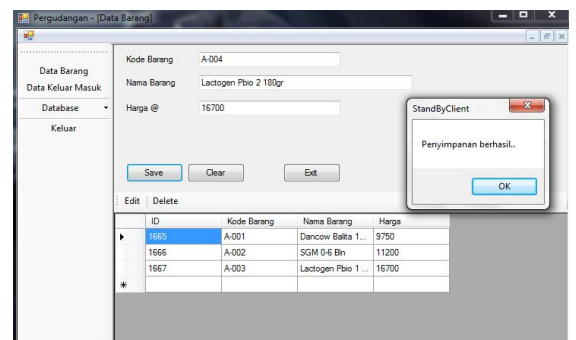


Gambar 5.2. Input Data Barang

VI. HASIL UJI COBA

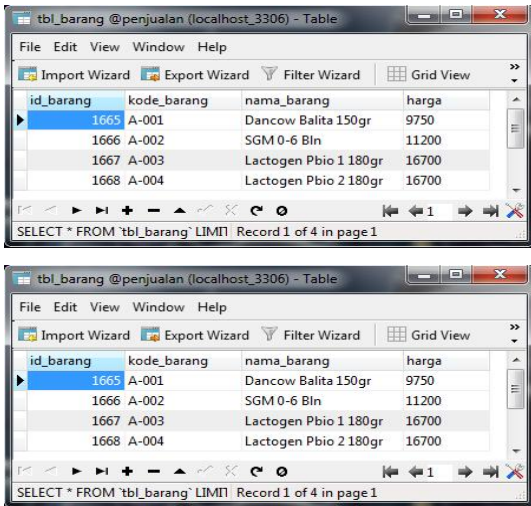
Peneliti melakukan simulasi di laboratorium dengan menggunakan jaringan LAN yang terisolasi dan melakukan penelitian terhadap tiga skenario. Tujuan simulasi ini dilakukan untuk mengumpulkan data yang diperlukan, yaitu mengetahui proses replikasi (*logging*) yang terjadi antara *primary database* dan *standby database*, proses aktifnya *failover* untuk membuat *standby database* menjadi *primary database* yang baru karena *primary database* yang sebelumnya mengalami kerusakan (*failure*), serta menghitung jumlah waktu respon dan *throughput* yang dihasilkan pada saat *primary database* mengirimkan *archived log* ke *standby database* dan jumlah waktu respon serta *throughput* yang terjadi pada saat *standby database* yang baru melakukan replikasi terhadap *primary database* yang baru.

Berikut adalah proses transaksi data yang terjadi dari admin menuju server *primary database* dan kemudian direplikasikan secara *realtime* menuju server *standby database*.



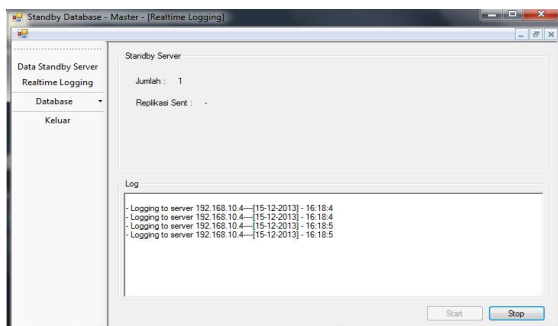
Gambar 6.1. Input Data

Admin memasukkan data barang ke dalam aplikasi pergudangan yang akan di simpan di dalam *primary database*. Namun sebelumnya *admin* harus melakukan pengaturan IP server yang menuju server *primary database* yaitu 192.168.10.1. Setelah data berhasil disimpan penulis melakukan pemeriksaan terhadap *primary database* dan *standby database*.



Gambar 6.2. (a) Data pada *Primary Database*, (b) Data pada *Standby Database*

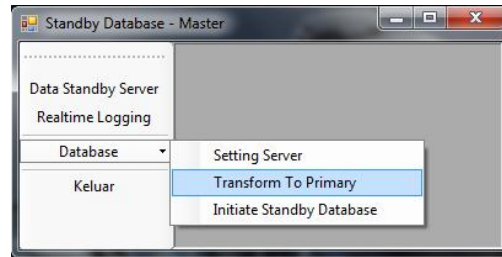
Setiap transaksi data yang di-*input*, *edit* maupun *delete* akan masuk ke *primary database* dan tereplikasi secara *incremental backup* menuju server *standby database*. Sehingga replikasi terjadi secara *realtime*. Berikut adalah gambar tabel *logging* yang membuktikan bahwa proses replikasi berjalan secara *realtime*.



Gambar 6.3 Tabel *Logging*

Dari tabel *logging* dapat dilihat bahwa proses replikasi yang terjadi dari server *primary database* menuju server *standby database* berjalan secara *realtime*. Karena data yang di-*input* oleh *admin* dari komputer klien yang masuk ke server *primary database* juga langsung ter-*input* di dalam server *standby database*.

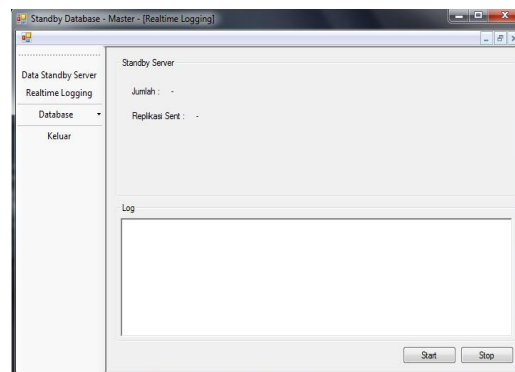
Fungsi lain dari *standby database* adalah menggantikan *primary database* yang mengalami kerusakan. Pada saat *primary database* rusak, *standby database* akan menjadi *primary database* yang baru. Sehingga transaksi data yang dibutuhkan oleh *admin* tidak mengalami gangguan. Berikut adalah proses aktifnya failover sehingga *standby database* menjadi *primary database* yang baru:



Gambar 6.4. Proses Aktifnya *Failover*

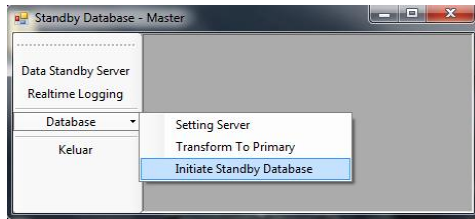
Proses failover dilakukan pada server *standby database* dengan cara mengaktifkan menu *transform to primary*. Sebelum melakukan failover, IP server harus di setting menjadi 192.168.10.4, karena server ini yang kedepannya akan menjadi server *primary database*. Sekarang *admin* melakukan transaksi data melalui server *primary database* yang baru. *Admin* tidak dapat mengembalikan fungsi *primary database* yang lama untuk menjadi *primary database* kembali dengan cara *role back*. *Primary database* yang lama akan disiapkan menjadi *standby database* yang baru. Dengan demikian *database* tersebut dapat menjadi *primary database* kembali apabila *primary database* yang berjalan saat ini juga mengalami kerusakan (*failure*). Berikut adalah langkah kerja aplikasi master mempersiapkan *database* yang telah siap untuk digunakan kembali menjadi *standby database*:

- Pastikan proses *realtime logging* pada server *primary database* yang baru dalam keadaan tidak aktif. Hal ini dilakukan agar proses reset pada *standby database* yang baru dapat berjalan dengan baik. Sebab reset tidak akan selesai apabila masih ada data yang terkirim dari *primary database*.



Gambar 6.5. Tabel *Realtime Logging* tidak Aktif

- Jalankan aplikasi master pada *standby database* yang baru. Dan kemudian set *IP Address* pada *standby database* yang baru. Kemudian klik *initiate standby database* pada menu aplikasi master yang terdapat pada server *standby database*.



Gambar 6.6. Proses *Initiate Standby Database*

Semua data di dalam *standby database* harus di format (hapus) sebelum melakukan teknik *initiate standby database*. Hal ini dilakukan untuk menghindari redundansi data yang mungkin terjadi apabila *standby database* yang baru mereplikasi data yang ada pada *primary database*. Fungsi utama dari *initiate standby database* adalah mereplikasi dari awal semua data yang ada pada *primary database* yang baru.

- c. Proses *realtime logging* dapat diaktifkan kembali. Pada proses ini *standby database* yang baru sudah dapat digunakan kembali. Dengan demikian proses *standby database* telah selesai dilakukan.

VII. KESIMPULAN

Berikut adalah beberapa kesimpulan yang penulis ambil dari penelitian ini:

1. Proses replikasi yang terjadi dari server *primary database* menuju *standby database* berjalan secara *realtime*. Hal ini karena replikasi dilakukan dengan metode *incremental backup*. Maka setiap ada perubahan data yang terjadi pada *primary database* akan diikuti oleh *standby database* (tersinkronisasi).
2. *Primary database* yang telah tergantikan oleh *standby database* tidak dapat menjadi *primary database* kembali tanpa melalui proses perputaran *standby database*.
3. Fungsi utama *standby database* adalah sebagai *database* cadangan yang bekerja mereplikasi semua data yang ada

pada *primary database* pada saat *admin* melakukan transaksi data.

VIII. DAFTAR PUSTAKA

- [1] Connolly, T & Beag, C. 2002. Database System: A Practical Approach in Design, Implementation, and Management. Third Edition. Addison Wesley.
- [2] Gregory, P. 2009. CISA Certified Information Systems Auditor All-in-One Exam Guide. ISBN 978-0-07-148755-9. Page 480.
- [3] Hoffer, J.A, Prescott, M.B, Topi, H. 2005. Modern Database Management. Seventh Edition. Pearson Education Inc.
- [4] Noraziah, A. 2012. Replication Techniques in Data Grid Environment. Volume 7197.
- [5] Navathe, S.B & Elmasri, R. 2011. Fundamentals of Database Systems. Sixth Edition. Pearson Education Inc. ISBN 13: 978-0-136-08620-8
- [6] Qian, C, Huang, Y, Zhao, X, Nakagawa, T. 2010. Optimal Backup Policies for a Database System with Periodic Incremental Backup. ISBN 978-952-5726-02-2.
- [7] Radulescu, S.B. 2002. High Availability Solution for a Transactional Database System. Thesis for the Degree of Master of Computer Science at Concordia University. Canada.
- [8] Smith, D.M. 2003. The Cost of Lost Data. Volume 6. Graziadio.
- [9] Singh, S, Rishi, R, Taneja, G, Manocha, A. 2013. Reability and Availability Analysis of Database System with Standby Unit Provide by the System Provider. ISSN 2231-2307.
- [10] Tawar & Wahyuningsih, S. 2011. Perbandingan Metode Back Up Database My SQL antara Replikasi dan MySQLDump. ISSN 2087-8737.
- [11] Wiesmann, M. 2000. Database Replication Techniques: a Three Parameter Classification. Proceedings The 19th IEEE Symposium. 0-7695-0543-0.
- [12] Waliyanto. 2000. Sistem Basis Data Analisis dan Pemodelan Data. Yogyakarta: J&J Learning.