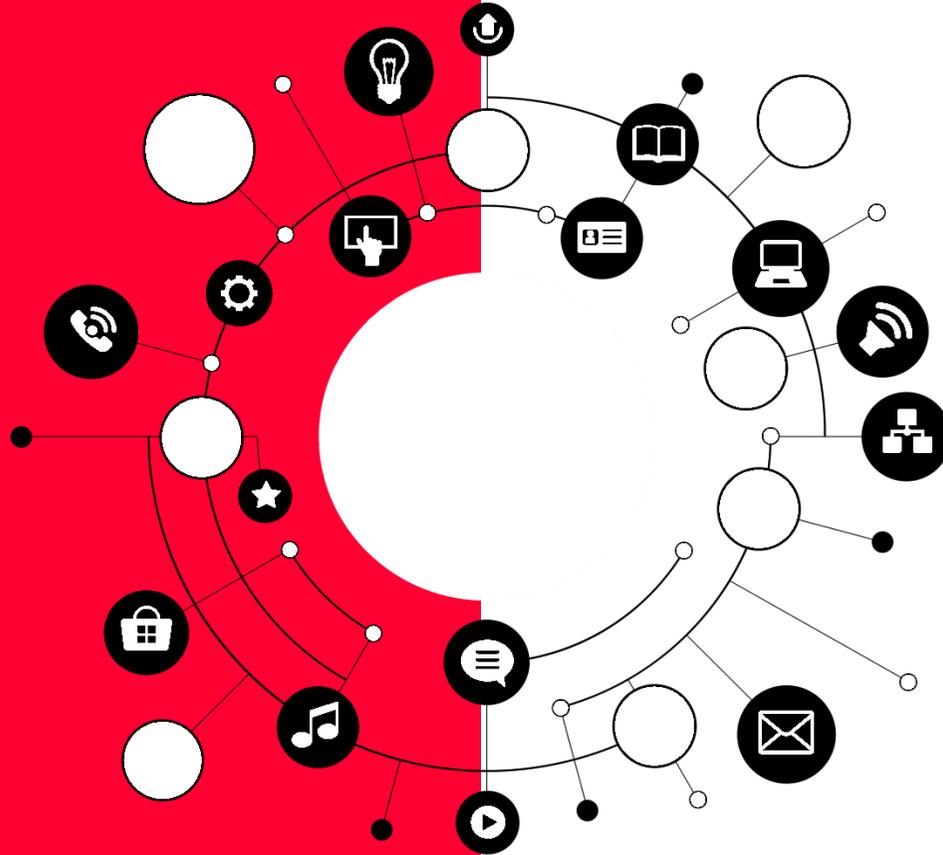


# Membangun Jaringan WARNET di JAYA.net



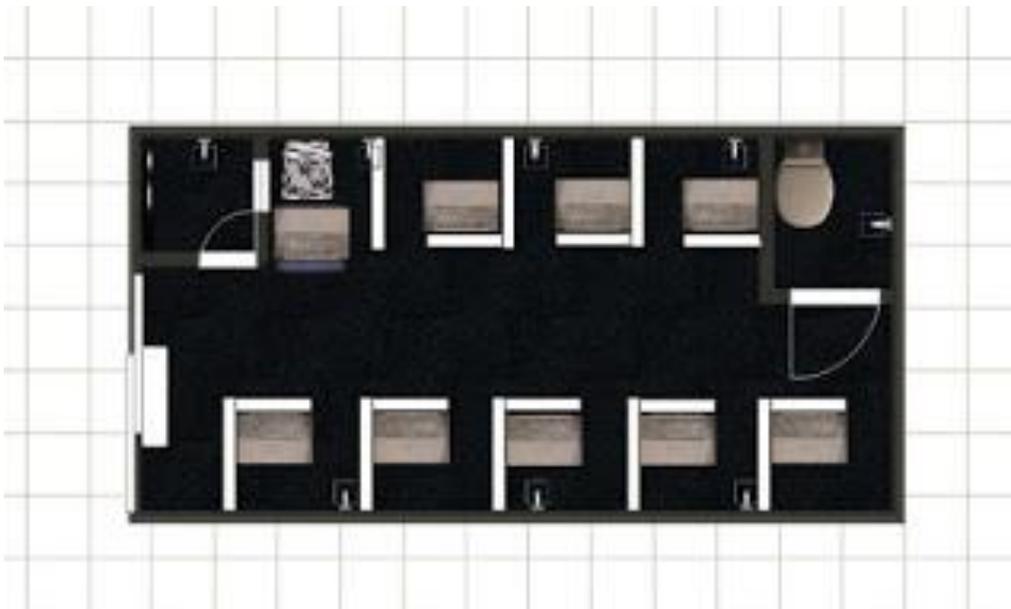
**CV MTI GROUP**

> Novita Anggraini > Rika Seftiana > Sulis

# PENDAHULUAN



# DENAH



## ■ LANTAI 1

■ 8 Komputer Client

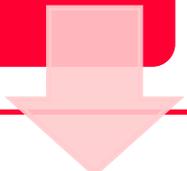
■ 1 Komputer Billing

Jam  
Operasional

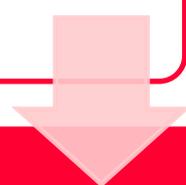
07:00 – 22:00  
WIB

# PROSES BISNIS

Jaringan Efektif

A light red downward-pointing arrow indicating the flow from the first step to the second.

Budget Rp 100,000,000

A light red downward-pointing arrow indicating the flow from the second step to the third.

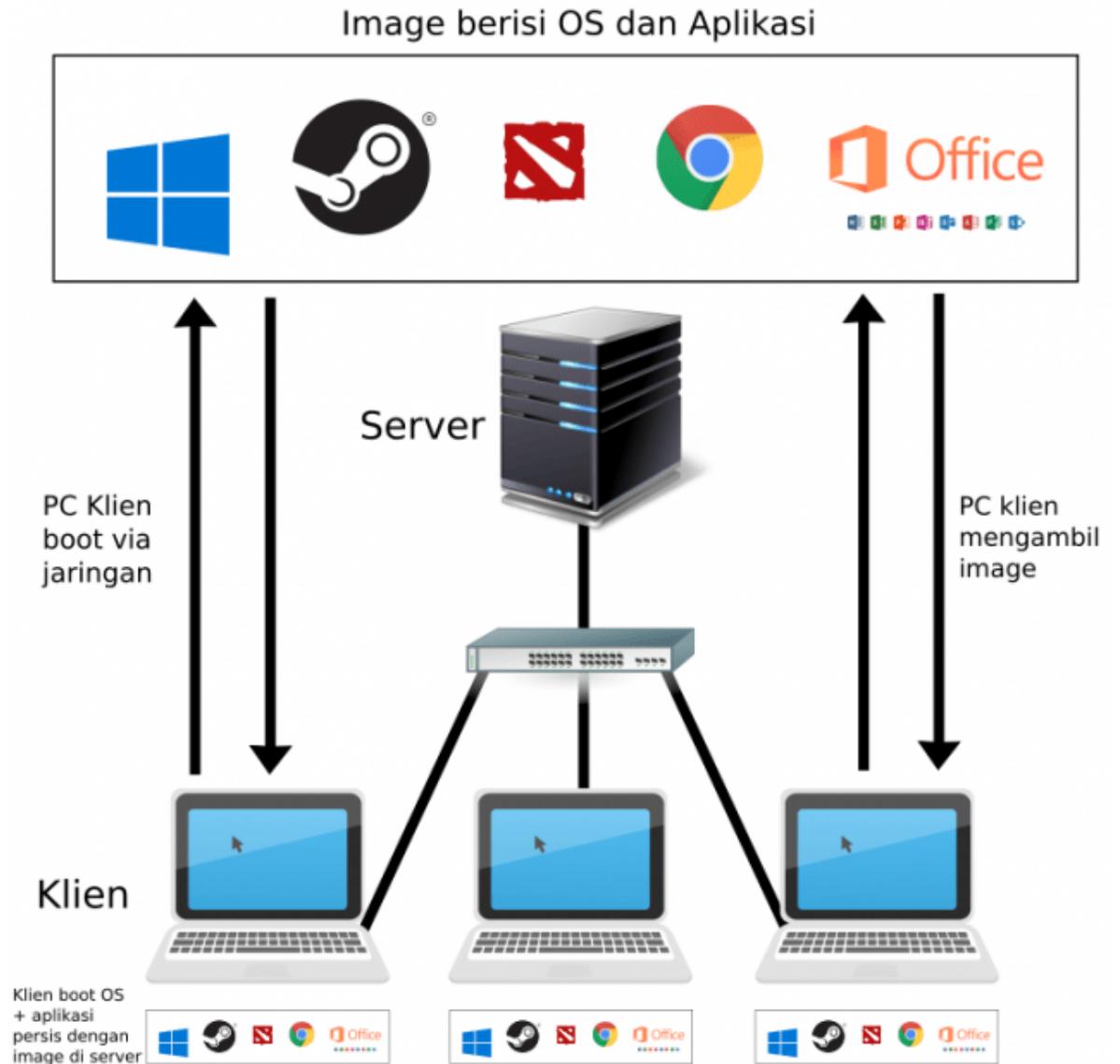
Support untuk Prospek Kedepan

# SOLUSI

**DISKLESS**

**HOW?**

**ILUSTRASI**



# FUNGSIONAL

FIX

Sulitnya untuk menambah aplikasi/game

Sulitnya untuk selalu update aplikasi/game

Sulitnya untuk maintenance

# ALAT DAN BAHAN

- SERVER UNBK DELL T30 Xeon Quad E3-1225v5/16Gb/1TB// 2LAN PORT
- PC client (Komputer Core I3 & LED AOC 16 Full)

- *MIKROTIK RB450G routerboard gigabit ( router indoor)*
- Switch Hub TPLINK 16 Port

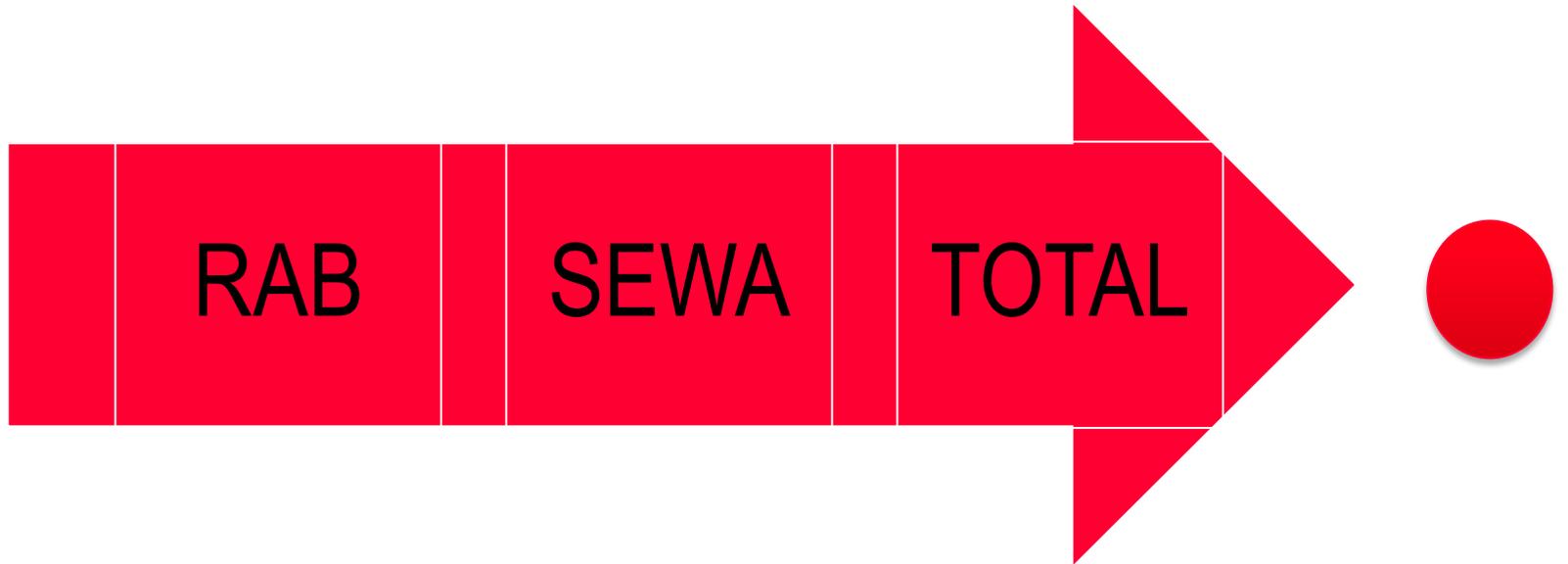
- *Cable UTP Cat 6*
- *Connector RJ45 Cat 6 AMP/COMMSCOPE (100pcs)*

- *Cyberindo Disk*
- UPS 600 VA

- *Camera CCTV SPC 2 Megapixel Tornado series Hybird*
- *Windows 7*

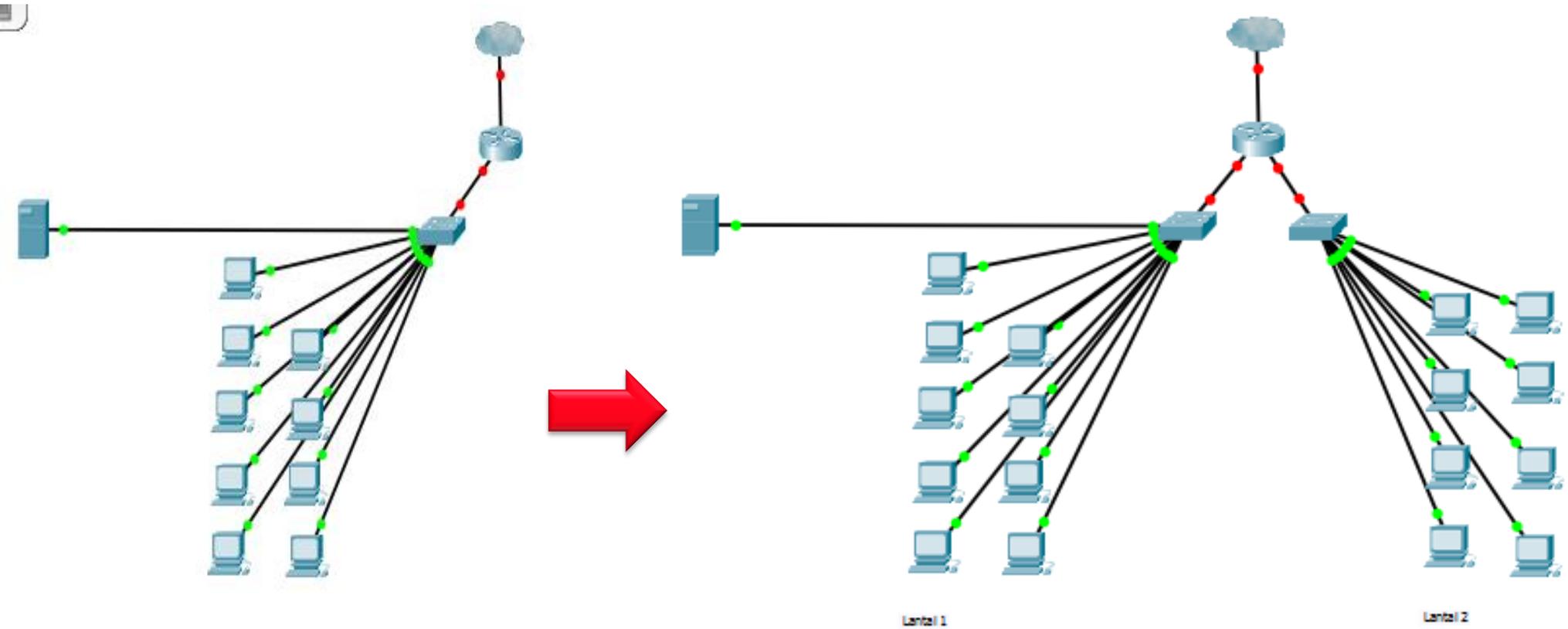
- Stop Kontak
- Kabel Tukuiki

# RAB



RAB dapat dilihat di lampiran, dalam bentuk M.S. Word

# DESIGN



# BENEFIT

Kemudahan Akses data



Biaya lebih murah tanpa hardisk



Serangan Trojan, virus, dan perubahan setting bisa dihindari



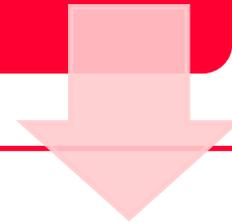
Tidak membayar banyak operator hanya satu saja operator untuk mengelola Komputer Server



Cocok untuk jenis usaha warnet, sekolah, dll

# KRITIK DAN SARAN

Perubahan mengenai penggunaan lebih dari 1 OS (*Operating System*)

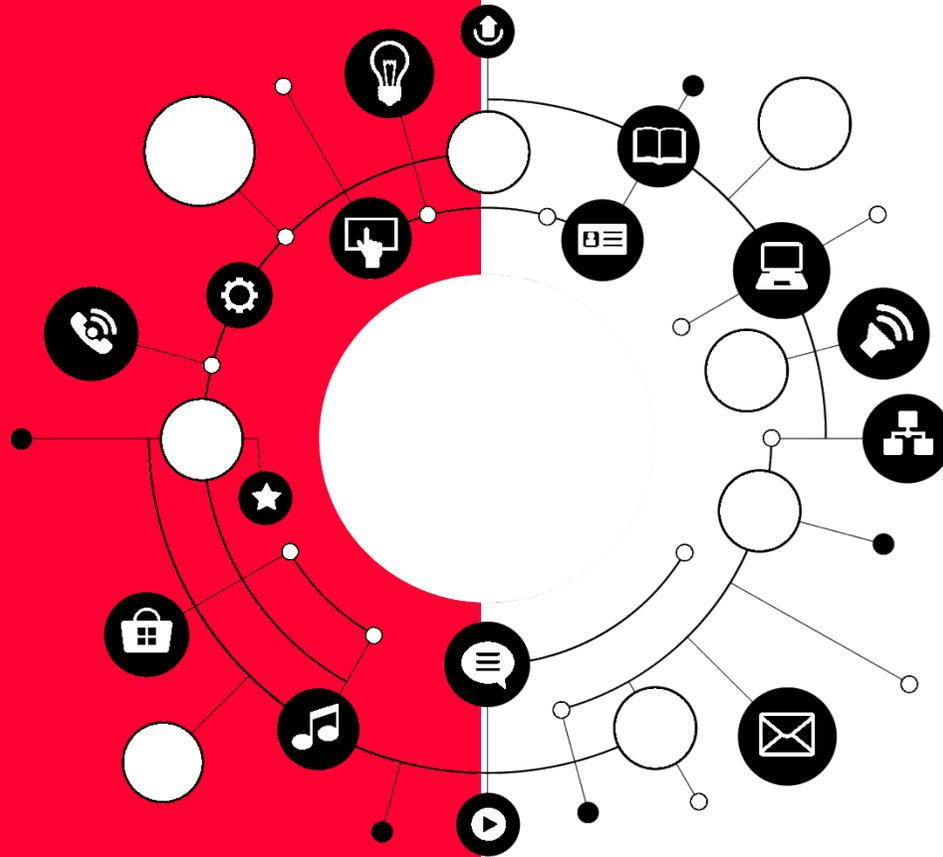


Kebutuhan internet kedepannya disesuaikan oleh kebutuhan user

**SEKIAN**



# Membangun Jaringan WARNET di JAYA.net



**CV MTI GROUP**

> Novita Anggraini > Rika Seftiana > Sulis

# PENDAHULUAN



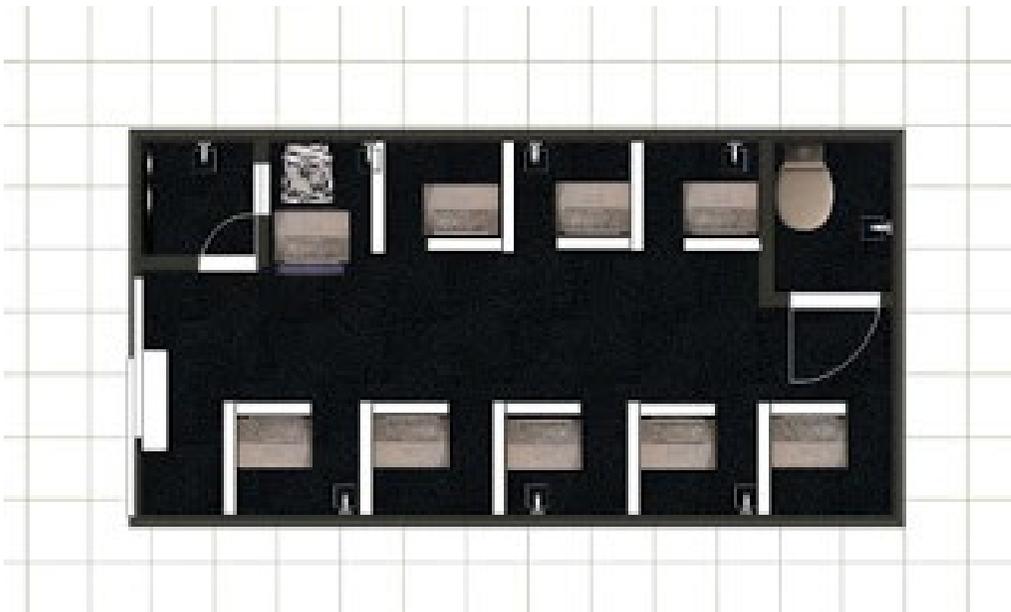
Lokasi

Kebutuhan

Keadaan

Prospek

# DENAH



## ■ LANTAI 1

■ 8 Komputer Client

■ 1 Komputer Billing

Jam  
Operasional

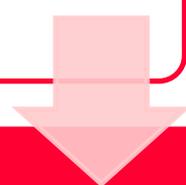
07:00 – 22:00  
WIB

# PROSES BISNIS

Jaringan Efektif

A light red downward-pointing arrow indicating a flow from the first step to the second.

Budget Rp 100,000,000

A light red downward-pointing arrow indicating a flow from the second step to the third.

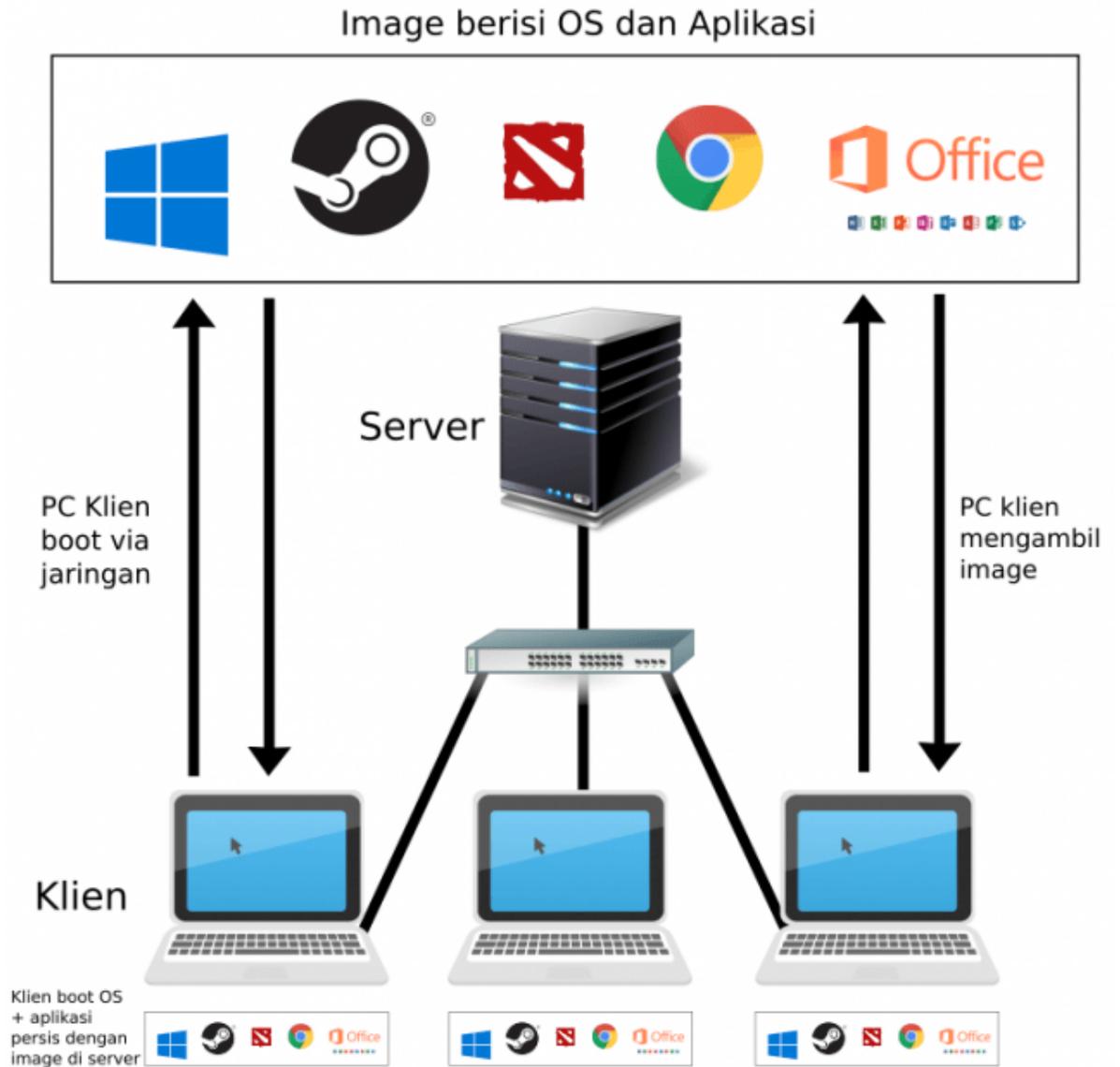
Support untuk Prospek Kedepan

# SOLUSI

DISKLESS

HOW?

ILUSTRASI



# FUNGSIONAL

FIX

Sulitnya untuk menambah aplikasi/game

Sulitnya untuk selalu update aplikasi/game

Sulitnya untuk maintenance

# ALAT DAN BAHAN

- SERVER UNBK DELL T30 Xeon Quad E3-1225v5/16Gb/1TB// 2LAN PORT
- PC client (Komputer Core I3 & LED AOC 16 Full)

- *MIKROTIK RB450G routerboard gigabit ( router indoor)*
- Switch Hub TPLINK 16 Port

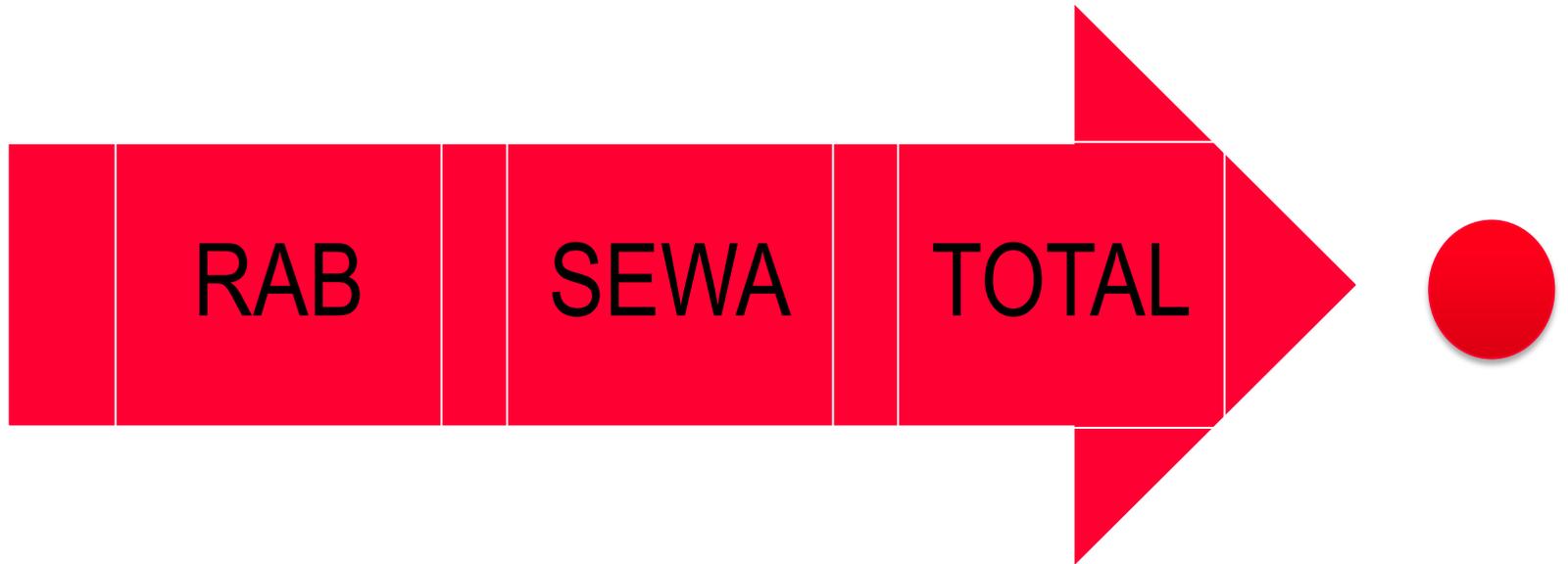
- *Cable UTP Cat 6*
- *Connector RJ45 Cat 6 AMP/COMMSCOPE (100pcs)*

- *Cyberindo Disk*
- UPS 600 VA

- *Camera CCTV SPC 2 Megapixel Tornado series Hybird*
- *Windows 7*

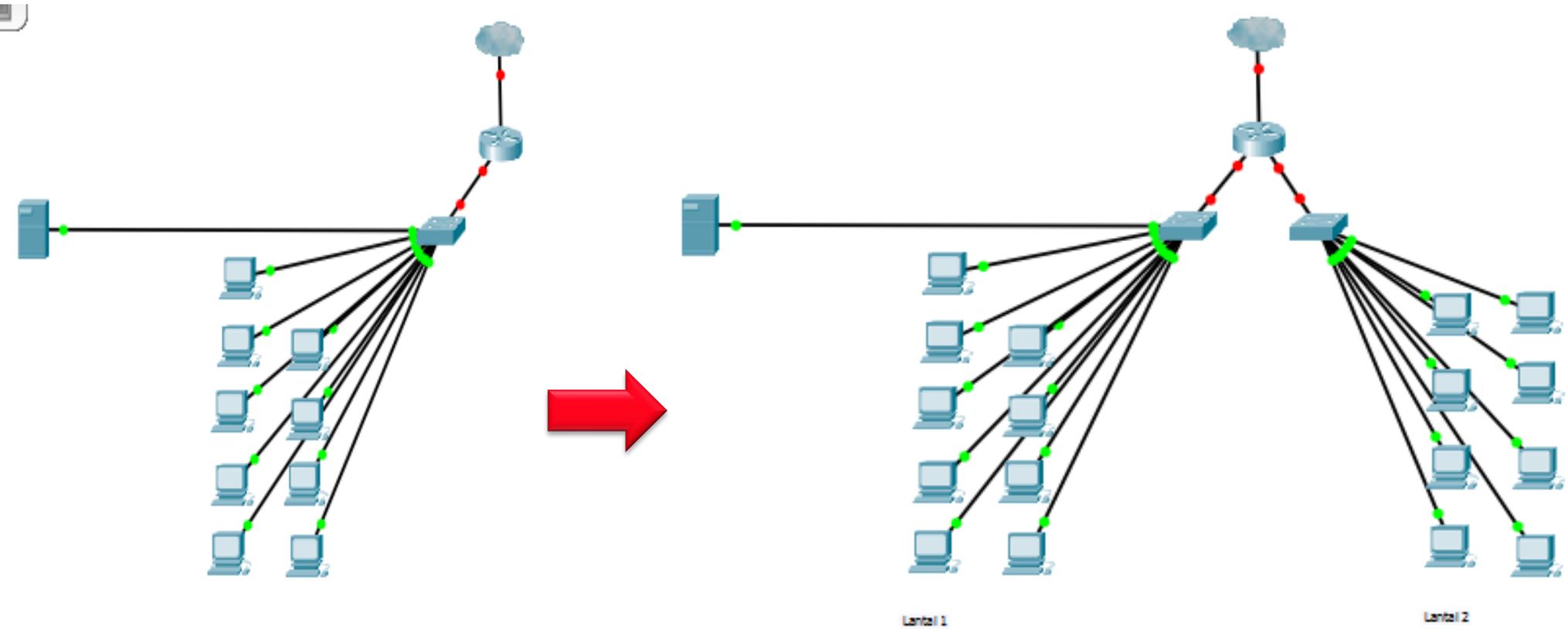
- Stop Kontak
- Kabel Tukuiki

# RAB



RAB dapat dilihat di lampiran, dalam bentuk M.S. Word

# DESIGN



# BENEFIT

Kemudahan Akses data



Biaya lebih murah tanpa hardisk



Serangan Trojan, virus, dan perubahan setting bisa dihindari



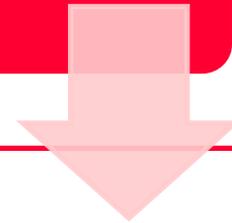
Tidak membayar banyak operator hanya satu saja operator untuk mengelola Komputer Server



Cocok untuk jenis usaha warnet, sekolah, dll

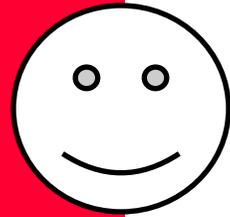
# KRITIK DAN SARAN

Perubahan mengenai penggunaan lebih dari 1 OS (*Operating System*)



Kebutuhan internet kedepannya disesuaikan oleh kebutuhan user

**SEKIAN**



**RANCANGAN ANGGARAN BIAYA 1 LANTAI**

- **Biaya Alat Dan Sarana**

Alat & Sarana	Uraian	Satuan	Biaya
Komputer server <i>Diskless AMD FM 2 + Seri A</i>	Rp 12.150.000	1 buah	Rp 12.150.000
PC client (Komputer Core I3 & LED AOC 16 Full)	Rp 2.175.000	9 buah	Rp 19.575.000
<i>MIKROTIK RB450G routerboard gigabit (router indoor)</i>	Rp Rp1.670.000	1 buah	Rp Rp1.670.000
TPLINK TL-SF1016D SWITCH HUB 16PORT	Rp 209.000	1 buah	Rp 209.000
<i>Cable UTP Cat 6</i>	Rp 1.295.000	1 roll	Rp 1.295.000
<i>Connector RJ45 Cat 6 AMP/COMMSCOPE (100pcs)</i>	Rp 970.000	1 buah	Rp 970.000
<i>Cyberindo Disk</i>	Free	1 buah	Free
UPS 600 VA	Rp 355.000	1 buah	Rp 355.000
Camera CCTV SPC 2 <i>Megapixel Tornado series Hybird</i>	Rp 168.900	1 buah	Rp 168.900
<i>Stop Contact Kuningan 6 lubang</i>	Rp 30.0000	12 buah	Rp 360.000
<i>Kabel Tukuiki</i>	Rp 163.000	1 roll	Rp 163.000
<i>Windows 7</i>	Free	1 buah	Free
<b>TOTAL</b>			<b>Rp 36.915.000</b>

- **Biaya Sewa dan Internet**

Sewa	Satuan	Biaya
ISP Telkom Unlimited	/bulan	Rp 700.000
Perawatan PC, perlengkapan dan biaya cadangan	/bulan	Rp 500.000
Listrik	/bulan	Rp 300.000
<b>Total</b>		<b>Rp 1.800.000</b>

- **Total Pengeluaran**

<b>Modal awal</b>	<b>Rp. 100.000.000</b>
Alat dan Bahan	Rp 36.915.000
Biaya Sewa	Rp 1.800.000
<b>Total</b>	<b>Rp 38.715.000</b>
Sisa	Rp 61.285.000

Biaya sewa untuk 1 lantai adalah Rp 1,440.000/ bulan sehingga estimasi harus lebih besar dari itu. Jika perjamnya adalah sebesar 3000 maka sehari pendapatan harus minimal Rp 48.000/hari x 30 1,440,000 untuk menutupi biaya *maintenance* dan internet setiap bulannya. Dengan rincian 2 pengunjung mengakses 2 jam di 8 pc setiap harinya dikali 30 hari.

Rp 6000 x 8 pc x 30 hari = Rp 1.440.000/bulan

**INVESTASI KEDEPANNYA 2 LANTAI**

● **Biaya Alat Dan Sarana**

Alat & Sarana	Uraian	Satuan	Biaya
PC client (Komputer Core I3 & LED AOC 16 Full)	Rp 2.175.000	8 buah	Rp 17.400.000
TPLINK TL-SF1016D SWITCH HUB 16PORT	Rp 209.000	1 buah	Rp 209.000
Camera CCTV SPC 2 <i>Megapixel Tornado series Hybird</i>	Rp 168.900	1 buah	Rp 168.900
<i>Stop Contact Kuningan</i> 6 lubang	Rp 30.0000	12 buah	Rp 360.000
<b>TOTAL</b>			<b>Rp18.137.900</b>

● **Biaya Sewa dan Internet**

Sewa	Satuan	Biaya
ISP Telkom Unlimited	/bulan	Rp 700.000
Perawatan PC, perlengkapan dan biaya cadangan	/bulan	Rp 800.000
<b>Listrik</b>	/bulan	Rp 700.000
Total		Rp 2.200.000

● **Total Pengeluaran**

<b>Modal awal</b>	<b>Rp. 100.000.000</b>
Alat dan Bahan Lantai 1	Rp 36.915.000
Alat dan Bahan Lantai 2	Rp 18.137.900
Sewa	<b>Rp 2.200.000</b>
Total	Rp 57.252.900
Sisa	Rp 42.747.100

Sehingga untuk bulan pertama pengeluaran adalah sebesar Rp 59.427.900 untuk dua lantai

Biaya sewa untuk 2 lantai adalah Rp 2.200.000/ bulan sehingga estimasi harus lebih besar dari itu. Jika perjamnya adalah sebesar 3000 maka sehari pendapatan harus minimal Rp 96.000/hari x 30 = 2,880,000 untuk menutupi biaya *maintenance* dan internet setiap bulannya. Dengan rincian 2 pengunjung mengakses 2 jam di 16 pc setiap harinya dikali 30 hari.

Rp 6000 x 16 pc x 30 hari = Rp 2.880.000/bulan

Sekian  
**CV MTI Group**

## Tugas

**Nama : Riyan Pratama**

**NIM : 192420041**

**Kelas : MTI 21 AR2**

**Pertanyaan : Mengapa Internet melambat ketika jaringan sedang sibuk dari studi kasus diberikan melalui video yang diunggah oleh Dr Richard Mortier**

A. Latar Belakang :

Menurut jurnal Ilmu Komputer Universitas California :

Dalam kasus tersebut dimana pertumbuhan internet semakin pesat dan banyak transisi perpindahan akses peneliti menganalisis dengan dua metode yaitu single Connection Behavior dan Parallel Connection Behavior

1. Single Connection Behavior :

- a. Loss Recovery : Bagaimanakah Pemulihan dan efektifitas pemulihan data dalam menghindari saat timeout
- b. Receiver Bottleneck : Berapa Seringkah Kinerja TCP menerima halaman dengan waktu yang terbatas
- c. Ack compression : Seberapa sering terjadi ack compression saat kehilangan data

2. Parallel Connection Behavior :

- a. Bagaimana klien melihat jumlah koneksi saat server dibuka
- b. Congestion control: bagaimana reaksi koneksi paralel menerima saat kehilangan data yang disebabkan koneksi timeout
- c. Loss behavior : Bagaimana data yang didistribusikan saat koneksi sedang padat

B. Analisa :

Didalam Penelitian tersebut dijelaskan bagaimana cara menganalisa tentang koneksi TCP saat koneksi sedang sibuk dan seberapa baik pemulihan yang

terjadi. Ada beberapa analisa tentang bagaimana cara memperbaiki data yang diterima saat melakukan koneksi jaringan antara lain :

### 1. Single Connection Behavior

Total connections	16501	
With packet re-ordering	9703	6
With receiver window as	2339	14
Total packets	78216	
During slow-start	66620	85
# of slow-start pkts lost	3545	5
During congestion	11595	15
# of congestion	8218	7
Total retransmissions	8571	
Fast retransmissions	3753	44
Slow-start retransmissions	5981	7
Coarse timeouts	4220	49
Avoidable with SACKs	1871	4
Avoidable with enhanced	1042	25

Pada tabel diatas saat melakukan single connection dimana ukuran data yang panggil lebih besar daripada data yang diterima

### 2. Parallel Connection Behavior

Analisa Koneksi yang terbuang saat melakukan koneksi secara bersamaan

### 3. Key Results of Trace Analysis

Dari penelitian yang dikembangkan dimana koneksi tunggal lebih baik dalam pemulihan pengiriman data ketimbang dengan koneksi yang bersamaan

## C. Pembahasan

Ada beberapa metode yang dapat dilakukan antara lain :

### 1. Application-level Multiplexing

Solusi tingkat aplikasi menghindari penggunaan beberapa koneksi TCP paralel, dan masalah yang dihasilkan, dengan multiplexing beberapa

aliran data ke koneksi TCP tunggal. Karena TCP hanya menyediakan abstraksi byte-stream tunggal

Tapi ada beberapa kekurangan antara lain :

- a. Mereka membutuhkan tambahan dan diperlukan server dan klien yang baik.
- b. Mereka tidak mengizinkan multiplexing transfer yang dilakukan oleh lebih dari satu aplikasi.
- c. Multiplexing melalui koneksi TCP tunggal memperkenalkan kopling yang tidak diinginkan antara transfer data yang secara logis independen

## **2. TCP-INT Implementation**

Menggambarkan penerapan kontrol kemacetan terintegrasi dan skema pemulihan kehilangan yang hanya mengubah TCP pada pengirim. Untuk setiap host yang terkait dengan satu mesin, tumpukan TCP / IP membuat struktur (Gambar 9) untuk menyimpan informasi tentang komunikasi apa pun. Struktur baru ini memungkinkan kontrol kemacetan bersama yang diinginkan dan pemulihan kerugian dengan menyediakan satu titik koordinasi untuk semua koneksi ke host tertentu. Struktur host berisi variabel-variabel standar TCP yang terkait dengan pemeliharaan jendela kongesti TCP (cwnd, ssthresh dan count). Struktur ini juga memperkenalkan beberapa variabel baru untuk membantu dalam kontrol kemacetan (ownd, decr\_ts) dan variabel lain untuk mendukung pemulihan kerugian terintegrasi (pkts []). Di subbagian berikut, kami menjelaskan bagaimana berbagai rutinitas TCP menggunakan dan memperbarui informasi baru ini. Rutin pengiriman data baru: Ketika suatu koneksi ingin mengirim suatu paket, ia memeriksa untuk melihat apakah jumlah byte yang sudah ada dalam "pipa" dari pengirim (ownd) lebih besar daripada ukuran "pipa" yang diinginkan (cwnd). Jika tidak, koneksi akan menyiapkan paket yang akan dikirim dengan menambahkan entri pada ekor daftar paket yang beredar. Entri ini berisi ukuran nomor urut dan stempel waktu dari paket yang dikirimkan. Ketika paket dikirim, koneksi menambah kepemilikan

dengan ukuran paket. Kami menggunakan penjadwalan round-robin di seluruh koneksi meskipun itu bukan persyaratan penting. Rutin `recv ack` baru: Ketika `ack` baru tiba, pengirim meningkatkan variabel `cwnd` yang sesuai. Juga pada saat kedatangan `ACK` baru, pengirim menghapus paket dari daftar `pkts []` yang telah mencapai

### **3. Integrated Congestion Control/Loss Recovery**

Motivasi untuk kontrol kemacetan terintegrasi dan pemulihan kerugian adalah untuk memungkinkan aplikasi menggunakan koneksi TCP terpisah untuk setiap transfer (seperti yang mereka lakukan hari ini), tetapi untuk menghindari masalah yang disebutkan dalam Bagian 3.2 dengan membuat modifikasi yang sesuai untuk tumpukan jaringan. Kami membagi fungsionalitas TCP ke dalam dua kategori: yang berkaitan dengan abstraksi TCP byte-stream yang andal, dan yang berkaitan dengan kontrol kemacetan dan pemulihan kerugian berbasis data. Yang terakhir ini dilakukan secara terintegrasi di set koneksi paralel. Kami menyebutnya versi modifikasi TCP `TCP-Int`. Dengan membuka  $n$  koneksi TCP terpisah untuk transfer, aplikasi memiliki  $n$ -stream stream yang andal dan independen untuk digunakan. Kontrol aliran untuk setiap koneksi terjadi secara independen dari yang lain, sehingga pengiriman data ke aplikasi penerima juga terjadi secara independen untuk setiap koneksi. Pada saat yang sama, kontrol kemacetan terintegrasi di seluruh koneksi TCP. Ada satu jendela kemacetan untuk set koneksi TCP antara klien dan server yang menentukan jumlah total data luar biasa yang dimiliki oleh set koneksi dalam jaringan. Ketika terjadi kerugian pada salah satu koneksi,

### **4. Simulation Results: One Client Host Case**

Cara ini menjelaskan hasil dari simulasi `ns` yang dirancang untuk menguji kontrol kemacetan dengan pengenal integer dan pemulihan kehilangan pada koneksi TCP simultan. Tes pertama menggunakan topologi pada Gambar 7. Ukuran buffer router diatur ke 3 paket. Ini cukup kecil untuk memaksa transfer untuk memiliki jendela kemacetan kecil dan sering mengalami kerugian. Sekali lagi, topologi dan parameter dipilih untuk menciptakan kembali situasi yang sering terjadi di jejak kami, dan

tidak meniru jaringan yang sebenarnya. Dalam tes ini, node pengirim melakukan 4 transfer TCP ke penerima. Transfer mulai dari 0, 2, 4 dan 6 detik dan semua berakhir pada 10 detik. Pilihan aktual dari nilai 0, 2, 4, dan 6 tidak penting, hanya saja waktu mulai setiap koneksi terhuyung-huyung dalam waktu. Gambar 10 menunjukkan plot urutan untuk pengujian menggunakan pengirim berbasis SACK. Ini menunjukkan bahwa biasanya hanya satu koneksi berperforma memuaskan pada satu waktu. Misalnya, pada waktu 2 detik, koneksi yang memulai mengalami beberapa kerugian awal dan dipaksa untuk memulihkannya melalui timeout kasar. Faktanya, koneksi ini tidak mengirim sejumlah besar data hingga 4 detik kemudian (pada waktu 6 detik). Selama periode 10 detik, koneksi dimulai pada waktu 2 detik dan waktu 6 detik. Akun untuk fraksi sangat kecil (<10%) dari total byte yang ditransfer. Ketidakadilan dan kinerja yang tidak dapat diprediksi (karena timeout kasar) tidak diinginkan dari sudut pandang aplikasi karena koneksi yang membawa data penting dapat diperlambat sementara yang lain yang membawa data yang kurang penting lebih baik.

##### **5. Simulation Results: Multiple Client Hosts Case**

Simulasi ini menggunakan topologi jaringan dimana saat waktu mulai koneksi di waktu 0, transfer TCP tunggal dimulai setengah dari bandwidth dapat diterima kembali. berfungsi untuk mengimplementasikan fungsi yang lebih kompleks dalam limit bandwidth, dimana penggunaan packet mark nya memiliki fungsi yang lebih baik. Digunakan untuk membatasi satu arah koneksi saja baik itu download maupun upload. Secara umum Queue Tree ini tidak terlihat berbeda dari Simple Queue. Perbedaan yang bisa kita lihat langsung yaitu hanya dari sisi cara pakai atau penggunaannya saja. Dimana Queue Simple secara khusus memang dirancang untuk kemudahan konfigurasi sementara Queue Tree dirancang untuk melaksanakan tugas antrian yang lebih kompleks dan butuh pemahaman yang baik tentang aliran trafik



# Ade Saputra

Nim : 192420027

Kelas : MTI

## Jaringan PAKET CV CEMERLANG

Sistem Layanan Jaringan Profesional

*Copyright © 2019 CV. CV CEMERLANG . All right reserved*

# Rencana Desain Warnet





# Agenda Style

01

## Profil

Menawarkan hal-hal yang kami layani dalam bidang IT

02

## Permasalahan Pada Layanan IT

Penjelasan mengenai masalah yang terjadi saat ini

03

## Sistem IT CV Cemerlang

Memberikan gambaran umum tentang cara pemasangan jaringan computer dalam Usaha Warnet

04

## Fitur IT CV CEMERLANG

Mengenalkan paket-paket instalasi jaringan Usaha Warnet yang kami sediakan

# Profil

Profil Perusahaan



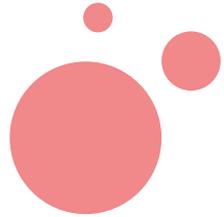


## VISI

“Untuk menjadi perusahaan layanan teknologi yang terbaik, memberikan nilai berkelanjutan kepada para pemangku kepentingan Proses Bisnis

## MISI

- “Membangun hubungan mitra terpercaya melalui penyediaan solusi terbaik”
- “Menjalankan filosofi perusahaan kami dengan standar kualitas tinggi Sumber Daya Manusia, untuk memastikan pertumbuhan yang berkelanjutan dari perusahaan kami”



# CV CEMERLANG

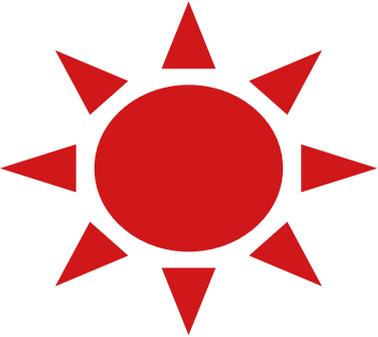
Sistem Layanan Jaringan Nasional

## A. Latar Belakang

Dengan semakin meningkatnya kebutuhan masyarakat akan informasi dan komunikasi, menuntut akan adanya kemudahan dalam mengaksesnya oleh karena itu kami berinisiatif untuk membangun dan mengembangkan bisnis internet. Bisnis ini selalu dibutuhkan karena informasi yang dijualnya.

Sebagai sarana komunikasi, serta sebagai media hiburan. Apalagi dengan kehadiran jejaring social yang saat ini sangat di gandrungi masyarakat, kenyataan ini tentu akan meningkatkan para pengguna internet .

# Tujuan Penawaran Bisnis Warnet



**Service Informasi**  
Kebutuhan User dalam mendapatkan Informasi Yang Cepat



**Pendidikan**  
Kebutuhan sumber Ilmu ter baru



**Peluang Usaha**  
Sangat Profit dan Di Butuh kan



**Ekonomi**  
Mendapatkan profit Cukup Besar



**Service Yang Komplitt**  
Fasilatas Cepat dan Nyaman



**Kepatuhan**  
Memblok situs situs yang berbahaya

### **Total Seluruh Biaya Modal Yang Digunakan**

Server	: Rp. 5.224.000,-
Client	: Rp. 17.472.000,-
Jaringan	: Rp. 2.155.000,-
Lainnya	: Rp. 1.615.000,-
<b>Total</b>	<b>: Rp. 27.876.000,-</b>



## Alat dan Bahan

Motherboard Ecs H61H2-MV HDMI + Gigabit Lan intel socket (1155)

Processor Intel Core i3-3220 Processor (3M Cache, 3.30 GHz) (1155)

Ram Venomrx 8Gb Pc 1333 Venomrx DDR3

HDD Sata Hitachi Sata III 1Tb 7200 Rpm 3.5Inch Int 3,5 I

HDD SSD Intel 335 Series 80 Gb SSD

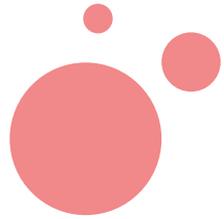
Power Supply Dazumba DZ 500W Box

Casing Mid Tower Dazumba D-Vito 683

Speaker Simbadda 5100n Aktif

UPS Prolink Pro 700v 650Va, True AVR, Wide Input Voltag

Total Copyright © 2019 CV. CEMERLANG. All right reserved



# CV CEMERLANG

Sistem Layanan Jaringan Nasional

## BIAYA ALAT & SARANA UNTUK CLIENT (Rakitan) 7 Unit Komputer

Alat dan Bahan	Harga
Motherboard Gigabyte GA-G41M-Combo (775)	Rp. 4.424.000,-
Processor Intel E5700 Dual Core (Tray + Fan) (3.0Ghz, C2 Mb, Fsb) (775)	Rp. 805.000,-
Ram Vgen 2Gb Pc 12800/1600 Vgen DDR3	Rp. 1.309.000,-
VGA Pixelview GT 440 2Gb 128bit DDR3 Nvidia PCI Exp.	Rp. 3.724.000,-
Power Supply Dazumba DZ 380W OEM	Rp. 910.000,-
Casing Mini Atx Dazumba DE-210	Rp. 1.155.000,-
Advance 15.6 Inch LM 1670 Lumine LED+Speaker	Rp. 3.850.000,-
Keyboard Logitech Classic Keyboard PS/2 K100 Keyboard	Rp. 581.000,-
Mouse Logitech B100 Optical USB Black Mouse	Rp. 343.000,-
Mouse Pad Oem Mouse Pad Standard/Bantal Mouse Pad	Rp. 56.000,-
Headset Komic Hed103	Rp. 315.000,-
Total	Rp. 17.472.000,-



**CV CEMERLANG**

## BIAYA ALAT & SARANA UNTUK JARINGAN WARNET

<b>Alat dan Bahan</b>	<b>Harga</b>
NIC Gigabit (Lan Card) Untuk PC client dan server (11 Unit)	Rp. 1.320.000,-
Router MikroTik Seri RB951-2n	Rp. 530.000,-
TP-Link TL-SG1016D : 16 Port Gigabyte	Rp. 870.000,-
Kabel UTP STURDY (100m)	Rp. 110.000,-
RG-45 (100 Unit)	Rp. 45.000,-
Total	Rp. 2.875.000,-



## BIAYA ALAT & SARANA LAINNYA (KEPERLUAN WARNET)



<b>Alat dan Bahan</b>	<b>Harga</b>
Meja Bersekat (7 unit)	Rp. 700.000,-
Meja Operator	Rp. 225.000,-
Karpet Biru Standard Ukuran 2x3m (2 Unit)	Rp. 100.000,-
Kipas angin GMC Wall Fan 509 (3 Unit)	Rp. 360.000,-
Kenmaster Stop Kontak F1-44 Switch (4 lubang) (6 Unit)	Rp. 180.000,-
Crimping Tools	Rp. 50.000,-
<b>Total</b>	<b>Rp. 1.615.000,-</b>

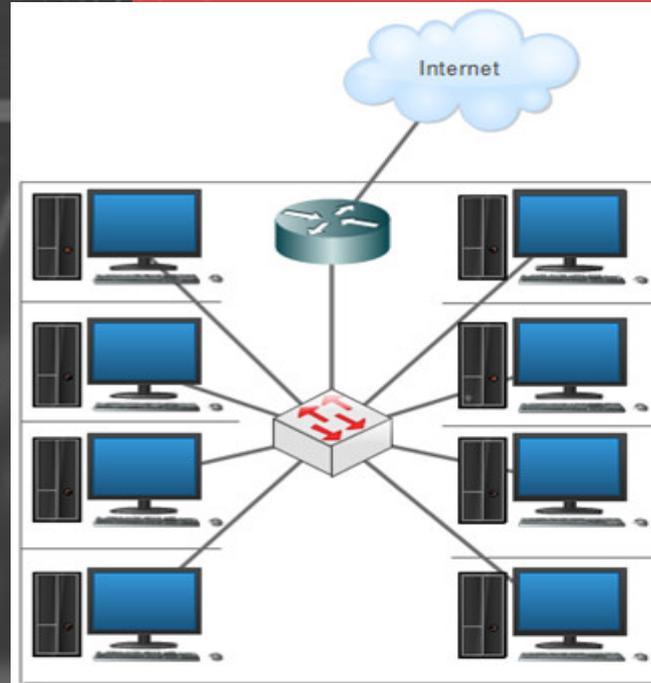
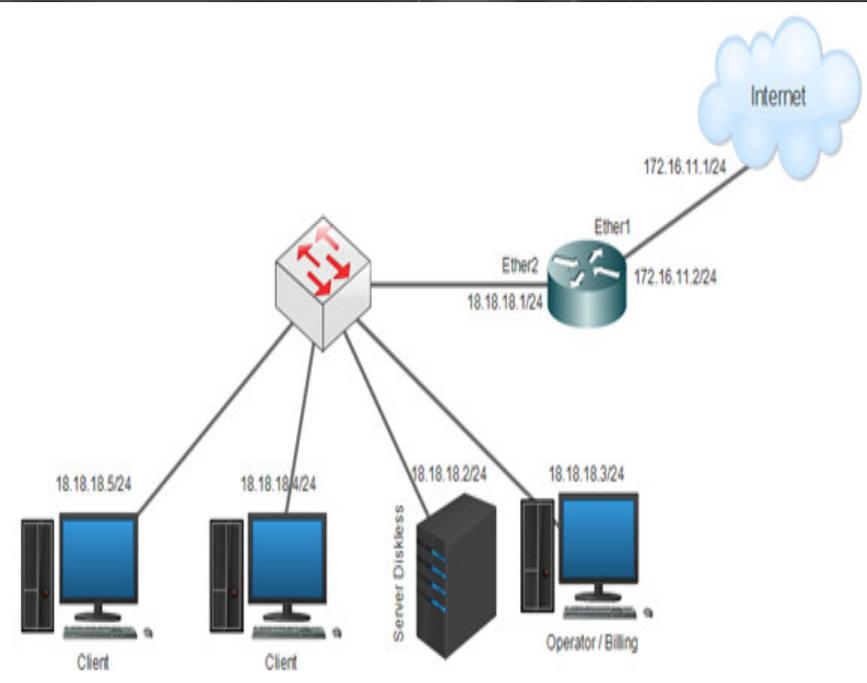
**CV CEMERLANG**



# Desaing Jaringan Yang Di Tawarkan

Konfigurasi IP Warnet – Server, Operator dan Client

Topologi Warnet

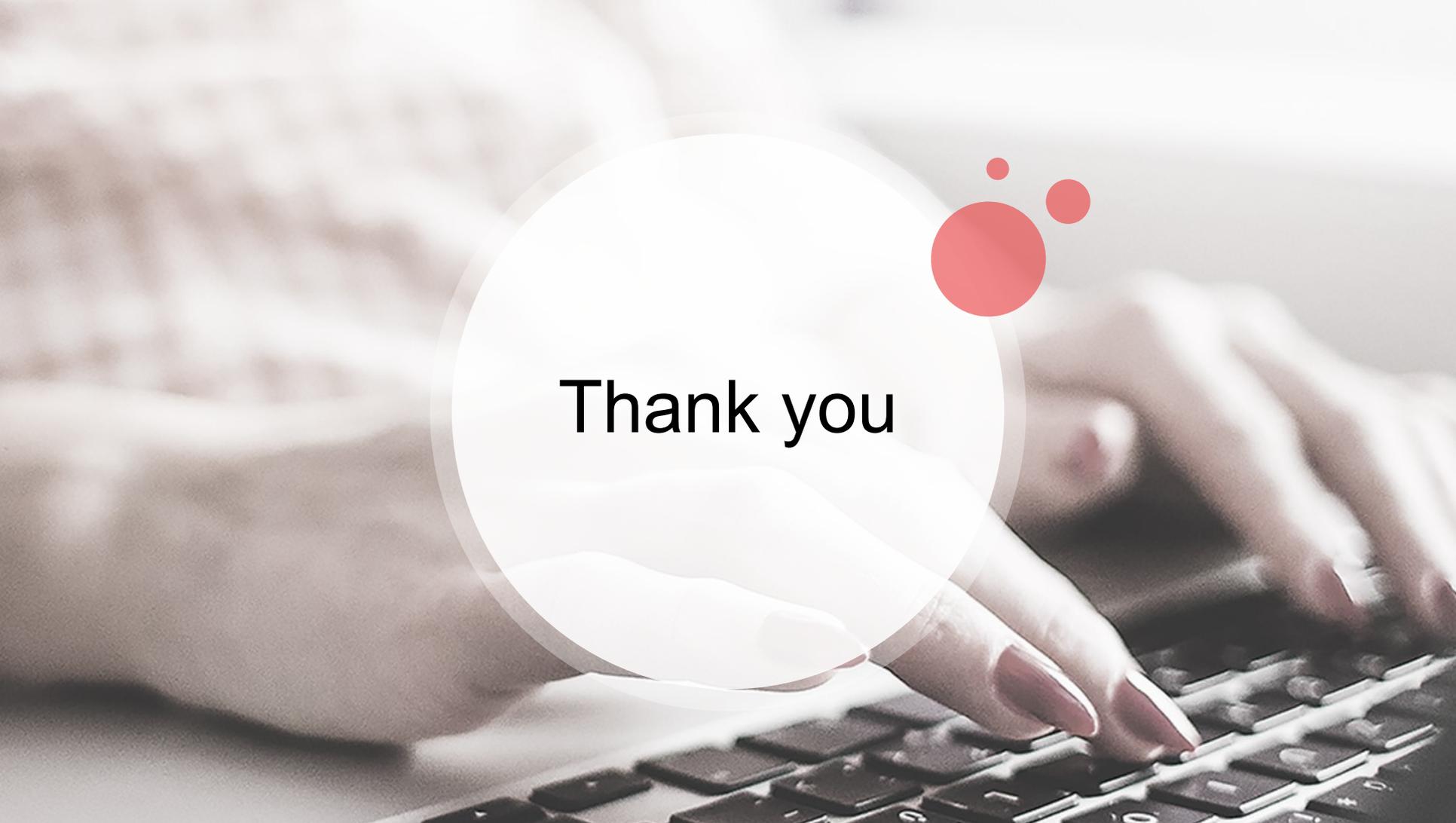


IP Publik : 172.16.11.0/24  
IP Lokal : 18.18.18.0/24

**HARGA RENTAL WARNET**

1	Jam	=	Rp.	3.000,-
2	Jam	=	Rp.	5.000,-
3	Jam	=	Rp.	7.000,-
4	Jam	=	Rp.	10.000,-
5	Jam	=	Rp.	13.000,-
7	Jam	=	Rp.	15.000,-

Keuntungan omset: misal 1 pc 3000 x 14= 42000

A close-up, shallow depth-of-field photograph of a person's hands typing on a laptop keyboard. The hands are positioned in the lower half of the frame, with the fingers resting on the keys. The background is a blurred, warm-toned surface, possibly a desk or another part of the laptop. Overlaid on the center of the image is a large, semi-transparent white circle containing the text 'Thank you'. To the right of this circle, there are three red circles of varying sizes, arranged in a cluster. The overall aesthetic is clean, modern, and professional.

Thank you

Nama : Hasirul Qodar dan Ryan Andrian

Kelas : MTI 21 A R1

UTS : Computer Network and Communications

Pertumbuhan aktifitas di internet yang cepat mempengaruhi lalu lintas internet sehingga menyebabkan terjadinya kemacetan dan pengiriman data yang tidak lengkap. Dari permasalahan tersebut dapat diantisipasi dengan beberapa alternatif solusi diantaranya, yaitu sebagai berikut :

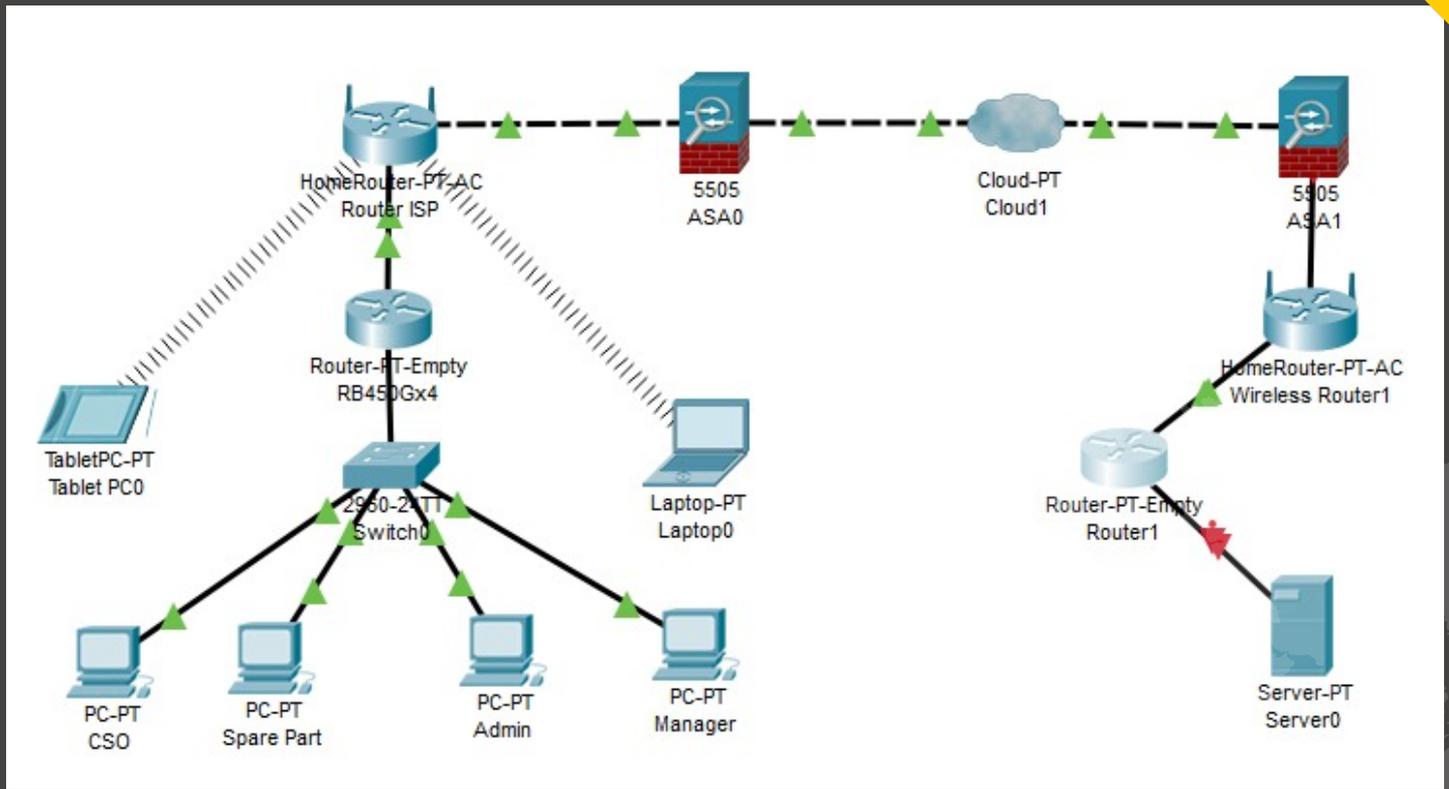
1. Modifikasi protokol tcp dalam pengiriman data yang sebelumnya menggunakan koneksi single di rubah menjadi koneksi paralel
2. Transfer terpisah berdasarkan prioritas jenis data misalnya voice, financial transaction dan gambar
3. Management dalam merespon request dari client di server secara bertahap untuk mengurangi beban memory di server dan merperkecil persentasi data loss



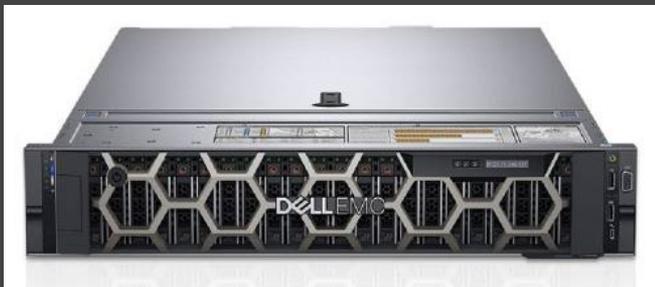
# PT. 2MB INDONESIA



Jl. Ahmad Yani No.500 Palembang  
Sumatera Selatan  
INDONESIA



# Alat dan Bahan



Server

DELL PowerEdge R740

DELL PowerEdge R740

(Dual Xeon Silver 4110, 32GB, 2x 1.92TB SSD)

2 x Xeon Silver 4110

2 x 16GB 2666MT/s Dual Ranked x8 Data Width RDIMMs

2 x 1.92TB SAS 2.5 Inch SSD

1 x Broadcom 5720 QP 1Gb Network Daughter Card

2 x 750W

Rackmount (2U)

# Alat dan Bahan

Router

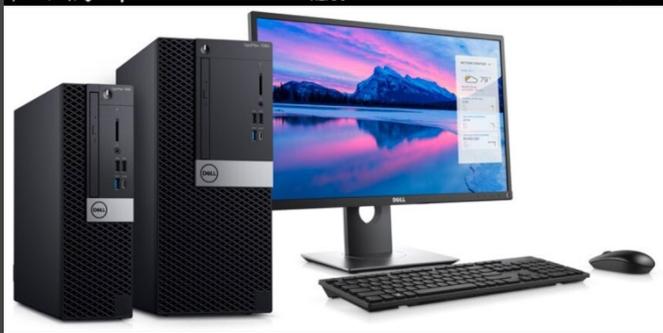
Routerboard RB450Gx4



Routerboard RB450Gx4

(716MHz Quad Core CPU, 1 GB DDR RAM, 512MB NAND Storage)

# Alat dan Bahan



Personal Komputer (PC)

DELL OptiPlex 7060

Intel Core™ i5-8500 (6 Cores/9MB/6T/up to 4.1GHz/65W); supports Windows 10/Linux

Windows 10 Pro 64bit English, French, Spanish

Intel® Integrated Graphics

8GB 2X4GB DDR4 2666MHz UDIMM Non-ECC

3.5" 500GB 7200rpm SATA Hard Disk Drive

## Rencana Anggaran Biaya

No	Komponen	Merk	Quantity	Harga Satuan	Jumlah
1	Server	DELL PowerEdge R740 (Dual Xeon Silver 4110, 32GB, 2x 1.92TB SSD)	1 pc	226.000.000	Rp 226.000.000
2	Router	Routerboard RB450Gx4 (716MHz Quad Core CPU, 1 GB DDR RAM, 512MB NAND Storage)	1 pc	1.617.000	Rp 1.617.000
3	Personal Komputer (PC)	DELL OptiPlex 7060	4 pcs	10.486.000	Rp 41.944.000
4	Kabel LAN UTP	CAT 5e	1 roll	1.325.000	Rp 1.325.000
<b>Total Harga</b>					<b>Rp 270.886.000</b>





Thank you

Feel Our Services

Nama : 1. Sela Taramita (192420038)  
2. Yuni Astuti (192420004)

Mata Kuliah : Computer Network and Business Information

Kelas : AR 1

## “ Why Internet Slows Down When Its Busy “

Permasalahan :

Permasalahan yang terdapat pada video yang telah kami tonton yaitu “*Why internet slows down when it's busy – Computerphile*” terdapat penjelasan yang menyebutkan bahwa penyedia jasa jaringan internet selalu menjanjikan kapasitas besar dan cepat dalam jaringan akan tetapi yang sebenarnya terjadi ketika kita terkoneksi dengan internet system jaringan tersebut mengalami queue (antrian) dimana semakin banyak pengguna semakin sibuk sistem antrian itu mengatur traffic input output datanya. Misalnya yang terdapat pada video, penyedia jasa jaringan internet memiliki total kapasitas jaringan server sebesar 1 (satu) Gb/sec dan mereka memberikan kecepatan dan kapasitas jaringan internet kepada pengguna (konsumen) 100 Mb/sec.

Solusi :

1. Harus meningkatkan standar ukuran *socket buffer* untuk menghindari hambatan dari penerima yang lain
2. Menggunakan 2 metode yaitu :
  - a. Ack bandwidth ratio  
Pada metode ini, mengkalkulasikan ack bandwidth pada jendela TCP untuk semua jejak koneksi
  - b. Perbandingan data dan Ack Bandwidth  
Pada metode kedua ini, mempertimbangkan aliran dari data dan ack bandwidth dimana mempertimbangkan data yang di transmisikan oleh pengirim
3. Meningkatkan kinerja beberapa koneksi TCP  
Untuk dapat meningkatkan kinerja koneksi maka koneksi tersebut dibagi dua dimana kinerja koneksi baru tidak harus menjalani *slow-start* untuk memperkirakan *traffic* yang

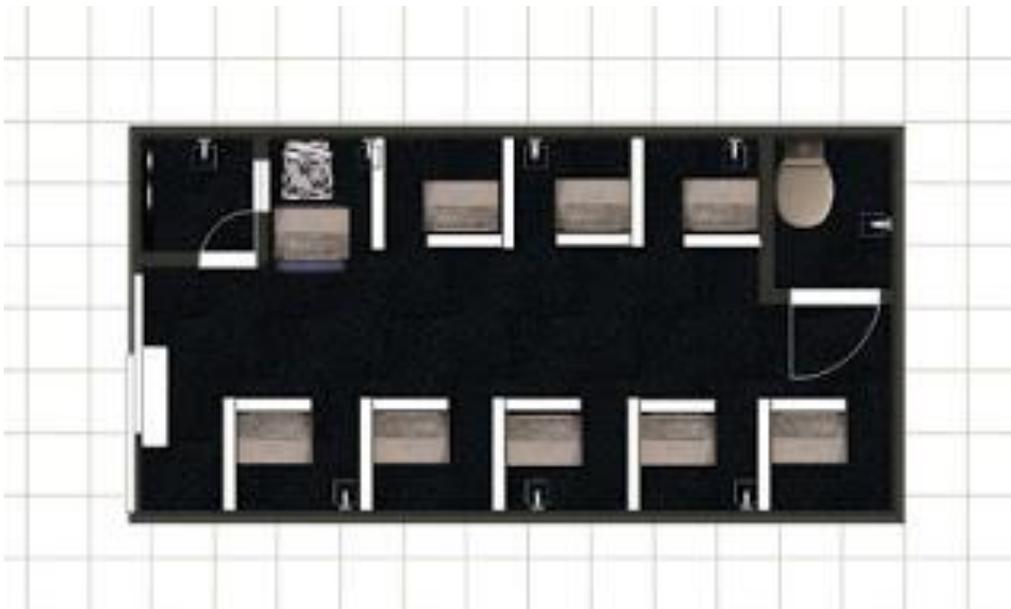
benar, ini mengarah untuk peningkatan kinerja koneksi tambahan. Data-driven recovery terintegrasi di set koneksi TCP. Ketika sebuah paket yang hilang pada satu koneksi, ada keberhasilan pengiriman paket di koneksi yang lain yang memungkinkan pengiriman dapat mendeteksi paket yang loss tanpa menggunakan timeout untuk meningkatkan kinerja tambahan.



# PENDAHULUAN



# DENAH



## ■ LANTAI 1

■ 8 Komputer Client

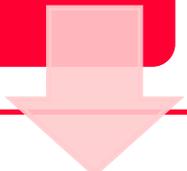
■ 1 Komputer Billing

Jam  
Operasional

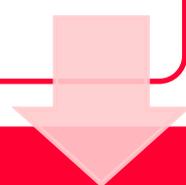
07:00 – 22:00  
WIB

# PROSES BISNIS

Jaringan Efektif

A light red downward-pointing arrow indicating a flow from the first step to the second.

Budget Rp 100,000,000

A light red downward-pointing arrow indicating a flow from the second step to the third.

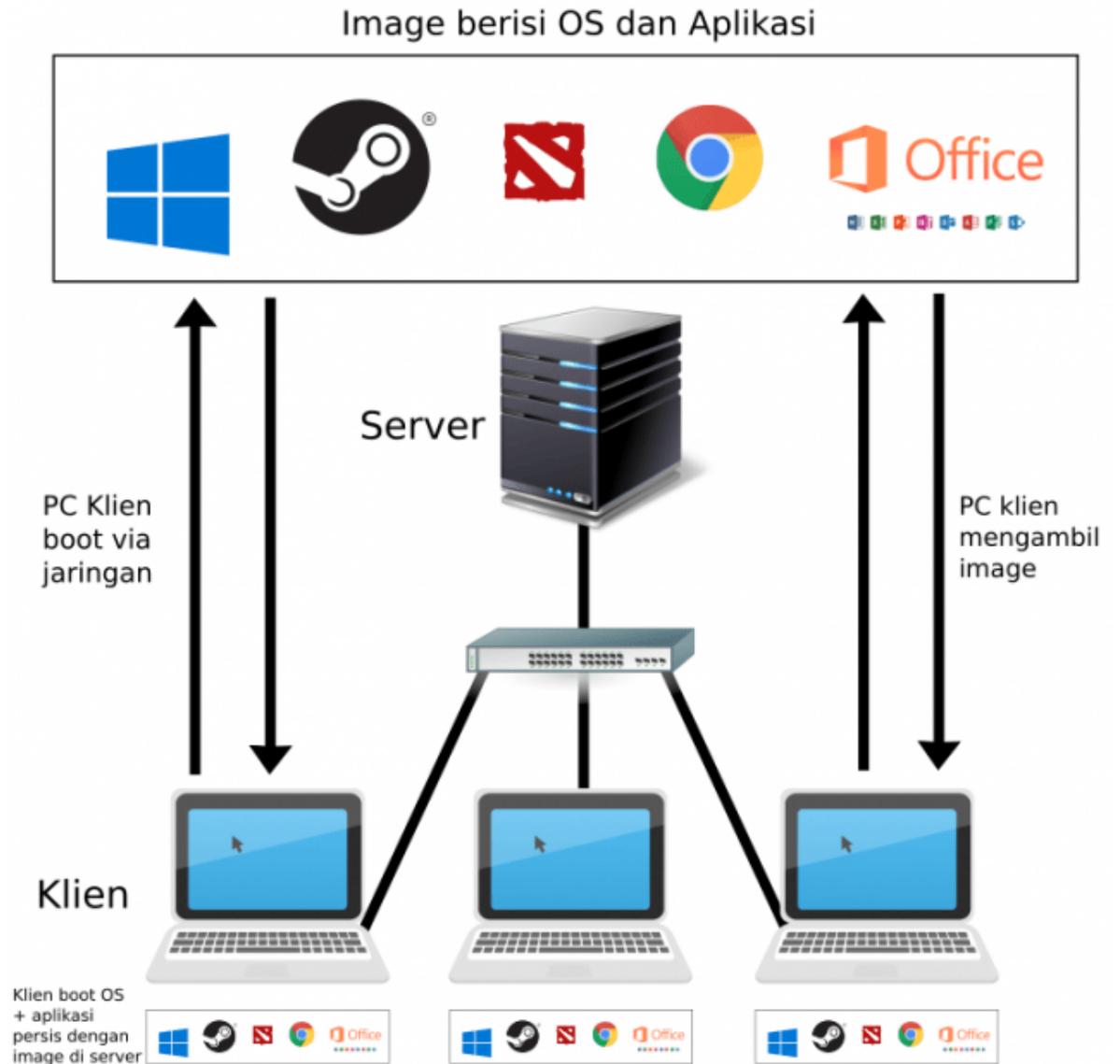
Support untuk Prospek Kedepan

# SOLUSI

**DISKLESS**

**HOW?**

**ILUSTRASI**



# FUNGSIONAL

**FIX**

Sulitnya untuk menambah aplikasi/game

Sulitnya untuk selalu update aplikasi/game

Sulitnya untuk maintenance

# ALAT DAN BAHAN

- SERVER UNBK DELL T30 Xeon Quad E3-1225v5/16Gb/1TB// 2LAN PORT
- PC client (Komputer Core I3 & LED AOC 16 Full)

- *MIKROTIK RB450G routerboard gigabit ( router indoor)*
- Switch Hub TPLINK 16 Port

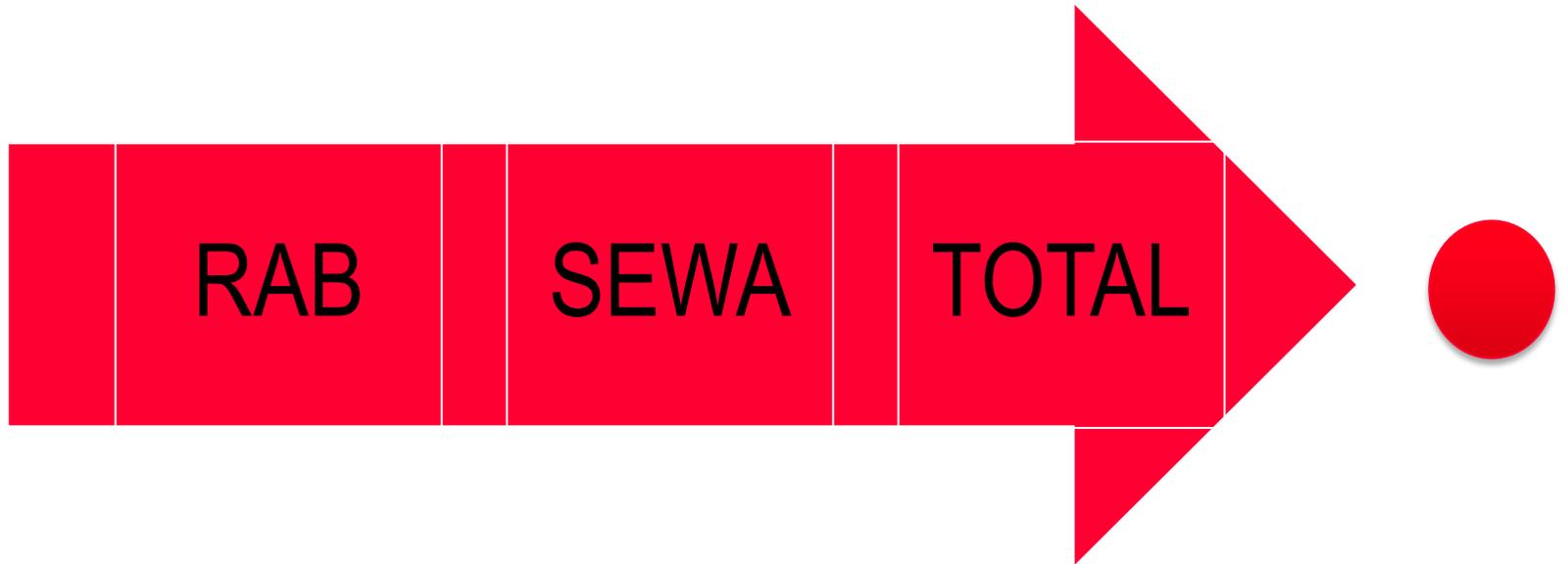
- *Cable UTP Cat 6*
- *Connector RJ45 Cat 6 AMP/COMMSCOPE (100pcs)*

- *Cyberindo Disk*
- UPS 600 VA

- *Camera CCTV SPC 2 Megapixel Tornado series Hybird*
- *Windows 7*

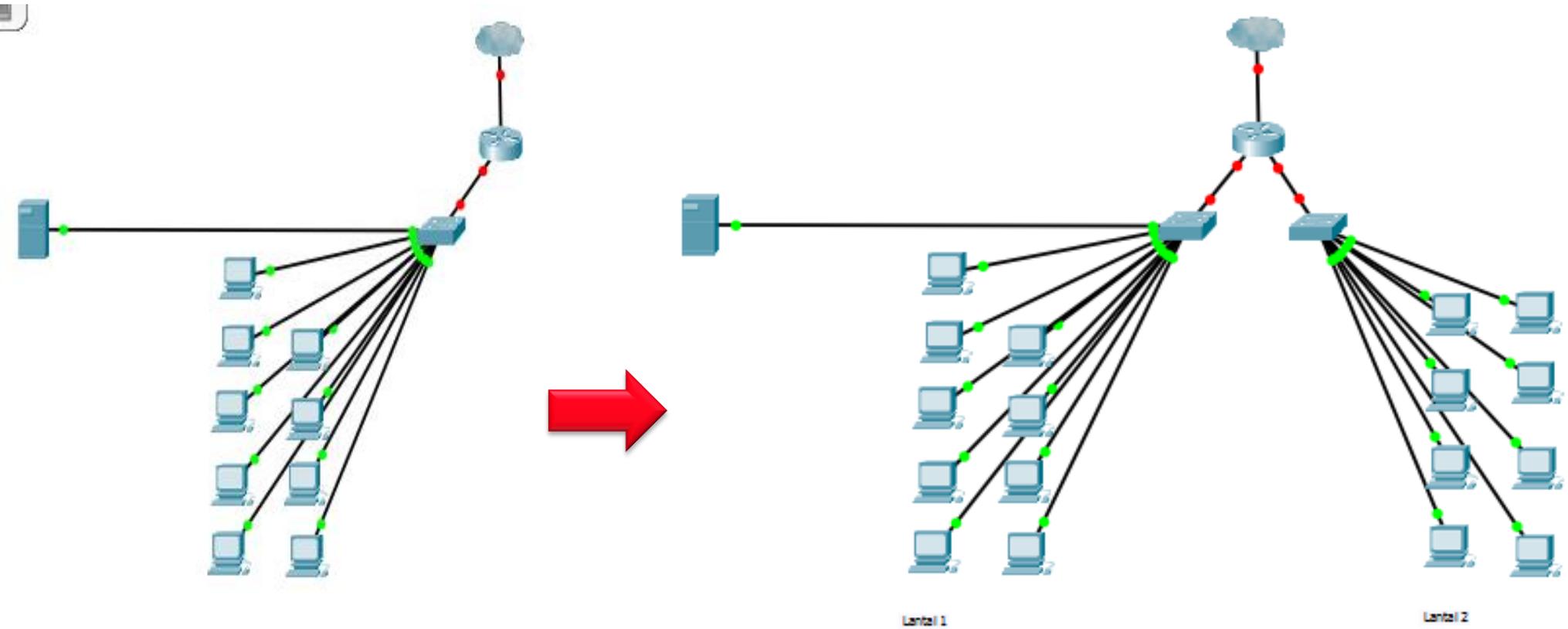
- Stop Kontak
- Kabel Tukuiki

# RAB



RAB dapat dilihat di lampiran, dalam bentuk M.S. Word

# DESIGN



# BENEFIT

Kemudahan Akses data



Biaya lebih murah tanpa hardisk



Serangan Trojan, virus, dan perubahan setting bisa dihindari



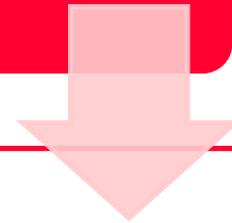
Tidak membayar banyak operator hanya satu saja operator untuk mengelola Komputer Server



Cocok untuk jenis usaha warnet, sekolah, dll

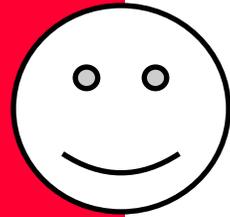
# KRITIK DAN SARAN

Perubahan mengenai penggunaan lebih dari 1 OS (*Operating System*)



Kebutuhan internet kedepannya disesuaikan oleh kebutuhan user

**SEKIAN**



**RANCANGAN ANGGARAN BIAYA 1 LANTAI**

- **Biaya Alat Dan Sarana**

Alat & Sarana	Uraian	Satuan	Biaya
Komputer server <i>Diskless AMD FM 2 + Seri A</i>	Rp 12.150.000	1 buah	Rp 12.150.000
PC client (Komputer Core I3 & LED AOC 16 Full)	Rp 2.175.000	9 buah	Rp 19.575.000
<i>MIKROTIK RB450G routerboard gigabit ( router indoor)</i>	Rp Rp1.670.000	1 buah	Rp Rp1.670.000
TPLINK TL-SF1016D SWITCH HUB 16PORT	Rp 209.000	1 buah	Rp 209.000
<i>Cable UTP Cat 6</i>	Rp 1.295.000	1 roll	Rp 1.295.000
<i>Connector RJ45 Cat 6 AMP/COMMSCOPE (100pcs)</i>	Rp 970.000	1 buah	Rp 970.000
<i>Cyberindo Disk</i>	Free	1 buah	Free
UPS 600 VA	Rp 355.000	1 buah	Rp 355.000
Camera CCTV SPC 2 <i>Megapixel Tornado series Hybird</i>	Rp 168.900	1 buah	Rp 168.900
<i>Stop Contact Kuningan 6 lubang</i>	Rp 30.0000	12 buah	Rp 360.000
<i>Kabel Tukuiki</i>	Rp 163.000	1 roll	Rp 163.000
<i>Windows 7</i>	Free	1 buah	Free
<b>TOTAL</b>			<b>Rp 36.915.000</b>

- **Biaya Sewa dan Internet**

Sewa	Satuan	Biaya
ISP Telkom Unlimited	/bulan	Rp 700.000
Perawatan PC, perlengkapan dan biaya cadangan	/bulan	Rp 500.000
Listrik	/bulan	Rp 300.000
<b>Total</b>		<b>Rp 1.800.000</b>

- **Total Pengeluaran**

<b>Modal awal</b>	<b>Rp. 100.000.000</b>
Alat dan Bahan	Rp 36.915.000
Biaya Sewa	Rp 1.800.000
<b>Total</b>	<b>Rp 38.715.000</b>
Sisa	Rp 61.285.000

Biaya sewa untuk 1 lantai adalah Rp 1,440.000/ bulan sehingga estimasi harus lebih besar dari itu. Jika perjamnya adalah sebesar 3000 maka sehari pendapatan harus minimal Rp 48.000/hari x 30 1,440,000 untuk menutupi biaya *maintenance* dan internet setiap bulannya. Dengan rincian 2 pengunjung mengakses 2 jam di 8 pc setiap harinya dikali 30 hari.

Rp 6000 x 8 pc x 30 hari = Rp 1.440.000/bulan

**INVESTASI KEDEPANNYA 2 LANTAI**

● **Biaya Alat Dan Sarana**

Alat & Sarana	Uraian	Satuan	Biaya
PC client (Komputer Core I3 & LED AOC 16 Full)	Rp 2.175.000	8 buah	Rp 17.400.000
TPLINK TL-SF1016D SWITCH HUB 16PORT	Rp 209.000	1 buah	Rp 209.000
Camera CCTV SPC 2 <i>Megapixel Tornado series Hybird</i>	Rp 168.900	1 buah	Rp 168.900
<i>Stop Contact Kuningan</i> 6 lubang	Rp 30.0000	12 buah	Rp 360.000
<b>TOTAL</b>			<b>Rp18.137.900</b>

● **Biaya Sewa dan Internet**

Sewa	Satuan	Biaya
ISP Telkom Unlimited	/bulan	Rp 700.000
Perawatan PC, perlengkapan dan biaya cadangan	/bulan	Rp 800.000
<b>Listrik</b>	/bulan	Rp 700.000
Total		Rp 2.200.000

● **Total Pengeluaran**

<b>Modal awal</b>	<b>Rp. 100.000.000</b>
Alat dan Bahan Lantai 1	Rp 36.915.000
Alat dan Bahan Lantai 2	Rp 18.137.900
Sewa	<b>Rp 2.200.000</b>
Total	Rp 57.252.900
Sisa	Rp 42.747.100

Sehingga untuk bulan pertama pengeluaran adalah sebesar Rp 59.427.900 untuk dua lantai

Biaya sewa untuk 2 lantai adalah Rp 2.200.000/ bulan sehingga estimasi harus lebih besar dari itu. Jika perjamnya adalah sebesar 3000 maka sehari pendapatan harus minimal Rp 96.000/hari x 30 = 2,880,000 untuk menutupi biaya *maintenance* dan internet setiap bulannya. Dengan rincian 2 pengunjung mengakses 2 jam di 16 pc setiap harinya dikali 30 hari.

Rp 6000 x 16 pc x 30 hari = Rp 2.880.000/bulan

Sekian  
**CV MTI Group**

## TUGAS UTS

MATA KULIAH : COMPUTER NETWORK AND BUSINESS INFORMATION

Dosen : Dr. Edi Surya Negara, M.Kom

Nama Kelompok :

1. Istikomah

2. Suriani

Analisa permasalahan dari video yang di tampilkan , Yakni; Permasalahan SIZE. pengiriman data dari klien ke server dengan kapasitas size yg sangat besar. Di server menampung besar size 1 Gb, sedangkan kecepatan pengiriman klien 100 Mb dari 50 klien, sehingga banyak data-data dari klien yang tertinggal atau hilang, rusak banyak tidak sampai.

Cara mengatasi permasalahan tersebut :

1. Mengusulkan mekanisme Pembagian TCP yang ditingkatkan dengan Teknik pembagian segmentasi data . di bagi per segmen sesuaikan kebutuhan klien. Sebagai contoh kapasitas di server 1 Gb di bagi 100mb/klien atau setiap klien 100 mb

Aplikasi tingkat Multiplexing solusi aplikasi-tingkat menghindari penggunaan beberapa sambungan paralel TCP, dan masalah yang dihasilkan, oleh multiplexing beberapa data stream ke sebuah koneksi TCP tunggal. Sejak TCP hanya menyediakan satu, mulus byte-stream abstraksi, solusi tingkat aplikasi ini termasuk skema framing untuk demarkasi aliran data individu. Contoh ini meliputi Persistent-koneksi HTTP (P-HTTP) [13], Session Control Protocol (SCP) [14] dan protokol MUX [8]. penurunan yang signifikan dalam latency dari Web pengaksesan menggunakan P-HTTP dilaporkan dalam Metode 1: Ack Bandwidth Ratio [11] Dalam metode ini, kami menghitung bandwidth ack untuk TCP windows untuk semua koneksi dalam penelusuran. The Jack bandwidth yang sehubungan dengan pengakuan A mulai didefinisikan sebagai jumlah byte yang beredar pada saat A diterima dibagi dengan waktu untuk menerima ack untuk byte luar biasa terakhir. Setiap sampel

bandwidth ack individu kemudian dibandingkan dengan bandwidth median ack untuk semua jendela dalam koneksi. Rasio ini  $( \text{bandwidth ack} ) / ( \text{bandwidth median ack} )$  memungkinkan kita untuk mengukur tingkat kompresi ack . Gambar 2 menunjukkan hasil analisis ini. The x-sumbu adalah (terkuantisasi) ack bandwidth yang rasio, dan sumbu y adalah probabilitas bahwa jendela berikutnya berisi packet loss. Kami melakukan dua pengamatan penting. Pertama, probabilitas kerugian meningkat tajam untuk nilai kecil rasio bandwidth ack . Tapi di luar titik, itu rata karena jaringan telah ditempatkan dalam keadaan macet. Metode 2: Perbandingan Dinamis Data dan Bandwidth Ack Metode kedua mempertimbangkan aliran data dan ack . Pertimbangkan jendela data yang dikirimkan oleh pengirim. Bandwidth data selama jendela dihitung dengan membagi jumlah data yang dikirim pada saat mengirimkan jendela1. Demikian pula, bandwidth ack dihitung dengan membagi jumlah data yang diakui oleh perbedaan waktu antara ack pertama dan terakhir

## 2. Pembagian waktu pengaksesan /Pemuatan TIME-WAIT

Transmission Control Protocol (TCP ) menyediakan transportasi byte-stream yang andal ke host di Internet. TCP digunakan oleh sebagian besar layanan jaringan yang membutuhkan transportasi yang andal, termasuk Hypertext Transport Protocol (HTTP ) . Metode TCP untuk mengisolasi koneksi lama dari yang baru menghasilkan akumulasi keadaan pada server yang sibuk yang dapat mengurangi tingkat throughput dan koneksi mereka. Efek pada server HTTP sangat menarik karena membawa sejumlah besar lalu lintas Internet. TCP mengharuskan titik akhir yang menutup koneksi memblokir koneksi lebih lanjut pada pasangan host / port yang sama hingga tidak ada paket dari koneksi yang tersisa di jaringan . Di bawah HTTP, host ini biasanya adalah server . Untuk memblokir sementara koneksi, satu titik akhir menyimpan salinan blok kontrol TCP (TCB) yang menunjukkan bahwa koneksi telah dihentikan baru-baru ini. Koneksi seperti itu ada dalam TIME state WAIT state . Koneksi dalam WAKTU TUNGGU dipindahkan ke TUTUP dan TCB mereka dibuang setelah cukup waktu berlalu bahwa semua paket dari koneksi yang sama telah meninggalkan jaringan. Paket meninggalkan jaringan dengan tiba di salah satu titik akhir dan ditolak, atau tiba dengan bidang TTL (time-to-live) kadaluwarsa di router dan dihapus. Untuk titik akhir yang menjadi target banyak koneksi, ribuan koneksi mungkin dalam WAKTU – MENUNGGU keadaan kapan saja, yang memperkenalkan overhead memori yang signifikan. Kami

menyebut kondisi ini sebagai pemuatan TIME-WAIT. Jika implementasi TCP titik akhir mencari semua TCB saat mengirimkan paket, WAKTU loading MENUNGGU akan secara langsung mempengaruhi kinerjanya. Kehadiran banyak WAKTU IT TUNGGU TCB dapat meningkatkan waktu demultipleks untuk koneksi aktif. Kami hav e dilihat throughput yang penurunan sebesar 50% pada titik akhir dimuat, dan efek pada server komersial telah dicatat di tempat lain . Beberapa implementasi TCP

mengatasi masalah demultiplexing tanpa mengatasi beban memori. Desain TCP menempatkan TIME – TUNGGU TCB di titik akhir yang menutup koneksi; keputusan ini bertentangan dengan semantik banyak protokol aplikasi.

Fungsi WAKTU –TUNGGU adalah untuk mencegah paket-paket tertunda dari satu koneksi diterima oleh koneksi kemudian. Koneksi konkuren diisolasi oleh mekanisme lain, terutama oleh alamat, port, dan nomor urut.

## UJIAN TENGAH SEMESTER

Mata Kuliah : COMPUTER BUSSINESS DAN NETWORK

Dosen : Dr.Edy Surya Negara, M.Kom

Nama Mahasiswa :

1. Andriansyah , NIM 192420020

2. Wahyudi Saputra, NIM 192420015

### **Masalah :**

1. Beban Kerja server yang berlebihan dari kapasitas yang ada sehingga terjadi Congestion pelayanan dalam pengiriman dan penerimaan paket-paket data dari host satu ke host yang lainnya dan pengaruh limit bandwith pun mempengaruhi beban kerja sehingga server busy kemungkinan terjadi.
2. Terjadi collision didalam jaringan yang membuat tabrakan paket-paket data yang dikirimkan oleh masing masing host sehingga hal itu membuat paket data yang ada menjadi menunggu sehingga server menjadi busy.

### **Solusi:**

1. Pengurangan beban kerja atau memberikan limit terhadap server yang terhubung dengan client misalkan pemberian akses client untuk 500-1000 host saja, sehingga server tidak terlalu busy dalam melayani paket-paket data yang ada.
2. Membuat sistem keamanan jaringan yang baik sekaligus meningkatkan keamanan sehingga tidak terjadi serangan DDOS yang sangat memungkinkan server menjadi busy.
3. Melakukan Network Congesti Control yang difokuskan untuk mengurangi kongesti pada sistem jaringan.

# Tentukan masalah dan solusi dari video “Why internet slows down when it's busy – Computerphile”

**Yudy Pranata**

*Mahasiswa Magister Teknik Informatika Universitas Bina Darma*

*Jl. Jend. A. Yani No. 12, Plaju, Palembang 30264*

*yudypranata26@gmail.com*

Pada video “*Why internet slows down when it's busy – Computerphile*” yang telah diberikan, terdapat penjelasan yang menyebutkan bahwa penyedia jasa jaringan internet selalu menjanjikan kapasitas besar dan cepat dalam jaringan untuk menarik banyak peminat. Seperti didalam video dengan contoh penyedia jasa jaringan internet memiliki total kapasitas jaringan server sebesar 1 (satu) Gb/sec dan mereka memberikan “janji” kecepatan dan kapasitas jaringan internet kepada pengguna (konsumen) 100 Mb/sec. Tetapi yang sebenarnya mereka tidak bisa menjanjikan bahwa kapasitas tersebut dengan penuh dapat digunakan, karena pada saat menggunakan internet kapasitas jaringan yang diberikan tetap mengacu pada jumlah pengguna (konsumen) dan seberapa besar menggunakan kapasitas jaringan tersebut pada saat yang bersamaan.

## **Solusi**

Berdasarkan pada penelitian *TIME-WAIT loading* [1] dalam percobaan dilakukan pada sebuah *workstation* yang terhubung ke Myrinet LAN sebesar 640 Mb/sec [2]. Dengan adanya 3 (tiga) penelitian yang pertama membuktikan bahwa beban TCB menurunkan kinerja server dan modifikasi yang digunakan akan menurunkan degradasi tersebut. Penelitian kedua menggambarkan dari kedua solusi TCP [3] dan HTTP [4] dapat meningkatkan kinerja server dibawah tolak ukur WebSTONE [5]. Dan penelitian yang terakhir menggambarkan bahwa modifikasi yang dibuat memungkinkan server untuk mendukung beban HTTP yang tidak dapat digunakan pada konfigurasi standar. Dapat dijelaskana bagaimana TCP berinteraksi dengan protokol aplikasi tertentu untuk memuat server dengan *TIME-WAIT* TCB dan menunjukkan dari tindakan ini. Interaksi tersebut adalah hasil langsung dari penyederhanaan asumsi bahwa titik akhir dalam penutupan koneksi tidak bergantung untuk selanjutnya membangun kembali koneksi tersebut. Dengan mangajukan usulan pada titik akhir yang dapat menahan *TIME-WAIT* selama pembentukan kembali koneksi. System tersebut akan memerlukan fungsi TCP untuk mengalokasikan *TIME-WAIT* TCB ke ujung koneksi yang tepat tanpa mengganggu protokol aplikasi atau meninggalkan kelemahan pada TCP. Pada tahap implementasi dan menguji solusi pada TCP yang lebih sederhana dan solusi pada HTTP untuk menunjukkan beban

*TIME-WAIT* dari klien ke server dan dalam kondisi tersebut dapat dipastikan bahwa beban *TIME-WAIT* dapat mempengaruhi kinerja server dibawah SunOS 4.1.3 yang hasilnya dapat berkurang sebanyak 50 %. Sementara itu, dengan menggunakan WebSTONE dapat dibuktikan bahwa HTTP pada klien dan server yang menggunakan system mereka memiliki hasil yang lebih besar dibandingkan dengan SunOS 4.1.3. Klien dan server yang telah menggunakan system konfigurasi mereka dapat mendukung tingkat koneksi yang lebih tinggi dari pada system yang belum dimodifikasi pada konfigurasi tertentu. Mereka juga telah menunjukkan bahwa sistemnya dapat dikombinasikan dengan koneksi HTTP yang persisten menggunakan lebih sedikit memori daripada system SunOS 4.1.3 yang tidak dimodifikasi menggunakan koneksi untuk beban klien tertentu.

	TCP with TIME- WAIT Negotiation	TCP with Clien <RST>	CLIENT_CLOSE HTTP Extension
Reduces TIME-WAIT loading	Yes	Yes	Yes
Compatible with Current Protocols	Yes	Yes	Yes
Changes are Effective if only the client is modified	No	Yes	Yes
Allows system to prevent TIME-WAIT assassination	Yes	No	Yes
No changes to Transport Protocol	No	No	Yes
No changes to Application Protocols	Yes	Yes	No
Adds no packet exchanges to Modified Protocol	Yes	No	No
TIME-WAIT allocation is a requirement of connection establishment	Yes	No	No

**Table 4: Summary of Proposed Systems [1]**

## **Referensi**

1. Faber Theodore, Joseph D. Touch, and W. Yue. The TIME-WAIT state in TCP and its effect on busy servers. In *Proceedings of IEEE Infocom* (March 1999).
2. Myricom, Inc., Nannette J. Boden, Danny Cohen, Robert E. Felderman, Alan E Kulawik, Charles L. Seitz, Jakov N. Selovic, and Wen-King Su, "Myrinet: A Gigabit-per-second Local Area Network." *IEEE Micro*, pp.29-36, IEEE (February 1995).
3. Jon Postel, ed., "Transmission Control Protocol," *RFC-793/STD-7* (September, 1981).
4. R. Fielding, J. Gettys, J. Mogul, H. Frystyk, and T. Berners-Lee, "Hypertext Transport Protocol – HTTP/1.1," *RFC-2068* (January, 1997).
5. Gene Trent and Mark Sake, "WebSTONE: The First Generation in HTTP Server Benchmarking," *white paper*, Silicon Graphics International (February, 1995).

Penyedia jaringan atau hosting membuat asumsi tentang bagaimana banyak orang menggunakan layanan pada tampilan sendiri dan asumsi yang memungkinkan, yaitu disebut Multiplex, ketentuan tersebut memberikan setiap pengguna seratus megabyte per detik.

Sebagian dari penyedia jaringan berasumsi bahwa jaringan mereka benar-benar digunakan oleh pengguna sepanjang waktu. sehingga apa yang mereka lakukan yang saya bisa, Vogel saya melihat yang Anda akan hening sehingga kami juga tidak dapat membuat bambu ini hari rabu, kamu akan mendapatkan cukup waktu untuk itu apakah kamu pergi untuk menarik kapasitas dan kamu punya banyak dan banyak berlangganan saffron akan datang hari ini beberapa dari ini kehilangan sebagian besar orang-orang ini kanvas jika kita menggunakan aplikasi ini sepenuhnya benar semua waktu jadi kita semua pergi ke sini maka itu tidak akan cocok bahkan bisa dimiliki.

koneksi paralel antara klien dan server adalah pengguna jaringan yang lebih agresif daripada aplikasi yang menggunakan koneksi TCP tunggal. Throughput berkorelasi positif dengan jumlah koneksi aktif. Ketika beberapa koneksi secara bersamaan aktif dan salah satunya mengalami kerugian, hanya setengah dari yang tersisa yang rata-rata mengalami kerugian telah menyajikan analisis rinci tentang perilaku TCP dari server Web yang sibuk. Analisis kami telah berfokus pada dua bidang utama: memeriksa kinerja koneksi TCP individual yang membawa muatan HTTP, dan menguji efek merugikan dari bagaimana browser Web menggunakan koneksi TCP paralel pada kinerja jaringan secara keseluruhan

Teknik pemulihan kerugian yang ada tidak efektif dalam menangani packet loss dan teknik baru harus dikembangkan untuk menanganinya. Hampir 50% dari semua kerugian membutuhkan waktu tunggu yang kasar untuk pulih. Pengiriman ulang cepat pulih dari kurang dari 45% dari semua kerugian. The sisa kerugian yang selama start lambat berikut timeout. • Implementasi jaringan di masa depan harus meningkatkan ukuran buffer socket default untuk menghindari jendela penerima dari menjadi Gambar 12. Topologi untuk tes simulasi. node R1 node R2 node S router router Waktu (detik) Nomor Urutan Paket Gambar 13. Tiga transfer TCP-INT dari dua host melalui router bottleneck tunggal. Koneksi A berasal dari host pertama dan dimulai pada waktu 0. Koneksi B & C berasal dari host kedua dan mulai pada waktu 0 dan 15 detik masing-masing. ACB 0 100 200 300 400 500 600 700 800 900 1000 0 5 10 15 20 25 30 35 40 0 100 200 300 400 500 600 700 800 900 1000 0 5 10 15 20 25 30 35 40 Waktu (dt) Paket Urutan Paket Nomor Gambar 14 Tiga transfer TCP-INT menggunakan algoritma kontrol kemacetan untuk proksi dari dua host melalui router bottleneck tunggal. Koneksi A berasal dari host pertama dan dimulai pada waktu 0. Koneksi B & C berasal dari host kedua dan mulai pada waktu 0 dan 15 detik masing-masing. Bottleneck ACB. Ukuran buffer socket membatasi throughput sekitar 14% dari semua koneksi yang diamati.

Hasil simulasi menunjukkan bahwa pendekatan kami mencapai perilaku start-up yang jauh lebih baik, pemulihan kerugian, dan berbagi bandwidth di antara koneksi paralel dari sejumlah host. Kami saat ini menerapkan pemulihan kerugian yang ditingkatkan dan teknik koneksi terintegrasi dan memeriksa kinerjanya di lingkungan server yang sibuk di dunia nyata. Selain itu, sebagai bagian dari pengumpulan data kami, kami juga menggunakan traceroute untuk mengumpulkan informasi topologi jaringan untuk sebagian besar klien yang mengunjungi server Web. Kami berencana untuk menggunakan informasi topologi ini untuk memeriksa kemungkinan memperluas kontrol kemacetan terintegrasi dan metode pemulihan kehilangan untuk berbagi informasi di seluruh koneksi dari host terdekat serta dari koneksi yang berasal dari host yang sama, memperluas untuk hasil yang disajikan.

Nama : 1. Sela Taramita (192420038)  
2. Yuni Astuti (192420004)

Mata Kuliah : Computer Network and Business Information

Kelas : AR 1

## “ Why Internet Slows Down When Its Busy “

Permasalahan :

Permasalahan yang terdapat pada video yang telah kami tonton yaitu “*Why internet slows down when it's busy – Computerphile*” terdapat penjelasan yang menyebutkan bahwa penyedia jasa jaringan internet selalu menjanjikan kapasitas besar dan cepat dalam jaringan akan tetapi yang sebenarnya terjadi ketika kita terkoneksi dengan internet system jaringan tersebut mengalami queue (antrian) dimana semakin banyak pengguna semakin sibuk sistem antrian itu mengatur traffic input output datanya. Misalnya yang terdapat pada video, penyedia jasa jaringan internet memiliki total kapasitas jaringan server sebesar 1 (satu) Gb/sec dan mereka memberikan kecepatan dan kapasitas jaringan internet kepada pengguna (konsumen) 100 Mb/sec.

Solusi :

1. Harus meningkatkan standar ukuran *socket buffer* untuk menghindari hambatan dari penerima yang lain
2. Menggunakan 2 metode yaitu :
  - a. Ack bandwidth ratio  
Pada metode ini, mengkalkulasikan ack bandwidth pada jendela TCP untuk semua jejak koneksi
  - b. Perbandingan data dan Ack Bandwidth  
Pada metode kedua ini, mempertimbangkan aliran dari data dan ack bandwidth dimana mempertimbangkan data yang di transmisikan oleh pengirim
3. Meningkatkan kinerja beberapa koneksi TCP  
Untuk dapat meningkatkan kinerja koneksi maka koneksi tersebut dibagi dua dimana kinerja koneksi baru tidak harus menjalani *slow-start* untuk memperkirakan *traffic* yang

benar, ini mengarah untuk peningkatan kinerja koneksi tambahan. Data-driven recovery terintegrasi di set koneksi TCP. Ketika sebuah paket yang hilang pada satu koneksi, ada keberhasilan pengiriman paket di koneksi yang lain yang memungkinkan pengiriman dapat mendeteksi paket yang loss tanpa menggunakan timeout untuk meningkatkan kinerja tambahan.

**Nama : A. Firdaus**

**Nim : 192420043**

**Kelas : MTIR2**

## **Tugas**

**Pertanyaan : Mengapa Internet melambat ketika jaringan sedang sibuk dari studi kasus diberikan melalui video yang diunggah oleh Dr Richard Mortier**

A. Latar Belakang :

Menurut jurnal Ilmu Komputer Universitas California :

Dalam kasus tersebut dimana pertumbuhan internet semakin pesat dan banyak transisi perpindahan akses peneliti menganalisis dengan dua metode yaitu single Connection Behavior dan Parallel Connection Behavior

1. Single Connection Behavior :

- a. Loss Recovery : Bagaimanakah Pemulihan dan efektifitas pemulihan data dalam menghindari saat timeout
- b. Receiver Bottleneck : Berapa Seringkah Kinerja TCP menerima halaman dengan waktu yang terbatas
- c. Ack compression : Seberapa sering terjadi ack compression saat kehilangan data

2. Parallel Connection Behavior :

- a. Bagaimana klien melihat jumlah koneksi saat server dibuka
- b. Congestion control: bagaimana reaksi koneksi paralel menerima saat kehilangan data yang disebabkan koneksi timeout
- c. Loss behavior : Bagaimana data yang didistribusikan saat koneksi sedang padat

B. Analisa :

Didalam Penelitian tersebut dijelaskan bagaimana cara menganalisa tentang koneksi TCP saat koneksi sedang sibuk dan seberapa baik pemulihan yang

terjadi. Ada beberapa analisa tentang bagaimana cara memperbaiki data yang diterima saat melakukan koneksi jaringan antara lain :

### 1. Single Connection Behavior

Total connections	16501	
With packet re-ordering	9703	6
With receiver window as	2339	14
Total packets	78216	
During slow-start	66620	85
# of slow-start pkts lost	3545	5
During congestion	11595	15
# of congestion	8218	7
Total retransmissions	8571	
Fast retransmissions	3753	44
Slow-start retransmissions	5981	7
Coarse timeouts	4220	49
Avoidable with SACKs	1871	4
Avoidable with enhanced	1042	25

Pada tabel diatas saat melakukan single connection dimana ukuran data yang panggil lebih besar daripada data yang diterima

### 2. Parallel Connection Behavior

Analisa Koneksi yang terbuang saat melakukan koneksi secara bersamaan

### 3. Key Results of Trace Analysis

Dari penelitian yang dikembangkan dimana koneksi tunggal lebih baik dalam pemulihan pengiriman data ketimbang dengan koneksi yang bersamaan

## C. Pembahasan

Ada beberapa metode yang dapat dilakukan antara lain :

### 1. Application-level Multiplexing

Solusi tingkat aplikasi menghindari penggunaan beberapa koneksi TCP paralel, dan masalah yang dihasilkan, dengan multiplexing beberapa

aliran data ke koneksi TCP tunggal. Karena TCP hanya menyediakan abstraksi byte-stream tunggal

Tapi ada beberapa kekurangan antara lain :

- a. Mereka membutuhkan tambahan dan diperlukan server dan klien yang baik.
- b. Mereka tidak mengizinkan multiplexing transfer yang dilakukan oleh lebih dari satu aplikasi.
- c. Multiplexing melalui koneksi TCP tunggal memperkenalkan kopling yang tidak diinginkan antara transfer data yang secara logis independen

## **2. TCP-INT Implementation**

Menggambarkan penerapan kontrol kemacetan terintegrasi dan skema pemulihan kehilangan yang hanya mengubah TCP pada pengirim. Untuk setiap host yang terkait dengan satu mesin, tumpukan TCP / IP membuat struktur (Gambar 9) untuk menyimpan informasi tentang komunikasi apa pun. Struktur baru ini memungkinkan kontrol kemacetan bersama yang diinginkan dan pemulihan kerugian dengan menyediakan satu titik koordinasi untuk semua koneksi ke host tertentu. Struktur host berisi variabel-variabel standar TCP yang terkait dengan pemeliharaan jendela kongesti TCP (cwnd, ssthresh dan count). Struktur ini juga memperkenalkan beberapa variabel baru untuk membantu dalam kontrol kemacetan (ownd, decr\_ts) dan variabel lain untuk mendukung pemulihan kerugian terintegrasi (pkts []). Di subbagian berikut, kami menjelaskan bagaimana berbagai rutinitas TCP menggunakan dan memperbarui informasi baru ini. Rutin pengiriman data baru: Ketika suatu koneksi ingin mengirim suatu paket, ia memeriksa untuk melihat apakah jumlah byte yang sudah ada dalam "pipa" dari pengirim (ownd) lebih besar daripada ukuran "pipa" yang diinginkan (cwnd). Jika tidak, koneksi akan menyiapkan paket yang akan dikirim dengan menambahkan entri pada ekor daftar paket yang beredar. Entri ini berisi ukuran nomor urut dan stempel waktu dari paket yang dikirimkan. Ketika paket dikirim, koneksi menambah kepemilikan

dengan ukuran paket. Kami menggunakan penjadwalan round-robin di seluruh koneksi meskipun itu bukan persyaratan penting. Rutin `recv ack` baru: Ketika `ack` baru tiba, pengirim meningkatkan variabel `cwnd` yang sesuai. Juga pada saat kedatangan `ACK` baru, pengirim menghapus paket dari daftar `pkts []` yang telah mencapai

### **3. Integrated Congestion Control/Loss Recovery**

Motivasi untuk kontrol kemacetan terintegrasi dan pemulihan kerugian adalah untuk memungkinkan aplikasi menggunakan koneksi TCP terpisah untuk setiap transfer (seperti yang mereka lakukan hari ini), tetapi untuk menghindari masalah yang disebutkan dalam Bagian 3.2 dengan membuat modifikasi yang sesuai untuk tumpukan jaringan. Kami membagi fungsionalitas TCP ke dalam dua kategori: yang berkaitan dengan abstraksi TCP byte-stream yang andal, dan yang berkaitan dengan kontrol kemacetan dan pemulihan kerugian berbasis data. Yang terakhir ini dilakukan secara terintegrasi di set koneksi paralel. Kami menyebutnya versi modifikasi TCP `TCP-Int`. Dengan membuka `n` koneksi TCP terpisah untuk transfer, aplikasi memiliki `n-stream` stream yang andal dan independen untuk digunakan. Kontrol aliran untuk setiap koneksi terjadi secara independen dari yang lain, sehingga pengiriman data ke aplikasi penerima juga terjadi secara independen untuk setiap koneksi. Pada saat yang sama, kontrol kemacetan terintegrasi di seluruh koneksi TCP. Ada satu jendela kemacetan untuk set koneksi TCP antara klien dan server yang menentukan jumlah total data luar biasa yang dimiliki oleh set koneksi dalam jaringan. Ketika terjadi kerugian pada salah satu koneksi,

### **4. Simulation Results: One Client Host Case**

Cara ini menjelaskan hasil dari simulasi `ns` yang dirancang untuk menguji kontrol kemacetan dengan pengenalan integer dan pemulihan kehilangan pada koneksi TCP simultan. Tes pertama menggunakan topologi pada Gambar 7. Ukuran buffer router diatur ke 3 paket. Ini cukup kecil untuk memaksa transfer untuk memiliki jendela kemacetan kecil dan sering mengalami kerugian. Sekali lagi, topologi dan parameter dipilih untuk menciptakan kembali situasi yang sering terjadi di jejak kami, dan

tidak meniru jaringan yang sebenarnya. Dalam tes ini, node pengirim melakukan 4 transfer TCP ke penerima. Transfer mulai dari 0, 2, 4 dan 6 detik dan semua berakhir pada 10 detik. Pilihan aktual dari nilai 0, 2, 4, dan 6 tidak penting, hanya saja waktu mulai setiap koneksi terhuyung-huyung dalam waktu. Gambar 10 menunjukkan plot urutan untuk pengujian menggunakan pengirim berbasis SACK. Ini menunjukkan bahwa biasanya hanya satu koneksi berperforma memuaskan pada satu waktu. Misalnya, pada waktu 2 detik, koneksi yang memulai mengalami beberapa kerugian awal dan dipaksa untuk memulihkannya melalui timeout kasar. Faktanya, koneksi ini tidak mengirim sejumlah besar data hingga 4 detik kemudian (pada waktu 6 detik). Selama periode 10 detik, koneksi dimulai pada waktu 2 detik dan waktu 6 detik. Akun untuk fraksi sangat kecil (<10%) dari total byte yang ditransfer. Ketidakadilan dan kinerja yang tidak dapat diprediksi (karena timeout kasar) tidak diinginkan dari sudut pandang aplikasi karena koneksi yang membawa data penting dapat diperlambat sementara yang lain yang membawa data yang kurang penting lebih baik.

##### **5. Simulation Results: Multiple Client Hosts Case**

Simulasi ini menggunakan topologi jaringan dimana saat waktu mulai koneksi di waktu 0, transfer TCP tunggal dimulai setengah dari bandwidth dapat diterima kembali. berfungsi untuk mengimplementasikan fungsi yang lebih kompleks dalam limit bandwidth, dimana penggunaan packet mark nya memiliki fungsi yang lebih baik. Digunakan untuk membatasi satu arah koneksi saja baik itu download maupun upload. Secara umum Queue Tree ini tidak terlihat berbeda dari Simple Queue. Perbedaan yang bisa kita lihat langsung yaitu hanya dari sisi cara pakai atau penggunaannya saja. Dimana Queue Simple secara khusus memang dirancang untuk kemudahan konfigurasi sementara Queue Tree dirancang untuk melaksanakan tugas antrian yang lebih kompleks dan butuh pemahaman yang baik tentang aliran trafik

UTS

**Computer Network and Communication (MTIK 111)**

Nama : 1. M. Nang Alhafiz (192420008)

2. Andrian Perdana (192420017)

Permasalahan :

- Dalam satu keadaan dimana kapasitas internet yang diberikan oleh ISP (Internet Service Provider) sebesar 1 Gbps dan dibagikan ke pelanggan sebesar 100 Mbps per pelanggan, jika suatu waktu ada 50 pelanggan yang mengakses internet dengan masing-masing pengguna menggunakan 100 mbps, maka kapasitas total menjadi 5 Gbps, sehingga menyebabkan antrian paket data untuk melewati oleh server jaringan karena melebihi kapasitas bandwidth yang tersedia dan menyebabkan akses internet menjadi lambat. Ketika memori server sudah penuh, ini dapat menyebabkan server membuang paket data yang ada dalam antrian data untuk ditransfer.

Solusi :

1. Memodifikasi waktu tunggu / Time-Wait server dalam pertukaran data sehingga bisa mengurangi 50 % kehilangan paket akibat antrian;
2. Memodifikasi TCP untuk bertukar waktu tunggu TCB setelah penutupan yang sukses;
3. Memodifikasi HTTP untuk mendorong klien untuk menutup koneksi TCP;
4. Penerapan IP versi 6 dapat meningkatkan kecepatan koneksi pertukaran data;
5. Implementasi jaringan di masa depan harus meningkatkan ukuran buffer socket default untuk menghindari hambatan jendela penerima;
6. Menggunakan kumpulan koneksi secara parallel antara client dan server sehingga pengguna lebih agresif daripada menggunakan koneksi TCP tunggal;
7. Memodifikasi node server untuk menunjukkan kapan pengiriman ulang TCP terjadi dengan mengirimkan paket yang di cache.

## UJIAN TENGAH SEMESTER

Mata Kuliah : COMPUTER BUSSINESS DAN NETWORK

Dosen : Dr.Edy Surya Negara, M.Kom

Nama Mahasiswa :

1. Andriansyah , NIM 192420020

2. Wahyudi Saputra, NIM 192420015

### **Masalah :**

1. Beban Kerja server yang berlebihan dari kapasitas yang ada sehingga terjadi Congestion pelayanan dalam pengiriman dan penerimaan paket-paket data dari host satu ke host yang lainnya dan pengaruh limit bandwith pun mempengaruhi beban kerja sehingga server busy kemungkinan terjadi.
2. Terjadi collision didalam jaringan yang membuat tabrakan paket-paket data yang dikirimkan oleh masing masing host sehingga hal itu membuat paket data yang ada menjadi menunggu sehingga server menjadi busy.

### **Solusi:**

1. Pengurangan beban kerja atau memberikan limit terhadap server yang terhubung dengan client misalkan pemberian akses client untuk 500-1000 host saja, sehingga server tidak terlalu busy dalam melayani paket-paket data yang ada.
2. Membuat sistem keamanan jaringan yang baik sekaligus meningkatkan keamanan sehingga tidak terjadi serangan DDOS yang sangat memungkinkan server menjadi busy.
3. Melakukan Network Congesti Control yang difokuskan untuk mengurangi kongesti pada sistem jaringan.

# WHY INTERNET SLOWS DOWN WHEN ITS BUSY ???

Mata Kuliah : *Computer Network And Business Information*  
Dosen : Dr. Edi Surya Negara, M.Kom.  
Kelas : Reguler **AR1**

Nama : Ardiansyah (192420013)  
M.Afdhaluddin (192420012)

Dari video yang berjudul “*WHY INTERNET SLOWS DOWN WHEN ITS BUSY ???*” maka, kami berdua mempunyai analisa tentang masalah yang dihadapi jaringan internet dalam pengiriman data yang mempunyai multi client terhadap sebuah server

## **Problem :**

Akses internet kepada suatu server dalam waktu yang bersamaan, dimana server yang diakses oleh 50 orang secara bersamaan kedalam suatu server yang mempunyai size sebesar 1 GB, tetapi setiap user mempunyai akses pengiriman sebesar 100 mb /s. yang menyebabkan server menjadi down sehingga membuat beberapa data menjadi rusak dan hilang.

## **Solusi :**

1. Memberikan **pembagian akses data** dari kecepatan berdasarkan data prioritas *Hight Priority, Mid Priority, Low Priority*, yang dimana data tertinggi terlebih dahulu diterima lalu data yang terendah menggunakan *traffic monitoring system*.
2. Menggunakan **control waktu**, dimana client dibagi bedasarkan golongan atau type. Lalu diberikan hak waktu akses yang berbeda , sehingga tidak menyebabkan waktu akses yang bertabrakan atau waktu koneksi berdasarkan wilayah yang dinamakan waktu tunggu yang disimpan didalam mbufs.
3. **Protocol communication** yang diperbarui dengan model *OSI* yang dimana model protocol server yang mengontrol akses di server.

# PENGATURAN JARINGAN

Pada Suatu Dinas Provinsi Sumatera Selatan

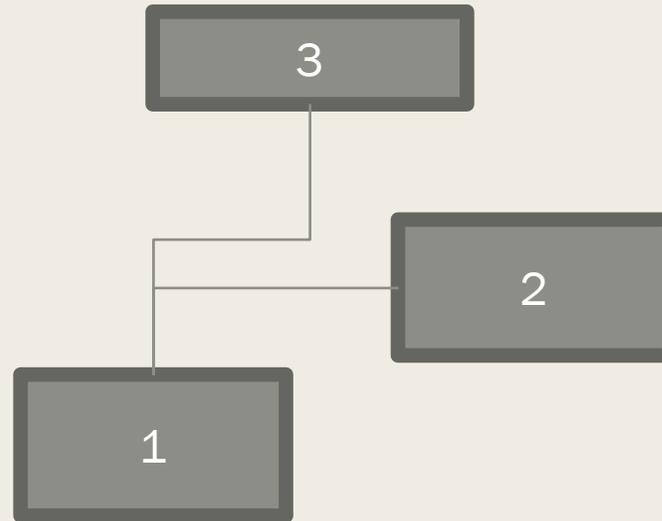
Anggota Kelompok:

- Bhagaskara
- Marhadi Wijaya

# Pemmasalahan/Permintaan

- Terdapat koneksi internet setiap ruangan di kantor dinas
- Setiap ruangan terdapat pengaturan bandwidth masing-masing
- **Rencana kedepan** terdapat server yang terpusat

# Ruang Lingkup Kantor Dinas

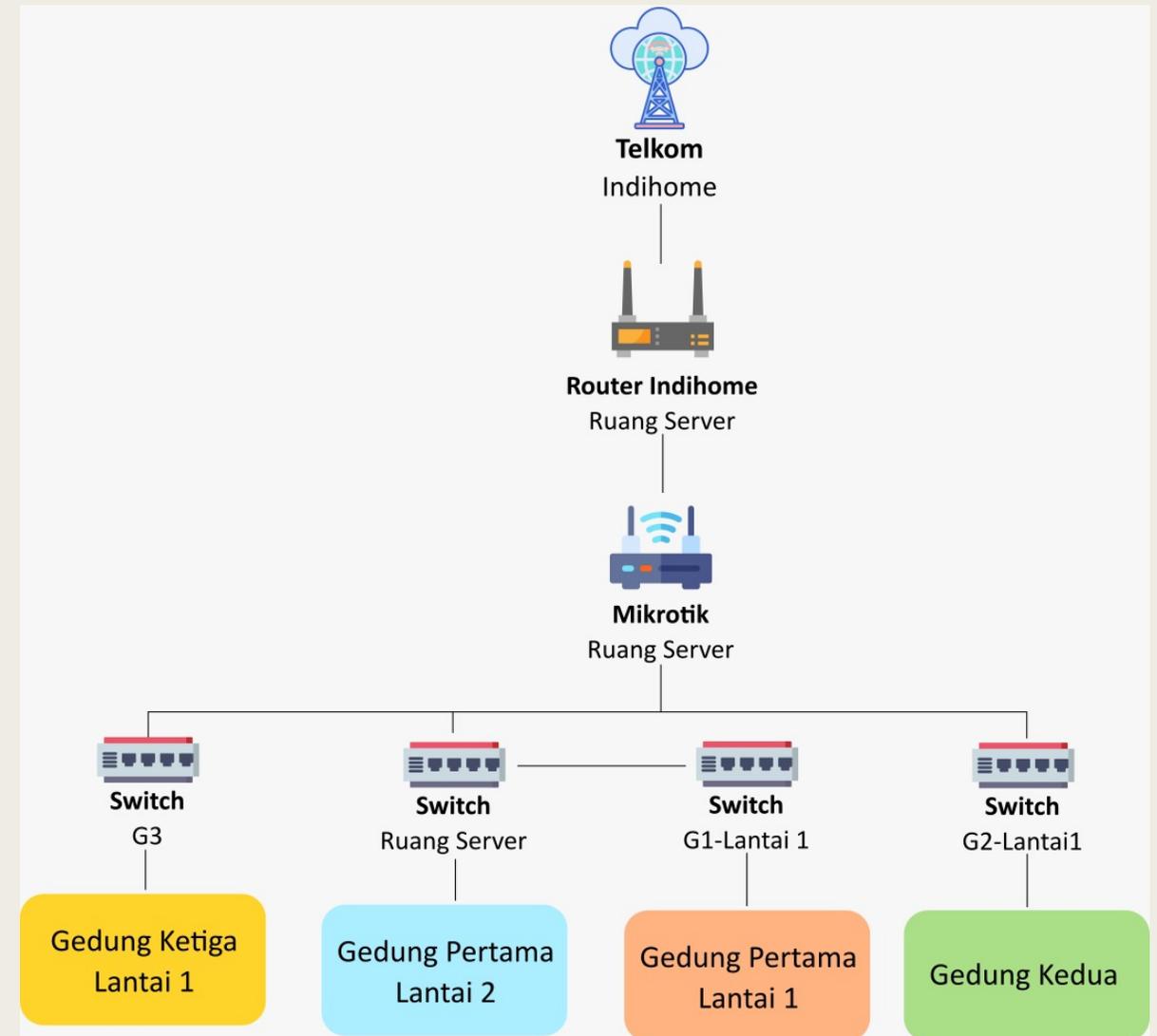


Gedung 1	Gedung 2	Gedung 3
Lantai 1: - Terdapat 7 Ruangan	Lantai 1: - Terdapat 2 Ruangan	Lantai 1: - Terdapat 1 Ruangan
Lantai 2: - Terdapat 7 Ruangan	Lantai 2: - Terdapat 1 Ruangan	

# Topologi Jaringan

Untuk menjawab permasalahan yang ada maka dibentuklah topologi seperti disamping:

1. Untuk menjawab masalah setiap ruang terdapat koneksi internet dan pengaturan bandwidth, maka setiap ruangan diberikan Access point masing-masing dengan perangkat **Mikrotik** sehingga bandwidth dapat diatur.
2. Untuk menjawab rencana kedepan terdapat server yang terpusat, maka disediakan ruang khusus untuk pusat pengaturan jaringan dan server nantinya yang disebut Ruang Server



# Rencana Anggaran Biaya

## Biaya Instalasi

No	Uraian	Jumlah	Biaya Satuan	Total
1	Mikrotik RB 941	19	380.000	7.220.000
2	Kabel Cat6	1	1.625.000	1.625.000
3	RJ 45	1	100.000	100.000
4	Switch D-Link	4	130.000	520.000
5	Tang Crimping	2	50.000	100.000
6	LAN Tester	2	50.000	100.000
<b>Total</b>				<b>9.665.000</b>

## Biaya Bulanan

ISP Indihome Up to 100 Mbps dengan biaya Rp 1.560.000/bulan

The image features two thick black L-shaped corner brackets. One is positioned in the top-left corner, and the other is in the bottom-right corner. They are oriented towards each other, framing the central text.

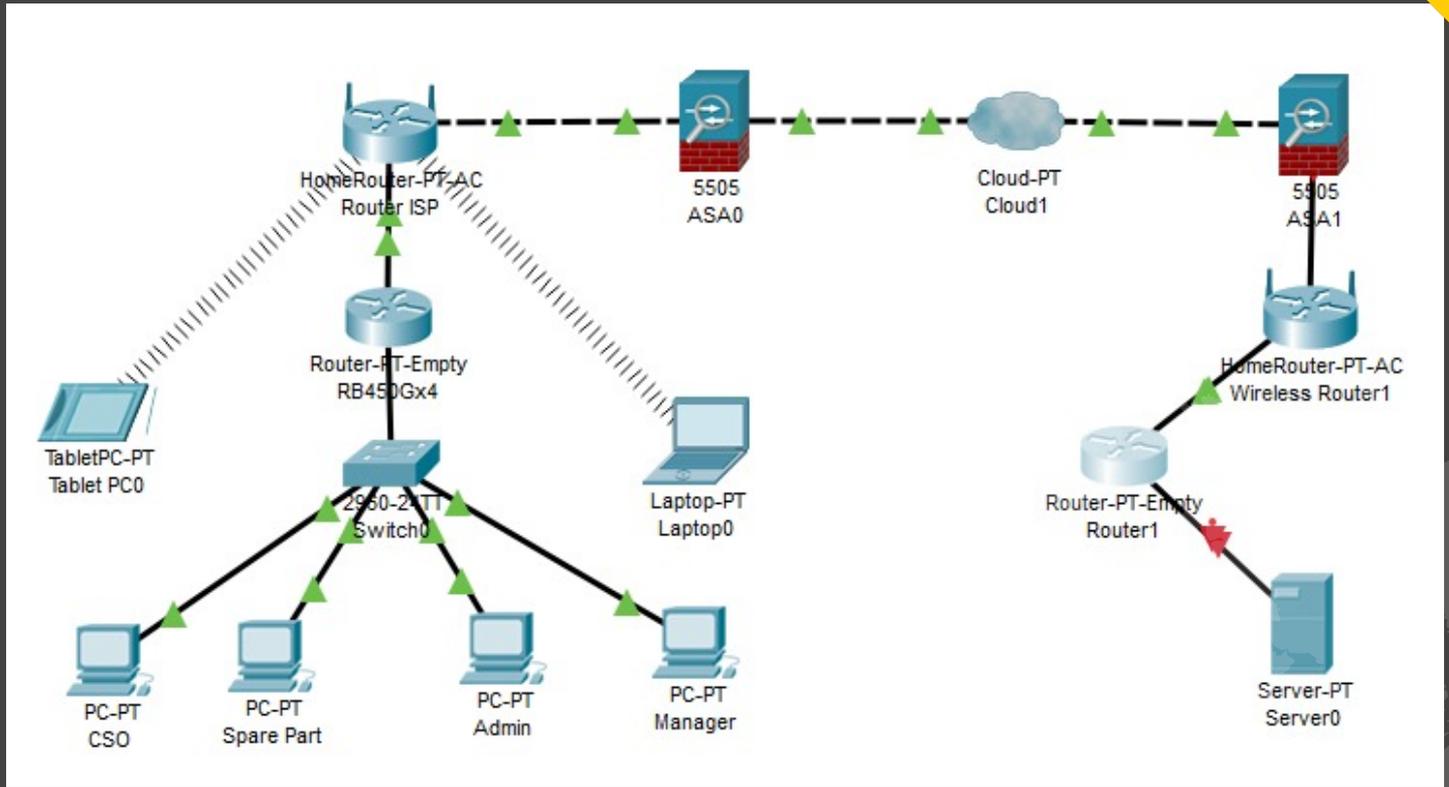
TERIMA KASIH



# PT. 2MB INDONESIA



Jl. Ahmad Yani No.500 Palembang  
Sumatera Selatan  
INDONESIA



# Alat dan Bahan



Server

DELL PowerEdge R740

DELL PowerEdge R740

(Dual Xeon Silver 4110, 32GB, 2x 1.92TB SSD)

2 x Xeon Silver 4110

2 x 16GB 2666MT/s Dual Ranked x8 Data Width RDIMMs

2 x 1.92TB SAS 2.5 Inch SSD

1 x Broadcom 5720 QP 1Gb Network Daughter Card

2 x 750W

Rackmount (2U)

# Alat dan Bahan

Router

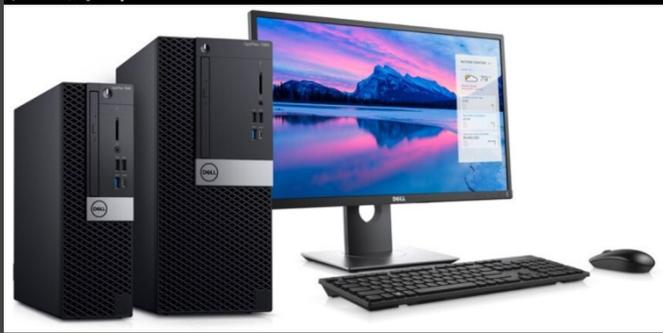
Routerboard RB450Gx4



Routerboard RB450Gx4

(716MHz Quad Core CPU, 1 GB DDR RAM, 512MB NAND Storage)

# Alat dan Bahan



Personal Komputer (PC)

DELL OptiPlex 7060

Intel Core™ i5-8500 (6 Cores/9MB/6T/up to 4.1GHz/65W); supports Windows 10/Linux

Windows 10 Pro 64bit English, French, Spanish

Intel® Integrated Graphics

8GB 2X4GB DDR4 2666MHz UDIMM Non-ECC

3.5" 500GB 7200rpm SATA Hard Disk Drive

## Rencana Anggaran Biaya

No	Komponen	Merk	Quantity	Harga Satuan	Jumlah
1	Server	DELL PowerEdge R740 (Dual Xeon Silver 4110, 32GB, 2x 1.92TB SSD)	1 pc	226.000.000	Rp 226.000.000
2	Router	Routerboard RB450Gx4 (716MHz Quad Core CPU, 1 GB DDR RAM, 512MB NAND Storage)	1 pc	1.617.000	Rp 1.617.000
3	Personal Komputer (PC)	DELL OptiPlex 7060	4 pcs	10.486.000	Rp 41.944.000
4	Kabel LAN UTP	CAT 5e	1 roll	1.325.000	Rp 1.325.000
<b>Total Harga</b>					<b>Rp 270.886.000</b>





Thank you

Feel Our Services

## TUGAS

Nama mahasiswa:

1. DEDE TRISEPTIAWAN
2. M DANIAL SENTOSA

Permasalahan : ketika user menggunakan internet di wilayah tertentu, kecepatan internet menurun, tidak seperti yang dijanjikan para provider, dikarenakan setiap provider membatasi pengguna pada suatu daerah, misal suatu daerah dibatasi 1 GB/s dan penggunanya ada 10 orang, maka 1 GB/s akan dibagikan kepada 10 orang, maka setiap orang mendapatkan kecepatan 100 Mb/s sehingga semakin banyak pengguna maka internet akan semakin lambat bahkan terjadi packet loss

Solusi : pada jurnal dengan judul *The TIME-WAIT state in TCP and Its Effect on Busy Servers* oleh Theodore Faber menjelaskan dengan menggunakan sistem TIME-WAIT

Jurnal ini telah menerapkan dan menguji solusi TCP yang lebih sederhana dan solusi HTTP untuk menunjukkan pergeseran waktu TIME-WAIT dari klien ke server. Kami telah menyajikan bukti eksperimental bahwa loading TIME-WAIT dapat mempengaruhi kinerja server di bawah SunOS 4.1.3. Dalam kondisi ini, throughput dapat dikurangi hingga 50%.

Dengan menggunakan system ini dan di kombinasikan dengan koneksi TCP persistent, hal ini menggunakan lebih sedikit memori dari pada Sun OS 4.1.3 yang tidak dimodifikasi menggunakan koneksi pertant, untuk beban klien tertentu. System yang digunakan berkoneksi terus-menerus, masalah pemuatan memori secara langsung. Ini dapat mengurangi biaya penyebaran server, yang dapat menjadi sangat penting server tertanam/baterai

Menurut theodore faber, pendistribusian TIME-WAIT TCBS ke client dengan memanfaatkan angka pertumbuhan mereka untuk mengurangi memory load pada server. Kelebihannya yaitu mendukung koneksi yang persistant dan berbagi manfaat lainnya seperti menghindari extra TCP 3 handshakes.

Pada jurnal ini percaya bahwa TCP pada akhirnya harus dimodifikasi untuk mendukung TIME-WAIT negotiation. Upgrade yang akan datang dari IP versi 4 ke versi 6 juga merupakan peluang untuk meninjau kembali implementasi dan desain TCP. Ini akan menjadi waktu yang tepat untuk memasukkan TIME-WAIT state negotiation.

Algoritma TIME-WAIT state negotiation yaitu mengizinkan busy server untuk menerima koneksi dari klien yang memiliki TIME-WAIT overhead. Penggunaan algoritma ini tidak harus mengubah penggunaan koneksi dekat dengan protokol mereka dan tidak mengalami perubahan performa yang terkait dengan pendistribusian TIME-WAIT TCBS.

Selain memodifikasi TCP, menurut theodore faber, memodifikasi HTTP juga diperlukan karena itu merupakan salah satu komponen terbesar pada internet traffic. Untuk mensupport koneksi yang persistent, indikasi end-of-connection dan end of transaction harus dipisahkan. Pemisahan ini mengizinkan kita untuk memodifikasi HTTP untuk mengizinkan klien mengaktifkan kembali close connections dan menahan TIME-WAIT state.

**TUGAS COMPUTER NETWORK AND BUSINESS INFORMATION**  
**DOSEN : Dr.Edi Surya Negara, M.Kom.**

**Nama : Muhammad Hadrifiansyah**

NIM : 192420010

Kelas : Reguler A R1

**Nama : Eko Purwanto**

NIM : 19240019

Kelas : Reguler A R1

Kasus :

Transmisi data sebesar 1 GB/s tidak merata kepada penerima paket, user menerima masing-masing 100Mbps sedangkan banyak user sebesar 50 user. Bagaimana cara untuk stabil pengiriman data sedangkan sering terjadi gagal pengiriman data. Di waktu tertentu trafik menunjukkan data terlalu lambatnya dalam transmisi internet.atau sering di sebut Backbone atau server terlalu sibuk.

Solusi :

1. Dalam membangun sebuah jaringan dan server harus melakukan beberapa tahapan yang harus di perhatikan yaitu :
  1. Pemilihan ISP Dedicated dan bandwidth yang stabil sesuai dengan kebutuhan perusahaan.
  2. Membangun perangkat Server sesuai dengan standard internasional, serta pemilihan penyimpanan memori yang besar untuk jangka panjang.
  3. Pembagian Paket internet menjadi user priority and user guest.
  
2. Dalam kasus yang telah terjadi ada pun yang harus di perhatikan dan menindak lanjuti :
  1. Melakukan Perubahan Paket Data atau ISP pada Server dan Maintenance Server karena dalam Off-loading memori, semakin banyak klien mengurangi memori server dan kecepatan koneksi pada server. Tumbuhnya kecepatan dan lambatnya akses karena peningkatan jumlah klien dan bandwidth yang tersedia di jaringan. Apabila semakin banyak user koneksi ke pada server, maka server akan semakin sibuk.
  2. Pembagian Paket internet menjadi user priority and user guest.  
Yang tujuannya untuk membagi internet prioritas dan internet yang Non-Prioritas.
  3. Data yang tidak terkirim dan Hilang akan tersimpang didalam Cookies Memori.
  4. Dalam Pembagian Kuota Internet masing-masing user bisa di atur melalui server.  
Dengan cara mendaftarkan IP Perangkat tersebut dan memmanagement sesuai kebutuhan.

Nama : Hasirul Qodar dan Ryan Andrian

Kelas : MTI 21 A R1

UTS : Computer Network and Communications

Pertumbuhan aktifitas di internet yang cepat mempengaruhi lalu lintas internet sehingga menyebabkan terjadinya kemacetan dan pengiriman data yang tidak lengkap. Dari permasalahan tersebut dapat diantisipasi dengan beberapa alternatif solusi diantaranya, yaitu sebagai berikut :

1. Modifikasi protokol tcp dalam pengiriman data yang sebelumnya menggunakan koneksi single di rubah menjadi koneksi paralel
2. Transfer terpisah berdasarkan prioritas jenis data misalnya voice, financial transaction dan gambar
3. Management dalam merespon request dari client di server secara bertahap untuk mengurangi beban memory di server dan merperkecil persentasi data loss



# Paparan Study Case Network Topology

Palembang, 20 Desember 2019

By. Hendra Yada Putra & Ichan Syaputra (MTI-Reg B Angk 21)



# Agenda

**Pendahuluan**

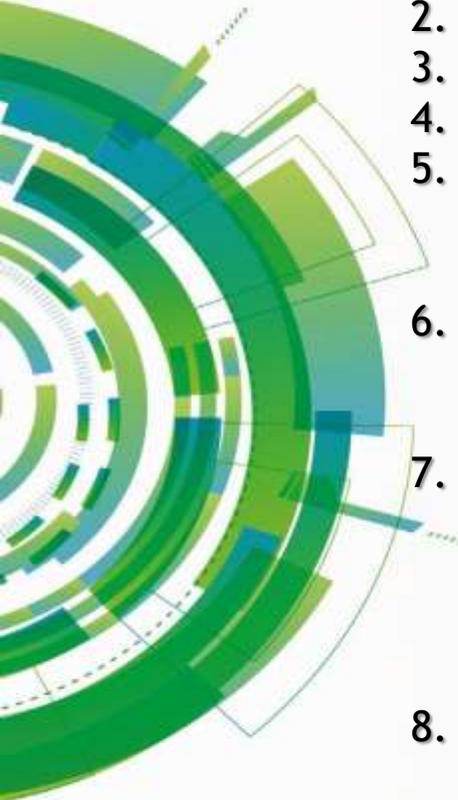
**Topologi existing**

**Assessment**

**Rekomendasi**

**Budget**

# Skenario

- 
1. Perusahaan Jasa Pengantaran Barang sejak 2012
  2. Memiliki 1 Kantor pusat dan 12 Kantor cabang
  3. Jumlah pegawai 300 orang
  4. Bisnis menggunakan Channel Kantor dan internet
  5. Mengadopsi teknologi Digital kedalam bisnis perusahaan pada tahun 2016, transaksi perusahaan meningkat pesat hampir 500%.
  6. Peningkatan transaksi didominasi pada Channel Internet, dimana transaksi melalui kantor menurun tidak lebih dari 20% terhadap total transaksi.
  7. Peningkatan transaksi diiringi dengan peningkatan keluhan dari customer dan internal staff.
    1. aplikasi sering dikeluhkan lambat,
    2. SLA Layanan setiap bulannya tidak tercapai
    3. Virus dan Malware
  8. Telah melakukan Tuning terhadap Aplikasi, Server, dan internet, namun keluhan masih tinggi.
  9. Dari sisi Compliant. perusahaan juga diharuskan untuk mengikuti standar ISO27001.
  10. Manajemen meminta untuk diadakan assessment dari sisi Jaringan

## PENDAHULUAN

### Latar Belakang

“Pemenuhan Tugas Mata Kuliah Computer Network and Communication”

### Tujuan

“Memaparkan Rekomendasi Topologi Network beserta Perangkat pendukungnya untuk Perusahaan kelas Menengah Keatas, dengan Konsentrasi Peningkatan Availbaility, Performance dan keamamanan”

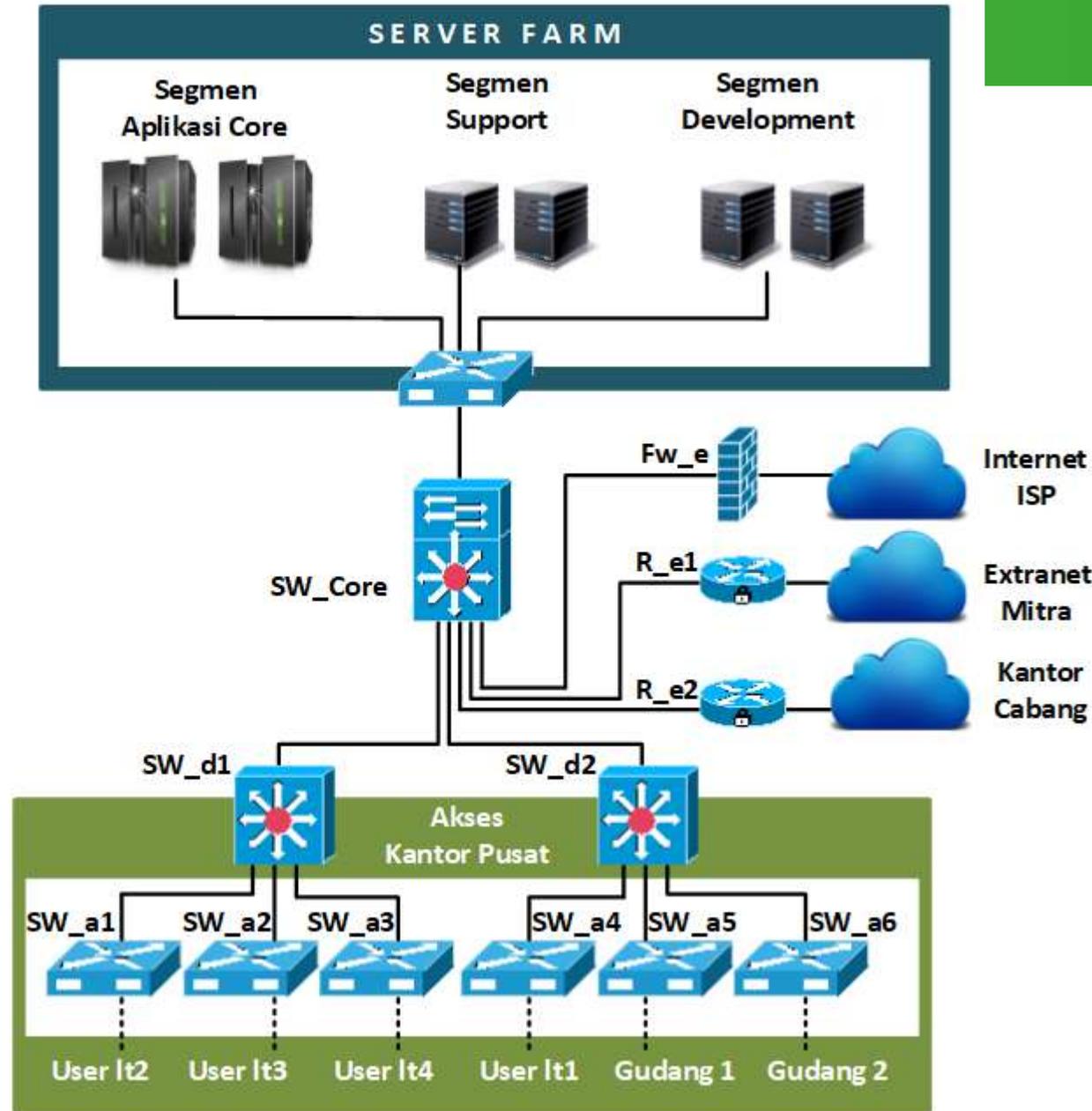
# TOPOLOGI

NETWORK  
DISTRIBUSI

NETWORK  
CORE

NETWORK  
DISTRIBUSI

NETWORK  
ACCESS



Perangkat terdiri dari:

1. 1 Switch Core
2. 7 Switch Access
3. 2 Switch Distribusi
4. 2 Router
5. 1 Firewall Edge



# Pengumpulan data

## ASSESSMENT

- Load trafik internet selalu mencapai level 80-90% di jam operasional kantor.
- Load trafik pada Switch Core yang menuju Server farm tidak lebih dari 70% dengan teknologi ethernet yang digunakan 1GbE
- Peak Cpu usage pada Switch Core 70%
- Pemisahan fungsi pada server Core untuk aplikasi, Database dan Web dengan SSL Sertifikat ditanam pada Web Server
- Serverfarm berada dalam 1 segemen untuk semua server: Server Core, Server Mobile API, Server Gateway, Development, AD, DNS-NTP, mail, Servicedesk, Antivirus, Monitoring, HR.
- Terdapat Cold Backup untuk server Core.
- Firewall menggunakan teknologi sebelum Next Generation
- Terdapat Perangkat yang telah EOL yaitu Swicth Core, Firewall, Switch Distribusi, Router, dan beberapa Switch Access



# Hasil analisa

## ASSESSMENT

- 
- Harus dibedakan akses Internet untuk akses transaksi dan User
  - Segera dilakukan penggantian perangkat EOL (risiko tinggi)
  - Dibutuhkan Penambahan Backup yang tergolong Critical poin
  - Perlu Menambahkan Link Internet baru dengan mengimplementasikan perangkat Link Controller untuk Redudansi Internet dan penempatan SSL Sertifikat
  - Perlu Menempatkan Server yang membutuhkan koneksi internet ke area DMZ sesuai area jaringan internet
    - Server support (Antivirus, DNS-NTP, Mail, Server Gateway)
    - Server Web Core dan Server Mobile API
  - Pemisahan Segement pada Serverfarm dan menambahkan NGT Firewall untuk proteksi masing-masing segmen
  - Dibutuhkan pembagi Beban untuk WEB Core Backup
  - Perlu penggantian firewall Internet dengan NGT Firewall.

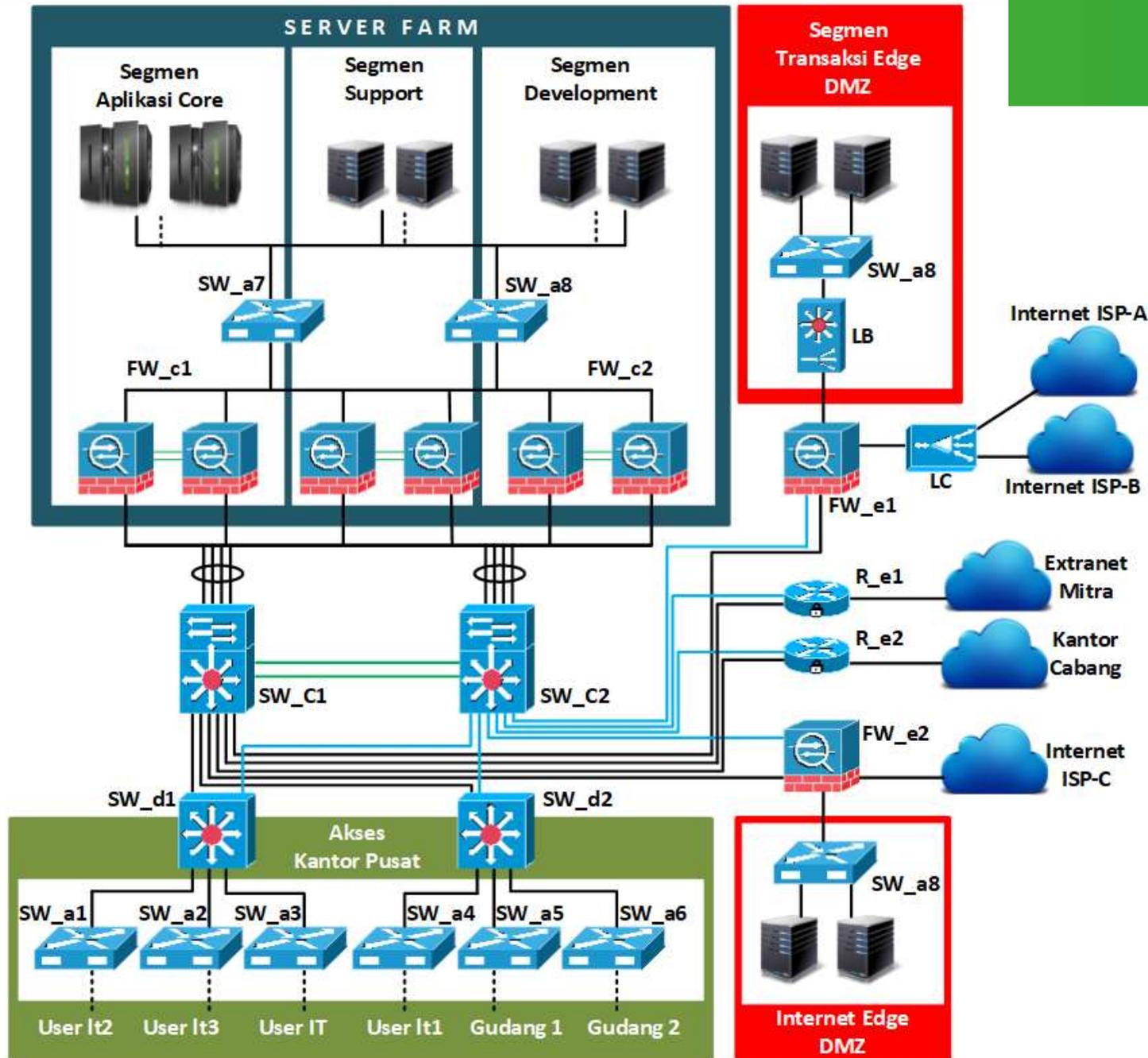
# REKOMENDASI

NETWORK  
DISTRIBUSI

NETWORK  
CORE

NETWORK  
DISTRIBUSI

NETWORK  
ACCESS



Perangkat terdiri dari:

1. 2 Switch Core
2. 8 Switch Access
3. 4 Switch Distribusi
4. 2 Router
5. 1 Load Balancer
6. 1 Load Controller
7. 2 Firewall Core
8. 2 Firewall Edge



# Segmentasi

DETIL

## **Segmen Aplikasi Core**

**Server Aplikasi Core Bisnis**

**Server Database**

**Server Interface**

## **Segmen Support**

**Server Active Directory**

**Server DNS internal**

**Server Aplikasi HR**

**Server Aplikasi Helpdesk**

**Server Email**

**Server Monitoring & Manajement**

## **Segmen Development**

**Server Development Core**

**Server Development Interface**

**Server Development Support**

## **Segemen Transaksi Edge**

**Server Web Korporasi**

**Server API Mobile**

**Server Web Transaksi**

## **Segemen Internet Edge**

**Server DNS Forwader**

**Server Antivirus**

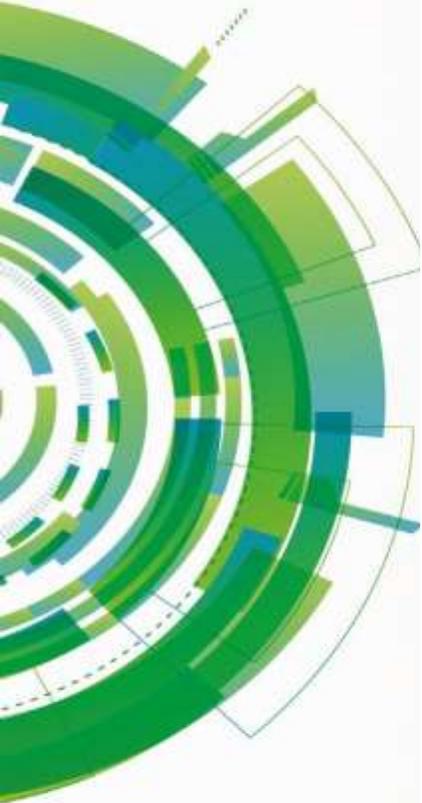
**Server NTP**

**Akses Internet User**



- **Segmentasi lebih dalam untuk area server Farm sehingga keamanan lebih terjamin**
- **Kualitas Layanan dapat ditinggkatkan**
- **Pemisahan Jalur internet user dengan internet untuk transaksi**
- **Pengamanan berlapis dengan pemenuhan standard iso27001**
- **Availability layanan dengan link controller dan link balancer**
- **Dengan memisahkan akses SSL sertifikat pada server dapat meningkatkan performace Server**

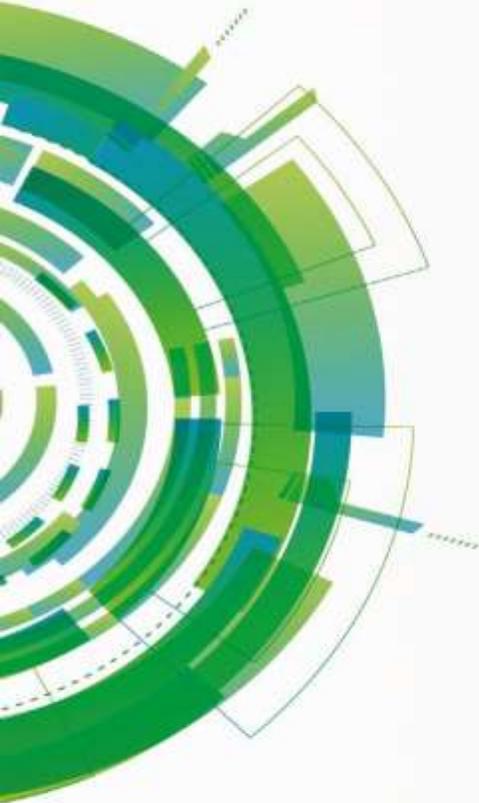




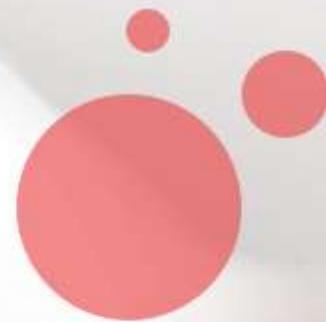
No	Item	Jumlah	Harga Satuan	Harga
1	Switch Core Cisco Catalyst 9300 C9300-48P-E	2	250.000.000	500.000.000
2	Switch Distribusi Cisco Catalyst 3850 Ws-C3850-24t-S	4	80.000.000	320.000.000
3	Switch Access Cisco Catalyst 2960-x (WS-C2960X-48FPD-L)	4	50.000.000	200.000.000
4	Router Cisco Cisco ASR 1002-X	2	70.000.000	140.000.000
5	Firewall Core Fortigate 1000D	1	260.000.000	260.000.000
6	Firewall Edge Fortigate 100e	1	80.000.000	80.000.000
7	Load Balancer A10 Thunder 840 ADC	1	150.000.000	150.000.000
8	Link Controller f5 2200 LTM	1	250.000.000	250.000.000
9	Cabling,Soket, dll (Panduit)	1	20.000.000	20.000.000
10	Implementasi	20	5.000.000	100.000.000
	TOTAL			2.020.000.000

# Q & A





Thank you



**TUGAS**  
**COMPUTER NETWORK AND BUSINESS INFORMATION**



**Nama : Hendri Donan**

**NIM : 192420039**

**Kelas : AR2**

**Program Studi S2 Teknik Informatika**

# TUGAS

Problem

Why Internet Slows Down when it's Busy?

Pembahasan.

**TCP/IP (*Transmission Control Protocol/Internet Protocol*)**

## **A. Pendahuluan**

Jaringan komputer sudah menjadi sesuatu kebutuhan yang sangat penting, dimana komputer tidak lagi berdiri sendiri tetapi saling terhubung satu sama lain. Dan seiring dengan makin berkembangnya jumlah komputer yang terdapat pada suatu jaringan, maka makin bertambah pula tingkat kesulitan untuk mengelola jaringan tersebut.

Sistem jaringan komunikasi data yang sangat populer dan tersebar di seluruh dunia adalah jaringan internet. Jaringan ini menghubungkan antara satu host komputer dengan komputer lainnya yang tersebar di seluruh dunia dengan menggunakan protokol IP atau disebut juga dengan internet protokol.

Pada awalnya komputer dihubungkan dengan komputer lain dengan tujuan pertukaran data secara sederhana. Teknologi yang ada yaitu *Serial Interface* (RS232) dapat digunakan untuk menghubungkan dua komputer secara langsung (direct) atau melalui modem dan untuk selanjutnya dihubungkan dengan fasilitas seperti telepon atau lainnya. Bentuk hubungan ini disebut dengan *Wide Area Network* (WAN). Munculnya teknologi Local Area Network (LAN) yang menggunakan ethernet atau lainnya pada tahun 1970-an memungkinkan komputer untuk saling berhubungan dengan kecepatan 10.000bps (10 Mbps) memberikan inovasi baru. Berbeda dengan WAN, umumnya komputer LAN terletak satu dengan yang lainnya dalam radius yang tidak terlalu jauh (dalam satu gedung atau kompleks).

Jaringan komputer merupakan teknologi yang bermanfaat, efektif, dan efisien dalam menyelesaikan berbagai pekerjaan, diantaranya adalah untuk pekerjaan-pekerjaan yang berkaitan dengan pengiriman data atau transmisi data. Contoh umum dari aktifitas ini adalah aktifitas para pengajar pada suatu perguruan tinggi. Bahan atau materi perkuliahan umumnya diambil dari internet, disamping beberapa buku referensi. Proses ini biasanya memiliki intensitas yang cukup tinggi pada aktifitas pencarian informasi terkini dan download data. Pada kondisi ini, keberhasilan mengakses informasi, sangat bergantung kepada kecepatan akses transmisi data, dimana secara langsung juga sangat berkaitan erat dengan sistem jaringan komputer yang terdapat pada lembaga tersebut.

## **B. TCP/IP (*Transmission Control Protocol/Internet Protocol*)**

Saat Vint Cerf dan Bob Kahn pertama menulis spesifikasi atas protokol ini tahun 1973, internet masih merupakan media yang dilarang untuk keperluan komersil. Di bulan May 1974, IEEE mempublikasikan makalah berjudul "*Protokol sebagai paket interkoneksi jaringan.*"<sup>1</sup> Sebagai penulis makalah tersebut, Cerf dan Kahn menjabarkan sebuah protokol interkoneksi untuk berbagi layanan dengan memanfaatkan pertukaran-paket antara node jaringan yang terhubung. Komponen pengontrol utama atas model ini disebut sebagai "*Program pengontrol transmisi*" yang menggabungkan baik tautan koneksi terorientasi dan layanan datagram antar *host*.

Program pengontrol transmisi monolitik ini kemudian dipecah ke dalam arsitektur modular yang terdiri atas *Protokol pengontrol transmisi* pada lapisan koneksi terorientasi dan *Protokol internet* pada lapisan datagram. Model inilah yang kemudian dikenal dengan istilah *TCP/IP*, meski secara formal istilah yang digunakan adalah *Paket protokol internet*. *TCP/IP (Transmission Control Protocol/Internet Protocol)* memungkinkan hubungan virtual antar komputer, dimana dua komputer atau lebih akan dapat saling berhubungan untuk pertukaran data serta layanan aplikasi jaringan lainnya.

Tujuan daripada desain TCP/IP adalah sebagai berikut :

1. Standart protokol yang open. Artinya spesifikasi dapat diperoleh dengan bebas dan dikembangkan sesuai dengan hardware yang dimiliki. Dengan

demikian TCP/IP dapat diimplementasikan pada *platform hardware* yang beragam.

2. Tidak tergantung pada jaringan fisik hardware. TCP/IP dapat diintegrasikan pada jaringan fisik yang bermacam-macam melalui *ethernet, token ring, dial up* (telepon) RS232 dan media transmisi lainnya.
3. Skema address yang luas. Skema address internet memungkinkan komputer mempunyai identitas tunggal (IP-address), sehingga walaupun mempunyai jangkauan *international (wordwide)*, komputer manapun dapat dicapai dengan mudah karena mempunyai identitas yang jelas.
4. Standar aplikasi. Utilitas standar yang akan memudahkan pemakaiannya dalam melakukan *file transfer, remote login* dan *remote execution*.

TCP memiliki karakteristik sebagai berikut:

- Berorientasi sambungan (*connection-oriented*): Sebelum data dapat ditransmisikan antara dua host, dua proses yang berjalan pada lapisan aplikasi harus melakukan negosiasi untuk membuat sesi koneksi terlebih dahulu. Koneksi TCP ditutup dengan menggunakan proses terminasi koneksi TCP (TCP connection termination).
- *Full-duplex*: Untuk setiap host TCP, koneksi yang terjadi antara dua host terdiri atas dua buah jalur, yakni jalur keluar dan jalur masuk. Dengan menggunakan teknologi lapisan yang lebih rendah yang mendukung full-duplex, maka data pun dapat secara simultan diterima dan dikirim. Header TCP berisi nomor urut (TCP sequence number) dari data yang ditransmisikan dan sebuah acknowledgment dari data yang masuk.
- Dapat diandalkan (*reliable*): Data yang dikirimkan ke sebuah koneksi TCP akan diurutkan dengan sebuah nomor urut paket dan akan mengharapkan paket positive acknowledgment dari penerima. Jika tidak ada paket Acknowledgment dari penerima, maka segmen TCP (protocol data unit dalam protokol TCP) akan ditransmisikan ulang. Pada pihak penerima, segmen-segmen duplikat akan diabaikan dan segmen-segmen yang datang tidak sesuai dengan urutannya akan diletakkan di belakang untuk mengurutkan segmen-segmen TCP. Untuk menjamin integritas setiap segmen TCP, TCP mengimplementasikan penghitungan TCP Checksum.

- *Byte stream*: TCP melihat data yang dikirimkan dan diterima melalui dua jalur masuk dan jalur keluar TCP sebagai sebuah *byte stream* yang berdekatan (kontigu). Nomor urut TCP dan nomor acknowledgment dalam setiap header TCP didefinisikan juga dalam bentuk byte. Meski demikian, TCP tidak mengetahui batasan pesan-pesan di dalam byte stream TCP tersebut. Untuk melakukannya, hal ini diserahkan kepada protokol lapisan aplikasi (dalam DARPA Reference Model), yang harus menerjemahkan byte stream TCP ke dalam "bahasa" yang ia pahami.
- Memiliki layanan *flow control*: Untuk mencegah data terlalu banyak dikirimkan pada satu waktu, yang akhirnya membuat "macet" jaringan internetwork IP, TCP mengimplementasikan layanan *flow control* yang dimiliki oleh pihak pengirim yang secara terus menerus memantau dan membatasi jumlah data yang dikirimkan pada satu waktu. Untuk mencegah pihak penerima untuk memperoleh data yang tidak dapat disangganya (*buffer*), TCP juga mengimplementasikan *flow control* dalam pihak penerima, yang mengindikasikan jumlah *buffer* yang masih tersedia dalam pihak penerima.
- Melakukan segmentasi terhadap data yang datang dari lapisan aplikasi (dalam DARPA Reference Model)
- Mengirimkan paket secara "one-to-one": hal ini karena memang TCP harus membuat sebuah sirkuit logis antara dua buah protokol lapisan aplikasi agar saling dapat berkomunikasi. TCP tidak menyediakan layanan pengiriman data secara *one-to-many*.

TCP umumnya digunakan ketika protokol lapisan aplikasi membutuhkan layanan transfer data yang bersifat andal, yang layanan tersebut tidak dimiliki oleh protokol lapisan aplikasi tersebut. Contoh dari protokol yang menggunakan TCP adalah HTTP dan FTP.

### C. Transmission Control Protocol (TCP)

TCP merupakan protokol yang terletak di layer transport. Protokol ini menyediakan service yang dikenal sebagai *connection oriented, reliable, byte stear service*. *Connection oriented* berarti sebelum melakukan pertukaran data, dua aplikasi pengguna TCP harus melakukan pembentukan hubungan (handshake) terlebih dahulu. *Reliable* berarti TCP menerapkan proses deteksi kesalahan paket

dan retransmisi. *Byte Stear Service* berarti paket dikirimkan dan sampai ke tujuan secara berurutan.

Untuk menjamin kehandalan, TCP melakukan hal-hal berikut ini :

1. Data yang diterima oleh aplikasi dipecah menjadi segmen-segmen yang besarnya menurut TCP paling sesuai untuk mengirimkan data.
2. Ketika TCP menerima data dari mitranya, TCP mengirimkan acknowledge (pemberitahuan bahwa data telah diterima).
3. Ketika TCP mengirimkan sebuah data, TCP mengaktifkan pewaktu (software time) yang akan menunggu acknowledge dari penerima segmen data tersebut. Jika sampai waktu yang ditentukan acknowledge tidak diterima, data tersebut akan dikembalikan ke TCP.
4. Sebelum segmen data dikirim, TCP melakukan perhitungan checksum pada header dan datanya. Hal ini berbeda pada protokol IP yang hanya melakukan perhitungan pada headernya saja. Jika segmen yang diterima memiliki checksum yang tidak valid, TCP akan membuang segmen ini dan berharap sisi pengirim akan melakukan retransmisi. Karena segmen TCP dikirim menggunakan IP dan datagram IP dapat sampai ke tujuan dalam keadaan tidak berurutan, segmen TCP yang dikirimpun dapat mengalami hal yang sama. Karenanya sisi penerima paket TCP harus mampu melakukan pengurutan kembali segmen TCP yang ia terima (resequencing) dan memberikan data dengan urutan yang benar kepada aplikasi penggunaannya.
5. Karena paket IP dapat terduplikasi di perjalanan, penerima TCP harus membuang data tersebut.
6. Untuk mencegah agar server yang cepat tidak membanjiri server yang lambat, TCP melakukan proses flow control. Setiap koneksi TCP memiliki buffer dengan ukuran yang terbatas. Sisi penerima TCP hanya memperbolehkan sisi pengirim data sebesar buffer yang ia miliki.

### **TCP Three-way handshake**



Proses pembuatan koneksi (TCP Three way handshake)

Proses pembuatan koneksi TCP disebut juga dengan "Three-way Handshake". Tujuan metode ini adalah agar dapat melakukan sinkronisasi terhadap nomor urut dan nomor acknowledgement yang dikirimkan oleh kedua pihak dan saling bertukar ukuran TCP Window. Prosesnya dapat digambarkan sebagai berikut:

- *Host* pertama (yang ingin membuat koneksi) akan mengirimkan sebuah segmen TCP dengan flag SYN diaktifkan kepada *host* kedua (yang hendak diajak untuk berkomunikasi).
- *Host* kedua akan meresponsnya dengan mengirimkan segmen dengan acknowledgment dan juga SYN kepada *host* pertama.
- *Host* pertama selanjutnya akan mulai saling bertukar data dengan *host* kedua.

TCP menggunakan proses jabat tangan yang sama untuk mengakhiri koneksi yang dibuat. Hal ini menjamin dua *host* yang sedang terkoneksi tersebut telah menyelesaikan proses transmisi data dan semua data yang ditransmisikan telah diterima dengan baik. Itulah sebabnya, mengapa TCP disebut dengan koneksi yang *reliable*.

### Kelebihan TCP

- Mendukung mekanisme *flow control*
- Dapat memecah data menjadi lebih kecil dan menyusunnya kembali
- Koneksinya Reliable karena melakukan handshaking terlebih dahulu

### Kekurangan TCP

- Jika mengirimkan data yang kecil akan kerepotan karena harus handshaking dulu, sehingga lebih lambat daripada **UDP**
- Tidak bisa broadcast, soalnya **TCP** ini sifatnya one to one

#### Daftar Pustaka

- Yunus, M dan As'ad. (2012). *Protokol TCP/IP sebagai sarana dalam proses transfer data*. Jurnal teknologi informasi: teori, konsep, dan implementasi vol. 3 no. 1
- Purbo, Onno W, (1998), *TCP/IP Standard Desain dan Implementasi*, PT. Elex Media Komputindo, Jakarta
- Suryadi, MT, (2000) *TCP/IP dan Internet sebagai Jaringan Komunikasi Data*, Andi Offset, Yogyakarta.
- Adipranata, R (2002) *Implementasi Protokol Tcp/Ip Untuk Pengendalian Komputer Jarak Jauh*. Fakultas Teknologi Industri, Jurusan Teknik Informatika – Universitas Kristen Petra e-mail : [rudya@petra.ac.id](mailto:rudya@petra.ac.id). Jurnal Informatika Vol. 3, No. 1, Mei 2002: 34 - 41
- Harahap, E. dkk (2007). Faktor Penghambat Transmisi Data Pada Jaringan Jurnal Matematika UNISBA Vol. 7 No. 1, November 2007 <http://ejournal.unisba.ac.id> ISSN: 1412-5056 Received: 07/07/2007 Accepted: 15/09/2007 Published online: 01/11/2007

NAMA : **RIA APRINDA ( 192420022)**

**ISTIANA RUSWITA ( 192420032)**

KELAS : REGULER A R1

DOSEN : Dr. EDI SURYA NEGARA, M.KOM

### **MASALAH :**

Ketika banyak klien yang mengakses ke sebuah server maka akan mengakibatkan permasalahan seperti koneksi melambat atau crash disebabkan data loss. Data tersebut harus di re-transmitted yang menyebabkan user experience menjadi jelek karena seringnya time out. Sehingga pokok permasalahan dari video tersebut adalah: Mengapa jaringan internet menjadi lambat ketika sibuk?

### **SOLUSI:**

#### **Berdasarkan jurnal The TIME-WAIT state in TCP and Its Effect on Busy Servers:**

Banyaknya jumlah akses klien menyebabkan kebutuhan memory pada server meningkat. Hal ini mengakibatkan waktu tunggu memory atau yang disebut dengan TIME-WAIT loading. Satu-satunya cara untuk menjaga kebutuhan memory TIME WAIT server adalah dengan memindahkan TIME WAIT TCBS kepada klien.

Ada 3 metode yang dapat digunakan:

1. TCP with TIME – WAIT Negotiation  
Dimana endpoint memegang TIME WAIT saat koneksi dikembangkan dan mengusulkan sebuah protokol negosiasi. Sistem ini memperpanjang fungsi TCP untuk mengalokasikan TIME WAIT TCBS ke akhir koneksi yang benar.
2. TCP with Clients  
Daalam kondisi ini, bandwidth dapat dikurangi hingga 50%.
3. Client\_Close HTTP Extension  
Sistem ini mendukung rata-rata koneksi yang lebih tinggi daripada sistem yang tidak dimodifikasi.

#### **Berdasarkan jurnal TCP Behaviour of a Busy Internet Server : Analysis and Improvements**

Untuk mengurangi terjadinya time out yaitu menggunakan skema peningkatan pemulihan kerugian (data yg hilang/koneksi terputus) yang memungkinkan TCP sender utk mengumpulkan koneksi yang terbagi-bagi (koneksi yang dilakukan bersamaan) menjadi 1 koneksi.

Ada 2 metode yang dapat digunakan :

1. Ack Bandwidth Ratio

Menghitung ack bandwidth untuk TCP windows pada semua koneksi pada history.

2. Dynamic Computing of Data and Ack Bandwidth

Data bandwidth dikomputasi dengan membagi jumlah data yang dikirim pada waktu mengirim data pada window

## TUGAS UTS

MATA KULIAH : COMPUTER NETWORK AND BUSINESS INFORMATION

Dosen : Dr. Edi Surya Negara, M.Kom

Nama Kelompok :

1. Istikomah

2. Suriani

Analisa permasalahan dari video yang di tampilkan , Yakni; Permasalahan SIZE. pengiriman data dari klien ke server dengan kapasitas size yg sangat besar. Di server menampung besar size 1 Gb, sedangkan kecepatan pengiriman klien 100 Mb dari 50 klien, sehingga banyak data-data dari klien yang tertinggal atau hilang, rusak banyak tidak sampai.

Cara mengatasi permasalahan tersebut :

1. Mengusulkan mekanisme Pembagian TCP yang ditingkatkan dengan Teknik pembagian segmentasi data . di bagi per segmen sesuaikan kebutuhan klien. Sebagai contoh kapasitas di server 1 Gb di bagi 100mb/klien atau setiap klien 100 mb

Aplikasi tingkat Multiplexing solusi aplikasi-tingkat menghindari penggunaan beberapa sambungan paralel TCP, dan masalah yang dihasilkan, oleh multiplexing beberapa data stream ke sebuah koneksi TCP tunggal. Sejak TCP hanya menyediakan satu, mulus byte-stream abstraksi, solusi tingkat aplikasi ini termasuk skema framing untuk demarkasi aliran data individu. Contoh ini meliputi Persistent-koneksi HTTP (P-HTTP) [13], Session Control Protocol (SCP) [14] dan protokol MUX [8]. penurunan yang signifikan dalam latency dari Web pengaksesan menggunakan P-HTTP dilaporkan dalam Metode 1: Ack Bandwidth Ratio [11] Dalam metode ini, kami menghitung bandwidth ack untuk TCP windows untuk semua koneksi dalam penelusuran. The Jack bandwidth yang sehubungan dengan pengakuan A mulai didefinisikan sebagai jumlah byte yang beredar pada saat A diterima dibagi dengan waktu untuk menerima ack untuk byte luar biasa terakhir. Setiap sampel

bandwidth ack individu kemudian dibandingkan dengan bandwidth median ack untuk semua jendela dalam koneksi. Rasio ini  $(\text{bandwidth ack}) / (\text{bandwidth median ack})$  memungkinkan kita untuk mengukur tingkat kompresi ack. Gambar 2 menunjukkan hasil analisis ini. The x-sumbu adalah (terkuantisasi) ack bandwidth yang rasio, dan sumbu y adalah probabilitas bahwa jendela berikutnya berisi packet loss. Kami melakukan dua pengamatan penting. Pertama, probabilitas kerugian meningkat tajam untuk nilai kecil rasio bandwidth ack. Tapi di luar titik, itu rata karena jaringan telah ditempatkan dalam keadaan macet. Metode 2: Perbandingan Dinamis Data dan Bandwidth Ack Metode kedua mempertimbangkan aliran data dan ack. Pertimbangkan jendela data yang dikirimkan oleh pengirim. Bandwidth data selama jendela dihitung dengan membagi jumlah data yang dikirim pada saat mengirimkan jendela. Demikian pula, bandwidth ack dihitung dengan membagi jumlah data yang diakui oleh perbedaan waktu antara ack pertama dan terakhir

## 2. Pembagian waktu pengaksesan /Pemuatan TIME-WAIT

Transmission Control Protocol (TCP) menyediakan transportasi byte-stream yang andal ke host di Internet. TCP digunakan oleh sebagian besar layanan jaringan yang membutuhkan transportasi yang andal, termasuk Hypertext Transport Protocol (HTTP). Metode TCP untuk mengisolasi koneksi lama dari yang baru menghasilkan akumulasi keadaan pada server yang sibuk yang dapat mengurangi tingkat throughput dan koneksi mereka. Efek pada server HTTP sangat menarik karena membawa sejumlah besar lalu lintas Internet. TCP mengharuskan titik akhir yang menutup koneksi memblokir koneksi lebih lanjut pada pasangan host / port yang sama hingga tidak ada paket dari koneksi yang tersisa di jaringan. Di bawah HTTP, host ini biasanya adalah server. Untuk memblokir sementara koneksi, satu titik akhir menyimpan salinan blok kontrol TCP (TCB) yang menunjukkan bahwa koneksi telah dihentikan baru-baru ini. Koneksi seperti itu ada dalam TIME state WAIT state. Koneksi dalam WAKTU TUNGGU dipindahkan ke TUTUP dan TCB mereka dibuang setelah cukup waktu berlalu bahwa semua paket dari koneksi yang sama telah meninggalkan jaringan. Paket meninggalkan jaringan dengan tiba di salah satu titik akhir dan ditolak, atau tiba dengan bidang TTL (time-to-live) kadaluwarsa di router dan dihapus. Untuk titik akhir yang menjadi target banyak koneksi, ribuan koneksi mungkin dalam WAKTU – MENUNGGU keadaan kapan saja, yang memperkenalkan overhead memori yang signifikan. Kami

menyebut kondisi ini sebagai pemuatan TIME-WAIT. Jika implementasi TCP titik akhir mencari semua TCB saat mengirimkan paket, WAKTU loading MENUNGGU akan secara langsung mempengaruhi kinerjanya. Kehadiran banyak WAKTU IT TUNGGU TCB dapat meningkatkan waktu demultipleks untuk koneksi aktif. Kami hav e dilihat throughput yang penurunan sebesar 50% pada titik akhir dimuat, dan efek pada server komersial telah dicatat di tempat lain . Beberapa implementasi TCP mengatasi masalah demultiplexing tanpa mengatasi beban memori. Desain TCP menempatkan TIME – TUNGGU TCB di titik akhir yang menutup koneksi; keputusan ini bertentangan dengan semantik banyak protokol aplikasi. Fungsi WAKTU –TUNGGU adalah untuk mencegah paket-paket tertunda dari satu koneksi diterima oleh koneksi kemudian. Koneksi konkuren diisolasi oleh mekanisme lain, terutama oleh alamat, port, dan nomor urut.

**Nama : Istiqomah Febrianty**

**Nim : 192420042**

**Kelas : MTIR2**

## **Tugas**

**Pertanyaan : Mengapa Internet melambat ketika jaringan sedang sibuk dari studi kasus diberikan melalui video yang diunggah oleh Dr Richard Mortier**

A. Latar Belakang :

Menurut jurnal Ilmu Komputer Universitas California :

Dalam kasus tersebut dimana pertumbuhan internet semakin pesat dan banyak transisi perpindahan akses peneliti menganalisis dengan dua metode yaitu single Connection Behavior dan Parallel Connection Behavior

1. Single Connection Behavior :

- a. Loss Recovery : Bagaimanakah Pemulihan dan efektifitas pemulihan data dalam menghindari saat timeout
- b. Receiver Bottleneck : Berapa Seringkah Kinerja TCP menerima halaman dengan waktu yang terbatas
- c. Ack compression : Seberapa sering terjadi ack compression saat kehilangan data

2. Parallel Connection Behavior :

- a. Bagaimana klien melihat jumlah koneksi saat server dibuka
- b. Congestion control: bagaimana reaksi koneksi paralel menerima saat kehilangan data yang disebabkan koneksi timeout
- c. Loss behavior : Bagaimana data yang didistribusikan saat koneksi sedang padat

B. Analisa :

Didalam Penelitian tersebut dijelaskan bagaimana cara menganalisa tentang koneksi TCP saat koneksi sedang sibuk dan seberapa baik pemulihan yang

terjadi. Ada beberapa analisa tentang bagaimana cara memperbaiki data yang diterima saat melakukan koneksi jaringan antara lain :

### 1. Single Connection Behavior

Total connections	16501	
With packet re-ordering	9703	6
With receiver window as	2339	14
Total packets	78216	
During slow-start	66620	85
# of slow-start pkts lost	3545	5
During congestion	11595	15
# of congestion	8218	7
Total retransmissions	8571	
Fast retransmissions	3753	44
Slow-start retransmissions	5981	7
Coarse timeouts	4220	49
Avoidable with SACKs	1871	4
Avoidable with enhanced	1042	25

Pada tabel diatas saat melakukan single connection dimana ukuran data yang panggil lebih besar daripada data yang diterima

### 2. Parallel Connection Behavior

Analisa Koneksi yang terbuang saat melakukan koneksi secara bersamaan

### 3. Key Results of Trace Analysis

Dari penelitian yang dikembangkan dimana koneksi tunggal lebih baik dalam pemulihan pengiriman data ketimbang dengan koneksi yang bersamaan

## C. Pembahasan

Ada beberapa metode yang dapat dilakukan antara lain :

### 1. Application-level Multiplexing

Solusi tingkat aplikasi menghindari penggunaan beberapa koneksi TCP paralel, dan masalah yang dihasilkan, dengan multiplexing beberapa

aliran data ke koneksi TCP tunggal. Karena TCP hanya menyediakan abstraksi byte-stream tunggal

Tapi ada beberapa kekurangan antara lain :

- a. Mereka membutuhkan tambahan dan diperlukan server dan klien yang baik.
- b. Mereka tidak mengizinkan multiplexing transfer yang dilakukan oleh lebih dari satu aplikasi.
- c. Multiplexing melalui koneksi TCP tunggal memperkenalkan kopling yang tidak diinginkan antara transfer data yang secara logis independen

## **2. TCP-INT Implementation**

Menggambarkan penerapan kontrol kemacetan terintegrasi dan skema pemulihan kehilangan yang hanya mengubah TCP pada pengirim. Untuk setiap host yang terkait dengan satu mesin, tumpukan TCP / IP membuat struktur (Gambar 9) untuk menyimpan informasi tentang komunikasi apa pun. Struktur baru ini memungkinkan kontrol kemacetan bersama yang diinginkan dan pemulihan kerugian dengan menyediakan satu titik koordinasi untuk semua koneksi ke host tertentu. Struktur host berisi variabel-variabel standar TCP yang terkait dengan pemeliharaan jendela kongesti TCP (cwnd, ssthresh dan count). Struktur ini juga memperkenalkan beberapa variabel baru untuk membantu dalam kontrol kemacetan (ownd, decr\_ts) dan variabel lain untuk mendukung pemulihan kerugian terintegrasi (pkts []). Di subbagian berikut, kami menjelaskan bagaimana berbagai rutinitas TCP menggunakan dan memperbarui informasi baru ini. Rutin pengiriman data baru: Ketika suatu koneksi ingin mengirim suatu paket, ia memeriksa untuk melihat apakah jumlah byte yang sudah ada dalam "pipa" dari pengirim (ownd) lebih besar daripada ukuran "pipa" yang diinginkan (cwnd). Jika tidak, koneksi akan menyiapkan paket yang akan dikirim dengan menambahkan entri pada ekor daftar paket yang beredar. Entri ini berisi ukuran nomor urut dan stempel waktu dari paket yang dikirimkan. Ketika paket dikirim, koneksi menambah kepemilikan

dengan ukuran paket. Kami menggunakan penjadwalan round-robin di seluruh koneksi meskipun itu bukan persyaratan penting. Rutin `recv ack` baru: Ketika `ack` baru tiba, pengirim meningkatkan variabel `cwnd` yang sesuai. Juga pada saat kedatangan `ACK` baru, pengirim menghapus paket dari daftar `pkts []` yang telah mencapai

### **3. Integrated Congestion Control/Loss Recovery**

Motivasi untuk kontrol kemacetan terintegrasi dan pemulihan kerugian adalah untuk memungkinkan aplikasi menggunakan koneksi TCP terpisah untuk setiap transfer (seperti yang mereka lakukan hari ini), tetapi untuk menghindari masalah yang disebutkan dalam Bagian 3.2 dengan membuat modifikasi yang sesuai untuk tumpukan jaringan. Kami membagi fungsionalitas TCP ke dalam dua kategori: yang berkaitan dengan abstraksi TCP byte-stream yang andal, dan yang berkaitan dengan kontrol kemacetan dan pemulihan kerugian berbasis data. Yang terakhir ini dilakukan secara terintegrasi di set koneksi paralel. Kami menyebutnya versi modifikasi TCP `TCP-Int`. Dengan membuka `n` koneksi TCP terpisah untuk transfer, aplikasi memiliki `n-stream` stream yang andal dan independen untuk digunakan. Kontrol aliran untuk setiap koneksi terjadi secara independen dari yang lain, sehingga pengiriman data ke aplikasi penerima juga terjadi secara independen untuk setiap koneksi. Pada saat yang sama, kontrol kemacetan terintegrasi di seluruh koneksi TCP. Ada satu jendela kemacetan untuk set koneksi TCP antara klien dan server yang menentukan jumlah total data luar biasa yang dimiliki oleh set koneksi dalam jaringan. Ketika terjadi kerugian pada salah satu koneksi,

### **4. Simulation Results: One Client Host Case**

Cara ini menjelaskan hasil dari simulasi `ns` yang dirancang untuk menguji kontrol kemacetan dengan pengenalan integer dan pemulihan kehilangan pada koneksi TCP simultan. Tes pertama menggunakan topologi pada Gambar 7. Ukuran buffer router diatur ke 3 paket. Ini cukup kecil untuk memaksa transfer untuk memiliki jendela kemacetan kecil dan sering mengalami kerugian. Sekali lagi, topologi dan parameter dipilih untuk menciptakan kembali situasi yang sering terjadi di jejak kami, dan

tidak meniru jaringan yang sebenarnya. Dalam tes ini, node pengirim melakukan 4 transfer TCP ke penerima. Transfer mulai dari 0, 2, 4 dan 6 detik dan semua berakhir pada 10 detik. Pilihan aktual dari nilai 0, 2, 4, dan 6 tidak penting, hanya saja waktu mulai setiap koneksi terhuyung-huyung dalam waktu. Gambar 10 menunjukkan plot urutan untuk pengujian menggunakan pengirim berbasis SACK. Ini menunjukkan bahwa biasanya hanya satu koneksi berperforma memuaskan pada satu waktu. Misalnya, pada waktu 2 detik, koneksi yang memulai mengalami beberapa kerugian awal dan dipaksa untuk memulihkannya melalui timeout kasar. Faktanya, koneksi ini tidak mengirim sejumlah besar data hingga 4 detik kemudian (pada waktu 6 detik). Selama periode 10 detik, koneksi dimulai pada waktu 2 detik dan waktu 6 detik. Akun untuk fraksi sangat kecil (<10%) dari total byte yang ditransfer. Ketidakadilan dan kinerja yang tidak dapat diprediksi (karena timeout kasar) tidak diinginkan dari sudut pandang aplikasi karena koneksi yang membawa data penting dapat diperlambat sementara yang lain yang membawa data yang kurang penting lebih baik.

##### **5. Simulation Results: Multiple Client Hosts Case**

Simulasi ini menggunakan topologi jaringan dimana saat waktu mulai koneksi di waktu 0, transfer TCP tunggal dimulai setengah dari bandwidth dapat diterima kembali. berfungsi untuk mengimplementasikan fungsi yang lebih kompleks dalam limit bandwidth, dimana penggunaan packet mark nya memiliki fungsi yang lebih baik. Digunakan untuk membatasi satu arah koneksi saja baik itu download maupun upload. Secara umum Queue Tree ini tidak terlihat berbeda dari Simple Queue. Perbedaan yang bisa kita lihat langsung yaitu hanya dari sisi cara pakai atau penggunaannya saja. Dimana Queue Simple secara khusus memang dirancang untuk kemudahan konfigurasi sementara Queue Tree dirancang untuk melaksanakan tugas antrian yang lebih kompleks dan butuh pemahaman yang baik tentang aliran trafik

**Jepri Yandi**  
**MTI 1AR2**  
**NIM**

## TOPIC

Why Internet Slows Down When it's Busy - Computerphile

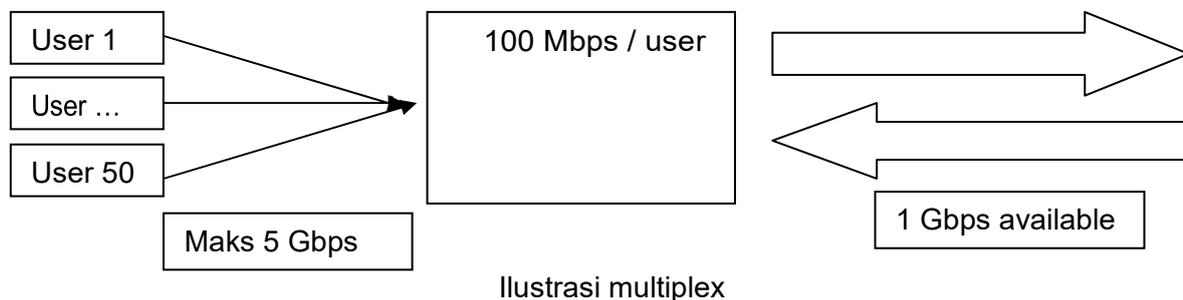
### summary of the video

“ISP / Network provider make assumption about how people will use their services. based on that assumption that allowed them to use multiplex”.( Computerphile )

“*Multiplexing is a method by which multiple analog message signals or digital data streams, coming from different sources are combined into one signal over a shared medium.* “  
(N.baharudin et.al,2013),

contoh skenario *multiplex* :

katakanlah ada 50 *user* yang membeli paket internet berkecepatan 100Mbps. Akan tetapi kenyataannya ISP hanya menyediakan link sebesar 1 gbps yang dishare ke 50 *user* tersebut.



Jika semua *user* menggunakan seluruh *bandwidth* yang dialokasikan kepada mereka bersamaan , maka jumlah beban yang akan diterima adalah 5 Gbps , sedangkan *bandwidth* yang tersedia hanya 1 Gbps, maka dampaknya akan terjadi, *congestion, delay , packet loss* dan sehingga mengakibatkan melambatnya koneksi internet.

Walaupun pada ada prakteknya adalah jarang terjadi kondisi dimana semua *user* memaksimalkan seluruh *bandwidth* yang mereka punya secara bersamaan, pada suatu waktu ada *user* yang menggunakan seluruh *bandwidth* yang dialokasikan kepada mereka, sedangkan beberapa *user* yang tidak menggunakan *bandwidth* yang dialokasikan secara maksimal kepada mereka , sehingga pada umumnya 1 Gbps yang dialokasikan ISP tersebut bisa “mencukupi” semua kebutuhan *user* , metode pembagian ini juga disebut *Statistical multiplexing*.

*Statistical multiplexing is a type of communication link sharing, very similar to dynamic bandwidth allocation (DBA). In statistical multiplexing, a communication channel is divided into an arbitrary number of variable bitrate digital channels or data streams. The link sharing is adapted to the instantaneous traffic demands of the data streams that are transferred over each channel.* (wikipedia.org)

Akan tetapi ada beberapa kondisi / waktu-waktu tertentu ketika orang-orang menggunakan seluruh *bandwidth* yang dialokasikan kepada mereka secara bersamaan, hal ini yang menyebabkan *users* merasakan *delay, slow internet connection or RTO*.

Teknik *multiplexing* telah diterapkan sejak lama pada era telpon rumah, tapi pada era internet, data / informasi yang diproses bergerak dengan cepat sehingga menimbulkan masalah kesulitan melakukan analisis dalam membagi *bandwidth* internet secara adil ke tiap-tiap user.

### Keywords / problems

- Bottleneck
- Congestion
- Delay
- Packet lost
- Mengakibatkan internet koneksi yang lambat

### Solutions

Ketika Koneksi internet semakin cepat, dan makin banyak user yang terhubung ke server tersebut maka server akan semakin sibuk, sehingga akan meningkatkan *Time-Wait loading* (theodore et al).

Ada 3 Solusi yang di berikan theodore et al. yang berfokus pada *Time-wait* dan cara mengurangi *TIME-WAIT loading* .

#### 1. Melakukan modifikasi pada algoritma *Time-Wait* pada saat *three-way handshake*

a. Client includes the *TW-Negotiate* option in the *<SYN>* packet for the connection. *TW Negotiate* contains the IP address of the end that will hold the TCB. The option's presence indicates that the client supports negotiation. Clients must send an IP address, to support the algorithm below for resolving a simultaneous open.

b. Server returns the *<SYN, ACK>* packet with *TW-Negotiate* set to its choice to keep The *TIME-WAIT* state. If it does not support negotiation, it sends no *TW-Negotiate* option.

c. The client decides if the server's choice is acceptable. If so, it acknowledges the *<SYN, ACK>* packet with the same value of *TW-Negotiate*. If not it aborts the connection with an *<RST>* packet. (The connection is aborted as though it failed to synchronize, and introduces no new failure modes to TCP.) Aborting the connection from this unsynchronized condition leaves no extra state at either endpoint; the server returns to *LISTEN*, and the client closes the connection. If the server returned no *TW-Negotiate* option, the connection will use current TCP semantics: the side that issues the active close will enter *TIME-WAIT* (or both will if they close simultaneously).

Apabila algoritma ini diterapkan akan mengurangi *server load* .

2. Mengurangi *Time-wait loading* di HTTP dan FTP Server
  - Memodifikasi TCP status Time-wait antara server dan *client* ketika koneksi ditutup, TCP *client* dimodifikasi sehingga setelah koneksi ditutup *client* akan mengirimkan <RST> paket ke server dan mengubah statusnya sendiri ke *TIME-WAIT*. <RST> paket akan menghapus status *TIME-WAIT* pada TCB di server.
3. Memodifikasi HTTP sehingga akan mengurangi kontribusi dari Protocol ke *TIME-WAIT loading*.
  - Modifikasi HTTP dilakukan karena HTTP merupakan sumber utama dari trafik internet pada *client/server*.
  - Modifikasi HTTP 1.1 termasuk notifikasi dari *client* ketika koneksi tertutup. Notifikasi ini berupa *extension request*, bernama *CLIENT\_CLOSE*.
  - *CLIENT\_CLOSE* tidak membutuhkan jawaban, *client* akan menutup *TCP connection* setelah mengirim *CLIENT\_CLOSE* ke server.
  - Modifikasi pada HTTP ini adalah solusi yang efektif untuk mengurangi *TIME-WAIT loading* akan tetapi memiliki banyak kendala, Server membutuhkan suatu mekanisme untuk berkomunikasi dengan client sehingga server dan client akan menghentikan koneksi yang sama.

H. Balakrishnan et al. berfokus pada meningkatkan performa pada *single connection* dan *multiple TCP connection* untuk meminimalisir *request time out* ke server ketika terjadi *congestion* dan *bottleneck*.

1. *Single connection performance*

- 1.1 *Enhance Loss Recovery*

Menggunakan algoritma *right-edge recovery*.

Ketika algoritma *right-edge recovery* digunakan, *sender* akan mengirimkan paket baru yang merespon setiap *duplicate acknowledgments*. Hal ini akan membuat *receiver* akan men *generate* tambahan *duplicate acknowledgments*, yang kemudian akan men *trigger fast retransmission* pada *sender*. Dengan mengeliminasi *coarse timeout*, *sender* yang menggunakan *right-edge recovery* akan menghasilkan performa transfer dua kali lebih cepat daripada standar *transmitter*.

2. *Multiple TCP Connection*

- 2.1 *Integrated Congestion Control/Loss Recovery*

Metode *Integrated Congestion Control/Loss Recovery* membuat aplikasi untuk menggunakan koneksi TCP terpisah untuk setiap transfer dengan menambah modifikasi pada *network stack*. Metode ini membagi fungsi TCP ke dalam dua kategori, yang pertama berhubungan dengan *reliable, ordered byte-stream abstraction of TCP* dan berkaitan dengan *congestion control and data-driven loss recovery*. *Congestion control and data-driven loss recovery* dilakukan terintegrasi secara paralel pada suatu set koneksi. Metode ini disebut jg *TPC-Int*

3. Solusi *Alternatif: Application-level Multiplexing*  
Application-level multiplexing menghindari penggunaan multiple parallel TCP connections dan masalahnya dengan melakukan *multiplexing* pada beberapa data streams ke dalam satu *TCP connection*. Solusi ini berhasil mengurangi *latency* secara signifikan ketika mengakses web saat menggunakan P-HTTP.
4. Meningkatkan *default socket buffer size* untuk mengurangi terjadinya *bottleneck*.

### References

1. Statistical time-division multiplexing – Wikipedia (2019, 26-10-2008)  
[https://en.wikipedia.org/wiki/Statistical\\_time-division\\_multiplexing](https://en.wikipedia.org/wiki/Statistical_time-division_multiplexing)
2. H. Balakrishnan et.al (1998). TCP Behavior of a Busy Internet Server: Analysis and Improvements. *IEEE Infocom*
3. N. Baharudin et.al (2013). Review on Multiplexing Techniques in Bandwidth Utilization. *Middle-East Journal of Scientific Research* 18 (10): 1510-1516, 2013
4. F. Theodore et.al. the TIME-WAIT state in TCP and Its Effect on Busy Servers.

## TUGAS

Nama mahasiswa:

### 1. M DANIAL SENTOSA

Permasalahan : ketika user menggunakan internet di wilayah tertentu, kecepatan internet menurun, tidak seperti yang dijanjikan para provider, dikarenakan setiap provider membatasi pengguna pada suatu daerah, misal suatu daerah dibatasi 1 GB/s dan penggunanya ada 10 orang, maka 1 GB/s akan dibagikan kepada 10 orang, maka setiap orang mendapatkan kecepatan 100 Mb/s sehingga semakin banyak pengguna maka internet akan semakin lambat bahkan terjadi packet loss

Solusi : pada jurnal dengan judul *The TIME-WAIT state in TCP and Its Effect on Busy Servers* oleh Theodore Faber menjelaskan dengan menggunakan sistem TIME-WAIT

Jurnal ini telah menerapkan dan menguji solusi TCP yang lebih sederhana dan solusi HTTP untuk menunjukkan pergeseran waktu TIME-WAIT dari klien ke server. Kami telah menyajikan bukti eksperimental bahwa loading TIME-WAIT dapat mempengaruhi kinerja server di bawah SunOS 4.1.3. Dalam kondisi ini, throughput dapat dikurangi hingga 50%.

Dengan menggunakan system ini dan di kombinasikan dengan koneksi TCP persistent, hal ini menggunakan lebih sedikit memori dari pada Sun OS 4.1.3 yang tidak dimodifikasi menggunakan koneksi pertant, untuk beban klien tertentu. System yang digunakan berkoneksi terus-menerus, masalah pemuatan memori secara langsung. Ini dapat mengurangi biaya penyebaran server, yang dapat menjadi sangat penting server tertanam/baterai

Menurut theodore faber, pendistribusian TIME-WAIT TCBS ke client dengan memanfaatkan angka pertumbuhan mereka untuk mengurangi memory load pada server. Kelebihannya yaitu mendukung koneksi yang persistant dan berbagi manfaat lainnya seperti menghindari extra TCP 3 handshakes.

Pada jurnal ini percaya bahwa TCP pada akhirnya harus dimodifikasi untuk mendukung TIME-WAIT negotiation. Upgrade yang akan datang dari IP versi 4 ke versi 6 juga merupakan peluang untuk meninjau kembali implementasi dan desain TCP. Ini akan menjadi waktu yang tepat untuk memasukkan TIME-WAIT state negotiation.

Algoritma TIME-WAIT state negotiation yaitu mengizinkan busy server untuk menerima koneksi dari klien yang memiliki TIME-WAIT overhead. Penggunaan algoritma ini tidak harus mengubah penggunaan koneksi dekat dengan protokol mereka dan tidak mengalami perubahan performa yang terkait dengan pendistribusian TIME-WAIT TCBS.

Selain memodifikasi TCP, menurut theodore faber, memodifikasi HTTP juga diperlukan karena itu merupakan salah satu komponen terbesar pada internet traffic. Untuk mensupport koneksi yang persistent, indikasi end-of-connection dan end of transaction harus dipisahkan. Pemisahan ini mengizinkan kita untuk memodifikasi HTTP untuk mengizinkan klien mengaktifkan kembali close connections dan menahan TIME-WAIT state.

# CV. Alumni Saudara

Politeknik negeri Sriwijaya



Muhammad Iqbal Rizky Tanjung  
Muhammad Wahyudi

# Profil Kampus Polsri

- Gedung Pendidikan
- Gedung KPA
- Gedung Kuliah 1 sd 6
- Gedung Kuliah DIV
- Gedung Laboratorium dan Bengkel

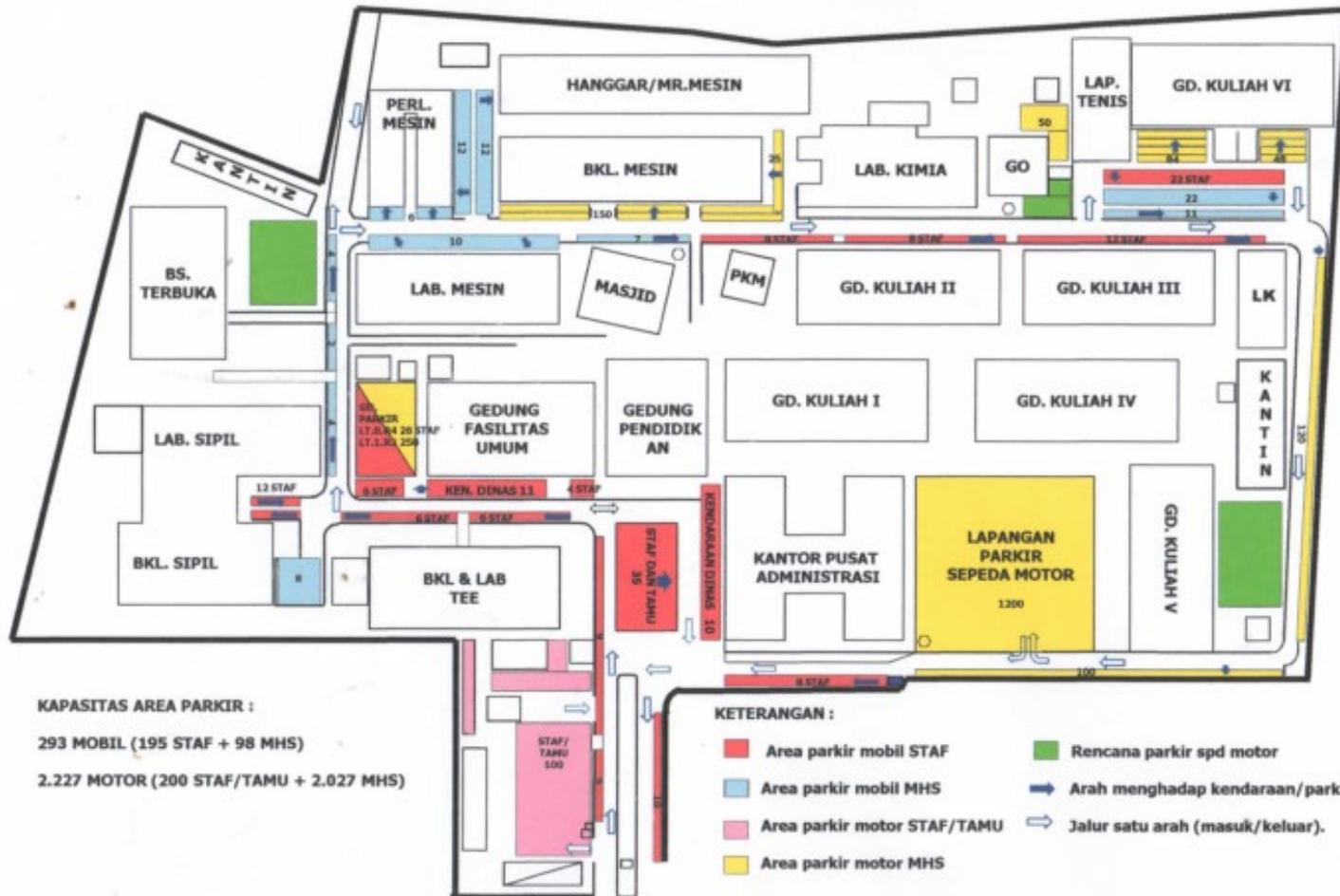


Jumlah User :  
Mahasiswa : 6.000  
Dosen/Staff : 600

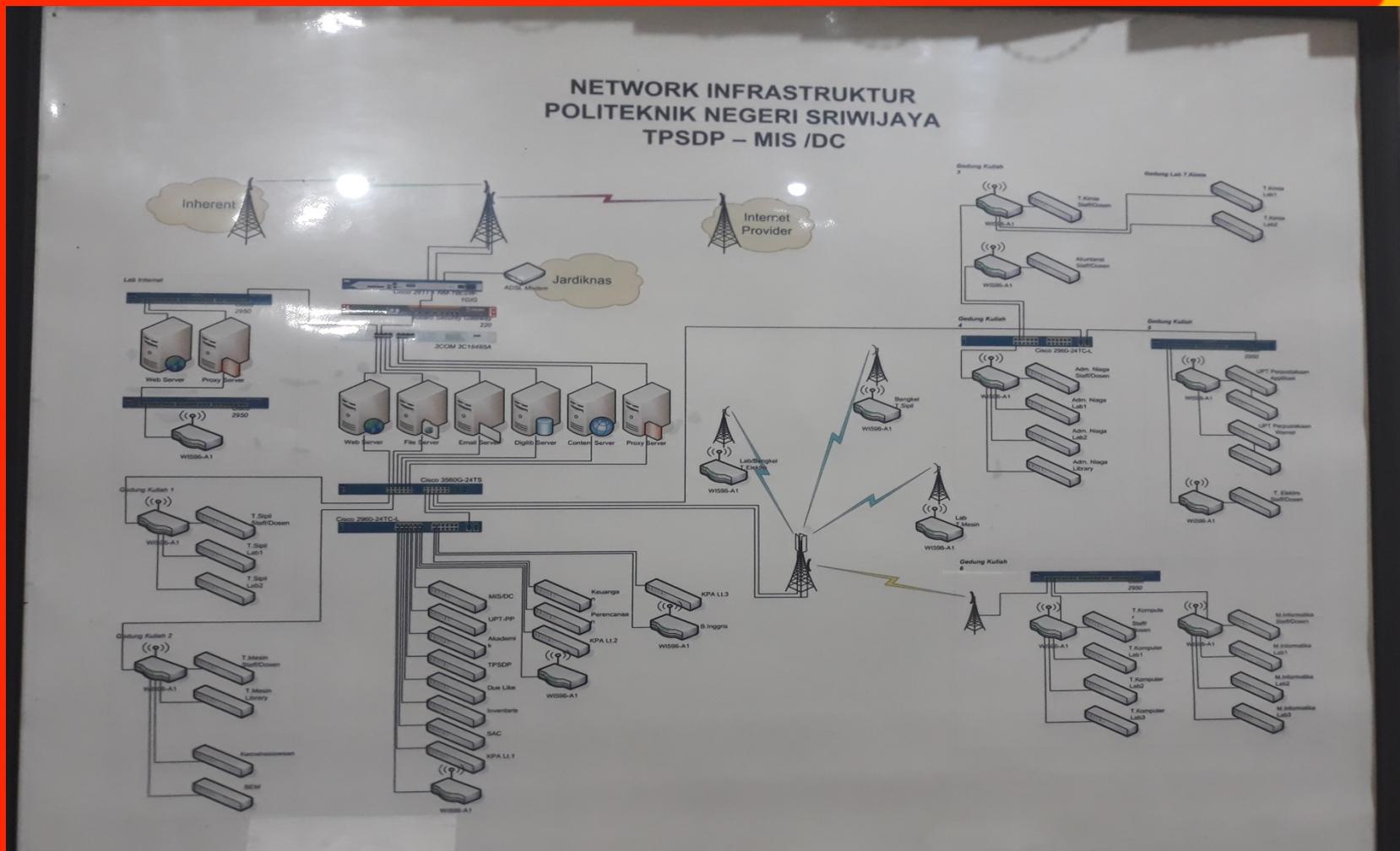


# Denah Kampus Polsri

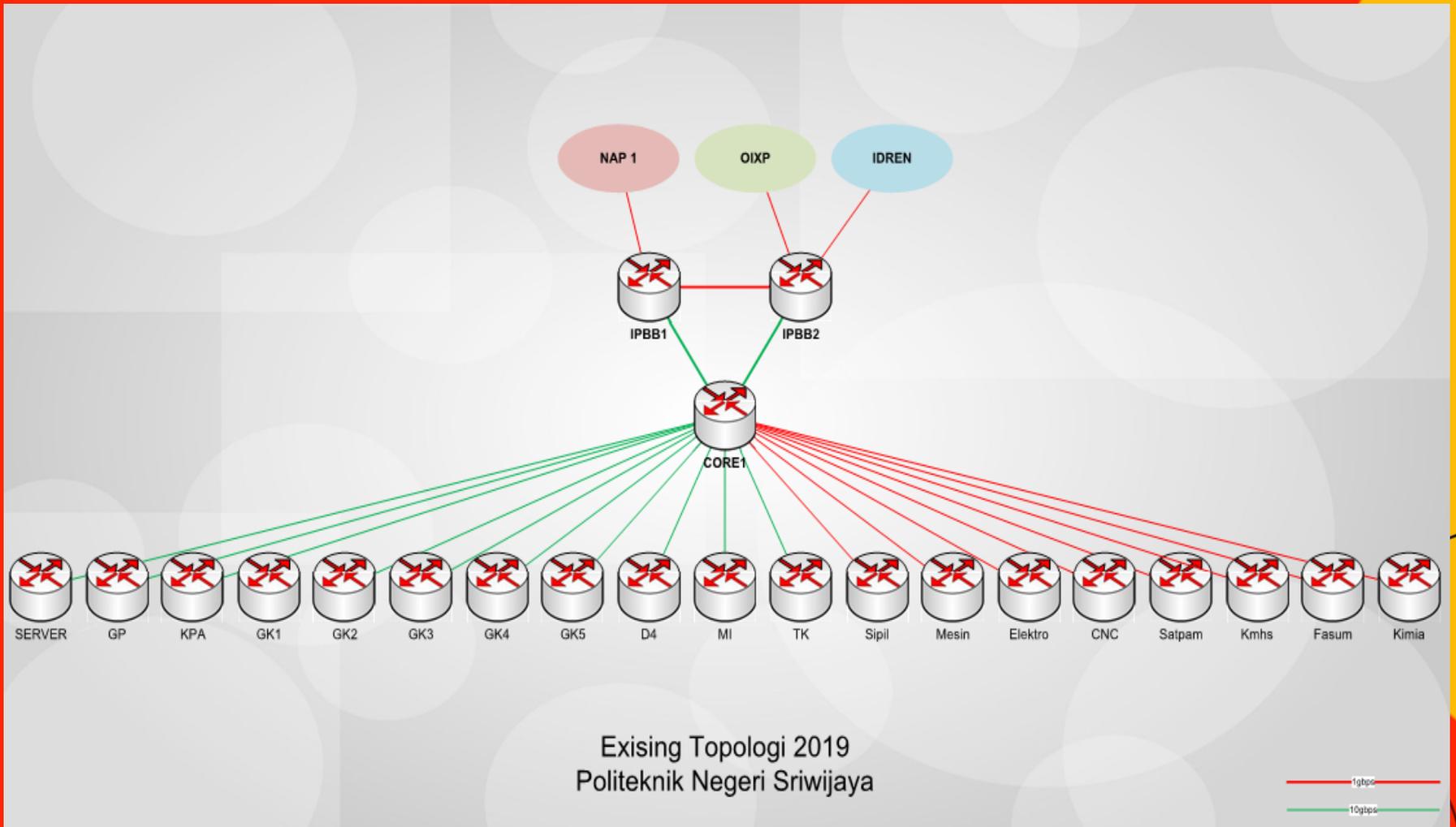
## SITUASI KAMPUS POLITEKNIK NEGERI SRIWIJAYA PARKIR DAN JALAN SATU ARAH



# Topologi Jaringan Lama

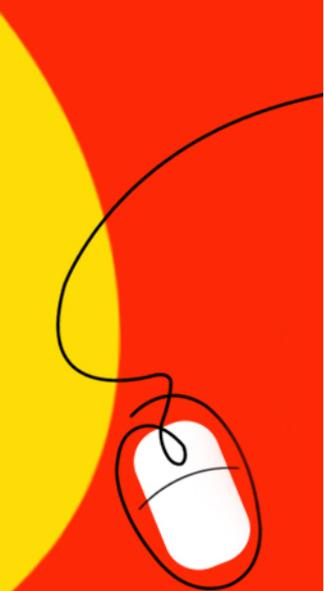


# Topologi Jaringan Baru



# Rencana Anggaran Biaya (RAB)

NO	Nama Barang	Jumlah	Harga Satuan	Total
1	Cisco ASR1001	1 buah	Rp.93.221.000,-	Rp.93.221.000,-
2	Mikrotik CCR1036 EM	2 buah	Rp.17.899.900,-	Rp.35.799.800,-
3	Cisco 4506 E	1 buah	Rp.20.100.000,-	Rp.20.100.000,-
4	Cisco FI 6454	1 buah	Rp.356.448.707,-	Rp.356.448.707,-
5	Cisco 3650 G	7 buah	Rp.2.500.000,-	Rp.17.500.000,-
6	Catalist 2960	1 buah	Rp.24.000.000,-	Rp.24.000.000,-
7	Mikrotik 2011 UiAS	9 buah	Rp.2.024.000,-	Rp.18.216.000,-
Jumlah				Rp.565.285.507,-



# Analisa dan Rekomendasi Topologi Jaringan Polsri

## Analisa

- Pada topologi jangsan polsri, terdapat perkembangan dari topologi lama dengan topologi yang baru. Seperti pada jaringan yang dulunya menggunakan inernet yaitu jaringan antar kampus, di ubah menggunakan fiberoptik yang lebih bagus dan lebih fleksibel.

## Rekomendasi

- Setelah kami analisa topologi jaringan polsri, kami merekomendasikan yaitu, pada Core1 itu sebaiknya di tambah menjadi 2 Core Yaitu Core1 dan Core2. agar, apabila terjadi kerusakan mesin pada Core1 maka tidak ada backupan Pengubung di setiap gedung. Maka dari itu sebaiknya dibuat menjadi 2 mesin Core. Jika memiliki 2 Core apabila salah satu mesin Core rusak maka akan ada backup mesin Core yang satunya untuk membantu menghubungkan jaringan di setiap gedung yang ada. Walaupun kecepatannya akan menurun.



Nama : Muhammad Fajar  
NIM : 192400237

## Permasalahan

Permasalahan yang terjadi di dalam video "why internet slow down when it's busy" adalah bottleneck antara kapasitas dari user dan kapasitas dari ISP. Network bottleneck atau hambatan jaringan mengacu pada kondisi di mana aliran data dibatasi oleh komputer atau sumber daya jaringan. Aliran data dikontrol sesuai dengan bandwidth berbagai sumber daya sistem. Jika sistem yang bekerja pada jaringan memberikan volume data yang lebih tinggi dari apa yang didukung oleh kapasitas jaringan yang ada, maka kemacetan jaringan akan terjadi. Kemacetan atau traffic pada internet inilah yang menjadi permasalahan. Traffic ini terjadi pada saat user mengakses keluar dengan bandwidth 100Mb/s setiap usernya dan bandwidth yang di sediakan oleh ISP sebesar 1 Gb/s. Jumlah user yang ada dan jumlah bandwidth yang di sediakan ISP tidak sebanding. User yang ada berjumlah 50, setiap user memiliki bandwidth 100Mb/s, dan jika seluruh user menggunakan seluruh bandwidth nya maka membutuhkan kapasitas jaringan 5Gb/s. kapasitas jaringan tersebut melebihi dari bandwidth yang disediakan oleh ISP. ISP juga tidak selalu benar menyatakan kapasitas yang diberikan, mereka bertaruh semua pelanggan tidak menggunakan semua bandwidth mereka sepanjang waktu. Angka maksimal yang diberikan pada ISP belum tentu benar. Secara teori kecepatan maksimal yang diberikan berdampak jika setiap pengguna lain tidak menggunakan bandwidth sama sekali. ISP juga tidak menjamin 100% layanan yang mereka berikan lancar dan tidak ada gangguan. Jadi ini bisa menjadi faktor internet melambat. Selain itu juga perangkat keras atau device yang digunakan harus sudah mampu dengan kapasitas, perangkat yang tidak mencukupi atau tidak support akan menimbulkan juga bottleneck. Penumpukan data yang akan dikirimkan biasanya jika terlalu lama data menunggu untuk dikirim maka akan langsung di buang paket data tersebut sehingga terjadinya packetloss.

## Solusi

Karena user input lebih banyak dari kapasitas yang diberikan, otomatis sistem output akan menekan dan memaksa sistem input untuk bekerja mengimbangnya. Untuk mengimbangnya maka di klasifikasikan pengiriman data tersebut menjadi data prioritas, seperti data voice akan di jadikan prioritas pengiriman di bandingkan data teks, jadi pembagian data tersebut dari klasifikasi prioritas. Dengan adanya klasifikasi data prioritas, data tersebut akan lebih dahulu sampai ke penerima. Jadi ini tidak akan membuang data prioritas. Paketloss hanya akan terjadi pada data yang prioritasnya rendah. Selanjutnya dengan cara membagi bandwidth dengan mengklasifikasikan user yang di prioritaskan, biasanya bandwidth yang tidak dibagi atau hanya satu saluran dan bandwidth yang diberikan pada setiap user sama besar, maka akan menimbulkan sebagian user mendapat penundaan paket Time-Wait. Untuk itu user yang penting atau user yang prioritas menggunakan jalur khusus yang bandwidthnya telah diatur sehingga tidak menimbulkan terlalu banyak delay yang diterima oleh user yang di prioritaskan. Akses yang banyak dalam satu network membuat menumpuknya IP yang membuat jalur bandwidth yang kita miliki penuh sehingga proses pengiriman data menjadi sangat terganggu dan terasa lambat. Hal ini juga bisa disebut dengan istilah Flooding. Flooding ini terjadi karena jalur bandwidth terbanjiri oleh IP address. Untuk mengatasi flooding ini yaitu mendisable proxy internal dan menambahkan filter rules untuk drop flooding attack. Penambahan rules bisa di setting melalui device, perangkat internet atau penambahan hardware tertentu untuk membatasi IP address yang masuk seperti firewall.

Lalu dilihat dari hardware yang digunakan. Hardware yang digunakan juga harus compatible dengan kapasitas yang disediakan oleh ISP. Jika ISP memberikan bandwidth sebesar 1Gb/s maka perangkat yang digunakan harus sudah support Gigabit Ethernet. Dengan perangkat yang sesuai dengan kapasitasnya maka kapasitas yang diberikan oleh ISP dapat terpakai seluruh bandwidth yang disediakan.

Nama : Muhammad Fajar  
NIM : 192400237

Selanjutnya mengimplementasikan ukuran buffet socket harus dibesarkan dari ukuran defaultnya. Dengan membesarkan socket buffer maka akan menghindari receiver window dari bottleneck. Buffer socket yang besar memungkinkan anggota user untuk mendistribusikan data dan events menjadi lebih cepat tetapi juga mengambil banyak memori.

TUGAS COMPUTER NETWORK AND BUSINESS INFORMATION  
DOSEN : Dr.Edi Surya Negara, M.Kom.

**Nama : Muhammad Hadrifiansyah**

NIM : 192420010

Kelas : Reguler A R1

**Nama : Eko Purwanto**

NIM : 19240019

Kelas : Reguler A R1

Kasus :

Transmisi data sebesar 1 GB/s tidak merata kepada penerima paket, user menerima masing-masing 100Mbps sedangkan banyak user sebesar 50 user. Bagaimana cara untuk stabil pengiriman data sedangkan sering terjadi gagal pengiriman data. Di waktu tertentu trafik menunjukkan data terlalu lambatnya dalam transmisi internet.atau sering di sebut Backbone atau server terlalu sibuk.

Solusi :

1. Dalam membangun sebuah jaringan dan server harus melakukan beberapa tahapan yang harus di perhatikan yaitu :
  1. Pemilihan ISP Dedicated dan bandwidth yang stabil sesuai dengan kebutuhan perusahaan.
  2. Membangun perangkat Server sesuai dengan standard internasional, serta pemilihan penyimpanan memori yang besar untuk jangka panjang.
  3. Pembagian Paket internet menjadi user priority and user guest.
2. Dalam kasus yang telah terjadi ada pun yang harus di perhatikan dan menindak lanjuti :
  1. Melakukan Perubahan Paket Data atau ISP pada Server dan Maintenace Server karena dalam Off-loading memori, semakin banyak klien mengurangi memori server dan kecepatan koneksi pada server. Tumbuhnya kecepatan dan lambatnya akses karena peningkatan jumlah klien dan bandwidth yang tersedia di jaringan. Apabila semakin banyak user koneksi ke pada server, maka server akan semakin sibuk.
  2. Pembagian Paket internet menjadi user priority and user guest.  
Yang tujuannya untuk membagi internet prioritas dan internet yang Non-Prioritas.
  3. Data yang tidak terkirim dan Hilang akan tersimpang didalam Cookies Memori.
  4. Dalam Pembagian Kuota Internet masing-masing user bisa di atur melakuai server.  
Dengan cara mendaftarkan IP Perangkat tersebut dan memangement sesuai kebutuhan.

UTS

### **Computer Network and Communication (MTIK 111)**

Nama : 1. M. Nang Alhafiz (192420008)

2. Andrian Perdana (192420017)

Permasalahan :

- Dalam satu keadaan dimana kapasitas internet yang diberikan oleh ISP (Internet Service Provider) sebesar 1 Gbps dan dibagikan ke pelanggan sebesar 100 Mbps per pelanggan, jika suatu waktu ada 50 pelanggan yang mengakses internet dengan masing-masing pengguna menggunakan 100 mbps, maka kapasitas total menjadi 5 Gbps, sehingga menyebabkan antrian paket data untuk melewati oleh server jaringan karena melebihi kapasitas bandwidth yang tersedia dan menyebabkan akses internet menjadi lambat. Ketika memori server sudah penuh, ini dapat menyebabkan server membuang paket data yang ada dalam antrian data untuk ditransfer.

Solusi :

1. Memodifikasi waktu tunggu / Time-Wait server dalam petukaran data sehingga bisa mengurangi 50 % kehilangan paket akibat antrian;
2. Memodifikasi TCP untuk bertukar waktu tunggu TCB setelah penutupan yang sukses;
3. Memodifikasi HTTP untuk mendorong klien untuk menutup koneksi TCP;
4. Penerapan IP versi 6 dapat meningkatkan kecepatan koneksi pertukaran data;
5. Implementasi jaringan di masa depan harus meningkatkan ukuran buffer socket default untuk menghindari hambatan jendela penerima;
6. Menggunakan kumpulan koneksi secara parallel antara client dan server sehingga pengguna lebih agresif daripada menggunakan koneksi TCP tunggal;
7. Memodifikasi node server untuk menunjukkan kapan pengiriman ulang TCP terjadi dengan mengirimkan paket yang di cache.

# CV. Alumni Saudara

Politeknik negeri Sriwijaya



Muhammad Iqbal Rizky Tanjung  
Muhammad Wahyudi

# Profil Kampus Polsri

- Gedung Pendidikan
- Gedung KPA
- Gedung Kuliah 1 sd 6
- Gedung Kuliah DIV
- Gedung Laboratorium dan Bengkel

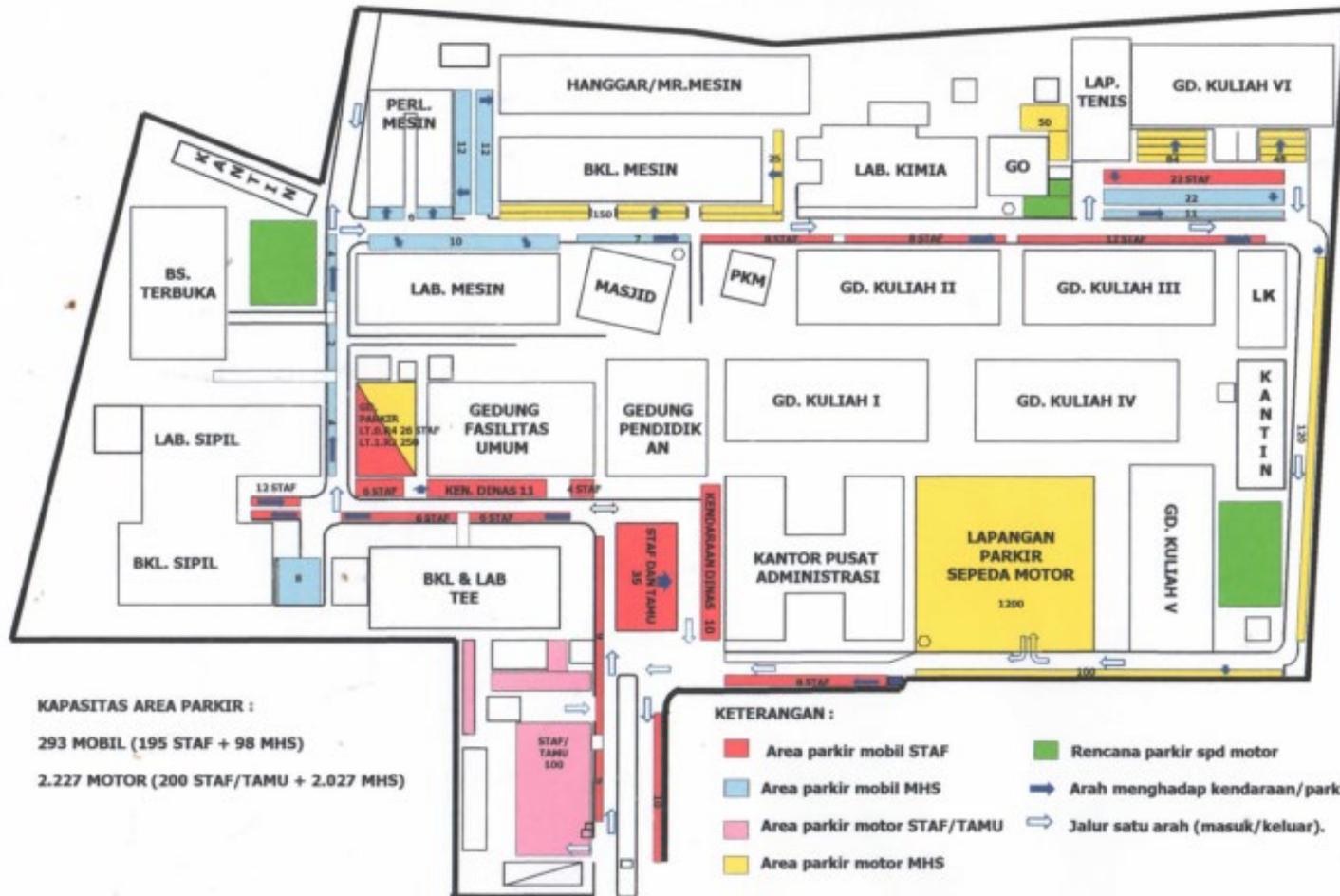


Jumlah User :  
Mahasiswa : 6.000  
Dosen/Staff : 600

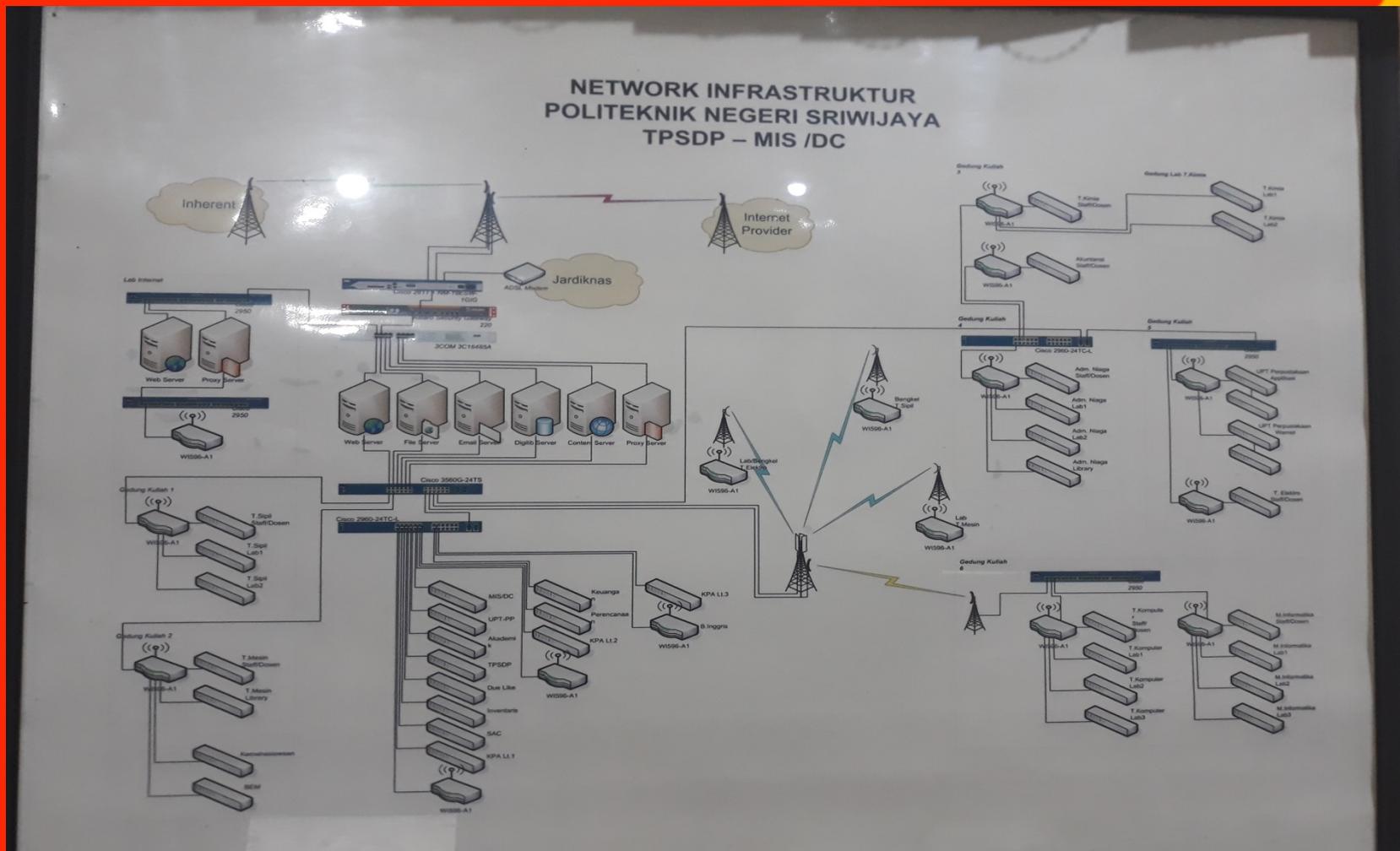


# Denah Kampus Polsri

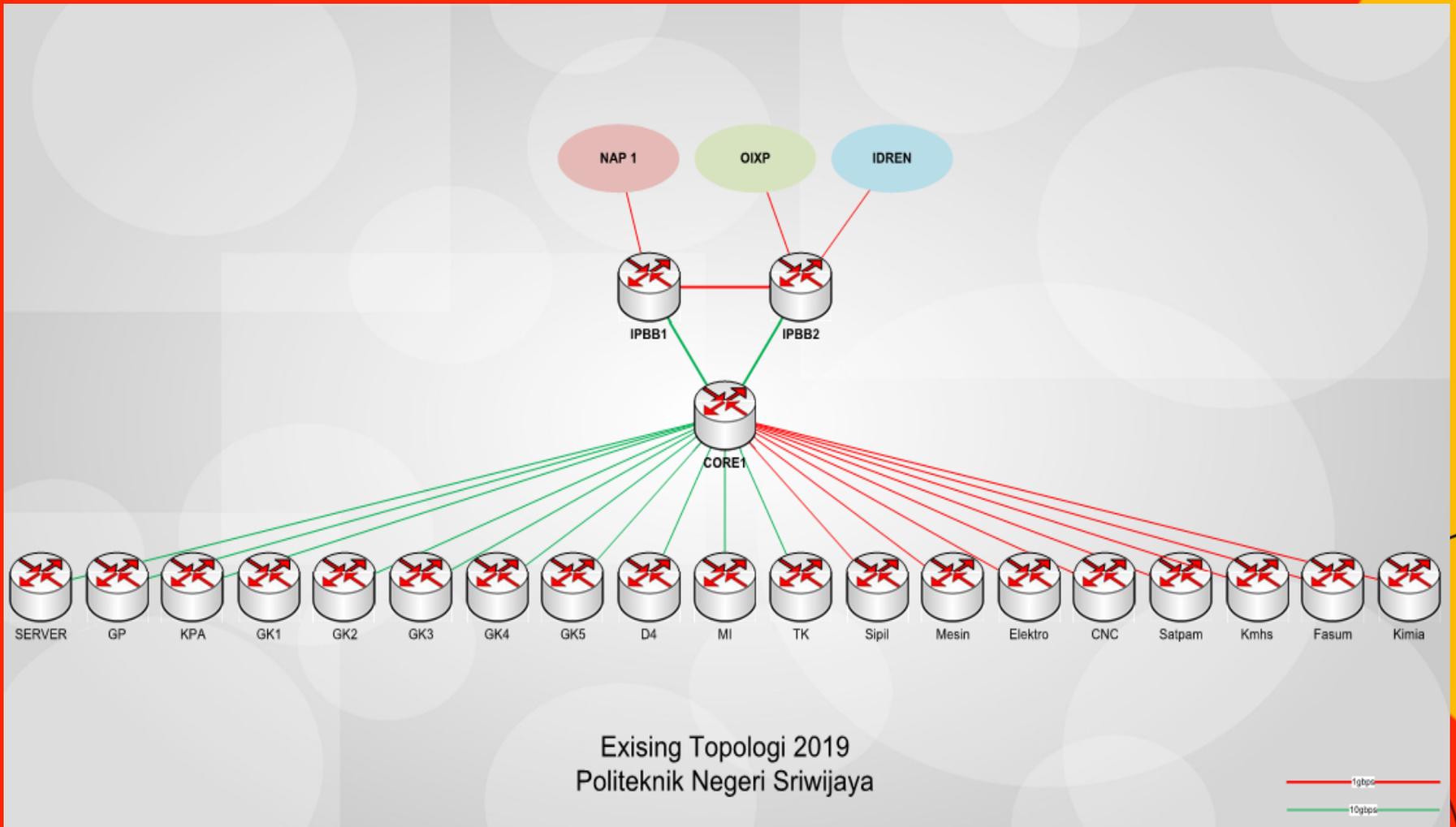
## SITUASI KAMPUS POLITEKNIK NEGERI SRIWIJAYA PARKIR DAN JALAN SATU ARAH



# Topologi Jaringan Lama

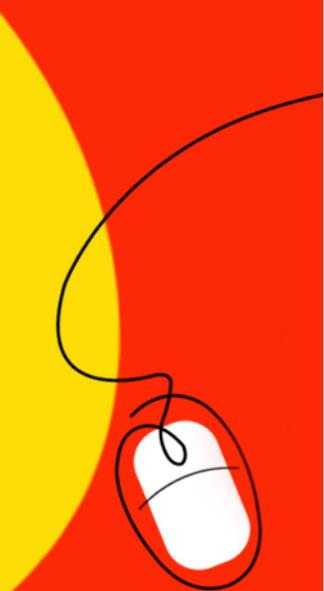


# Topologi Jaringan Baru



# Rencana Anggaran Biaya (RAB)

NO	Nama Barang	Jumlah	Harga Satuan	Total
1	Cisco ASR1001	1 buah	Rp.93.221.000,-	Rp.93.221.000,-
2	Mikrotik CCR1036 EM	2 buah	Rp.17.899.900,-	Rp.35.799.800,-
3	Cisco 4506 E	1 buah	Rp.20.100.000,-	Rp.20.100.000,-
4	Cisco FI 6454	1 buah	Rp.356.448.707,-	Rp.356.448.707,-
5	Cisco 3650 G	7 buah	Rp.2.500.000,-	Rp.17.500.000,-
6	Catalist 2960	1 buah	Rp.24.000.000,-	Rp.24.000.000,-
7	Mikrotik 2011 UiAS	9 buah	Rp.2.024.000,-	Rp.18.216.000,-
Jumlah				Rp.565.285.507,-



# Analisa dan Rekomendasi Topologi Jaringan Polsri

## Analisa

- Pada topologi jangsan polsri, terdapat perkembangan dari topologi lama dengan topologi yang baru. Seperti pada jaringan yang dulunya menggunakan inernet yaitu jaringan antar kampus, di ubah menggunakan fiberoptik yang lebih bagus dan lebih fleksibel.

## Rekomendasi

- Setelah kami analisa topologi jaringan polsri, kami merekomendasikan yaitu, pada Core1 itu sebaiknya di tambah menjadi 2 Core Yaitu Core1 dan Core2. agar, apabila terjadi kerusakan mesin pada Core1 maka tidak ada backupan Penghubung di setiap gedung. Maka dari itu sebaiknya dibuat menjadi 2 mesin Core. Jika memiliki 2 Core apabila salah satu mesin Core rusak maka akan ada backup mesin Core yang satunya untuk membantu menghubungkan jaringan di setiap gedung yang ada. Walaupun kecepatannya akan menurun.



## Computer Network and Communications.

Nama : 1. Ahmad Reza Fahlevi  
: 2. Nizar Firliansa  
Jurusan : Magister Teknik Informatika Kelas R1  
Dosen : Dr. Edi Surya Negara, S.Kom., M.Kom

Permasalahan :

Kenapa Internet lambat ketika banyak yang mengakses jaringan.

Internet lambat saat jam sibuk dan banyak yang mengakses jaringan disebut dengan kemacetan (congestion), karena terlalu banyak akses dalam kurun waktu yang bersamaan, hal ini terjadi karena jumlah pengakses lebih tinggi daripada kapasitas server, serta bisa menyebabkan paket hilang (loss packet).

Solusi :

1. Menambah kemampuan server utama dengan meningkatkan spesifikasi dari server tersebut.  
Maka dapat menampung lebih banyak akses data.
2. Kontrol kemacetan, dengan melakukan prioritas data.
3. Membagi paket yang kapasitasnya besar menjadi segmentasi yang sesuai dengan bandwidth
4. Mengurangi kehilangan data dengan retransmission and loss recovery..



# PENDAHULUAN



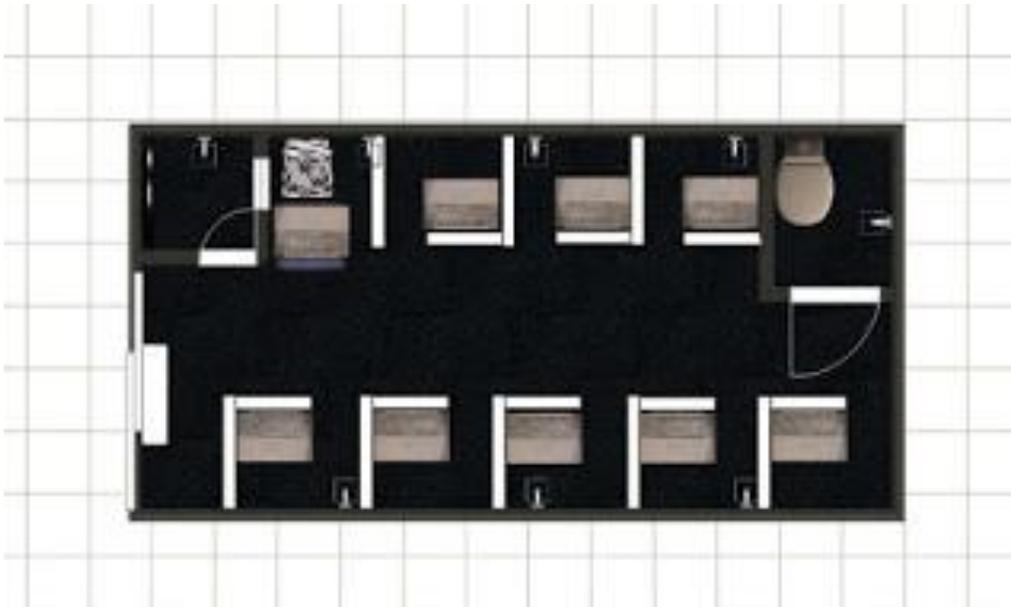
Lokasi

Kebutuhan

Keadaan

Prospek

# DENAH



## ■ LANTAI 1

■ 8 Komputer Client

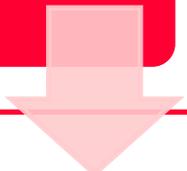
■ 1 Komputer Billing

Jam  
Operasional

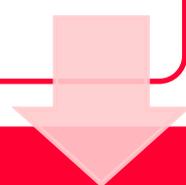
07:00 – 22:00  
WIB

# PROSES BISNIS

Jaringan Efektif

A light red downward-pointing arrow indicating a flow from the first step to the second.

Budget Rp 100,000,000

A light red downward-pointing arrow indicating a flow from the second step to the third.

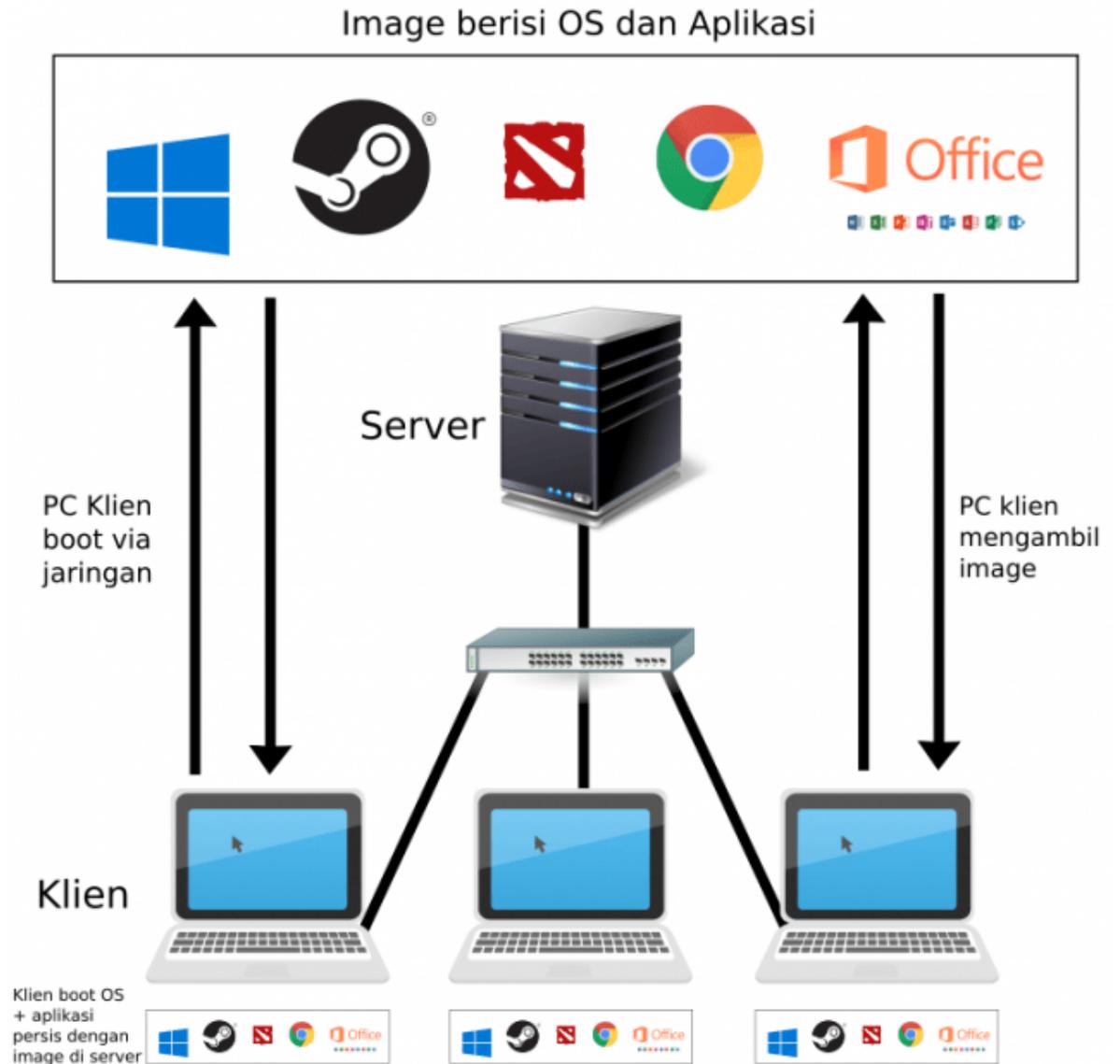
Support untuk Prospek Kedepan

# SOLUSI

DISKLESS

HOW?

ILUSTRASI



# FUNGSIONAL

FIX

Sulitnya untuk menambah aplikasi/game

Sulitnya untuk selalu update aplikasi/game

Sulitnya untuk maintenance

# ALAT DAN BAHAN

- SERVER UNBK DELL T30 Xeon Quad E3-1225v5/16Gb/1TB// 2LAN PORT
- PC client (Komputer Core I3 & LED AOC 16 Full)

- *MIKROTIK RB450G routerboard gigabit ( router indoor)*
- Switch Hub TPLINK 16 Port

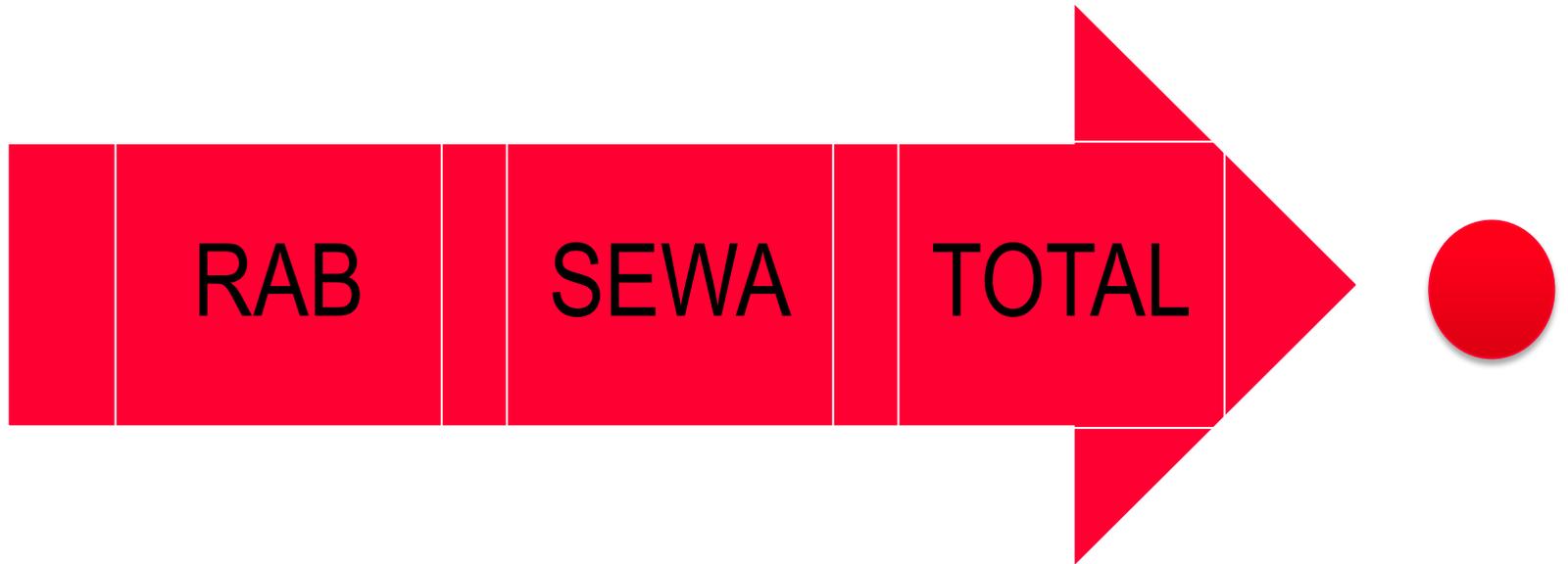
- *Cable UTP Cat 6*
- *Connector RJ45 Cat 6 AMP/COMMSCOPE (100pcs)*

- *Cyberindo Disk*
- UPS 600 VA

- *Camera CCTV SPC 2 Megapixel Tornado series Hybird*
- *Windows 7*

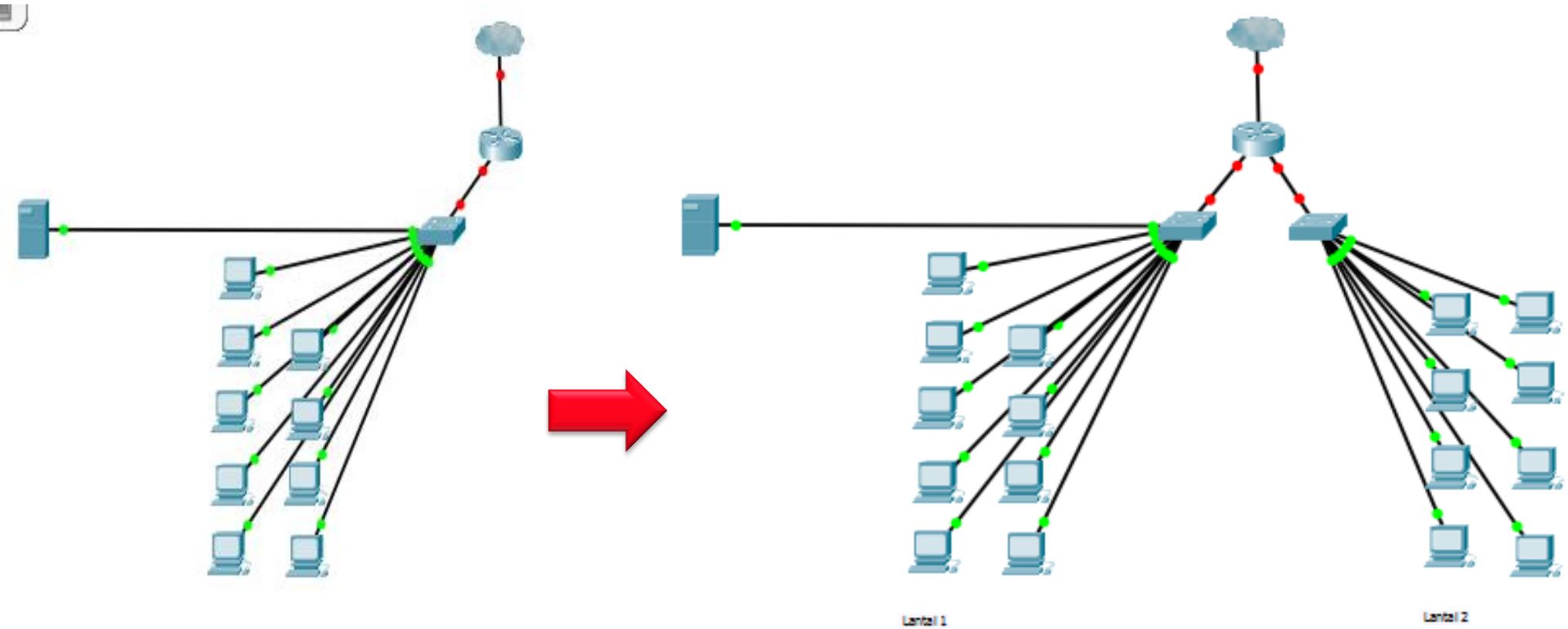
- Stop Kontak
- Kabel Tukuiki

# RAB



RAB dapat dilihat di lampiran, dalam bentuk M.S. Word

# DESIGN



# BENEFIT

Kemudahan Akses data



Biaya lebih murah tanpa hardisk



Serangan Trojan, virus, dan perubahan setting bisa dihindari



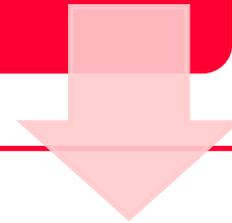
Tidak membayar banyak operator hanya satu saja operator untuk mengelola Komputer Server



Cocok untuk jenis usaha warnet, sekolah, dll

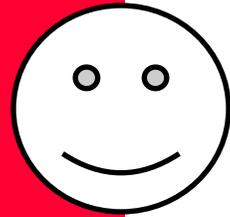
# KRITIK DAN SARAN

Perubahan mengenai penggunaan lebih dari 1 OS (*Operating System*)

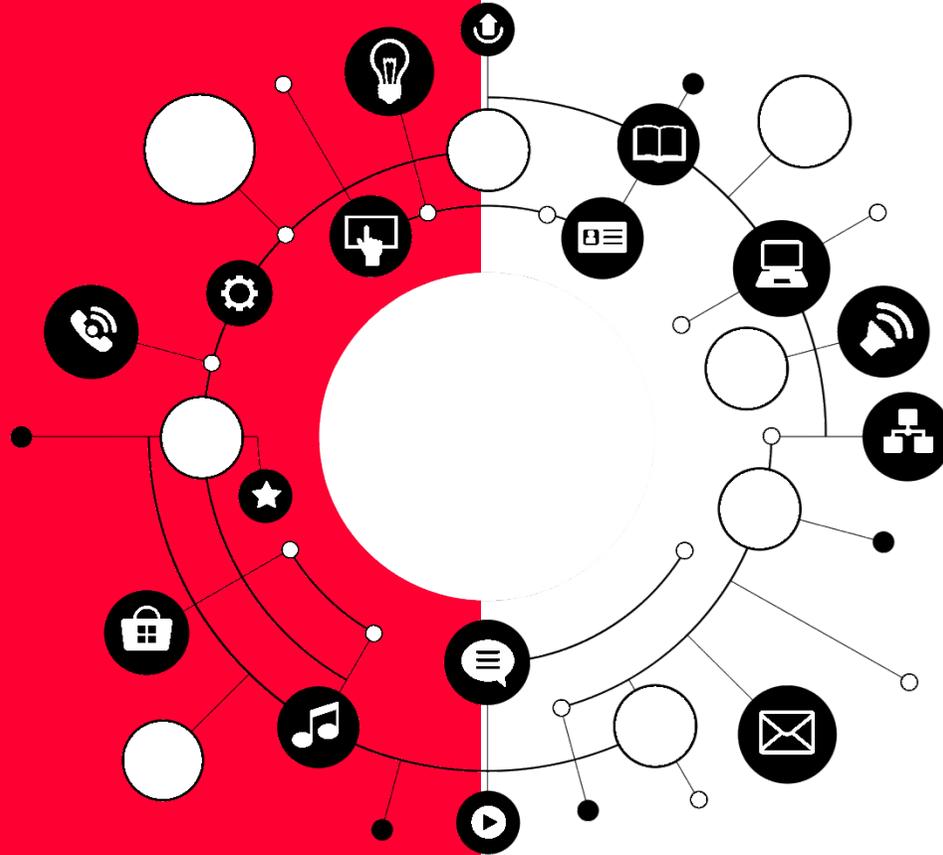


Kebutuhan internet kedepannya disesuaikan oleh kebutuhan user

**SEKIAN**



# Membangun Jaringan WARNET di JAYA.net



**CV MTI GROUP**

> Novita Angraini > Rika Seftiana > Sulis

# PENDAHULUAN



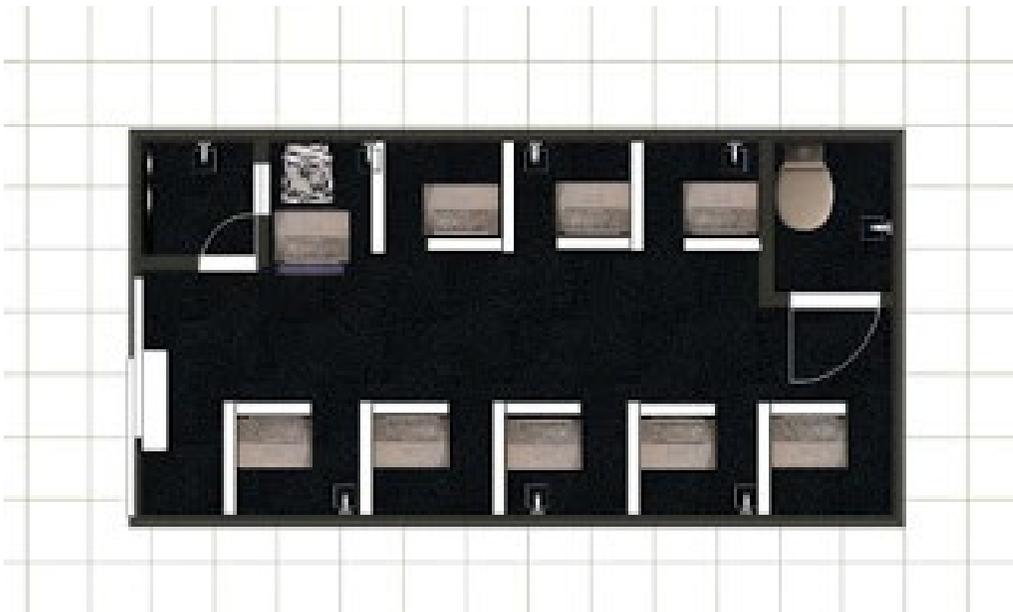
Lokasi

Kebutuhan

Keadaan

Prospek

# DENAH



## ■ LANTAI 1

■ 8 Komputer Client

■ 1 Komputer Billing

Jam  
Operasional

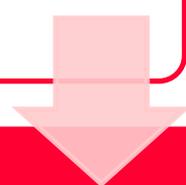
07:00 – 22:00  
WIB

# PROSES BISNIS

Jaringan Efektif

A light red downward-pointing arrow indicating a flow from the first step to the second.

Budget Rp 100,000,000

A light red downward-pointing arrow indicating a flow from the second step to the third.

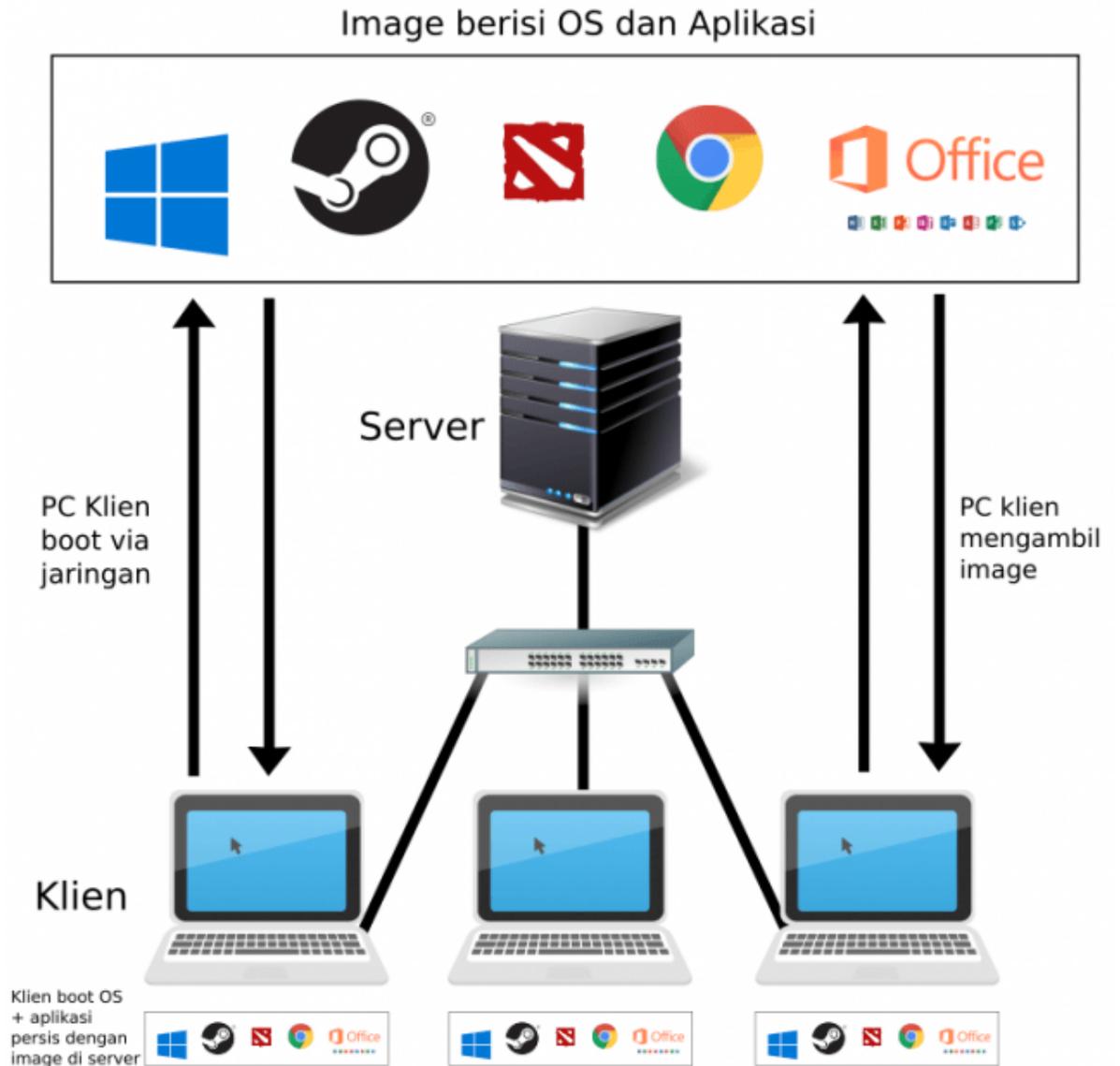
Support untuk Prospek Kedepan

# SOLUSI

DISKLESS

HOW?

ILUSTRASI



# FUNGSIONAL

**FIX**

Sulitnya untuk menambah aplikasi/game

Sulitnya untuk selalu update aplikasi/game

Sulitnya untuk maintenance

# ALAT DAN BAHAN

- SERVER UNBK DELL T30 Xeon Quad E3-1225v5/16Gb/1TB// 2LAN PORT
- PC client (Komputer Core I3 & LED AOC 16 Full)

- MIKROTIK RB450G routerboard gigabit ( router indoor)
- Switch Hub TPLINK 16 Port

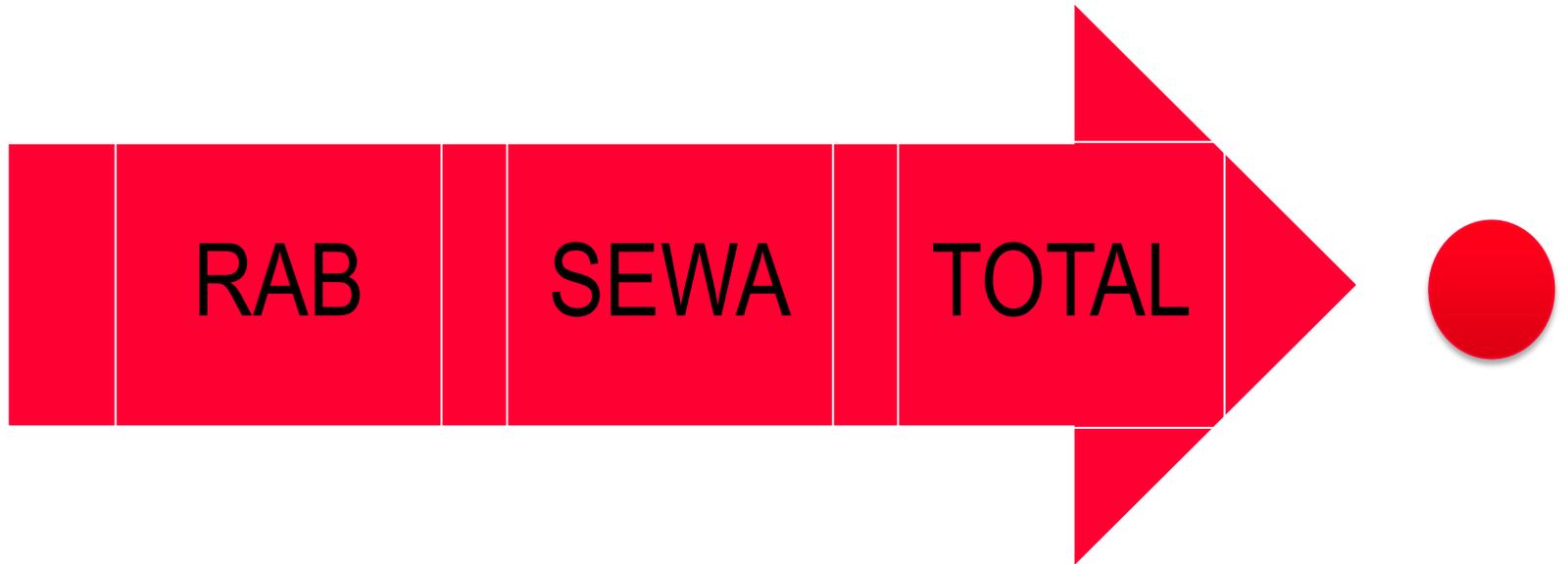
- Cable UTP Cat 6
- Connector RJ45 Cat 6 AMP/COMMSCOPE (100pcs)

- Cyberindo Disk
- UPS 600 VA

- Camera CCTV SPC 2 Megapixel Tornado series Hybird
- Windows 7

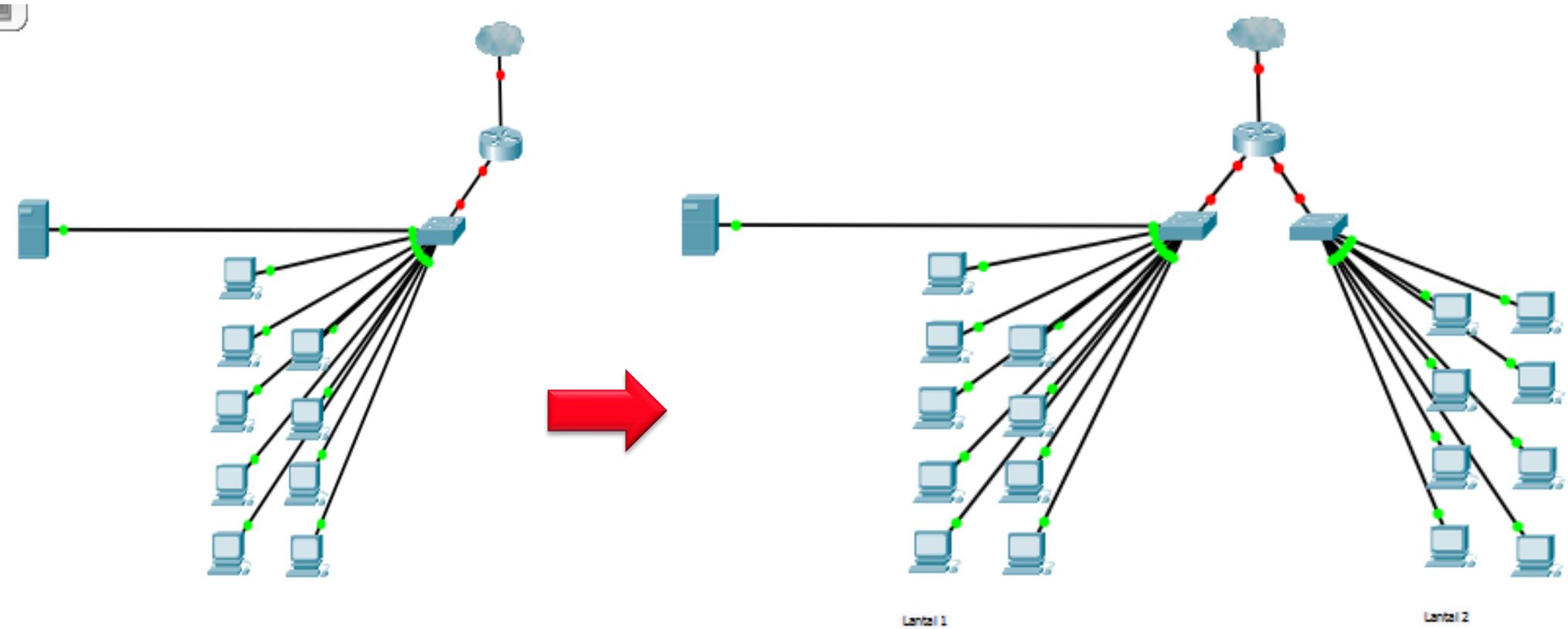
- Stop Kontak
- Kabel Tukuiki

# RAB



RAB dapat dilihat di lampiran, dalam bentuk M.S. Word

# DESIGN



# BENEFIT

Kemudahan Akses data



Biaya lebih murah tanpa hardisk



Serangan Trojan, virus, dan perubahan setting bisa dihindari



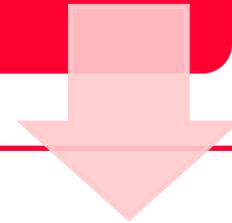
Tidak membayar banyak operator hanya satu saja operator untuk mengelola Komputer Server



Cocok untuk jenis usaha warnet, sekolah, dll

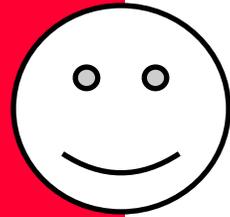
# KRITIK DAN SARAN

Perubahan mengenai penggunaan lebih dari 1 OS (*Operating System*)



Kebutuhan internet kedepannya disesuaikan oleh kebutuhan user

**SEKIAN**



**RANCANGAN ANGGARAN BIAYA 1 LANTAI**

- **Biaya Alat Dan Sarana**

Alat & Sarana	Uraian	Satuan	Biaya
Komputer server <i>Diskless AMD FM 2 + Seri A</i>	Rp 12.150.000	1 buah	Rp 12.150.000
PC client (Komputer Core I3 & LED AOC 16 Full)	Rp 2.175.000	9 buah	Rp 19.575.000
<i>MIKROTIK RB450G routerboard gigabit ( router indoor)</i>	Rp Rp1.670.000	1 buah	Rp Rp1.670.000
TPLINK TL-SF1016D SWITCH HUB 16PORT	Rp 209.000	1 buah	Rp 209.000
<i>Cable UTP Cat 6</i>	Rp 1.295.000	1 roll	Rp 1.295.000
<i>Connector RJ45 Cat 6 AMP/COMMSCOPE (100pcs)</i>	Rp 970.000	1 buah	Rp 970.000
<i>Cyberindo Disk</i>	Free	1 buah	Free
UPS 600 VA	Rp 355.000	1 buah	Rp 355.000
Camera CCTV SPC 2 <i>Megapixel Tornado series Hybird</i>	Rp 168.900	1 buah	Rp 168.900
<i>Stop Contact Kuningan 6 lubang</i>	Rp 30.0000	12 buah	Rp 360.000
<i>Kabel Tukuiki</i>	Rp 163.000	1 roll	Rp 163.000
<i>Windows 7</i>	Free	1 buah	Free
<b>TOTAL</b>			<b>Rp 36.915.000</b>

- **Biaya Sewa dan Internet**

Sewa	Satuan	Biaya
ISP Telkom Unlimited	/bulan	Rp 700.000
Perawatan PC, perlengkapan dan biaya cadangan	/bulan	Rp 500.000
Listrik	/bulan	Rp 300.000
<b>Total</b>		<b>Rp 1.800.000</b>

- **Total Pengeluaran**

<b>Modal awal</b>	<b>Rp. 100.000.000</b>
Alat dan Bahan	Rp 36.915.000
Biaya Sewa	Rp 1.800.000
<b>Total</b>	<b>Rp 38.715.000</b>
Sisa	Rp 61.285.000

Biaya sewa untuk 1 lantai adalah Rp 1,440.000/ bulan sehingga estimasi harus lebih besar dari itu. Jika perjamnya adalah sebesar 3000 maka sehari pendapatan harus minimal Rp 48.000/hari x 30 1,440,000 untuk menutupi biaya *maintenance* dan internet setiap bulannya. Dengan rincian 2 pengunjung mengakses 2 jam di 8 pc setiap harinya dikali 30 hari.

Rp 6000 x 8 pc x 30 hari = Rp 1.440.000/bulan

**INVESTASI KEDEPANNYA 2 LANTAI**

● **Biaya Alat Dan Sarana**

Alat & Sarana	Uraian	Satuan	Biaya
PC client (Komputer Core I3 & LED AOC 16 Full)	Rp 2.175.000	8 buah	Rp 17.400.000
TPLINK TL-SF1016D SWITCH HUB 16PORT	Rp 209.000	1 buah	Rp 209.000
Camera CCTV SPC 2 <i>Megapixel Tornado series Hybird</i>	Rp 168.900	1 buah	Rp 168.900
<i>Stop Contact Kuningan</i> 6 lubang	Rp 30.0000	12 buah	Rp 360.000
<b>TOTAL</b>			<b>Rp18.137.900</b>

● **Biaya Sewa dan Internet**

Sewa	Satuan	Biaya
ISP Telkom Unlimited	/bulan	Rp 700.000
Perawatan PC, perlengkapan dan biaya cadangan	/bulan	Rp 800.000
<b>Listrik</b>	/bulan	Rp 700.000
Total		Rp 2.200.000

● **Total Pengeluaran**

<b>Modal awal</b>	<b>Rp. 100.000.000</b>
Alat dan Bahan Lantai 1	Rp 36.915.000
Alat dan Bahan Lantai 2	Rp 18.137.900
Sewa	<b>Rp 2.200.000</b>
Total	Rp 57.252.900
Sisa	Rp 42.747.100

Sehingga untuk bulan pertama pengeluaran adalah sebesar Rp 59.427.900 untuk dua lantai

Biaya sewa untuk 2 lantai adalah Rp 2.200.000/ bulan sehingga estimasi harus lebih besar dari itu. Jika perjamnya adalah sebesar 3000 maka sehari pendapatan harus minimal Rp 96.000/hari x 30 = 2,880,000 untuk menutupi biaya *maintenance* dan internet setiap bulannya. Dengan rincian 2 pengunjung mengakses 2 jam di 16 pc setiap harinya dikali 30 hari.

Rp 6000 x 16 pc x 30 hari = Rp 2.880.000/bulan

Sekian  
**CV MTI Group**

## TOPIC

Why Internet Slows Down When it's Busy - Computerphile

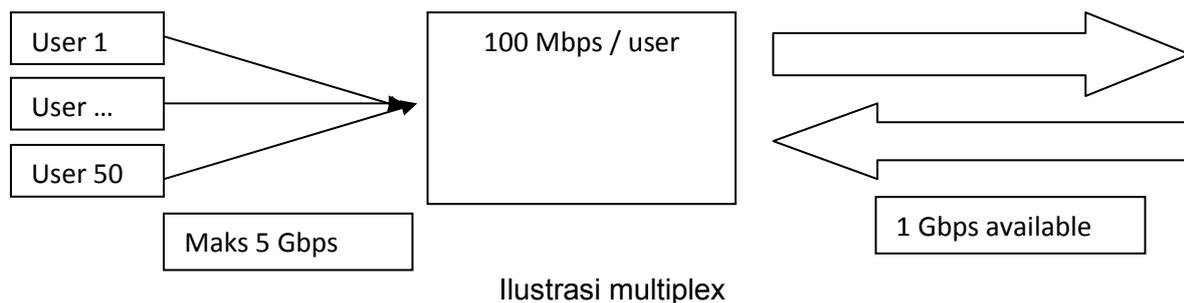
### summary of the video

“ISP / Network provider make assumption about how people will use their services. based on that assumption that allowed them to use multiplex”.( Computerphile )

“*Multiplexing is a method by which multiple analog message signals or digital data streams, coming from different sources are combined into one signal over a shared medium.* “  
(N.baharudin et.al,2013),

contoh skenario *multiplex* :

katakanlah ada 50 *user* yang membeli paket internet berkecepatan 100Mbps. Akan tetapi kenyataannya ISP hanya menyediakan link sebesar 1 gbps yang dishare ke 50 *user* tersebut.



Jika semua *user* menggunakan seluruh *bandwidth* yang dialokasikan kepada mereka bersamaan , maka jumlah beban yang akan diterima adalah 5 Gbps , sedangkan *bandwidth* yang tersedia hanya 1 Gbps, maka dampaknya akan terjadi, *congestion, delay , packet loss* dan sehingga mengakibatkan melambatnya koneksi internet.

Walaupun pada ada prakteknya adalah jarang terjadi kondisi dimana semua *user* memaksimalkan seluruh *bandwidth* yang mereka punya secara bersamaan, pada suatu waktu ada *user* yang menggunakan seluruh *bandwidth* yang dialokasikan kepada mereka, sedangkan beberapa *user* yang tidak menggunakan *bandwidth* yang dialokasikan secara maksimal kepada mereka , sehingga pada umumnya 1 Gbps yang dialokasikan ISP tersebut bisa “mencukupi” semua kebutuhan *user* , metode pembagian ini juga disebut *Statistical multiplexing*.

*Statistical multiplexing is a type of communication link sharing, very similar to dynamic bandwidth allocation (DBA). In statistical multiplexing, a communication channel is divided into an arbitrary number of variable bitrate digital channels or data streams. The link sharing is adapted to the instantaneous traffic demands of the data streams that are transferred over each channel.* (wikipedia.org)

Akan tetapi ada beberapa kondisi / waktu-waktu tertentu ketika orang-orang menggunakan seluruh *bandwidth* yang dialokasikan kepada mereka secara bersamaan, hal ini yang menyebabkan *users* merasakan *delay, slow internet connection or RTO*.

Teknik *multiplexing* telah diterapkan sejak lama pada era telepon rumah, tapi pada era internet, data / informasi yang diproses bergerak dengan cepat sehingga menimbulkan masalah kesulitan melakukan analisis dalam membagi *bandwidth* internet secara adil ke tiap-tiap user.

### Keywords / problems

- Bottleneck
- Congestion
- Delay
- Packet lost
- Mengakibatkan internet koneksi yang lambat

### Solutions

Ketika Koneksi internet semakin cepat, dan makin banyak user yang terhubung ke server tersebut maka server akan semakin sibuk, sehingga akan meningkatkan *Time-Wait loading* (theodore et al).

Ada 3 Solusi yang di berikan theodore et al. yang berfokus pada *Time-wait* dan cara mengurangi *TIME-WAIT loading* .

#### 1. Melakukan modifikasi pada algoritma *Time-Wait* pada saat *three-way handshake*

a. *Client includes the TW-Negotiate option in the <SYN> packet for the connection. TW Negotiate contains the IP address of the end that will hold the TCB. The option's presence indicates that the client supports negotiation. Clients must send an IP address, to support the algorithm below for resolving a simultaneous open.*

b. *Server returns the <SYN, ACK> packet with TW-Negotiate set to its choice to keep The TIME-WAIT state. If it does not support negotiation, it sends no TW-Negotiate option.*

c. *The client decides if the server's choice is acceptable. If so, it acknowledges the <SYN, ACK> packet with the same value of TW-Negotiate. If not it aborts the connection with an <RST> packet. (The connection is aborted as though it failed to synchronize, and introduces no new failure modes to TCP.) Aborting the connection from this unsynchronized condition leaves no extra state at either endpoint; the server returns to LISTEN, and the client closes the connection. If the server returned no TW-Negotiate option, the connection will use current TCP semantics: the side that issues the active close will enter TIME-WAIT (or both will if they close simultaneously).*

Apabila algoritma ini diterapkan akan mengurangi *server load* .

2. Mengurangi *Time-wait loading* di HTTP dan FTP Server
  - Memodifikasi TCP status Time-wait antara server dan *client* ketika koneksi ditutup, TCP *client* dimodifikasi sehingga setelah koneksi ditutup *client* akan mengirimkan <RST> paket ke server dan mengubah statusnya sendiri ke *TIME-WAIT*. <RST> paket akan menghapus status *TIME-WAIT* pada TCB di server.
3. Memodifikasi HTTP sehingga akan mengurangi kontribusi dari Protocol ke *TIME-WAIT loading*.
  - Modifikasi HTTP dilakukan karena HTTP merupakan sumber utama dari trafik internet pada *client/server*.
  - Modifikasi HTTP 1.1 termasuk notifikasi dari *client* ketika koneksi tertutup. Notifikasi ini berupa *extension request*, bernama *CLIENT\_CLOSE*.
  - *CLIENT\_CLOSE* tidak membutuhkan jawaban, *client* akan menutup *TCP connection* setelah mengirim *CLIENT\_CLOSE* ke server.
  - Modifikasi pada HTTP ini adalah solusi yang efektif untuk mengurangi *TIME-WAIT loading* akan tetapi memiliki banyak kendala, Server membutuhkan suatu mekanisme untuk berkomunikasi dengan client sehingga server dan client akan menghentikan koneksi yang sama.

H. Balakrishnan et al. berfokus pada meningkatkan performa pada *single connection* dan *multiple TCP connection* untuk meminimalisir *request time out* ke server ketika terjadi *congestion* dan *bottleneck*.

1. *Single connection performance*

- 1.1 *Enhance Loss Recovery*

Menggunakan algoritma *right-edge recovery*.

Ketika algoritma *right-edge recovery* digunakan, *sender* akan mengirimkan paket baru yang merespon setiap *duplicate acknowledgments*. Hal ini akan membuat *receiver* akan men *generate* tambahan *duplicate acknowledgments*, yang kemudian akan men *trigger fast retransmission* pada *sender*. Dengan mengeliminasi *coarse timeout*, *sender* yang menggunakan *right-edge recovery* akan menghasilkan performa transfer dua kali lebih cepat daripada standar *transmitter*.

2. *Multiple TCP Connection*

- 2.1 *Integrated Congestion Control/Loss Recovery*

Metode *Integrated Congestion Control/Loss Recovery* membuat aplikasi untuk menggunakan koneksi TCP terpisah untuk setiap transfer dengan menambah modifikasi pada *network stack*. Metode ini membagi fungsi TCP ke dalam dua kategori, yang pertama berhubungan dengan *reliable, ordered byte-stream abstraction of TCP* dan berkaitan dengan *congestion control and data-driven loss recovery*. *Congestion control and data-driven loss recovery* dilakukan terintegrasi secara paralel pada suatu set koneksi. Metode ini disebut jg *TPC-Int*

3. Solusi *Alternatif: Application-level Multiplexing*  
Application-level multiplexing menghindari penggunaan multiple parallel TCP connections dan masalahnya dengan melakukan *multiplexing* pada beberapa data streams ke dalam satu *TCP connection*. Solusi ini berhasil mengurangi *latency* secara signifikan ketika mengakses web saat menggunakan P-HTTP.
4. Meningkatkan *default socket buffer size* untuk mengurangi terjadinya *bottleneck*.

### References

1. Statistical time-division multiplexing – Wikipedia (2019, 26-10-2008)  
[https://en.wikipedia.org/wiki/Statistical\\_time-division\\_multiplexing](https://en.wikipedia.org/wiki/Statistical_time-division_multiplexing)
2. H. Balakrishnan et.al (1998). TCP Behavior of a Busy Internet Server: Analysis and Improvements. *IEEE Infocom*
3. N. Baharudin et.al (2013). Review on Multiplexing Techniques in Bandwidth Utilization. *Middle-East Journal of Scientific Research* 18 (10): 1510-1516, 2013
4. F. Theodore et.al. the TIME-WAIT state in TCP and Its Effect on Busy Servers.