



**MANAJEMEN *BANDWIDTH* DENGAN MIKROTIK PADA  
DINAS PENDIDIKAN SURABAYA**



UNIVERSITAS  
**Dinamika**

**Oleh :**

**NURULLAH YULI SAPRIYANTO**

**14.41020.0060**

---

**FAKULTAS TEKNOLOGI DAN INFORMATIKA  
UNIVERSITAS DINAMIKA 2020**

**MANAJEMEN *BANDWIDTH* DENGAN MIKROTIK PADA  
DINAS PENDIDIKAN SURABAYA**

Diajukan sebagai salah satu syarat untuk menempuh ujian Tahap Akhir

Program Strata Satu (S1)



Disusun Oleh :

**Nama** : Nurullah Yuli Sapriyanto

**Nim** : 14.41020.0060

**Program** : S1 (Strata Satu)

**Jurusan** : Teknik Komputer

**FAKULTAS TEKNOLOGI DAN INFORMATIKA**

**UNIVERSITAS DINAMIKA**

**2020**

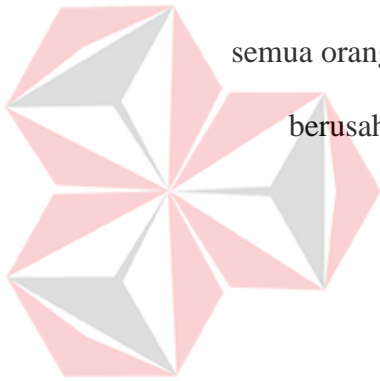
*“Uang bukan segalanya, tetapi segalanya butuh uang.”*

**Nurullah Yuli Sapriyanto**



UNIVERSITAS  
**Dinamika**

Dipersembahkan kepada Bapak, Ibu dan Keluarga saya yang selalu mendukung, memotivasi dan memberi doa yang terbaik kepada saya, Beserta semua orang yang selalu membantu, mendukung dan memotivasi agar tetap berusaha, belajar, berdoa agar menjadi lebih baik dari sebelumnya.



UNIVERSITAS  
**Dinamika**

**LEMBAR PENGESAHAN  
LAPORAN KERJA PRAKTIK**

**MANAJEMEN *BANDWIDTH* DENGAN MIKROTIK PADA DINAS  
PENDIDIKAN SURABAYA**

Laporan Kerja Praktik oleh  
**NURULLAH YULI SAPRIYANTO**  
**NIM : 14.41020.0060**

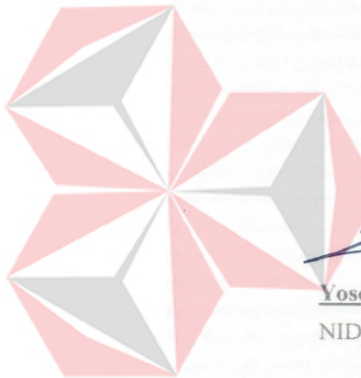
Telah diperiksa, diuji dan disetujui


Surabaya, 20 Juli 2020

Disetujui :

Pembimbing

Penyelia



  
**Yosefine Trividyastuti, M.T.**  
NIDN. 0729038504

  
**Parivem**  
NIP. 005934



Mengetahui :

Ketua Prodi S1 Teknik Komputer

Digitally signed by  
Pauladie Susanto  
Date: 2020.07.26  
09:54:03 +07'00'

**Pauladie Susanto, S.Kom., M.T.**

NIDN 0729047501

**SURAT PERNYATAAN**  
**PERSETUJUAN PUBLIKASI DAN KEASLIAN KARYA ILMIAH**

Sebagai mahasiswa Institut Bisnis dan Informatika Stikom Surabaya, saya :

Nama : Nurullah Yuli Sapriyanto  
NIM : 14.41020.0060  
Program Studi : S1 Teknik Komputer  
Fakultas : Fakultas Teknologi dan Informatika  
Jenis Karya : Laporan Kerja Praktik  
Judul Karya : **MANAJEMEN BANDWIDTH DENGAN MIKROTIK  
PADA DINAS PENDIDIKAN SURABAYA**

Menyatakan dengan sesungguhnya bahwa:

1. Demi pengembangan Ilmu Pengetahuan, Teknologi dan Seni, saya menyetujui memberikan kepada Institut Bisnis dan Informatika Stikom Surabaya Hak Bebas Royalti Non-Eksklusif (*Non-Exclusive Royalti Free Right*) atas seluruh isi/ sebagian karya ilmiah saya tersebut di atas untuk disimpan, dialihmediakan dan dikelola dalam bentuk pangkalan data (*database*) untuk selanjutnya didistribusikan atau dipublikasikan demi kepentingan akademis dengan tetap mencantumkan nama saya sebagai penulis atau pencipta dan sebagai pemilik Hak Cipta.
2. Karya tersebut di atas adalah karya asli saya, bukan plagiat baik sebagian maupun keseluruhan. Kutipan, karya atau pendapat orang lain yang ada dalam karya ilmiah ini adalah semata hanya rujukan yang dicantumkan dalam Daftar Pustaka saya.
3. Apabila dikemudian hari ditemukan dan terbukti terdapat tindakan plagiat pada karya ilmiah ini, maka saya bersedia untuk menerima pencabutan terhadap gelar keserjanaan yang telah diberikan kepada saya.

Demikian surat pernyataan ini saya buat dengan sebenarnya.

Surabaya, 20 Juli 2020

Yang menyatakan



NIM : 14.41020.0060

## ABSTRAK

Perkembangan teknologi saat ini sangatlah cepat, terutama dibidang jaringan komputer, seiring berkembangnya jaringan internet sekarang dengan mudahnya kita mendapatkan internet gratis, hampir semua institusi maupun lembaga organisasi dan pendidikan menggunakan layanan internet ini, karena internet bukanlah termasuk hal yang mewah lagi, dan bahkan saat ini sudah menjadi kebutuhan pokok.

*Router* mikrotik memiliki fitur *queue* yang dapat melakukan pengaturan alokasi *bandwidth* untuk setiap *user*. Dengan melakukan manajemen *bandwidth*, maka telah dilakukan usaha perbaikan terhadap *Quality of Service* (kualitas jaringan). *Quality of Service* akan memberikan jaminan alokasi *Bandwidth* minimum pada setiap *user* di dalam jaringan, sehingga setiap *user* tidak perlu khawatir akan tidak mendapatkan jatah *Bandwidth*.

Dikarenakan hal tersebut, maka pembagian atau manajemen *bandwidth* sangat diperlukan agar setiap *user* menikmati jaringan internet secara merata tanpa ada yang mendominasi dan dapat dilakukan pembagian *bandwidth* yang sesuai dengan prioritas dari setiap *user*. Manajemen *bandwidth* menggunakan mikrotik juga memiliki sistem keamanan yang lebih tinggi jika dibandingkan dengan perangkat *Router* yang lainnya.

**Kata Kunci:** *Bandwidth, Queue Tree, Quality of Service, Peer Connection Queue, Mikrotik, Delay, Jitter, Throughput.*

## KATA PENGANTAR



*Assalamu'alaikum Wr. Wb.*

Segala puji dan syukur penulis panjatkan kehadirat Allah SWT atas rahmat dan hidayah-Nya. Shalawat seiring salam semoga tercurah kepada junjungan kita Nabi besar Muhammad SAW, keluarga, sahabat, dan para pengikutnya sehingga penulis dapat penyusunan Laporan Kerja Praktik yang berjudul : Manajemen *Bandwidth* Dengan Mikrotik Pada Dinas Pendidikan Surabaya.

Penulis menyadari bahwa banyak pihak yang telah membantu, memberikan bimbingan dan motivasi kepada penulis dalam menyelesaikan penulisan makalah ini. Untuk itu, iringan do'a dan ucapan terima kasih penulis sampaikan kepada:

1. Allah SWT, karena dengan rahmatnya dan hidayahnya penulis dapat menyelesaikan Laporan Kerja Praktik ini.
2. Orang Tua dan Seluruh Keluarga penulis tercinta yang telah memberikan dorongan dan bantuan baik moral maupun materi sehingga penulis dapat menempuh dan menyelesaikan Kerja Praktik serta Laporan ini.
3. Dinas Pendidikan Kota Surabaya atas segala kesempatan dan pengalaman kerja yang telah diberikan kepada penulis selama melaksanakan Kerja Praktik.
4. Kepada Bapak Pariyem selaku Penyelia. Terima kasih atas bimbingan yang diberikan sehingga penulis dapat melaksanakan Kerja Praktik di Dinas Pendidikan Kota Surabaya.



5. Kepada Bapak Pauladie Susanto, S.Kom., M.T. selaku Ketua Program Studi Teknik Komputer Surabaya sekaligus Ibu Yosefine Triwidyastuti, M.T. selaku dosen pembimbing atas ijin yang diberikan untuk melaksanakan Kerja Praktik di Dinas Pendidikan Kota Surabaya.

Tiada kata yang pantas penulis ucapkan selain do'a Semoga Allah memberikan balasan yang setimpal atas bantuan dan pemikirannya. Akhir kata penulis berharap buku laporan KP ini bermanfaat dan dapat menjadi inspirasi bagi pembaca pada umumnya serta menambah khasanah ilmu pengetahuan. Amin

*Wassalamu'alaikum Wr.Wb*



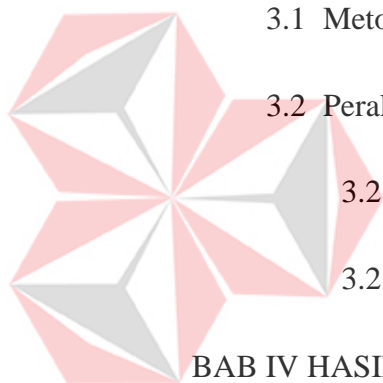
UNIVERSITAS  
**Dinamika**  
Surabaya, 20 Juli 2020

Penulis

## DAFTAR ISI

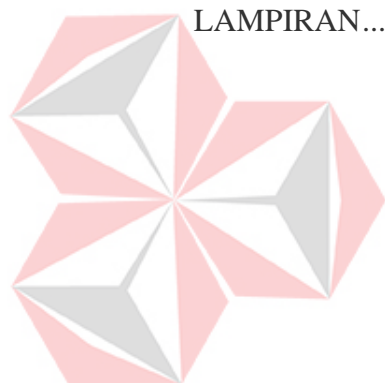
	Halaman
ABSTRAK.....	v
KATA PENGANTAR .....	vi
DAFTAR ISI.....	viii
DAFTAR GAMBAR .....	xi
DAFTAR TABEL.....	xiv
DAFTAR LAMPIRAN.....	xv
<b>BAB I PENDAHULUAN.....</b>	<b>2</b>
1.1. Latar Belakang Masalah .....	2
1.2. Perumusan Masalah.....	2
1.3. Batasan Masalah.....	3
1.4. Tujuan.....	3
1.5. Kontribusi .....	3
<b>BAB II LANDASAN TEORI.....</b>	<b>4</b>
2.1 Jaringan Komputer .....	4
2.2 Internet.....	5
2.3 <i>Bandwidth</i> .....	6
2.4 Manajemen <i>Bandwidth</i> .....	8
2.5 <i>Queue Tree</i> .....	9
2.6 <i>Peer Connection Queue</i> .....	9

2.7	<i>Quality of Service</i>	10
2.8	<i>Jitter</i>	11
2.9	<i>Throughput</i>	11
2.10	<i>IP Address</i>	12
2.11	<i>DHCP</i>	14
2.12	<i>Router Mikrotik</i>	16
2.13	<i>Winbox</i>	17
<b>BAB III METODOLOGI PENELITIAN</b>		19
3.1	Metode Pengumpulan Data	19
3.2	Peralatan Penelitian	19
3.2.1	Perangkat Keras	19
3.2.2	Perangkat Lunak	20
<b>BAB IV HASIL ANALISA</b>		22
4.1	Konfigurasi dengan Winbox	22
4.2	<i>Setting DHCP Client</i>	28
4.3	<i>Setting DHCP Server</i>	30
4.4	<i>Setting DHCP Server Untuk Wireless</i>	30
4.5	<i>Setting NAT</i>	32
4.6	<i>Sharing Koneksi Via Wireless</i>	34
4.7	<i>Setting Queue Tree dan PCQ</i>	37
4.8	<i>Setting Limit Alokasi Bandwidth di Jaringan Wireless</i>	45



UNIVERSITAS  
Dinamika

4.9 Pengujian Menggunakan <i>Wireshark</i> .....	54
4.9.1 <i>Delay</i> .....	57
4.9.2 <i>Jitter</i> .....	59
4.9.3 <i>Throughput</i> .....	62
BAB V PENUTUP .....	64
5.1 Kesimpulan.....	64
5.2 Saran .....	65
DAFTAR PUSTAKA .....	63
LAMPIRAN.....	64



UNIVERSITAS  
**Dinamika**

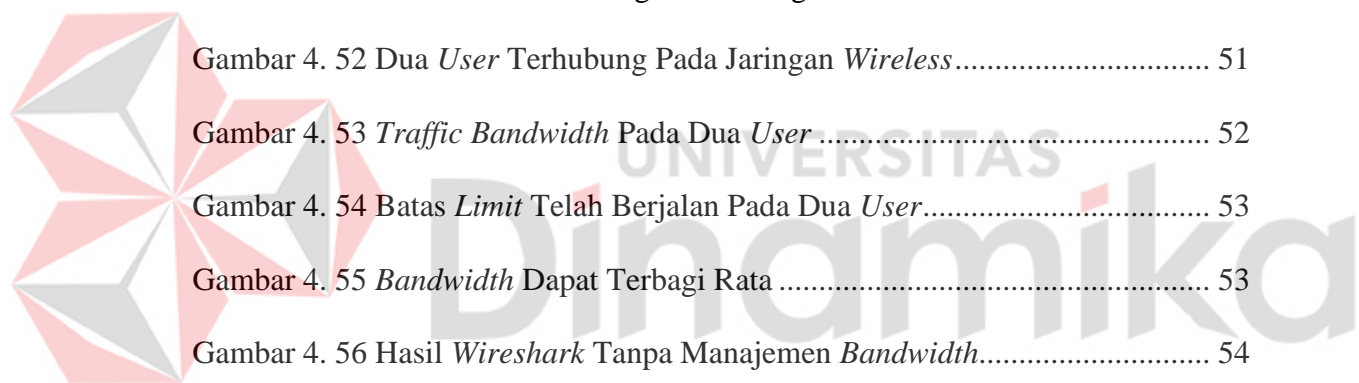
## DAFTAR GAMBAR

Halaman

Gambar 2. 1 Jaringan Komputer .....	4
Gambar 2. 2 Router Mikrotik.....	17
Gambar 2. 3 Konfigurasi Awal Winbox .....	18
Gambar 3. 1 Topologi Simulasi Manajemen <i>Bandwidth</i> .....	21
Gambar 4. 1 Jendela Awal <i>Winbox</i> .....	22
Gambar 4. 2 Konfigurasi <i>Interface BridgeWired</i> .....	23
Gambar 4. 3 Konfigurasi <i>Ether2</i> .....	24
Gambar 4. 4 Konfigurasi <i>Ether3</i> .....	24
Gambar 4. 5 Konfigurasi <i>Ether4</i> .....	25
Gambar 4. 6 Konfigurasi <i>BridgeWlan</i> .....	25
Gambar 4. 7 Konfigurasi <i>Port Wlan1</i> .....	26
Gambar 4. 8 Hasil Konfigurasi Semua <i>Port</i> .....	26
Gambar 4. 9 Konfigurasi <i>IP Address BridgeWired</i> .....	27
Gambar 4. 10 Konfigurasi <i>IP Address BridgeWlan</i> .....	28
Gambar 4. 11 Hasil Konfigurasi Semua <i>IP Address</i> .....	28
Gambar 4. 12 Konfigurasi <i>DHCP Client</i> .....	29
Gambar 4. 13 Hasil Konfigurasi <i>DHCP Client</i> .....	29
Gambar 4. 14 Konfigurasi <i>DHCP Server Wired</i> .....	30
Gambar 4. 15 Konfigurasi <i>DHCP Server Wlan</i> .....	31
Gambar 4. 16 Hasil Konfigurasi <i>DHCP Server Wired dan Wlan</i> .....	31
Gambar 4. 17 Konfigurasi NAT pada <i>ether1</i> .....	32

Gambar 4. 18 Konfigurasi NAT pada <i>Tab Action</i> .....	32
Gambar 4. 19 Hasil Konfigurasi NAT .....	33
Gambar 4. 20 Cek Koneksi Internet Melalui <i>New Terminal</i> .....	33
Gambar 4. 21 Cek Koneksi Internet Melalui <i>CMD</i> Pada PC.....	34
Gambar 4. 22 Mengaktifkan <i>Interface Wlan1</i> .....	34
Gambar 4. 23 Membuat Jaringan Untuk Akses <i>Wireless</i> .....	35
Gambar 4. 24 Membuat <i>Password Wifi</i> .....	36
Gambar 4. 25 Konfigurasi <i>Password</i> kedalam <i>Interface Wlan1</i> .....	36
Gambar 4. 26 Tampilan Koneksi <i>Wireless SSID</i> dan <i>Password</i> .....	37
Gambar 4. 27 Konfigurasi <i>Mangle</i> pada <i>Ether1</i> .....	38
Gambar 4. 28 Konfigurasi <i>Action</i> dengan <i>Koneksi Lokal Wired</i> .....	38
Gambar 4. 29 Konfigurasi <i>Mangle</i> dengan <i>Koneksi Lokal Wired</i> .....	39
Gambar 4. 30 Konfigurasi <i>Action</i> dengan <i>Mark Packet</i> .....	39
Gambar 4. 31 Konfigurasi <i>Queue</i> untuk <i>Upload</i> .....	40
Gambar 4. 32 Konfigurasi <i>Queue</i> untuk <i>Download</i> .....	40
Gambar 4. 33 Hasil Konfigurasi <i>Queue</i> .....	41
Gambar 4. 34 Pengujian Koneksi <i>Wired</i> Sebelum Manajemen .....	41
Gambar 4. 35 Konfigurasi <i>Queue Tree</i> pada <i>Ether1</i> .....	42
Gambar 4. 36 Konfigurasi <i>Queue Tree</i> pada <i>Parent Upload</i> .....	43
Gambar 4. 37 Konfigurasi <i>Queue Tree</i> untuk <i>Download</i> .....	43
Gambar 4. 38 Konfigurasi <i>Queue Tree</i> dengan <i>Parent Download</i> .....	44
Gambar 4. 39 Pengujian Koneksi <i>Wired</i> Setelah Manajemen .....	44
Gambar 4. 40 Konfigurasi <i>Limit Bandwidth</i> pada <i>Ether1</i> .....	45
Gambar 4. 41 Konfigurasi <i>Mark Connection</i> pada <i>Koneksi Lokal Wlan</i> .....	46

Gambar 4. 42 Konfigurasi <i>Mark Packet</i> pada <i>KoneksiLokalWlan</i> .....	46
Gambar 4. 43 Hasil Konfigurasi <i>KoneksiLokalWlan</i> .....	47
Gambar 4. 44 Menghubungkan <i>Wireless</i> dari PC.....	47
Gambar 4. 45 Pengujian Koneksi <i>Wireless</i> Sebelum Manajemen.....	48
Gambar 4. 46 Konfigurasi <i>Queues</i> untuk <i>UploadWlan</i> .....	48
Gambar 4. 47 Konfigurasi <i>Queues</i> dengan <i>Parent UploadWlan</i> .....	49
Gambar 4. 48 Konfigurasi <i>Queues</i> untuk <i>DownloadWlan</i> .....	49
Gambar 4. 49 Konfigurasi <i>Queues</i> dengan <i>Parent DownloadWlan</i> .....	50
Gambar 4. 50 Pengujian Koneksi <i>Wireless</i> Setelah Manajemen .....	50
Gambar 4. 51 Satu <i>User</i> Terhubung Pada Jaringan <i>Wireless</i> .....	51
Gambar 4. 52 Dua <i>User</i> Terhubung Pada Jaringan <i>Wireless</i> .....	51
Gambar 4. 53 <i>Traffic Bandwidth</i> Pada Dua <i>User</i> .....	52
Gambar 4. 54 Batas <i>Limit</i> Telah Berjalan Pada Dua <i>User</i> .....	53
Gambar 4. 55 <i>Bandwidth</i> Dapat Terbagi Rata .....	53
Gambar 4. 56 Hasil <i>Wireshark</i> Tanpa Manajemen <i>Bandwidth</i> .....	54
Gambar 4. 57 Hasil <i>Capture Wireshark</i> Tanpa Manajemen <i>Bandwidth</i> .....	55
Gambar 4. 58 Hasil <i>Wireshark</i> dengan Manajemen <i>Bandwidth</i> .....	56
Gambar 4. 59 Hasil <i>Capture Wireshark</i> dengan Manajemen <i>Bandwidth</i> .....	56



## DAFTAR TABEL

	Halaman
Tabel 4. 1 Data <i>Transfer Rate</i> dari <i>Wireshark</i> .....	57
Tabel 4. 2 Perhitungan <i>Delay</i> sebelum manajemen .....	58
Tabel 4. 3 Perhitungan <i>Delay</i> dengan manajemen.....	58
Tabel 4. 4 Perhitungan Rata-rata <i>Jitter</i> sebelum Konfigurasi.....	60
Tabel 4. 5 Perhitungan <i>Jitter</i> Setelah Manajemen .....	61
Tabel 4. 6 perhitungan <i>Throughput</i> sebelum manajemen.....	62
Tabel 4. 7 hasil perhitungan <i>Throughput</i> dengan manajemen .....	63



UNIVERSITAS  
**Dinamika**



## DAFTAR LAMPIRAN

	Halaman
Lampiran 1 Form KP-3 (Surat Balasan Perusahaan) .....	64
Lampiran 2 Form KP-5 (Acuan Kerja) .....	66
Lampiran 3 Form KP-5 (Garis Besar Rencana Kerja Mingguan) .....	67
Lampiran 4 Form KP-6 (Log Harian dan Catatan Perubahan Acuan Kerja) .....	68
Lampiran 5 Form KP-7 (Kehadiran Kerja Praktik) .....	69
Lampiran 6 Kartu Bimbingan Kerja Praktik .....	70



UNIVERSITAS  
**Dinamika**

# BAB I

## PENDAHULUAN

### 1.1. Latar Belakang Masalah

Perkembangan teknologi saat ini sangatlah cepat, terutama dibidang jaringan komputer, seiring berkembangnya jaringan internet sekarang dengan mudahnya kita mendapatkan internet gratis, hampir semua institusi maupun lembaga organisasi dan pendidikan menggunakan layanan internet ini, karena internet bukanlah termasuk hal yang mewah lagi, dan bahkan saat ini sudah menjadi kebutuhan pokok.

Besarnya *bandwidth* mempengaruhi kecepatan koneksi internet, sehingga setiap *user* berharap untuk mendapatkan jatah *bandwidth* yang besar, dengan kapasitas yang besar itu diperlukan sebuah manajemen *bandwidth* yang baik, karena sering kali terjadi permasalahan koneksi internet yang lambat, sinyal wifi yang kadang hilang dan timbul, serta keluhan lainnya yang tidak dapat terkoneksi ke internet disebabkan ada yang memutuskan koneksinya dengan sengaja, maka berdasarkan permasalahan tersebut, perlu kiranya manajemen *bandwidth*, supaya semua jaringan komputer yang ada bisa dimonitoring dengan baik, dan supaya tidak adanya monopoli *bandwidth* yang menyebabkan komputer yang lain tidak mendapat jatah *bandwidth* yang adil. *Router* mikrotik memiliki fitur *queue* yang dapat melakukan pengaturan alokasi *bandwidth* untuk setiap *user*.

Dengan melakukan manajemen *bandwidth*, maka telah dilakukan usaha perbaikan terhadap *Quality of Service* (kualitas jaringan). *Quality of Service* akan memberikan jaminan alokasi *bandwidth* minimum pada setiap *user* di dalam jaringan, sehingga setiap *user* tidak perlu khawatir akan tidak mendapatkan jatah *bandwidth*. Pada proses manajemen *bandwidth* ini nantinya akan menggunakan metode PCQ (*Peer Connection Queue*), metode ini dipilih karena pada umumnya sebuah jaringan sering terjadi pembagian *bandwidth* yang tidak merata pada setiap *user* dalam suatu jaringan yang mengakibatkan satu *user* dapat mengakses internet lebih cepat, sedangkan satu *user* yang lainnya lebih lambat. Dengan menggunakan *simple queue* yang mempunyai sebuah metode PCQ (*Peer Connection Queue*) bekerja dengan sebuah algoritma yang akan membagi *bandwidth* secara merata ke sejumlah *client* yang aktif. PCQ (*Peer Connection Queue*) ideal diterapkan apabila dalam pengaturan *bandwidth* kesulitan dalam penentuan *bandwidth* per *client*.

## 1.2. Perumusan Masalah

Berdasarkan uraian dari latar belakang diatas maka dapat kita rumuskan beberapa permasalahan yang ada, yakni :

1. Bagaimana mengimplementasikan metode PCQ menggunakan *Queue Tree* untuk mengoptimalkan *bandwidth*?
2. Bagaimana hasil dari pengujian parameter QoS (*Quality of Service*) sebelum dan setelah penerapan metode PCQ (*Peer Connection Queue*) menggunakan *Queue Tree* ?

### 1.3. Batasan Masalah

Melihat permasalahan yang ada, maka penulis membatasi masalah dari bahasan yang lebih luas terhadap manajemen *Bandwidth*, yaitu:

1. Manajemen *bandwidth* menggunakan *Queue Tree* dengan metode *Peer Connection Queue*.
2. Menggunakan *software Winbox* sebagai konfigurasi mikrotik.
3. Jaringan yang dipakai menggunakan *Local Area Network (LAN)* dan *Wireless*.

### 1.4. Tujuan

Tujuan umum dari kegiatan Kerja Praktik yang dilaksanakan mahasiswa adalah agar mahasiswa dapat melihat serta berlatih pada kondisi dan keadaan nyata yang ada pada dunia kerja sehingga mendapatkan pengalaman yang lebih banyak dan dapat memperdalam kemampuan pada suatu bidang. Tujuan khusus adalah sebagai berikut:

1. Mengimplementasikan manajemen *bandwidth* menggunakan *Queue Tree* dengan metode *Peer Connection Queue*.
2. Memberikan *Quality of Service* yang baik dalam sebuah jaringan internet.

### 1.5. Kontribusi

Memberikan kontribusi ke Dinas Pendidikan Kota Surabaya dalam memperbaiki dan mengoptimisasi manajemen *bandwidth* pada ruang lingkup perkantoran, sehingga tercipta suasana kerja yang nyaman tanpa hambatan koneksi internet.

## BAB II

### LANDASAN TEORI

#### 2.1 Jaringan Komputer

Jaringan komputer adalah jaringan telekomunikasi yang memungkinkan antar komputer untuk saling menukar data. Tujuan dari jaringan komputer ialah agar bisa mencapai tujuannya, pada bagian dari setiap jaringan komputer bisa memberikan serta meminta layanan (*service*). Pihak yang menerima/meminta layanan disebut dengan klien (*Client*) serta yang melakukan pengiriman/yang memerikan layanan disebut dengan peladen (*Server*). Desain tersebut bernama sistem *Client-Server*, serta dipakai pada hampir semua aplikasi jejaring komputer.



Gambar 2. 1 Jaringan Komputer

Dua buah komputer, masing-masing mempunyai kartu jaringan, lalu dihubungkan dengan kabel ataupun nirkabel untuk medium transmisi data, serta ada perangkat lunak sistem operasi jaringan akan membentuk sebuah jaringan komputer yang sederhana. Jika ingin membuat jaringan komputer yang lebih luas

jangkauanya, maka diperlukan sebuah peralatan tambahan seperti, *Gateway, Hub, Switch, Bridge, Router* untuk peralatan interkoneksinya.

## 2.2 Internet

Internet adalah singkatan dari *Interconnected Networking* yang apabila diartikan dalam Bahasa Indonesia berarti rangkaian komputer yang terhubung di dalam beberapa rangkaian jaringan. Internet merupakan salah satu hasil dari kecanggihan dan kemajuan ilmu pengetahuan dan teknologi buatan manusia. Internet merupakan sebuah sebutan untuk sekumpulan jaringan komputer yang dapat menghubungkan berbagai situs akademik, pemerintahan, komersial, organisasi, hingga perorangan. Lebih lanjut dijelaskan bahwa internet mampu untuk menyediakan akses untuk layanan telekomunikasi dan berbagai sumber daya informasi untuk jutaan pemakaiannya yang tersebar di seluruh dunia. Internet memiliki berbagai macam layanan-layanan internet meliputi komunikasi secara langsung seperti *email* dan juga *chatting*, diskusi seperti *Usenet News*, email dan juga milis serta sumber daya informasi yang terdistribusi (*World Wide Web, Gopher*), *remote Login*, dan lalu lintas *file (Telnet, FTP)*, dan lain-lainnya (Rahmadi, 2017).

Internet adalah seluruh jaringan yang saling terhubung satu sama lain. Beberapa komputer- komputer dalam jaringan ini menyimpan *file*, seperti halaman *web*, yang dapat diakses oleh seluruh jaringan komputer (Strauss, 2006).

Pengertian internet secara umum (menurut bahasa) adalah kumpulan dari jaringan komputer yang terhubung dan bekerja sebagai suatu sistem. Sedangkan pengertian Internet secara khusus adalah suatu jaringan komputer terbesar di dunia

karena menghubungkan seluruh jaringan komputer yang ada di dunia ini. Sedangkan Jaringan adalah cara untuk menghubungkan beberapa komputer sehingga setiap komputer yang ada di dalamnya bisa saling berhubungan dan berbagi sumber daya.

Beberapa layanan populer di Internet yang menggunakan protokol di atas, ialah email/surat elektronik, *Usenet*, *Newsgroup*, berbagi berkas (*File Sharing*), *WWW (World Wide Web)*, *Gopher*, akses sesi (*Session Access*), *WAIS*, *finger*, *IRC*, *MUD*, dan *MUSH*. Di antara semua ini, email/surat elektronik dan *World Wide Web* lebih kerap digunakan, dan lebih banyak servis yang dibangun berdasarkannya, seperti milis (*Mailing List*) dan *Weblog*. Internet memungkinkan adanya servis terkini (*Real-time service*), seperti *web radio*, dan *webcast*, yang dapat diakses di seluruh dunia. Selain itu melalui Internet dimungkinkan untuk berkomunikasi secara langsung antara dua pengguna atau lebih melalui program pengirim pesan instan seperti *Camfrog*, *Pidgin (Gaim)*, *Trilian*, *Kopete*, *Yahoo! Messenger*, *MSN Messenger* dan *Windows Live Messenger*.

### 2.3 *Bandwidth*

*Bandwidth* adalah suatu nilai konsumsi transfer data yang dihitung dalam *bit/detik* atau yang biasanya di sebut dengan *bit per second (bps)*, antara *Server* dan *Client* dalam waktu tertentu. Atau bisa didefinisikan sebagai lebar cakupan frekuensi yang dipakai oleh sinyal dalam medium transmisi (Situmorang, 2019).

*Bandwidth* juga dapat diartikan sebuah kapasitas yang dapat digunakan pada kabel *ethernet* agar dapat dilewati *trafik* paket data dengan maksimal tertentu. Pengertian lain dari *bandwidth* internet adalah jumlah konsumsi transfer data yang

dihitung dalam satuan waktu *bit per second* (bps). Sehingga secara ringkas dapat diartikan bahwa *bandwidth* internet merupakan kapasitas maksimal jalur komunikasi untuk melakukan proses pengiriman dan penerimaan data dalam hitungan detik.

Terdapat juga istilah *bandwidth* analog. *Bandwidth* analog adalah perbedaan antara frekuensi paling rendah dan frekuensi paling tinggi pada suatu rentang frekuensi yang dapat diukur dengan satuan *Hertz* (Hz) untuk mengetahui data atau informasi yang dapat ditransmisikan pada suatu waktu. *Bandwidth* sering dianalogikan dengan lebar jalan raya, sedangkan data yang masuk melewati *bandwidth* diibaratkan kendaraan yang melintasi jalan tersebut. Semakin sedikit kendaraan yang lewat maka lalu lintas akan semakin lancar. Kebalikannya, jika kendaraan yang lewat banyak maka lalu lintas di jalan tersebut akan tersendat sehingga akan mempengaruhi aktivitas kendaraan lain.

Semakin besar jalan (*bandwidth*) maka akan semakin banyak pula kendaraan yang dapat melaluinya. Maka tidak salah jika *bandwidth* menjadi pertimbangan pengguna jaringan internet. Dikarenakan semakin besar *bandwidth* maka semakin cepat pertukaran data yang terjadi dan semakin banyak data yang dapat melaluinya dalam satu waktu. Menurut (Mujiono, 2012) *bandwidth* dibagi menjadi 2 yaitu *bandwidth* analog dan *bandwidth* digital.

#### A. *Bandwidth* Analog

*Bandwidth* analog merupakan perbedaan antara frekuensi terendah dan frekuensi tertinggi dalam sebuah rentang frekuensi yang diukur dalam satuan Hz (*hertz*) yang dapat menentukan banyaknya informasi yang dapat ditransmisikan dalam suatu saat.



## B. *Bandwidth* Digital

*Bandwidth* digital merupakan jumlah atau *volume* suatu data (dalam satuan *bit* per detik/bps) yang dapat dikirimkan melalui sebuah saluran komunikasi tanpa adanya distorsi.

### 2.4 Manajemen *Bandwidth*

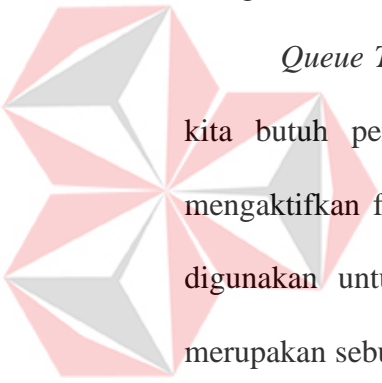
Management *bandwidth* merupakan suatu alat yang dapat digunakan untuk management dan mengoptimalkan berbagai jenis jaringan dengan menerapkan layanan *Quality of Service* (QoS) untuk menetapkan tipe-tipe lalu lintas jaringan. Sedangkan QoS adalah kemampuan untuk menggambarkan suatu tingkatan pencapaian didalam suatu sistem komunikasi data. Ada juga yang mengartikan bahwa manajemen *bandwidth* adalah suatu pengalokasian yang tepat dari suatu *Bandwidth* untuk mendukung kebutuhan atau keperluan aplikasi atau suatu layanan jaringan. Pengalokasian *bandwidth* yang tepat dapat menjadi salah satu metode dalam memberikan jaminan kualitas suatu layanan jaringan QoS (*Quality of Services*).

Sebuah proses mengukur dan mengontrol komunikasi (lalu lintas, paket) pada link jaringan, untuk menghindari mengisi link untuk kapasitas atau overflowing link, yang akan mengakibatkan kemacetan jaringan dan kinerja yang buruk dapat juga diartikan sebagai manajemen *bandwidth*. Selain itu maksud dari manajemen *bandwidth* ini adalah bagaimana kita menerapkan pengalokasian atau pengaturan *bandwidth* dengan menggunakan sebuah PC *Router* Mikrotik. Manajemen *bandwidth* memberikan kemampuan untuk mengatur *bandwidth* jaringan dan

memberikan level layanan sesuai dengan kebutuhan dan prioritas sesuai dengan permintaan pelanggan.

## 2.5 *Queue Tree*

*Queue Tree* adalah salah satu fitur yang terdapat dalam mikrotik yang digunakan untuk mengatur jumlah *bandwidth*. Berfungsi untuk mengimplementasikan fungsi yang lebih kompleks dalam *limit bandwidth*. Biasanya digunakan oleh admin warnet untuk membatasi satu arah koneksi untuk *Download* maupun untuk upload. Cara membuat *Queue Tree* sering dibilang agak rumit oleh orang oleh karena itu orang lebih banyak memilih *simple queue* (Widayanto, 2018).



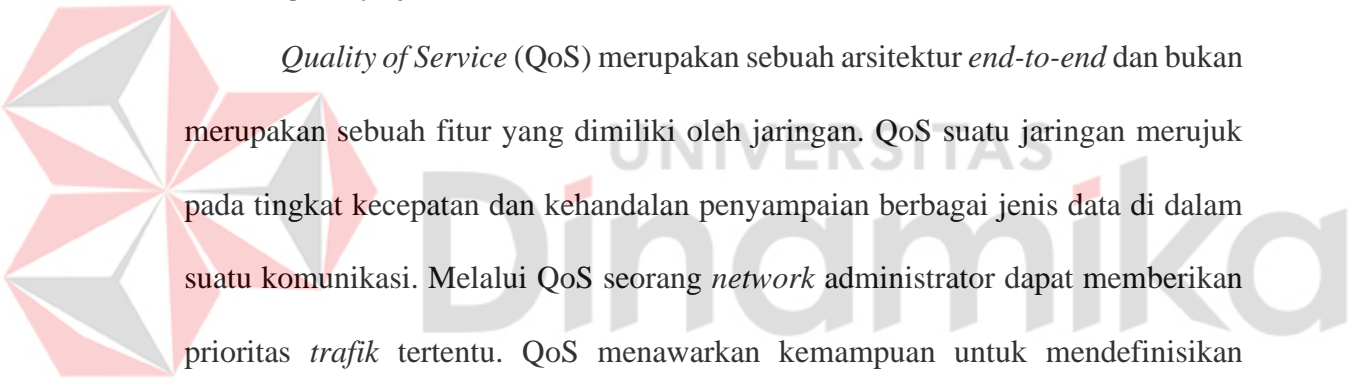
*Queue Tree* dirancang untuk menjalankan tugas yang lebih kompleks dan kita butuh pemahaman yang bagus tentang aliran *trafik* dan kita harus mengaktifkan fitur *mangle* pada *firewall*. Sedangkan *Simple Queue* kebanyakan digunakan untuk memudahkan konfigurasi. Secara keseluruhan *Queue Tree* merupakan sebuah konfigurasi *queue* yang bersifat *one way* (satu arah), ini berarti sebuah konfigurasi *queue* hanya akan mampu melakukan *queue* terhadap 1 (satu) arah jenis *traffic*. Jika sebuah konfigurasi *queue* pada *Queue Tree* ditujukan untuk melakukan *queue* terhadap *bandwidth download*, maka konfigurasi tersebut tidak dapat melakukan *queue* terhadap *bandwidth upload*, demikian sebaliknya.

## 2.6 *Peer Connection Queue*

*Peer Connection Queue* (PCQ) pada *queue type* adalah salah satu fitur dari MikroTik untuk membantu memanager *traffic rate* dan *traffic packet*. Dalam OS mikrotik, PCQ adalah program untuk mengelola jaringan Lalu Lintas Kualitas

Layanan (QoS). Tujuan utama dari metode ini adalah untuk melakukan *Bandwidth sharing* otomatis dan merata ke *multi Client*. Kerja prinsip PCQ dengan menerapkan *Simple Queue* atau *Queue Trees* dimana hanya ada satu klien aktif yang menggunakan *bandwidth*, sementara klien lain berada dalam posisi idle maka klien aktif tersebut dapat menggunakan *bandwidth* maksimum yang tersedia, tetapi jika klien lain aktif, maka *bandwidth* yang maksimal dapat digunakan oleh kedua klien (*bandwidth* atau jumlah klien yang aktif) sehingga *bandwidth* dapat terdistribusi secara adil untuk semua klien.

## 2.7 *Quality of Service*



*Quality of Service* (QoS) merupakan sebuah arsitektur *end-to-end* dan bukan merupakan sebuah fitur yang dimiliki oleh jaringan. QoS suatu jaringan merujuk pada tingkat kecepatan dan kehandalan penyampaian berbagai jenis data di dalam suatu komunikasi. Melalui QoS seorang *network administrator* dapat memberikan prioritas *trafik* tertentu. QoS menawarkan kemampuan untuk mendefinisikan atribut-atribut layanan yang disediakan, baik secara kualitatif maupun kuantitatif. Tujuan QoS menyediakan kualitas layanan yang berbeda-beda berdasarkan kebutuhan layanan di dalam jaringan. *Quality of Service* (QoS) atau kualitas layanan adalah metode pengukuran yang digunakan untuk menentukan kapabilitas jaringan, seperti aplikasi jaringan, *host*, atau *Router* untuk menyediakan layanan jaringan yang lebih baik dan lebih terencana yang memenuhi kebutuhan layanan.

## 2.8 *Jitter*

*Jitter* adalah variasi *delay* antara blok-blok yang berurutan yang nilainya sangat dipengaruhi oleh beban *trafik* dan besarnya tumbukan antar paket dalam sebuah jaringan. Semakin besar beban *trafik* dalam sebuah jaringan akan menyebabkan semakin besar peluang terjadinya tumbukan antar paket dalam jaringan tersebut dan hal ini menyebabkan nilai *Jitter*nya semakin besar. *Jitter* dapat dihitung menggunakan rumus berikut :

$$Jitter = \text{total variasi } delay / (\text{total paket data} - 1)$$

*Jitter* juga dapat diartikan sebagai variasi dari waktu kedatangan paket data. Dalam komunikasi digital dimana data yang dikirim dalam bentuk paket-paket yang memungkinkan pengirimannya akan dilakukan secara bersamaan namun paket-paket tersebut belum tentu sampai secara bersamaan karena melalui jalur yang berbeda. Perbedaan waktu sampai inilah yang dinamakan *Jitter* (Lisnawita, 2006).

## 2.9 *Throughput*

*Throughput* adalah kecepatan rata-rata data yang diterima oleh suatu node dalam selang waktu pengamatan tertentu. *Throughput* merupakan *Bandwidth* aktual saat itu juga dimana kita sedang melakukan koneksi. Satuan yang dipakai adalah bps (*bit persecond*). *Bandwidth* sering kali disebut dengan istilah *Throughput*. Walaupun kadang kala maknanya sama namun seringkali membawa pengertian yang berbeda, sesuai dengan situasi dan kondisinya. Sebagai contoh sebuah *network* 10 Mbps ternyata hanya dapat melayani aplikasi video dengan *Throughput* 2 Mbps. Sebab ada berbagai faktor yang mempengaruhi *network* tersebut, seperti sambungan kabel , perangkat, pengaturan aplikasi dan sebagainya. Dalam hal ini

*Throughput* berkaitan dengan “nilai *actual*” dari nilai *Bandwidth* riil yang dapat dicapai oleh suatu aplikasi. *Throughput* dapat dirumuskan sebagai berikut:

$$\textit{Throughput} = \textit{Transfer Size} / \textit{Transfer Time}$$

## 2.10 IP Address

*IP Address (Internet Protocol Address)* adalah nomor biner atau identitas numerik yang dipakai disetiap komputer agar komputer tersebut dapat saling berhubungan. *IP Address* terdiri dari 4 blok angka desimal dimana angka tersebut tidak boleh melebihi nilai 255. Sebenarnya komputer hanya mengirim dan menerima data dalam bentuk kode biner (hanya angka satu dan nol). *IP Address* yang terdapat dalam komputer juga merupakan sebuah kode biner yang di terjemahkan kedalam angka – angka.

Sebagai Contoh *IP Address* adalah :192.168.1.3 atau bisa juga dalam bentuk biner adalah : 11000000.10101000.00000001.00000011. *IP Address* memiliki 2 jenis yaitu IPv4 dan IPv6 . IPv4 atau IP versi 4 adalah IP yang memiliki panjang angka 32 bit sedangkan IPv6 atau IP versi 6 adalah IP yang memiliki panjang angka 128 bit. Terbentuknya IPv6 karena antisipasi dari melonjaknya pengguna internet dari hari ke hari untuk IPv4 bisa menampung *host* sebanyak 4.294.967.296 sedangkan IPv6 bisa menampung pangkat 4 dari IPv4.

IPv4 terbagi menjadi beberapa kelas,yaitu: Kelas A, Kelas B, Kelas C, Kelas D, dan Kelas E. Berikut adalah data pembagian kelas IP :

### 1. Kelas A:

Nilai oktet pertama: 1 – 126

Bagian untuk *Network Identifier* : W

Bagian untuk *Host Identifier* : X.Y.Z

Jumlah jaringan maksimum : 126

Jumlah *host* dalam satu jaringan maksimum : 16.777.214

2. Kelas B:

Nilai oktet pertama: 127 – 191

Bagian untuk *Network Identifier* : W.X

Bagian untuk *Host Identifier* : Y.Z

Jumlah jaringan maksimum : 16.384

Jumlah *host* dalam satu jaringan maksimum : 65.534

3. Kelas C:

Nilai oktet pertama: 192 – 223

Bagian untuk *Network Identifier* : W.X.Y

Bagian untuk *Host Identifier* : Z

Jumlah jaringan maksimum : 2,097,152

Jumlah *host* dalam satu jaringan maksimum : 254

4. Kelas D

Nilai oktet pertama: 224 – 239

Bagian untuk *Network Identifier* : *Multicast IP Address*

Bagian untuk *Host Identifier* : *Multicast IP Address*

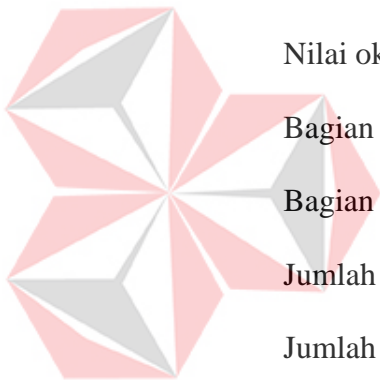
Jumlah jaringan maksimum : *Multicast IP Address*

Jumlah *host* dalam satu jaringan maksimum : *Multicast IP Address*

5. Kelas E

Nilai oktet pertama: 240 – 255

Bagian untuk *Network Identifier* : Dicadangkan; eksperimen



Bagian untuk *Host Identifier* : Dicapangkan; eksperimen

Jumlah jaringan maksimum : Dicapangkan; eksperimen

Jumlah *host* dalam satu jaringan maksimum : Dicapangkan; eksperimen

## 2.11 DHCP

DHCP (*Dynamic Host Configuration Protocol*) adalah sebuah protokol yang digunakan pada sebuah jaringan untuk memberikan konfigurasi *IP Address* kepada komputer yang membutuhkan secara otomatis. Konfigurasi *IP Address* yang diberikan kepada komputer agar dapat terhubung dengan internet terdiri dari *IP Address*, *subnet mask*, *default gateway* dan *DNS Sever*. DHCP terdiri dari 2 jenis, yaitu *DHCP Server* dan *DHCP Client*. *DHCP Server* adalah sebuah perangkat yang membagi konfigurasi *IP Address*. Sedangkan *DHCP Client* adalah sebuah seperangkat yang akan meminta *IP Address* pada *DHCP Server*. Router MikroTik dapat digunakan sebagai *DHCP Server*.

Tujuan dari penerapan DHCP ini adalah memudahkan setiap perangkat user mendapatkan sebuah *IP Address*. DHCP banyak diimplementasikan pada *hotspot*, baik *private* maupun *hotspot* untuk publik. Sehingga dalam penerapannya kita tidak harus memberikan IP secara manual kepada setiap *user* baru yang ingin menggunakan jaringan internet. Hal ini merupakan manfaat atau fungsi dari DHCP.

Pilihan yang harus dikonfigurasi untuk menggunakan DHCP adalah sebagai berikut:

1. *Network Address*; pilihan ini merupakan *network Address* beserta prefix (subnet mask) yang harus diberikan pada komputer pengguna.

2. *Gateway*; komputer pengguna harus memiliki *gateway* yang akan digunakan mengakses Internet.
3. *IP Address Pool*; pilihan ini berfungsi untuk menentukan jumlah *IP Address* yang akan diberikan kepada beberapa komputer pengguna.
4. *DNS Server*; pilihan pada DHCP yang digunakan oleh komputer *Client* untuk mengakses internet.
5. *Lease Time*; DHCP *Server* bekerja dengan teknik peminjaman *IP Address*. Komputer pengguna meminjam *IP Address* pada DHCP *Server*. Dalam proses peminjaman ini, DHCP *Server* akan memberikan batas waktu peminjaman *IP Address* yang ditentukan oleh pilihan *Lease Time* ini. Selama *Lease Time* dari sebuah *IP Address* belum habis, maka *IP Address* tersebut tidak akan dipinjamkan DHCP *Server* kepada komputer lain, sekalipun komputer pengguna yang meminjam pertama kali tidak lagi menggunakan *IP Address* tersebut. Jangan menggunakan *Lease Time* terlalu panjang jika pengguna jaringan sangat dinamis, karena dapat menyebabkan *IP Address Pool* akan kehabisan *IP Address* untuk dipinjamkan.

Sekalipun sudah menyediakan DHCP *Server* dan membatasi penggunaan *IP Address*, namun pengguna jaringan bisa saja jahil dan menggunakan konfigurasi *IP Address* yang dibuatnya sendiri. Misal DHCP *Server* hanya mempunyai *IP Pool* sebanyak 10 *IP Address* (192.168.1.1-192.168.1.10). Bila DHCP *Server* tidak diamankan, user dapat mengkonfigurasi *IP Address*nya sendiri, dan dapat terhubung ke Router MikroTik. Untuk mencegah hal tersebut, maka user dapat dipaksa untuk menggunakan DHCP *Server*. Dengan sistem pengamanan ini, jika



ada komputer yang berusaha untuk mengkonfigurasi *IP Address* secara statik, maka pengguna tersebut tidak akan terhubung ke Router MikroTik.

## 2.12 Router Mikrotik

Mikrotik SIA, atau yang lebih dikenal dengan MikroTik, adalah perusahaan pembuat peralatan jaringan komputer yang berbasis di Latvia. Perusahaan yang dibentuk pada tahun 1996 oleh John Tully dan Arnis Riekstins ini menjual produk-produk nirkabel dan router yang populer di pasar teknologi nirkabel di negara-negara berkembang. Perangkat keras terbaik dari Mikrotik adalah :

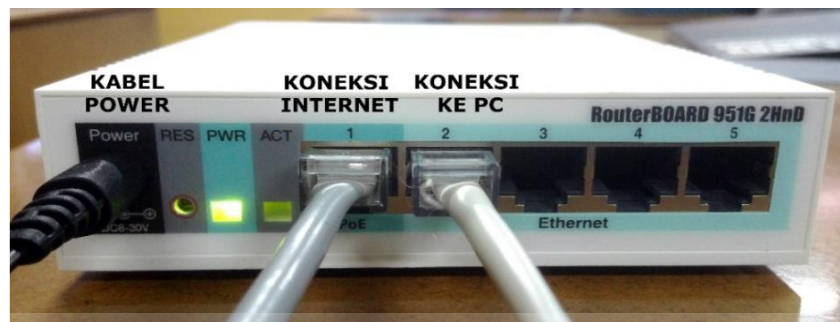
1. Router
2. Switch
3. Antena

Sedangkan Perangkat lunak terbaik Mikrotik hanya MikroTik RouterOS.

MikroTik *RouterOS* adalah sistem operasi dan perangkat lunak yang dapat digunakan untuk membuat komputer menjadi router *network* yang handal, mencakup berbagai fitur yang dibuat untuk *IP network* dan jaringan *wireless*, cocok digunakan oleh ISP dan *provider hotspot*. Untuk instalasi MikroTik tidak dibutuhkan piranti lunak tambahan atau komponen tambahan lain. MikroTik didesain untuk mudah digunakan dan sangat baik digunakan untuk keperluan administrasi jaringan komputer seperti merancang dan membangun sebuah sistem jaringan komputer skala kecil hingga yang kompleks sekalipun.

*RouterBoard* adalah router *embedded* produk dari mikrotik. *RouterBoard* seperti sebuah PC mini yang terintegrasi karena dalam satu board tertanam

prosesor, RAM, ROM, dan memori flash. *RouterBoard* menggunakan sistem operasi *RouterOS* yang berfungsi sebagai router jaringan, *bandwidth management*, *proxy Server*, *dhcp*, *DNS Server* dan bisa juga berfungsi sebagai *hotspot Server*. Ada beberapa seri *RouterBoard* yang juga bisa berfungsi sebagai *wifi* seperti seri RB411, RB433, RB600, RB950.



Gambar 2. 2 Router Mikrotik

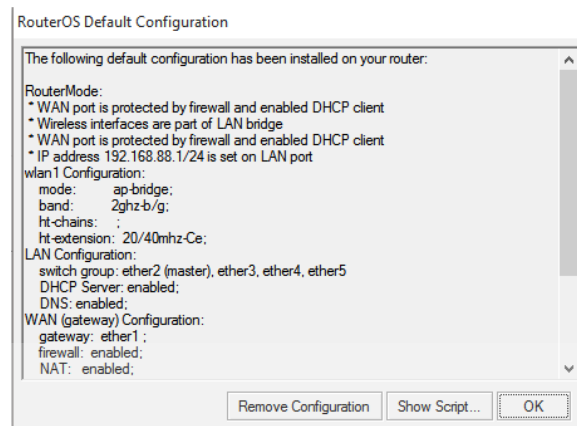
Kebanyakan Router MikroTik diletakkan diantara jaringan LAN dengan jaringan Internet. Jaringan LAN biasanya memiliki beberapa komputer yang terhubung. Beberapa produk Router MikroTik dilengkapi dengan *fitur wireless interface* sehingga dapat difungsikan sebagai *access point* pada jaringan nirkabel. Biasanya *access point* tersebut digunakan pada jaringan berskala kecil.

### 2.13 Winbox

Winbox adalah perangkat lunak (*software*) milik MikroTik yang digunakan untuk mengkonfigurasi MikroTik *RouterBoard*. Winbox telah berbasis *Guest User Interface* sehingga memudahkan *user* dalam pemakaiannya. Terdapat beberapa tahap dalam konfigurasi awal dengan menggunakan Winbox :

1. Buka aplikasi Winbox, klik tab *Neighbors*, dan tunggu beberapa saat.

2. Apabila *RouterBoard* sudah muncul, klik pada *MAC Address* atau *IP Address* perangkat tersebut. Maka *MAC Address* atau *IP Address* akan otomatis tersalin ke kotak '*Connect To*'.
3. Login dengan *username 'admin'* dan tanpa *password*.
4. Terakhir, klik *Connect*.



Gambar 2. 3 Konfigurasi Awal Winbox

Saat pertama kali terhubung ke MikroTik *RouterBoard* dan masuk ke Winbox, akan muncul jendela *Default Configuration* yang menunjukkan ringkasan konfigurasi pabrikan (*default*) dari *RouterBoard*. Bila memilih 'OK', maka konfigurasi pabrikan akan diterapkan. Bila tidak menginginkan demikian, maka dapat memilih '*Remove Configuration*' yang mana akan menghapus konfigurasi pabrikan dan memulai konfigurasi MikroTik dari nol (tanpa konfigurasi apapun).

## BAB III

### METODOLOGI PENELITIAN

#### 3.1 Metode Pengumpulan Data

Metodologi yang dilakukan dalam proses pengumpulan data yang dilakukan adalah dengan mengumpulkan beberapa referensi berdasarkan kajian pustaka, implementasi materi yang didapat selama perkuliahan serta mempelajari pola dari struktur jaringan internet yang terdapat pada Dinas Pendidikan Kota Surabaya.

Proses pengumpulan referensi dilakukan dengan membaca buku, jurnal maupun beberapa artikel terkait yang terdapat pada internet, serta melakukan pengamatan maupun analisis terhadap berbagai metode dalam optimalisasi manajemen *Bandwidth* sebagai bahan acuan pembelajaran.

#### 3.2 Peralatan Penelitian

Peralatan atau perangkat yang dibutuhkan dalam proses pengerjaan ini dibedakan mejadi dua jenis, yaitu perangkat keras (*hardware*) dan perangkat lunak (*software*).

##### 3.2.1 Perangkat Keras

Perangkat keras yang dibutuhkan dalam melakukan pengujian ini membutuhkan sebuah *Router*. Adapun tipe *Router* yang digunakan adalah Mikrotik RB951G-2HnD dengan spesifikasi sebagai berikut :

1. CPU : AR9344 600MHz
2. *Main Storage/NAND* : 64MB

3. RAM : 128 MB
4. *RouterOS License* : Level4

### 3.2.2 Perangkat Lunak

Perangkat lunak yang dibutuhkan dalam melakukan pengujian ini membutuhkan beberapa *software* untuk melakukan sebuah simulasi. Dimana dalam proses pengujiannya tidak memungkinkan untuk melakukan perubahan secara langsung terhadap jaringan, sehingga dilakukan sebuah simulasi dengan menggunakan beberapa *software* berikut :

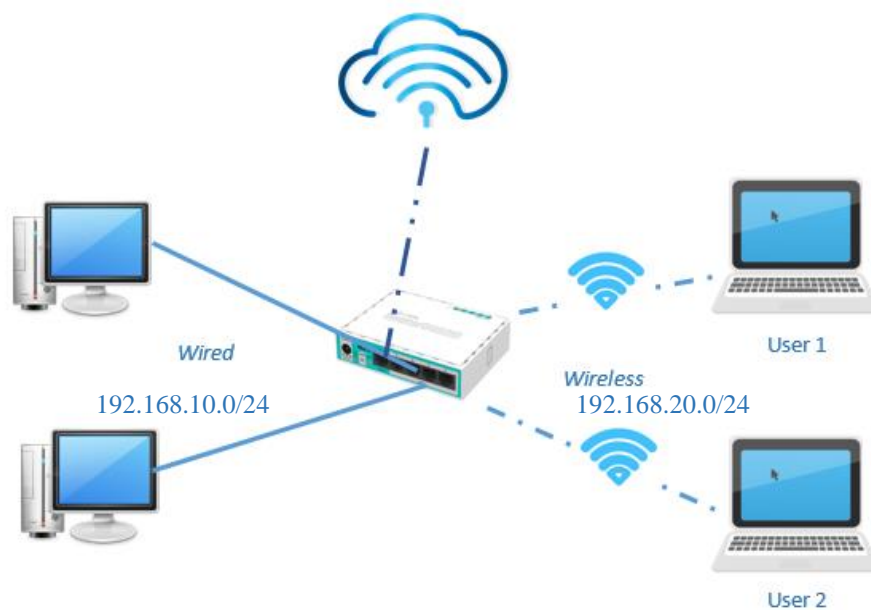
1. *Windows 10* : Sebagai sistem operasi PC
2. Mikrotik : *RouterOS Level4*
3. *Winbox* : Untuk konfigurasi mikrotik

### 3.3 Perancangan Sistem

Pada tahap perancangan ini akan dibuat sebuah rangkaian topologi yang akan dibangun agar dapat mengimplementasikan manajemen *Bandwidth* menggunakan *Queue Tree* dengan metode *Peer Connection Queue*.

### 3.4 Skenario Simulasi

Skenario yang akan dibuat untuk jaringan dengan pengimplementasian manajemen *Bandwidth* menggunakan *Queue Tree* dengan metode *Peer Connection Queue* dapat diamati pada topologi Gambar 3.1 berikut.



Gambar 3. 1 Topologi Simulasi Manajemen *Bandwidth*

Berdasarkan topologi pada Gambar 3.1 maka nantinya untuk setiap konfigurasi akan menggunakan *network* 192.168.10.0/24 sebagai *IP Address* untuk jalur *wired* dan menggunakan *network* 192.168.20.0/24 sebagai *IP Address* untuk jalur *wireless*. Serta menggunakan DNS dari *google* yaitu 8.8.8.8 saat melakukan uji ping atau uji koneksi ke internet.

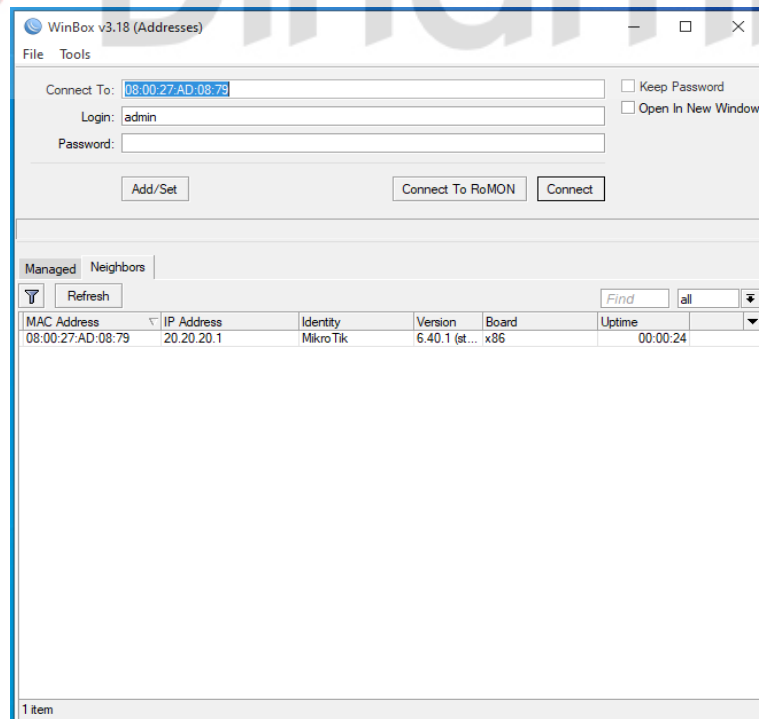
## BAB IV

### HASIL ANALISA

Pada bab ini akan dibahas terkait hasil analisa dengan berdasarkan pada proses manajemen *bandwidth* dengan mikrotik pada Dinas Pendidikan Surabaya. Dimana dalam proses penerapannya penulis melakukan analisa dengan menggunakan jaringan lokal dengan sampel beberapa user serta jaringan tersendiri agar tidak mengganggu aktifitas jaringan dari Dinas Pendidikan Surabaya.

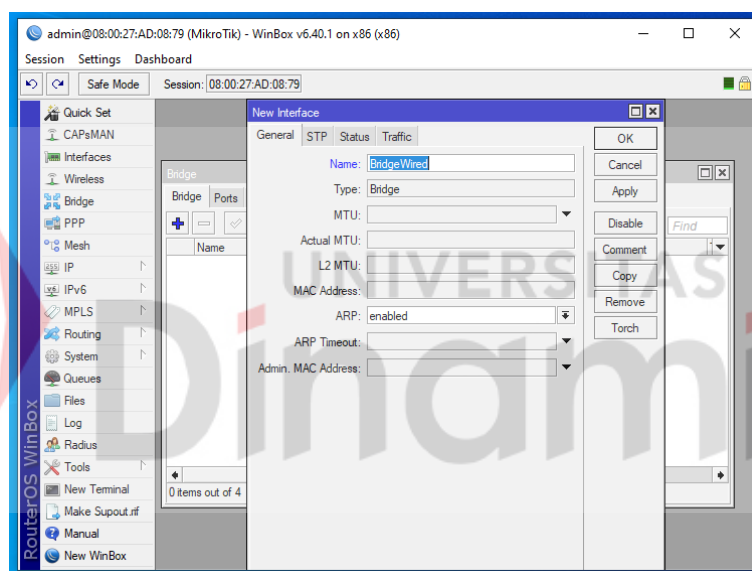
#### 4.1 Konfigurasi dengan Winbox

Pertama hal yang harus dilakukan adalah membuka aplikasi *Winbox* pada PC yang akan digunakan untuk menyeting *Router* Mikrotik. Untuk login kedalam winbox, klik pada bagian *MAC Address* lalu klik *connect*.



Gambar 4. 1 Jendela Awal Winbox

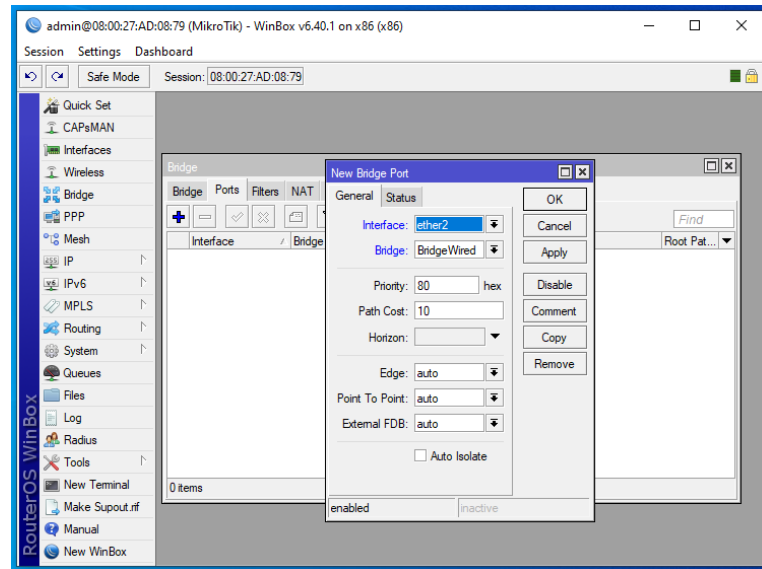
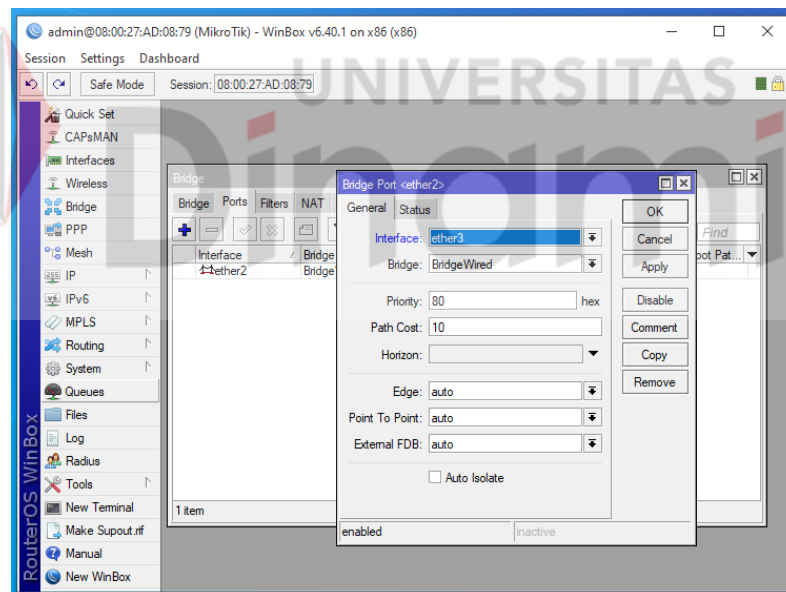
Setelah melakukan proses *Login*, maka jendela awal *Winbox* akan terbuka. Terdapat beberapa tahapan yang perlu dilakukan sebelum melakukan konfigurasi manajemen *bandwidth*. Langkah pertama adalah melakukan konfigurasi *bridge*. Pembuatan *interface bridge* dilakukan dengan melalui menu *bridge* -> tombol *add*. Kemudian pada kolom nama dapat diberi nama sesuai dengan kehendak. Dalam proses konfigurasi ini, kolom nama diisi dengan nama *BridgeWired* untuk memudahkan dalam mengingatnya nanti. Dimana ini nantinya akan berfungsi sebagai *bridge* atau jembatan penghubung untuk jalur kabel/*wired*.

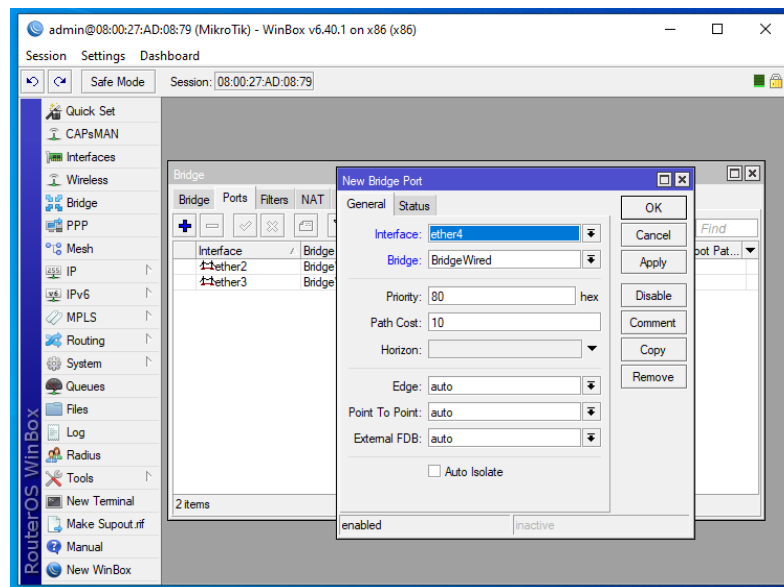


Gambar 4. 2 Konfigurasi *Interface BridgeWired*

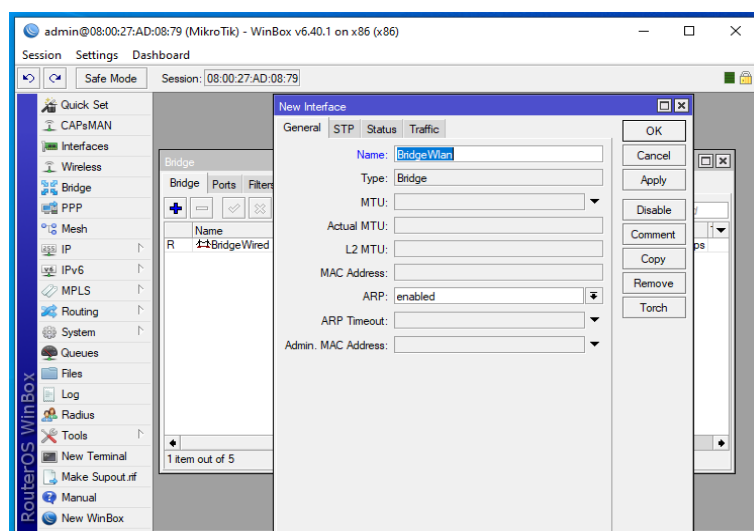
Setelah membuat *interface BridgeWired*, langkah selanjutnya adalah memasukkan *interface ether2*, *ether3*, dan *ether4* ke dalam *interface* tersebut. Hal ini berfungsi untuk mengaktifkan semua *port ether* yang terdapat pada router Mikrotik. Langkah-langkah untuk melakukan konfigurasi tersebut melalui menu *Bridge* -> tab *Ports* -> tombol *add*, dan pilih *ether* yang akan kita *bridging*.



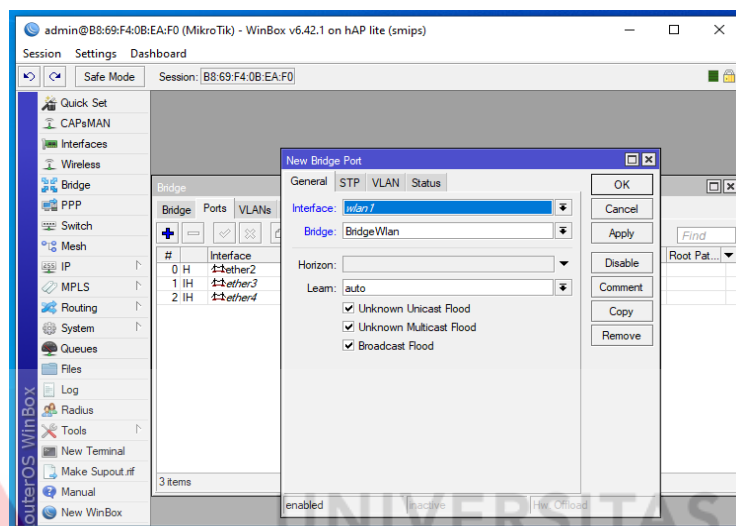
Gambar 4. 3 Konfigurasi *Ether2*Gambar 4. 4 Konfigurasi *Ether3*

Gambar 4. 5 Konfigurasi *Ether4*

Setelah membuat *interface BridgeWired*, maka selanjutnya membuat sebuah *interface* dengan nama *BridgeWlan*. *Interface* ini memiliki fungsi yang sama dengan *BridgeWired* yaitu untuk membuat sebuah jembatan penghubung namun untuk jalur *wireless/nirkabel*. Tahapan untuk membuat *Interface BridgeWlan* sama dengan tahapan yang sebelumnya, yaitu dengan masuk ke menu *Bridge* -> tombol *add*, dan beri nama *BridgeWlan*.

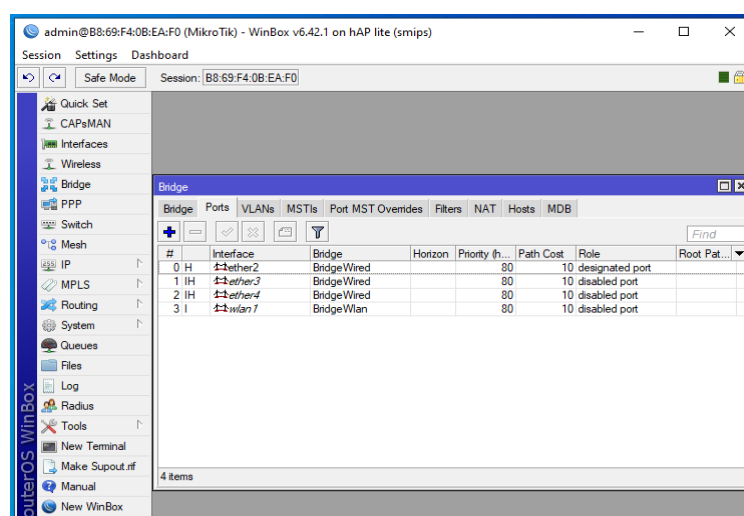
Gambar 4. 6 Konfigurasi *BridgeWlan*

Selanjutnya adalah dengan membuat sebuah *bridge port* yang akan di-*bridge* pada *interface BridgeWlan*. Adapun cara untuk melakukan konfigurasi dengan masuk kedalam menu *Bridge* -> tab *Ports* -> tombol *add*, dan beri nama *BridgeWlan*. Proses konfigurasi dapat diamati seperti pada Gambar 4.7 berikut.



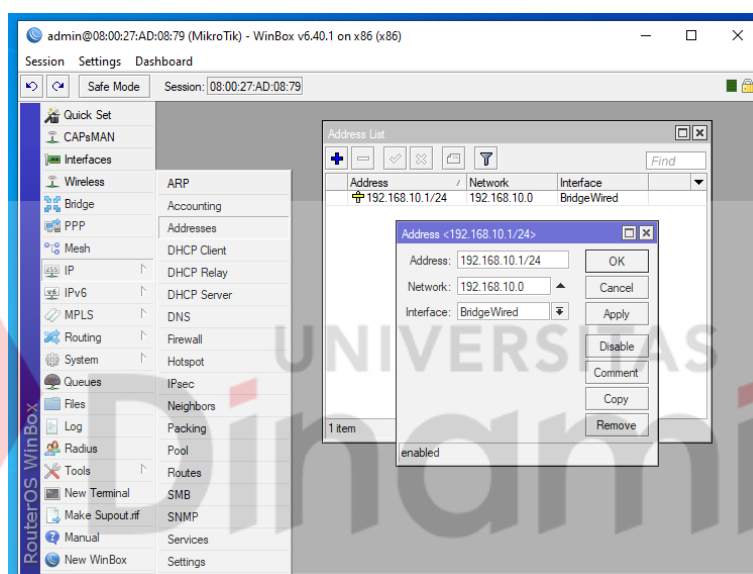
Gambar 4. 7 Konfigurasi Port Wlan1

Setelah dilakukan beberapa konfigurasi pada *port* dengan *interface ether2*, *ether3*, dan *ether4* telah ter-*bridge* pada *BridgeWired* dan *interface Wlan1* telah ter-*bridge* pada *interface BridgeWlan*. Maka hasilnya akan terlihat seperti Gambar 4.8.



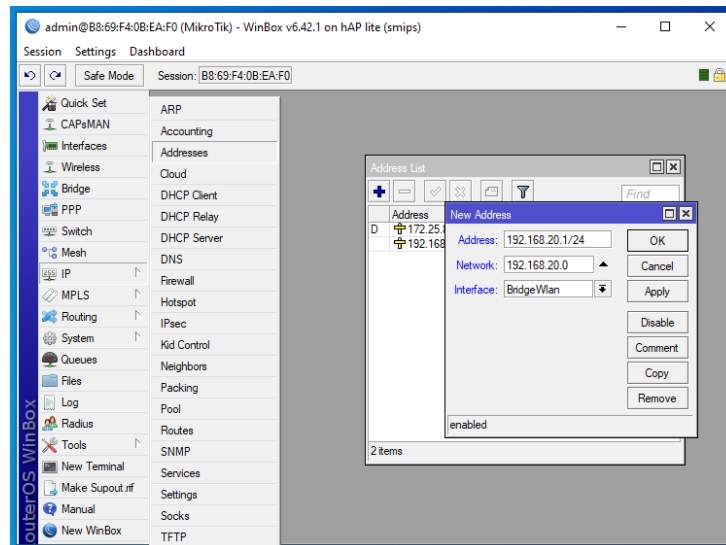
Gambar 4. 8 Hasil Konfigurasi Semua Port

Tahap selanjutnya adalah *setting* IP untuk *BridgeWired* dan *BridgeWlan*. Masuk ke dalam menu IP -> *Addresses* -> tombol *add*. Kemudian membuat sebuah IP Address: 192.168.10.1/24 yang dimana nantinya IP ini berfungsi sebagai IP DHCP yang akan dibagikan kepada *user*. IP ini dapat di isi sesuai dengan range dan kelas IP yang diinginkan. Lalu ubah *Interface: BridgeWired*, kemudian klik *apply* dan OK. Dengan demikian *bridge* untuk jalur kabel/*wired* telah mempunyai IP 192.168.10.1/24 seperti pada Gambar 4.9.



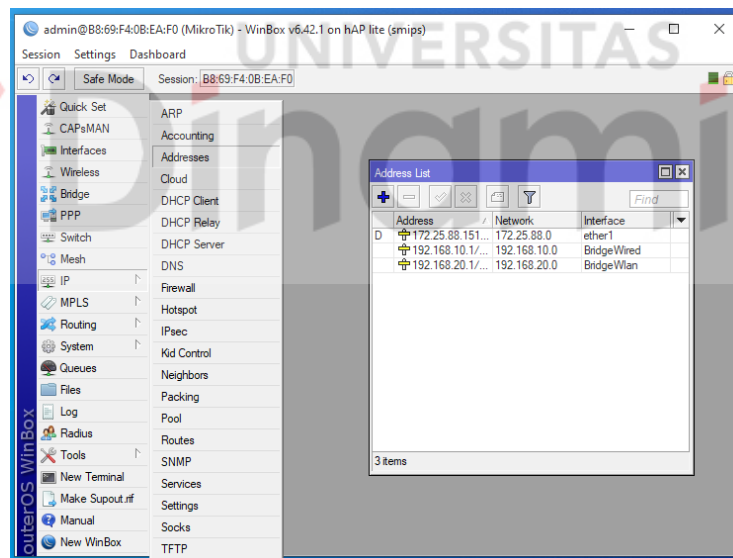
Gambar 4. 9 Konfigurasi IP Address BridgeWired

Setelah memberikan IP pada jalur kabel, maka selanjutnya adalah memberikan sebuah IP pula kepada *bridge* untuk jalur nirkabel. Adapun tahapannya tetap sama dengan sebelumnya, yaitu dengan masuk pada menu *Bridge* -> tombol *add*, lalu masukan Address: 192.168.20.1/24, *interface: BridgeWlan* dan klik *Apply* kemudian klik OK. IP address tersebut merupakan IP yang akan dibagikan kepada *user* yang terhubung pada jaringan internet pada jalur *wireless* nantinya. Konfigurasi jalur nirkabel tersebut dapat diamati pada Gambar 4.10.



Gambar 4. 10 Konfigurasi *IP Address BridgeWan*

Jika telah berhasil melakukan konfigurasi atau pembuatan *IP Address*-nya pada setiap *interface*, maka hasilnya dapat diamati seperti pada Gambar 4.11 berikut.

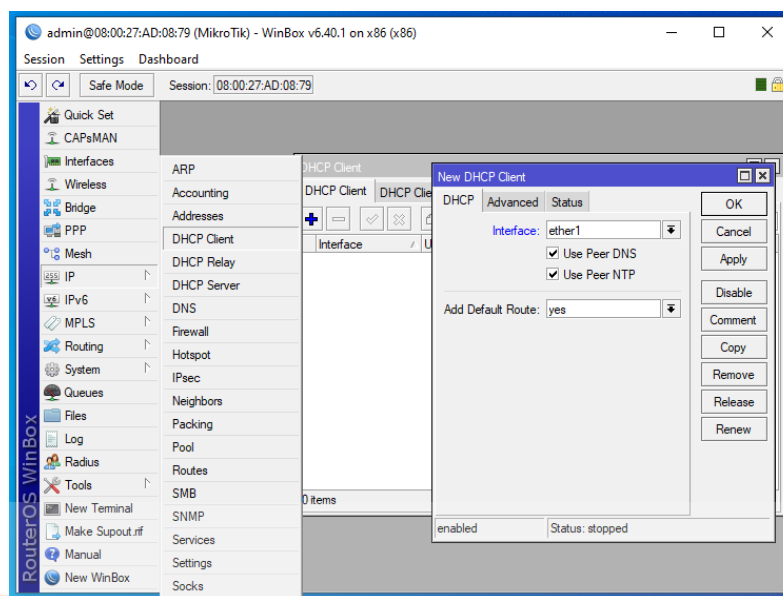


Gambar 4. 11 Hasil Konfigurasi Semua *IP Address*

#### 4.2 Setting DHCP Client

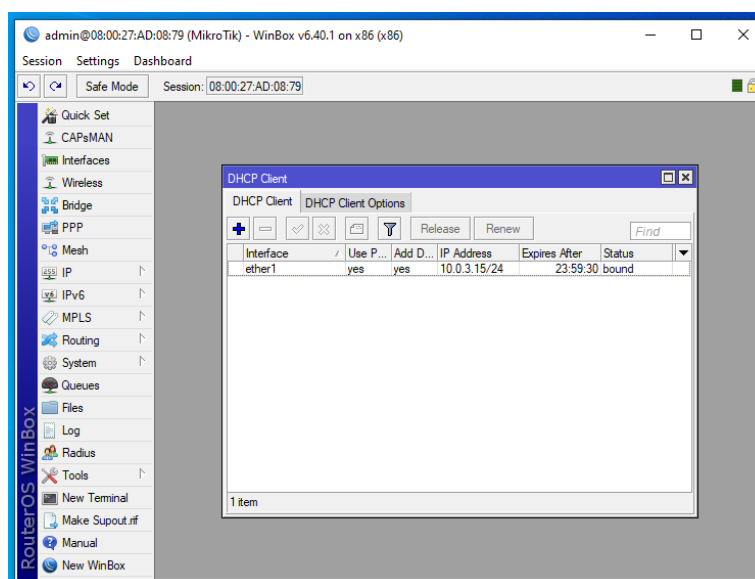
Tahap yang kedua adalah untuk mendapatkan *IP Address*, *Gateway*, dan *DNS Server* dari *Router ISP* secara otomatis. Adapun langkah yang harus dilakukan

adalah dengan membuat DHCP *Client*. Tahap yang perlu dilakukan ialah dengan masuk pada menu IP -> DHCP *Client* -> pilih tombol *add* (*interface* tetap di *ether1*, karena *ether 1* merupakan *ether* yang terhubung ke jaringan internet) lalu klik OK.



Gambar 4. 12 Konfigurasi DHCP *Client*

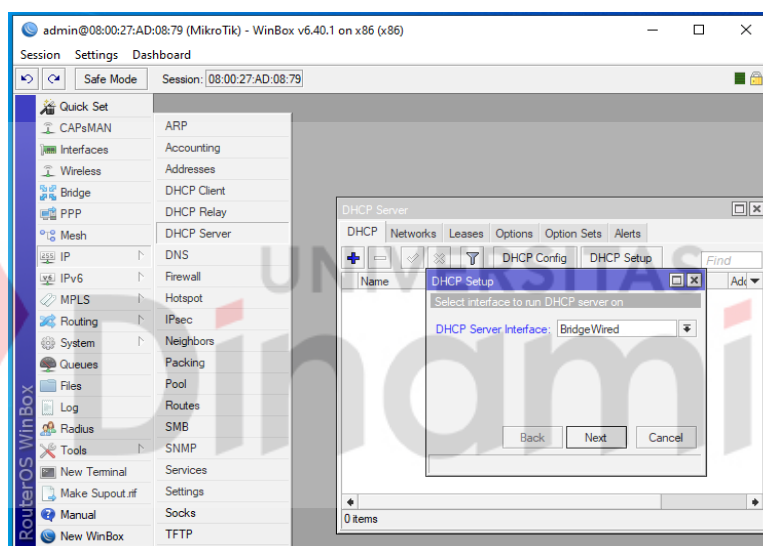
Hasilnya dapat dilihat pada Gambar 4.13 seperti berikut ini, dimana telah didapatkan IP Address 172.23.26.12 secara otomatis. IP tersebut didapatkan dari *provider/ISP* yang memberikan koneksi internet.



Gambar 4. 13 Hasil Konfigurasi DHCP *Client*

### 4.3 Setting DHCP Server

Tahap yang ketiga adalah membuat dua buah konfigurasi untuk DHCP Server, dimana nantinya satu akan digunakan pada jaringan *Wired* atau *BridgeWired* dan satu buah DHCP Server untuk jaringan *Wireless* atau *BridgeWlan*. Proses konfigurasi dapat dilakukan dengan cara masuk pada menu IP -> DHCP Server. Kemudian klik tab DHCP *setup* dan ganti DHCP Server Interface dengan *BridgeWired*, klik *Next* sampai proses selesai. DHCP ini berfungsi untuk menyediakan IP kepada user yang akan masuk pada jalur kabel LAN.

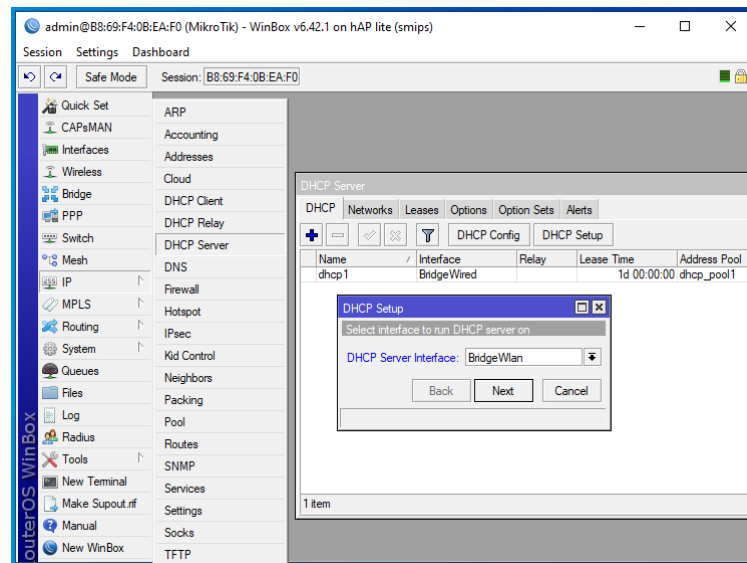


Gambar 4. 14 Konfigurasi DHCP Server Wired

### 4.4 Setting DHCP Server Untuk Wireless

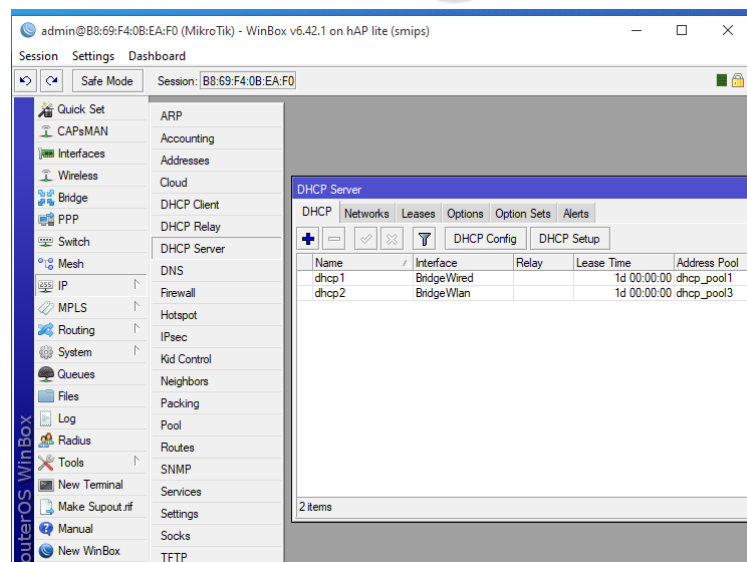
Setelah berhasil melakukan konfigurasi pada DHCP Server untuk kabel atau *wired*, maka tahapan selanjutnya adalah dengan melakukan konfigurasi pada DHCP Server untuk *wireless*-nya. Untuk proses konfigurasi dapat dilakukan dengan cara masuk pada menu IP -> DHCP Server -> klik DHCP *setup* dan ganti DHCP

*ServerInterface* dengan *BridgeWlan*. Klik *Next* sampai selesai. Cara ini hampir sama dengan sebelumnya, hanya berbeda pada pembuatan *interface*-nya.



Gambar 4. 15 Konfigurasi DHCP Server Wlan

Hasilnya dari konfigurasi DHCP Server untuk jaringan kabel dan nirkabel dapat diamati pada Gambar 4.16 berikut.

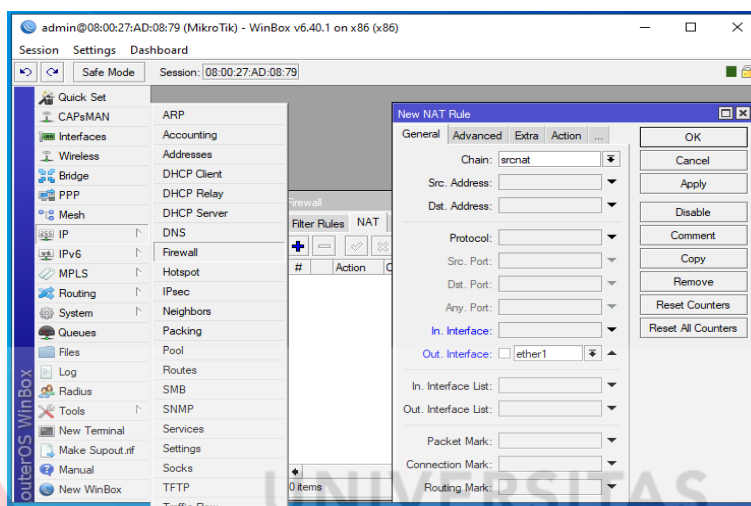


Gambar 4. 16 Hasil Konfigurasi DHCP Server Wired dan Wlan



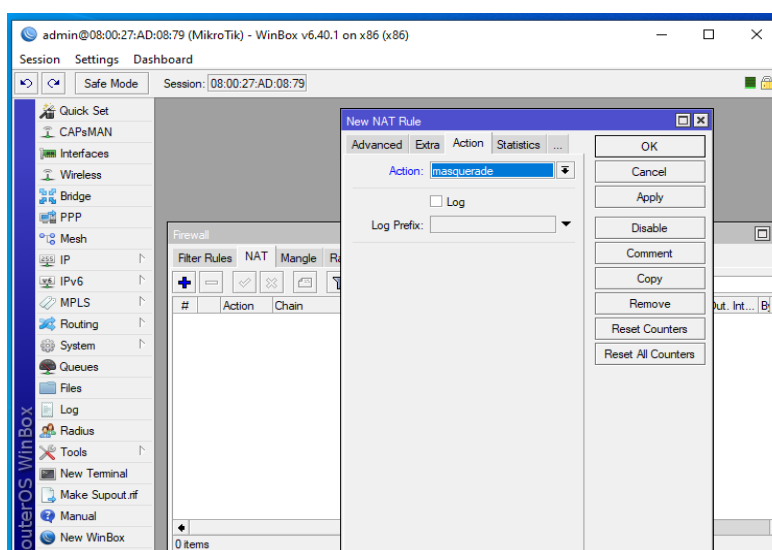
#### 4.5 Setting NAT

Tahap yang selanjutnya adalah melakukan *setting* pada NAT. Adapun cara untuk konfigurasinya ialah masuk ke menu IP -> *Firewall* -> tab NAT -> klik *add* lalu ganti Out. *Interface: ether1*, klik OK. Proses ini berfungsi untuk melakukan perubahan IP asli dari *user* saat akan dikirim ke internet.



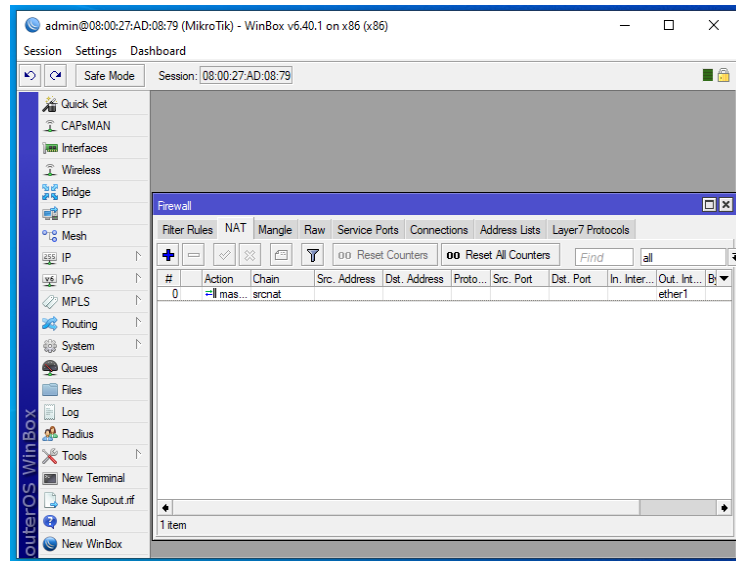
Gambar 4. 17 Konfigurasi NAT pada *ether1*

Kemudian klik Tab *Action* dan ganti *Action: masquerade*, klik OK. Dengan demikian saat user melakukan aktifitas ke internet, maka IP yang dipakai adalah IP dari *router* dan bukan IP asli dari *user*.



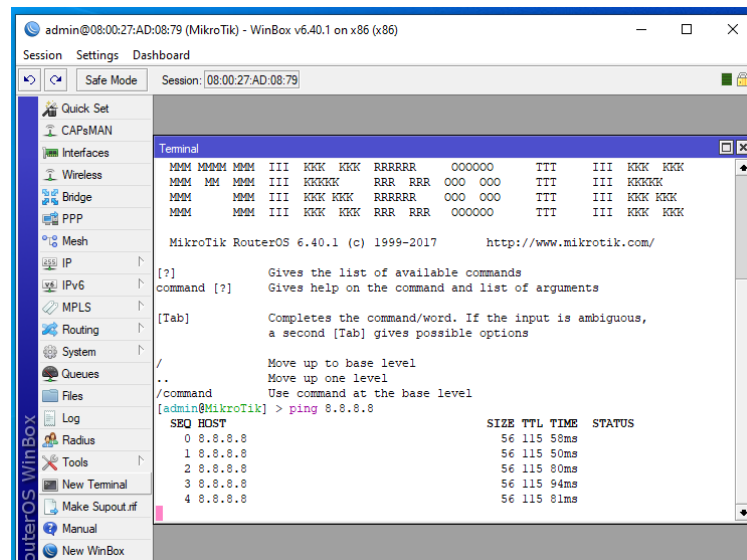
Gambar 4. 18 Konfigurasi NAT pada *Tab Action*

Jika konfigurasi dilakukan dengan benar, maka hasil dari konfigurasi NAT akan seperti Gambar 4.19 berikut.



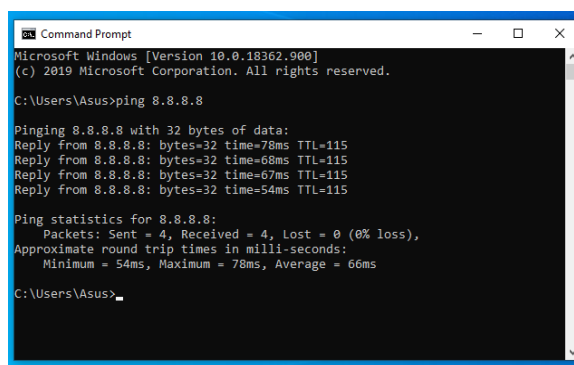
Gambar 4. 19 Hasil Konfigurasi NAT

Setelah dilakukan konfigurasi NAT, untuk mengetahui apakah hasil konfigurasi yang dilakukan telah berhasil atau tidak, maka dapat dilakukan pengecekan koneksi internet melalui terminal *router* dengan cara pilih tab *New Terminal* -> kemudian ping ke DNS *Google* dengan ketik: `ping 8.8.8.8` lalu *enter*.



Gambar 4. 20 Cek Koneksi Internet Melalui *New Terminal*

Jika uji koneksi internet dengan melakukan ping ke DNS *Google* melalui *terminal* router telah berhasil, maka selanjutnya adalah melakukan cek koneksi juga terhadap perangkat PC kita. Dengan membuka *Command Prompt* dan ketik *ping 8.8.8.8* seperti halnya pada *new terminal* lalu *enter*.



```

Microsoft Windows [Version 10.0.18362.900]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\Asus>ping 8.8.8.8

Pinging 8.8.8.8 with 32 bytes of data:
Reply from 8.8.8.8: bytes=32 time=78ms TTL=115
Reply from 8.8.8.8: bytes=32 time=68ms TTL=115
Reply from 8.8.8.8: bytes=32 time=67ms TTL=115
Reply from 8.8.8.8: bytes=32 time=54ms TTL=115

Ping statistics for 8.8.8.8:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 54ms, Maximum = 78ms, Average = 66ms

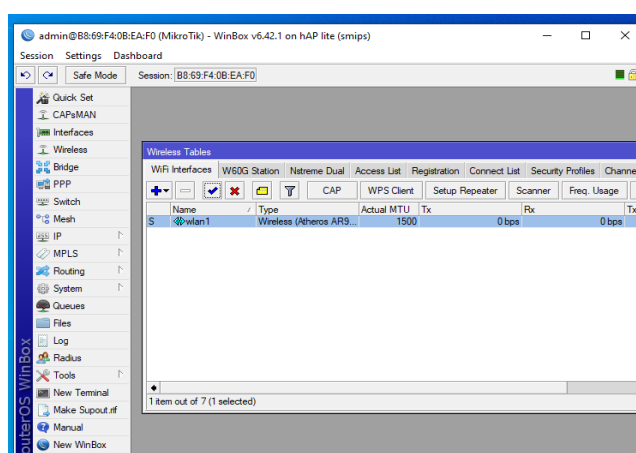
C:\Users\Asus>

```

Gambar 4. 21 Cek Koneksi Internet Melalui *CMD* Pada PC

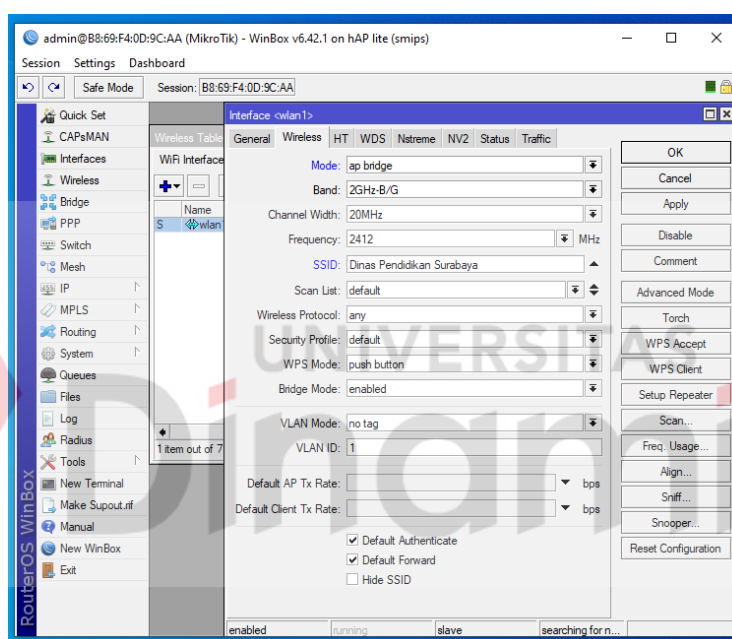
#### 4.6 Sharing Koneksi Via *Wireless*

Setelah memastikan bahwa koneksi internet telah berhasil atau router dipastikan sudah terhubung dan memiliki koneksi internet, maka selanjutnya adalah membuat sebuah koneksi jaringan secara nirkabel. Adapun langkah-langkahnya adalah masuk pada menu *Wireless*, kemudian klik pada *interface Wlan1*, klik tanda centang untuk mengaktifkan atau *enable interface Wlan1* tersebut.



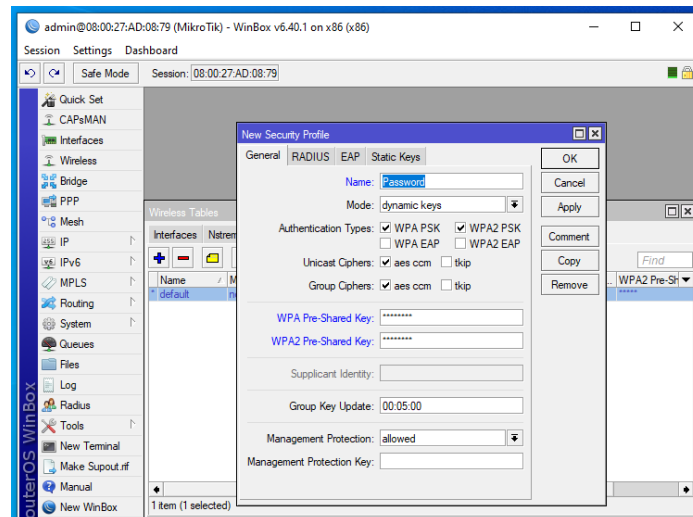
Gambar 4. 22 Mengaktifkan *Interface Wlan1*

Selanjutnya membuat sebuah jaringan untuk akses jalur *wireless* atau *Wifi*. Caranya adalah dengan melakukan klik dua kali pada *interface Wlan1*, kemudian klik tab *Wireless* dan ganti Mode dengan *AP Bridge*, agar konfigurasi yang sebelumnya telah dibuat dapat digunakan. Setelah itu ganti *Band*: 2GHz-B/G/N, hal ini bermaksud agar data *rate* bisa mencapai maksimum 300Mbps. Lalu memberi nama pada jaringan *Wifi* atau nama *SSID*: Dinas Pendidikan Surabaya, kemudian klik OK. Proses konfigurasi tersebut dapat diamati pada Gambar 4.23.



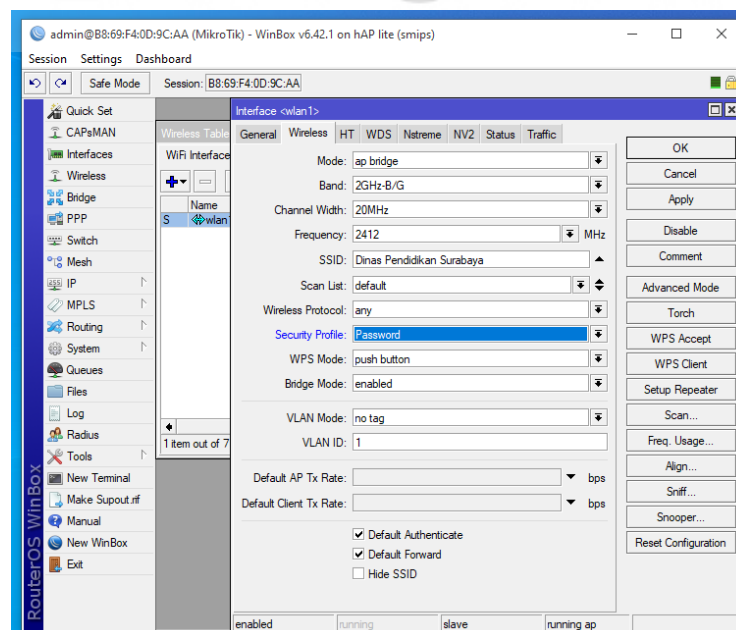
Gambar 4. 23 Membuat Jaringan Untuk Akses *Wireless*

Adapun untuk meningkatkan keamanan jaringan dan mencegah pihak diluar Dinas Pendidikan dapat mengakses jaringan internet, maka buat sebuah *password*. Dalam melakukan pembuatan *password* ini dengan masuk dan klik tab *Security Profiles* -> tombol *add*, ganti Name: *Password*. Masukkan *Password* yang sama pada kotak *WPA Pre-Shared Key* dan *WPA2 Pre-Shared Key* (8 digit, bisa angka maupun huruf). Disini menggunakan *nurullah*, kemudian klik OK.



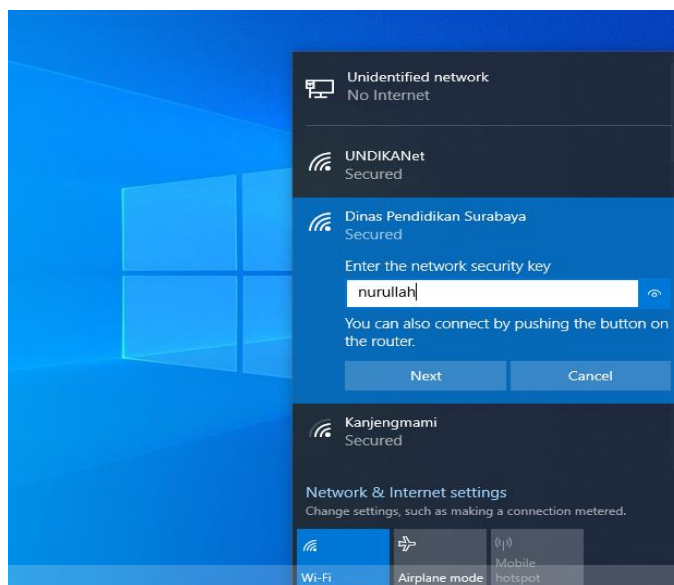
Gambar 4. 24 Membuat *Password Wifi*

Kemudian kembali ke tab *Interface*, dan klik dua kali pada *interface Wlan1*. Ganti *Security Profile* dengan *Profile Password* yang telah kita buat sebelumnya dan klik OK. Hal ini berfungsi agar user memasukan sandi yang telah dibuat sebelumnya sebagai syarat *login* kedalam jaringan internet jalur nirkabel. Proses konfigurasi tersebut dapat diamati pada Gambar 4.25 berikut.



Gambar 4. 25 Konfigurasi *Password* kedalam *Interface Wlan1*

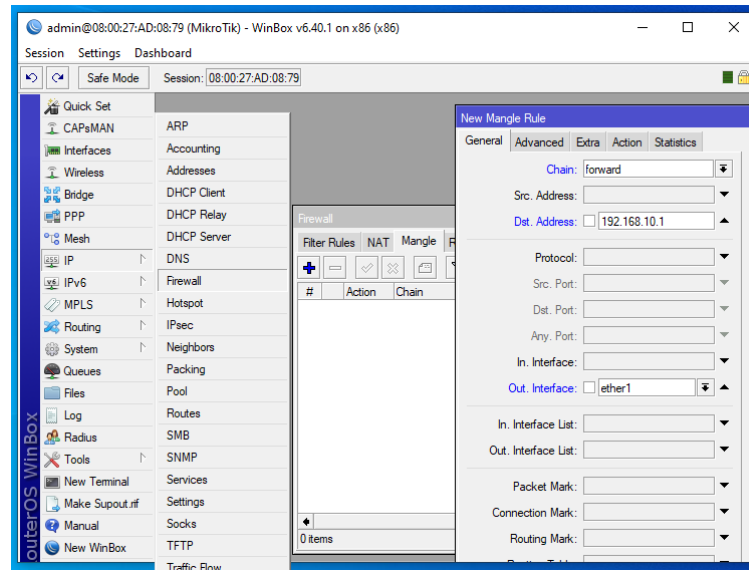
Setelah itu cek pada icon koneksi internet pada pojok kanan bawah PC. Pilih dan masukkan *password* sesuai dengan yang telah dibuat sebelumnya.



Gambar 4. 26 Tampilan Koneksi *Wireless SSID* dan *Password*

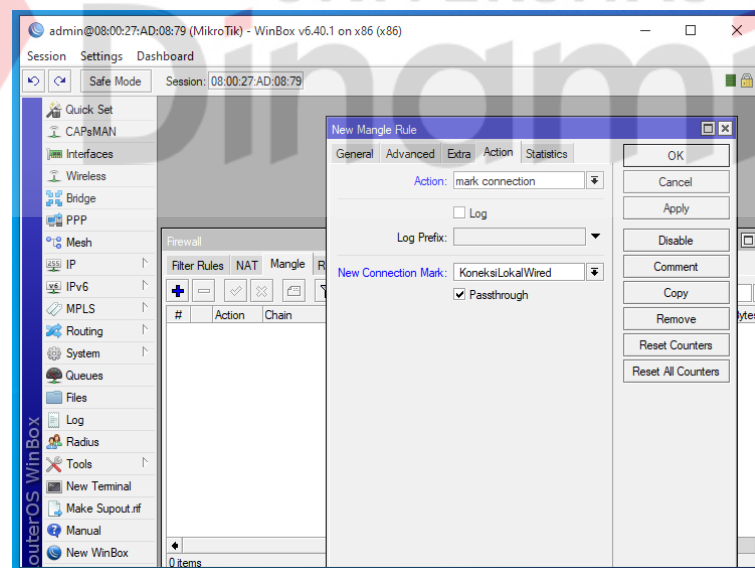
#### 4.7 Setting *Queue Tree* dan *PCQ*

Berikut merupakan tahap inti dalam proses manajemen *bandwidth*. Konfigurasi queue berfungsi untuk memlimit penggunaan koneksi internet secara merata. Terdapat beberapa tahapan dalam melakukan konfigurasi tersebut. Pertama, buka menu IP -> *Firewall* -> tab *Mangle* -> tombol *add* dan ganti *Chain: Forward*, hal ini bertujuan untuk melakukan filter terhadap IP yang dipakai saat melakukan koneksi ke internet. Kemudian ganti *Src. Address: 192.168.10.0/24* dan *Out Interface: ether1*.



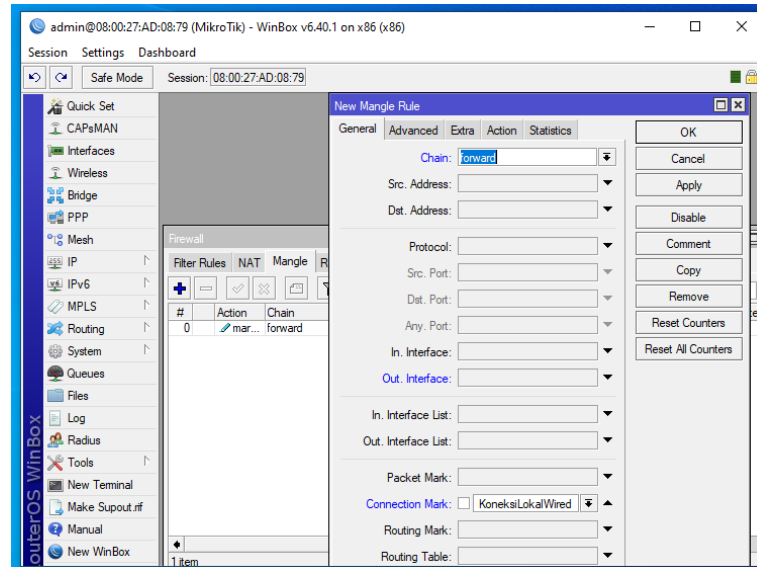
Gambar 4. 27 Konfigurasi *Mangle* pada *Ether1*

Kemudian masuk kedalam Tab *Action*, ganti *Action: mark connection*, ganti *New Connection Mark: KoneksiLokalWired*. Konfigurasi dapat dilihat pada Gambar 4.28.



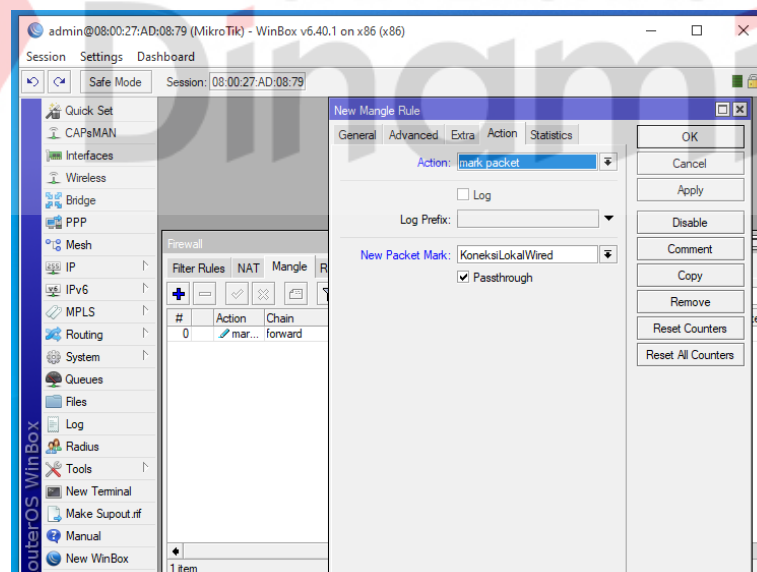
Gambar 4. 28 Konfigurasi *Action* dengan *KoneksiLokalWired*

Selanjutnya membuat sebuah mangle baru lagi dengan cara klik tombol *add* -> tab *General*. Kemudian ganti *Chain: forward*, *Connection Mark: KoneksiLokalWired*.



Gambar 4. 29 Konfigurasi *Mangle* dengan *KoneksiLokalWired*

Kemudian masuk kedalam tab *Action*, lalu ganti *Action: mark packet, New Packet Mark: KoneksiLokalWired*. Klik OK.

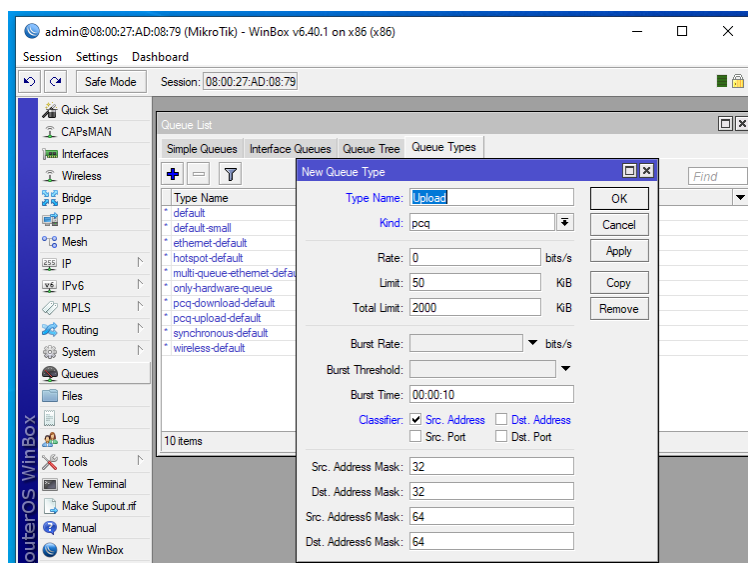


Gambar 4. 30 Konfigurasi *Action* dengan *Mark Packet*

Selanjutnya melakukan konfigurasi untuk *queue* dengan cara masuk ke menu *Queues* -> tab *Queues Types* -> tombol *add*. Pada tahap ini konfigurasi *queue*

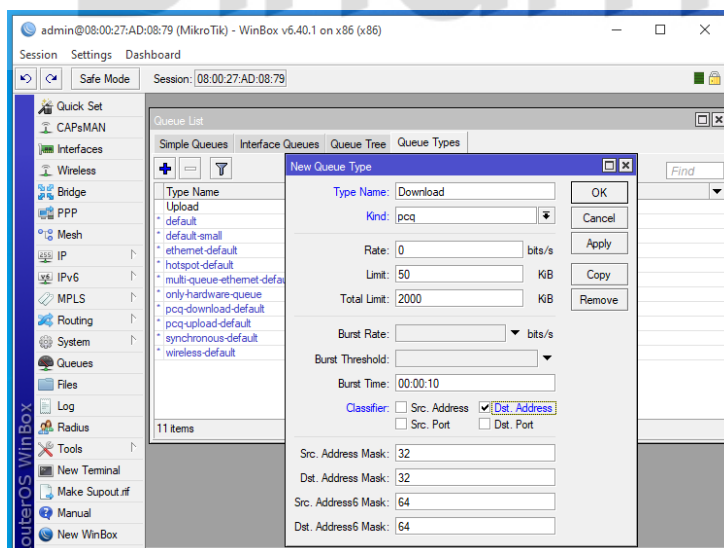


berfungsi untuk memisahkan antara *upload* dan *download*. Kemudian ganti juga *Type Name: Upload, Kind: pcq, Classifier: Src Address*, klik OK.



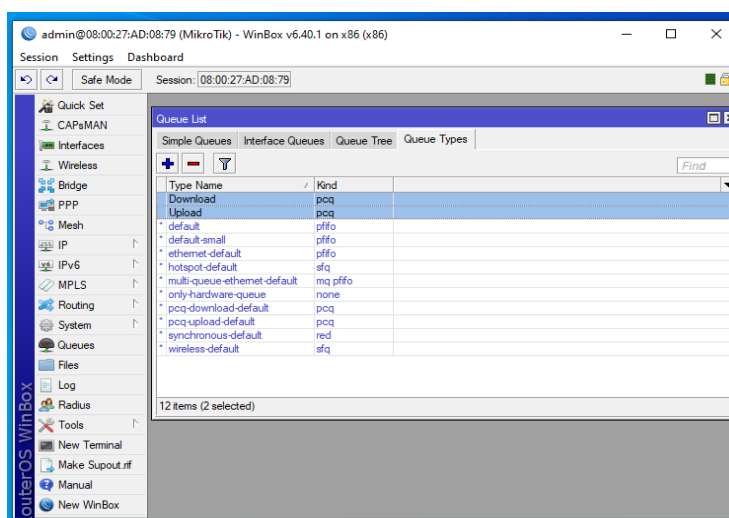
Gambar 4. 31 Konfigurasi *Queue* untuk *Upload*

Setelah itu klik tombol *add* ganti *Type Name: Download, Kind: pcq, Classifier: Dst. Address*, klik OK.



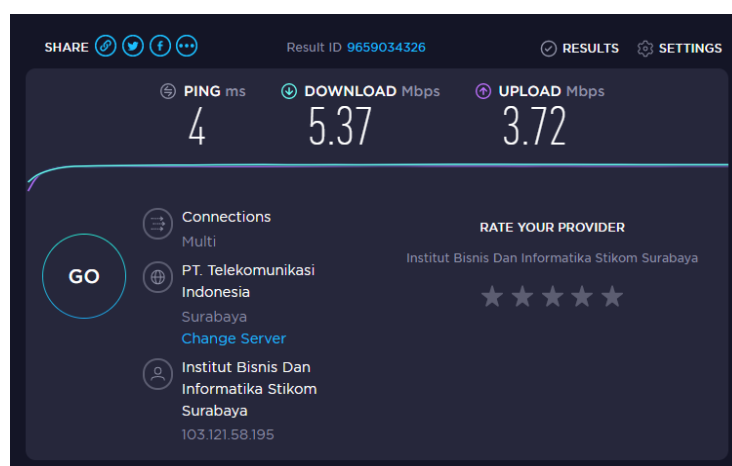
Gambar 4. 32 Konfigurasi *Queue* untuk *Download*

Hasil dari konfigurasi tersebut jika telah dilakukan dengan benar dan sesuai, maka dapat diamati pada Gambar 4.33 sebagai berikut.



Gambar 4. 33 Hasil Konfigurasi *Queue*

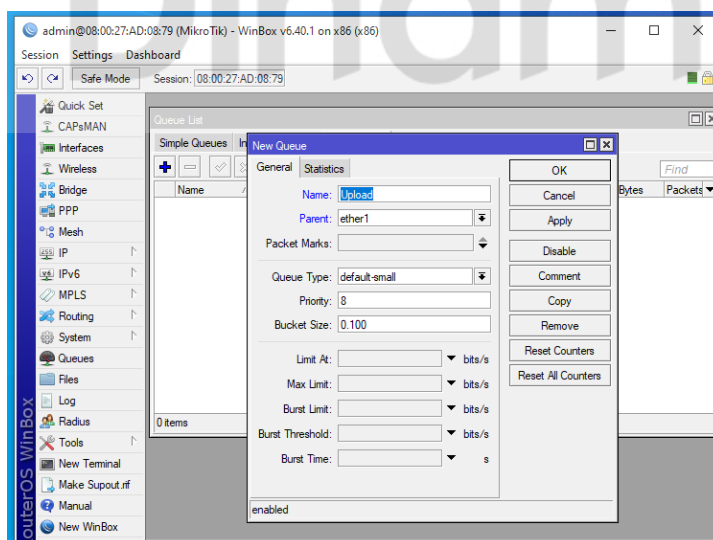
Bila setiap konfigurasi telah dilakukan, tahapan selanjutnya adalah melakukan tes kecepatan internet yang didapat dari ISP. Caranya dengan buka *web browser* kemudian ketik *speedtest.net* pada laman *url* lalu *enter*. Dari jaringan yang telah dibuat maka teruji bahwa koneksi internet lancar dan mempunyai *bandwidth* seperti pada Gambar 4.33 berikut.



Gambar 4. 34 Pengujian Koneksi *Wired* Sebelum Manajemen

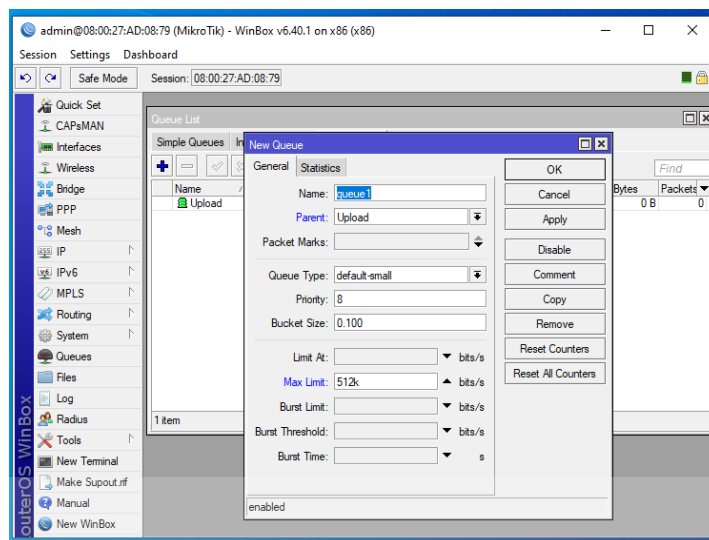
Seperti halnya yang sudah dijelaskan pada awal sub bab, bahwa dimana dalam proses penerapannya penulis melakukan analisa dengan menggunakan jaringan lokal dengan sampel beberapa *user* serta jaringan tersendiri agar tidak mengganggu aktifitas jaringan dari Dinas Pendidikan Surabaya. Oleh karena itu penulis menggunakan jaringan internet dari Universitas Dinamika yang masih mempunyai *user server name* Institut Bisnis dan Informatika Stikom Surabaya sebagai pengganti dari koneksi internet Dinas Pendidikan.

Dari data diatas dapat diketahui bahwa konfigurasi router telah berhasil dan sudah memiliki koneksi internet dengan kecepatan 5.37Mbps untuk *download* dan 3.72Mbps untuk *upload* pada jalur *wired* atau kabel LAN. Selanjutnya akan dilakukan konfigurasi terhadap kecepatan *bandwidth* tersebut agar bisa terbagi secara merata. Hal tersebut dapat dilakukan dengan cara klik Tab *Queue Tree*, ganti Name: *Upload*, Parent: *ether1* lalu klik OK.



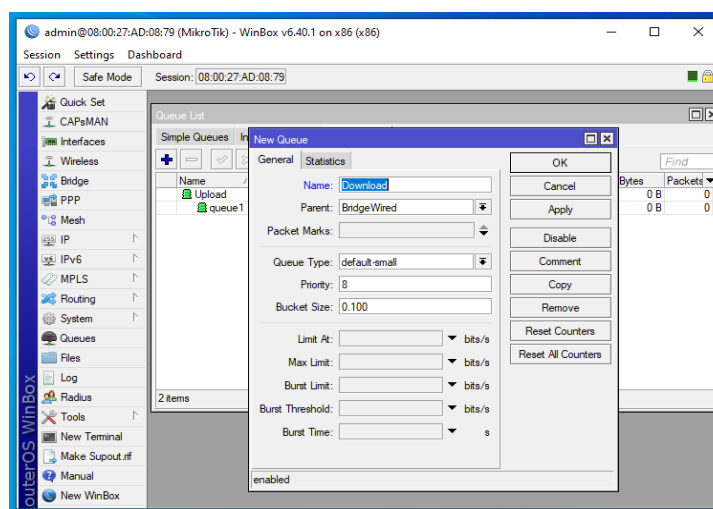
Gambar 4. 35 Konfigurasi *Queue Tree* pada *Ether1*

Selanjutnya klik tombol *add* dan ganti Name: *queue1*, Parent: *Upload*, Packet Mark: *KoneksiLokalWired*, Queue Type: *Upload*, Max Limit: 512k lalu klik OK. Konfigurasi ini akan membatasi kecepatan *upload* pada *queue1* menjadi 512k.



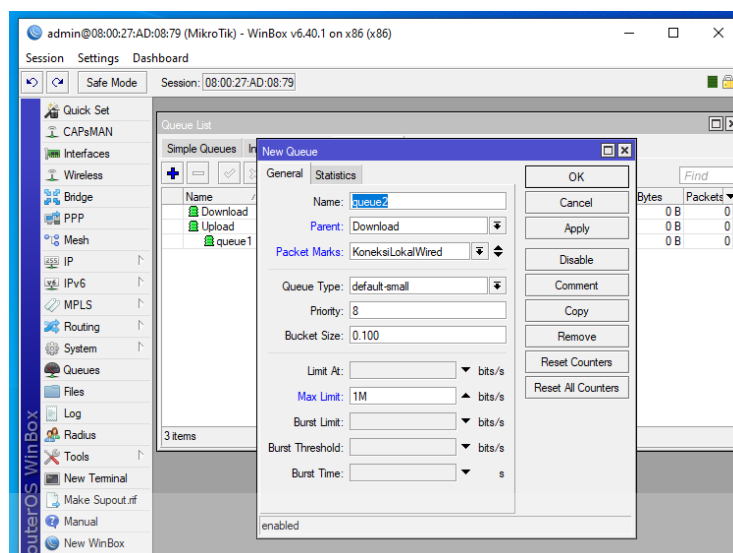
Gambar 4. 36 Konfigurasi *Queue Tree* pada Parent *Upload*

Setelah membuat konfigurasi untuk *upload*, selanjutnya adalah membuat *Queue Tree* untuk *download*. langkah membuat konfigurasi adalah dengan mengganti Name: *Download*, Parent: *BridgeWired* lalu klik OK.



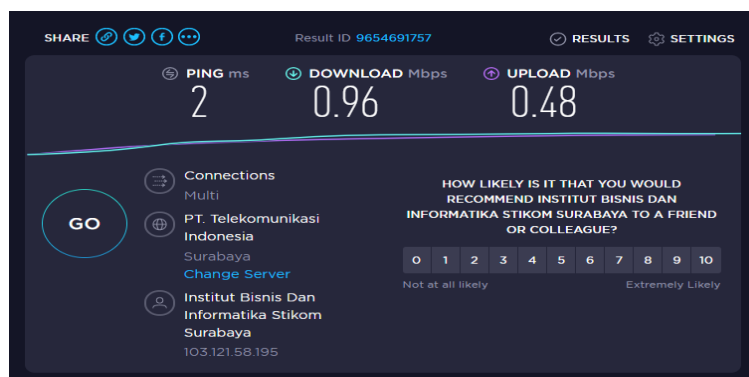
Gambar 4. 37 Konfigurasi *Queue Tree* untuk *Download*

Dan buat cabang *queue* seperti halnya konfigurasi pada *upload* dengan cara klik tombol *add*, kemudian ganti *Name: queue2*, *Parent: Download*, *Packet Mark: KoneksiLokalWired*, *Queue Type: Download*, *Max Limit: 1M* lalu klik OK.



Gambar 4. 38 Konfigurasi *Queue Tree* dengan *Parent Download*

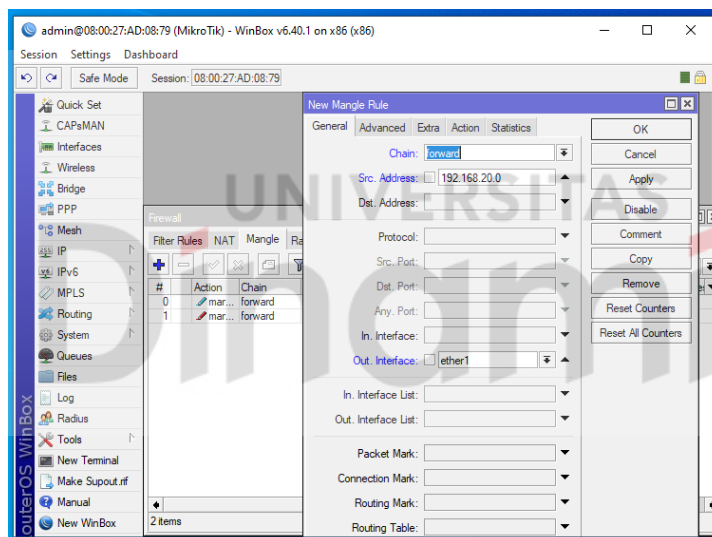
Dengan demikian kecepatan *bandwidth* untuk *download* adalah 1Mbps. Untuk mengetahui keberhasilan konfigurasi yang telah dilakukan, maka buka kembali *browser*, lalu cek kecepatan internet seperti halnya cara yang sebelumnya. Pada Gambar 4.39 di bawah ini menunjukkan bahwa telah berhasil melakukan *limit* alokasi *bandwidth* di jaringan *Wired* sesuai dengan konfigurasi yang dilakukan.



Gambar 4. 39 Pengujian Koneksi *Wired* Setelah Manajemen

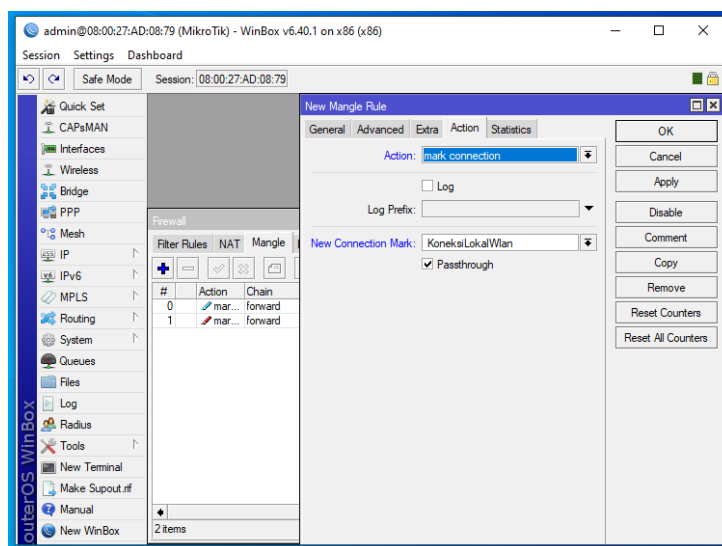
#### 4.8 Setting Limit Alokasi Bandwidth di Jaringan Wireless

Setelah berhasil melakukan konfigurasi *limit* pada jalur kabel, maka yang selanjutnya adalah melakukan konfigurasi pada jalur nirkabel atau wireless. Adapun tahapan dalam konfigurasi ini adalah dengan masuk kedalam menu IP -> Firewall -> Tab Mangle -> tombol *add*. Ganti *Chain: Forward*, *Src. Address: 192.168.20.0/24*, *Out Interface: ether1* lalu klik OK. Dalam konfigurasi ini berfungsi untuk membuat sebuah *interface wireless* dengan memberikan sebuah alokasi untuk IP yang akan digunakan atau diberikan kepada *user* nanti adalah 192.168.20.2 - 192.168.22.254 berdasarkan dari *network* yang telah dibuat.



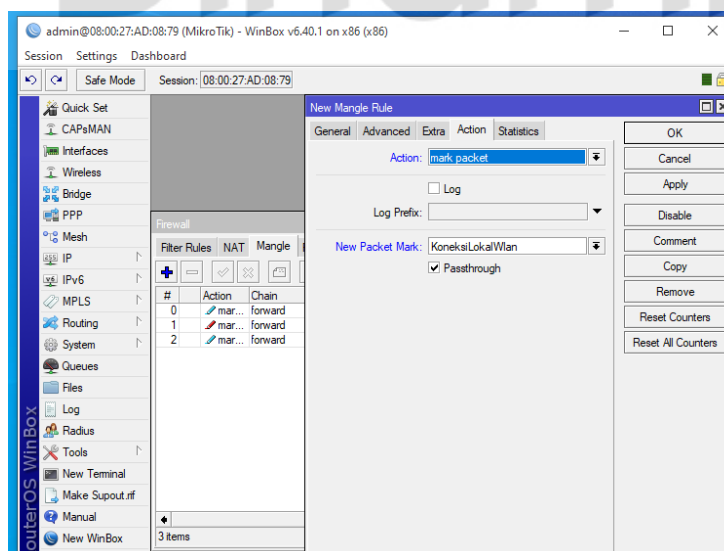
Gambar 4. 40 Konfigurasi *Limit Bandwidth* pada *Ether1*

Seperti halnya koneksi lokal pada jalur kabel, maka perlu dibuat sebuah konek lokal untuk jalur nirkabel, yaitu dengan cara masuk kedalam tab *Action*, kemudian ganti *Action: mark connection*, *New Connection Mark:KoneksiLokalWlan* lalu klik OK.



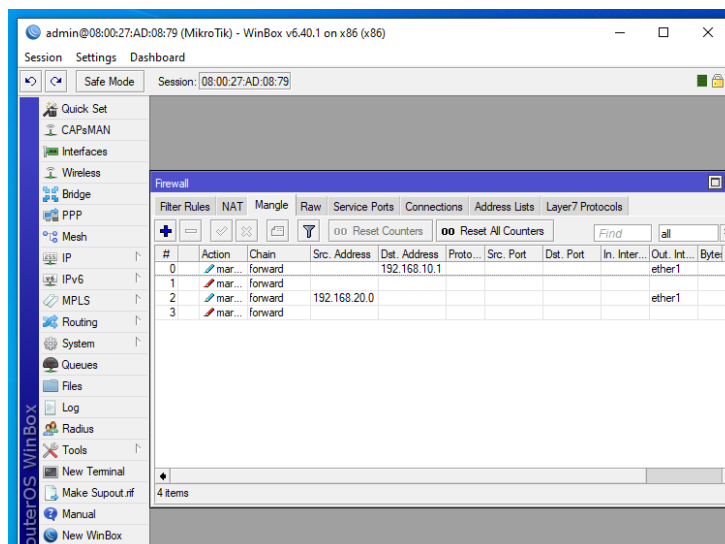
Gambar 4. 41 Konfigurasi *Mark Connection* pada *KoneksiLokalWlan*

Serta lakukan beberapa konfigurasi untuk *mark packet*, yaitu dengan masuk ke dalam Tab *Mangle*, klik tombol *add* dan ganti *chain: forward*, *Connection Mark:KoneksiLokalWlan*. Kemudian pilih Tab *Action*, ganti *Action: mark packet*, *New Packet Mark: KoneksiLokalWlan* lalu klik *OK*.



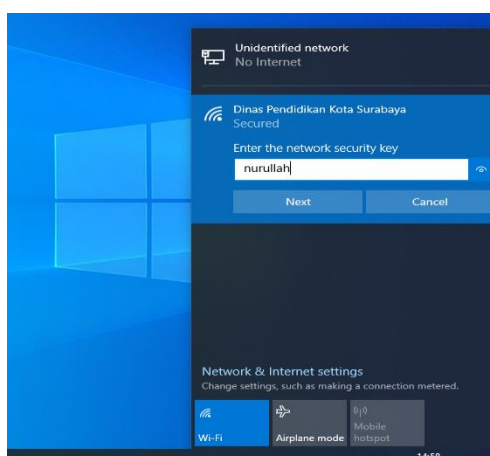
Gambar 4. 42 Konfigurasi *Mark Packet* pada *KoneksiLokalWlan*

Hasilnya akan tampak seperti pada Gambar 4.43 berikut jika setiap konfigurasi telah berhasil dilakukan dengan benar.



Gambar 4. 43 Hasil Konfigurasi *KoneksiLokalWlan*

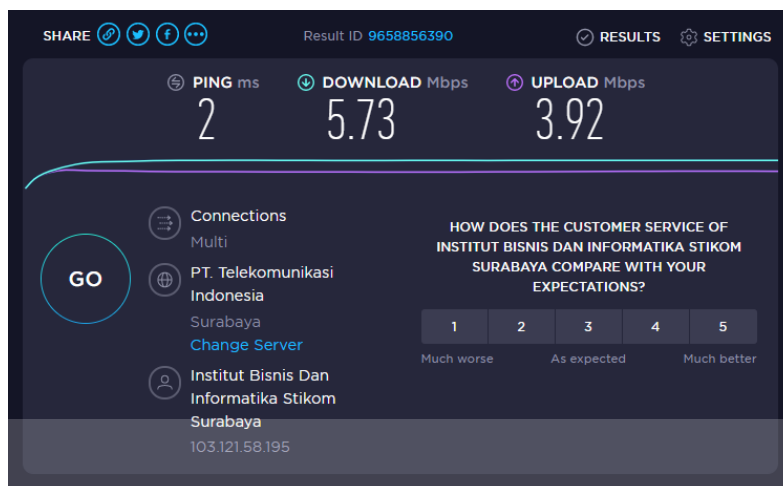
Seperti halnya dengan jalur kabel yang sebelumnya, maka diperlukan sebuah pengujian terhadap koneksi internya. Dimana pada konfigurasi kali ini adalah konfigurasi untuk wireless maka perlu menghubungkan perangkat PC dengan WiFi yang telah dibuat sebelumnya. Nama SSID yang telah dibuat sebelumnya adalah Dinas Pendidikan dengan *password* nurullah.



Gambar 4. 44 Menghubungkan *Wireless* dari PC

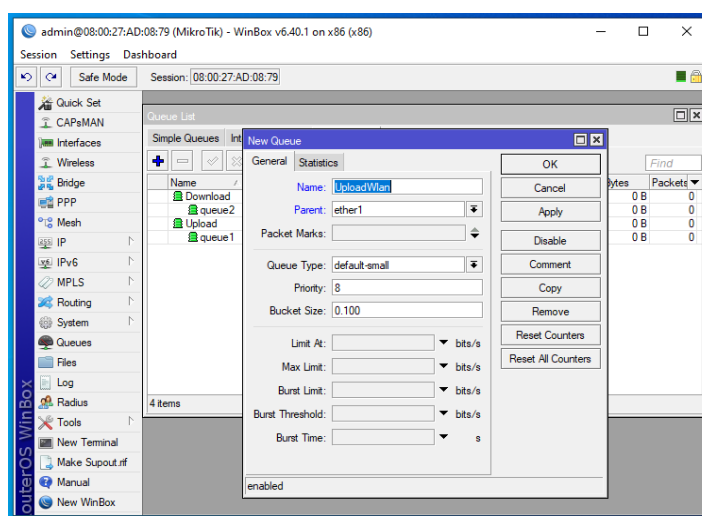


Bila telah berhasil kedalam jaringan *wireless*, maka lakukan pengujian kecepatan koneksinya. Pada Gambar 4.45 dapat diketahui bahwa pada jaringan *wireless* memiliki kecepatan *download* sebesar 5.73 Mbps dan untuk kecepatan *upload* sebesar 3.92 Mbps.



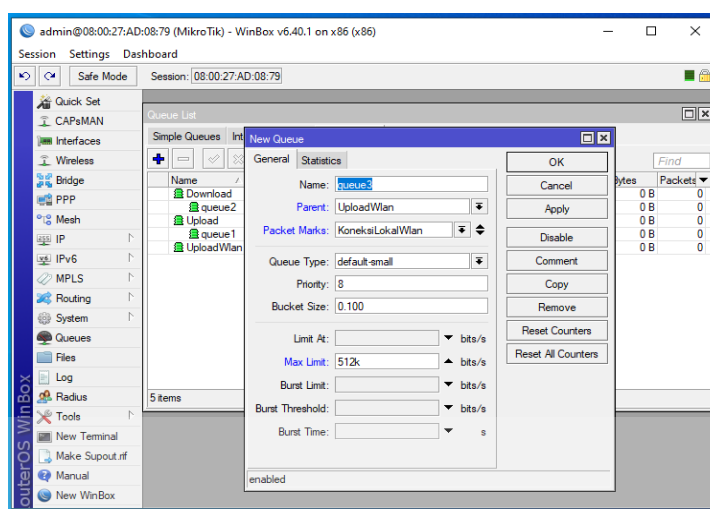
Gambar 4. 45 Pengujian Koneksi *Wireless* Sebelum Manajemen

Untuk melakukan pemerataan *bandwidth* maka perlu dilakukan pengaturan *queue* pula pada jaringan *wireless*, yaitu dengan cara masuk pada menu *Queues* -> tab *Queue Tree* -> tombol *add*, lalu ganti *Name: UploadWlan*, *Parent: ether1* kemudian klik OK.



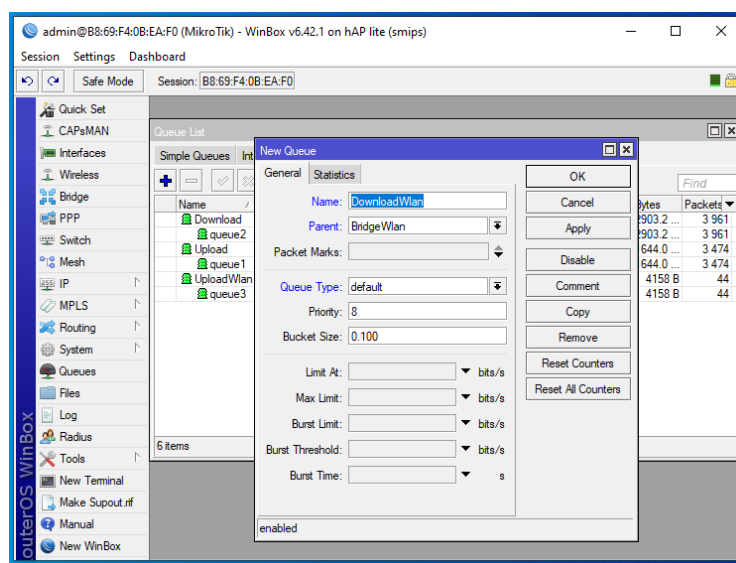
Gambar 4. 46 Konfigurasi *Queues* untuk *UploadWlan*

Selanjutnya buat batas *limit* untuk *upload* pada jalur nirkabel dengan melakukan konfigurasi pada *queue*. Adapun konfigurasinya adalah dengan klik tombol *add*, kemudian ganti *Name: queue3*, *Parent: UploadWlan*, *Packet Marks: KoneksiLokalWlan*, *Queue Type: Upload*, *Max Limit: 512k* lalu klik OK.



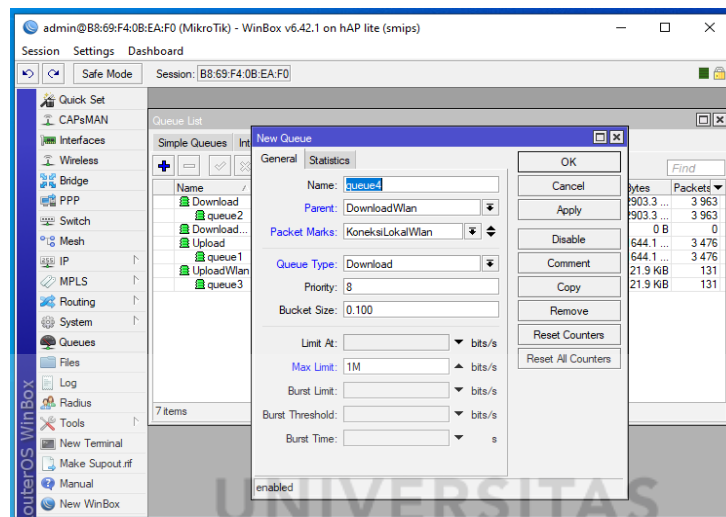
Gambar 4. 47 Konfigurasi *Queues* dengan *Parent UploadWlan*

Setelah itu tambahkan juga queue untuk download pada jalur nirkabel dengan mengklik tanda *add* lalu ganti *Name: DownloadWlan*, *Parent: BridgeWlan*, lalu klik OK.



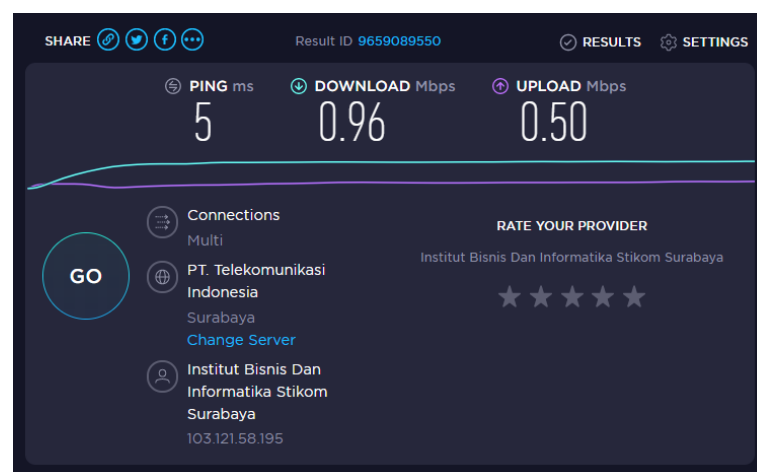
Gambar 4. 48 Konfigurasi *Queues* untuk *DownloadWlan*

Dan buat batas *limit* juga untuk *download* dengan cara klik tombol *add*, ganti *Name*: *queue4*, *Parent*: *DownloadWlan*, *Packet Mark*: *KoneksiLokalWlan*, *Queue Type*: *Download*, *Max Limit*: 1M lalu klik OK. Dengan demikian telah dilakukan konfigurasi batas *limit* terhadap jalur nirkabel atau wireless dengan batas *upload* sebesar 512k dan batas *download* sebesar 1Mb.



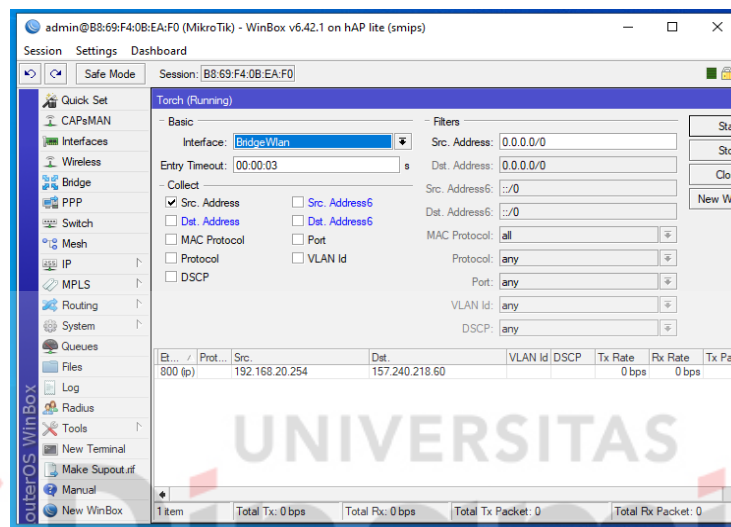
Gambar 4. 49 Konfigurasi *Queues* dengan *Parent DownloadWlan*

Untuk mengetahui tingkat keberhasilan atau perubahan dari konfigurasi, maka lakukan pengecekan kecepatan internet pada *user* di jaringan *Wireless* seperti sebelumnya. Pada Gambar 4.50 terlihat bahwa batas *limit* telah berhasil diterapkan.



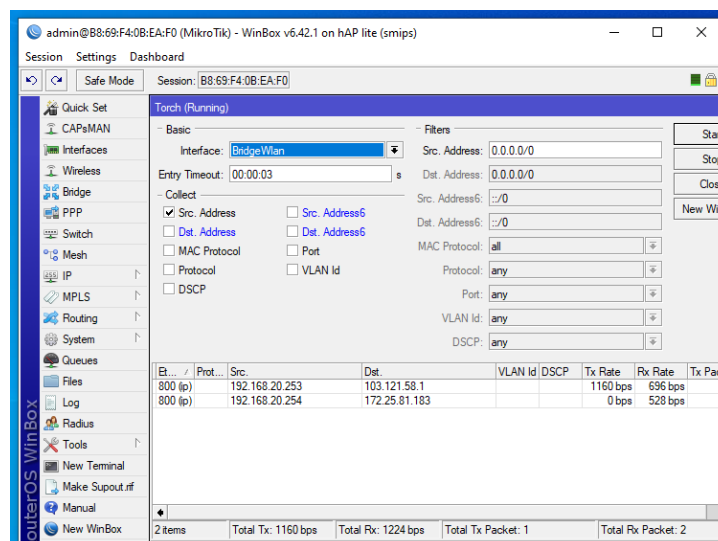
Gambar 4. 50 Pengujian Koneksi *Wireless* Setelah Manajemen

Dari beberapa proses dan hasil yang didapatkan bisa diketahui bahwa proses manajemen *bandwidth* dengan pembatasan dan pembagian *bandwidth* telah berhasil dan dapat diimplementasikan kepada *user*. Dimana proses untuk mengecek apakah fungsi manajemen *bandwidth* berjalan dengan lancar maka dapat dilakukan pengujian dengan menghubungkan *user* pada jaringan *Wireless*.



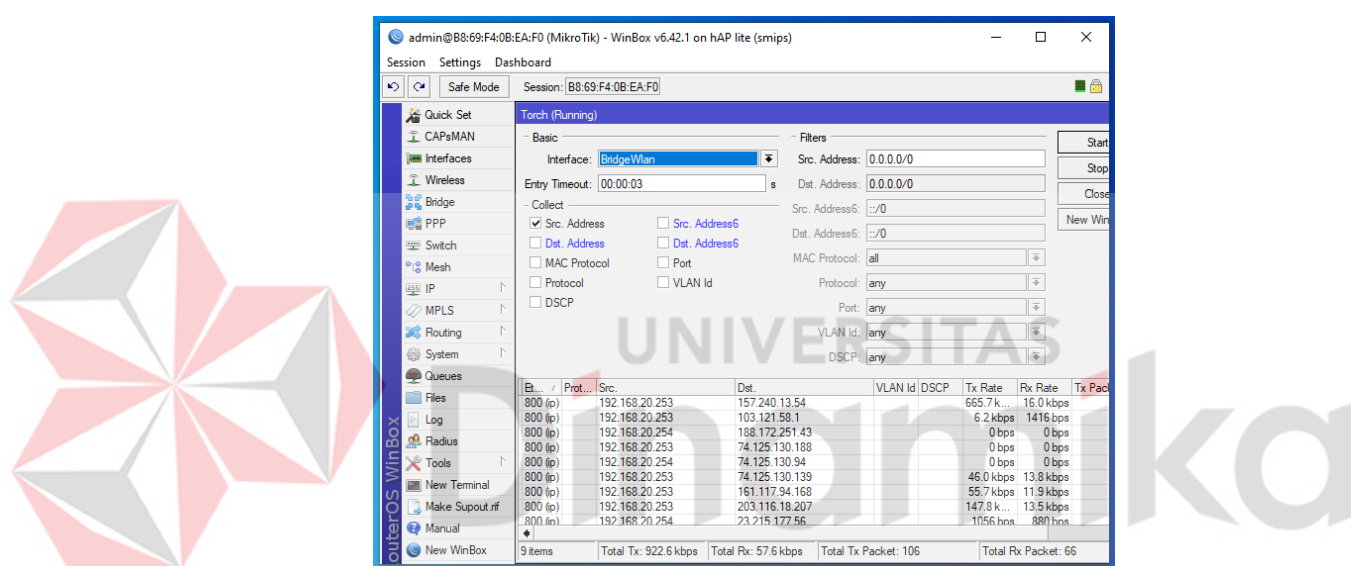
Gambar 4. 51 Satu User Terhubung Pada Jaringan Wireless

Dengan menggunakan tab *torch* dapat diamati bahwa pada Gambar 4.51 terlihat satu *user* di jaringan *Wireless* dengan dengan IP DHCP 192.168.20.254.



Gambar 4. 52 Dua User Terhubung Pada Jaringan Wireless

Dan dapat diamati pula bahwa pada Gambar 4.52 terdapat dua user dengan IP DHCP yang didapat 192.168.20.253 dan 192.168.20.254 serta memiliki kecepatan *bandwidth* dibawah 1Mb, dimana hal tersebut sudah sesuai dengan manajemen *bandwidth* yang sudah kita buat sebelumnya. Untuk melakukan pengujian lebih lanjut maka dilakukan proses browsing ataupun searching pada koneksi internet untuk mengetahui kinerja dari batas *limit* yang telah dibuat sebelumnya. Hal tersebut dapat diamati pada Gambar 4.53.



Gambar 4. 53 *Traffic Bandwidth* Pada Dua User

Dari Gambar 4.53 terlihat bahwa *bandwidth* dapat terbagi dengan rata sesuai dengan batas *limit* yang telah dilakukan. Selanjutnya akan dilakukan pula pengujian terhadap batas *limit* terhadap jaringan internet yang menggunakan kabel atau LAN. Dengan menggunakan proses dan perangkat yang sama tetapi memakai jalur koneksi yang berbeda. Hasil dari pengujian tersebut dapat diamati pada Gambar 4.54.

Et.	Prot.	Src.	Dst.	VLAN Id	DSCP	Tx Rate	Rx Rate	Tx Packet
800 (ip)		192.168.20.254	74.125.24.190			131.8 k...	10.0 kbps	
800 (ip)		192.168.20.254	74.125.24.119			415.7 k...	16.6 kbps	
800 (ip)		192.168.20.254	103.121.58.1			0 bps	680 bps	
800 (ip)		192.168.20.254	103.121.58.2			0 bps	0 bps	
800 (ip)		192.168.20.254	118.98.30.207			221.3 k...	13.6 kbps	
800 (ip)		192.168.20.254	74.125.130.94			528 bps	440 bps	
800 (ip)		192.168.20.254	172.217.194.132			228.3 k...	11.9 kbps	

7 items Total Tx: 997.8 kbps Total Rx: 53.2 kbps Total Tx Packet: 96 Total Rx Packet: 86

Gambar 4. 54 Batas *Limit* Telah Berjalan Pada Dua *User*

Dari Gambar 4.54 dan Gambar 4.55 terlihat bahwa *user* dapat menggunakan *bandwidth* secara penuh pada jaringan *Wired* dan dapat diamati bahwa *bandwidth* dapat terbagi secara merata pada kedua *user* di jaringan *Wired*.

Et.	Prot.	Src.	Dst.	VLAN Id	DSCP	Tx Rate	Rx Rate	Tx Packet
800 (ip)		192.168.20.254	118.98.30.207			988.2 k...	19.4 kbps	
800 (ip)		192.168.20.254	74.125.130.94			0 bps	0 bps	
800 (ip)		192.168.20.253	103.121.58.1			0 bps	0 bps	
800 (ip)		192.168.20.253	74.125.68.95			0 bps	0 bps	
800 (ip)		192.168.20.253	161.117.94.168			0 bps	0 bps	
800 (ip)		192.168.20.254	118.98.30.208			528 bps	440 bps	
800 (ip)		192.168.20.253	157.240.218.61			3.2 kbps	2.4 kbps	
800 (ip)		192.168.20.254	74.125.24.190			3.7 kbps	7.5 kbps	

7 items Total Tx: 400.6 kbps Total Rx: 10.7 kbps Total Tx Packet: 40 Total Rx Packet: 23

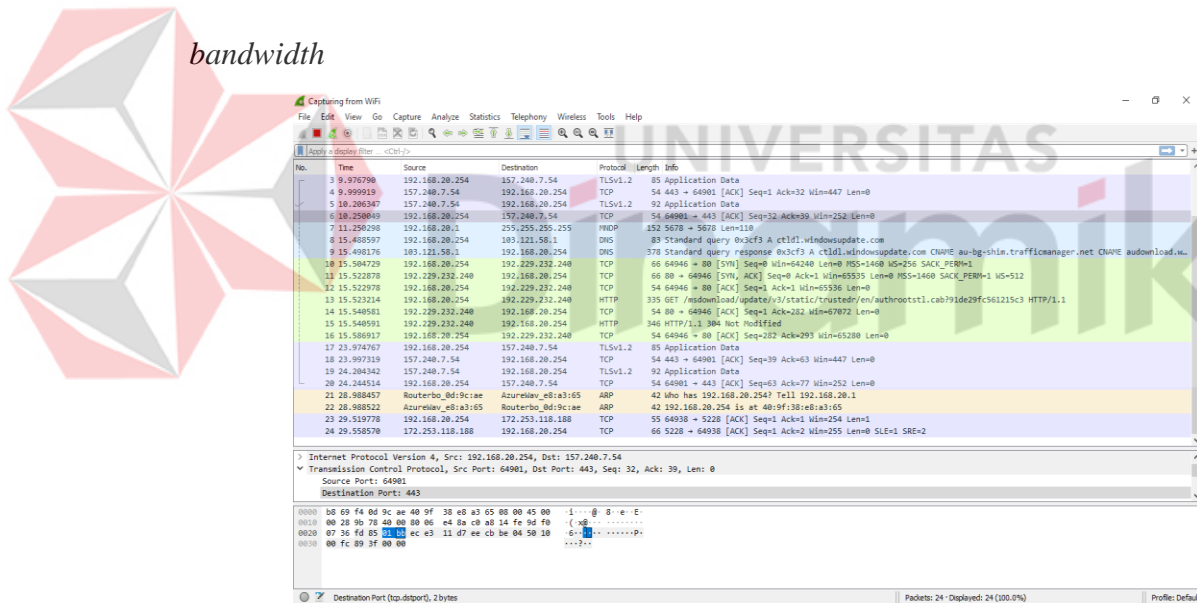
Gambar 4. 55 *Bandwidth* Dapat Terbagi Rata

## 4.9 Pengujian Menggunakan Wireshark

Pada tahap pengujian ini akan dilakukan dengan menggunakan *wireshark* agar mendapatkan data yang lebih spesifik dalam melakukan perhitungan, perbandingan atau analisa terhadap penggunaan manajemen *bandwidth* dan tanpa penggunaan manajemen *bandwidth*.

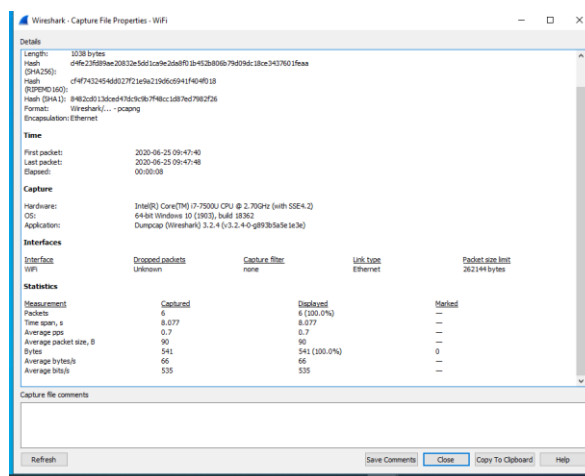
Pada proses pengujian yang pertama adalah analisa dari jaringan router yang sudah dapat terhubung kedalam jaringan internet, namun belum dilakukan konfigurasi terhadap manajemen *bandwidth* atau memberi batas *limit* dan pembagian *bandwidth*. Hal tersebut dapat diamati dari data yang diambil dari *wireshark* saat perangkat terhubung pada router sebelum dilakukan manajemen

*bandwidth*



Gambar 4. 56 Hasil *Wireshark* Tanpa Manajemen *Bandwidth*

Pada Gambar 4.56 diatas merupakan data dari jaringan *Wireless* dengan menggunakan *Wireshark* tanpa manajemen *bandwidth Queue Tree* dan *PCQ*. Dikarenakan data dari *wireshark* terlalu banyak maka dengan menggunakan fungsi *capture* pada tab *option wireshark* didapatkan data yang lebih spesifik.

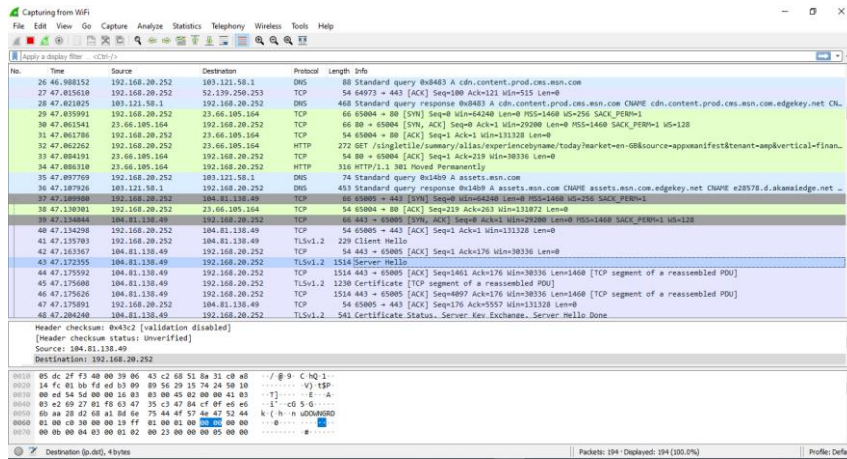


Gambar 4. 57 Hasil *Capture Wireshark* Tanpa Manajemen *Bandwidth*

Dapat diamati pada Gambar 4.57 yang merupakan hasil *capture* data dari *Wireshark* dengan koneksi internet tanpa menggunakan Manajemen *Bandwidth Queue Tree* dan PCQ.

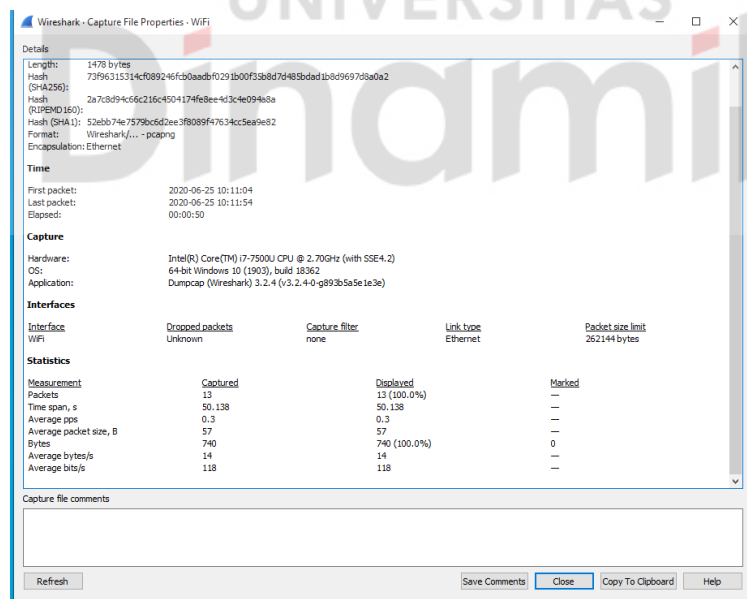
Selanjutnya untuk proses pengujian yang kedua adalah analisa dari jaringan router yang sudah dapat terhubung kedalam jaringan internet dan juga telah dilakukan konfigurasi terhadap manajemen *bandwidth* atau memberi batas *limit* dan pembagian *bandwidth*. Hal tersebut dapat diamati dari data yang diambil dari *wireshark* saat perangkat terhubung pada router setelah dilakukan manajemen *bandwidth* sebagai bahan perbandingan untuk analisa. Dapat diamati pada Gambar 4.58 yang merupakan hasil *wireshark* dengan manajemen *bandwidth*.





Gambar 4. 58 Hasil *Wireshark* dengan Manajemen *Bandwidth*

Serta hasil *capture wireshark* dengan manajemen *bandwidth* untuk proses analisa yang lebih spesifik pada data tertentu.



Gambar 4. 59 Hasil *Capture Wireshark* dengan Manajemen *Bandwidth*

Data yang diambil dari *capture* pada *wireshark* yang merupakan nilai dari pengiriman sebuah paket data atau kecepatan sebuah perangkat dalam melakukan sebuah koneksi internet atau secara ringkasnya adalah kecepatan sebuah data dapat terkirim melalui koneksi tersebut. Data didapatkan dari *capture* sebanyak 5 kali pengambilan data seperti halnya yang dapat diamati pada Gambar 4.57 dan Gambar 4.59 sebelumnya. Adapun data tersebut dapat diamati pada Tabel 4.1.

Tabel 4. 1 Data *Transfer Rate* dari *Wireshark*

Paket ke-	Tanpa <i>Queue Tree</i> dan PCQ (dalam <i>second</i> )	Dengan <i>Queue Tree</i> dan PCQ (dalam <i>second</i> )
1	1.375233	1.178492
2	1.537823	1.419270
3	1.578931	1.436903
4	1.730933	1.570238
5	1.752012	1.592501
Jumlah	7.974932	7.197404
Total variasi	376.779	414,009

Dalam Tabel 4.1 juga terdapat total variasi yang berasal dari variasi tiap paket data, dimana total variasi ini merupakan hasil dari penjumlahan data antar paket dengan rumus perhitungan  $(\text{paket ke } 2 - \text{paket ke } 1) + (\text{paket ke } 3 - \text{paket ke } 2)$ , dan seterusnya.

#### 4.9.1 *Delay*

*Delay* merupakan keterlambatan dalam waktu transmisi data dari pengirim dan penerima, satuan dari *delay* adalah sekon (detik). Dalam pengujian ini akan dilakukan perhitungan terhadap *delay* tersebut. Hasil pengujian *delay* tanpa menggunakan Manajemen *Bandwidth*:

$$\begin{aligned}
 \text{Rata-rata } \textit{delay} &= \text{Total } \textit{delay} / \text{Total paket yang diterima} \\
 &= 7.974932 / 14451 \\
 &= 0.00055186 \text{ s} \\
 &= 0,55186 \text{ ms}
 \end{aligned}$$

Total *delay* didapat dengan cara menjumlahkan keseluruhan *delay* yang ada antara paket satu dengan paket lainnya. Tabel menunjukkan hasil perhitungan rata-rata *delay* sebelum menggunakan *Queue Tree* dan PCQ.

Tabel 4. 2 Perhitungan *Delay* sebelum manajemen

Parameter	Nilai
Total paket yang diterima	14451 <i>packet</i>
Total <i>delay</i>	7.974932 s
Rata-rata <i>delay</i>	0,55186 ms

Hasil pengujian *delay* dengan menggunakan Manajemen *Bandwidth*:

$$\begin{aligned}
 \text{Rata-rata } \textit{delay} &= \text{Total } \textit{delay} / \text{Total paket yang diterima} \\
 &= 7.197404 / 24649 \\
 &= 0.000292 \text{ s} \\
 &= 0,292 \text{ ms}
 \end{aligned}$$

Total *delay* didapat dengan cara menjumlahkan keseluruhan *delay* yang ada antara paket satu dengan paket lainnya. Tabel menunjukkan hasil perhitungan rata-rata *delay* dengan menggunakan *Queue Tree* dan PCQ.

Tabel 4. 3 Perhitungan *Delay* dengan manajemen

Parameter	Nilai
Total paket yang diterima	24649 <i>packet</i>

Total <i>delay</i>	7.197404 s
Rata-rata <i>delay</i>	0,292 ms

Dari pengujian yang dilakukan didapatkan bahwa nilai *delay* antara manajemen *Bandwidth* sebelum dan setelah menggunakan metode *Queue Tree* dan PCQ adalah berbeda dimana *delay* pada manajemen *Bandwidth* tanpa metode *Queue Tree* dan PCQ lebih besar daripada setelah menggunakan metode *Queue Tree* dan PCQ.

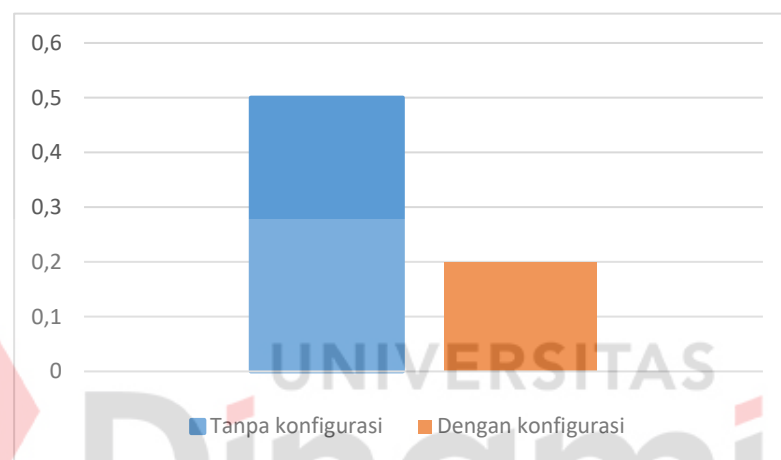


Diagram 4. 1 Diagram Perbandingan *Delay* tanpa dan dengan Manajemen

Bandwidth

Diagram perbandingan *delay* tanpa menggunakan Manajemen *Bandwidth* dengan yang menggunakan Manajemen *Bandwidth* melalui metode *Queue Tree* dan PCQ dapat diamati pada Diagram 4.1.

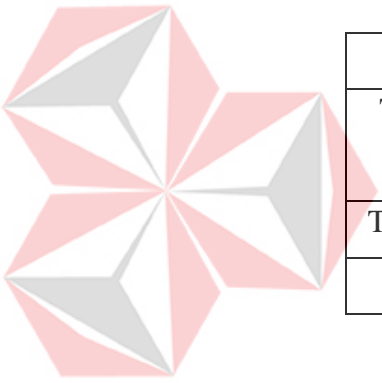
#### 4.9.2 *Jitter*

*Jitter* merupakan variasi dari *delay* atau selisih antara *delay* pertama dengan *delay* selanjutnya. Dalam pengujian ini akan dilakukan perhitungan terhadap *jitter* tersebut. Hasil pengujian *Jitter* tanpa menggunakan metode *Queue Tree* dan PCQ. Berikut perhitungan terhadap nilai *jitter* sebelum manajemen :

$$\begin{aligned}
 \text{Jitter} &= \text{total variasi } \textit{delay} / (\text{total packet yang diterima} - 1) \\
 &= 376,779 / (14451 - 1) \\
 &= 376,779 / 14450 \\
 &= 0,026074671 \text{ s} \\
 &= 26,074671 \text{ ms}
 \end{aligned}$$

Total variasi *delay* didapat dengan cara menjumlahkan keseluruhan selisih *delay* antara paket satu dengan yang lainnya. Tabel menunjukkan hasil perhitungan rata-rata *Jitter* sebelum menggunakan *Queue Tree* dan PCQ.

Tabel 4. 4 Perhitungan Rata-rata *Jitter* sebelum Konfigurasi



Parameter	Nilai
Total paket yang diterima	14451 <i>packet</i>
Total variasi <i>delay</i>	376,779 s
<i>Jitter</i>	26,074671 ms

Selanjutnya akan kita lakukan pengujian *Jitter* dengan menggunakan Manajemen *Bandwidth* melalui metode *Queue Tree* dan PCQ. Berikut perhitungan nilai *jitter* setelah manajemen :

$$\begin{aligned}
 \text{Jitter} &= \text{Total variasi } \textit{delay} / (\text{total packet yang diterima} - 1) \\
 &= 414,009 / (24649 - 1) \\
 &= 414,009 / 24648 \\
 &= 0.01679686 \text{ s} \\
 &= 16,79686 \text{ ms}
 \end{aligned}$$

Total variasi *delay* didapat dengan cara menjumlahkan keseluruhan selisih *delay* antara paket satu dengan yang lainnya. Tabel menunjukkan hasil rata-rata *Jitter* setelah menggunakan *Queue Tree* dan PCQ.

Tabel 4. 5 Perhitungan *Jitter* Setelah Manajemen

Parameter	Nilai
Total paket yang diterima	24649 <i>packet</i>
Total variasi <i>delay</i>	414,009 s
<i>Jitter</i>	16,79686 ms

Dari pengujian yang dilakukan didapatkan bahwa nilai *Jitter* antara manajemen *Bandwidth* sebelum dan setelah menggunakan metode *Queue Tree* dan PCQ adalah berbeda dimana *Jitter* pada manajemen *Bandwidth* tanpa metode *Queue Tree* dan PCQ lebih besar daripada setelah menggunakan metode *Queue Tree* dan PCQ.

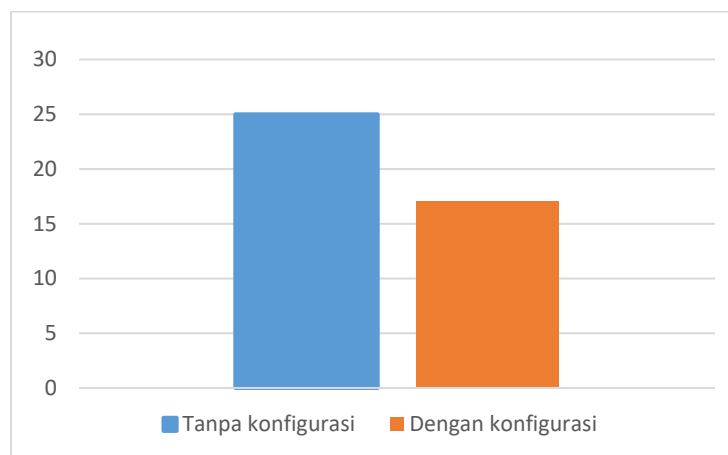


Diagram 4. 2 Perbandingan *Jitter* dengan dan tanpa Konfigurasi

Diagram perbandingan *Jitter* tanpa menggunakan metode *Queue Tree* dan PCQ dengan yang menggunakan metode *Queue Tree* dan PCQ dapat diamati pada Diagram 4.2.

### 4.9.3 Throughput

*Throughput* adalah kecepatan transfer data yang diukur dalam bps. *Throughput* merupakan jumlah total kedatangan paket yang sukses yang diamati pada destination selama interval waktu tersebut.

Pengujian *Throughput* tanpa menggunakan metode *Queue Tree* dan PCQ.

$$\begin{aligned}
 \textit{Throughput} &= \text{Paket yang diterima/lama pengamatan} \\
 &= 14451 / 72,688 \\
 &= 198.809 \textit{ bytes/s} \\
 &= 0.19414903 \textit{ kbps}
 \end{aligned}$$

Tabel menunjukkan hasil perhitungan *Throughput* sebelum menggunakan *Queue Tree* dan PCQ.

Tabel 4. 6 perhitungan *Throughput* sebelum manajemen

Parameter	Nilai
Total paket yang diterima	14451 <i>packet</i>
Total variasi <i>delay</i>	72,688 s
<i>Jitter</i>	0.19414903 kbps

Pengujian *Throughput* dengan menggunakan metode *Queue Tree* dan PCQ.

$$\textit{Throughput} = \text{Paket yang diterima/lama pengamatan}$$

$$= 24649 / 213,290$$

$$= 115.566 \text{ bytes/s}$$

$$= 0.112857 \text{ kbps}$$

Tabel menunjukkan hasil perhitungan *Throughput* sebelum menggunakan *Queue Tree* dan PCQ

Tabel 4. 7 hasil perhitungan *Throughput* dengan manajemen

Parameter	Nilai
Total paket yang diterima	24649 <i>packet</i>
Total variasi <i>delay</i>	213,290 s
<i>Jitter</i>	0.112857 kbps

Dari pengujian yang dilakukan didapatkan *Throughput* yang dihasilkan menurun namun tidak terlalu jauh berbeda karena pembatasan *Bandwidth* bagi setiap *user* yang menyebabkan kecepatan transfer data menjadi menurun. Berdasarkan tabel pengujian diatas dapat disimpulkan bahwa nilai *delay*, *Jitter* dan *troughput* dengan menggunakan metode *Queue Tree* dan PCQ lebih kecil dibandingkan tanpa menggunakan metode *Queue Tree* dan PCQ. Hal ini disebabkan karena pengalokasian *bandwidth* untuk semua *user*.



## BAB V

### PENUTUP

#### 5.1 Kesimpulan

Kesimpulan yang diperoleh dalam proses optimalisasi manajemen *Bandwidth* menggunakan *Queue Tree* dengan metode *Peer Connection Queue* pada ruang lingkup Dinas Pendidikan Kota Surabaya sebagai berikut :

1. Mikrotik merupakan sebuah router yang dapat melakukan konfigurasi secara efisien dalam melakukan manajemen *bandwidth*. Serta memiliki tingkat keamanan yang lebih tinggi jika dibandingkan dengan router pada umumnya.
2. Manajemen *bandwidth* dapat dilakukan pada berbagai tipe koneksi internet, baik koneksi secara *wired* maupun koneksi secara *wireless*.
3. Pada penerapannya, manajemen *bandwidth* dengan menggunakan *simple queue* dan juga PCQ memiliki beberapa keunggulan selain membagi rata setiap *bandwidth* serta dapat memberikan batasan *limit* kepada setiap *user*, yakni memiliki *transfer rate* data yang lebih cepat.
4. PCQ juga dapat bekerja dengan baik pada mikrotik dijalur *wired* maupun *wireless*, sehingga setiap *user* bisa mendapatkan alokasi *bandwidth* secara merata.
5. Nilai *delay*, *Jitter* dan *throughput* dengan menggunakan metode *Queue Tree* dan PCQ lebih kecil jika dibandingkan tanpa menggunakan metode *Queue Tree* dan PCQ.

## 5.2 Saran

Saran dari dalam proses optimalisasi manajemen *Bandwidth* menggunakan *Queue Tree* dengan metode *Peer Connection Queue* pada ruang lingkup Dinas Pendidikan Kota Surabaya sebagai berikut :

1. Metode tersebut masih memiliki kelemahan dalam implementasi manajemen *bandwidth*, terutama pada pembagian alokasi *bandwidth* jika salah satu *host* dapat menggunakan *bandwidth* secara maksimal sehingga menyebabkan pada alokasi *bandwidth host* lain tidak bisa digunakan.
2. Mencoba untuk menggunakan metode lain dalam melakukan manajemen *bandwidth* lain yang lebih baik lagi.



UNIVERSITAS  
**Dinamika**

## DAFTAR PUSTAKA

Ardiansa, G. F. (2017). Manajemen Bandwidth dan Manajemen Pengguna pada Jaringan Wireless Mesh Network dengan Mikrotik. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 10.

Lisawita. (2006). Manajemen Bandwidth Menggunakan Metode Hierarchical Token Bucket. *Jurnal Teknologi Informasi & Komunikasi Digital Zone*, 8.

Rahmadi. (2017, Juni 6). *CloudHost*. Retrieved from Mengenal Pengertian Apa itu IP Address: <https://idcloudhost.com/>

Situmorang, H. P. (2019). Implementasi Manajemen Bandwidth Menggunakan Metode Peer Connection Queue Pada Smk Budi Mulia Tangerang. *Jurnal Idealis*, 7.

Strauss. (2006). Internet Of Things. *Jurnal The World*, 12.

Widayanto, A. E. (2018). Manajemen Bandwidth Dengan Simple Queue Di Laboratorium Komputer Universitas Sahid Surakarta. *Jurnal Unsas*, 12.