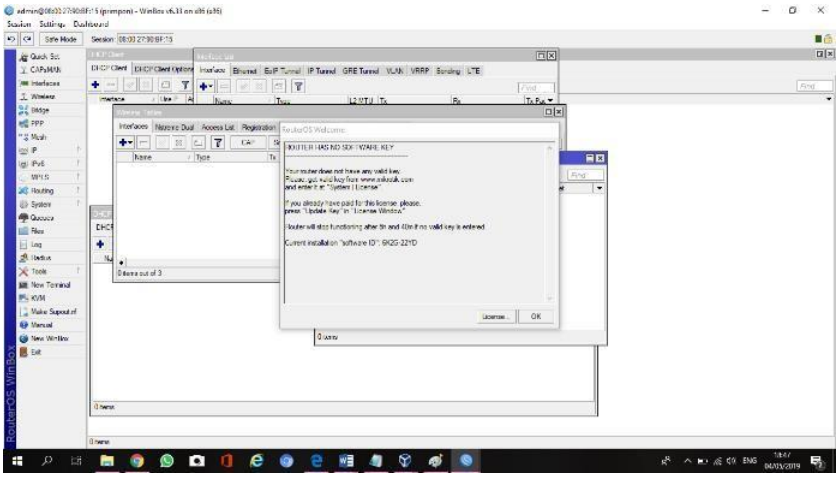
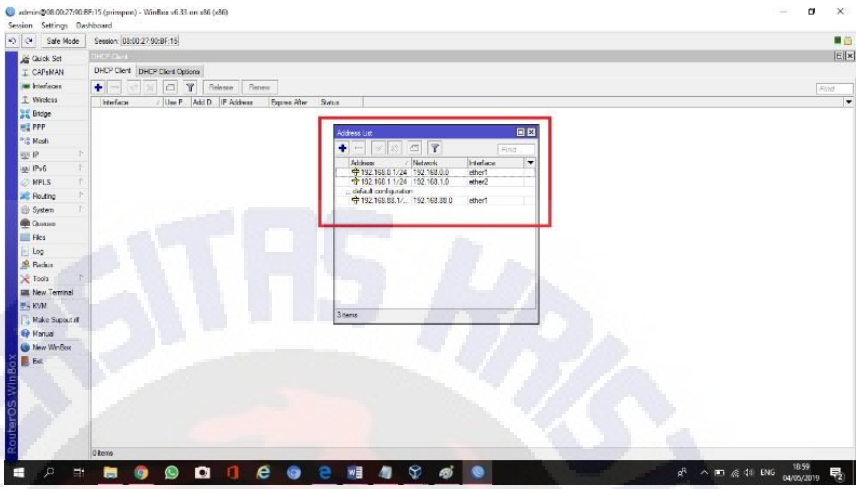


Gambar 4.3. Router OS terhubung

```
[admin@MikroTik] > ip add pr
Flags: X - disabled, I - invalid, D - dynamic
 #    ADDRESS            NETWORK         INTERFACE
 0    192.168.1.1/24     192.168.1.0     ether2
 1    192.168.0.1/24     192.168.0.0     ether1
line 2 of 2> _
```

Gambar 4.4. tampilan IP *Address*

Untuk pemberian IP pada MikroTik, dilakukan secara otomatis *(dhcp)* dan manual *(static)*, namun untuk terkoneksi ke Internet MikroTik menggunakan *dhcp client* karena IP sudah diberikan laboratorium dan jika diganti secara *static* maka tidak bisa terkoneksi ke Internet dan untuk menghindari IP yang sama agar tidak terjadi error . Maka dipakailah konfigurasi *static* dan *manual*



Gambar 4.5. konfigurasi IP pada Mikrotik

Pada gambar di atas adalah IP yang digunakan untuk konfigurasi *server* pada implementasi ini. Pada kolom *interface* terdapat 3 interface yang digunakan yaitu *ether1*, *ether2*, *ether1 (default configuration)*

4.1.3 Pengaturan *IP Route*

Perintah yang diberikan untuk membuat *gateway router* :

```
/ip route add gateway=192.168.0.1
```

Alamat *gateway* dapat dicek menggunakan perintah :

```
/ip route print
```

4.1.4 Pengaturan *IP* DNS

Alamat DNS diberikan untuk dapat mengakses internet.

```
/ip dns set primarydns=192.168.1.47
allow-remoterequest=yes
```

Jika dicek maka terlihat seperti ini :

Servers : 8.8.8.8,8.8.4.4

Dynamic-servers : -

Allow-remote-req : yes

4.1.5 IP DHCP Setup

Untuk *setup* dapat dilakukan dengan menulis perintah :

```
/ip dhcp-server setup
Dhcp server interface : ether2
Dhcp address space : 192.168.1.0/24
Gateway for dhcp network :
192.168.1.1
Addresses to give out : 192.168.1.2-
192.168.1.254
Dns servers : 8.8.8.8,8.8.4.4
Lease time : 3d
```

4.1.6 IP Hotspot

Membuat Hotspot dapat dilakukan dengan perintah sebagai berikut :

```
/ip hotspot setup
Hotspot interface : ether1
Local address of network : 192.168.0.254
Masquerade : yes
Select certificate :import-other-certificate
Passphrase : none
Dns server : 8.8.8.8,8.8.4.4
Dns name : primpon.com
Password for the user : qwerty123
```

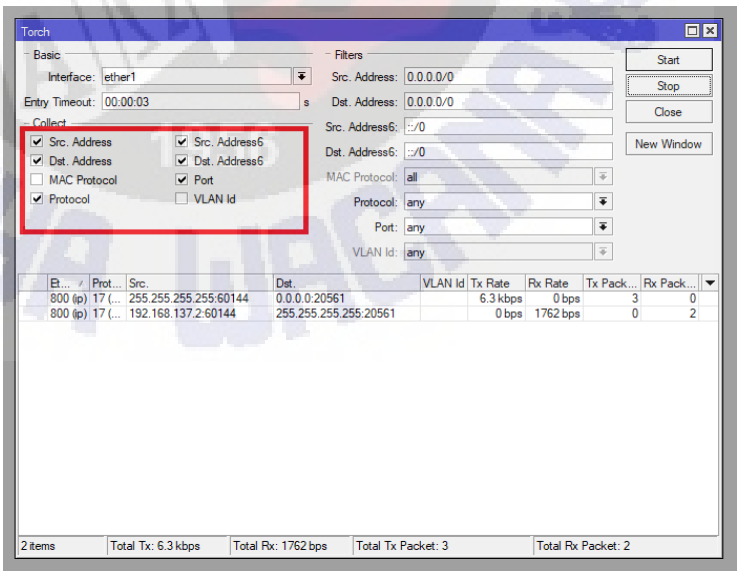
Gambar 4.6. Tampilan *Login*

4.1.7 Pengaturan NAT

Karena pengguna belum dapat terhubung ke jaringan internet publik maka *output* pada MikroTik harus diberikan pengaturan *masquerade. IP Masquerade* adalah salah satu fasilitas di MikroTik yang memungkinkan komputer pengguna terhubung ke internet dengan alamat IP privat melalui MikroTik sebagai penerjemah alamat jaringan. Menggunakan perintah sebagai berikut :

```
/ip firewall NAT add
chain=srnataction=masquerade
Out-interface=ether1
```

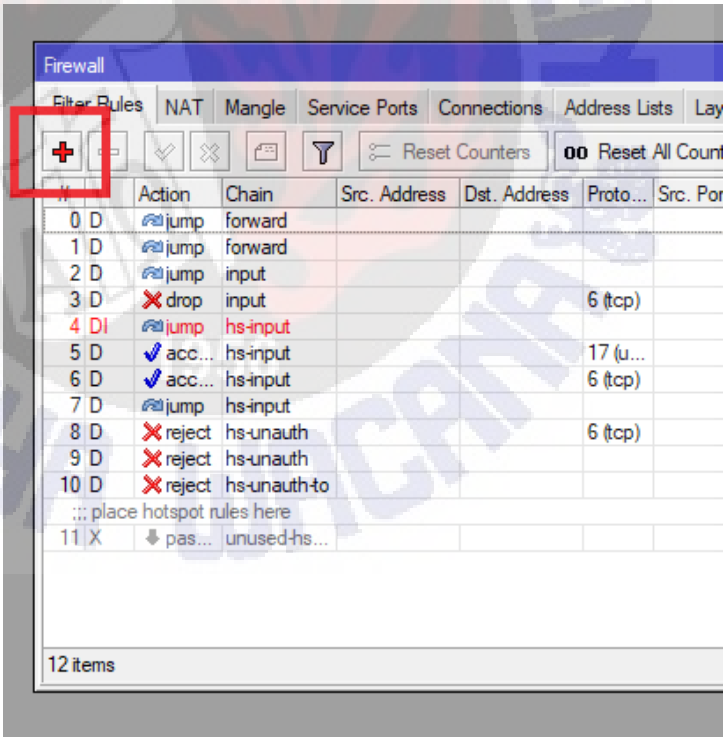
4.1.8 Torch



Gambar 4.7. Tampilan *torch*

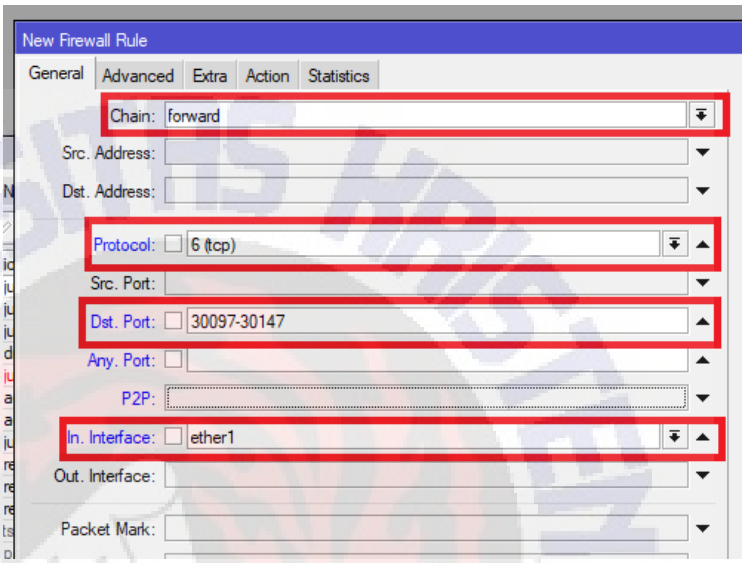
Lakukan *torch* ini pada saat perangkat yang terhubung ke *hotspot* kemudian centang pada *Src.Address, Dst.Address, Protocol, Port, Src.Address6, Dst.Address*. Atau bisa dengan mencari di internet berbagai macam *port-port game online* yang telah tersedia. Dan didapat *port game online* yaitu *Mobile Legend* yang penulis gunakan untuk melakukan ujicoba. Dengan *port game* 30097-30147

4.1.9 Firewall Rule



Gambar 4.8. Tampilan *firewall*

Pilih yang symbol (+) karena akan menambah *firewall* yang baru.

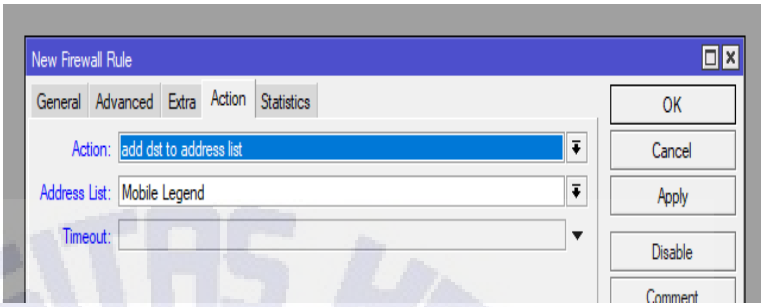


Gambar 4.9. *New firewall rule*

Chain pilih *forward* karena paket berasal dari luar Router dan menuju keluar Router. Dan forward biasanya digunakan untuk memproses trafik paket data yang hanya melewati router.

Untuk *protocol game* tersebut. Bisa TCP atau UDP untuk *port game online* yang tadi sudah didapat di letakan di *Dst.Port* atau port tujuan.

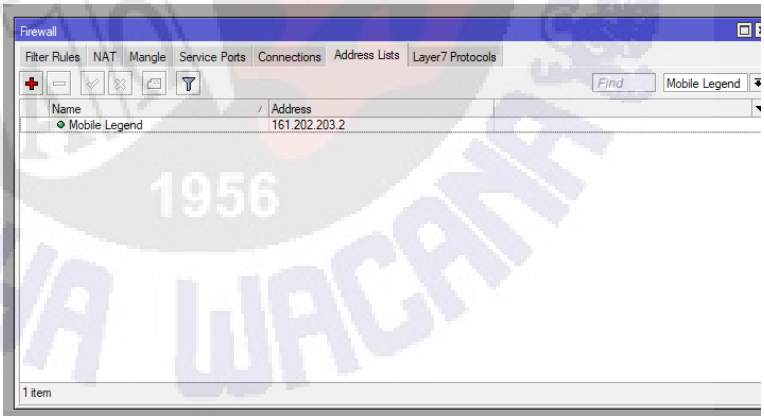
In. Interface pilih yang menuju ke perangkat yang terhubung.



Gambar 4.10. *Action*

Action menggunakan yang *add dst to address list*, fungsinya untuk mendapatkan ip server dari *game* tersebut.

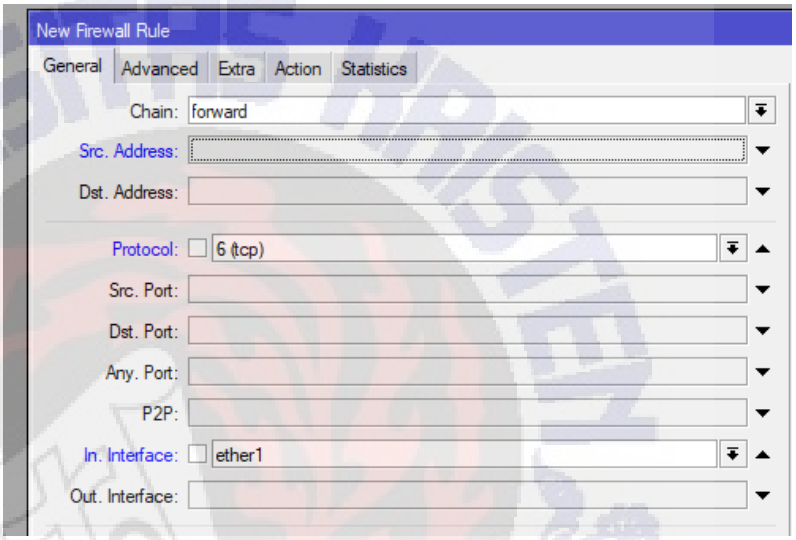
Address list untuk memberi nama pada list IP. Berikut adalah IP server yang didapat untuk *Mobile Legend dapat dilihat dengan cara IP → Firewall → Address List*



Gambar 4.11. *List IP server mobile legend*

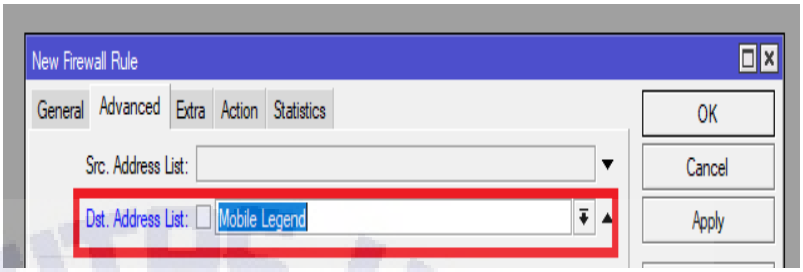
4.1.10 Rule Firewall Filter

Setelah mendapatkan *list IP Mobile Legend*, kemudian membuat *rule firewall filter* untuk memblokir pada hari dan jam yang diinginkan.



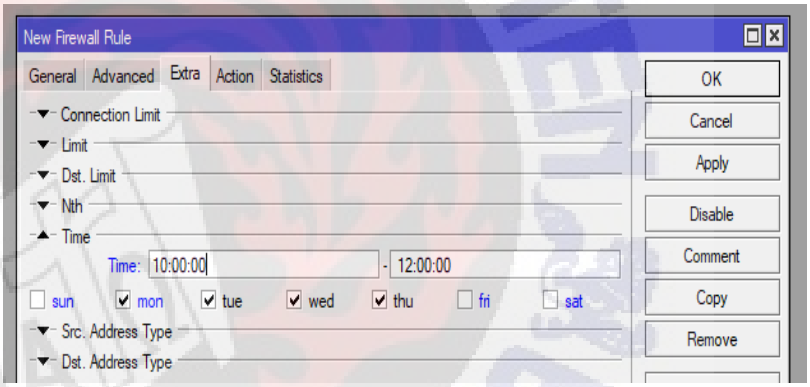
Gambar 4.12. *New firewall filter*

Sama seperti saat membuat *firewall filter*, chainnya masih sama, *Forward*. Karena trafik hanya melewati Router. dari luar Router menuju keluar Router. *Interface* menuju ke *client*. Tetapi untuk *Dst. Port* dikosongkan.



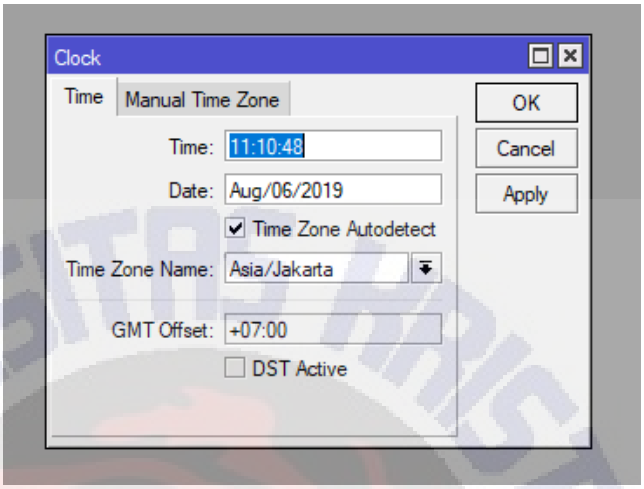
Gambar 4.13. *New firewall advance*

Dst. Address List pilih *list IP* pada *Address List* yang sudah kita buat sebelumnya.



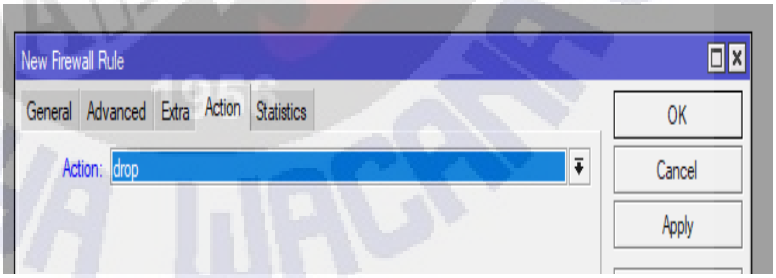
Gambar 4.14. *New firewall extra*

Pilih *extra* untuk menentukan waktu sesuai kebutuhan. Dan *time* pada Router sudah benar dan cocok dengan waktu saat ini. Jika waktu tidak sesuai bisa diatur dengan cara sebagai berikut :



Gambar 4.15. *Clock*

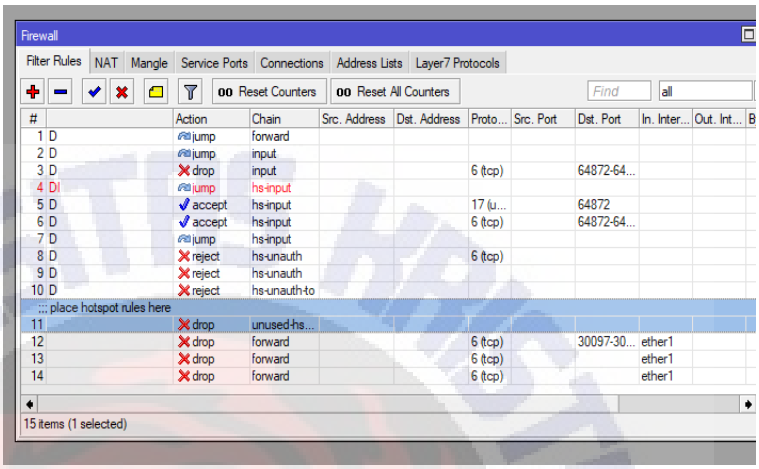
Pengaturan waktu pada Router Mikrotik bisa dilakukan pada *system → Clock karena pada saat Router reboot* waktu pada Router akan kembali ke pengaturan awal.



Gambar 4.16. *Drop*

Action pilih yang *drop*. Karena akan membuang data dari *client* yang dilakukan oleh Router. Yang dilakukan secara diam-diam.

4.1.11 Filter Rules

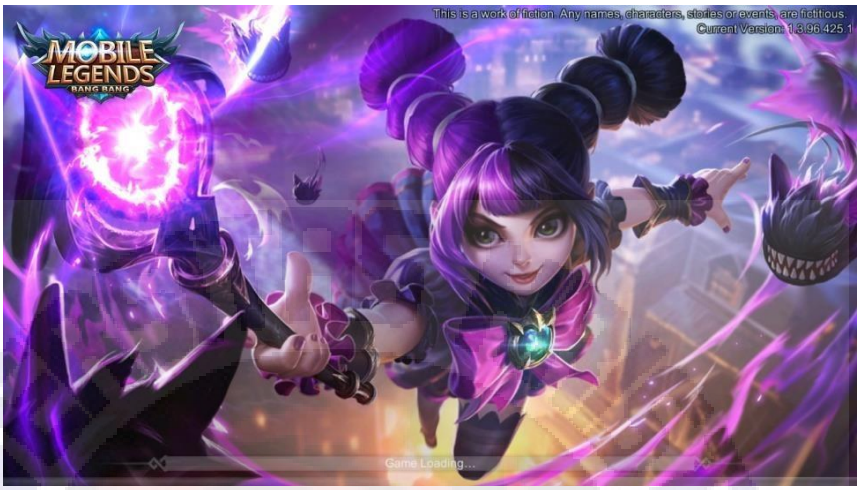


Gambar 4.17. *Firewall*

Daftar tampilan *firewall* yang telah dibuat akan muncul pada *firewall rules.*

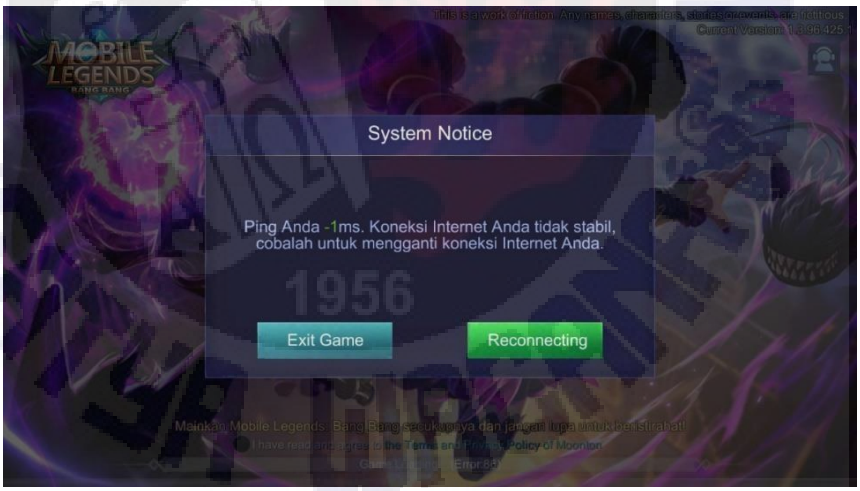
4.2 Hasil Pengujian

Berdasarkan hasil pengujian, sudah dapat berjalan sesuai dengan rancangan. Hasil pengujian menggunakan perangkat *smartphone* yang telah ter-*instal Mobile Legend* yang sudah tersambung ke *Access Point* yang telah dibuat. Dan sebagai berikut :



Gambar 4.18. Proses *Loading*

Jika menjalankan *game* tersebut pertama akan mengalami *loading* yang tidak dapat me-*load* data dari internet.



Gambar 4.19. Eror

Dan tampilan di atas merupakan notice saat Mobile Legend gagal me-load, karena ping atau pergerakan transfer data yang tidak jalan.

4.3 Analisis Sistem

Pada percobaan kali ini yang melakukan blokir *game online* pada *smartphone* yaitu *Mobile Legend* untuk sebuah *client* menggunakan *address list* untuk mencari *port* dari *game online* tersebut. Terlebih dahulu sudah sukses membuat sebuah jaringan menggunakan Rourer OS dan sudah mendapatkan *ip client* secara otomatis dan sudah membuat sebuah server. Dari percobaan yang telah dilakukan, dapat diketahui bahwa dalam cara melakukan blokir *game* yakni menggunakan *firewall filter rules* pada MikroTik. Pada *firewall rule* akan melakukan *drop* pada *port* yang sudah ditemukan dan yang akan diblokir dengan cara melakukan pencarian port dengan menggunakan *torch* pada MikroTik dan pada menu *firewall rule*, *action* nya pilih yang *drop* karena akan membuang data dari *client* yang dilakukan oleh Router. Yang dilakukan secara diam-diam.

Secara topologi memang sangat simpel dan mudah diterapkan oleh siapapun. Tapi secara keseluruhan masih belum dapat mencari *port-port game online* yang ada di perangkat komputer lainnya. Kelemahan lainnya yang masih terjadi, tidak adanya pencarian *port game online* secara otomatis yang mengakibatkan harus meng-input satu per satu. Alangkah lebih baiknya dapat dilakukan pada *game-game online* lainnya yang ada di *smartphone* maupun di komputer. Sistem yang penulis rancang ini sudah dapat digunakan untuk

membatasi mahasiswa untuk bermain *game online* saat jam perkuliahan berlangsung.

UNIVERSITAS KRISTEN
SATYA WACANA
1956