PENERAPAN INTERNET SEHAT SEBAGAI INTERNET SERVICE PROVIDER MENGGUNAKAN NETWORK MONITORING SYSTEM ZABBIX DAN SQUID PROXY

Saleh Dwiyatno[1], **Edy Rakhmat**[2] , **Sulistiyono**[3], **M. Rafi Mahruzzaman**[4]
[1,3,4]Universitas Serang Raya
Jl. Raya Serang Cilegon Drangong Taktakan Kota Serang Banten
[2]Universitas Banten Jaya
Jl. Ciwaru Raya II No. 73 Warung Pojok Cipare Kota Serang Banten

Email : salehdwiyatno*@gmail.com*[1], *edyrakhmat@unbaja.ac.id*[2], *sulistiyonoputro@gmail.com*[3], *rafi.mahruz@gmail.com*[4]

ABSTRACT

Internet users continue to grow. However, there are still many who have not experienced the Internet. One of the contributing factors is the cost of the Internet which is not cheap. Most Internet users get Internet access by subscribing to their respective ISPs. In addition, today more and more crimes are committed via the Internet. This study aims to provide a solution to reduce the cost of subscribing to an ISP and provide healthy Internet access. At Ciruas Center Shops, all traders and buyers use the Internet. Even if it's just for entertainment or even for buying and selling. The system applied in this research is where Internet users can get Internet access only by subscribing to one ISP. That way the fees paid to the ISP can be paid by mutual cooperation. The result of this research is an Internet provider system in which there is Zabbix Network Monitoring System (NMS) software as a network traffic monitor and Squid Proxy as a barrier as well as security when connected to the Internet.

***Keywords** : Internet, ISP, NMS Zabbix, Squid Proxy.*

Pendahuluan

Internet adalah salah satu dampak modernisasi dalam bidang teknologi informasi. Berdasarkan situs wearesocial.com terungkap bahwa pengguna Internet di seluruh dunia pada bulan April 2020 telah mencapai angka 4,57 miliar pengguna, meningkat 7,1% dari tahun lalu. Angka tersebut lebih dari separuh populasi manusia di bumi yang berjumlah 7,77 miliar jiwa (Hidayat et al., n.d.). Namun, banyak sekali penyalahgunaan dalam penggunaan Internet. Mulai dari penyalahgunaan yang merugikan diri sendiri atau orang lain hingga penyalahgunaan yang melanggar hukum.

Untuk mengurangi penyalahgunaan perlu adanya pemantauan pada aktivitas dan perangkat jaringan (Wijayanto & Waspada, 2016). Pemantauan ini bertujuan supaya terciptanya lingkungan dengan *Internet* yang sehat. Pengertian *Internet* sehat sendiri adalah aktivitas *Internet* yang disesuaikan dengan kebutuhan pengguna *Internet* secara kriteria umur, profesi, dan keyakinan yang bertujuan adanya konten yang pas dan tidak melanggar aturan hukum *Cyber* yang berlaku (Purnama, 2019). Saat ini di Pertokoan Ciruas Center belum ada layanan *Internet* yang terintegrasi dengan *server* sebagai pemantau lalu lintas jaringan.

Dengan kata lain, layanan *Internet* yang ada di Pertokoan Ciruas Center masing-masing toko berlangganan *Internet* secara individu ataupun membeli paket *mobile broadband,* informasi ini didapatkan dari hasil survei penggunaan *ISP* yang tersaji pada pada tabel 1.

Tabel 1 Hasil Survei Minat Calon *Client*

No.	Nama	Nama Toko	Jabatan	Akses *Internet*	Penggunaan *Internet* (/Bulan)	Biaya *Internet* (/Bulan)	Minat
1	Mansyur M	Pasar Ciruas	Karyawan	*Wi-Fi*	< 20 GB	< Rp. 200.000	Ya
2	Bahar	Caem	Karyawan	Data Seluler	> 50 GB	< Rp. 100.000	Mungkin
3	Tjung Aleng	Toko Perak Ming-Ping	Pemilik	Data Seluler	< 20 GB	< Rp. 100.000	Ya
4	Sanah	Toko Perak Ming-Ping	Karyawan	Data Seluler	< 5 GB	< Rp. 50.000	Ya
5	Ikhwan M	Caem Two Plastik	Karyawan	Data Seluler	< 50 GB	< Rp. 150.000	Ya
6	Wahyu	Caem Two Plastik	Karyawan	Data Seluler	> 50 GB	< Rp. 150.000	Ya
7	Sarmah	Kejora Accesories	Karyawan	Data Seluler	< 20 GB	< Rp. 100.000	Ya
8	Andi	Redo Putra	Pemilik	Data Seluler	< 50 GB	< Rp. 150.000	Tidak
9	Wahid	Redo Putra	Karyawan	Data Seluler	< 20 GB	< Rp. 150.000	Ya
10	Ronal	Redo Putra	Karyawan	Data Seluler	< 20 GB	< Rp. 150.000	Ya
11	Ugin	Ely Kosmetik	Pemilik	Data Seluler	< 5 GB	< Rp. 200.000	Ya
12	Iwan	Assesoris	Pemilik	Data Seluler	< 5 GB	< Rp. 50.000	Mungkin
13	Siti	Peralatan Sekolah	Karyawan	Data Seluler	< 5 GB	< Rp. 50.000	Ya
14	Hamifah	Purnama Gordyn	Karyawan	Data Seluler	< 5 GB	< Rp. 50.000	Ya
15	Idoh	RM. Bugih Lamo	Karyawan	Data Seluler	< 10 GB	< Rp. 50.000	Ya
16	Kustanti	Anur Parsel	Pemilik	Data Seluler	< 20 GB	< Rp. 50.000	Ya
17	Saefullah	Arya Plastik	Karyawan	Data Seluler	< 20 GB	< Rp. 100.000	Ya
18	Yanti Susanti	Arya Plastik	Karyawan	Data Seluler	< 5 GB	< Rp. 50.000	Ya
19	Nurmala	Warkop	Pemilik	Data Seluler	< 20 GB	< Rp. 100.000	Ya
20	Imron	Lahan Parkir	Pemilik	Data Seluler	< 5 GB	< Rp. 50.000	Ya
21	Ipin	Caem Two Plastik	Karyawan	Data Seluler	< 50 GB	< Rp. 100.000	Ya
22	Yanto	Caem Two Plastik	Karyawan	Data Seluler	< 50 GB	< Rp. 100.000	Ya
23	M. Soleh	Caem Two Plastik	Pemilik	Data Seluler	< 20 GB	< Rp. 200.000	Ya

Berlanggganan *Internet* secara individu memiliki keuntungan dalam menggunakannya tidak ada yang membatasi atau bebas, namun memiliki kerugian berupa biaya langganannya bisa dibilang cukup mahal. Jika ini terus dibiarkan, maka dapat merugikan banyak orang. Selain itu terdapat beberapa *user* yang melakukan pelanggaran dalam mengakses *Internet*, khususnya dalam menonton film dari sumber yang ilegal. Informasi mengenai pelanggaran oleh *user* dalam mengakses *Internet* tersaji pada tabel 2.

Tabel 2 Hasil Survei Akses Situs Ilegal

No.	Nama	Media Menonton Film
1.	Bahar	Situs Ilegal (Website gudangfilm)
2.	Sarmah	Bioskop
3.	Wahid	Situs Ilegal (Website nagaxxi)
4.	Imron	Situs Ilegal (Website lk21)
5.	Hamifah	Situs Ilegal (Website drakorindo)

Untuk mengatasinya maka layanan *Internet* tersebut dapat diatur oleh suatu sistem yang memiliki *server* dengan berlangganan pada satu *ISP* yang berfungsi sebagai pengatur jaringan *Internet* dan memblokir akses ke situs-situs ilegal. Masalah yang timbul adalah pembagian bandwith yang tidak stabil ketika semua menggunakan akses internet, sebagai contoh ketika salah satu client melakukan download terutama menggunakan *download accelerator/ download manager* yang sangat mungkin bisa menghabiskan bandwith, karena bandwith akan tersedot pada client tersebut, sehingga client yang lain akan merasakan koneksi yang lambat (Ontoseno et al., 2017). Jika sistem ini diterapkan maka perlu adanya pengaturan pembatasan bandwith pada masing-masing user sehingga selain penggunaan *Internet* menjadi lebih sehat dan juga biaya tagihan bulanan bisa lebih hemat. Biaya tagihan bisa lebih hemat karena biaya tagihan yang tadinya dibayarkan ke *ISP* secara individu menjadi dibayar secara gotong royong, karena satu *ISP* digunakan untuk banyak *user*. Semakin banyak pengguna yang berlangganan maka biaya bisa semakin murah. Selain itu, juga dapat mengurangi kebiasaan untuk mengakses situs-situs illegal.

Metode Penelitian

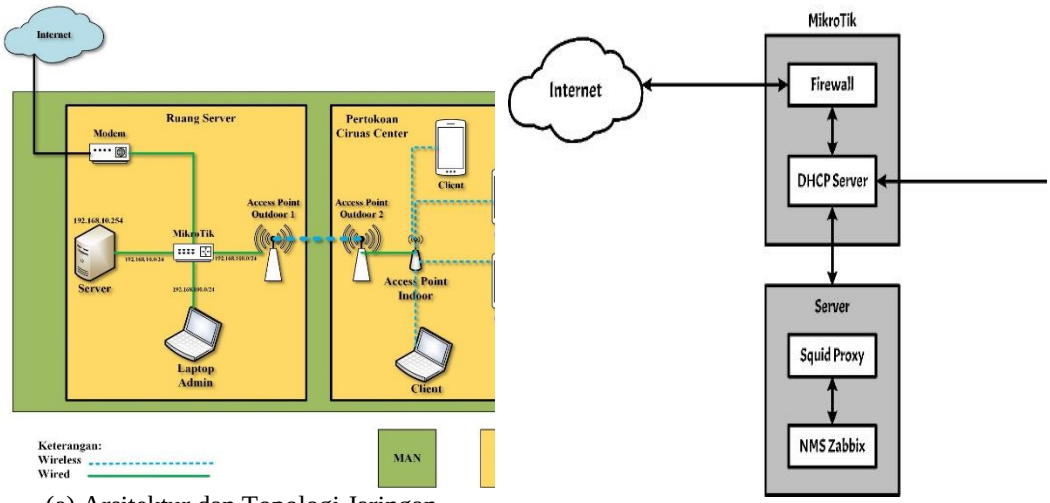
Tujuan dilakukannya penelitian ini adalah untuk memberikan solusi dimasyarakat agar memiliki akses *Internet* yang lebih terjangkau dan juga lebih sehat. Pada umumnya masyarakat untuk mendapatkan akses *Internet* harus membeli paket *mobile broadband* atau berlangganan pada salah satu *ISP* yang cukup memakan biaya. Dengan hasil penelitian ini masyarakat cukup berlangganan dengan salah satu *ISP* sudah bisa digunakan untuk bersama, dengan begitu untuk biaya berlangganan bisa dibayar secara gotong royong dan menjadi lebih murah.

Dalam penelitian ini selain menghemat biaya berlangganan ke *ISP*, penelitian ini juga bertujuan menyediakan layanan *Internet* yang lebih sehat. Sehat dalam artian akses *Internet* yang

diberikan diatur menggunakan sistem yang dapat memantau dan membatasi akses supaya tidak terjadi pelanggaran hukum. Pelanggaran yang sering dilakukan oleh pengguna *Internet* adalah mengakses konten dewasa dan situs pembajakan film.

Hasil rancangan arsitektur dan topologi jaringannya adalah seperti tersaji pada gambar 1(a). Berdasarkan gambar 1(a) dapat diketahui bahwa pada ruang *server* dan pertokoan Ciruas Center akan menggunakan arsitektur *LAN*. Kedua tempat tersebut dihubungkan secara *wireless point-to-point* menggunakan *access point outdoor* yang berjarak 2 km. Sehingga dapat diasumsikan bahwa kedua jaringan *LAN* tersebut saling terhubung menggunakan arsitektur *MAN*

Dalam topologi tersebut dapat diketahui bahwa semua berpusat pada *MikroTik*. Selain *modem*, semua perangkat *client* dihubungkan dengan satu segmen *IP address* yaitu 192.168.5.0/24. *DHCP Server* berada di *MikroTik*, supaya perangkat yang ingin terhubung tidak perlu untuk mengatur *IP address* secara manual.



(a) Arsitektur dan Topologi Jaringan (b) Skema Jaringan

Gambar 1 Arsitektur, Topologi, dan Skema Jaringan

Sesuai dengan topologi jaringan yang telah dijelaskan sebelumnya, skema jaringan yang diterapkan tersaji pada gambar 1(b).

Dalam skema jaringan tersebut perangkat *client* mendapatkan *IP* dari *DHCP Server* yang merupakan salah satu pelayanan didalam *MikroTIk*. *Server* tersebut memiliki 2 layanan, yaitu *NMS Zabbix* dan *Squid Proxy*. *NMS Zabbix* digunakan untuk memantau lalu lintas pada jaringan pada *MikroTik*. *Squid Proxy* digunakan untuk membatasi akses ke situs-situs ilegal, untuk menghindari pelanggaran hukum. *Firewall* yang ada pada *MikroTik* berguna untuk mengalihkan *port* 80 *http* ke *port 3128* dan *port* 443 *https* ke *port 3129* yang merupakan *port Squid Proxy*. Pengalihan *port* ini berguna supaya *website request* oleh *client* diarahkan ke *proxy* terlebih dahulu, karena didalam *Squid Proxy* tersebut terdapat filter untuk membatasi akses ke *website* yang tidak dilarang. DHCP Server

digunakan untuk memberikan alamat *IP* secara otomatis kepada *client* yang terhubung ke jaringan tersebut, supaya tidak perlu repot untuk mengatur alamat *IP* secara manual pada setiap *client* yang ingin terhubung ke jaringan tersebut.

Sebelum sistem ini digunakan, ada beberapa konfigurasi yang harus dilakukan. Seperti konfigurasi *Firewall, DHCP Server, Squid Proxy,* dan *NMS Zabbix.* Namun, sebaiknya *IP address Server* diatur menjadi *static IP* supaya alamat *IP* pada *Server* tidak berubah-ubah. Pada sistem operasi *Linux CentOS 7* cara mengaturnya adalah sebagai berikut:

1. Langkah pertama adalah ketik perintah *"vim /etc/sysconfig/network-script/ifcfg-enp3s0"*. Pada bagian *enp3s0* disesuaikan dengan nama *interface ethernet* yang terhubung ke *MikroTik.* Setelah itu tekan *"Enter"*. Tersaji pada gambar 2.

```
[root@localhost ~]# vim /etc/sysconfig/network-scripts/ifcfg-enp3s0
```

Gambar 2. Perintah Untuk Mengatur *IP Address*

2. Langkah kedua, setelah itu yang harus diperhatikan adalah pada baris *"BOOTPROTO"* dan *"IPV6INIT"* keduanya harus bertuliskan *"no"*. Selain itu perhatikan juga baris *"DEFROUTE"* berbeda dengan sebelumnya, *DEFROUTE* harus bertuliskan *"yes"*. Selanjutnya tambahkan "*IPADDR"* sebagai alamat *IP* yang diinginkan, *"GATEWAY"* sebagai *gateway* atau dalam penelitian ini *gateway* adalah alamat *IP* pada *MikroTik*, *"DNS1"* dan *"DNS2"* sebagai *dns* yang akan digunakan dalam topologi. Untuk mengetikkan konfigurasi tersebut tekan huruf I pada *keyboard* untuk *"Insert"*. Hasil konfigurasinya bisa dilihat pada gambar 4.17. Setelah selesai simpan lalu keluar dari konfigurasi tersebut, dengan cara tekan tombol *"esc"* pada *keyboard* lalu ketikkan *":wq"*. Tersaji pada gambar 3.

```
TYPE="Ethernet"
PROXY_METHOD="none"
BROWSER_ONLY="no"
BOOTPROTO="no"
IPADDR=192.168.10.254
GATEWAY=192.168.10.1
DNS1=192.168.10.1
DNS2=192.168.1.1
DEFROUTE="yes"
IPV4_FAILURE_FATAL="no"
IPV6INIT="no"
IPV6_AUTOCONF="yes"
IPV6_DEFROUTE="yes"
IPV6_FAILURE_FATAL="no"
IPV6_ADDR_GEN_MODE="stable-privacy"
NAME="enp3s0"
UUID="9bb91553-24b1-4f31-99e0-9de5e687ba84"
DEVICE="enp3s0"
ONBOOT="yes"
```

Gambar 3. Konfigurasi *IP Static* Pada *Server*

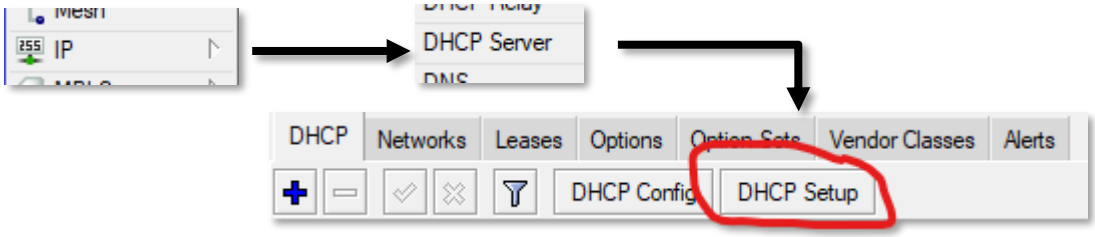
3. Langkah terakhir adalah dengan menyalakan ulang sistem jaringan pada *server*. Caranya adalah dengan mengetikkan perintah *"systemctl restart network"* lalu tekan *"Enter"* pada *keyboard*. Tersaji pada gambar 4.

```
[root@localhost ~]# systemctl restart network
```

Gambar 4. Perintah Menyalakan Ulang Sistem Jaringan Pada Server

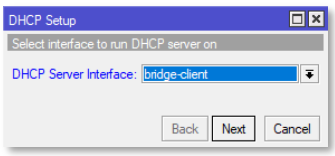
Hasil dan Pembahasan

Konfigurasi *DHCP Server* diperlukan untuk kemudahan *client* yang ingin terhubung ke jaringan. *DHCP Server* ini sangat berguna untuk jaringan yang memiliki jumlah *user*-nya tidak menentu atau berubah-ubah. Konfigurasi *DHCP Server* pada *MikroTik* dapat dilakukan pada menu *IP* > *DHCP Server* > *DHCP Setup* atau seperti tersaji pada gambar 5.

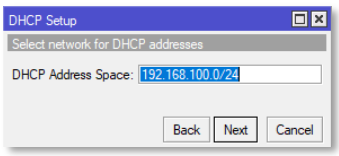


Gambar 5 Tahap Awal Konfigurasi *DHCP Server*

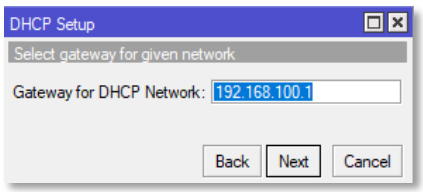
Dengan menekan tombol *"DHCP Setup"*, sistem pada *MikroTik* akan menuntun untuk melakukan konfigurasi dengan menampilkan kotak-kotak dialog pada setiap tahapnya. Tersaji pada gambar 6(a).



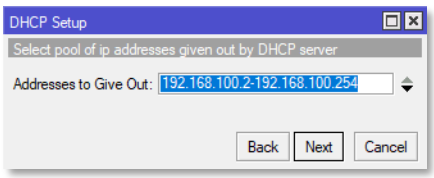
(a) Memilih *Interface*



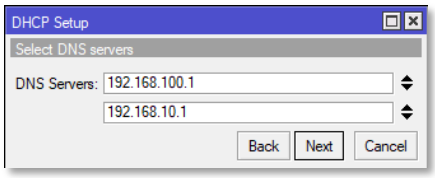
(b) Menentukan Segmen *IP Address*



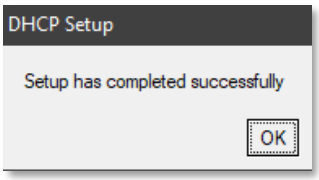
(c) Menentukan *Default Gateway* Untuk *Client*



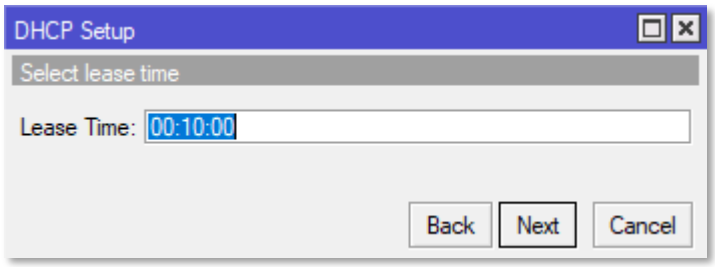
(d) Menentukan Alamat *IP* Untuk *Client*



(e) Mengatur *DNS* Pada *DHCP Server*



(f) Konfigurasi *DHCP Server* Selesai



(g) Menentukan *Lease Time*

Gambar 6 Langkah-langkah konfigurasi DHCP Server

Langkah pertama, menentukan pada *interface* mana *DHCP Server* akan aktif. Pada penelitian ini *"bridge-client"* dipilih sebagai *interface* yang akan tersedia *DHCP Server*. Lalu klik *"Next"*. Tersaji pada gambar 6(b).

Dalam penelitian ini segmen *IP address* yang digunakan adalah 192.168.100.0/24. Jika *interface* sebelumnya sudah terdapat alamat *IP*, maka *DHCP Address Space* akan otomatis mengambil segmen *IP* yang sama. Tersaji pada gambar 6(c).

Selanjutnya, menentukan alamat *IP* yang akan digunakan sebagai *default-gateway* oleh *DHCP Client* nantinya. Tersaji pada gambar 6(d).

Tentukan alamat *IP* yang akan didistribusikan ke *client*. Secara otomatis sistem akan mengisi *host IP* pada segmen yang telah digunakan. Pada penelitian ini, *IP* 192.168.100.1 tidak masuk dalam *"Addresses ti Give Out"*, sebab *IP* tersebut sudah digunakan sebagai *gateway* dan tidak akan didistribusikan ke *client*. Tersaji pada gambar 6(e).

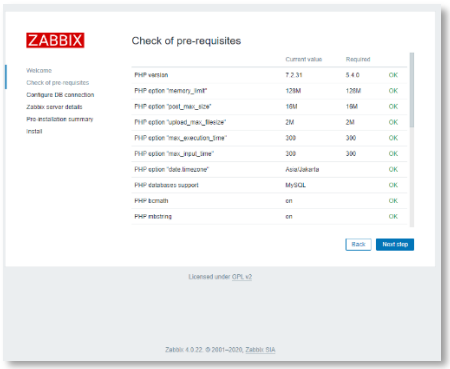
DNS perlu ditentukan supaya nantinya *client* melakukan *request DNS* ke *DNS* yang sudah ditentukan. Tersaji pada gambar 6(g). Langkah terakhir adalah menentukan *Lease-Time*, yaitu berapa lama waktu sebuah alamat *IP* akan dipinjamkan ke *client*. Untuk menghindari penuh atau kehabisan *IP*, atur *Lease-Time* jangan terlalu lama. Sampai langkah ini, jika diklik *"Next"* akan tertampil pesan yang menyatakan bahwa konfigurasi *DHCP Server* telah selesai. Tersaji pada gambar 6(f).

Konfigurasi *NMS Zabbix*

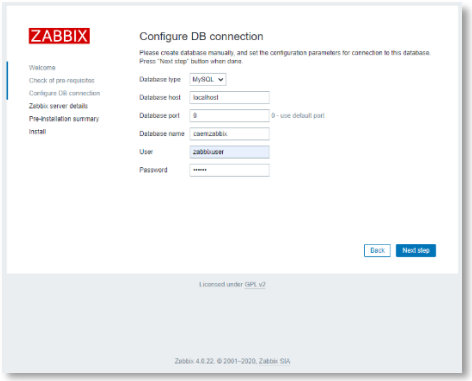
Setelah *NMS Zabbix* sudah terinstal dengan benar, maka *NMS Zabbix* bisa diakses pada *browser* dengan mengetikkan "*IPServer/zabbix"*. Jika sesuai akan muncul tampilan seperti pada gambar 7(a). Lalu klik "*Next Step"* dan akan muncul tampilan seperti pada gambar 7(b). Pada bagian tersebut pastikan semua parameter bertuliskan "*OK*" yang berarti *PHP* dan *Database* sudah terinstal dengan benar. Jika sudah, selanjutnya klik *"Next Step"*. Gambar 7(c) merupakan tahap untuk menghubungkan *database* yang sudah dibuat dalam sistem operasi dengan aplikasi *zabbix* tersebut. Pastikan nama *database, user database,* dan *password* sudah sesuai. Setelah itu klik *"Next Step"*. Tersaji pada gambar 7(d). Pada tahap ini diminta untuk memberikan sebuah nama untuk *Server*

zabbix. Lalu klik *"Next Step"*. Tersaji pada gambar 7(e). Selanjutnya, sistem akan memperlihatkan apa saja yang sebelumnya sudah dikonfigurasi. Untuk memastikan apakah ada kesalahan dalam penulisan yang tidak sesuai dengan *database* yang ada. Jika semua sudah sesuai, lalu klik *"Next Step"*. Konfigurasi dan instalasi *zabbix* telah selesai tersaji pada gambar 7(f). Selanjutnya klik *"Finish"*. Lalu sistem akan menampilkan tampilan *login* untuk mengakses *zabbix*. Untuk *username default*-nya adalah "Admin" sedangkan untuk *password default*-nya adalah "zabbix". Setelah berhasil login maka akan muncul tampilan awal *zabbix*. Tersaji pada gambar 7(g).

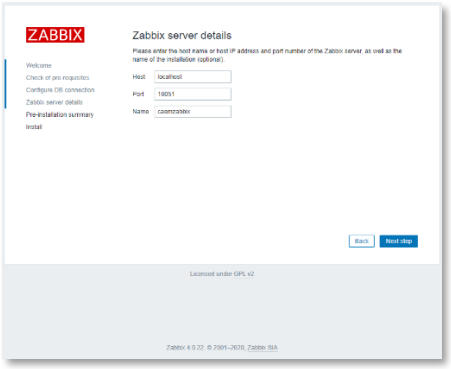
(a) Tampilan Utama *NMS Zabbix*



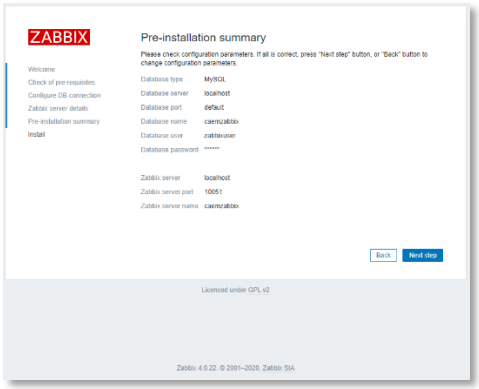
(b) Tampilan Cek Parameter *PHP* dan *Database* yang Terinstal

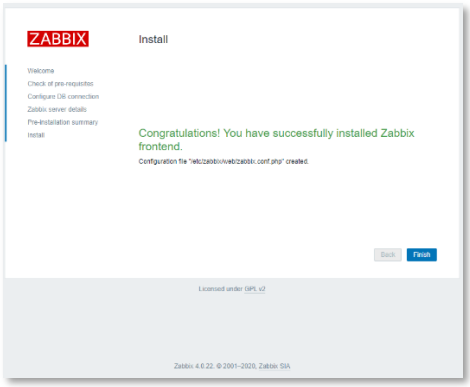


(c) Mengatur *Database Untuk Zabbix*

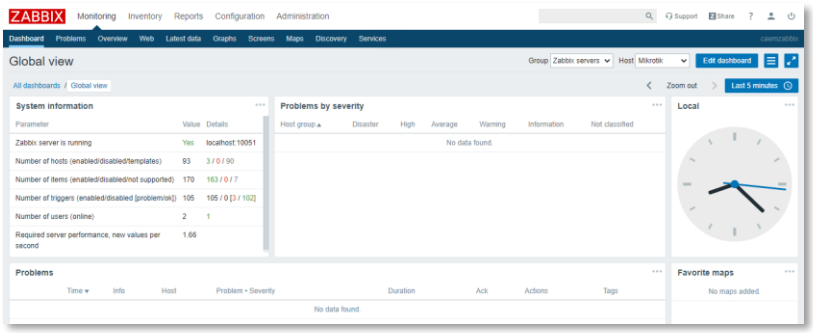


(d) Memberi Nama *Server Zabbix*





(e) Konfirmasi Sebelum Instalasi Selesai (f) Konfigurasi dan Instalasi Zabbix Selesai

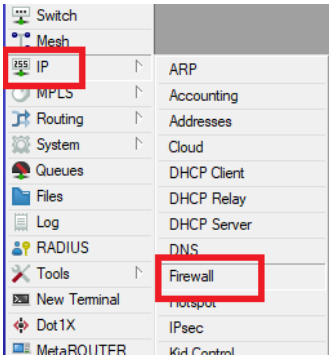


(h) Tampilan Awal *Zabbix*

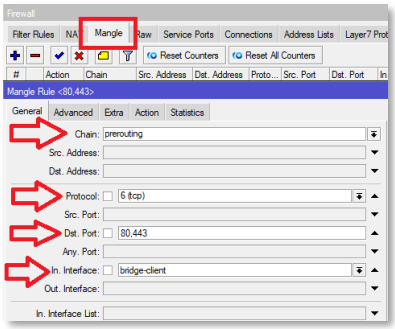
Gambar 7. Langkah-langkah konfigurasi *Zabbix*

Konfigurasi *Firewall* Pada *MikroTik*

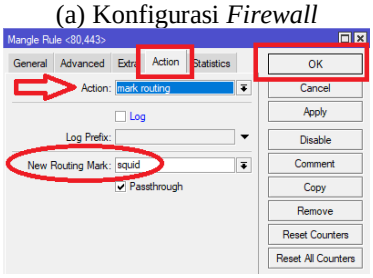
Saat *client* mengakses situs berarti *client* melakukan *request http/https* yang pada dasarnya masing-masing memiliki *port* 80 dan *port* 443. Normalnya *resquest http/https* dari *client* langsung dilayani oleh *ISP*, namun pada penelitian ini hal tersebut dirubah. Sebelumnya *request http/https* langsung dilayani oleh *ISP* kini dilayani oleh *Squid Proxy* melalui *port* 3128 untuk *http* dan *port* 3129 untuk *https.* Untuk melakukan itu semua diperlukan konfigurasi pada *firewall* dalam *router* dan *Squid Proxy* itupun sendiri. Setelah berhasil mengakses *MikroTik* pilih menu *"IP > Firewall"* tersaji pada gambar 8(a).



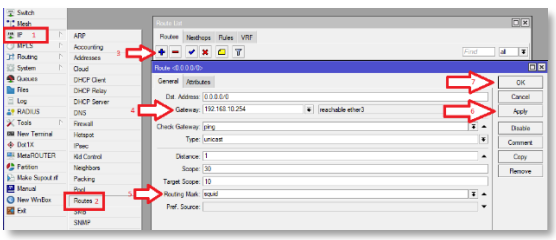
(a) Konfigurasi *Firewall*



(b) Aturan Baru Pada *Mangle*



(c) *Action* Pada Aturan *Mangle*



(d) Konfigurasi *Routing* Untuk *Squid*

Gambar 8. Konfigurasi *Firewall* pada *MikroTik*

Pada tab *"Mangle"* tambahkan aturan baru. Pada kolom *"Chain"* pilihlah *"prerouting"*, untuk *"protocol"* pilih *"tcp"*, untuk *"Dst. Port"* ketikkan "80,443", dan untuk *"In. Interface"* pilih *interface* yang mengarah ke *client.* Adapun *interface* yang mengarah ke *client* pada penelitian adalah *interface "brige-client"* tersaji pada gambar 8(b). Selanjutnya pilih tab *"Action"*, lalu untuk kolom *"Action"* pilih *"mark routing"*, dan jangan lupa untuk memberi identitas untuk aturan yang baru tersebut pada kolom *"New Routing Mark"* tersaji pada gambar 8(c). Setelah aturan *Mangle* telah dibuat selanjutnya adalah mengatur *routing*. Pilih tab *"IP > Routes"*. Lalu buat *routing* baru, isikan *"Gateway"* dengan alamat *IP Squid Proxy*, pada *"Routing Mark"* isikan nama aturan yang sebelumnya ditambahkan pada *Mangle*, lalu klik *"Apply"* dan terakhir klik *"OK"* tersaji pada gambar 8(d)..

Konfigurasi *Squid Proxy*

Ada dua *port* yang akan dialihkan oleh *router*, yaitu *port 80* dan *433* yang masing-masing adalah *port* yang digunakan oleh *http* dan *https*. *Server squid* harus mampu menangani pengalihan *port*. Adapun pengalihan yang akan dilakukan oleh *proxy server* adalah pengalihan dari *port* 80 ke *port 3128* yakni sebagai *port http* dan pengalihan dari *port 443* ke *port 3129* yakni sebagai *port https*. Langkah pertama yang dilakukan adalah mengaktifkan *nat masquerade* pada *firewall server* dengan mengetikkan perintah yang tersaji pada gambar 9. Langkah selanjutnya adalah membuka dua *port* untuk *squid*, yaitu *port* 3128 dan 3129 dengan mengetikkan perintah yang tersaji pada gambar 10. Selanjutnya adalah meneruskan *port* 80 dan 443 ke *port* 3128 dan 3129, dengan mengetikkan perintah yang tersaji pada gambar 11.

```
[root@192 ~]# firewall-cmd --zone=public --add-masquerade --permanent
```

Gambar 9. Perintah Mengaktifkan *NAT Masquerade*

```
[root@192 ~]# firewall-cmd --permanent --zone=public --add-port=3128/tcp
[root@192 ~]# firewall-cmd --permanent --zone=public --add-port=3129/tcp
```

Gambar 10 Perintah Membuka *Port* Pada *Server*

```
firewall-cmd --permanent --zone=public --add-forward-port=port=80:proto=tcp:toport=3128:toaddr=192.168.10.254
firewall-cmd --permanent --zone=public --add-forward-port=port=443:proto=tcp:toport=3129:toaddr=192.168.10.254
```

Gambar 11 Perintah Meneruskan *Port* Pada *Server*

Pada gambar 12. Terdapat alamat *IP*, alamat *IP* tersebut disesuaikan dengan alamat *IP* pada *proxy server*. Jika sudah meneruskan *port* selanjutnya adalah memuat ulang *firewall* pada *server*. Perintah untuk melakukannya tersaji pada gambar 12. Selanjutnya adalah memeriksa apakah konfigurasi pada *firewall server* udah benar atau belum, dengan mengetikkan perintah yang tersaji pada gambar 13.

```
[root@192 ~]# firewall-cmd --reload
```

Gambar 12. Perintah Memuat Ulang *Firewall* Pada *Server*

```
[root@192 ~]# firewall-cmd --list-all
public (active)
  target: default
  icmp-block-inversion: no
  interfaces: enp3s0
  sources:
  services: dhcpv6-client http https ssh
  ports: 10051/tcp 10050/tcp 3128/tcp 3129/tcp
  protocols:
  masquerade: yes
  forward-ports: port=80:proto=tcp:toport=3128:toaddr=192.168.10.254
        port=443:proto=tcp:toport=3129:toaddr=192.168.10.254
  source-ports:
  icmp-blocks:
  rich rules:
```

Gambar 13. Perintah Memeriksa Konfigurasi *Firewall Server*

Untuk memblokir situs yang tidak diinginkan, perlu dilakukan konfigurasi pada *Squid Proxy*. Untuk melakukan konfigurasi ketikkan perintah *"vim /etc/squid/squid.conf"* hapus semua isinya dan tuliskan konfigurasi yang tersaji pada gambar 14. Sedangkan untuk mendaftarkan situs yang ingin diblokir bisa dituliskan dalam file *"blokweb.txt"* yang ada didalam direktori *"etc/squid"* tersaji pada gambar 15.

```
[root@192 ~]# vim /etc/squid/squid.conf
acl our_network src 192.168.100.0/24
acl blok dstdomain -n "/etc/squid/blokweb.txt"
acl CONNECT method CONNECT

http_access deny blok
http_access allow all

http_port 3130
http_port 3128 intercept
https_port 3129 intercept ssl-bump generate-host-certificates=on dynamic_cert_mem_cache_size=4MB cert=/etc/squid/ssl_cert/squid35.pem

sslcrtd_program /usr/lib64/squid/ssl_crtd -s /var/lib/ssl_db -M 8MB
sslcrtd_children 16 startup=5 idle=1

ssl_bump peek all
ssl_bump splice all
```

Gambar 14. Isi Konfigurasi *Squid Proxy*

```
[root@192 ~]# vim /etc/squid/blokweb.txt

onnocenter.or.id
```

Gambar 15. Daftar Situs yang Diblokir *Squid*

Berikan akses kepada *squid* supaya *squid* dapat membaca atau mengeksekusi file *blokweb.txt* tersebut. Cara untuk memberikan akses kepada *squid* dengan mengetikkan perintah yang tersaji pada gambar 16. Langkah terakhir adalah menyalakan ulang sistem *Squid Proxy*, perintah untuk menyalakan ulang sistem *Squid Proxy* adalah *"systemctl restart squid"* tersaji pada gambar 17.

```
[root@192 ~]# chown -R squid:squid /etc/squid/blokweb.txt
[root@192 ~]# chmod 777 /etc/squid/blokweb.txt
```

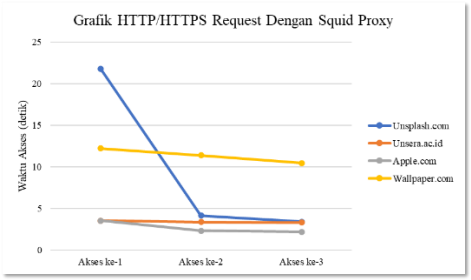
Gambar 16. Memberikan Akses Kepada *Squid*

```
[root@192 ~]# systemctl restart squid
```

Gambar 17. Perintah Menyalakan Ulang Sistem *Squid Proxy*

Pada penelitian ini menghasilkan sistem penyedia akses *Internet* yang didalamnya terdapat *software monitoring* jaringan dan *proxy server*. Sebelumnya masing-masing pengguna *Internet* harus berlangganan ke salah satu *IP* untuk pemakaian pribadi. Saat ini, biaya berlangganan *Internet* dapat dibayarkan dengan bergotong royong. Akses *Internet* yang disediakan oleh sistem tersebut dirancang lebih aman dengan menggunakan *Squid Proxy*. Selain itu, lalu lintas jaringan pada sistem tersebut juga terpantau secara *real-time*.

Squid proxy selain berperan untuk memblokir situs, *Squid Proxy* juga menggantikan peran *client* yang melakukan *http/https request* ke *ISP*. Situs yang melewati *Squid Proxy* akan tersimpan di dalam *cache Squid Proxy*. Ini berguna untuk mempercepat proses memuat situs yang sama untuk *request* selanjutnya. Berikut perbandingan kecepatan memuat situs yang sama tanpa *Squid Proxy* dengan memuat situs yang sama menggunakan *Squid Proxy* tersaji pada gambar 18.



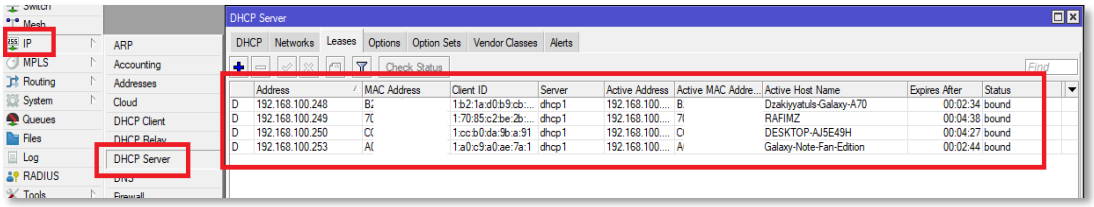
Gambar 18. Grafik *HTTP/HTTPS Request* Dengan *Squid Proxy*

Selama penelitian ini berlangsung, sudah terbangun jaringan *Internet* yang sehat dan aman. Didalam jaringan tersebut terdapat satu *server*, satu *router MikroTik*, dua *access point outdoor*, dan satu *access point indoor*. Pada *server* sudah terpasang aplikasi *monitoring* jaringan yaitu *Zabbix* dan juga aplikasi *Squid Proxy*. *Zabbix* bertugas untuk merekam lalu lintas jaringan, mulai dari *bandwidth usage*, kecepatan unggah, kecepatan unduh, hingga memantau kinerja perangkat jaringan dan juga *server* itu sendiri.

Namun, karena produk *access point* yang digunakan dalam penelitian ini merupakan *brand* yang tidak terlalu banyak orang tau. Maka *Zabbix* belum memiliki *template* untuk menampilkan grafik dari *access point* yang digunakan dalam penelitian ini. Walaupun *access point outdoor* yang digunakan dalam penelitian ini sudah mendukung protokol *SNMP (Simple Network Management Protocol)* sekalipun. Protokol *SNMP* merupakan protokol yang digunakan untuk me-*monitor* dan mengelola berbagai perangkat di jaringan *Internet* secara jarak jauh. Sebagai gantinya, jika ingin me-*monitor access point* yang digunakan dalam penelitian ini, *monitoring* tersebut dapat dilakukan melalui *router MikroTik*.

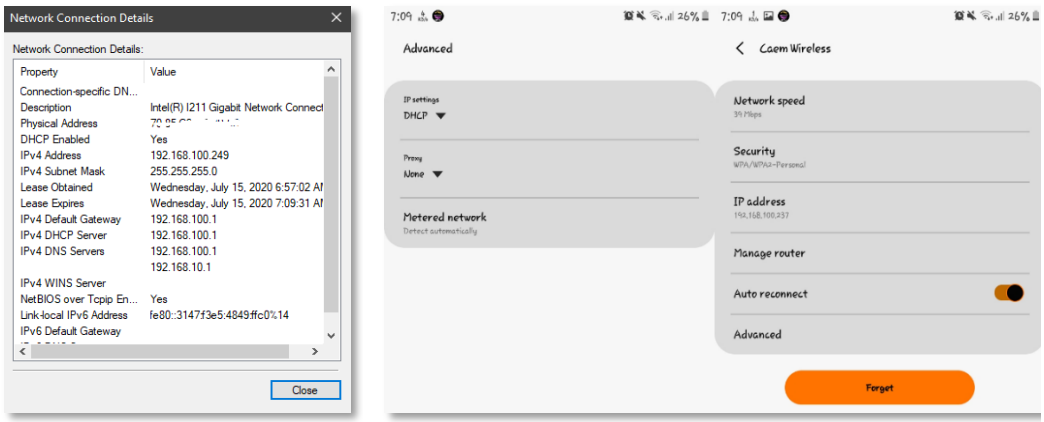
Router MikroTik dalam penelitian ini berperan untuk mengalihkan masing-masing *port* 80 sebagai *http request* ke *port 3128* dan *port 443* sebagai *https request* ke *port 3129*. Sebagaimana sudah dijelaskan pada subbab 4.2.2.3 tentang konfigurasi *firewall* pada *MikroTik*. Selain itu, *router MikroTik* juga berperan sebagai pembatas akses ke situs-situs yang dilarang.

Hasil pengujian *DHCP Server* jika ada *client* yang terhubung ke jaringan lokal, maka tersaji pada pada gambar 19.



Gambar 19 Pengujian *DHCP Server*

Dari sisi *client* jika sudah terhubung, maka *IP address* yang akan tampil seperti pada gambar 20.

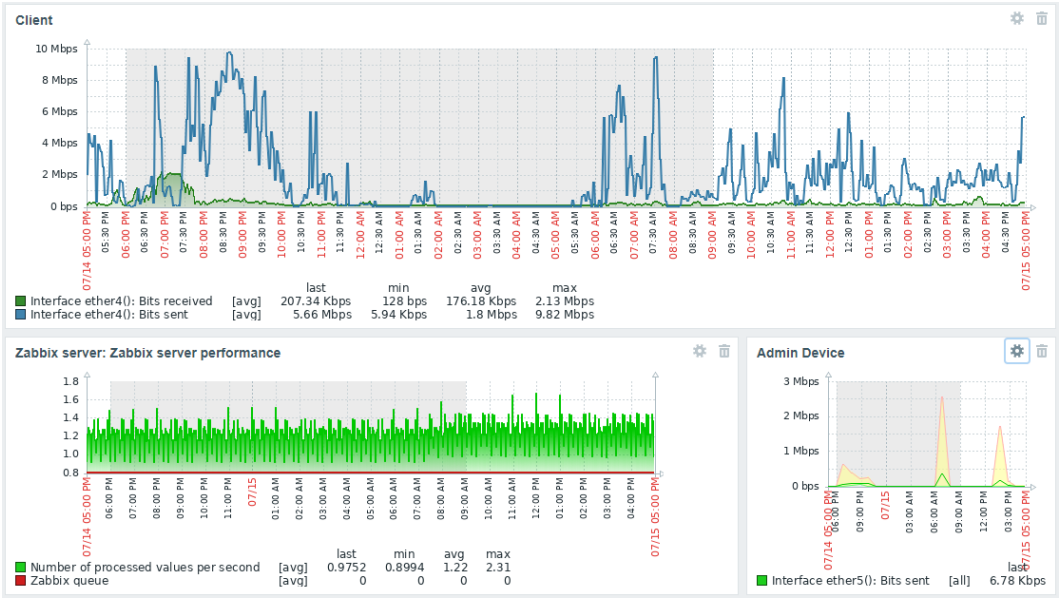


(a) Tampilan *IP* Pada *Client* Dengan *Windows OS*

(b) Tampilan *IP* Pada *Client* Dengan *Android OS*

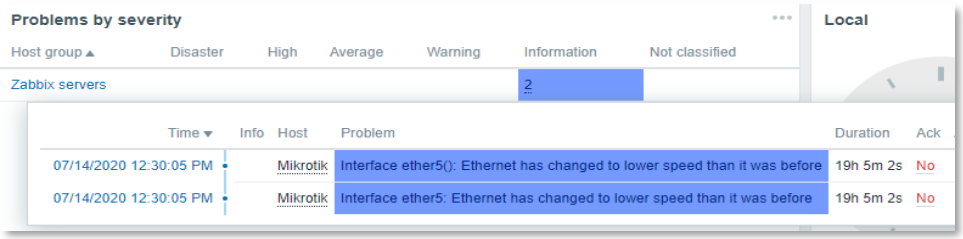
Gambar 20. Tampilan *IP* pada *Client*

Pada gambar 21 tersaji hasil pengujian *NMS Zabbix* yang telah didapat setelah merekam lalu lintas jaringan selama 24 jam.



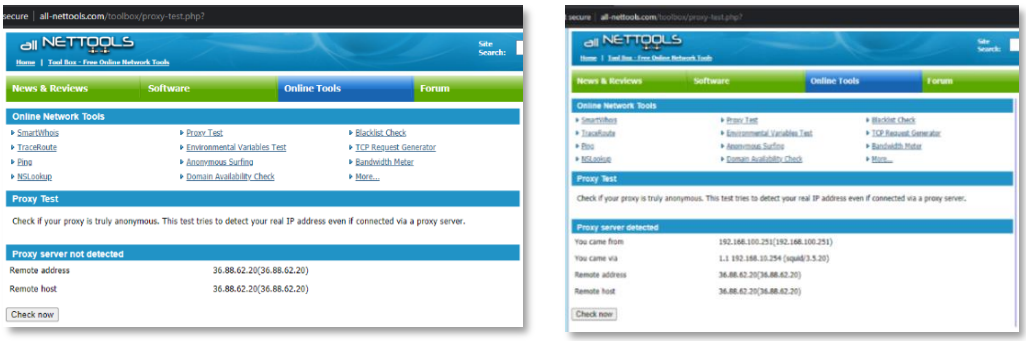
Gambar 21. Tampilan Lalu Lintas Jaringan Dalam 24 Jam

Pada gambar 21 adalah hasil pengujian *NMS Zabbix* untuk merekam lalu lintas jaringan selama 24 jam. Terhitung mulai dari tanggal 14 Juli 2020 pada jam 5 sore hingga tanggal 15 Juli 2020 pada jam 5 sore. Grafik tersebut menggambarkan bahwa *Server* dan *router MikroTik* bekerja sama untuk melayani *client.* Namun, perangkat yang digunakan oleh *Admin* hanya beberapa kali aktif untuk melakukan pengecekan apakah terjadi masalah atau tidak. Selain itu, adapun pesan yang akan ditampilkan oleh *NMS Zabbix* jika ada yang tidak normal pada jaringan tersebut. Misalnya saat salah satu *Interface* diturunkan *bandwidth*-nya, maka akan tampil seperti pada gambar 22.



Gambar 22. Tampilan *Problem* yang Terdeteksi Oleh *NMS Zabbix*

Fungsi utama *firewall* dalam penelitian ini adalah mengalihkan *port http/https request* ke *port Squid Proxy*. Hasil pengujian firewall disajikan pada gambar 23 dengan melakukan pengujian melalui situs *"allnettools.com"*.



(a) Tanpa *Proxy* (b) Dengan *Proxy*

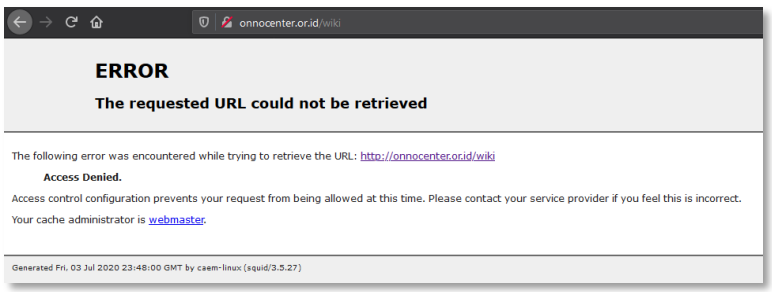
Gambar 23. Pengujian *Proxy* Melalui Situs

Cara kedua untuk menguji *firewall* telah bekerja adalah melalui *server*. Hasil pengujian kinerja firewall tersaji pada gambar 24. Dimana *firewall* berhasil melakukan pengalihan *port*.

```
1594443075.645    157 192.168.100.249 TCP_MISS/200 36243 GET http://192.168.10.254/zabbix/chart2.php? - ORIGINAL_DST/192.168.10.254 image/png
1594443077.461     71 192.168.100.249 TCP_MISS/200 1061 POST http://192.168.10.254/zabbix/zabbix.php? - ORIGINAL_DST/192.168.10.254 application/json
1594443078.477     87 192.168.100.249 TCP_MISS/200 678 GET http://192.168.10.254/zabbix/chart.php? - ORIGINAL_DST/192.168.10.254 image/png
1594443078.619    138 192.168.100.249 TCP_MISS/200 26012 GET http://192.168.10.254/zabbix/chart.php? - ORIGINAL_DST/192.168.10.254 image/png
1594443083.390  19993 192.168.100.253 TCP_MISS/502 3944 POST http://c.whatsapp.net/chat - ORIGINAL_DST/69.171.250.61 text/html
1594443083.486     98 192.168.100.249 TCP_MISS/200 679 GET http://192.168.10.254/zabbix/chart2.php? - ORIGINAL_DST/192.168.10.254 image/png
1594443083.650    160 192.168.100.249 TCP_MISS/200 34696 GET http://192.168.10.254/zabbix/chart2.php? - ORIGINAL_DST/192.168.10.254 image/png
1594443084.671     23 192.168.100.253 TAG_NONE/200 0 CONNECT 157.240.19.54:443 - HIER_NONE/- -
1594443085.492     94 192.168.100.249 TCP_MISS/200 678 GET http://192.168.10.254/zabbix/chart2.php? - ORIGINAL_DST/192.168.10.254 image/png
1594443085.514  20192 192.168.100.248 TCP_TUNNEL/200 22295 CONNECT 36.91.232.25:443 - ORIGINAL_DST/36.91.232.25 -
1594443085.655    159 192.168.100.249 TCP_MISS/200 36119 GET http://192.168.10.254/zabbix/chart2.php? - ORIGINAL_DST/192.168.10.254 image/png
1594443086.084  40092 192.168.100.249 TCP_TUNNEL/200 1813 CONNECT 118.98.113.26:443 - ORIGINAL_DST/118.98.113.26 -
1594443087.696     74 192.168.100.249 TCP_MISS/200 393 POST http://91.108.56.140/api - ORIGINAL_DST/91.108.56.140 application/octet-stream
1594443088.500     88 192.168.100.249 TCP_MISS/200 678 GET http://192.168.10.254/zabbix/chart.php? - ORIGINAL_DST/192.168.10.254 image/png
1594443088.525    124 192.168.100.250 TAG_NONE/200 0 CONNECT 52.229.171.86:443 - ORIGINAL_DST/52.229.171.86 -
1594443088.639    134 192.168.100.249 TCP_MISS/200 26183 GET http://192.168.10.254/zabbix/chart.php? - ORIGINAL_DST/192.168.10.254 image/png
1594443089.388  25043 192.168.100.249 TCP_MISS/200 393 POST http://91.108.56.140/api - ORIGINAL_DST/91.108.56.140 application/octet-stream
1594443089.839  21858 192.168.100.248 TCP_TUNNEL/200 2739 CONNECT 118.97.158.217:443 - ORIGINAL_DST/118.97.158.217 -
1594443090.025  25044 192.168.100.249 TCP_MISS/200 393 POST http://91.108.56.140/api - ORIGINAL_DST/91.108.56.140 application/octet-stream
```

Gambar 24. Tampilan *Access.log*

Dalam penelitian ini *Squid Proxy* diberi tugas untuk membatasi akses ke situs-situs tertentu. Hasil pengujian *Squid Proxy* tersaji pada gambar 25 yang menunjukkan hasil pengujian terhadap situs yang diblokir.



Gambar 25. Hasil Pengujian *Squid Proxy*

Kesimpulan

Berdasarkan penelitian yang telah dilakukan, maka ada beberapa hal yang dapat disimpulkan, yaitu sebagai berikut:

1. Dengan adanya Lembaga yang menyediakan layanan *Internet* untuk *customer*-nya disuatu tempat yang tetap, dapat mengurangi biaya berlangganan untuk akses *Internet* ke salah satu *ISP* secara individu pada satu lokasi yang tetap.
2. Dengan adanya peran *NMS Zabbix* untuk memantau jaringan, juga peran *router MikroTik* untuk mengatur jaringan, dan peran *Squid Proxy* untuk menjaga jaringan tersebut dengan membatasi akses ke situs-situs terlarang. Sehingga tercipta jaringan *Internet* yang sehat

Saran

Dengan menyadari bahwa dalam penelitian ini banyak sekali terdapat kekurangan dan kelemahan, maka dari itu berikut ini adalah beberapa hal yang dapat disarankan oleh penulis untuk memperbaiki kekurangan dan kelemahan tersebut.

1. Belum tersedianya *template* untuk menampilkan grafik dari *monitoring access point Tenda* pada *NMS Zabbix*. Maka perlu dibuatkan *template* supaya dapat menampilkan grafik *monitoring access point Tenda* pada *NMS Zabbix.*
2. Perlu adanya peningkatan dalam keamanan jaringan tersebut, karena dalam penelitian ini untuk *user* yang ingin terhubung keamanannya hanya mengandalkan password pada *Wi-Fi.*
3. Untuk pengembangan selanjutnya diharapkan dapat menggunakan *software proxy server* yang lain, karena masih sedikit yang menggunakan *software proxy server* selain *squid*. Contoh *software proxy server* yang lain seperti, *Privoxy, Polipo, Tiny Proxy,*

Referensi

Hidayat, T., Studi, P., Informatika, T., Teknik, J., Politeknik, I., & Bengkalis, N. (n.d.). *Desain prototipe aplikasi sistem monitoring browser ponsel anak untuk menerapkan internet sehat dengan kontrol orang tua*. 43–49.

Ontoseno, R. D. H., Haqqi, M. N., & Hatta, M. (2017). *LIMITASI PENGGUNA AKSES INTERNET BERDASARKAN KUOTA WAKTU DAN DATA MENGGUNAKAN PC ROUTER OS MIKROTIK (Studi Kasus : SMK YPM 7 Tarik)*. *1*, 125–130.

Purnama, R. A. (2019). *Filtering Domain Name Server (DNS) untuk Membangun Internet Sehat Menggunakan Routerboard Mikrotik (Development of Safety Internet with Filtering Domain Names Server (DNS) on Mikrotik Routerbroad)*. *VII*, 43–48.

Wijayanto, D., & Waspada, I. (2016). *Perangkat dan Aktivitas Pengguna pada Jaringan Menggunakan Protocol SNMP dan Squid Proxy*. *02*(03), 11–20.