BAB 4
HASIL PENELITIAN

4.1 HASIL PENELITIAN

Penelitian ini bertujuan untuk mengimplementasikan Pi-Hole untuk mengurangi risiko serangan siber berupa *malvertising*. Sistem ini dikembangkan dengan konfigurasi perangkat Raspberry Pi 3 model B+. Penelitian dilakukan dalam jangka waktu 14 hari, yaitu pada tanggal 26 Juli sampai dengan 8 Agustus 2022. Sistem diimplementasikan pada DMZ (*Demilitarized Zone*) dalam jaringan FTTI Universitas Jenderal Achmad Yani Yogyakarta. Proses ini dilakukan sebagai upaya pengambilan data objek analisis. Pengembangan sistem pertahanan ini dilakukan dalam beberapa tahapan sebagai berikut.

4.1.1 Instalasi Sistem Operasi Raspberry Pi OS

Proses pertama yang dilakukan pada penelitian yaitu mempersiapkan sistem operasi pada perangkat Raspberry Pi 3 model B+. Pada dasarnya, Raspberry Pi tidak memiliki sistem operasi bawaan, sehingga perlu untuk melakukan instalasi sistem operasi pada setiap perangkat yang akan digunakan. Pada penelitian ini menggunakan Raspberry Pi Imager seperti pada Gambar 4.1, yang bertujuan untuk mempermudah proses instalasi sistem operasi Raspberry Pi OS Lite atau Pi OS Lite. Kedua perangkat lunak tersebut dapat diunduh melalui https://www.raspberrypi.com/software/. Penulis memilih Pi OS versi Lite karena pada penelitian fokus dengan antarmuka teks atau CLI (*Command Line Interface*).

Gambar 4.1 Raspberry Pi Imager

Selain itu, penelitian juga mempersiapkan beberapa perangkat lainnya, sehingga dapat diklasifikasikan kebutuhan alat sebagai berikut.

1. Kebutuhan perangkat keras
    a. Raspberry Pi 3 model B+
    b. MicroSD Sandisk 16GB
    c. MicroSD *Adaptor*
    d. Monitor
    e. Keyboard
    f. Kabel HDMI
    g. Kabel *ethernet*
2. Kebutuhan perangkat lunak
    a. Raspberry Pi Imager
    b. Raspberry Pi OS Lite

Proses *burning* sistem operasi Pi OS Lite pada MicroSD menggunakan Raspberry Pi Imager. Pada dasarnya, MicroSD menjadi media penyimpanan Pi OS dan nantinya akan diaplikasikan ke dalam Raspberry Pi 3 model B+. Proses *burning*

Pi OS Lite membutuhkan waktu sekitar 10 menit sampai dinyatakan telah sukses terpasang.

MicroSD yang sudah terpasang Pi OS Lite dimasukkan ke dalam MicroSD *slot* Raspberry Pi 3 model B+ untuk melakukan proses konfigurasi awal. Persiapan perangkat Raspberry Pi 3 model B+ dilakukan dengan memasangkan beberapa perangkat eksternal seperti *monitor* sebagai media *output* berupa *visual*, yang dihubungkan menggunakan kabel HDMI/VGA. Sementara itu menghubungkan perangkat *keyboard* untuk mendukung proses konfigurasi lanjutan. Apabila semuanya sudah terpasang dan Raspberry Pi 3 model B+ telah menyala, muncul beberapa instruksi terkait penyesuaian pengaturan pada Raspberry Pi 3 model B+. Pada penelitian ini, memilih pengaturan *default* dan mengikuti instuksi lainnya hingga persiapan Raspberry Pi 3 model B+ berhasil. Kemudian, melakukan *update* pada versi Pi OS Lite untuk mengurangi kemungkinan kesalahan pada tahapan konfigurasi. Berikut perintah untuk melakukan *update*.

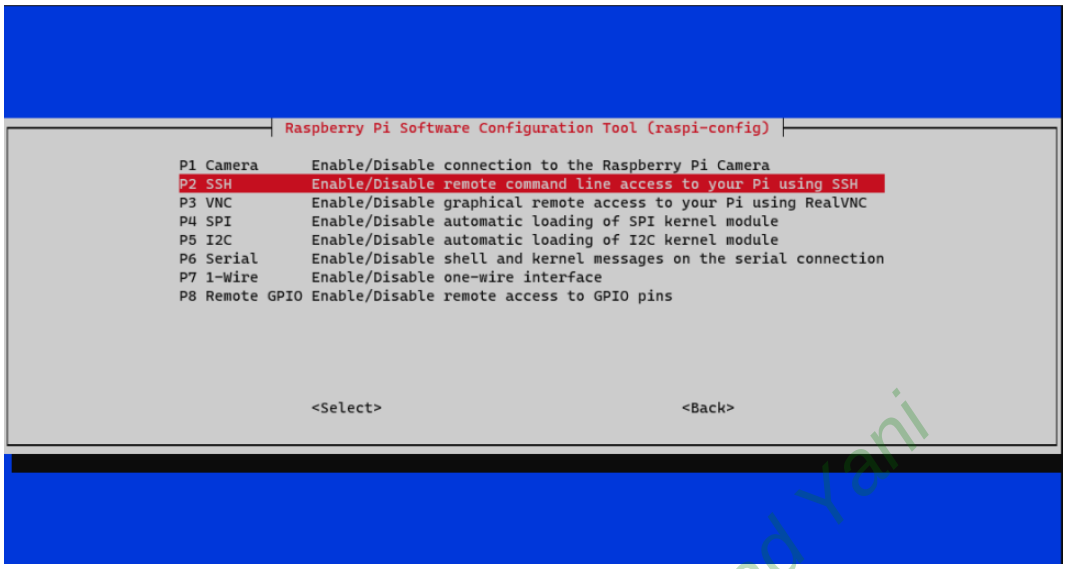
```
sudo apt update && sudo apt dist-upgrade -y
```

4.1.2 Konfigurasi Pi-Hole pada Raspberry Pi 3 model B+

Tahapan selanjutnya yaitu melakukan konfigurasi Pi-Hole pada perangkat Raspberry Pi 3 model B+. Proses ini diawali dengan mempersiapkan SSH *protocol* pada perangkat. SSH berfungsi untuk dapat melakukan akses *terminal* Raspberry Pi secara *remote*. SSH diaktifkan dengan menjalankan perintah pada *terminal* Raspberry Pi

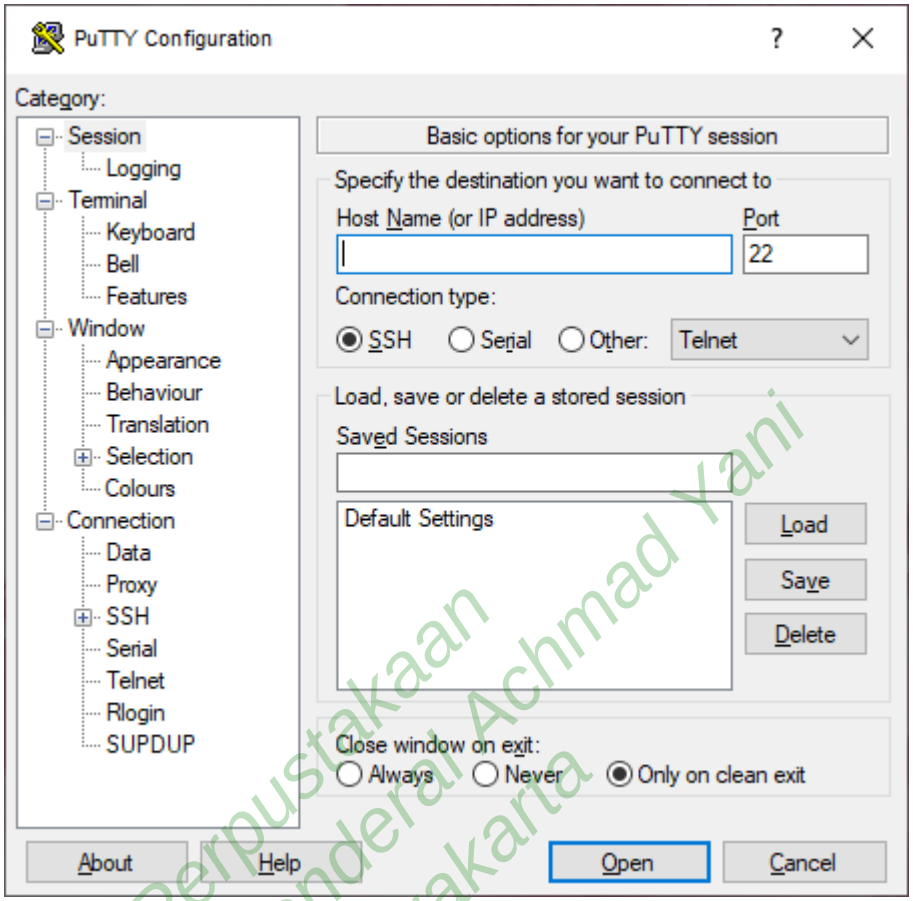
```
raspi-config
```

Maka, muncul tampilan GUI (*Graphical User Interface*) yang menunjukkan beberapa pengaturan awal Raspberry Pi. Konfigurasi SSH ditunjukkan seperti pada Gambar 4.2. Dalam hal ini memilih "***enable* SSH**" untuk mengizinkan *terminal* Raspberry Pi diakses menggunakan SSH. Apabila SSH sudah diaktifkan, kembali pada *terminal* Raspberry Pi dengan menekan **<*Back*>**.



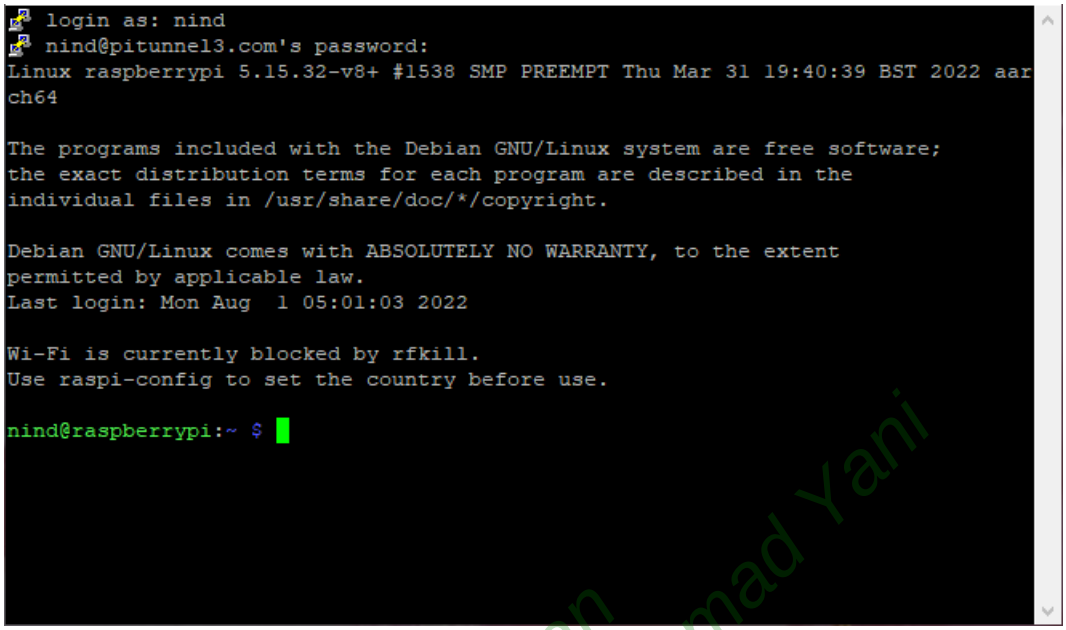
Gambar 4.2 Aktivasi SSH pada Raspberry Pi 3 model B+

Selanjutnya adalah mempersiapkan *terminal emulator* pada perangkat *workstation*, yaitu Laptop Lenovo ThinkPad T420i. Pada penelitian ini menggunakan perangkat lunak PuTTY. Penggunaan PuTTY dapat mempermudah penulis dalam melakukan konfigurasi lanjutan dan proses instalasi Pi-Hole pada Raspberry Pi 3 model B+, serta berfungsi sebagai SSH *client*. Maka dari itu, penulis perlu melakukan instalasi PuTTY dan mempersiapkan beberapa tahapan sebelum PuTTY dapat dijalankan.



Gambar 4.3 PuTTY

Dapat dilihat pada Gambar 4.3, untuk memulai sesi pada perangkat lunak PuTTY maka perlu mengisi kolom *Host Name* dengan alamat IP perangkat Raspberry Pi 3 model B+ yang digunakan yaitu 172.16.13.245, dengan membiarkan *port* ada pada *port* 22, dan memilih tipe koneksi SSH. Sesi baru dapat dimulai dengan menekan ***Open*** untuk mengakses *terminal* Raspberry Pi 3 model B+. Dengan itu, PuTTY mendapatkan akses ke *terminal* Raspberry Pi 3 model B+ dan dapat menjalankan berbagai perintah sebagaimana menuliskan perintah pada Raspberry Pi. Pada Gambar 4.4 menunjukkan tampilan *terminal* ketika *client* sudah melakukan *login* ke *server* melalui PuTTY. Pada sesi tersebut, *client* diminta untuk memasukkan *username* dan *password* yang sudah dibuat pada tahapan konfigurasi Pi OS sebelumnya.



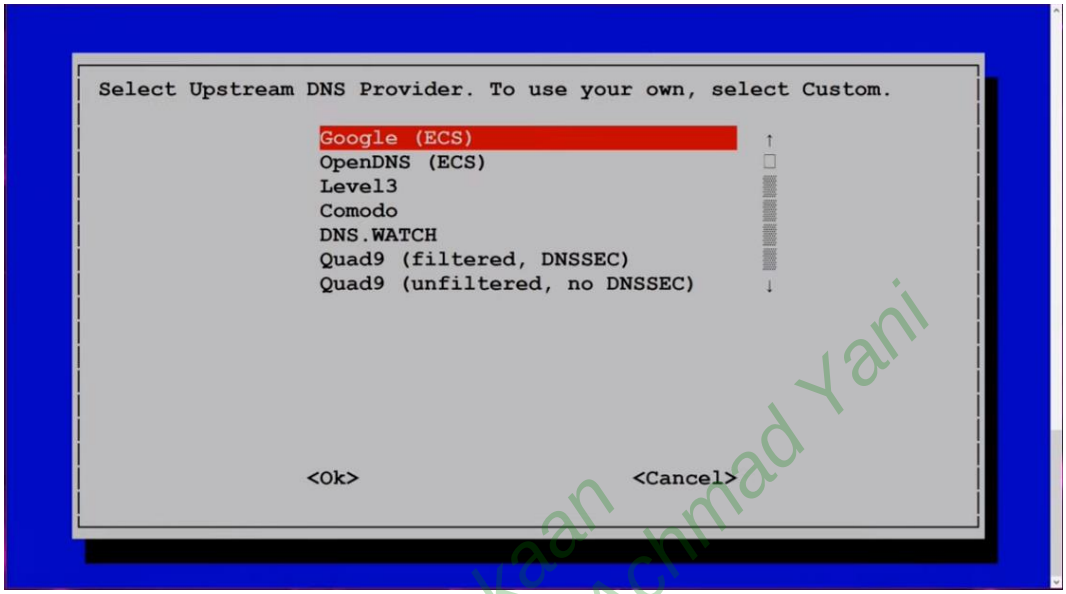
Gambar 4.4 *Terminal* Raspberry Pi 3 model B+ pada PuTTY

Tahapan selanjutnya yaitu melakukan konfigurasi Pi-Hole melalui PuTTY. Proses konfigurasi Pi-Hole dilakukan dengan masuk ke dalam bagian *root* terlebih dahulu, guna menjalankan perintah yang mengandalkan *root*. Konfigurasi Pi-Hole dilakukan dengan menjalankan perintah sebagai berikut.

```
sudo su -
curl -sSL https://install.pi-hole.net | bash
```

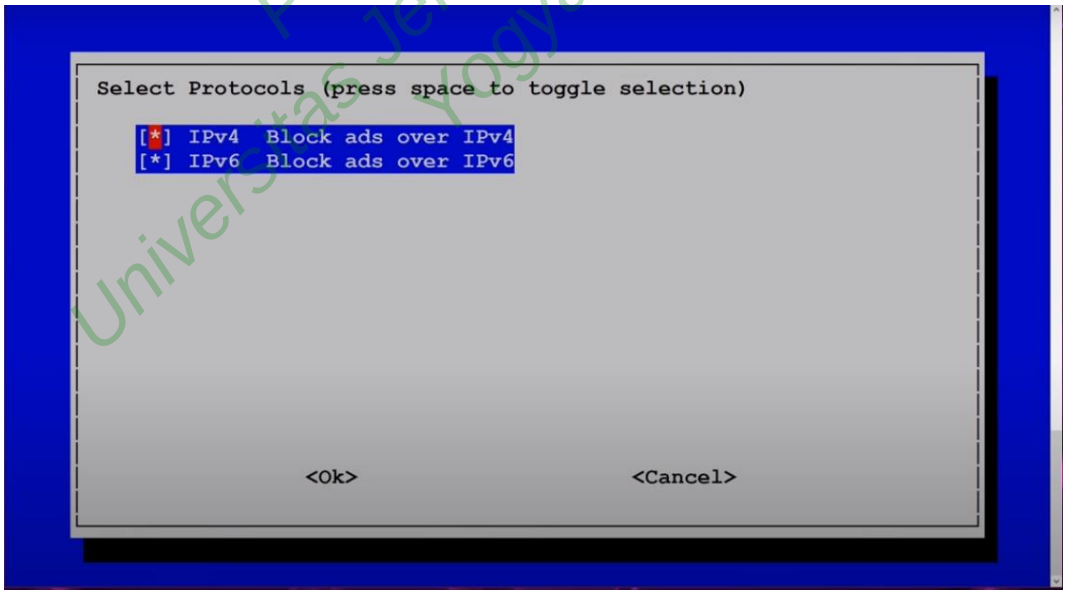
Pada skenario tersebut, Raspberry Pi memerlukan beberapa konfigurasi tambahan untuk mendukung proses instalasi Pi-Hole. Hal ini agar perangkat dapat menyesuaikan kebutuhan pengelola jaringan terkait penggunaan Pi-Hole. Pada proses konfigurasi ini, *terminal root* Raspberry Pi akan langsung beralih ke tampilan GUI dan terdapat beberapa instruksi terkait pengaturan sistem. Pada penelitian, ada beberapa poin yang perlu diperhatikan dan diatur sesuai kebutuhan penelitian, diantaranya sebagai berikut.

1. Pengaturan *Upstream DNS Provider*, dimana penelitian menggunakan *provider* Google (ECS) seperti pada Gambar 4.5.



Gambar 4.5 Pengaturan *Upstream DNS Provider*

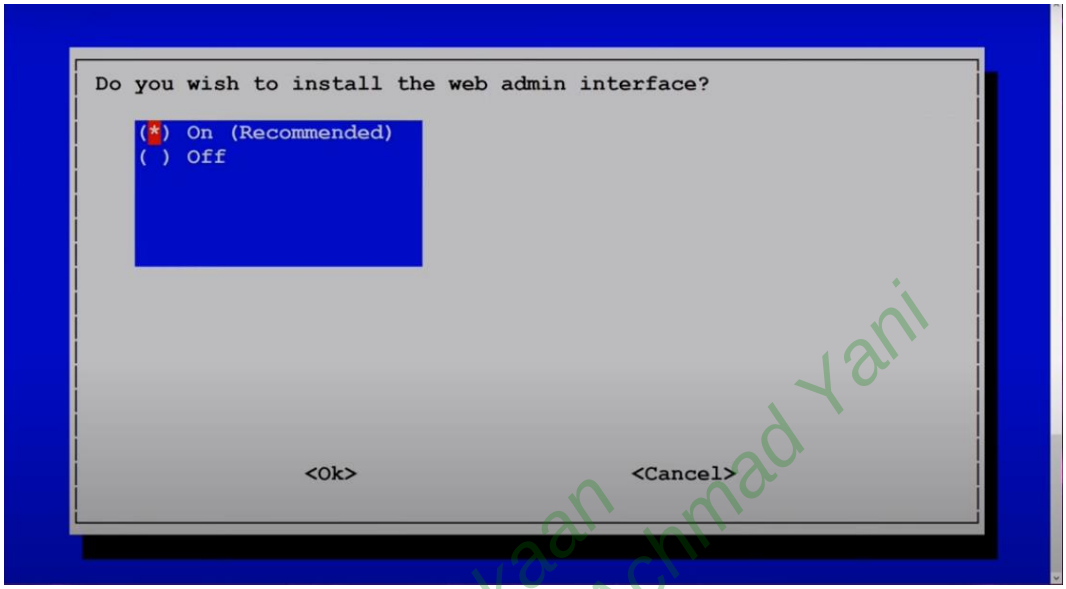
2. Pengaturan *Internet Protocol*. Pada penelitian ini memilih protokol IPv4, seperti pada Gambar 4.6.



Gambar 4.6 Pengaturan *Internet Protocol*

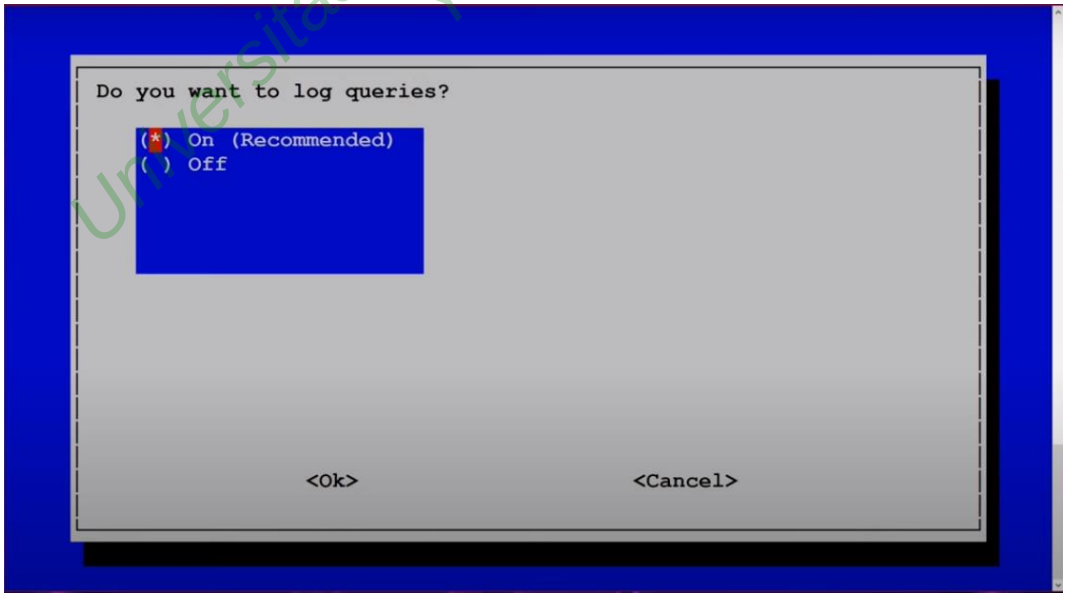
3. Pengaturan *web admin*, penelitian membutuhkan *interface* untuk *web admin* agar mempermudah proses pemantauan sistem dan pengambilan data. Maka

dari itu, pengaturan *web admin* harus dipastikan ***On*** seperti pada Gambar 4.7.



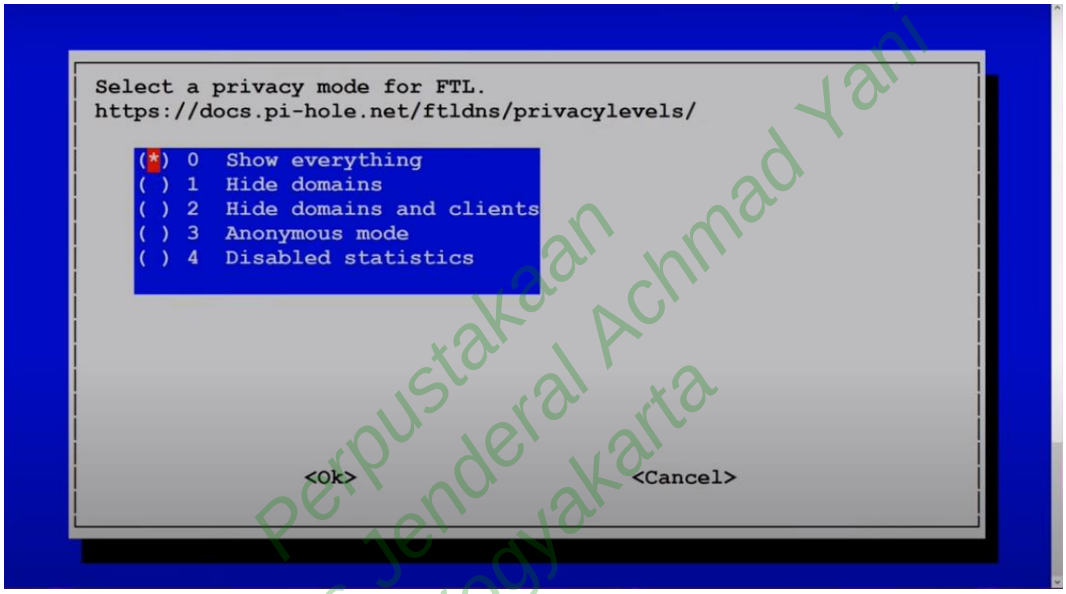
Gambar 4.7 Pengaturan Instalasi *Web Admin Interface*

4. Pengaturan *log query*. Pada pilihan yang tertera pada Gambar 4.8, menunjukkan apakah Pi-Hole diizinkan untuk mencatat seluruh *query*-nya ke dalam *log* agar dapat dilakukan pemantauan *log query*. Pastikan untuk mengaktifkan *log query* dengan memilih ***On***.



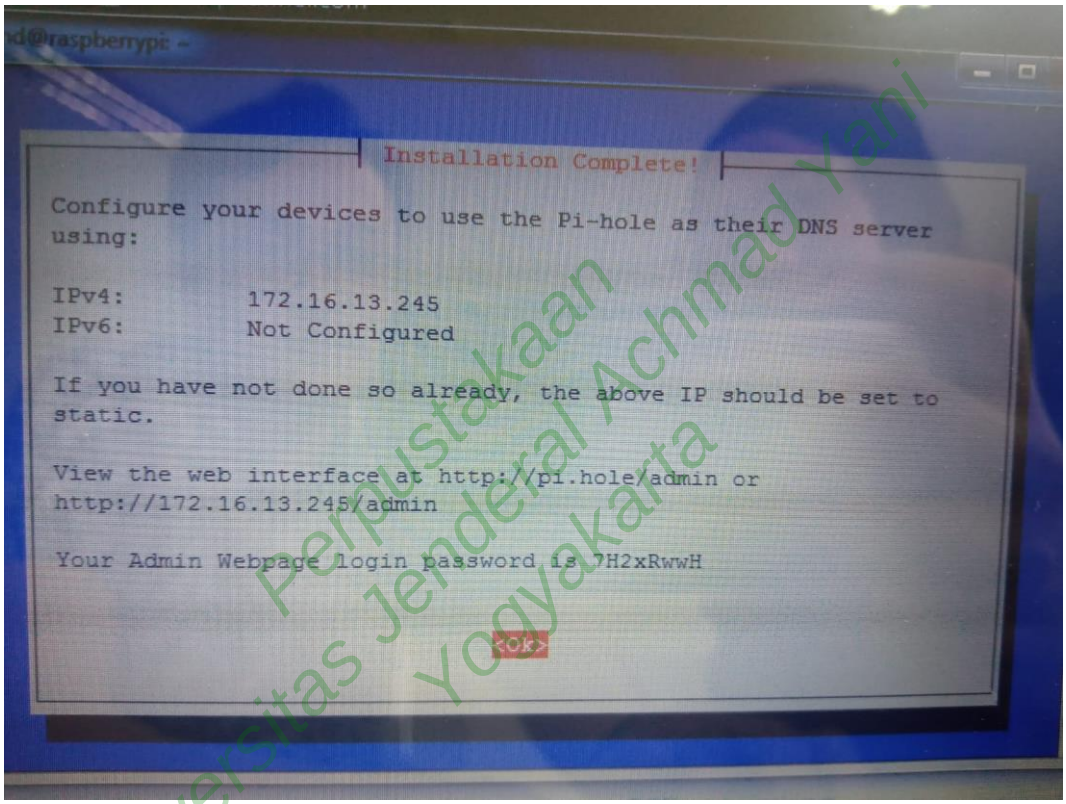
Gambar 4.8 Pengaturan *Log Query*

5. Pengaturan privasi pada FTL. FTL (*Faster Than Light*) berfungsi sebagai penyimpanan seluruh informasi DNS *resolver* dan disajikan dalam API (*Application Programming Interface*) yang bersifat interaktif, selain itu FTL menghasilkan statistik yang dapat ditampilkan ke *dashboard web admin* Pi-Hole. Pastikan untuk memilih ***Show Everything*** agar dapat memantau seluruh *query log* yang masuk ke dalam Pi-Hole. Pengaturan privasi FTL seperti pada Gambar 4.9.



Gambar 4.9 Pengaturan Privasi FTL Pi-Hole

6. Pada tahapan akhir dari instalasi Pi-Hole, Raspberry Pi menampilkan informasi mengenai alamat URL yang dapat digunakan untuk mengakses *web admin* Pi-Hole. Informasi tersebut juga menunjukkan *password* yang dapat dipakai untuk melakukan *login* pada *web admin* Pi-Hole. Selain itu, IP yang tertera dapat digunakan untuk konfigurasi Pi-Hole sebagai DNS *server* seperti halnya pada Gambar 4.10.



Gambar 4.10 Informasi Akhir pada Konfigurasi Pi-Hole

7. Setelah proses konfigurasi selesai dijalankan, Raspberry Pi akan kembali ke tampilan *terminal* dan menjalankan instalasi Pi-Hole secara keseluruhan. Raspberry Pi menjalankan proses instalasi sesuai pengaturan yang sudah ditentukan sebelumnya.

8. Melakukan *update* pada Pi-Hole. Tahapan ini bertujuan agar Pi-Hole dapat digunakan dalam versi terbaru dan mengurangi *troubleshoot* saat mengakses Pi-Hole. Proses *update* Pi-Hole dilakukan dengan menjalankan perintah sebagai berikut.

   ```
   pihole version
   pihole updatePihole
   ```

9. Pi-Hole memiliki opsi untuk mengganti *password* yang sebelumnya diberikan oleh Pi-Hole. *Password* tersebut berfungsi sebagai kode *login* pada saat mengakses *web admin* Pi-Hole. Pengubahan *password* dilakukan agar mempermudah penulis pada saat *login* untuk mengakses *web admin* Pi-Hole. Maka dari itu, dilakukan dengan menjalankan perintah sebagai berikut.

   ```
   pihole -a -p
   ```

   Pada proses ini, Pi-Hole meminta untuk menuliskan *password* baru yang lebih mudah diingat oleh penulis. Demikian tahapan persiapan selesai dengan terpasangnya Pi-Hole pada perangkat Raspberry Pi 3 model B+.

4.1.3 Konfigurasi Jaringan Pada Raspberry Pi 3 Model B+

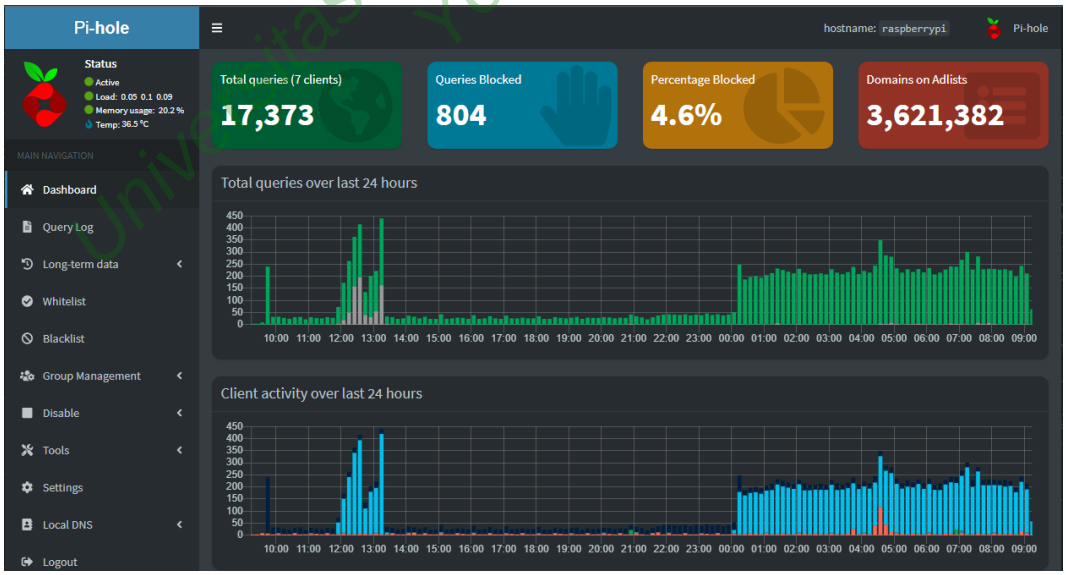
Tahapan konfigurasi jaringan pada Raspberry Pi 3 model B+ yang sudah terinstal Pi-Hole, kemudian diintegrasikan dalam DMZ (*Demilitarized Zone*) yang terdapat pada jaringan FTTI Universitas Jenderal Achmad Yani Yogyakarta. Pada skenario ini, penulis melakukan koordinasi dengan *Network Engineer* yang menangani komunikasi jaringan FTTI Universitas Jenderal Achmad Yani Yogyakarta untuk melakukan konfigurasi DNS *server* pada ISP (*Internet Service Provider)* sehingga dapat diarahkan ke IP Pi-Hole. Selain itu, penulis berkoordinasi kepada *Network Engineer* FTTI Universitas Jenderal Achmad Yani Yogyakarta untuk konfigurasi DHCP *server* pada Raspberry Pi 3 model B+. Pengaturan DHCP *server* dilakukan dengan tujuan mendistribusikan alamat IP ke komputer *client* secara otomatis selama berada dalam satu jaringan yang sama.

4.1.4 Pengujian Sistem

Tahapan pengujian sistem dilakukan dengan tujuan memantau hasil pengembangan sistem pertahanan menggunakan Pi-Hole yang dikonfigurasikan

pada perangkat Raspberry Pi 3 model B+ guna memerangi serangan *malvertising*. Pengujian juga dilakukan untuk memastikan apakah sistem sudah berfungsi dengan baik tanpa adanya eror pada sistem. Skenario tersebut dimulai dengan mengakses *web admin* Pi-Hole yaitu http://172.16.12.94/admin. URL *web admin* Pi-Hole menunjukkan IP yang berbeda dengan IP Pi-Hole sebelumnya, yaitu http://172.16.13.245/admin. Hal ini terjadi karena perangkat Raspberry Pi 3 model B+ telah terintegrasi dengan DMZ pada jaringan FTTI Universitas Jenderal Achmad Yani Yogyakarta, sehingga IP Pi-Hole secara otomatis berubah menyesuaikan IP pada ISP yang digunakan FTTI Universitas Jenderal Achmad Yani Yogyakarta.

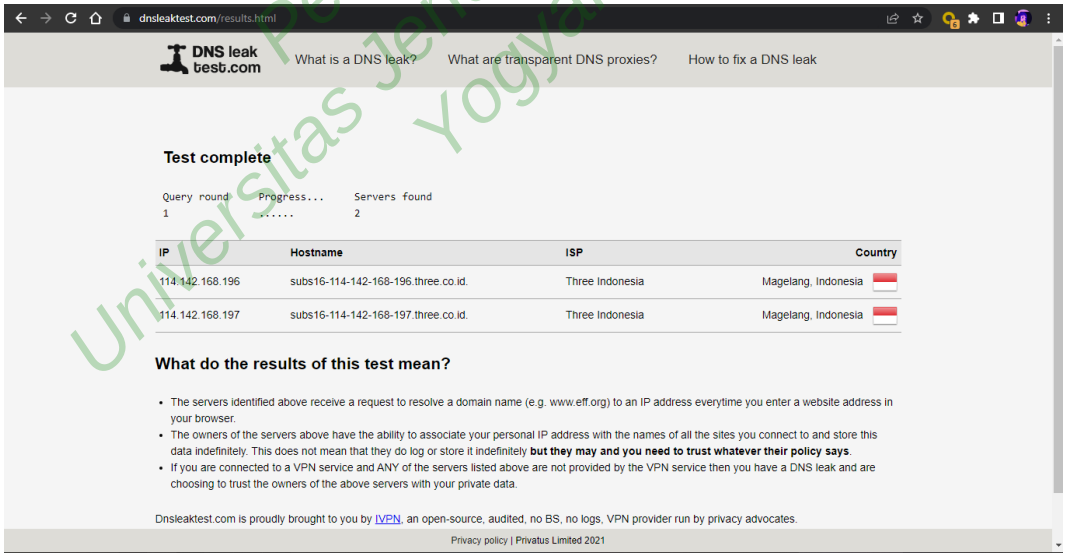
Laptop Lenovo ThinkPad T420i yang sudah terhubung dengan *web admin* Pi-Hole membutuhkan *login* terlebih dahulu untuk dapat mengakses dan memantau *log query* secara lebih *detail* dengan menuliskan *password. Password* yang diisi merupakan *password* yang sebelumnya sudah dibuat oleh penulis pada saat proses konfigurasi Pi-Hole berlangsung. Proses *login* dilakukan setiap kali penulis mengakses Pi-Hole walaupun melalui perangkat yang sama. Maka, seluruh data yang berisi performa Pi-Hole akan ditampilkan dalam *dashboard* seperti pada Gambar 4.11.



Gambar 4.11 *Dashboard Web Admin* Pi-Hole

Selain itu, dengan berjalannya Pi-Hole pada jaringan perlu dibuktikan apakah DNS yang dimiliki Pi-Hole sudah terenkripsi atau justru rentan terjadi kebocoran DNS. Kebocoran DNS atau DNS *leak* mengacu pada cacat keamanan pada DNS *request* ketika *client* mengakses situs *web* melalui ISP yang bersifat non-anonim, tidak seperti VPN dan *dedicated DNS server*. Hal ini disebabkan karena saat menggunakan layanan *internet*, *client* akan menjalankan DNS *request* pada ISP dan seluruh trafik jaringan akan tercatat oleh DNS *server*. Apabila DNS *server* mengalami kebocoran karena tidak terenkripsi, setiap musuh yang melacak trafik jaringan tersebut dapat mencatat seluruh aktivitas *client*. Akibatnya hal ini juga menjadi ancaman privasi bagi *client* karena jaringan mungkin menjadi salah satu sebab bocornya data pribadi *client*.

Pada penelitian ini menggunakan situs DNSLeakTest yang dapat diakses melalui https://www.dnsleaktest.com/results.html. Pada skenario ini dilakukan perbandingan antara komunikasi DNS yang terenkripsi dan yang belum. Berikut ini merupakan kondisi jaringan sebelum terhubung ke jaringan FTTI Universitas Jenderal Achmad Yani Yogyakarta.

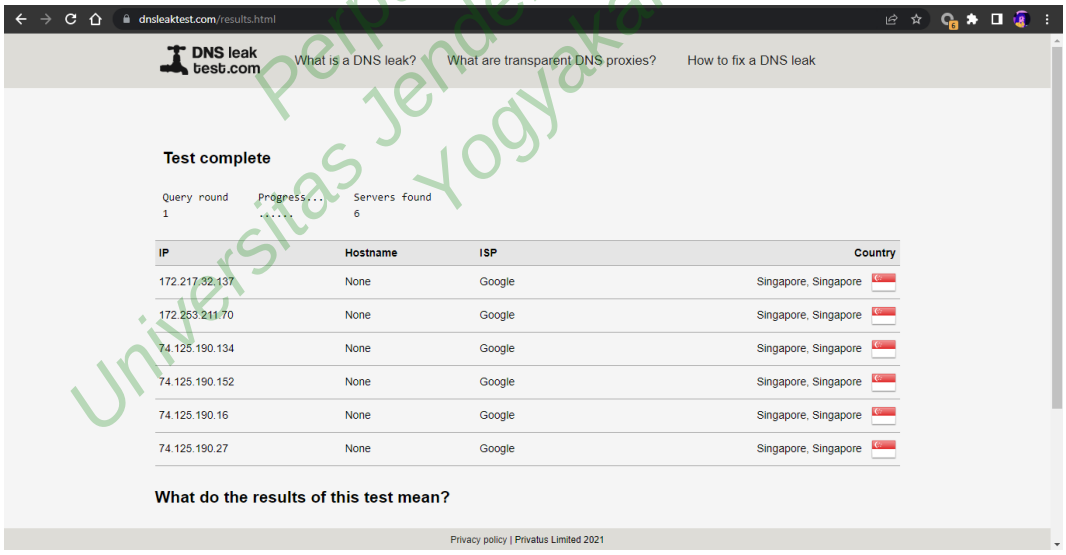


Gambar 4.12 Pengujian DNSLeakTest Sebelum Terhubung ke Jaringan FTTI Universitas Jenderal Achmad Yani Yogyakarta

Berdasarkan Gambar 4.12 menunjukkan bahwa ditemukan dua IP perangkat *client* yang terhubung ke dalam satu ISP. IP *client* diketahui menjadi DNS *resolver*

atas pemetaan nama *domain* (misalnya www.nama.com) setiap melakukan pencarian di *web browser*. Maka, dapat disimpulkan bahwa DNS *server* yang disediakan ISP menjadi penangkap setiap DNS *request* oleh *client*. Pada Gambar 4.12 tersebut juga menunjukkan informasi *server* yang ditemukan dalam jaringan, mulai dari ISP yang digunakan, lokasi akses, bahkan IP perangkatnya merupakan informasi yang sebenarnya.

Kondisi seperti ini dapat menjadi sebuah ancaman karena pada dasarnya DNSLeakTest dapat membaca informasi DNS *query* pada ISP yang tidak terenkripsi. Pada akhirnya, apabila DNS *server* yang muncul merupakan DNS *server* milik ISP, maka sangat rentan terjadi kebocoran DNS. Penyebab utamanya adalah tidak adanya enkripsi terhadap setiap DNS *query* yang disimpan dalam DNS *server* ISP, dalam hal ini disebut sebagai DoH (DNS-*over*-HTTPS). Apabila ISP tidak memiliki DoH pada DNS *server*, maka berisiko terjadinya kebocoran DNS. Akan tetapi, Pi-Hole dapat mengamankan jalur komunikasi sehingga tidak dapat terbaca oleh DNSLeakTest. Hal ini dapat ditunjukkan pada Gambar 4.13.



Gambar 4.13 Pengujian DNSLeakTest Setelah Terhubung ke Jaringan FTTI Universitas Jenderal Achmad Yani Yogyakarta

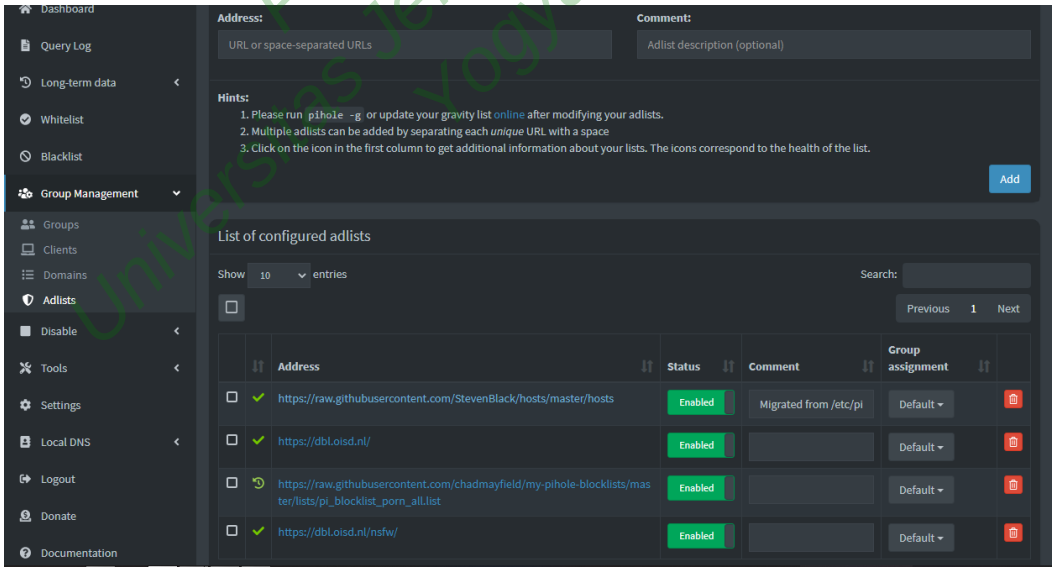
Gambar 4.13 menunjukkan bahwa perangkat *client* telah terhubung ke dalam jaringan FTTI Universitas Jenderal Achmad Yani Yogyakarta. Hasil DNSLeakTest menampilkan lokasi, IP, dan ISP yang hanya sebuah ketidakaturan. Dalam artian, DNSLeakTest tidak menampilkan informasi *server* yang sebenarnya.

Hal ini menjadi tanda bahwa jaringan FTTI Universitas Jenderal Achmad Yani Yogyakarta yang sudah terintegrasi dengan Pi-Hole telah menerapkan DoH pada DNS *server* sehingga setiap *query*-nya terenkripsi dengan baik. Maka demikian, pengujian tersebut menyatakan Pi-Hole mampu mengamankan jalur komunikasi dengan baik.

4.1.5 Update Blocklist

Pada tahapan ini melakukan *update* pada *blocklist* yaitu *adlist* dan *blacklist. Adlist* merupakan daftar *domain* terkait layanan iklan daring yang mungkin disisipkan pada situs *web* maupun aplikasi berbasis *web*. Jenis iklan yang disisipkan dapat berupa iklan *pop up*, iklan *banner*, iklan *pop under*, bahkan iklan *push*. Seluruh iklan daring yang muncul pada situs *web* dapat mengganggu kenyamanan *client* pada saat mengaksesnya. Pada kemungkinan terburuknya, iklan daring tersebut dapat disisipi *malware* dan menyerang perangkat *client* yang mengakses iklan daring tersebut secara disengaja maupun tidak. *Update adlist* dilakukan dengan cara sebagai berikut.

1. *Update Adlist*

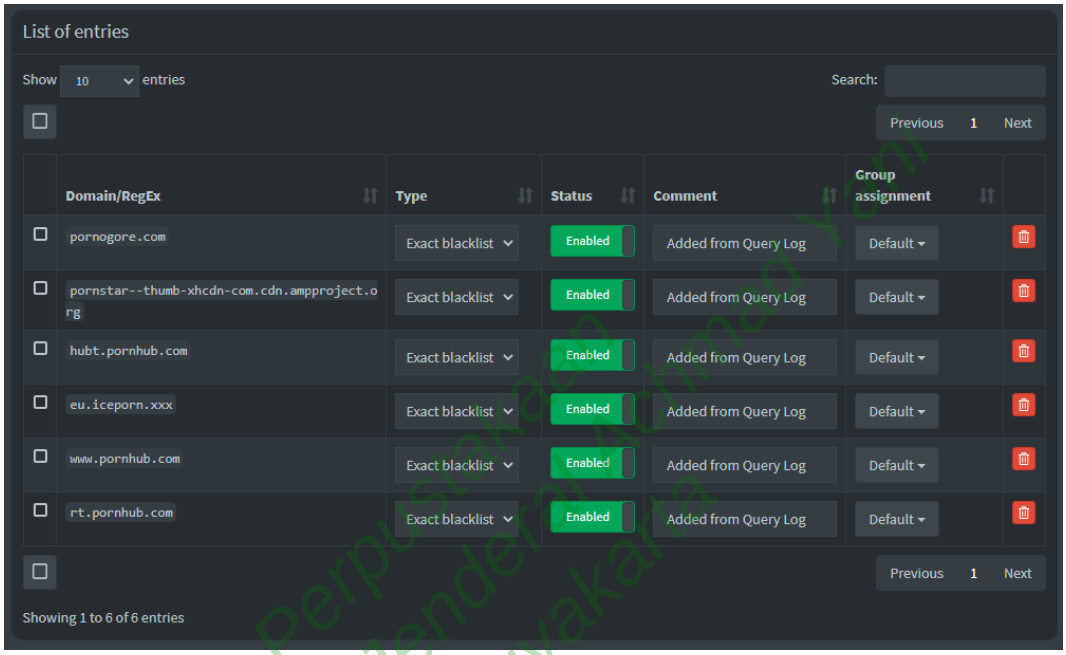


Gambar 4.14 *Adlist* Pi-Hole

Pada Gambar 4.14 berikut ini merupakan tampilan *adlist*, dimana *adlist* dapat ditambahkan ke dalam *input* yang tersedia. Pada umumnya, seluruh

adlist sudah disajikan dalam satu URL khusus dan pencarian *adlist* dapat ditemukan melalui GitHub atau *blog* berbasis komunitas. Pada penelitian, beberapa URL berisi ribuan *adlist* telah ditambahkan ke dalam daftar dan siap untuk diproses ke tahapan pemblokiran.

2. *Update Blacklist*

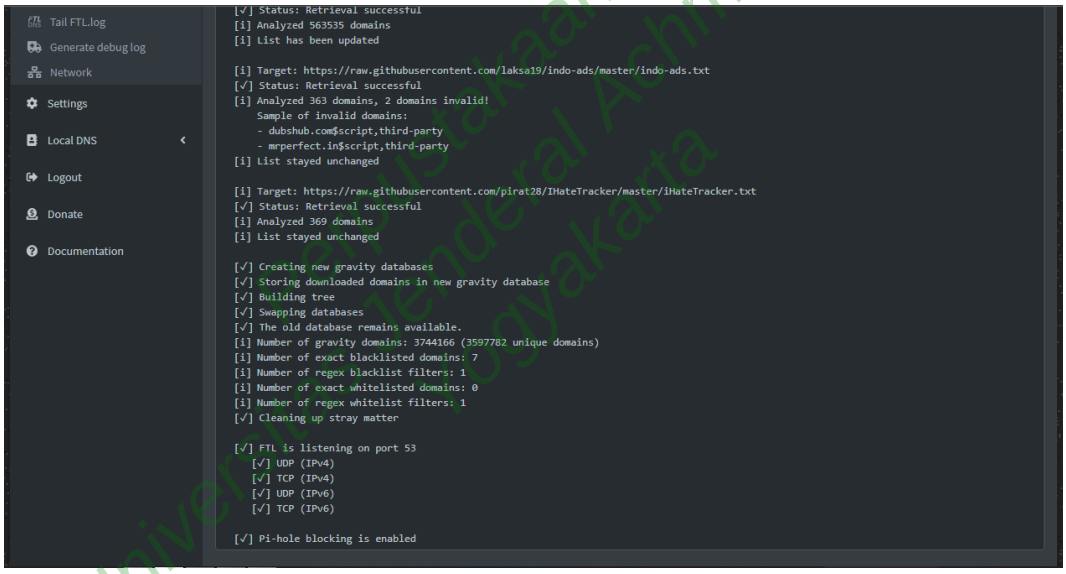


Gambar 4.15 *Blacklist* Pi-Hole

Selain melakukan *update adlist*, pengelola jaringan juga dapat menambahkan *blocklist* yaitu *blacklist* pada *domain*. Berdasarkan Gambar 4.15 menunjukkan bahwa *domain* yang termasuk ke dalam daftar merupakan *domain* yang telah dimasukkan ke dalam *blacklist.* Maka dari itu, dapat dikatakan bahwa tidak hanya *adlist* yang dapat ditambahkan ke dalam *blocklist*, tetapi pemblokiran *domain* juga dapat dilakukan. Hal ini bertujuan untuk memblokir situs-situs yang mengandung konten tidak baik. Pada daftar yang ditunjukkan Gambar 4.15 merupakan *domain* situs pornografi. Untuk menambahkan *domain* ke dalam daftar *blacklist* domain juga dilakukan secara *manual*.

3. *Update Gravity*

Setelah penambahan daftar *adlist* dan *blacklist* seperti yang dilakukan pada poin **1** dan **2** sebelumnya, dapat diketahui bahwa penambahan *blocklist* berupa *adlist* dan *blacklist* hanya bisa dilakukan secara *manual*. Akan tetapi, dengan menambahkan *blocklist* juga perlu update *gravity* Pi-Hole setelahnya. Hal ini dikarenakan seluruh *domain* yang masuk ke dalam daftar blokir belum sepenuhnya diproses oleh Pi-Hole, sehingga *gravity* berfungsi untuk memproses seluruh *blocklist* agar dapat diblokir oleh Pi-Hole. Pada penelitian, membarui *gravity* Pi-Hole dilakukan setiap kali adanya penambahan *blocklist.* Pengelola jaringan juga dapat melakukan *update gravity* secara langsung tanpa perlu menambahkan *blocklist* terlebih dahulu.



Gambar 4.16 *Update Gravity*

Seperti yang ditunjukkan pada Gambar 4.16, proses pemblokiran tidak hanya terjadi pada *adlist*, namun berlaku juga untuk menyelesaikan proses pemblokiran Pi-Hole yang lain, yaitu *blacklist*. Lamanya proses *update gravity* bergantung dari banyaknya *blocklist* yang ditambahkan sebelumnya. Pada penelitian, proses *update gravity* dapat berjalan hingga 15 menit. Demikian proses *update gravity* selesai, maka Pi-Hole telah memproses seluruh *adlist* sehingga *domain* dan *adlist* yang diblokir pun tidak dapat diakses *client*.

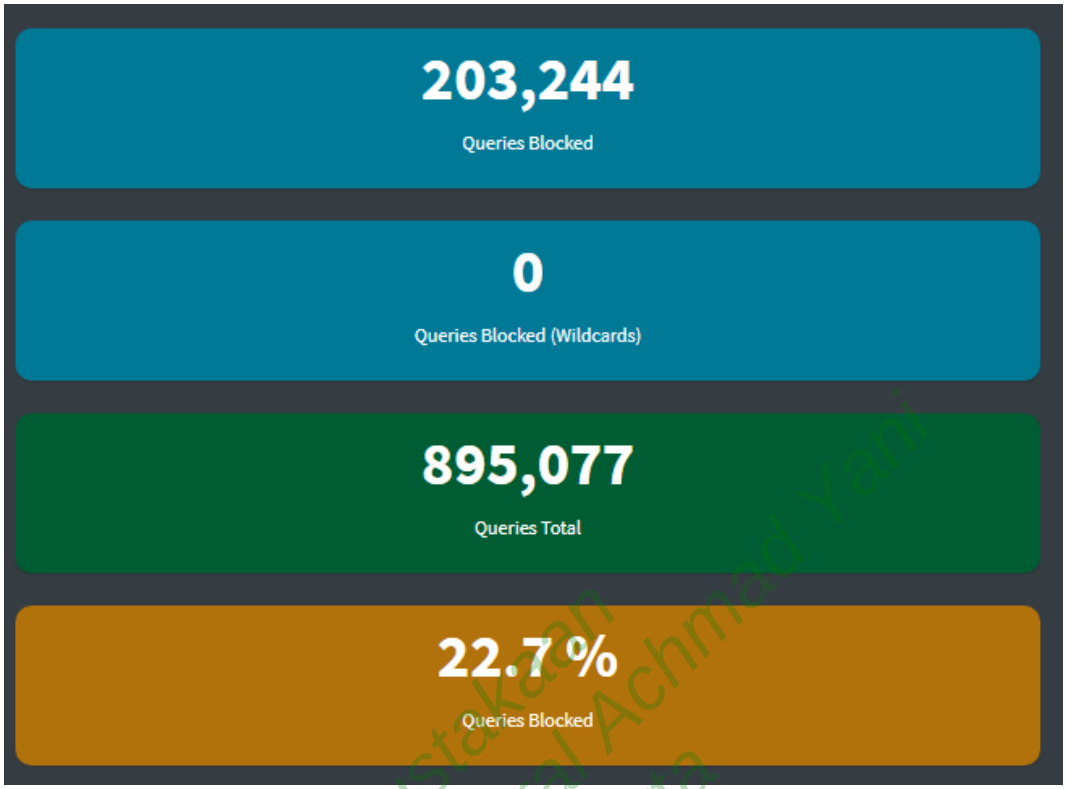
4.2 PEMBAHASAN

Penelitian mengenai implementasi Pi-Hole yang dikonfigurasikan pada perangkat Raspberry Pi 3 model B+ dan diintegrasikan dalam DMZ jaringan FTTI Universitas Jenderal Achmad Yani Yogyakarta menghasilkan beberapa temuan yang dapat dianalisis secara lebih lanjut. Hasil yang diperoleh dari penelitian yaitu data *log query* yang dicatat oleh Pi-Hole selama 14 hari, yaitu tanggal 26 Juli sampai dengan 8 Agustus 2022. Pemantauan performa Pi-Hole dilakukan dengan menghubungkan perangkat Laptop Lenovo ThinkPad T420i dengan jaringan FTTI Universitas Jenderal Achmad Yani Yogyakarta dan melakukan akses pada URL http://172.16.12.94/admin. Jumlah *log query* ditentukan dari seberapa banyak *client* yang mengakses situs *web* dan *domain* yang diakses.

Penelitian ini dilakukan pengujian terhadap sistem pertahanan. Pengujian tersebut dapat digunakan untuk mengukur tingkat efektivitas Pi-Hole dalam memerangi serangan *malvertising* pada jaringan FTTI Universitas Jenderal Achmad Yani Yogyakarta. Demikian hal yang diuji adalah sebagai berikut.

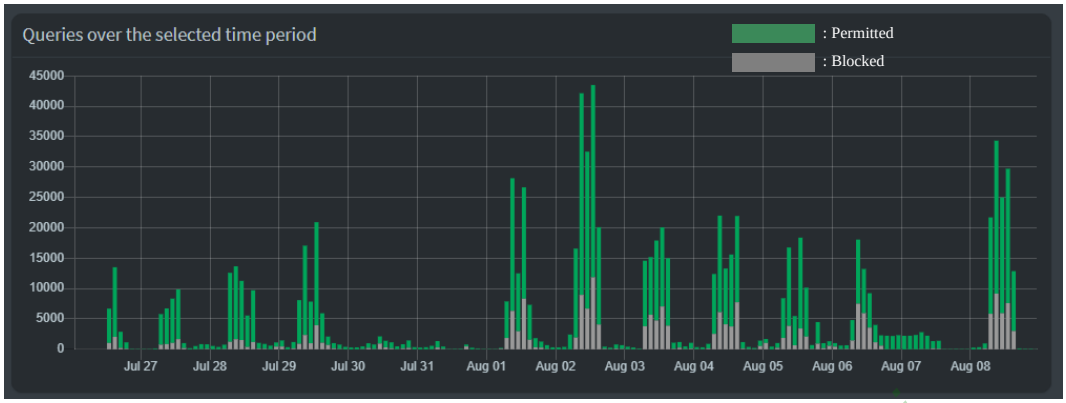
1. Jumlah *log query* yang diblokir Pi-Hole dalam jangka waktu 14 hari
2. Daftar teratas pada Pi-Hole
3. Pengujian akses situs *web* yang berpotensi serangan *malvertising*
4. Perbandingan jumlah iklan daring yang muncul pada 20 situs *web* dan 2 aplikasi berbasis *mobile*

Maka dari itu, penelitian menghasilkan beberapa temuan yang dapat diambil untuk analisis, seperti yang digambarkan pada Gambar 4.17.



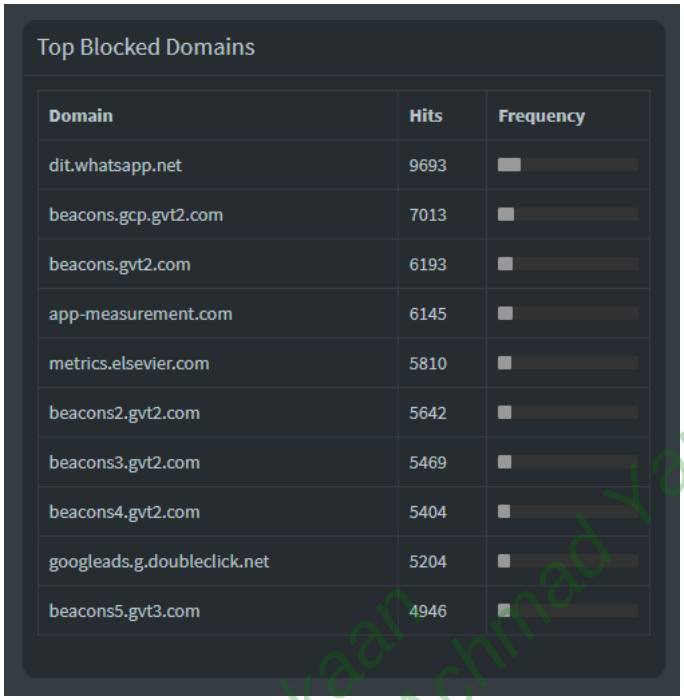
Gambar 4.17 Informasi *Log Query* Pi-Hole

Temuan penelitian yang pertama ditampilkan pada Gambar 4.17. Informasi tersebut menunjukkan *log query* secara keseluruhan dalam jangka waktu 14 hari. Gambar 4.16 menunjukkan sebanyak **203.244** *query* diblokir oleh Pi-Hole dari total *query* yang masuk yaitu sebanyak **895.077** *query*. Dengan demikian, sistem implementasi Pi-Hole berhasil memblokir sebanyak **22,7%** dari total *query* yang masuk ke dalam Pi-Hole. *Query* yang merupakan *wildcard* menunjukkan angka nol, yang artinya tidak ada *client* yang mengakses situs dengan *domain* yang terdapat dalam *wildcard*. Fitur *wildcard* berfungsi memblokir segala jenis kombinasi *domain* yang memiliki istilah yang masuk ke dalam kamus Pi-Hole. *Domain* yang bukan termasuk *wildcard*, akan ditetapkan sebagai *Exact Blacklist*. Sementara, *domain* yang termasuk dalam *wildcard* akan ditetapkan sebagai *Regex Blacklist*.



Gambar 4.18 Grafik *Query Log* Pi-Hole

Sementara itu, grafik yang ditunjukkan pada Gambar 4.18 menggambarkan banyaknya *query* yang masuk pada jangka waktu 14 hari. Berdasarkan grafik tersebut, dapat dikatakan bahwa semakin banyak akses yang dilakukan *client*, maka potensi serangan *malvertising* juga semakin tinggi. Pernyataan tersebut dibuktikan dengan fakta bahwa *client* yang terhubung pada jaringan FTTI Universitas Jenderal Achmad Yani Yogyakarta paling banyak tanggal 2 Agustus 2022, dengan jumlah *query* yang terblokir pun mencetak angka terbanyak. Namun seiring dengan jumlah akses yang dilakukan pada tanggal 3 Agustus 2022, dengan jumlah akses lebih sedikit dibandingkan dengan hari sebelumnya, membuktikan bahwa lebih sedikit melakukan akses jaringan, maka potensi serangan *malvertising* juga semakin berkurang. Namun dengan berkurangnya akses jaringan tidak menutup kemungkinan terjadinya serangan *malvertising*.

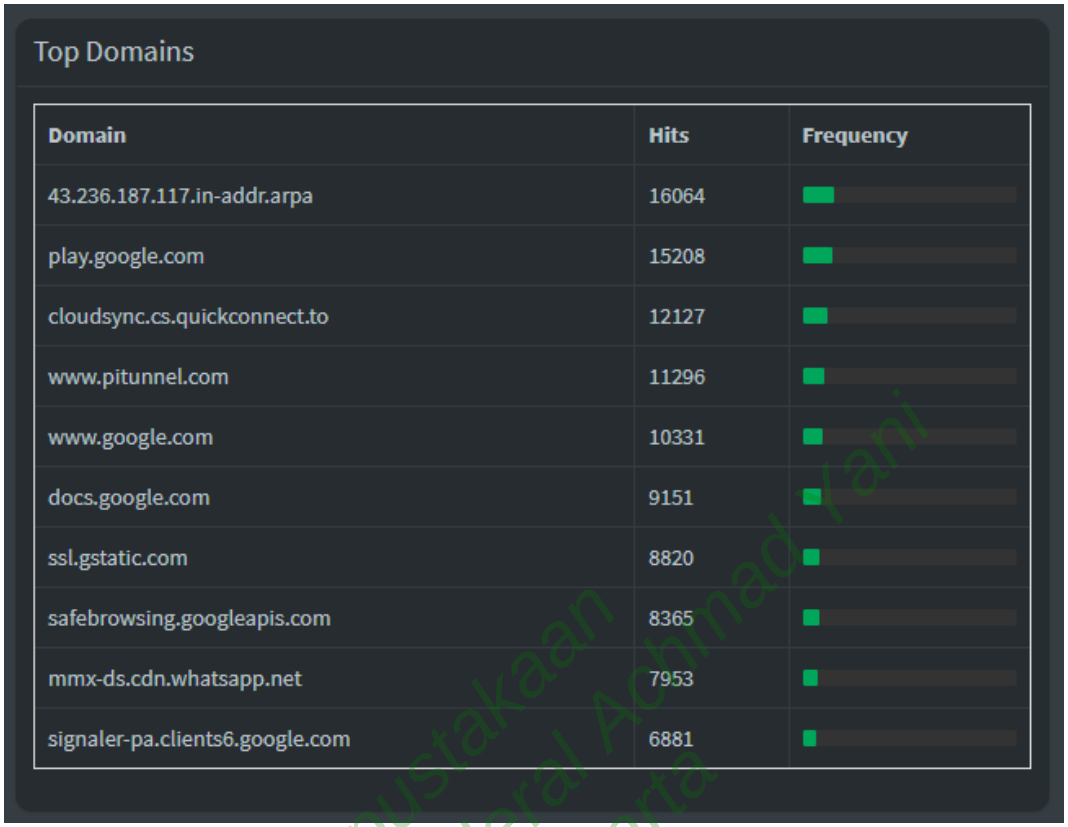


Top Blocked Domains

Domain	Hits	Frequency
dit.whatsapp.net	9693	
beacons.gcp.gvt2.com	7013	
beacons.gvt2.com	6193	
app-measurement.com	6145	
metrics.elsevier.com	5810	
beacons2.gvt2.com	5642	
beacons3.gvt2.com	5469	
beacons4.gvt2.com	5404	
googleads.g.doubleclick.net	5204	
beacons5.gvt3.com	4946	

Gambar 4.19 *Domain* yang Paling Sering Diblokir Pi-Hole

Sementara itu, pada Gambar 4.19 menunjukkan 10 *domain* teratas yang paling banyak diblokir oleh Pi-Hole selama 14 hari. *Domain* dit.whatsapp.net dinyatakan sebagai *domain* yang paling sering diblokir, yaitu sebanyak 9693 kali. Maka dapat dikatakan bahwa Pi-Hole berhasil mengurangi risiko serangan *malvertising* dengan memblokir iklan daring yang muncul pada situs *web* dan aplikasi berbasis *web*.

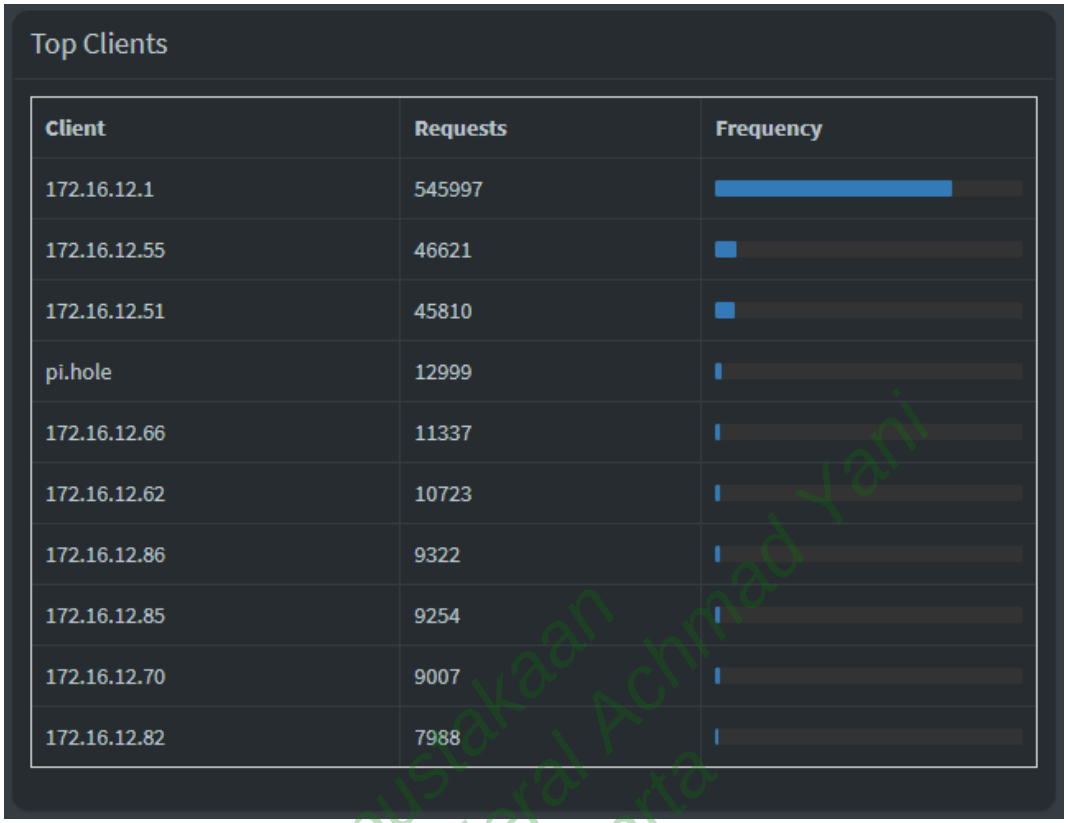


Top Domains

Domain	Hits	Frequency
43.236.187.117.in-addr.arpa	16064	
play.google.com	15208	
cloudsync.cs.quickconnect.to	12127	
www.pitunnel.com	11296	
www.google.com	10331	
docs.google.com	9151	
ssl.gstatic.com	8820	
safebrowsing.googleapis.com	8365	
mmx-ds.cdn.whatsapp.net	7953	
signaler-pa.clients6.google.com	6881	

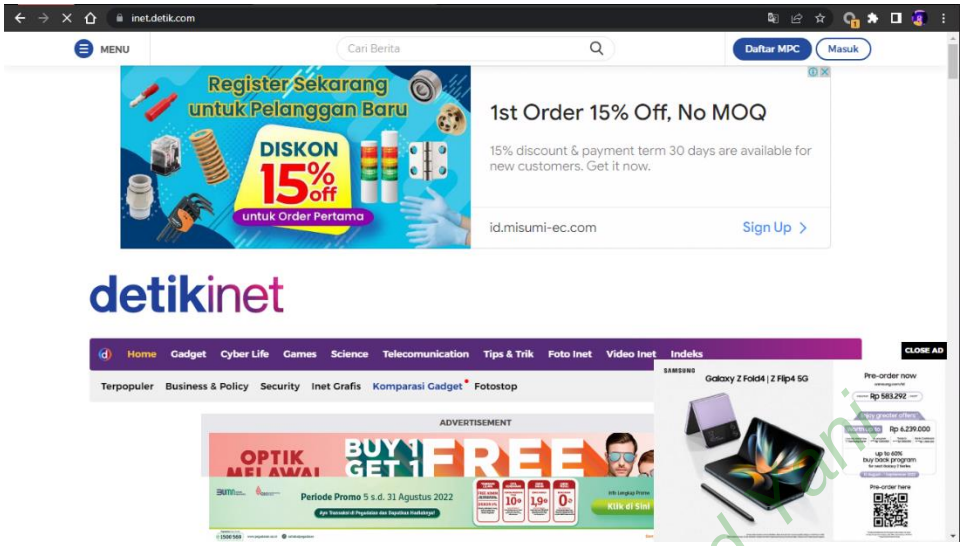
Gambar 4.20 *Domain* yang Paling Sering Diakses *Client*

Sedangkan pada Gambar 4.20 menunjukkan 10 daftar *domain* yang paling sering diakses oleh *client*. Pada data yang ditangkap Pi-Hole mendapatkan temuan bahwa *domain* dengan nama 43.236.177.in-addr.arpa diakses sebanyak 16064 kali dalam jangka waktu 14 hari. Sehingga dapat dikatakan bahwa Pi-Hole dapat menampilkan daftar *domain* yang paling sering diakses oleh *client*.



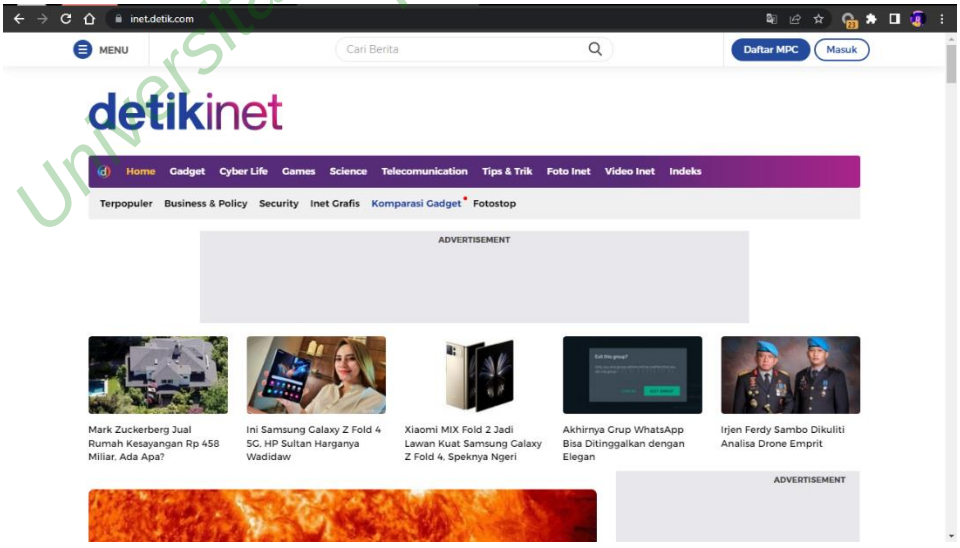
Gambar 4.21 *Client* yang Paling Banyak Melakukan Akses *Internet*

Kemudian pada Gambar 4.21 memperlihatkan daftar *client* yang paling banyak melakukan akses pada jaringan FTTI Universitas Jenderal Achmad Yani Yogyakarta. Daftar tersebut menunjukkan IP perangkat *client* yang dapat berupa PC ataupun *smartphone*. Dalam hal ini, seluruh informasi IP *client* akan dicatat oleh Pi-Hole sehingga dapat dilakukan pengumpulan data yang kemudian disajikan pada *dashboard* dalam bentuk tabel. Selain itu, penelitian melakukan pengujian akses jaringan menggunakan perangkat PC dan perangkat mobile yaitu smartphone sebagai *client*. Sebagai contoh, penelitian mengakses *domain* inet.detik.com menggunakan jaringan yang tidak menggunakan Pi-Hole. Maka, situs akan tampak seperti pada Gambar 4.22.



Gambar 4.22 Situs inet.detik.com yang Mengandung *Malvertising*

Banyaknya iklan daring yang muncul pada situs Kumparan mengganggu kenyamanan akses yang dilakukan *client*. Maka dari itu, penulis melakukan pengujian dengan mengganti jaringan menggunakan jaringan FTTI Universitas Jenderal Achmad Yani Yogyakarta. Antarmuka pada situs Detik menjadi seperti pada Gambar 4.23 di bawah ini. Gambar menunjukkan bahwa iklan daring yang muncul pada situs Detik berkurang, bahkan proses akses menjadi lebih cepat karena berkurangnya serangan *malvertising*.



Gambar 4.23 Situs inet.detik.com Terbebas dari *Malvertising*

Selain pengujian akses jaringan melalui PC, penelitian juga melakukan pengujian dengan akses jaringan melalui *smartphone*. Hal ini bertujuan untuk memastikan apakah Pi-Hole hanya bekerja pada perangkat PC saja. Penelitian menggunakan situs parapuan.co sebagai objek perbandingan.

Gambar 4.24 Situs parapuan.co yang Mengandung *Malvertising*

Pada Gambar 4.24 menunjukkan bahwa beberapa iklan daring muncul di tengah artikel. Hal ini terjadi sebelum perangkat *smartphone* terpasang dengan jaringan FTTI Universias Jenderal Achmad Yani Yogyakarta. Demikian iklan daring akan terus bermunculan dan mengganggu kenyamanan *client*. Penelitian melakukan percobaan pada perangkat *smartphone* dengan menghubungkan ke

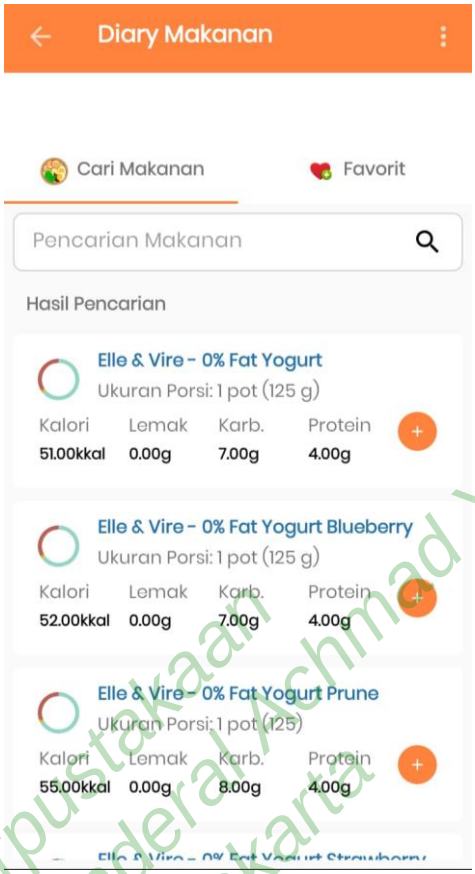
jaringan FTTI Universitas Jenderal Achmad Yani Yogyakarta dan mengakses ulang situs tersebut.

Gambar 4.25 Situs parapuan.co Terbebas dari *Malvertising*

Berikut pada Gambar 4.25 menunjukkan situs Parapuan telah bersih dari *malvertising* setelah perangkat terhubung ke jaringan FTTI Universitas Jenderal Achmad Yani Yogyakarta. Berdasarkan hasil pengujian yang dilakukan, membuktikan bahwa sistem Pi-Hole berfungsi dengan baik dan berhasil mengurangi risiko serangan *malvertising*. Demikian dapat dibuktikan dengan terblokirnya iklan daring yang muncul pada perangkat *client* serta pengaksesan ke situs *web* berjalan lebih cepat dibandingkan sebelumnya.

Gambar 4.26 Aplikasi HitungKalori yang Mengandung *Malvertising*

Sementara itu, penelitian melakukan pengujian *client* pada aplikasi berbasis *web*. Seperti yang ditunjukkan pada Gambar 4.26, bahwa iklan daring tetap muncul ketika jaringan tidak terhubung ke jaringan FTTI Universitas Jenderal Achmad Yani Yogyakarta. Dengan demikian, penelitian menguji aplikasi tersebut dengan menghubungkan *smartphone client* dengan jaringan FTTI Universitas Jenderal Achmad Yani Yogyakarta sehingga mendapatkan hasil sebagai berikut.



Gambar 4.27 Aplikasi HitungKalori Terbebas dari *Malvertising*

Berdasarkan hasil yang ditunjukkan pada Gambar 4.27, aplikasi HitungKalori yang sebelumnya mengandung *malvertising* telah bersih setelah perangkat *smartphone* terhubung dengan jaringan FTTI Universitas Jenderal Achmad Yani Yogyakarta. Demikian Pi-Hole dinyatakan dapat mengurangi serangan *malvertising* pada aplikasi *mobile.*

Pengujian selanjutnya dilakukan dengan analisis perbandingan terhadap situs *web* yang terindikasi mengandung *malvertising*. Pada penelitian ini diambil 20 objek untuk analisis. Pengujian ini dilakukan dengan membuka masing-masing 10 situs berbeda untuk perangkat PC dan *smartphone* menggunakan *web browser* Google Chrome. Selain itu, pengujian berlanjut dengan pengaksesan 2 aplikasi *mobile* melalui *smartphone.* Perangkat mengakses situs *web* dengan dua situasi, yaitu pada saat jaringan FTTI Universitas Jenderal Achmad Yani Yogyakarta belum diimplementasikan dengan Pi-Hole dan pengaksesan saat jaringan FTTI

Universitas Jenderal Achmad Yani Yogyakarta setelah diimplementasikan dengan Pi-Hole. Kemudian, dilakukan penghitungan iklan daring yang muncul pada situs *web* tersebut. Maka dari itu, hasil pengujian dapat dilihat pada Tabel 4.1 dan Tabel 4.2.

Tabel 4.1 Hasil Pengujian *Client* pada PC

No	Situs	Jumlah *Malvertising*	
		Sebelum	Sesudah
1.	cekresi.com	3	0
2.	id.geekmarkt.com	7	0
3.	detik.com	12	0
4.	kapanlagi.com	6	0
5.	freepik.com	3	0
6.	kumparan.com	8	0
7.	medcom.id	8	0
8.	sinonimkata.com	3	0
9.	editpad.org	4	0
10.	duplichecker.com	4	0

Berdasarkan tabel yang ditampilkan pada Tabel 4.1, dapat diketahui bahwa situs yang diakses pada PC dengan jaringan FTTI Universitas Jenderal Achmad Yani Yogyakarta sebelum diintegrasikan dengan Pi-Hole menunjukkan perbedaan hasil pada jumlah iklan daring yang muncul. Dalam kondisi Sebelum, dapat diartikan bahwa ISP tidak menjalankan proses pemblokiran secara otomatis terhadap situs *web* yang mengandung *malvertising*. Sedangkan, pada kondisi Sesudah menunjukkan bahwa situs tidak memunculkan iklan daring sama sekali, kemudian *slot* iklan daring hanya terisi dengan ikon yang menunjukkan halaman kosong atau *empty page*.

Pengujian selanjutnya dilakukan dengan melakukan pengujian *client* pada perangkat *smartphone*. Tahapan yang dilakukan sama halnya dengan pengujian *client* dengan perangkat PC. Maka dari itu, hasil pengujian *client* dengan perangkat *smartphone* adalah sebagai berikut.

Tabel 4.2 Hasil Pengujian *Client* pada Perangkat *Smartphone*

No	Situs	Jumlah *Malvertising*	
		Sebelum	Sesudah
1.	kompas.com	11	0
2.	liputan6.com	7	0
3.	m.bisnis.com	10	0
4.	jabarekspres.com	10	0
5.	hops.id	8	0
6.	m.vidio.com	1	0
7.	thesimsresource.com	2	0
8.	picrew.me	4	0
9.	modthesims.info	3	0
10.	loop.co.id	9	0

Tabel 4.2 menggambarkan hasil pengujian yang dilakukan pada perangkat *smartphone*. Hasil pengujian menunjukkan bahwa ketika perangkat *smartphone* pun tidak luput dari serangan *malvertising*. Namun, ketika perangkat *smartphone* mengakses situs *web* tersebut kembali pada saat jaringan FTTI Universitas Jenderal Achmad Yani Yogyakarta telah diintegrasikan dengan Pi-Hole, hasil menunjukkan bahwa sistem pertahanan juga berfungsi dengan baik pada perangkat *smartphone*.

Pengujian selanjutnya dilakukan pada aplikasi berbasis *web*. Pada dasarnya, *malvertising* tidak hanya terjadi pada situs *web* saja, tetapi juga bisa terjadi dalam aplikasi *mobile*. Pada penelitian, dilakukan pengujian pada aplikasi mobile yaitu HitungKalori dan Background Eraser, sehingga didapatkan hasil seperti pada Tabel 4.3.

Tabel 4.3 Pengujian *Client* pada Aplikasi *Mobile*

No	Aplikasi	Jumlah Malvertising	
		Sebelum	Sesudah
1.	HitungKalori	1	0
2.	Background Eraser	2	0

Hasil pengujian menunjukkan bahwa kedua aplikasi pada awalnya memunculkan dua iklan daring pada saat aplikasi dijalankan. Tetapi pada saat aplikasi tersebut dijalankan pada perangkat *smartphone* yang terhubung ke jaringan FTTI Universitas Jenderal Achmad Yani Yogyakarta, dimana telah diintegrasikan dengan Pi-Hole, menunjukkan hasil nol untuk jumlah iklan daring yang muncul pada aplikasi tersebut.