BAB 4
HASIL PENELITIAN

4.1 RINGKASAN HASIL PENELITIAN

Metode *footprinting* merupakan metode yang di gunakan untuk pengumpulan data anomali jaringan. Terdapat 5 tahapan metode *footprinting* yaitu *Identifikasi* dan *prepare*, *port mirroring*, *setup configurasi*, analisis, dan pelaporan hasil.

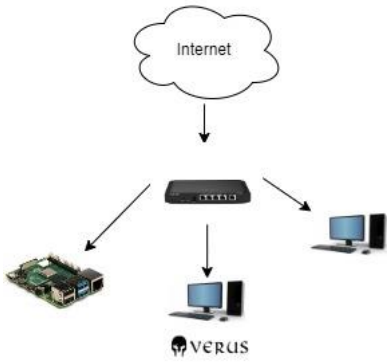
4.1.1 Identifikasi dan Prepare.

Mempersiapkan alat dan rancangan bangun ruang jaringan yang akan diuji kemanan dan kelayakan jaringan yang berada ada kampus Universitas Jend Ahmad yani Yogyakarta, mempersiapkan alat berupa laptop, satu buah *mikrotic* RB941-2^{nd,}, satu buah *memory card 16GB,* kabel rj45 2 buah, satu buah *rasbery pi 3b+*, mengunduh aplikasi *Verus coin* pada leptop atau *pc* dan *Verus miner 9000 smartphone android*, melakuakn konfigurasi *mikrotic* dan *instalasi* aplikasi *Maltrail malicious traffic detection system* pada *memerory rasbery pi 3b+*, pengkoneksian jaringan pada *mikrotic dan Rasbery pi 3b+*.

4.2 PORT MIRORRING.

Port mirorring adalah teknik dalam jaringan komputer yang digunakan untuk memonitor lalu lintas jaringan yang berada dalam jangkauan satu atau lebih *port* pada sebuah *switch*. Dalam hal ini, lalu lintas jaringan yang disalin untuk tujuan pemantauan penggunaan jaringan melewati proses *port* itu sendiri, selain digunakan untuk pemantauan, juga untuk pemecahan masalah jaringan, analisis data jaringan, dan keamanan lalu lintas jaringan. Dengan menggunakan *port mirroring*, *admin* dapat memantau aktivitas jaringan secara *real time*, juga dapat menganalisis paket jaringan yang terhubung dengan *port mirroring* serta dapat mengidentifikasi jaringan secara aman *Setting* port *mirorring* menggunakan *RB 94-2nd proxy* untuk *monitoring* dan *copy* tanpa merubah data asli, *setting* kabel biru pada *ether* 1 digunakan untuk *network source* agar terhubung ke *internet*, kabel kuning pada *drive plug pada ether* 2 digunakan untuk membagi sinyal antara, kabel abu-abu *Ether* 3 digunakan untuk sensor *Rasbery Pi 3b+* untuk memantau jaringan yang saat ini berjalan di *Ether 2*, *port mirroring* melalui *Ether* 3 digunakan untuk replikasi data melalui *Ether* 2 dengan masuk ke antarmuka *web Maltrail* dapat melihat *port*

mirroring berjalan dan dikonfigurasi, setelah berhasil mengonfigurasi *proxy*, pencerminan *port* dapat berjalan di *web nanopool.org*(Suharyanto & Gopama, 2019).



Gambar 4. 1 *port mirroing.*

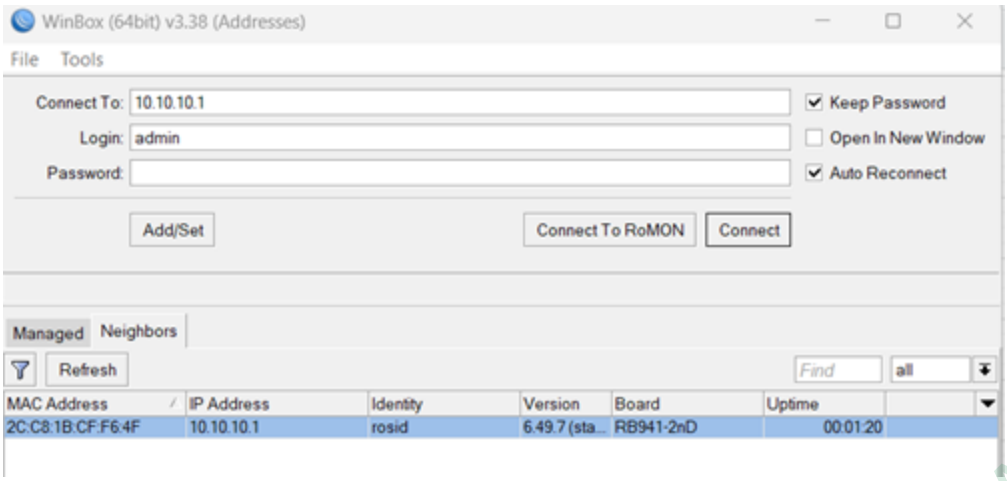
4.3 CONFIGURATION.

Konfigurasi merujuk pada sernagkaian pengaturan yang diterapkan pada system, perangkat, atau aplikasi untuk mengatur fungsional, perilaku, dan karakteristik tertentu sesuai dengan kebutuhan atau preferensi. Dalam konteks teknologi informasi dan jaringan, konfigurasi sering melibatkan penyesuaian parameter dan opsi yang memungkinkan system atau perangkat beroperasi degan cara yang diinginka, berikut adalah penjelasan tentang konfigurasi:

4.3.1 Setup mikrotik

Konfigurasi dasar dalam jaringan adalah *Setup* Mikrotik, perlu di perhatikan perangkat mikrotik membutuhkan daya Listrik untuk menghidupkan dan jaringan *computer* yang sesuai. Dalam penelitian ini pengkoneksian mikrotik pada *router* yang sudah terkoneksi pada jaringan menggunakan kabel *ethernet* dan diperlukan manajemen *remote* agar dapat mengakses dam pengkonfigurasian berhasil menggunakan *Winbox*, penulis menggunakan alar *IP* dengan *static* menggunakan nomor 10.10.10.1/24 .

Beberapa hal yang harus di perhatikan dalam konfigurasi mikrotik pada *Winbox* yaitu memasukan nama pengguna (*userame*) dan kata sandi (*password)* untuk *Autotokensi*, ketentuan ini sebenarnya sudah ada dari bawaan *setup* mikrotik sehingga perlu di ubah supaya mengamankan rangkaian *setup* mikrotik dari serangan kejahatan.



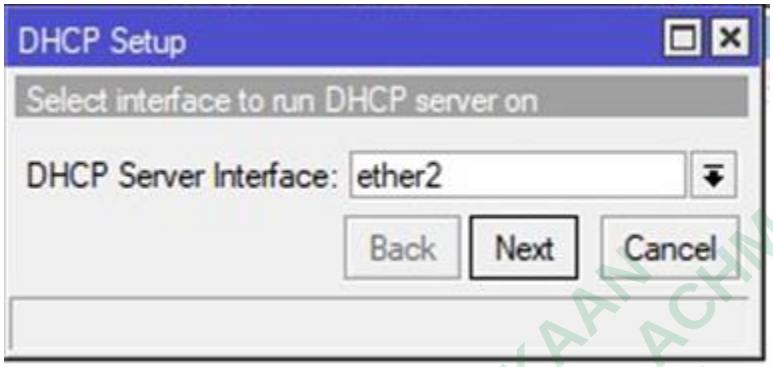
Gambar 4. 2 masuk *winbox* dengan *Mac address default.*

Tampilan di atas merupakan *userame* dan *password* yang telah diganti, setelah dilakukan penggantian *Identity* pada *Wibox* untuk meminimalisir kesalahan daalam k*onfigurasi*, kemudian dilakukan *konfigurasi* berikut:

1. Konfigurasi *ip address* pada antar muka *Ethernet* pada *Microtic* yaitu *ethernet* 1 sebagai sumber internet, *Ethernet* 2 sebagai *DHCP Server*, *ethernet* 3 sebagai *DHCP Client.*
2. Mikrotik digunakan sebagai *Router internet* dengan menggunakan 3 kabel, *Ethernet* biru sebagai internet tertancap di *Ethernet* 1,*Eternet* kuning ssebagai distribusi tertancap di *Ethernet* 2, *Ethernet* abu-abu sebaga i*port mirroring data* tertancap pada *ethernet*
3. Mikrotik mengaktifkan fitur *DHCP (Dynamic Host Configuration Protocol) server. DHCP* ini digunakan untuk memudahkan dalam pendistribuan *IP* secara otomatis keperangkat lain.

4.3.2 *Setting DHCP Setup*

Setelah melakukan *log in* pada mikrotk maka langkah awal adalah *setting DHCP SETUP* dengan klik 2 kali maka jendela dhcp akan terbuka, *Port Ether2* digunakan sebagai jembatan yang menghubungkan perangkat dijaringan lokal, dengan mengkonfigurasi *DHCP* di *Ether 2*, perangkat mendapatkan alamat seacara otomatis mendapatkan *ip*. Sehingga dapat berkomunikasi dan berinteraksi secara mudah. Seperti gambar 4.3

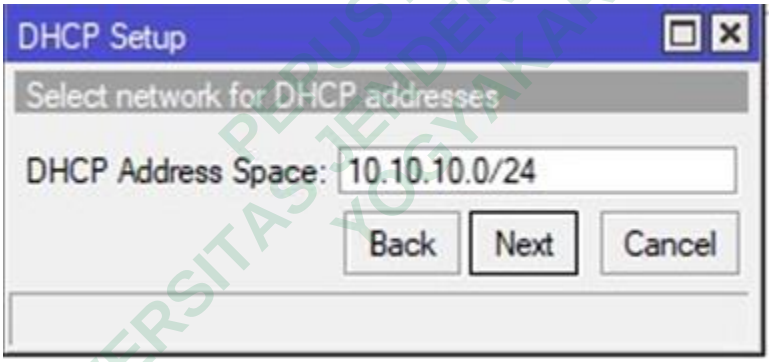


Gambar 4. 3 *Dhcp server interface Ether 2*

.

4.3.3 Konfigrasi *DHCP SETUP*

Langkah selanjutnya setelah menyetel alamat *ip* ke 10.10.10.0/24, Langkah ini ditujukan untuk pemrosesan konfigurasi yang lebih mendalam mengenai alamat *IP* yang akan digunakan oleh *server DHCP* untuk menetapkan rentang alamat *IP* pada perangkat yang terhubung ke jaringan. Klik "Berikutnya" dengan alamat *ip* 10.10.10.0/24, Sebelum mengklik "Berikutnya", dapat melihat ringkasan konfigurasi yang telah ditentukan sebelumnya. Ini termasuk alamat *ip* 10.10.10.0/24 yang akan digunakan di jaringan.Beberapa konfigurasi *DHCP* juga memungkinkan alamat *IP* statis permanen ditetapkan ke perangkat tertentu di jaringan. Setelah pengaturan rentang alamat *IP* dan waktu selesai, biasanya ada langkah verifikasi atau ringkasan konfigurasi sebelum benar-benar melakukan perubahan. Kemudian pengguna dapat mengklik tombol "Terapkan" atau "OK" untuk melanjutkan , dapat di lihat pada gambar 4.4

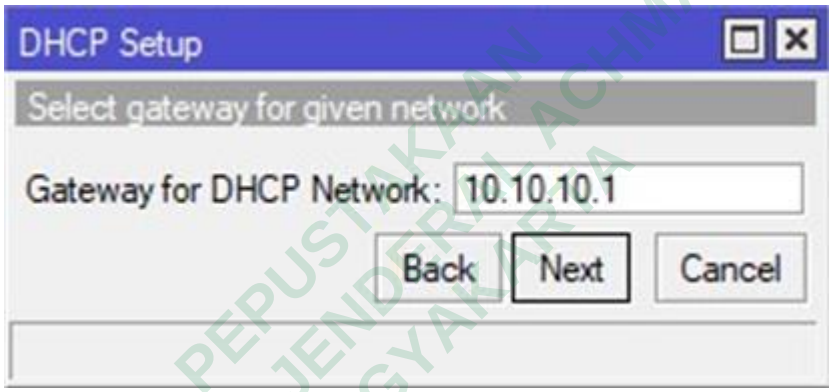


Gambar 4. 4 konfigurasi *address space.*

4.3.4 DHCP Network

Setelah berhasil melakukan konfigurasi alamat *IP address*, langkah selanjutnya yang dilakukan adalah melakukan konfigurasi *DHCP* dengan pengaturan alamat *IP* 10.0.10.1. Proses ini termasuk mengonfigurasi layanan *Dynamic Host Configuration Protocol (DHCP*) untuk secara otomatis menetapkan alamat *IP* ke perangkat di jaringan yang terhubung. Alamat *IP* yang digunakan adalah 10.0.10.1, penulis mengakses atau mengkonfigurasi server untuk bertindak sebagai layanan *DHCP* di jaringan. Ini bisa berupa *router*, *server* khusus, atau perangkat lain

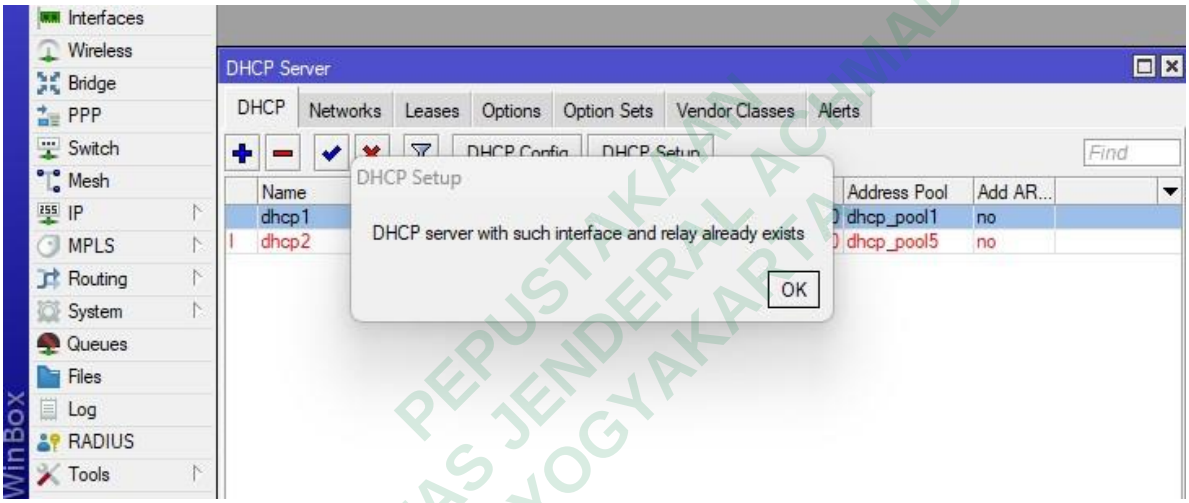
yang mampu menyediakan layanan *DHCP*. Layanan *DHCP* diaktifkan di *server* dengan mengaktifkan fitur *DHCP*. Hal ini memungkinkan server untuk mendistribusikan alamat *IP* dan parameter jaringan lainnya ke perangkat jaringan. Penulis menentukan rentang alamat *IP* yang diberikan oleh layanan *DHCP*. Dalam hal ini, alamat *IP* 10.0.10.1, Selain alamat *IP*, penulis juga dapat mengkonfigurasi pilihan lain di layanan *DHCP*, seperti alamat server *DNS*, *gateway default*, *subnet mask*, dan *konfigurasi* tambahan sesuai kebutuhan. Pengaturan ini perangkat dijaringan secara otomatis mendapatkan alamat *IP* 10.0.10.1 dan konfigurasi jaringan lainnya, sehingga mengurangi kerumitan konfigurasi manual. Layanan *DHCP* mengelola alokasi alamat *IP* secara dinamis, sehingga mengurangi risiko potensi konflik alamat *IP*. Jika perubahan pada pengaturan jaringan atau alamat *IP* perlu dilakukan, perubahan ini dapat dilakukan di *server DHCP* pusat. Dapat di lihat pada gambar 4.5



Gambar 4. 5 konfigurasi *DHCP Network.*

4.3.5 Tampilan *DHCP Server*

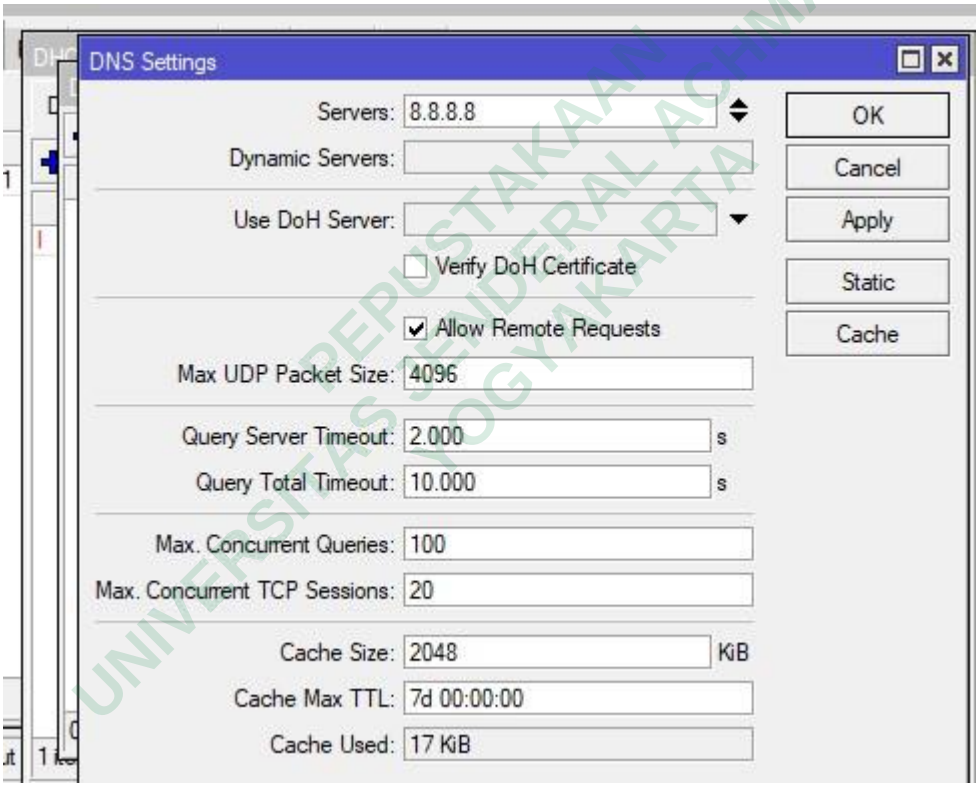
Dengan konfigurasi ini, perangkat yang terhubung secara otomatis memperoleh alamat *IP*, *subnet mask, port*, dan informasi jaringan lainnya. Di layar awal, dapat melihat status layanan *DHCP* yang diaktifkan. Status ini menunjukkan apakah layanan *DHCP* berfungsi dengan baik. Biasanya status ini akan menandakan layanan sudah aktif atau telah selesai di buat.dapat di lihat Pada gambar 4.6



Gambar 4. 6 *DHCP setup* sukses.

4.3.6 DNS Server

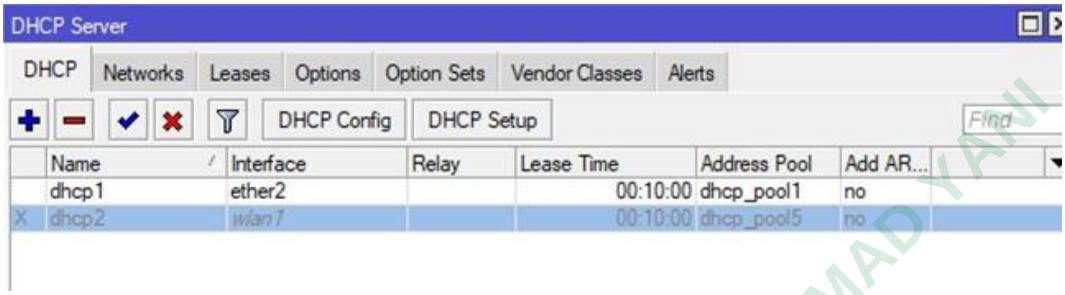
Setelah berhasil melakukan konfigurasi *DHCP*, langkah selanjutnya adalah melakukan konfigurasi *DNS server*. Konfigurasikan Server *DNS* dengan *DNS* 8.8.8.8 dan opsi Izinkan Kueri Jarak Jauh atau *allow remore request*. Setelah mengkonfigurasi *DHCP*, masuk ke pengaturan server *DNS*. Hal ini dapat dilakukan melalui antarmuka administrasi perangkat atau server bertindak sebagai *server DNS*. Mengkonfigurasi *server DNS* dengan menentukan alamat *DNS*. *server DNS* dikonfigurasi dengan alamat *DNS* 8.8.8.8, yang merupakan *server DNS* publik yang dioperasikan oleh *Google*. Memilih opsi ini akan mengaktifkan fitur yang menerima pertanyaan dari jaringan eksternal atau perangkat yang terhubung dari luar jaringan lokal. Setelah mengonfigurasi alamat *DNS* dan memilih opsi Izinkan permintaan jarak jauh, Anda akan menyimpan dan menerapkan perubahan. Dapat di lihat pada gambar 4.7



Gambar 4. 7 *DNS Server*.

4.3.7 Tampilan *DHC Server*

Setelah dibuat dan dikonfigurasi menggunakan fitur *DHCP* untuk secara otomatis mengkonfigurasi jaringan dan konektivitas yang efisien. Di layar awal, akan melihat status layanan *DHCP*. Status ini akan menunjukkan apakah layanan *DHCP* diaktifkan dan berfungsi dengan baik. Biasanya status ini akan menandakan layanan sedang aktif , dapat dilihat pada gambar 4.



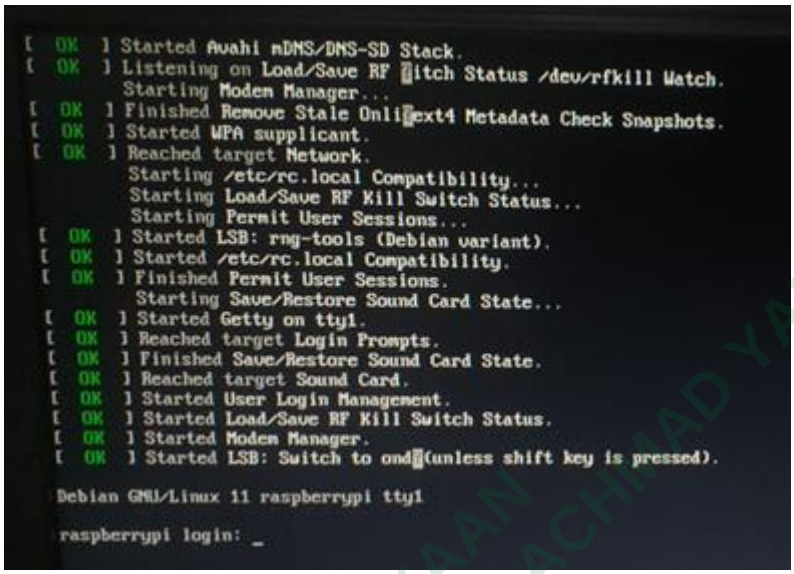
Gambar 4. 8 Tampilan *DHCP Server.*

Berdasarkan proses kolaborasi pengumpulan data penelitian ini penulis menggunakan metode *Mirorring data* untuk pendeteksi *maltrail* pada aplikasi *verus mining data traffic,* di mulai dengan menginstal *OS raspberry pi* dan menginstal *maltrail* pada *raspbery pi 3b+* serta penyettingan *mikrotic* untuk jaringan data yang di gunakan dalam pemantauan jaringan, dimulai dengan dilakuakan proses pengoneksian *raspbery pi 3b+* yang sudah terinstal aplikasi *Maltrail* dengan *setup* dan *convigurasi* mikrotik dasar menggunakan *WinBox*. Kemudian dilakukan analisis hasil dari penelitian dengan melakukan pengambilan *endpoint* data pada *verus mining.*

4.3.8 Installasi *Raspberry*

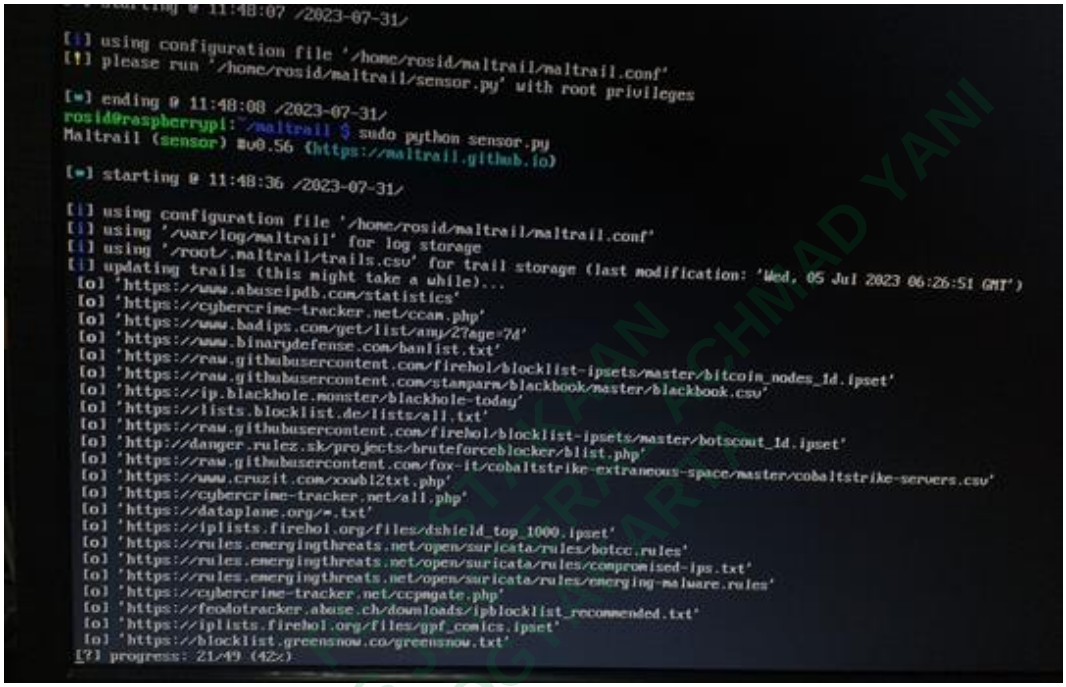
Dalam penelitian ini penulis melakukan instalasi *raspberry pi* di mulai dengan instalasi *imager*, pengunduhan melalui situs resmi *raspberry pi* dan *instasli* di lakukan di pc atau leptop penulis dengan mengekstak pada kartu *micro sd card* sebagai media penyimpanan utama, Kartu *Sd card* yang digunakan penulsi adalah *sandisk 16Gb* sebagai media penyimapana *Maltrail*, ada beberapa model *raspberry pi* tetapi penulis menggunakan *raspberry pi* model 3B+, *raspbery pi* membutuhkan system operasi agar dapat di gunakan, system operasi yang digunakan oleh penulis adalah *Ubuntu* 20,4. Di butuhkan juga *Hardware* seperti monitor, *keyboard, mouse,* sebagai komponen pendukung yang menghubungkan ke *Raspberry pi* ,setelah r*aspberry Booting*

akan di arahkan ke antarmuka konfigurasi untuk mengatur pengaturan jaringan, Bahasa, waktu, DLL langkah langkah instalasi sebagai berikut. Tampilan setelah *booting* akan diminta *Login* serta *password* untuk memulai proses *linuk*



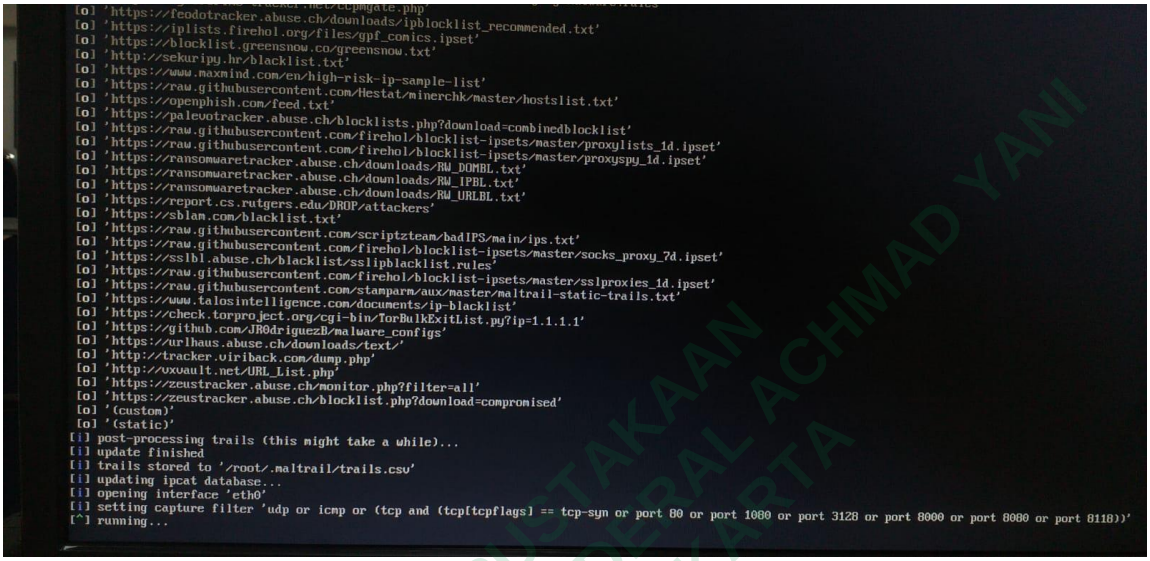
Gambar 4. 9 tampilan awal masuk

Setelah berhasil masuk ke tampilan awal, langkah selanjutnya adalah mengupdate *Python sensor.pi* dengan perintah *sudo update*, proses ini melibatkan instalasi dan pembaharuan terbaru yang memerlukan beberapa waktu untuk diselesaikan. Selama proses ini berbagai komponen dan *dedepensi* akan di *install* dan di perbaharui untuk memastikan ketersediaan yang terbaru dan optimal dalam pembaharuan *python sensor.pi*. Seperti gambar 4.10



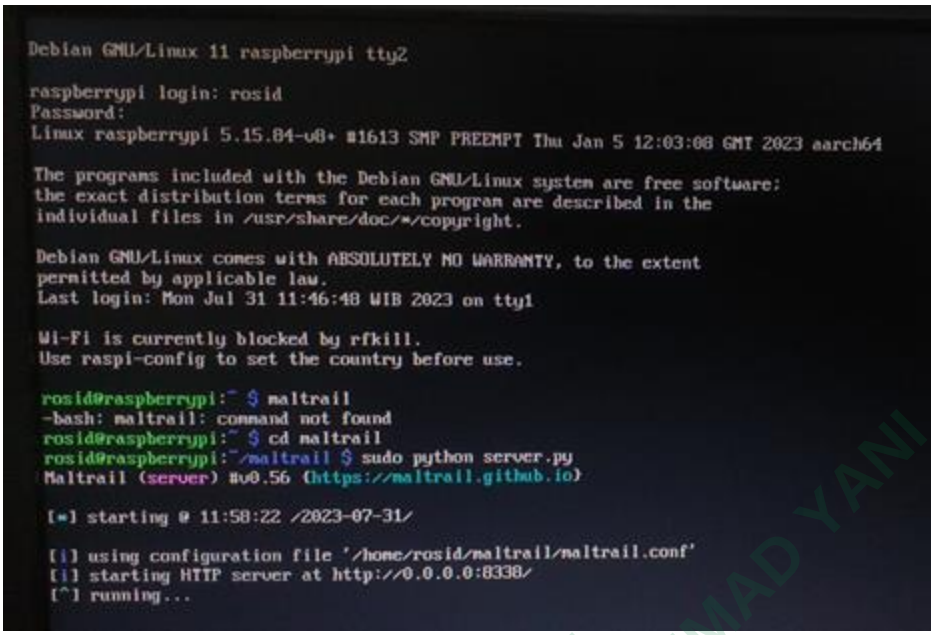
Gambar 4. 10 *sensor update*

Setelah sensor berhasil diperbaharui dan ter-*update* dengan langkah –langkah sebelumnya, langkah selanjutnya adalah mengetik perintah *sudo python.py*, tindakan ini akan menginisiasi eksekusi dari *file python.py*, yang telah di perbaharui. *File* ini kemungkinan memiliki peran dalam proses lebih lanjut yang berkaitan dengan fungsionalitas sensor atau alat yang digunakan. Proses ini akan dimulai setelah perintah dieksekusi. Seperti gambar 4.11



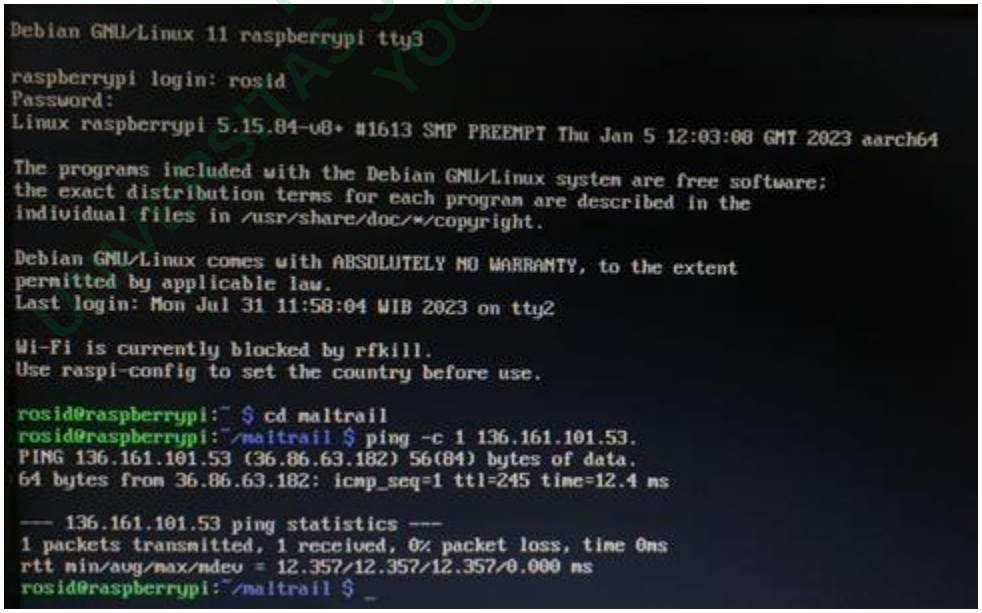
Gambar 4. 11 *sensor* berjalan

Langkah berikutnya adalah membuka *tab* baru atau *tab* kedua pada antarmuka yang digunakan. Hal ini bertujuan untuk menjalankan *server maltrail.* Dengan mengetikkan perintah *sudo python server.py*. Proses ini akan menginisiasi ekseskusi dari *file server.py*, yang merupakan bagian dari *server maltrail*. Tindakan ini dibuktikan dengan gambar 4.12 yang memberikan visualisasi tentang langkah ini dalam praktik



Gambar 4. 12 *server.py* sudah berjalan

Pengujian *maltrail malicious traffic detection system* sudah terhubung dengan internet melalui langkah pengujian berupa *ping ip* –c 1 136.161.101.53, hasil dari pengujian menunjukan bahwa 1 paket berhasil di kirim, 1 paket diterima, 0% hilang.



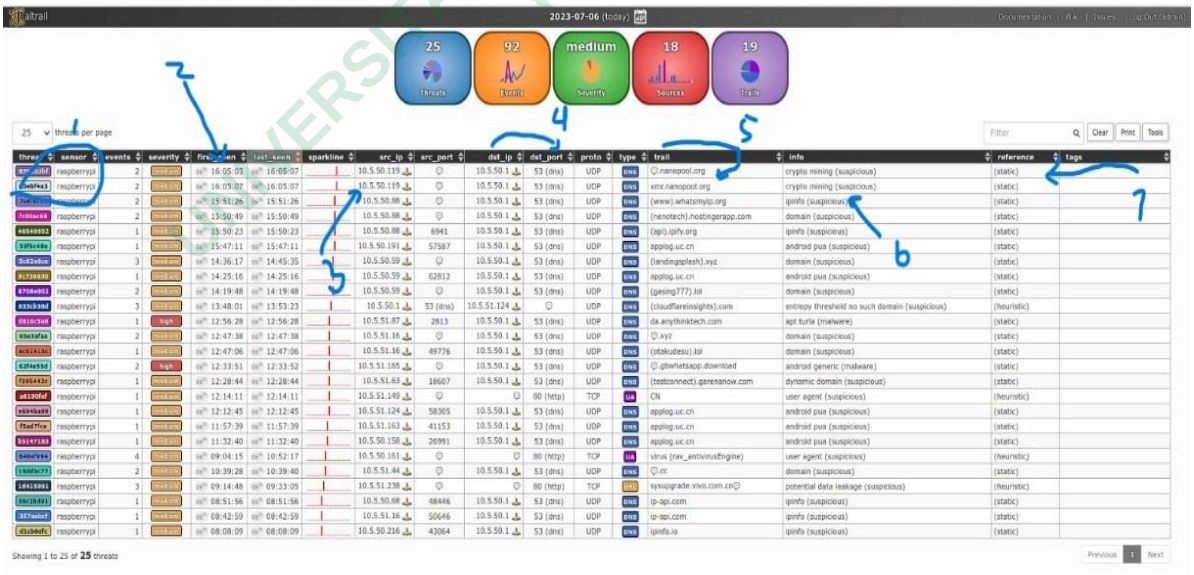
Gambar 4. 13 uji coba aplikasi *maltrail*

4.4HASIL

4.1 PELAPORAN HASIL ANALISIS

Untuk menjalankan aplikasi maltrail dimulai dengan menyusup ke proses lalu lintas berbahaya di jaringan, melepaskan lalu lintas jaringan masuk dan keluar jaringan tanpa mengetahui data yang diterima dan dikirim, dan mengkonfigurasi jaringan agar terhubung ke *proxy* di jaringan yang sama dengan jaringan *raspberry*. jaringan ter-segmentasi dilacak untuk diekstrak atau menyalin data.

Saat menerapkan *port mirorring*, analisis lalu lintas berhasil dilakukan dalam konteks penambangan mata uang kripto, mengalihkan salinan lalu lintas jaringan dari *port* terkait penambangan ke *port* pemantauan, dalam analis Ini mengidentifikasi koneksi yang terkait dengan *port* tertentu yang digunakan dalam protokol penambangan seperti *port* 3333 untuk penambangan. Berkat pemantauan aktif ini tidak mendeteksi lonjakan koneksi ke *port* yang mencerminkan aktivitas penambangan, juga tidak meningkatkan *bandwidth* yang digunakan saat menambang, akses analisis jenis lalu lintas melewati *port* ini, mengidentifikasi perangkat tambahan terkait penambangan, seperti penambang. *rig* dan *server* penambangan, analisis ini digunakan untuk mengklasifikasikan kemungkinan serangan atau aktivitas mencurigakan.



Gambar 4. 1 Hasil *monitoring*

Berkat aplikasi pencerminan *port miroring* yang terpasang pada sensor *Raspberry Pi* [1], telah berhasil mengumpulkan data terkait aktivitas mata uang kripto di jaringan, sensor *Raspberry Pi* mendapatkan salinan lalu lintas, yang dapat diakses. Mengekspor data yang akurat dan mendalam tentang aktivitas penambangan *cryptocurrency* di jaringan yang berjalan pada satu *segmen* jaringan, data ini membantu mendeteksi potensi masalah keamanan jaringan, memantau kinerja jaringan, dan merespons dengan cepat terhadap kelainan yang terdeteksi. Pendekatan ini memberikan pandangan *komprehensif* tentang aktivitas kripto di lingkungan, memberikan manfaat signifikan dalam tata kelola dan manajemen jaringan.

Pengumpulan data pertama dilakukan dengan metode pengujian aplikasi *Verus* yang terintegrasi dengan sistem deteksi *real-time* atau secara langsung, terhubung ke jaringan yang sama dengan sensor *Raspberry Pi 3B+*. Sensor ini berhasil mencatat rata-rata penggunaan jaringan dalam berbagai kondisi. Selama pengujian, sensor berhasil mengidentifikasi serangan terhadap penambangan *cryptocurrency* pada pukul 16.05.03 [2]. Proses verifikasi data berhasil diselesaikan pada pukul 16.05.07 [3], memastikan data akurat dan dapat diandalkan. Pengujian dan pemantauan jaringan dilakukan dalam *mode real-time,* dengan durasi perekaman *IP* 10.5.20.119 [4] hingga peralihan ke alamat *IP* 10.5.50.1 [4]. Data yang dikumpulkan selama periode ini mencerminkan pola lalu lintas dan penggunaan sumber daya jaringan.

Namun perlu diingat bahwa selama pemantauan pada *web nanopool.org* [5], terdapat *indeks* yang masih dicurigai sebagai sumber aktivitas mencurigakan. Indeks ini mengacu pada alamat *IP* 10.5.50.1. terus memantau secara dekat indikator ini, dengan tujuan untuk mengklarifikasi apakah ada kemungkinan aktivitas merugikan terkait penambangan *cryptocurrency*. Akuisisi data dilakukan dengan menguji aplikasi Verus yang terhubung ke sistem deteksi *real-time* dan sensor *Raspberry Pi 3B+*. Data yang dikumpulkan memberikan gambaran akurat tentang penggunaan jaringan dan serangan kripto yang terjadi pada pukul 16.05.03. Sensor memantau indeks mencurigakan[7] pada alamat *IP* 10.5.50.1 di *nanopool.org* [6] untuk memastikan keamanan jaringan yang optimal. Bisa di lihat pada monitoring pertama pada gambar 4.15 dibawah ini.

Gambar 4. 2 hasil *monitoring*

Pada tahap pengumpulan data selanjutnya, melakukan pengujian pada *website pool.supportxmr.com* menggunakan sensor *Raspberry Pi* [1]. Dalam pengujian ini memantau lalu lintas yang dihasilkan oleh *sensor Raspberry Pi*. Situs *web* dibuat untuk lebih memahami penambangan *cryptocurrency*. Hasil pengujian menunjukkan jaringan digunakan dalam kondisi aman, tidak ada tanda-tanda serangan mencurigakan.

Pengujian dilaksanakan pada pukul 08.03.19 [2] dan berhasil pada pukul 15.59.19 [2]. Pengujian difokuskan pada alamat *IP* 10.5.50.64 [3], dengan lalu lintas yang diamati dan dianalisis. Selama pengujian tidak terdeteksi aktivitas yang mengindikasikan serangan atau ancaman keamanan signifikan. Namun perlu dicatat bahwa pada periode selanjutnya, data pengujian menunjukkan indikator yang masih dianggap mencurigakan pada penambangan kripto. Informasi ini merujuk secara khusus ke alamat *IP* 10.5.50.1. menganalisis lebih lanjut indikator ini untuk memastikan bahwa tidak ada aktivitas berbahaya atau ilegal yang terkait dengan penambangan mata uang kripto.

pengujian yang dilakukan di *web pool.supportxmr.com* [4] melalui sensor *Raspberry Pi* menunjukkan bahwa penggunaan jaringan dibuat sepenuhnya aman dan tanpa tanda-tanda serangan. Namun keberadaan *indeks* yang diduga masih mencurigakan [5] pada alamat *IP* 10.5.50.1 tetap menjadi fokus *analisis* mendalam untuk menjaga integritas keamanan jaringan dan mencegah potensi risiko yang kemungkinan akan timbul.