

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 konfigurasi*, analisis, dan pelaporan hasil.

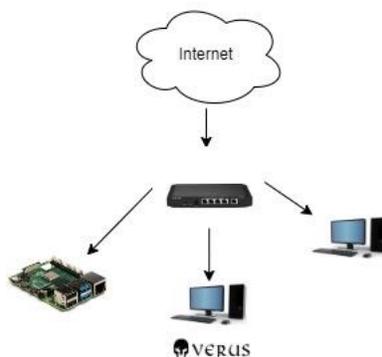
4.1.1 *Identifikasi dan Prepare.*

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

4.2 PORT MIRRORING.

Port mirroring 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 mirroring* 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.

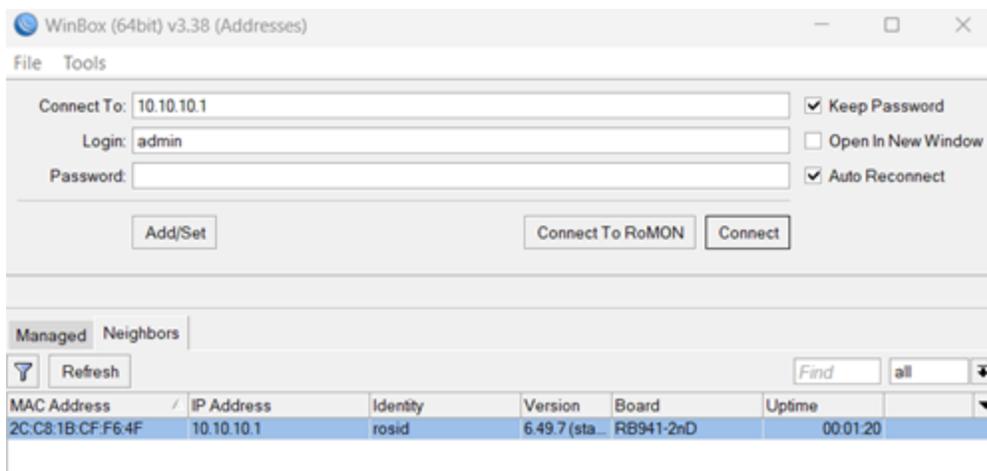
4.3 CONFIGURATION.

Konfigurasi merujuk pada serangkaian 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 dengan cara yang diinginkan, 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 dan 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 (*username*) 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.



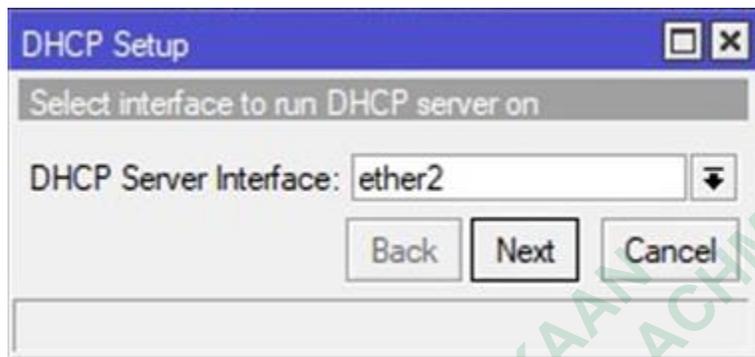
Gambar 4. 2 masuk winbox dengan Mac address default.

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

1. Konfigurasi *ip address* pada antar muka *Ethernet* pada *Mikrotik* 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*, *Ethernet* kuning sebagai distribusi tertancap di *Ethernet 2*, *Ethernet* abu-abu sebagai *iport 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

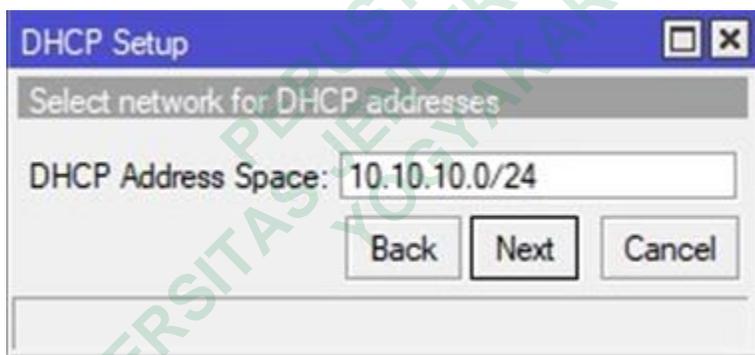
Setelah melakukan *log in* pada mikrotk maka langkah awal adalah *setting DHCP SETUP* dengan klik 2 kali maka jendela dhcp akan terbuka, *Port Ether2* digunakan sebagai jembatan yang menghubungkan perangkat di jaringan lokal, dengan mengkonfigurasi *DHCP* di *Ether 2*, perangkat mendapatkan alamat secara 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 Konfigurasi *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

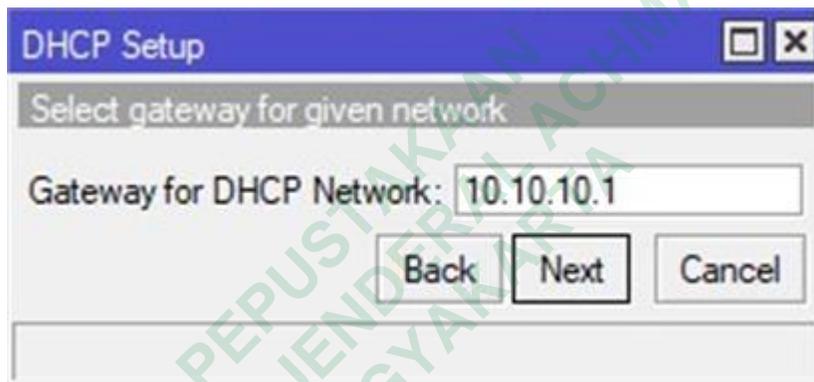


Gambar 4. 4 konfigurasi *address space*.

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

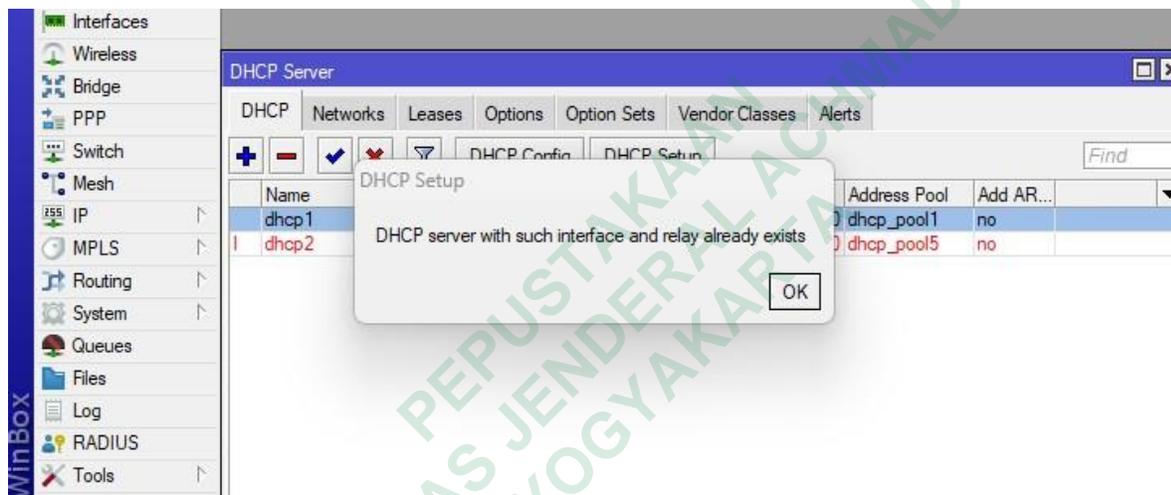
yang mampu menyediakan layanan *DHCP*. Layanan *DHCP* diaktifkan di *server* dengan mengaktifkan fitur *DHCP*. Hal ini memungkinkan server untuk mendistribusikan alamat *IP* dan parameter jaringan lainnya ke perangkat jaringan. Penulis menentukan rentang alamat *IP* yang diberikan oleh layanan *DHCP*. 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 di jaringan 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*

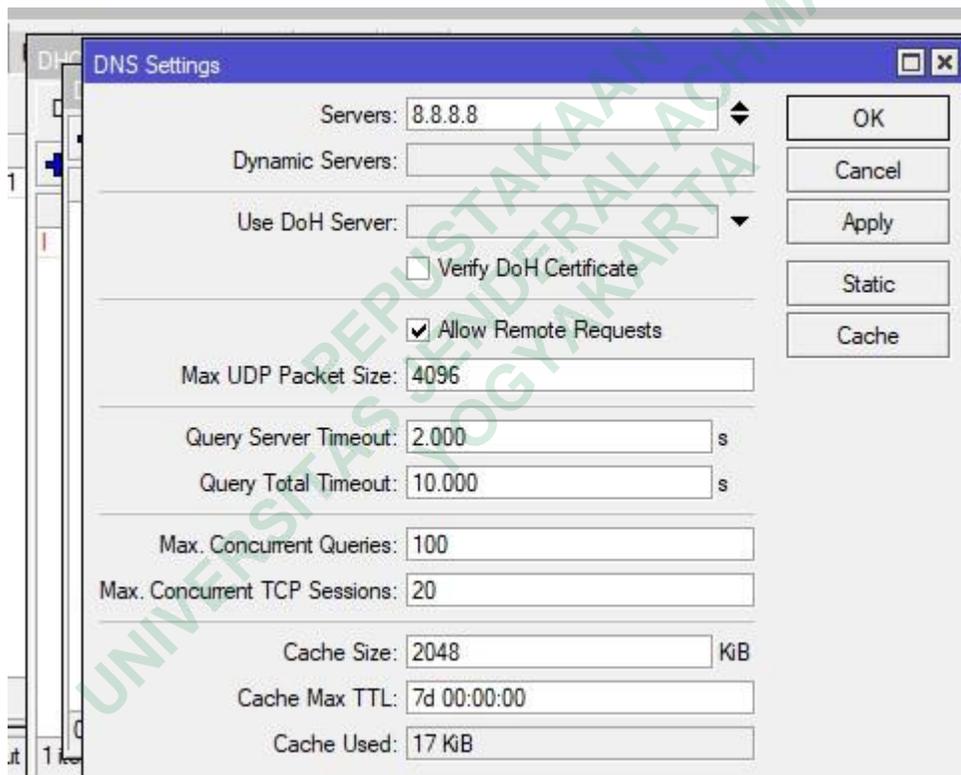
Dengan konfigurasi ini, perangkat yang terhubung secara otomatis memperoleh alamat *IP*, *subnet mask*, *port*, dan informasi jaringan lainnya. Di layar awal, dapat melihat status layanan *DHCP* yang diaktifkan. Status ini menunjukkan apakah layanan *DHCP* berfungsi dengan baik. Biasanya status ini akan 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 remote 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.

Name	Interface	Relay	Lease Time	Address Pool	Add AR...
dhcp1	ether2		00:10:00	dhcp_pool1	no
X dhcp2	wan7		00:10:00	dhcp_pool5	no

Gambar 4. 8 Tampilan *DHCP Server*.

Berdasarkan proses kolaborasi pengumpulan data penelitian ini penulis menggunakan metode *Mirroring data* untuk pendeteksi *maltrail* pada aplikasi *verus mining data traffic*, di mulai dengan menginstal *OS raspberry pi* dan menginstal *maltrail* pada *raspberry pi 3b+* serta penyettingan *mikrotic* untuk jaringan data yang di gunakan dalam pemantauan jaringan, dimulai dengan dilakukan proses pengoneksian *raspberry 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 Instalasi *Raspberrry*

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 mengekstack 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+, *raspberry 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 *Raspberrry pi*, setelah *raspberry 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 *linux*

```
[ OK ] Started Avahi mDNS/DNS-SD Stack.
[ OK ] Listening on Load/Save RF Kill Switch Status /dev/rfkill Watch.
      Starting Modem Manager...
[ OK ] Finished Remove Stale Onli[ext4 Metadata Check Snapshots.
[ OK ] Started WPA supplicant.
[ OK ] Reached target Network.
      Starting /etc/rc.local Compatibility...
      Starting Load/Save RF Kill Switch Status...
      Starting Permit User Sessions...
[ OK ] Started LSB: rng-tools (Debian variant).
[ OK ] Started /etc/rc.local Compatibility.
[ OK ] Finished Permit User Sessions.
      Starting Save/Restore Sound Card State...
[ OK ] Started Getty on tty1.
[ OK ] Reached target Login Prompts.
[ OK ] Finished Save/Restore Sound Card State.
[ OK ] Reached target Sound Card.
[ OK ] Started User Login Management.
[ OK ] Started Load/Save RF Kill Switch Status.
[ OK ] Started Modem Manager.
[ OK ] Started LSB: Switch to on[unless shift key is pressed).

Debian GNU/Linux 11 raspberrypi tty1
raspberrypi login: _
```

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

```

[!] starting @ 11:48:07 /2023-07-31/
[!] using configuration file '/home/rosid/maltrail/maltrail.conf'
[!] please run '/home/rosid/maltrail/sensor.py' with root privileges
[!] ending @ 11:48:08 /2023-07-31/
rosid@raspberrypi: ~$ sudo python sensor.py
Maltrail (sensor) v0.56 (https://maltrail.github.io)

[!] starting @ 11:48:36 /2023-07-31/
[!] using configuration file '/home/rosid/maltrail/maltrail.conf'
[!] using '/var/log/maltrail' for log storage
[!] using '/root/.maltrail/trails.csv' for trail storage (last modification: 'Wed, 05 Jul 2023 06:26:51 GMT')
[!] updating trails: (this might take a while)...
[!] 'https://www.abuseipdb.com/statistics'
[!] 'https://cybercrime-tracker.net/ccan.php'
[!] 'https://www.badips.com/get/list/any?2?age=7d'
[!] 'https://www.binarydefense.com/banlist.txt'
[!] 'https://raw.githubusercontent.com/firehol/blocklist-ipsets/master/bitcojn_nodes_id.ipset'
[!] 'https://raw.githubusercontent.com/stanpara/blackbook/master/blackbook.csv'
[!] 'https://ip.blackhole.nemster/blackhole-today'
[!] 'https://lists.blocklist.de/lists/all.txt'
[!] 'https://raw.githubusercontent.com/firehol/blocklist-ipsets/master/botscout_id.ipset'
[!] 'http://danger.rulez.sk/projects/bruteforceblocker/blist.php'
[!] 'https://raw.githubusercontent.com/fox-it/cobaltstrike-extraneous-space/master/cobaltstrike-servers.csv'
[!] 'https://www.cruzeit.com/xoubi2.txt.php'
[!] 'https://cybercrime-tracker.net/all.php'
[!] 'https://dataplane.org/*.txt'
[!] 'https://iplists.firehol.org/files/dshield_top_1000.ipset'
[!] 'https://rules.emergingthreats.net/open/suricata/rules/botcc.rules'
[!] 'https://rules.emergingthreats.net/open/suricata/rules/compromised-ips.txt'
[!] 'https://rules.emergingthreats.net/open/suricata/rules/emerging-malware.rules'
[!] 'https://cybercrime-tracker.net/ccpgate.php'
[!] 'https://feedotracker.abuse.ch/downloads/ipblocklist_recommended.txt'
[!] 'https://iplists.firehol.org/files/gpf_comics.ipset'
[!] 'https://blocklist.greensnow.co/greensnow.txt'
[?] progress: 21/49 (42%)

```

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

```

[0] https://feodotracker.abuse.ch/downloads/ipblocklist_recommended.txt'
[0] https://iplists.firehol.org/files/gpf_conics_ipset'
[0] https://blocklist.greensnow.co/greensnow.txt'
[0] http://securipg.hr/blacklist.txt'
[0] https://www.maxmind.com/en/high-risk-ip-sample-list'
[0] https://raw.githubusercontent.com/Hestat/minerchk/master/hostslist.txt'
[0] https://paleovotracker.abuse.ch/blocklists.php?download=combinedblocklist'
[0] https://raw.githubusercontent.com/firehol/blocklist-ipsets/master/proxyls_id_ipset'
[0] https://raw.githubusercontent.com/firehol/blocklist-ipsets/master/proxyls_id_ipset'
[0] https://ransomwaretracker.abuse.ch/downloads/RM_DOMPL.txt'
[0] https://ransomwaretracker.abuse.ch/downloads/RM_IPBL.txt'
[0] https://report.cs.rutgers.edu/DROP/attackers'
[0] https://sblam.com/blacklist.txt'
[0] https://raw.githubusercontent.com/scriptzteam/badIPS/main/ips.txt'
[0] https://raw.githubusercontent.com/firehol/blocklist-ipsets/master/socks_proxy_7d_ipset'
[0] https://sslbl.abuse.ch/blacklist/ssliblacklist.rules'
[0] https://raw.githubusercontent.com/firehol/blocklist-ipsets/master/sslproxies_id_ipset'
[0] https://raw.githubusercontent.com/stamparm/aux/master/maltrail-static-trails.txt'
[0] https://www.talosintelligence.com/documents/ip-blacklist'
[0] https://check.torproject.org/cgi-bin/TorBulkExitList.py?ip=1.1.1.1'
[0] https://github.com/00driguez0/malware_configs'
[0] https://urlhaus.abuse.ch/downloads/text/'
[0] http://tracker.viriback.com/dump.php'
[0] http://uxvault.net/URL_List.php'
[0] https://zeustracker.abuse.ch/monitor.php?filter=all'
[0] https://zeustracker.abuse.ch/blocklist.php?download=compromised'
[0] 'custom'
[0] 'static'
[1] post-processing trails (this might take a while)...
[1] update finished
[1] trails stored to '/root/.maltrail/trails.csv'
[1] updating ipcat database...
[1] opening interface 'eth0'
[1] setting capture filter 'udp or icmp or (tcp and (tcpflags == tcp-syn or port 80 or port 1000 or port 3128 or port 8000 or port 8080 or port 8118))'
[*] running...

```

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 eksekusi 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

```

Debian GNU/Linux 11 raspberrypi tty2
raspberrypi login: rosid
Password:
Linux raspberrypi 5.15.04-00+ #1613 SMP PREEMPT Thu Jan 5 12:03:08 GMT 2023 aarch64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Jul 31 11:46:48 WIB 2023 on tty1

Wi-Fi is currently blocked by rfkill.
Use raspi-config to set the country before use.

rosid@raspberrypi:~$ maltrail
-bash: maltrail: command not found
rosid@raspberrypi:~$ cd maltrail
rosid@raspberrypi:~/maltrail$ sudo python server.py
Maltrail (server) #v0.56 (https://maltrail.github.io)

[!] starting @ 11:58:22 /2023-07-31/
[!] using configuration file '/home/rosid/maltrail/maltrail.conf'
[!] starting HTTP server at http://0.0.0.0:8338/
[!] running...

```

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 menunjukkan bahwa 1 paket berhasil di kirim, 1 paket diterima, 0% hilang.

```

Debian GNU/Linux 11 raspberrypi tty3
raspberrypi login: rosid
Password:
Linux raspberrypi 5.15.04-00+ #1613 SMP PREEMPT Thu Jan 5 12:03:08 GMT 2023 aarch64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Jul 31 11:58:04 WIB 2023 on tty2

Wi-Fi is currently blocked by rfkill.
Use raspi-config to set the country before use.

rosid@raspberrypi:~$ cd maltrail
rosid@raspberrypi:~/maltrail$ ping -c 1 136.161.101.53.
PING 136.161.101.53 (36.86.63.182) 56(84) bytes of data:
64 bytes from 36.86.63.182: icmp_seq=1 ttl=245 time=12.4 ms

--- 136.161.101.53 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 12.357/12.357/12.357/0.000 ms
rosid@raspberrypi:~/maltrail$ _

```

Gambar 4. 13 uji coba aplikasi *maltrail*

4.4 HASIL

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 *raspberrypi*. jaringan ter-segmentasi dilacak untuk diekstrak atau menyalin data.

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

threat	source	events	severity	first	last	sparkline	src_ip	src_port	dst_ip	dst_port	proto	type	trail	info	reference	tags
0000000	raspberrypi	2	critical	16:05:33	16:05:07	10.5.50.119	10.5.50.1	53	dns	UDP	dns	Q	0xnetpost.org	crypto mining (suspicious)	(static)	
0000001	raspberrypi	2	critical	16:05:37	16:05:07	10.5.50.119	10.5.50.1	53	dns	UDP	dns	Q	ama-nanopool.org	crypto mining (suspicious)	(static)	
0000002	raspberrypi	2	critical	19:41:26	19:51:26	10.5.50.88	10.5.50.1	53	dns	UDP	dns	Q	(www) ahashtraylo.org	pinfns (suspicious)	(static)	
0000003	raspberrypi	2	critical	19:50:49	19:50:49	10.5.50.88	10.5.50.1	53	dns	UDP	dns	Q	(netotech) hostingrasp.com	domain (suspicious)	(static)	
0000004	raspberrypi	1	critical	19:50:23	19:50:23	10.5.50.88	4941	10.5.50.1	53	dns	UDP	dns	(api) safy.org	pinfns (suspicious)	(static)	
0000005	raspberrypi	1	critical	19:47:11	19:47:11	10.5.50.191	53987	10.5.50.1	53	dns	UDP	dns	appleq.uc.cn	android pua (suspicious)	(static)	
0000006	raspberrypi	3	critical	14:36:17	14:45:35	10.5.50.99	Q	10.5.50.1	53	dns	UDP	dns	(landing)push.vyz	domain (suspicious)	(static)	
0000007	raspberrypi	1	critical	14:25:16	14:25:16	10.5.50.99	62812	10.5.50.1	53	dns	UDP	dns	appleq.uc.cn	android pua (suspicious)	(static)	
0000008	raspberrypi	2	critical	14:19:48	14:19:48	10.5.50.99	Q	10.5.50.1	53	dns	UDP	dns	(gaming777).id	domain (suspicious)	(static)	
0000009	raspberrypi	3	critical	13:40:01	13:53:23	10.5.50.1	53	dns	UDP	UDP	dns	(cloud)flareinsights.com	entropy threshold no such domain (suspicious)	(heuristic)		
0000010	raspberrypi	1	high	12:56:28	12:56:28	10.5.51.87	2813	10.5.50.1	53	dns	UDP	dns	dx.anythinktech.com	api burle (malware)	(static)	
0000011	raspberrypi	2	critical	12:47:38	12:47:38	10.5.51.16	Q	10.5.50.1	53	dns	UDP	dns	Q.vyz	domain (suspicious)	(static)	
0000012	raspberrypi	1	high	12:47:06	12:47:06	10.5.51.16	49776	10.5.50.1	53	dns	UDP	dns	(otakuteku).id	domain (suspicious)	(static)	
0000013	raspberrypi	2	high	12:33:51	12:33:52	10.5.51.105	Q	10.5.50.1	53	dns	UDP	dns	Q.gwhatsapp.download	android generic (malware)	(static)	
0000014	raspberrypi	1	critical	12:28:44	12:28:44	10.5.51.63	18667	10.5.50.1	53	dns	UDP	dns	(testconnect) gareanow.com	dynamic domain (suspicious)	(static)	
0000015	raspberrypi	1	critical	12:14:11	12:14:11	10.5.51.149	Q	Q	80	http	TCP	dns	CN	user agent (suspicious)	(heuristic)	
0000016	raspberrypi	1	critical	12:12:45	12:12:45	10.5.51.124	58365	10.5.50.1	53	dns	UDP	dns	appleq.uc.cn	android pua (suspicious)	(static)	
0000017	raspberrypi	1	critical	11:57:39	11:57:39	10.5.51.163	41153	10.5.50.1	53	dns	UDP	dns	appleq.uc.cn	android pua (suspicious)	(static)	
0000018	raspberrypi	1	critical	11:32:40	11:32:40	10.5.50.158	26991	10.5.50.1	53	dns	UDP	dns	appleq.uc.cn	android pua (suspicious)	(static)	
0000019	raspberrypi	4	critical	09:04:15	10:52:17	10.5.50.101	Q	Q	80	http	TCP	dns	virus (zav_antivirus)gnje	user agent (suspicious)	(heuristic)	
0000020	raspberrypi	2	critical	10:39:28	10:39:40	10.5.51.44	Q	10.5.50.1	53	dns	UDP	dns	Q.nc	domain (suspicious)	(static)	
0000021	raspberrypi	3	critical	09:14:48	09:33:05	10.5.51.238	Q	Q	80	http	TCP	dns	synergygrade.viva.com.cn	potential data leakage (suspicious)	(heuristic)	
0000022	raspberrypi	1	critical	08:51:58	08:51:56	10.5.50.68	48446	10.5.50.1	53	dns	UDP	dns	ip-api.com	pinfns (suspicious)	(static)	
0000023	raspberrypi	1	critical	08:42:59	08:42:59	10.5.51.16	50646	10.5.50.1	53	dns	UDP	dns	ip-api.com	pinfns (suspicious)	(static)	
0000024	raspberrypi	1	critical	08:08:09	08:08:09	10.5.50.216	43064	10.5.50.1	53	dns	UDP	dns	pinfns.io	pinfns (suspicious)	(static)	

Gambar 4. 1 Hasil monitoring

Berkat aplikasi pencerminan *port mirroring* 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 16.05.03². Proses verifikasi data berhasil diselesaikan pada pukul 16.05.07³, memastikan data akurat dan dapat diandalkan. Pengujian dan pemantauan jaringan dilakukan dalam *mode real-time*, dengan durasi perekaman *IP* 10.5.20.119⁴ hingga peralihan ke alamat *IP* 10.5.50.1⁴. 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 mencurigakan⁷ 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 08.03.19² dan berhasil pada pukul 15.59.19². Pengujian difokuskan pada alamat IP 10.5.50.64³, 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 mencurigakan⁵ 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.