



UNIVERSITAS
GADJAH MADA

PENELITIAN PENGARUH MAC ADDRESS COLLISION PADA PERANGKAT YANG DAPAT MERUBAH MAC ADDRESS

Gabriel Possenti Kheisa Drianasta

19/442374/PA/19123

Pembimbing 1 : Bakhtiar Alldino A.S., S.Si, M.Cs

Pembimbing 2 : Prof. Jazi Eko Istiyanto, M.Sc., Ph.D

Pengubahan MAC address secara random

Wi-Fi 6



On



TelenetWiFree



TELENET

Advanced options

Manage Wi-Fi settings

Random hardware addresses

Use random hardware addresses to make it harder for people to track your location when you connect to this Wi-Fi network. This setting applies only to this network and any change will take effect the next time you connect.

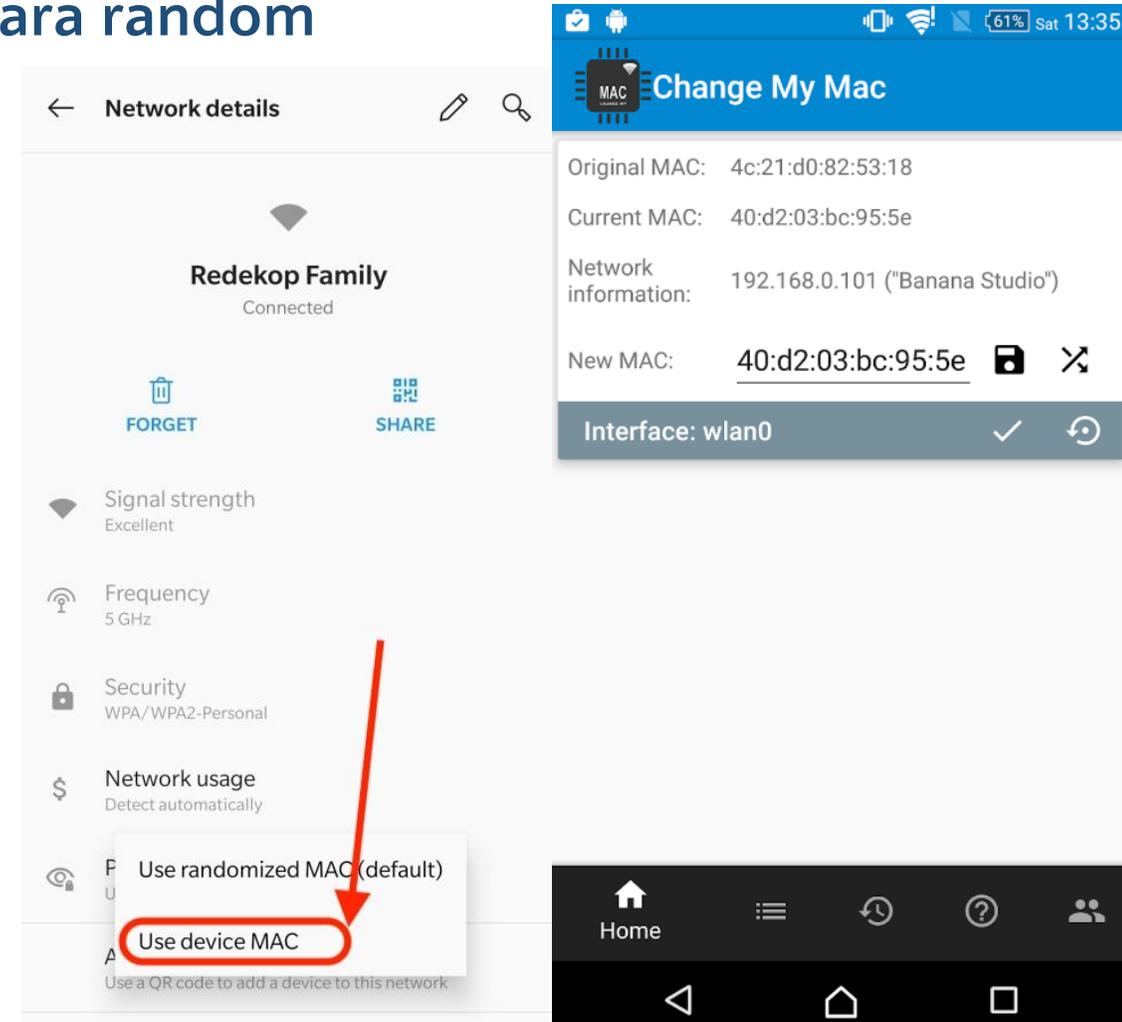
Use random addresses for this network

On

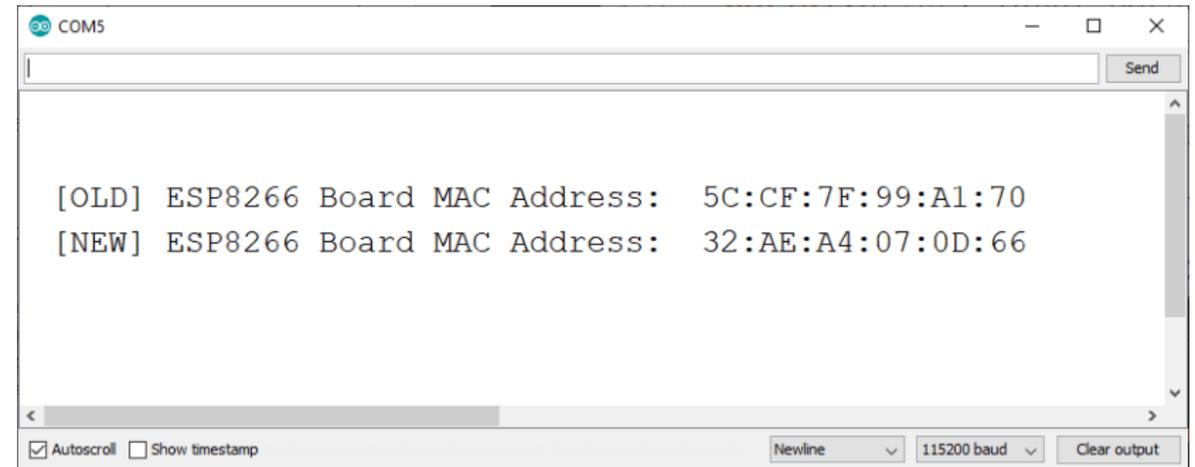
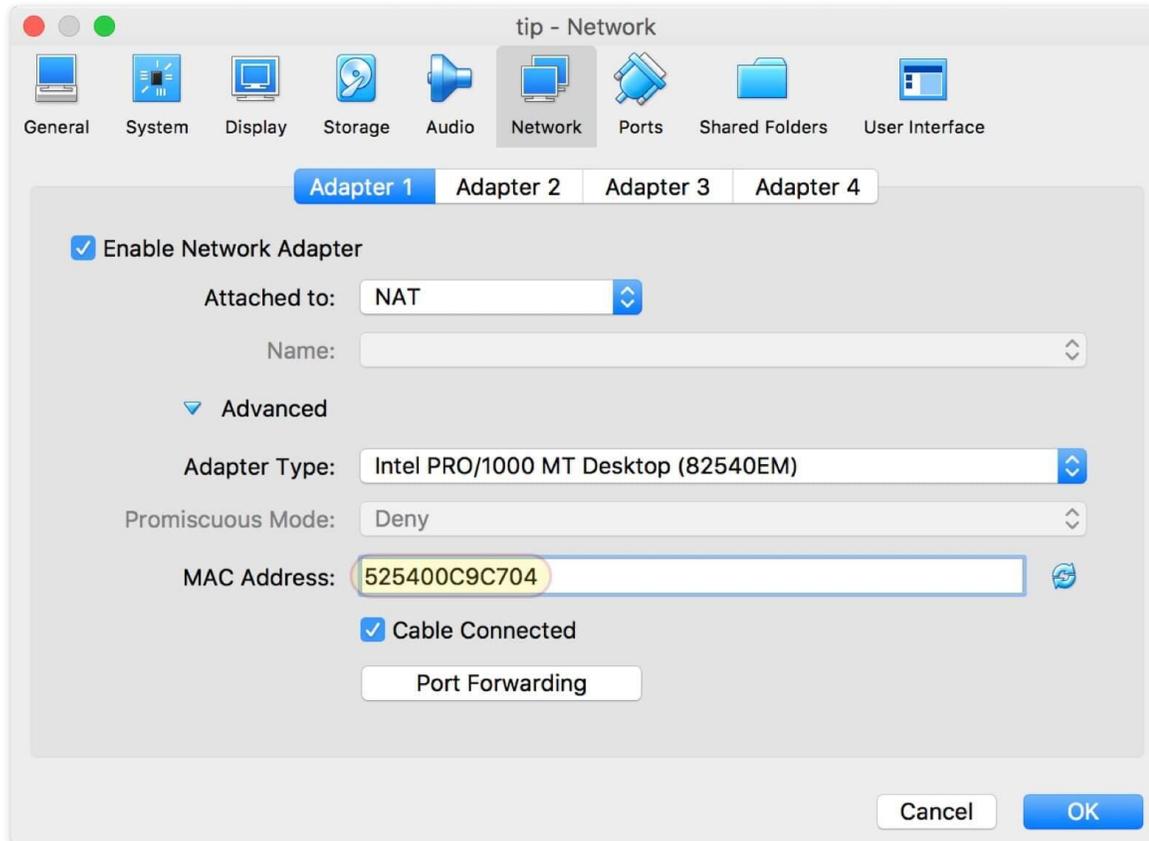
On

Off

Change daily



Pengubahan MAC address secara manual



Meskipun MAC Address umumnya tidak dapat diubah, namun beberapa perangkat bahkan manufaktur boleh merubah alamat yang sudah bersifat hard-coded tersebut. Dengan mengubah MAC Address, identitas perangkat secara fisik juga berubah. (Cardenas, Edgar D., 2013) Dengan mengubah 24-bit pertama, maka perangkat tersebut dapat memanipulasi kode penerbit perangkat yang sebenarnya. Beberapa penerbit perangkat tersebut, khususnya dengan sistem operasi (OS) dan dukungan driver Network Interface Card (NIC) terbaru yang memperbolehkan mengubah MAC address demi tujuan anonimitas penggunaannya. Tetapi jika MAC address perangkat tersebut memiliki address yang sama dengan perangkat lain dalam suatu jaringan yang sama, maka secara teoritis akan menimbulkan masalah pada jaringan tersebut.

Tujuan dan manfaat penelitian

Mengetahui dampak yang ditimbulkan pada jaringan apabila terjadi kesamaan MAC Address atau disebut juga dengan MAC Address collision.

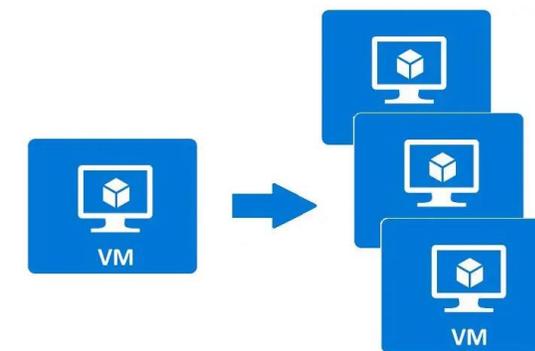
Hasil data berpotensi dapat diimplementasikan sebagai indikator MAC address collision dalam suatu jaringan

Pendahuluan



Batasan Masalah

1. Pengujian dilakukan dengan perangkat yang mendukung penggantian MAC Address.
2. Pengujian dilakukan dengan menggunakan simulasi pada virtual machine. Wireshark tidak dapat mengamati aliran data dari dua buah VM karena tipe jaringan *internal network* VirtualBox menggunakan switch, bukan hub.
3. Pengujian dilakukan dengan perangkat yang mendukung penggantian MAC Address acak pada perangkat yang mendukung *MAC Address randomization* seperti mikrokontroler ESP8266 atau perangkat lainnya dengan dukungan sistem operasi seperti Android 6.0 dan seterusnya, dan Windows 10 dan seterusnya, serta pada *virtual NIC* pada *virtual environment* seperti VirtualBox.
4. Variable yang diambil dari penelitian ini adalah waktu *ping* atau *latency*, data *traceroute*, serta hasil *request / response* pada *application layer* pada port HTTP (TCP 80 dan TCP 1111 untuk Android), serta beberapa data yang diambil dengan perangkat lunak penganalisa jaringan Wireshark.



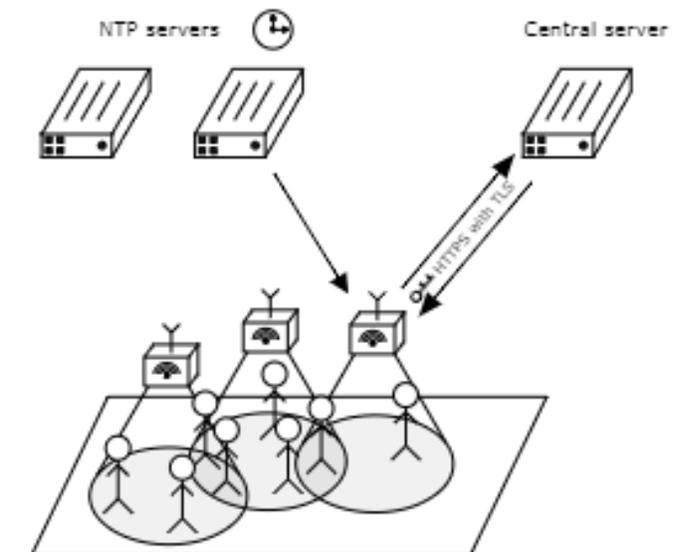
Metode randomization MAC address Practical Hash-Based Anonymity for MAC Addresses (Junade Ali, Vladimir Dyo. Cloudflare Inc)

- Input salt adalah nilai yang digunakan untuk menambah keacakan dan kompleksitas pada proses komputasi hash.
- Input salt harus unik dan tidak dapat diprediksi untuk membuat lebih sulit bagi penyerang untuk melakukan serangan pada data yang dihash.
- Input salt dapat berupa nilai apa pun yang memenuhi kriteria ini, seperti angka acak atau kombinasi nilai acak.
- Fungsi derivasi seperti Argon2, PBKDF2, BCrypt, atau SCrypt dapat digunakan dengan faktor kerja yang dapat dikonfigurasi untuk menghasilkan nilai salt.
- Input salt harus dijaga kerahasiaannya dan tidak diungkapkan kepada penyerang potensial.
- Nilai salt dapat diputar (rotate) sesuai keinginan untuk lebih meningkatkan keacakan dan kompleksitas proses komputasi hash.

```
1 def get_mac_address():
2     """Gets the MAC address of the current device."""
3     mac_address = os.environ.get("WLAN_MAC")
4     if mac_address is None:
5         raise ValueError("Unable to get MAC address")
6     return mac_address
7
8 def generate_salt(length):
9     """Generates a random salt of the specified length."""
10    salt = ""
11    for _ in range(length):
12        salt += chr(random.randint(32, 126))
13    return salt
14
15 def hash_mac_address(mac_address, salt):
16    """Hashes the MAC address with the specified salt."""
17    hash = hashlib.sha256()
18    hash.update(mac_address.encode("utf-8"))
19    hash.update(salt.encode("utf-8"))
20    return hash.hexdigest()
21
22 def main():
23    """Replaces the password with the MAC address and hashes it."""
24    mac_address = get_mac_address()
```

MAC Address Anonymization for Crowd Counting (Jean-François Determe, Sophia Azzagnuni, François Horlin, Philippe De Doncker)

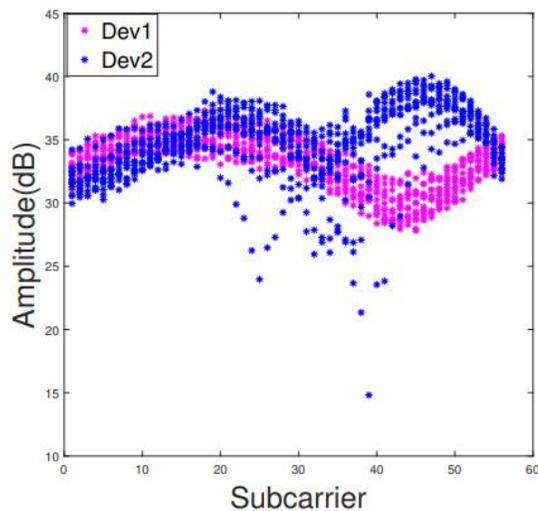
- Server pusat mempertahankan array server pepper terbaru yang terdiri dari 20 server pepper selama 20 menit.
- Sensor secara berkala mengambil server pepper menggunakan tautan HTTPS dengan Transport Layer Security (SSL/TLS).
- Sensor menggunakan setiap server pepper untuk satu kerangka waktu satu menit tertentu.
- Server pepper dibuat menggunakan pseudo-random number generator (PRNG).
- Server pusat menerima permintaan anonymized probe requests (PR) dari sensor.
- Server pusat menyinkronkan jam sensor menggunakan server Network Time Protocol (NTP), untuk crowd counting.
- Nilai MAC digunakan sebagai source address (SA) dari PR, nilai MAC dihash dengan fungsi SHA256 yang di "pepper" sebelum permintaan anonym PR dikirim ke server pusat
- **Pepper adalah nilai rahasia, sedangkan salt adalah nilai acak (random)**



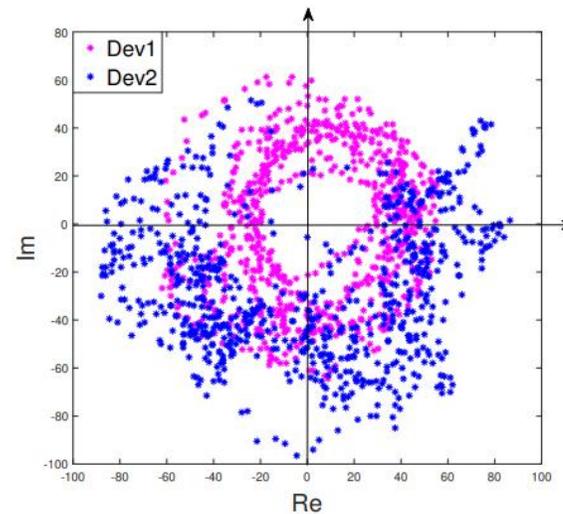
Tinjauan Pustaka



Peng Jiang, Hongyi Wu, Cong Wang, Chunsheng Xin “Virtual MAC Spoofing Detection through Deep Learning”. Department of ECE, Old Dominion University, Norfolk, VA, USA.

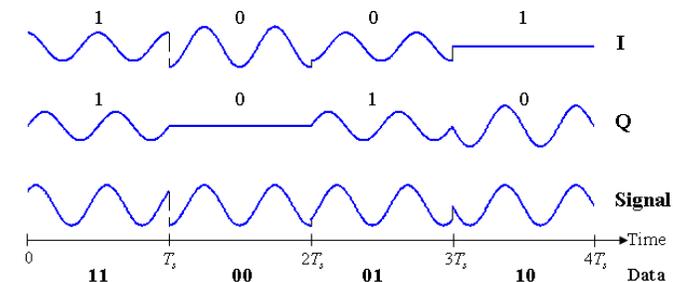
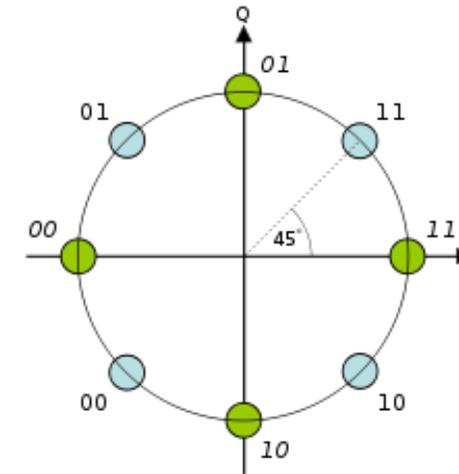


(a) Amplitude



(b) Amplitude and phase

Fig. 2: CSI of 20 packet transmissions from two devices at the same location, a) amplitude, b) amplitude and phase

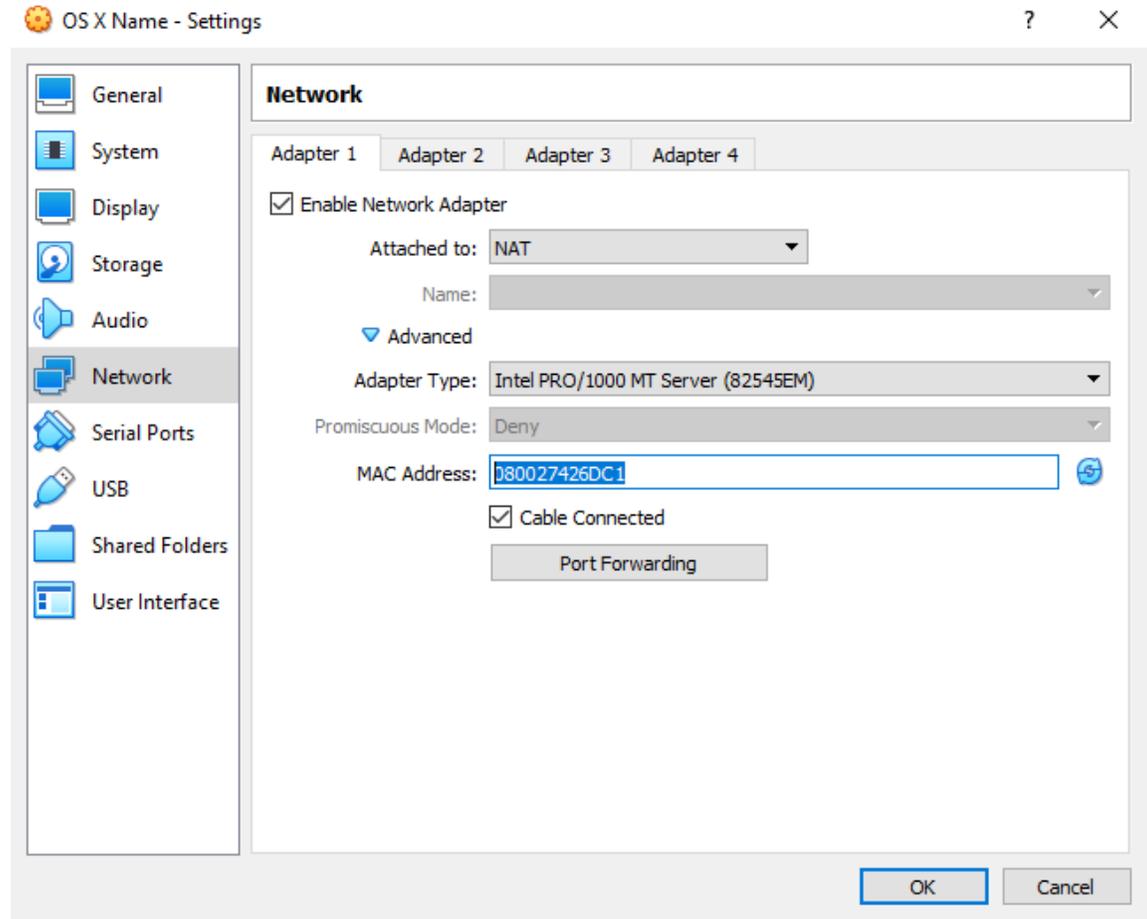


Metode Penelitian



Penggantian MAC address manual

```
1 #include <ESP8266WiFi.h>
2
3 void setup() {
4     uint8_t new_mac[6] = {0xAA, 0xBB, 0xCC, 0xDD, 0xEE, 0xFF};
5     wifi_set_macaddr(STATION_IF, new_mac);
6 }
7
8 void loop() {
9
10 }
11
```



Tabel 4.1 Daftar Komponen Penunjang

No.	Nama Komponen	Fungsi
1.	NodeMCU dengan mikrokontroler ESP8266	Perangkat <i>client</i> yang MAC Address nya akan diubah.
2.	Ponsel dengan sistem operasi Android (<i>rooted</i>).	Perangkat <i>client</i> yang MAC Address nya akan diubah.
3.	<i>Router</i> yang akan diuji	Perangkat <i>router</i> yang MAC Address nya akan disamakan.

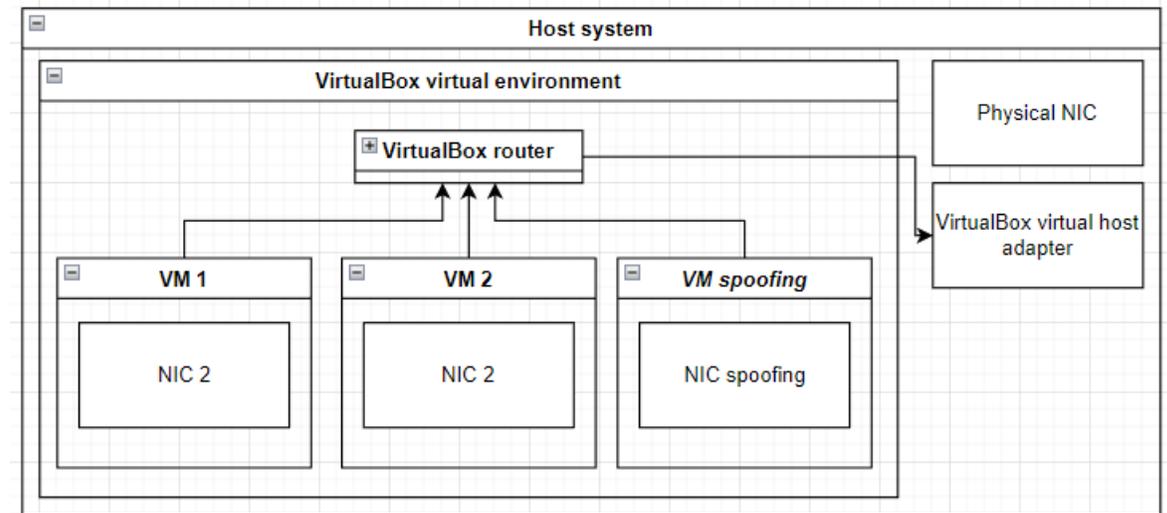
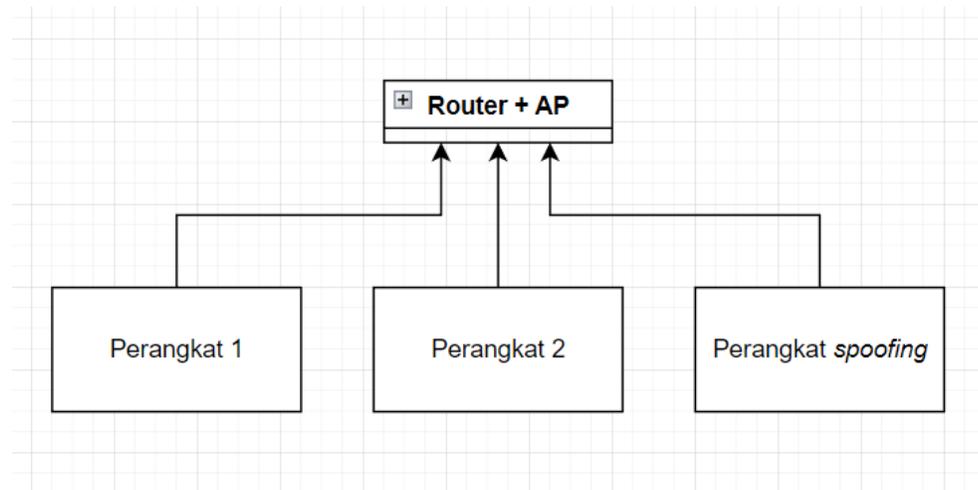
Tabel 4.2 Daftar Peralatan Lunak Penunjang

No.	Nama Alat	Fungsi
1.	Arduino IDE	Untuk perubahan MAC Address.
2.	MAC Address changer (Android)	Untuk perubahan MAC Address.
3.	VirtualBox	Untuk percobaan lanjutan pada mesin virtual.
4.	Wireshark	Untuk menganalisis paket data
5.	cURL atau browser	Untuk melakukan HTTP GET <i>request</i>

Metode Penelitian



Rancangan sistem



Rancangan pengujian

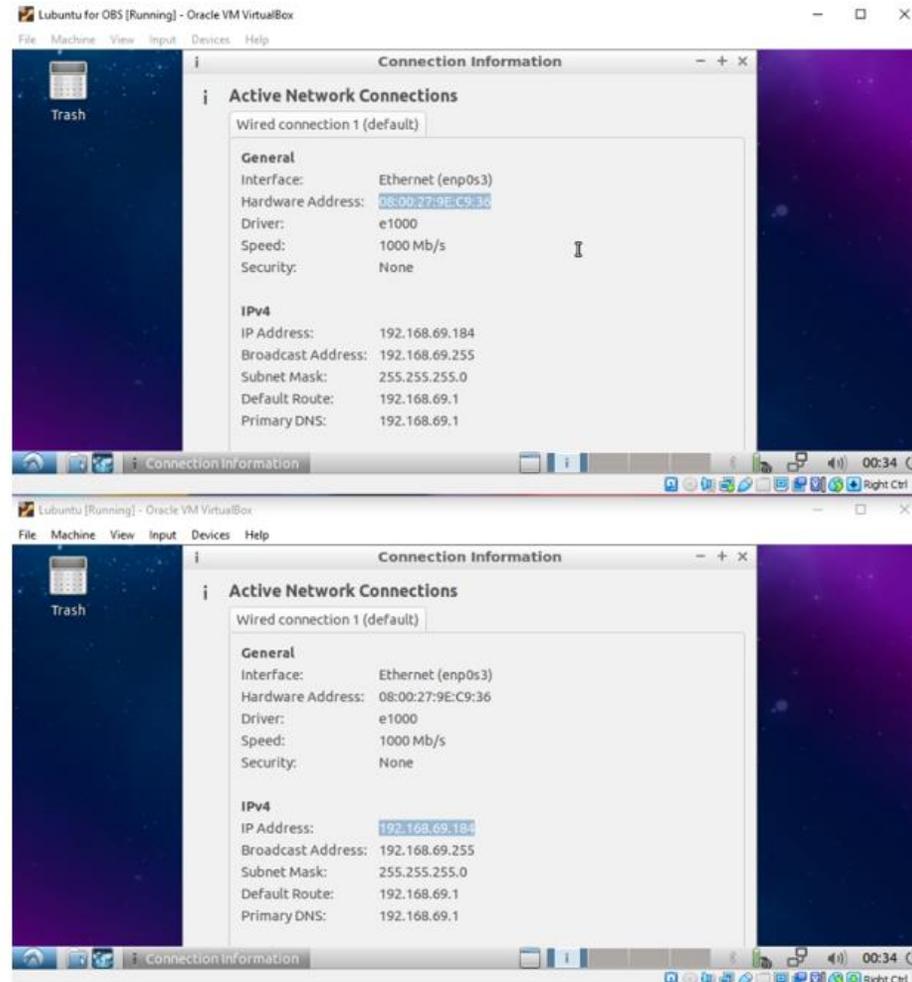
1. Mengambil data ping, traceroute, HTTP GET request lingkungan virtual
2. Mengambil data ping, traceroute, HTTP GET request lingkungan fisik
3. Membandingkan data pengujian antara lingkungan virtual dengan lingkungan fisik
4. Membandingkan keberhasilan penelitian antara lingkungan virtual dengan lingkungan fisik untuk beberapa kali pengujian

Metode Penelitian



Variabel bebas	MAC address yang disamakan	Disamakan dengan perangkat A, B, dan C
Variabel terikat	Ping	latency, berhasil/gagal
	Traceroute	latency, hop, berhasil/gagal
	HTTP GET	berhasil/gagal
Variabel kontrol	Spesifikasi VM	OS, disk, NIC

Implementasi

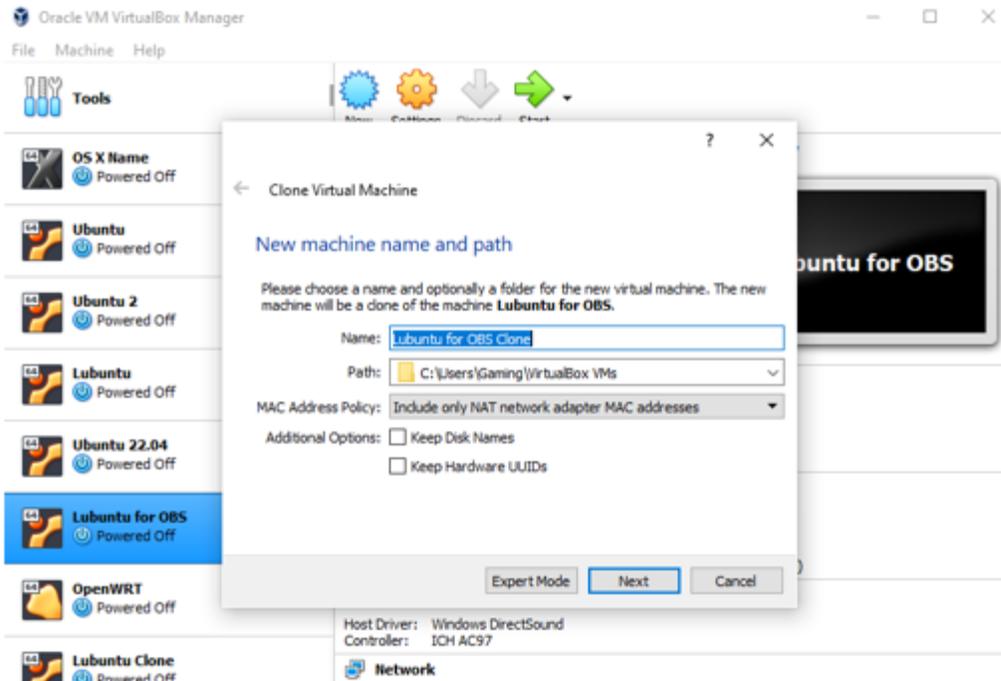


Gambar 4.5 Dua buah VM dengan MAC address yang sama

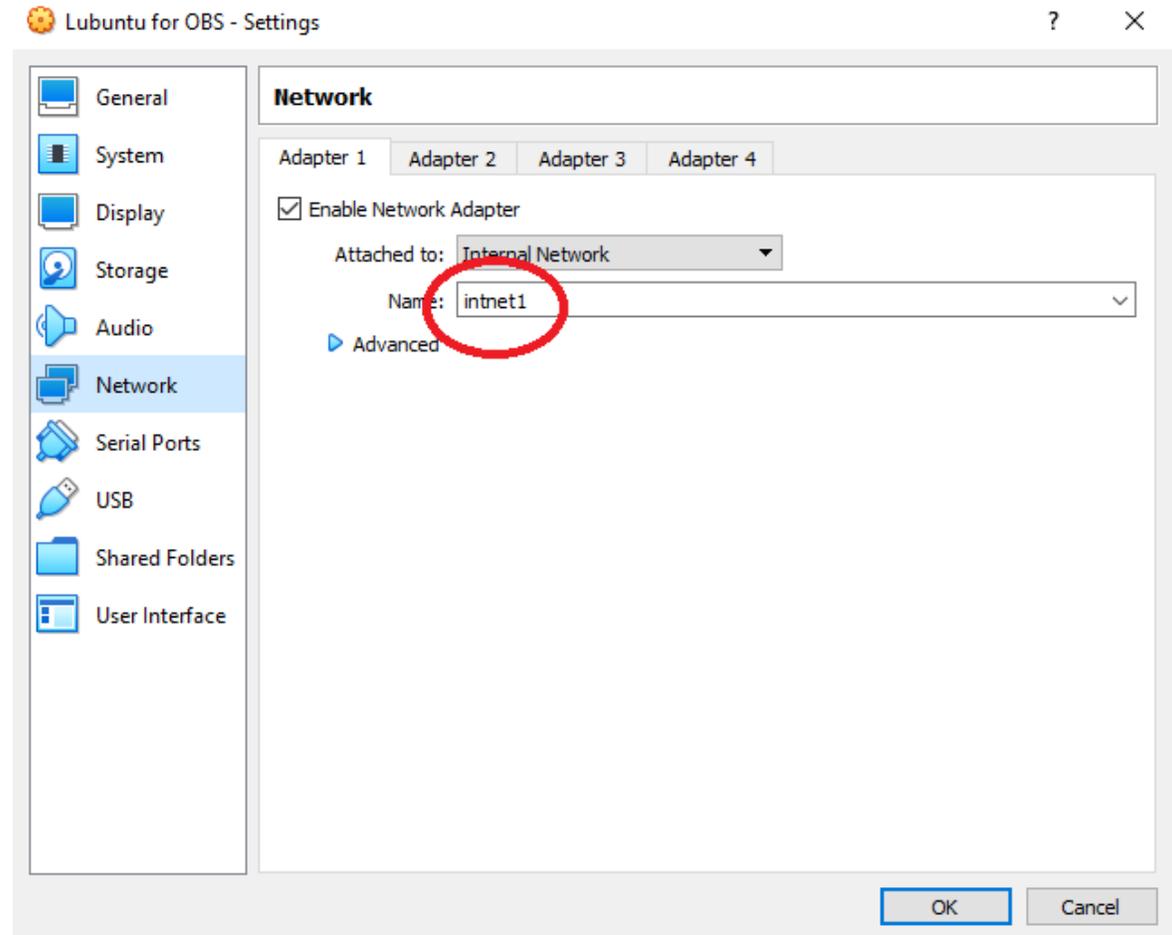
Interface: Ethernet (enp0s3)
Hardware Address: 08:00:27:9E:C9:36

Interface: Ethernet (enp0s3)
Hardware Address: 08:00:27:9E:C9:36

Implementasi



Gambar 4.6 Clone Virtual Machine



Implementasi



The screenshot displays four Oracle VM VirtualBox windows, each showing a Linux terminal window with network-related output. The windows are arranged in a 2x2 grid.

- Top-Left Window:** Shows a series of received packets on the eth0 interface. The output includes source addresses and VLAN IDs, such as (284.019317) br-lan: received packet on eth0 with own address as source address (addr:00:00:27:a0:df:71, vlan:0).
- Top-Right Window:** Shows the output of a ping command to 192.168.69.1. It lists 20 successful ping attempts with response times ranging from approximately 0.153 ms to 0.420 ms. The ping statistics at the bottom indicate 20 packets transmitted, 20 received, 0% packet loss, and a total time of 19444ms.
- Bottom-Left Window:** Shows the output of a ping command to 192.168.69.1. It lists 18 successful ping attempts with response times ranging from approximately 0.241 ms to 0.296 ms. The ping statistics at the bottom indicate 18 packets transmitted, 18 received, 0% packet loss, and a total time of 17406ms.
- Bottom-Right Window:** Shows the output of a ping command to 192.168.69.1. It lists 19 successful ping attempts with response times ranging from approximately 1.11 ms to 0.982 ms. The ping statistics at the bottom indicate 19 packets transmitted, 19 received, 0% packet loss, and a total time of 18031ms.

Hasil pada virtual (normal)



Virtual (normal)

Pengujian ke	Dari	Tujuan	PING		Traceroute			HTTP GET
			Berhasil	Latency	Berhasil	Latency (ms)	Hop	Berhasil
1	VM A	VM Router	1	0.301	1	0.306	1	1
2	VM A	VM Router	1	0.273	1	0.354	1	1
3	VM A	VM Router	1	0.273	1	0.343	1	1
4	VM A	VM Router	1	0.237	1	0.309	1	1
5	VM A	VM Router	1	0.329	1	0.295	1	1
6	VM A	VM Router	1	0.251	1	0.304	1	1
7	VM A	VM Router	1	0.237	1	0.292	1	1
8	VM A	VM Router	1	0.256	1	0.341	1	1
9	VM A	VM Router	1	0.282	1	0.347	1	1
10	VM A	VM Router	1	0.308	1	0.295	1	1

Keberhasilan (%)			100		100			100
Rata-rata				0.2747		0.3186	1	
Standar deviasi				0.029		0.023	0	

Pengujian ke	Dari	Tujuan	PING		Traceroute			HTTP GET
			Berhasil	Latency	Berhasil	Latency (ms)	Hop	Berhasil
1	VM A	VM B	1	0.301	1	0.297	1	1
2	VM A	VM B	1	0.304	1	0.304	1	1
3	VM A	VM B	1	0.312	1	0.299	1	1
4	VM A	VM B	1	0.299	1	0.302	1	1
5	VM A	VM B	1	0.313	1	0.298	1	1
6	VM A	VM B	1	0.298	1	0.297	1	1
7	VM A	VM B	1	0.298	1	0.297	1	1
8	VM A	VM B	1	0.297	1	0.299	1	1
9	VM A	VM B	1	0.292	1	0.297	1	1
10	VM A	VM B	1	0.299	1	0.297	1	1

Keberhasilan (%)			100		100			100
Rata-rata				0.3013		0.2987	1	
Standar deviasi				0.006		0.002	0	

Pengujian ke	Dari	Tujuan	PING		Traceroute			HTTP GET
			Berhasil	Latency	Berhasil	Latency (ms)	Hop	Berhasil
1	VM B	VM Router	1	0.255	1	0.294	1	1
2	VM B	VM Router	1	0.281	1	0.302	1	1
3	VM B	VM Router	1	0.242	1	0.292	1	1
4	VM B	VM Router	1	0.267	1	0.303	1	1
5	VM B	VM Router	1	0.298	1	0.294	1	1
6	VM B	VM Router	1	0.313	1	0.291	1	1
7	VM B	VM Router	1	0.299	1	0.293	1	1
8	VM B	VM Router	1	0.328	1	0.293	1	1
9	VM B	VM Router	1	0.328	1	0.302	1	1
10	VM B	VM Router	1	0.314	1	0.305	1	1

Keberhasilan (%)			100		100			100
Rata-rata				0.2925		0.2969	1	
Standar deviasi				0.029		0.005	0	

Pengujian ke	Dari	Tujuan	PING		Traceroute			HTTP GET
			Berhasil	Latency	Berhasil	Latency (ms)	Hop	Berhasil
1	VM B	VM A	1	0.299	1	0.301	1	1
2	VM B	VM A	1	0.299	1	0.297	1	1
3	VM B	VM A	1	0.296	1	0.297	1	1
4	VM B	VM A	1	0.304	1	0.302	1	1
5	VM B	VM A	1	0.297	1	0.304	1	1
6	VM B	VM A	1	0.298	1	0.302	1	1
7	VM B	VM A	1	0.299	1	0.303	1	1
8	VM B	VM A	1	0.298	1	0.298	1	1
9	VM B	VM A	1	0.297	1	0.297	1	1
10	VM B	VM A	1	0.301	1	0.298	1	1

Keberhasilan (%)			100		100			100
Rata-rata				0.2988		0.2999	1	
Standar deviasi				0.002		0.003	0	

Hasil pada virtual (VM-pengamat)



Pengujian ke	Dari	Tujuan	PING		Traceroute			HTTP GET	Pengujian ke	Dari	Tujuan	PING		Traceroute			HTTP GET
			Berhasil	Latency	Berhasil	Latency (ms)	Hop	Berhasil				Berhasil	Latency	Berhasil	Latency (ms)	Hop	Berhasil
1	VM variabel bebas	VM 1	1	0.299	1	0.392	1	1	1	VM variabel bebas	VM router	1	0.285	1	0.391	1	0
2	VM variabel bebas	VM 1	1	0.317	1	0.316	1	1	2	VM variabel bebas	VM router	1	0.306	1	0.302	1	0
3	VM variabel bebas	VM 1	1	0.261	1	0.341	1	1	3	VM variabel bebas	VM router	1	0.272	1	0.293	1	0
4	VM variabel bebas	VM 1	1	0.315	1	0.389	1	1	4	VM variabel bebas	VM router	1	0.353	1	0.286	1	0
5	VM variabel bebas	VM 1	1	0.267	1	0.378	1	1	5	VM variabel bebas	VM router	1	0.287	1	0.301	1	0
6	VM variabel bebas	VM 1	1	0.266	1	0.387	1	1	6	VM variabel bebas	VM router	1	0.268	1	0.276	1	0
7	VM variabel bebas	VM 1	1	0.298	1	0.385	1	1	7	VM variabel bebas	VM router	1	0.284	1	0.284	1	0
8	VM variabel bebas	VM 1	1	0.299	1	0.372	1	1	8	VM variabel bebas	VM router	1	0.322	1	0.293	1	0
9	VM variabel bebas	VM 1	1	0.288	1	0.278	1	1	9	VM variabel bebas	VM router	1	0.285	1	0.387	1	0
10	VM variabel bebas	VM 1	1	0.359	1	0.316	1	1	10	VM variabel bebas	VM router	1	0.276	1	0.285	1	0
Keberhasilan (%)			100		100			100	Keberhasilan (%)			100		100			0
Rata-rata				0.2969		0.3554	1		Rata-rata			0.2938		0.3098	1		
Standar deviasi				0.028		0.038	0		Standar deviasi			0.025		0.04	0		

Virtual (collision VM variabel bebas dan pengamat) dari VM 1

Pengujian ke	Dari	Tujuan	PING		Traceroute			HTTP GET	Pengujian ke	Dari	Tujuan	PING		Traceroute			HTTP GET
			Berhasil	Latency	Berhasil	Latency (ms)	Hop	Berhasil				Berhasil	Latency	Berhasil	Latency (ms)	Hop	Berhasil
1	VM 1	VM variabel bebas	1	0.299	1	0.395	1	0	1	VM 1	VM router	1	0.339	1	0.335	1	1
2	VM 1	VM variabel bebas	1	0.301	1	0.295	1	0	2	VM 1	VM router	1	0.295	1	0.297	1	1
3	VM 1	VM variabel bebas	1	0.304	1	0.916	1	0	3	VM 1	VM router	1	0.307	1	0.308	1	1
4	VM 1	VM variabel bebas	1	0.291	1	0.267	1	0	4	VM 1	VM router	1	0.355	1	0.581	1	1
5	VM 1	VM variabel bebas	1	0.282	1	0.282	1	0	5	VM 1	VM router	1	0.302	1	0.284	1	1
6	VM 1	VM variabel bebas	1	0.378	1	0.563	1	0	6	VM 1	VM router	1	0.301	1	0.622	1	1
7	VM 1	VM variabel bebas	1	0.296	1	0.932	1	0	7	VM 1	VM router	1	0.263	1	0.334	1	1
8	VM 1	VM variabel bebas	1	0.303	1	0.292	1	0	8	VM 1	VM router	1	0.281	1	0.733	1	1
9	VM 1	VM variabel bebas	1	0.311	1	0.591	1	0	9	VM 1	VM router	1	0.278	1	0.291	1	1
10	VM 1	VM variabel bebas	1	0.354	1	0.288	1	0	10	VM 1	VM router	1	0.313	1	0.381	1	1
Keberhasilan (%)			100		100			0	Keberhasilan (%)			100		100			100
Rata-rata				0.3119		0.4821	1		Rata-rata			0.3034		0.4166	1		
Standar deviasi				0.029		0.247	0		Standar deviasi			0.026		0.156	0		

Ping dengan label biru DUP!

VM variabel bebas dan VM pengamat (VM 2) memiliki alamat IP yang sama (diatur server DHCP)

Hasil pada virtual (VM-router)



Virtual (collision with OpenWrt) VM OpenWrt																	
Pengujian ke	Dari	Tujuan	PING		Traceroute			HTTP GET	Pengujian ke	Dari	Tujuan	PING		Traceroute			HTTP GET
			Berhasil	Latency	Berhasil	Latency (ms)	Hop	Berhasil				Berhasil	Latency	Berhasil	Latency (ms)	Hop	Berhasil
1	VM variabel bebas	VM OpenWrt	1	1.93	1	1.16	1	1	1	VM 2	VM OpenWrt	1	0.339	1	0.416	1	1
2	VM variabel bebas	VM OpenWrt	1	0.909	1	0.401	1	1	2	VM 2	VM OpenWrt	1	0.31	1	0.338	1	1
3	VM variabel bebas	VM OpenWrt	1	1	1	0.319	1	1	3	VM 2	VM OpenWrt	1	0.312	1	0.335	1	1
4	VM variabel bebas	VM OpenWrt	1	1.11	1	0.432	1	1	4	VM 2	VM OpenWrt	1	0.318	1	0.318	1	1
5	VM variabel bebas	VM OpenWrt	1	1.07	1	0.264	1	1	5	VM 2	VM OpenWrt	1	0.354	1	0.275	1	1
6	VM variabel bebas	VM OpenWrt	1	1.13	1	2.058	1	1	6	VM 2	VM OpenWrt	1	0.153	1	0.319	1	1
7	VM variabel bebas	VM OpenWrt	1	1.1	1	0.297	1	1	7	VM 2	VM OpenWrt	1	0.338	1	0.304	1	1
8	VM variabel bebas	VM OpenWrt	1	2.1	1	0.303	1	1	8	VM 2	VM OpenWrt	1	0.398	1	0.319	1	1
9	VM variabel bebas	VM OpenWrt	1	1.06	1	1.701	1	1	9	VM 2	VM OpenWrt	1	0.404	1	0.288	1	1
10	VM variabel bebas	VM OpenWrt	1	1.12	1	0.329	1	1	10	VM 2	VM OpenWrt	1	0.373	1	0.442	1	1
Keberhasilan (%)			100		100			100	Keberhasilan (%)			100		100			100
Rata-rata				1.2529		0.7264	1		Rata-rata				0.3299		0.3354	1	
Standar deviasi				0.388		0.633	0		Standar deviasi				0.067		0.051	0	

Pengujian ke	Dari	Tujuan	PING		Traceroute			HTTP GET
			Berhasil	Latency	Berhasil	Latency (ms)	Hop	Berhasil
1	VM 1	VM OpenWrt	1	0.251	1	0.851	1	1
2	VM 1	VM OpenWrt	1	0.246	1	0.301	1	1
3	VM 1	VM OpenWrt	1	0.241	1	0.401	1	1
4	VM 1	VM OpenWrt	1	0.244	1	0.307	1	1
5	VM 1	VM OpenWrt	1	0.526	1	0.311	1	1
6	VM 1	VM OpenWrt	1	0.275	1	0.285	1	1
7	VM 1	VM OpenWrt	1	0.351	1	0.377	1	1
8	VM 1	VM OpenWrt	1	0.295	1	0.319	1	1
9	VM 1	VM OpenWrt	1	0.637	1	0.858	1	1
10	VM 1	VM OpenWrt	1	0.364	1	0.327	1	1
Keberhasilan (%)			100		100			100
Rata-rata				0.343		0.4337	1	
Standar deviasi				0.129		0.213	0	

Hasil pada virtual (VM-router)



Virtual (collision with OpenWrt) VM 1

Pengujian ke	Dari	Tujuan	PING		Traceroute			HTTP GET
			Berhasil	Latency	Berhasil	Latency (ms)	Hop	Berhasil
1	VM variabel bebas	VM 1	1	0.337	1	0.409	1	1
2	VM variabel bebas	VM 1	1	0.322	1	0.326	1	1
3	VM variabel bebas	VM 1	1	0.331	1	0.698	1	1
4	VM variabel bebas	VM 1	1	0.374	1	0.286	1	1
5	VM variabel bebas	VM 1	1	0.308	1	0.309	1	1
6	VM variabel bebas	VM 1	1	0.326	1	0.678	1	1
7	VM variabel bebas	VM 1	1	0.358	1	0.502	1	1
8	VM variabel bebas	VM 1	1	0.332	1	0.572	1	1
9	VM variabel bebas	VM 1	1	0.307	1	0.562	1	1
10	VM variabel bebas	VM 1	1	0.335	1	0.581	1	1
Keberhasilan (%)			100		100			100
Rata-rata				0.333		0.4923	1	
Standar deviasi				0.019		0.144	0	

Pengujian ke	Dari	Tujuan	PING		Traceroute			HTTP GET
			Berhasil	Latency	Berhasil	latency (ms)	Hop	Berhasil
1	VM 2	VM 1	1	0.381	1	0.386	1	1
2	VM 2	VM 1	1	0.334	1	0.281	1	1
3	VM 2	VM 1	1	0.229	1	0.317	1	1
4	VM 2	VM 1	1	0.336	1	0.347	1	1
5	VM 2	VM 1	1	0.328	1	0.424	1	1
6	VM 2	VM 1	1	0.159	1	0.312	1	1
7	VM 2	VM 1	1	0.308	1	0.361	1	1
8	VM 2	VM 1	1	0.328	1	0.317	1	1
9	VM 2	VM 1	1	0.336	1	0.285	1	1
10	VM 2	VM 1	1	0.367	1	0.272	1	1
Keberhasilan (%)			100		100			100
Rata-rata				0.3106		0.3302	1	
Standar deviasi				0.063		0.047	0	

Virtual (collision with OpenWrt) VM duplicate

Pengujian ke	Dari	Tujuan	PING		Traceroute			HTTP GET
			Berhasil	Latency	Berhasil	Latency (ms)	Hop	Berhasil
1	VM 1	VM variabel bebas	1	0.352	1	0.696	1	1
2	VM 1	VM variabel bebas	1	0.297	1	0.294	2	1
3	VM 1	VM variabel bebas	1	0.331	1	0.321	1	1
4	VM 1	VM variabel bebas	1	0.314	1	0.403	1	1
5	VM 1	VM variabel bebas	1	0.296	1	0.294	1	1
6	VM 1	VM variabel bebas	1	0.332	1	0.301	2	1
7	VM 1	VM variabel bebas	1	0.296	1	0.295	1	1
8	VM 1	VM variabel bebas	1	0.385	1	0.824	1	1
9	VM 1	VM variabel bebas	1	0.317	1	0.263	1	1
10	VM 1	VM variabel bebas	1	0.308	1	0.308	1	1
Keberhasilan (%)			100		100			100
Rata-rata				0.3228		0.3999	1.2	
Standar deviasi				0.027		0.186	0.4	

Pengujian ke	Dari	Tujuan	PING		Traceroute			HTTP GET
			Berhasil	Latency	Berhasil	latency (ms)	Hop	Berhasil
1	VM 2	VM variabel bebas	1	0.369	1	0.473	1	1
2	VM 2	VM variabel bebas	1	0.369	1	0.575	2	1
3	VM 2	VM variabel bebas	1	0.431	1	0.294	1	1
4	VM 2	VM variabel bebas	1	0.389	1	0.534	2	1
5	VM 2	VM variabel bebas	1	0.323	1	0.317	2	1
6	VM 2	VM variabel bebas	1	0.308	1	0.277	1	1
7	VM 2	VM variabel bebas	1	0.319	1	0.555	1	1
8	VM 2	VM variabel bebas	1	0.312	1	0.261	1	1
9	VM 2	VM variabel bebas	1	0.391	1	0.551	2	1
10	VM 2	VM variabel bebas	1	0.329	1	0.525	1	1
Keberhasilan (%)			100		100			100
Rata-rata				0.354		0.4362	1.4	
Standar deviasi				0.04		0.125	0.49	

Ping dengan label biru DUP!

Traceroute dengan hop ganda

Hasil pada virtual (VM-router)



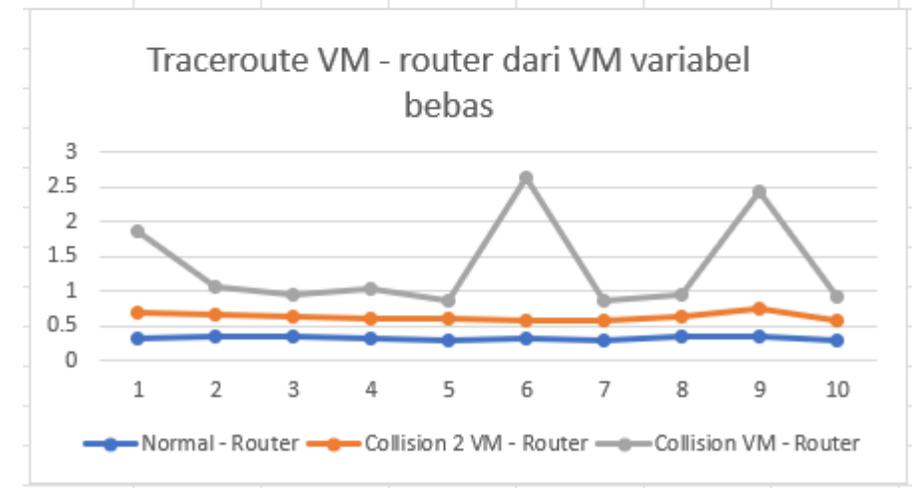
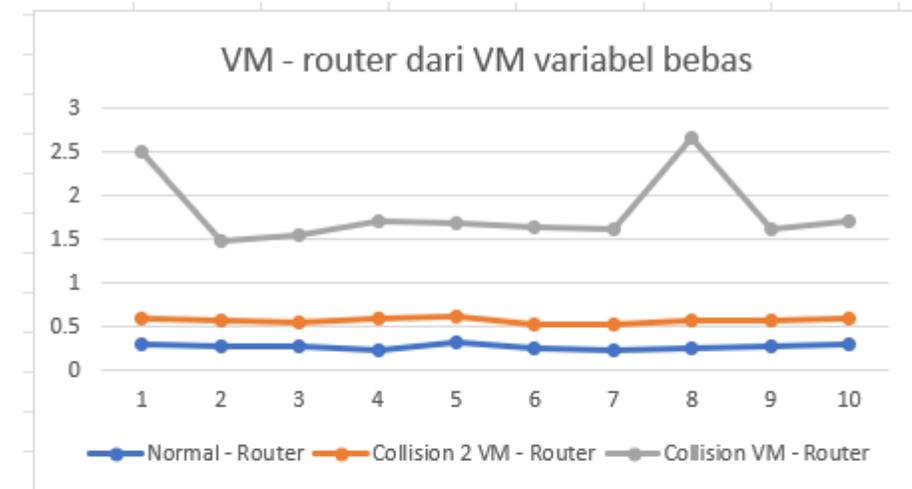
Virtual (collision with OpenWrt) VM 2

Pengujian ke	Dari	Tujuan	PING		Traceroute			HTTP GET
			Berhasil	Latency	Berhasil	Latency (ms)	Hop	Berhasil
1	VM variabel bebas	VM 2	1	0.282	1	0.581	1	1
2	VM variabel bebas	VM 2	1	0.341	1	0.314	1	1
3	VM variabel bebas	VM 2	1	0.341	1	0.361	1	1
4	VM variabel bebas	VM 2	1	0.287	1	0.735	1	1
5	VM variabel bebas	VM 2	1	0.268	1	0.345	1	1
6	VM variabel bebas	VM 2	1	0.261	1	0.296	1	1
7	VM variabel bebas	VM 2	1	0.332	1	0.282	1	1
8	VM variabel bebas	VM 2	1	0.341	1	0.323	1	1
9	VM variabel bebas	VM 2	1	0.345	1	0.301	1	1
10	VM variabel bebas	VM 2	1	0.334	1	0.294	1	1
Keberhasilan (%)			100		100			100
Rata-rata				0.3132		0.3832	1	
Standar deviasi				0.032		0.143	0	

Ping dengan label biru DUP!

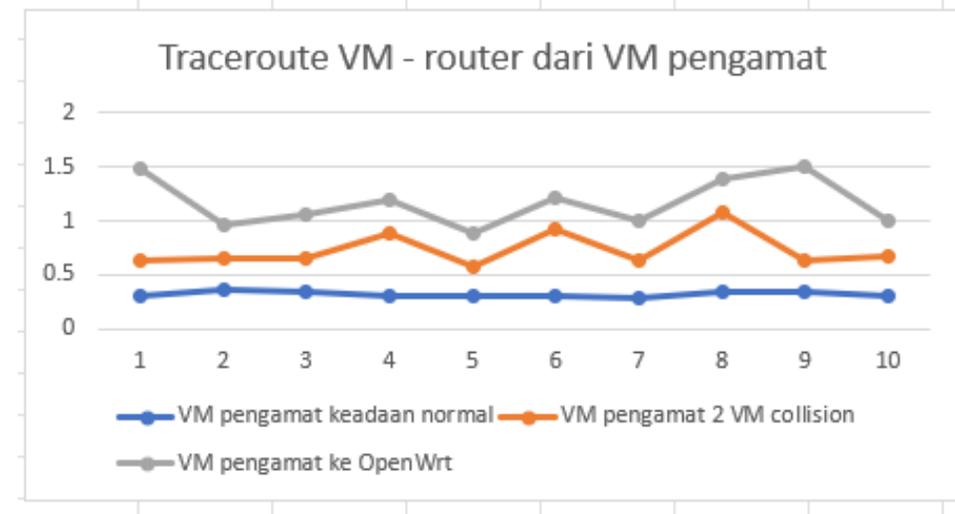
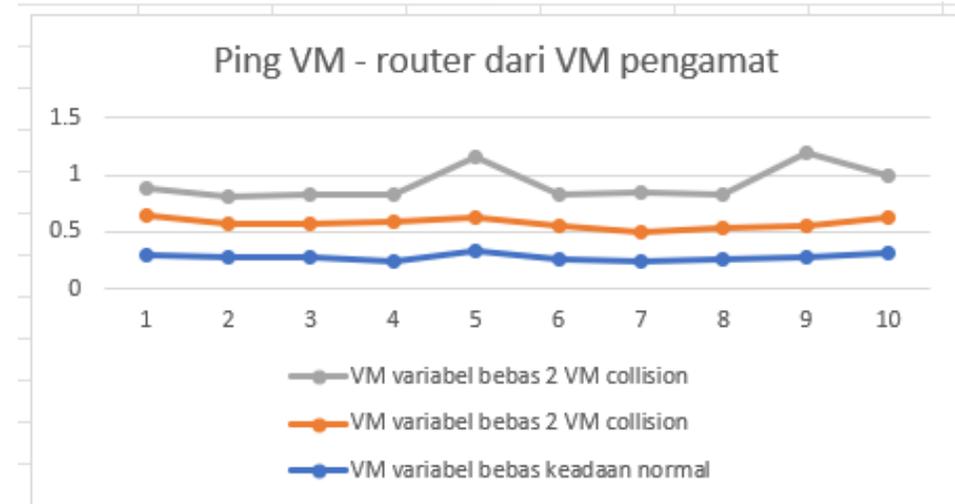


Statistik konektivitas ke router dari VM variabel bebas (data dari 3 skenario collision)





Statistik konektivitas ke router dari VM pengamat (data dari 3 skenario collision)



Hasil pada virtual (Wireshark, ping, VM-router)

No	Time	Source	Destination	Protocol	Length	Info
1	0	192.168.69.184	192.168.69.1	ICMP	100	Echo (ping) request, seq=1/256, ttl=64
2	0.0003	192.168.69.1	192.168.69.184	ICMP	100	Echo (ping) reply, seq=1/256, ttl=64
3	1.0172	192.168.69.184	192.168.69.1	ICMP	100	Echo (ping) request, seq=2/512, ttl=64
4	1.0175	192.168.69.1	192.168.69.184	ICMP	100	Echo (ping) reply, seq=2/512, ttl=64
5	2.0404	192.168.69.184	192.168.69.1	ICMP	100	Echo (ping) request, seq=3/768, ttl=64
6	2.0406	192.168.69.1	192.168.69.184	ICMP	100	Echo (ping) reply, seq=3/768, ttl=64
...

Gambar 5.14 Pengujian ping dari VM pengamat ke VM OpenWRT

No	Time	Source	Destination	Protocol	Length	Info
1	0	192.168.69.184	192.168.69.220	ICMP	100	Echo (ping) request, seq=1/256, ttl=64
2	0.0027	192.168.69.220	192.168.69.184	ICMP	100	Echo (ping) reply, seq=1/256, ttl=64
3	1.0174	192.168.69.184	192.168.69.220	ICMP	100	Echo (ping) request, seq=2/512, ttl=64
4	1.0177	192.168.69.220	192.168.69.184	ICMP	100	Echo (ping) reply, seq=2/512, ttl=64
5	2.0417	192.168.69.184	192.168.69.220	ICMP	100	Echo (ping) request, seq=3/768, ttl=64
6	2.0421	192.168.69.220	192.168.69.184	ICMP	100	Echo (ping) reply, seq=3/768, ttl=64
...

Gambar 5.15 Pengujian ping dari VM pengamat ke VM lainnya

Hasil pada virtual (Wireshark, ping, VM-router)

No	Time	Source	Destination	Protocol	Length	Info
1	0	192.168.69.184	192.168.69.181	ICMP	100	Echo (ping) request, seq=1/256, ttl=64
2	0.00016	192.168.69.181	192.168.69.184	ICMP	100	Echo (ping) reply, seq=1/256, ttl=64
3	0.00027	192.168.69.181	192.168.69.184	ICMP	100	Echo (ping) reply, seq=1/256, ttl=64
4	1.01348	192.168.69.184	192.168.69.181	ICMP	100	Echo (ping) request, seq=2/512, ttl=64
5	1.01375	192.168.69.181	192.168.69.181	ICMP	100	Echo (ping) reply, seq=2/512, ttl=64
6	1.01377	192.168.69.1	192.168.69.184	ICMP	128	(Redirect for host)
7	1.01386	192.168.69.181	192.168.69.184	ICMP	100	Echo (ping) reply, seq=2/512, ttl=64
...

Gambar 5.16 Pengujian ping dari VM pengamat ke VM duplicate

Hasil analisis Wireshark untuk ping duplicate

Protocol	Length	Info
ICMP	100	Echo (ping) request, seq=1/256, ttl=64
ICMP	100	Echo (ping) reply, seq=1/256, ttl=64
ICMP	100	Echo (ping) request, seq=2/512, ttl=64
ICMP	100	Echo (ping) reply, seq=2/512, ttl=64
ICMP	100	Echo (ping) request, seq=3/768, ttl=64
ICMP	100	Echo (ping) reply, seq=3/768, ttl=64
...

Protocol	Length	Info
ICMP	100	Echo (ping) request, seq=1/256, ttl=64
ICMP	100	Echo (ping) reply, seq=1/256, ttl=64
ICMP	100	Echo (ping) reply, seq=1/256, ttl=64
ICMP	100	Echo (ping) request, seq=2/512, ttl=64
ICMP	100	Echo (ping) reply, seq=2/512, ttl=64
ICMP	128	(Redirect for host)
ICMP	100	Echo (ping) reply, seq=2/512, ttl=64
...

Hasil pada lingkungan fisik (Collision NodeMCU - router)

MAC Variabel bebas (NodeMCU)	MAC Komputer	MAC Android	MAC router	avg Ping NodeMCU ke Router (ms)	avg Ping komputer ke Router (ms)	avg Ping Android ke Router (ms)		MAC Variabel bebas (NodeMCU)	MAC Komputer	MAC Android	MAC router	avg Ping NodeMCU ke Komputer (ms)	avg Ping komputer ke Komputer (ms)	avg Ping Android ke Komputer (ms)
84:F3:EB:CC:03:EF	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	8	1	30		84:F3:EB:CC:03:EF	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	17	0	28
50:D2:F5:2F:21:69	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	RTO	18	28		50:D2:F5:2F:21:69	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	RTO	RTO	51
MAC Variabel bebas (NodeMCU)	MAC Komputer	MAC Android	MAC router	avg Ping NodeMCU ke NodeMCU (ms)	avg Ping komputer ke NodeMCU (ms)	avg Ping Android ke NodeMCU (ms)		MAC Variabel bebas (NodeMCU)	MAC Komputer	MAC Android	MAC router	avg Ping NodeMCU ke Android (ms)	avg Ping komputer ke Android (ms)	avg Ping Android ke Android (ms)
84:F3:EB:CC:03:EF	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69					84:F3:EB:CC:03:EF	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	266	288	0
50:D2:F5:2F:21:69	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69					50:D2:F5:2F:21:69	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	RTO	RTO	0

Hasil pada lingkungan fisik (Collision NodeMCU - router)

MAC Variabel bebas (NodeMCU)	MAC Komputer	MAC Android	MAC router	HTTP GET NodeMCU ke Router (ms)	HTTP GET komputer ke Router (ms)	HTTP GET Android ke Router (ms)		MAC Variabel bebas (NodeMCU)	MAC Komputer	MAC Android	MAC router	HTTP GET NodeMCU ke Komputer (ms)	HTTP GET komputer ke Komputer (ms)	HTTP GET Android ke Komputer (ms)
84:F3:EB:CC:03:EF	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	Berhasil	Berhasil	Berhasil		84:F3:EB:CC:03:EF	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	Berhasil	Berhasil	Berhasil
50:D2:F5:2F:21:69	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	Gagal	Berhasil	Berhasil		50:D2:F5:2F:21:69	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	Gagal	Berhasil	Berhasil
MAC Variabel bebas (NodeMCU)	MAC Komputer	MAC Android	MAC router	HTTP GET NodeMCU ke NodeMCU (ms)	HTTP GET komputer ke NodeMCU (ms)	HTTP GET Android ke NodeMCU (ms)		MAC Variabel bebas (NodeMCU)	MAC Komputer	MAC Android	MAC router	HTTP GET NodeMCU ke Android (ms)	HTTP GET komputer ke Android (ms)	HTTP GET Android ke Android (ms)
84:F3:EB:CC:03:EF	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	Berhasil	Berhasil	Berhasil		84:F3:EB:CC:03:EF	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	Berhasil	Berhasil	Berhasil
50:D2:F5:2F:21:69	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	Gagal	Gagal	Gagal		50:D2:F5:2F:21:69	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	Gagal	Gagal	Berhasil

Hasil pada lingkungan fisik (Collision NodeMCU - Android)

MAC Variabel bebas (NodeMCU)	MAC Komputer	MAC Android	MAC router	avg Ping NodeMCU ke Router (ms)	avg Ping komputer ke Router (ms)	avg Ping Android ke Router (ms)		MAC Variabel bebas (NodeMCU)	MAC Komputer	MAC Android	MAC router	avg Ping NodeMCU ke Komputer (ms)	avg Ping komputer ke Komputer (ms)	avg Ping Android ke Komputer (ms)
84:F3:EB:CC:03:EF	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	8	1	30		84:F3:EB:CC:03:EF	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	17	0	28
04:E5:98:32:D0:1E	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	59	9	RTO		04:E5:98:32:D0:1E	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	37	0	RTO
MAC Variabel bebas (NodeMCU)	MAC Komputer	MAC Android	MAC router	avg Ping NodeMCU ke NodeMCU (ms)	avg Ping komputer ke NodeMCU (ms)	avg Ping Android ke NodeMCU (ms)		MAC Variabel bebas (NodeMCU)	MAC Komputer	MAC Android	MAC router	avg Ping NodeMCU ke Android (ms)	avg Ping komputer ke Android (ms)	avg Ping Android ke Android (ms)
84:F3:EB:CC:03:EF	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69					84:F3:EB:CC:03:EF	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	266	288	0
04:E5:98:32:D0:1E	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69					04:E5:98:32:D0:1E	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	RTO	RTO	0

Hasil pada lingkungan fisik (Collision NodeMCU - Android)

MAC Variabel bebas (NodeMCU)	MAC Komputer	MAC Android	MAC router	HTTP GET NodeMCU ke Router (ms)	HTTP GET komputer ke Router (ms)	HTTP GET Android ke Router (ms)		MAC Variabel bebas (NodeMCU)	MAC Komputer	MAC Android	MAC router	HTTP GET NodeMCU ke Komputer (ms)	HTTP GET komputer ke Komputer (ms)	HTTP GET Android ke Komputer (ms)
84:F3:EB:CC:03:EF	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	Berhasil	Berhasil	Berhasil		84:F3:EB:CC:03:EF	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	Berhasil	Berhasil	Berhasil
04:E5:98:32:D0:1E	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	Gagal	Berhasil	Berhasil		04:E5:98:32:D0:1E	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	Gagal	Berhasil	Gagal
MAC Variabel bebas (NodeMCU)	MAC Komputer	MAC Android	MAC router	HTTP GET NodeMCU ke NodeMCU (ms)	HTTP GET komputer ke NodeMCU (ms)	HTTP GET Android ke NodeMCU (ms)		MAC Variabel bebas (NodeMCU)	MAC Komputer	MAC Android	MAC router	HTTP GET NodeMCU ke Android (ms)	HTTP GET komputer ke Android (ms)	HTTP GET Android ke Android (ms)
84:F3:EB:CC:03:EF	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	Berhasil	Berhasil	Berhasil		84:F3:EB:CC:03:EF	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	Berhasil	Berhasil	Berhasil
04:E5:98:32:D0:1E	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69		Gagal	Gagal		04:E5:98:32:D0:1E	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	Gagal	Berhasil	Berhasil

Hasil pada lingkungan fisik (Collision NodeMCU - Komputer)

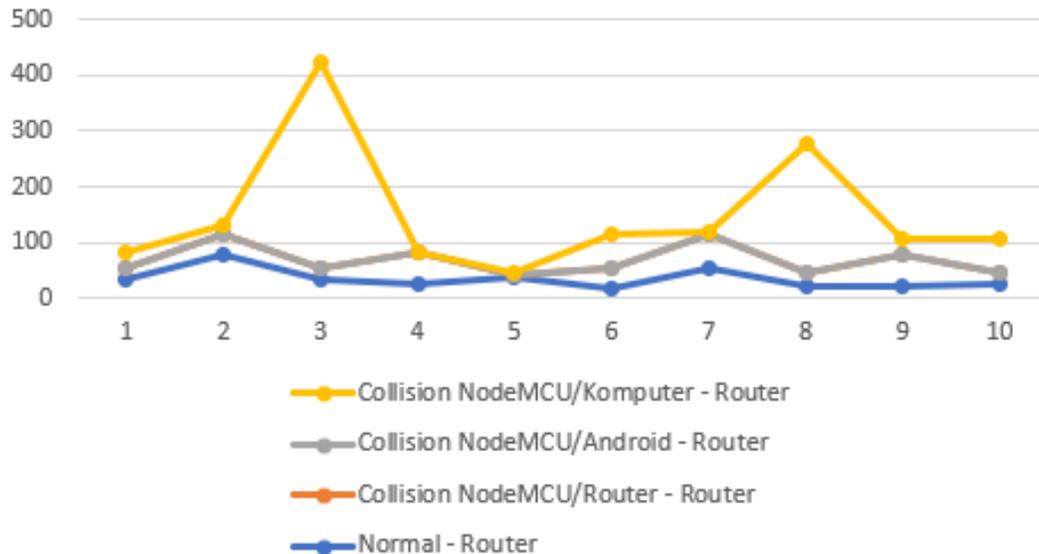
MAC Variabel bebas (NodeMCU)	MAC Komputer	MAC Android	MAC router	avg Ping NodeMCU ke Router (ms)	avg Ping komputer ke Router (ms)	avg Ping Android ke Router (ms)		MAC Variabel bebas (NodeMCU)	MAC Komputer	MAC Android	MAC router	avg Ping NodeMCU ke Komputer (ms)	avg Ping komputer ke Komputer (ms)	avg Ping Android ke Komputer (ms)
84:F3:EB:CC:03:EF	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	8	1	30		84:F3:EB:CC:03:EF	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	17	0	28
34:13:E8:C6:4C:42	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	RTO	1	83		34:13:E8:C6:4C:42	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	RTO	0	153
MAC Variabel bebas (NodeMCU)	MAC Komputer	MAC Android	MAC router	avg Ping NodeMCU ke NodeMCU (ms)	avg Ping komputer ke NodeMCU (ms)	avg Ping Android ke NodeMCU (ms)		MAC Variabel bebas (NodeMCU)	MAC Komputer	MAC Android	MAC router	avg Ping NodeMCU ke Android (ms)	avg Ping komputer ke Android (ms)	avg Ping Android ke Android (ms)
84:F3:EB:CC:03:EF	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69					84:F3:EB:CC:03:EF	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	266	288	0
34:13:E8:C6:4C:42	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69					34:13:E8:C6:4C:42	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	RTO	RTO	0

Hasil pada lingkungan fisik (Collision NodeMCU - Komputer)

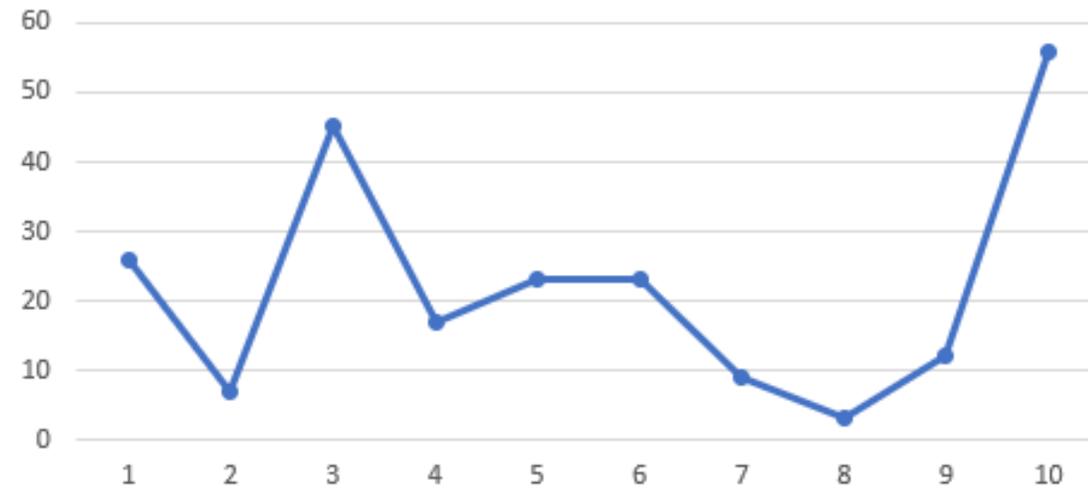
MAC Variabel bebas (NodeMCU)	MAC Komputer	MAC Android	MAC router	HTTP GET NodeMCU ke Router (ms)	HTTP GET komputer ke Router (ms)	HTTP GET Android ke Router (ms)		MAC Variabel bebas (NodeMCU)	MAC Komputer	MAC Android	MAC router	HTTP GET NodeMCU ke Komputer (ms)	HTTP GET komputer ke Komputer (ms)	HTTP GET Android ke Komputer (ms)
84:F3:EB:CC:03:EF	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	Berhasil	Berhasil	Berhasil		84:F3:EB:CC:03:EF	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	Berhasil	Berhasil	Berhasil
34:13:E8:C6:4C:42	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	Berhasil	Berhasil	Berhasil		34:13:E8:C6:4C:42	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	Gagal	Berhasil	Gagal
MAC Variabel bebas (NodeMCU)	MAC Komputer	MAC Android	MAC router	HTTP GET NodeMCU ke NodeMCU (ms)	HTTP GET komputer ke NodeMCU (ms)	HTTP GET Android ke NodeMCU (ms)		MAC Variabel bebas (NodeMCU)	MAC Komputer	MAC Android	MAC router	HTTP GET NodeMCU ke Android (ms)	HTTP GET komputer ke Android (ms)	HTTP GET Android ke Android (ms)
84:F3:EB:CC:03:EF	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	Berhasil	Berhasil	Berhasil		84:F3:EB:CC:03:EF	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	Berhasil	Berhasil	Berhasil
34:13:E8:C6:4C:42	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	Gagal	Gagal	Gagal		34:13:E8:C6:4C:42	34:13:E8:C6:4C:42	04:E5:98:32:D0:1E	50:D2:F5:2F:21:69	Gagal	Gagal	Berhasil

Statistik konektivitas ke router dari semua client ke router (data dari 3 skenario collision)

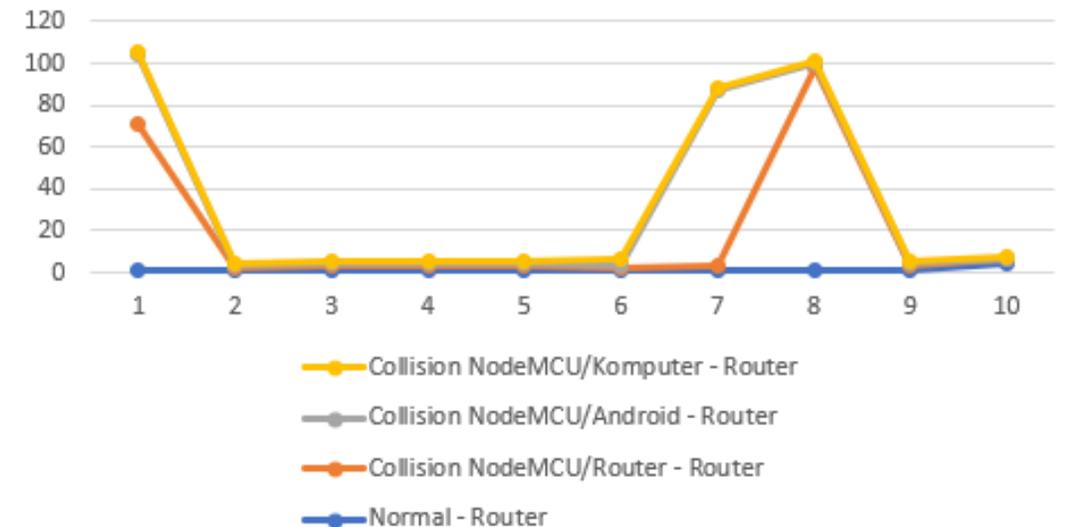
Ping dari Android ke Router



Ping NodeMCU ke router



Ping dari Komputer ke Router



1. Penelitian MAC address collision menghasilkan karakteristik anomali yang berbeda – berbeda, seperti peningkatan latency pada percobaan lingkungan virtual VirtualBox dan percobaan pada lingkungan fisik, namun pada lingkungan fisik perlu diuji kembali dengan perangkat identik.
2. Pada pengujian ping, fenomena MAC address collision dapat ditemukan dengan adanya pesan “DUP!” pada response dari ping pada sistem operasi Linux yang berjalan pada lingkungan VirtualBox, bertambahnya latency dari keadaan normal untuk penelitian baik dalam VirtualBox maupun lingkungan fisik, bahkan gagalnya ping, yang menggunakan parameter request timed out (RTO) terjadi pada lingkungan fisik.
3. Pada pengujian traceroute di VirtualBox, terjadi kenaikan nilai latency, serta beberapa kali melalui 2 buah hop, dimana pada keadaan normal seharusnya tidak terjadi.
4. Pada pengujian HTTP GET, semua perangkat pada VirtualBox berhasil menampilkan webpage kecuali pada beberapa perangkat pada penelitian collision antara 2 VM, baik melalui browser maupun cURL, namun pada lingkungan fisik hanya beberapa saja yang berhasil, dan adanya respon HTTP GET yang tidak seharusnya ditampilkan perangkat yang dituju, terutama pada saat melakukan HTTP GET dari Android ke komputer, dimana respon yang di dapat adalah respon dari NodeMCU.

1. Penelitian pada lingkungan fisik perlu diuji kembali dengan perangkat yang identik (Batasan penelitian)
2. Karena lingkungan fisik nilai latency nya lebih beragam (dilihat dari standar deviasi), maka penelitian perlu lebih dilakukan lebih dari 10 kali.
3. Jika memungkinkan, pengujian pada VirtualBox menggunakan hub, bukan switch agar VM pengamat dapat melihat payload antara VM kedua, VM ketiga, dan ke VM router (Batasan penelitian)

Lampiran



Data Penelitian	https://go.gabrielkheisa.xyz/data_penelitian_excel
Repository penelitian	https://repo.gabrielkheisa.xyz/gabrielkheisa/skripsi
Program NodeMCU	https://repo.gabrielkheisa.xyz/gabrielkheisa/skripsi/src/branch/master/source_nodemcu.cpp
IJEIS	https://go.gabrielkheisa.xyz/ijeis
Dokumentasi VirtualBox	https://go.gabrielkheisa.xyz/dokumentasi_virtualbox



UNIVERSITAS
GADJAH MADA

Terima Kasih

LOCALLY ROOTED, GLOBALLY RESPECTED

ugm.ac.id