DAFTAR PUSTAKA

[1] N. Gupta, R. Manaswini, B. Saikrishna, F. Silva, and A. Teles, “Authentication-based secure data dissemination protocol and framework for 5G-enabled VANET,” *Future Internet*, vol. 12, no. 4, p. 63, 2020.

[2] G. Xu *et al.*, “BAGKD: A batch authentication and group key distribution protocol for vanets,” *IEEE Commun. Mag.*, vol. 58, no. 7, pp. 35–41, 2020.

[3] X. Li, Y. Han, J. Gao, and J. Niu, “Secure hierarchical authentication protocol in VANET,” *IET Inf. Secur.*, vol. 14, no. 1, pp. 99–110, 2019.

[4] J. Ericka, “*Optimizing Aodv Performance In Vehicle To Vehicle Communicaton Using Static Intersection Node*,” 2016, doi: 10.13140/RG.2.1.1225.3680.

[5] C. J. Joshua, R. Duraisamy, and V. Varadarajan, “A reputation based weighted clustering protocol in VANET: A multi-objective firefly approach,” *Mob. Netw. Appl.*, vol. 24, no. 4, pp. 1199–1209, 2019.

[6] R. C. Muniyandi, F. Qamar, and A. N. Jasim, “Genetic Optimized Location Aided Routing Protocol for VANET Based on Rectangular Estimation of Position,” *Appl. Sci.*, vol. 10, no. 17, p. 5759, 2020.

[7] A. Kumar and M. Sinha, “Design and analysis of an improved AODV protocol for black hole and flooding attack in vehicular ad-hoc network (VANET),” *J. Discrete Math. Sci. Cryptogr.*, vol. 22, no. 4, pp. 453–463, 2019.

[8] K. Aravindhan and C. S. G. Dhas, “Destination-aware context-based routing protocol with hybrid soft computing cluster algorithm for VANET,” *Soft Comput.*, vol. 23, no. 8, pp. 2499–2507, 2019.

[9] M. A. Karabulut, A. S. Shah, and H. Ilhan, “OEC-MAC: A novel OFDMA based efficient cooperative MAC protocol for VANETS,” *IEEE Access*, vol. 8, pp. 94665–94677, 2020.

[10] S. D. Anggraini, K. Nugroho, and E. F. Cahyadi, “Analisis Perbandingan Performasi Protokol Routing AODV Dan DSR Pada Mobile Ad-Hoc Network,” p. 7, 2017.

[11] X. Li, Y. Han, J. Gao, and J. Niu, “Secure hierarchical authentication protocol in VANET,” *IET Inf. Secur.*, vol. 14, no. 1, pp. 99–110, 2019.

[12] N. Mohamed Noor, I. Idris, and M. Yamani, “Intersection and complement set (IACS) method to reduce redundant node in mobile WSN localization,” *Sensors*, vol. 18, no. 7, p. 2344, 2018.

[13] F. Nutrihadi, “Studi Kinerja VANET Scenario Generators: SUMO dan VanetMobisim untuk Implementasi Routing Protocol AODV menggunakan Network Simulator 2 (NS-2),” *J. Tek. ITS*, vol. 5, no. 1, Mar. 2016, doi: 10.12962/j23373539.v5i1.14307.

[14] A. Kumar and M. Sinha, “Design and analysis of an improved AODV protocol for black hole and flooding attack in vehicular ad-hoc network (VANET),” *J. Discrete Math. Sci. Cryptogr.*, vol. 22, no. 4, pp. 453–463, 2019.

[15] R. Anggoro, M. Husni, and R. Bastian, “Source route implementation using intersection node on GPSR protocol to increase VANET’s data transmission reliability,” in *2017 International Conference on Advanced Mechatronics, Intelligent Manufacture, and Industrial Automation (ICAMIMIA)*, Surabaya, Oct. 2017, pp. 356–358. doi: 10.1109/ICAMIMIA.2017.8387618.
[16] M. Ramalingam and R. Thangarajan, “Mutated k-means algorithm for dynamic clustering to perform effective and intelligent broadcasting in medical surveillance using selective reliable broadcast protocol in VANET,” *Comput. Commun.*, vol. 150, pp. 563–568, 2020.
[17] G. Sun, Y. Zhang, H. Yu, X. Du, and M. Guizani, “Intersection fog-based distributed routing for V2V communication in urban vehicular ad hoc networks,” *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 6, pp. 2409–2426, 2019.
[18] N. Mohamed Noor, I. Idris, and M. Yamani, “Intersection and complement set (IACS) method to reduce redundant node in mobile WSN localization,” *Sensors*, vol. 18, no. 7, p. 2344, 2018.
[19] Centerklik, “Inilah Kecepatan Jaringan Internet 5G Dalam Angka,” *Centerklik™*, Sep. 26, 2019. https://www.centerklik.com/kecepatan-internet-5g/ (accessed Feb. 26, 2021).

LAMPIRAN – LAMPIRAN

Kode program untuk membuat mobilitas

```
netconvert --osm-files map.osm -o map.net.xml --no-turnarounds
python ../tools/randomTrips.py -n map.net.xml -e 1 -o trip.trips.xml -p 0.04
duarouter -n map.net.xml -t trip.trips.xml -o route.rou.xml -s 500 --remove-loops --repair --ignore-errors
sumo -c skenario.sumo.cfg --fcd-output skenario.xml
python ../tools/traceExporter.py --fcd-input skenario.xml --ns2mobility-output mobility.tcl
```

Kode program hasil mobilitas (mobility.tcl)

```
$node_(0) set X_ 184.77
$node_(0) set Y_ 501.99
$node_(0) set Z_ 0
$ns_ at 0.0 "$node_(0) setdest 184.77 501.99 0.00" $node_(1) set X_ 151.29
$node_(1) set Y_ 200.27
$node_(1) set Z_ 0
$ns_ at 0.0 "$node_(1) setdest 151.29 200.27 0.00"
$ns_ at 1.0 "$node_(0) setdest 184.06 500.73 1.44"
$ns_ at 1.0 "$node_(1) setdest 149.81 201.07 1.67"
$node_(2) set X_ 327.55
$node_(2) set Y_ 159.34
$node_(2) set Z_ 0
$ns_ at 1.0 "$node_(2) setdest 327.55 159.34 0.00" $node_(3) set X_ 118.31
$node_(3) set Y_ 98.89
$node_(3) set Z_ 0
$ns_ at 1.0 "$node_(3) setdest 118.31 98.89 0.00" $node_(4) set X_ 382.66
$node_(4) set Y_ 393.5
$node_(4) set Z_ 0
$ns_ at 1.0 "$node_(4) setdest 382.66 393.5 0.00" $node_(5) set X_ 124.3
$node_(5) set Y_ 150.47
$node_(5) set Z_ 0
$ns_ at 1.0 "$node_(5) setdest 124.3 150.47 0.00" $node_(6) set X_ 109.1
$node_(6) set Y_ 368.02
$node_(6) set Z_ 0
$ns_ at 1.0 "$node_(6) setdest 109.1 368.02 0.00" $node_(7) set X_ 123.54
$node_(7) set Y_ 369.7
$node_(7) set Z_ 0
$ns_ at 1.0 "$node_(7) setdest 123.54 369.7 0.00" $node_(8) set X_ 212.93
$node_(8) set Y_ 324.65
$node_(8) set Z_ 0
```

Kode program koordinat Static Intersection Node

```
#RSU
$node_(52) set X_ 368.83
$node_(52) set Y_ 235.99
$node_(52) set Z_ 0
$node_(52) color "blue"
$ns_ at 0.0 "$node_(52) setdest 368.83 235.99 0"
$ns_ at 0.0 "$node_(52) color blue"
$node_(53) set X_ 191.03
$node_(53) set Y_ 509.71
$node_(53) set Z_ 0
$node_(53) color "blue"
$ns_ at 0.0 "$node_(53) setdest 191.03 509.71 0"
$ns_ at 0.0 "$node_(53) color blue"
$node_(54) set X_ 176.75
$node_(54) set Y_ 67.72
$node_(54) set Z_ 0
$node_(54) color "blue"
$ns_ at 0.0 "$node_(54) setdest 176.75 67.72 0"
$ns_ at 0.0 "$node_(54) color blue"
$node_(55) set X_ 110.70
$node_(55) set Y_ 105.24
$node_(55) set Z_ 0
$node_(55) color "blue"
$ns_ at 0.0 "$node_(55) setdest 110.70 105.24 0"
$ns_ at 0.0 "$node_(55) color blue"
$node_(56) set X_ 152.98
$node_(56) set Y_ 442.34
$node_(56) set Z_ 0
$node_(56) color "blue"
$ns_ at 0.0 "$node_(56) setdest 152.98 442.34 0"
$ns_ at 0.0 "$node_(56) color blue"
$node_(57) set X_ 445.65
$node_(57) set Y_ 273.31
$node_(57) set Z_ 0
$node_(57) color "blue"
$ns_ at 0.0 "$node_(57) setdest 445.65 273.31 0"
$ns_ at 0.0 "$node_(57) color blue"
$node_(58) set X_ 352.36
$node_(58) set Y_ 336.58
$node_(58) set Z_ 0
$node_(58) color "blue"
$ns_ at 0.0 "$node_(58) setdest 352.36 336.58 0"
$ns_ at 0.0 "$node_(58) color blue"
$node_(59) set X_ 408.53
$node_(59) set Y_ 207.94
$node_(59) set Z_ 0
$node_(59) color "blue"
$ns_ at 0.0 "$node_(59) setdest 408.53 207.94 0"
$ns_ at 0.0 "$node_(59) color blue"
$node_(60) set X_ 45.92
$node_(60) set Y_ 253.30
$node_(60) set Z_ 0
$node_(60) color "blue"
```

Kode program untuk mengirimkan paket data

```
#preset
set val(chan)    Channel/WirelessChannel
set val(prop)    Propagation/TwoRayGround
set val(netif)   Phy/WirelessPhy
set val(mac)     Mac/802_11
set val(ifq)     Queue/DropTail/PriQueue set val(ll) LL
set val(ant)     Antenna/OmniAntenna
set val(ifqlen)  50
set val(nn)      150
set val(rp)      AODV
set val(x)       600
set val(y)       600
set val(stop)    100

#simulatornya
set ns_ [new Simulator]

set f [open 100.tr w]
$ns_ trace-all $f
set nf [open 100.nam w]
$ns_ namtrace-all-wireless $nf  $val(x)  $val(y)

$ns_ use-newtrace

# set up topography object
set topo       [new Topography]

$topo load_flatgrid $val(x) $val(y)

#
# Create God
#

create-god $val(nn)

set chan_1_ [new $val(chan)]

$ns_ node-config   -adhocRouting $val(rp) \
                   -llType $val(ll) \
                   -macType $val(mac) \
                   -ifqType $val(ifq) \
                   -ifqLen $val(ifqlen) \
                   -antType $val(ant) \
                   -propType $val(prop) \
                   -phyType $val(netif) \
                    -channelType $val(chan) \
                   -topoInstance $topo \
                   -agentTrace ON \
                   -routerTrace ON \
                   -macTrace ON \
                   -movementTrace ON \
```

```
#limiting wifi range
Phy/WirelessPhy set CPThresh_ 10.0
Phy/WirelessPhy set CSThresh_ 1.42681e-08 ;#100m
Phy/WirelessPhy set RXThresh_ 1.42681e-08 ;#100m
Phy/WirelessPhy set bandwidth_ 10M
Phy/WirelessPhy set Pt_ 0.281838
Phy/WirelessPhy set freq_ 9.14e+08
Phy/WirelessPhy set L_ 1

#set nodes
for {set i 0} {$i < $val(nn) } { incr i } {
        set node_($i) [$ns_ node]}

#initial position
for {set i 0} {$i < $val(nn) } { incr i } {
        $ns_ initial_node_pos $node_($i) 20}

# load mobility
source "../mobility/100/mobility.tcl"

#load sin position
#source "../sin/djsin.tcl"

#sender position
$node_(50) set X_ 301
$node_(50) set Y_ 2
$node_(50) set Z_ 0
$ns_ at 0.0 "$node_(50) setdest 301 2 0"
$node_(50) color "red"
$ns_ at 0.0 "$node_(50) color red"

#receiver position
$node_(51) set X_ 216.48
$node_(51) set Y_ 554.10
$node_(51) set Z_ 0
$ns_ at 0.0 "$node_(51)
$node_(51) color "red"
$ns_ at 0.0 "$node_(51) color red"

# Set a UDP connection
set UDP [new Agent/UDP]
set null [new Agent/Null]
$ns_ attach-agent $node_(50) $UDP
$ns_ attach-agent $node_(51) $null
$ns_ connect $UDP $null
$UDP set fid_ 2

# Set a CBR over UDP
set cbr [new Application/Traffic/CBR]
$cbr attach-agent $UDP
$cbr set type_ cbr
$cbr set packetSize_ 100
$cbr set interval_ 1
$ns_ at 0.0 "$cbr start"
```

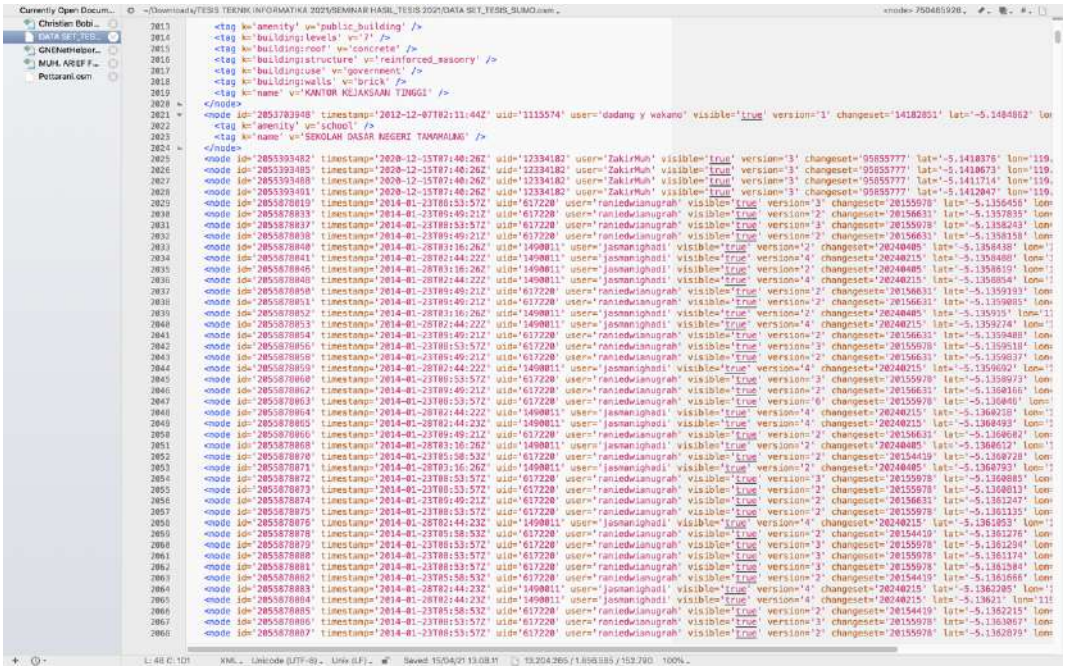
Kode program untuk memeriksa hasil pengiriman data

```
BEGIN {
print("\n\n******** Network Statistics ********\n");

# Change array size from 50 to any number of nodes for which u are doing simulation.
# i.e. change values of arrays packet_sent, packet_drop, packet_recvd,
packet_forwarded, energy_left,
packet_sent[133] = 100;
packet_drop[133] = 100;
packet_recvd[133] = 100;
packet_forwarded[133] = 100;

# Change energy assigned to initial node (as per your simulation tcl file)
# Initial Energy assigned to each node in Joules

energy_left[133] = 100;

total_pkt_sent=0;
total_pkt_recvd=0;
total_pkt_drop=0;
total_pkt_forwarded=0;
pkt_delivery_ratio = 0;
total_hop_count = 0;
avg_hop_count = 0;
overhead = 0;
start = 0.000000000;
end = 0.000000000;
packet_duration = 0.0000000000;
recvnum = 0;
delay = 0.000000000;
sum = 0.000000000;
i=0;
total_energy_consumed = 0.000000;
}

{
state	=	$1;
time	=	$3;

# For energy consumption statistics see trace file
node_num=$5;
energy_level=$7;

node_id=$9;
level=$19;
pkt_type=$35;
packet_id=$41;
no_of_forwards=$49;

# In for loop change values from 50 to number of nodes that u specify for your simulation
if((pkt_type == "cbr") && (state == "s") && (level=="AGT")) {
for(i=0;i<133;i++) {
	if(i == node_id) {
```

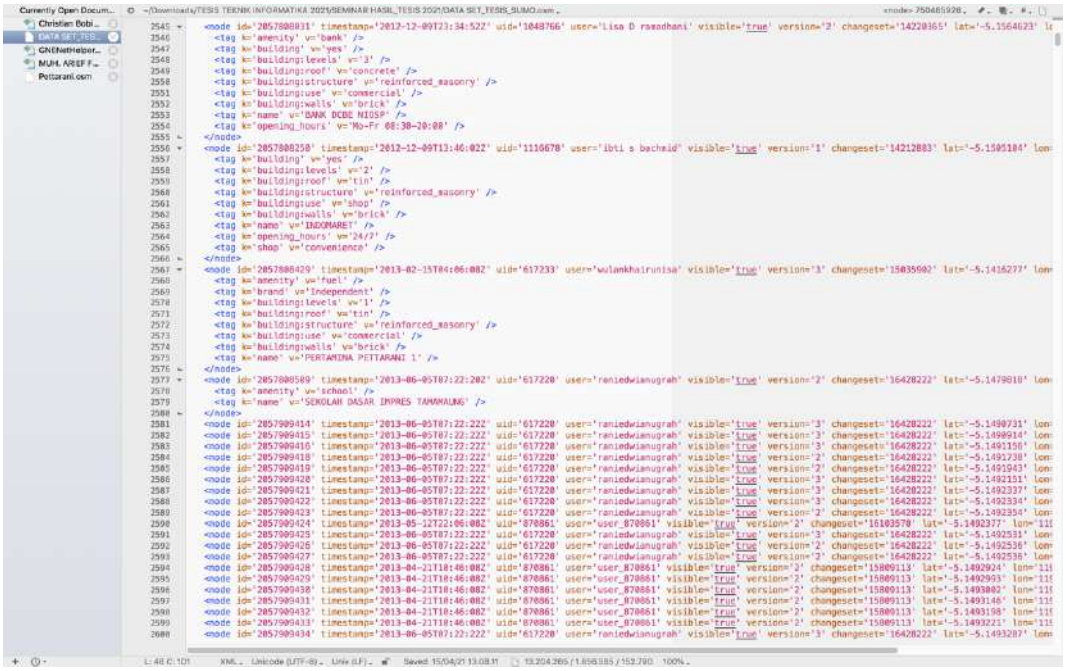
```
        packet_sent[i] = packet_sent[i] + 1; }

}
}else if((pkt_type == "cbr") && (state == "r") && (level=="AGT")) { for(i=0;i<133;i++) {
     if(i == node_id) {
     packet_recvd[i] = packet_recvd[i] + 1; }

}
}else if((pkt_type == "cbr") && (state == "d")) {
        for(i=0;i<133;i++) {
             if(i == node_id) {
             packet_drop[i] = packet_drop[i] + 1; }
        }
}else if((pkt_type == "cbr") && (state == "f")) {
        for(i=0;i<133;i++) {
              if(i == node_id) {
              packet_forwarded[i] = packet_forwarded[i] + 1; }
}
}

# To calculate total hop counts
if ((state == "r") && (level == "RTR") && (pkt_type == "cbr")) { total_hop_count
= total_hop_count + no_of_forwards; }


# Routing Overhead
if ((state == "s" || state == "f") && (level == "RTR") && (pkt_type == "message")) {
overhead = overhead + 1; }

# Calculating Average End to End Delay

if ( start_time[packet_id] == 0 ) { start_time[packet_id] = time; }

if (( state == "s") && ( pkt_type == "cbr" ) && ( level == "AGT" )) { start_time[packet_id] =
time; }

if (( state == "r") && ( pkt_type == "cbr" ) && ( level == "AGT" )) { end_time[packet_id] =
time; }

 else {  end_time[packet_id] = -1;  }

# To Calculate Average Energy Consumption

# Change number of nodes in this for loop also

if(state == "N") {
     for(i=0;i<133;i++) {
          if(i == node_num) {
                    energy_left[i] = energy_left[i] - (energy_left[i] - energy_level);
               }
}
}
```

Dataset untuk Simulasi memakai SUMO Iterasi 1

Dataset untuk Simulasi memakai SUMO Iterasi 2



Dataset untuk Simulasi memakai SUMO Iterasi 3



Dataset untuk Simulasi memakai SUMO Iterasi 4

Data tabel hasil Simulasi memakai SUMO

ROAD	MOBILITY 1	DENSITY 1	ROAD	MOBILITY 1	DENSITY 1
1	5	0,573	27	7	1,518
2	3	1,285	28	8	0,358
3	3	0,647	29	6	1,442
4	5	0,179	30	16	0,228
5	10	0,208	31	3	0,422
6	12	0,280	32	3	0,244
7	5	0,277	33	5	0,274
8	5	0,934	34	5	0,617
9	3	0,650	35	7	0,214
10	2	0,584	36	6	0,908
11	10	0,326	37	7	0,763
12	7	1,554	38	5	0,576
13	9	0,388	39	5	0,465
14	3	0,713	40	2	0,939
15	6	0,308	41	10	0,733
16	3	0,856	42	6	4,324
17	6	0,420	43	6	0,702
18	3	0,363	44	5	0,227
19	4	0,484	45	4	1,654
20	3	0.0044	46	7	0,664
21	3	0,113	47	1	0,632
22	6	0,277	48	2	0,363
23	6	0,458	49	4	0,381
24	17	0,388	50	7	0,149
25	10	0,536	51	6	0,481
26	14	0,761	52	9	0,447

ROAD	MOBILITY 2	DENSITY 2	ROAD	MOBILITY 2	DENSITY 2
1	2	0,433	27	7	1,819
2	4	0,285	28	7	0,569
3	2	1,008	29	5	0,492
4	4	0,178	30	6	0,291
5	12	0,435	31	1	0,889
6	8	0,191	32	5	0,448
7	12	0,285	33	3	0,160
8	10	0,123	34	11	0,562
9	18	0.008	35	10	1,951
10	3	0,416	36	4	0,215
11	9	0,093	37	5	1,587
12	8	0,166	38	5	0,373
13	2	0,187	39	5	0,457
14	2	1,108	40	7	1,080
15	12	0,876	41	2	0,143
16	3	0,544	42	4	0,380
17	1	1,749	43	3	0,281
18	3	0,556	44	4	0,147
19	4	0,718	45	7	0,621
20	1	0,208	46	23	0,782
21	6	0,725	47	8	0,610
22	9	0,208	48	18	5,851
23	3	0,498	49	11	0,756
24	3	0,428	50	6	1,068
25	3	0,274	51	7	1,344
26	2	0,166	52	5	0,378

ROAD	MOBILITY 3	DENSITY 3	ROAD	MOBILITY 3	DENSITY 3
1	7	0,247	27	3	1,299
2	6	0,538	28	3	0,756
3	9	0,623	29	4	1,068
4	6	1,088	30	5	0,224
5	2	0,484	31	9	0,802
6	4	0,549	32	6	0,671
7	5	0,191	33	3	0,621
8	6	0,857	34	9	0,939
9	10	0,831	35	10	1,099
10	9	1,022	36	7	0,210
11	14	0,490	37	4	1,410
12	8	1,554	38	4	0,215
13	2	0,374	39	3	4,070
14	12	0,277	40	4	0,769
15	9	0,560	41	9	0,455
16	16	0,416	42	7	0,210
17	6	0,332	43	3	0,304
18	9	0,285	44	6	0,281
19	4	1,083	45	9	0,654
20	1	1,142	46	16	0,889
21	1	0,168	47	3	0,549
22	6	0.0044	48	13	0,617
23	3	0.0081	49	6	0,214
24	8	0,499	50	12	0,244
25	2	0,268	51	4	1,985
26	6	0,869	52	5	0,285

ROAD	MOBILITY 4	DENSITY 4	ROAD	MOBILITY 4	DENSITY 4
1	8	0,222	27	10	0,508
2	3	1,426	28	11	0,635
3	3	0,679	29	4	1,843
4	5	0,657	30	9	0,358
5	18	0,435	31	8	1,153
6	5	1,554	32	16	0,378
7	14	0,185	33	12	0,224
8	9	0,226	34	9	0,802
9	3	0,416	35	4	0,469
10	5	0,816	36	4	0,610
11	1	0,276	37	9	4,578
12	5	0,092	38	5	0,351
13	6	0,359	39	6	0,563
14	6	0,485	40	4	0,440
15	5	0,280	41	6	0,357
16	17	0,669	42	4	0,931
17	5	0,715	43	7	1,324
18	13	1,201	44	5	0,285
19	5	0,975	45	13	1,875
20	4	0,713	46	7	1,059
21	3	1,008	47	3	1,374
22	7	0,333	48	5	0,298
23	5	0,447	49	1	0,274
24	3	0,435	50	6	0,154
25	8	0,277	51	5	0,210
26	9	0,374	52	4	0,380

Data tabel hasil Simulasi memakai SUMO dilihat dari Jarak

ROAD	LENGTH (m)	ROAD	LENGTH (m)	ROAD	LENGTH (m)
1	1270	19	172	37	273
2	798	20	148	38	198
3	630	21	275	39	291
4	594	22	195	40	298
5	622	23	227	41	142
6	128	24	271	42	195
7	579	25	398	43	254
8	282	26	198	44	293
9	614	27	364	45	243
10	1465	28	296	46	193
11	1293	29	248	47	275
12	630	30	291	48	345
13	1173	31	245	49	297
14	902	32	136	50	344
15	989	33	342	51	244
16	798	34	285	52	356
17	177	35	340	53	322
18	163	36	292	54	243

Tabel Perbandingan Protokol Routing AODV dan DSR

Parameter	Tanpa Serangan		Black Hole		Collaborative Black hole	
	AODV	DSR	AODV	DSR	AODV	DSR
Throughput (kbit/s)	216.146	73.423	207.332	70.180	197.909	64.755
Delay (m/s)	0.719	2.246	0.623	3.072	0.718	2.521
Packet Loss (%)	0.15%	0.01%	1.72%	2.16%	1.88	6.06%