BAB II
LANDASAN TEORI

2.1 Studi Literatur

Pada penelitian yang telah dilakukan sebelumnya membahas mengenai VANET antara lain oleh (Raharjo, 2017), memiliki tujuan penelitian untuk mengetahui performa dari *routing protocol* AODV dengan parameter 802.11p dalam lingkungan VANET pada skenario grid dan skenario riil. Pada penelitian tersebut menggunakan *Network Simulato*r 2 (NS-2) dan *Simulation of Urban Mobility* (SUMO) dengan parameter QoS yang dilakukan analisa adalah *Routing Overhead* (RO), *Average End to End Delay*, *Packet Delivery Ratio* (PDR). Skenario yang digunakan adalah dengan melakukan penambahan jumlah node 50, 100, 150 dan 200 pada skenario grid dan skenario riil. Hasil yang didapatkan dari penelitian ini menunjukan bahwa seiring dengan tingkat kepadatan yang meningkat pada kedua skenario nilai PDR dari *protocol* AODV meningkat cukup stabil dari tingkat kepadatan yang minimal dan semakin naik pada kepadatan yang maksimal. Pada *Average End to End Delay* dengan tingkat kepadatan yang rendah, nilai *Delay* masih tidak stabil, namun seiring bertambahnya kepadatan node nilai *Delay* semakin naik hingga kepadatan yang maksimal. Pada *Routing Overhead* (RO) memiliki kondisi awal yang cukup stabil dengan tingkat kepadatan yang masih rendah dan semakin meningkat seiring bertambahnya tingkat kepadatan jumlah node. Kelebihan dari penelitian ini adalah menggunakan skenario grid dan skenario riil. Kekurangan dari penelitian ini adalah hanya mengetahui performansi *routing protocol* AODV dan menggunakan kecepatan yang statis serta tidak menghitung parameter nilai *Throughput.*

Pada penelitian lain yang dilakukan oleh (Basil, Ismail, Altahrawi, Mahdi, & Ramli, 2017) memiliki tujuan penelitian untuk mengetahui performa dari *routing protocol* AODV dan OLSR dalam lingkungan VANET dengan skenario pada persimpangan lalu lintas. Pada penelitian tersebut menggunakan *Network Simulator 3* (NS-3) dan *Simulation of Urban Mobility* (SUMO) dengan parameter QoS yang dilakukan analisa adalah *Throughput, Average End to End Delay, Packet Delivery Ratio* (PDR). Skenario yang digunakan adalah melakukan penambahan jumlah node 18 dan 72 dengan kecepatan rata-rata 40 Km/h pada area persimpangan lalu lintas. Hasil yang didapatkan dari penelitian ini pada tingkat kepadatan rendah dan tingkat kepadatan yang tinggi adalah nilai *Throughput* dan *Average End to End Delay* dari kedua *routing protocol* semakin tinggi tingkat kepadatan semakin meningkat nilai

QoS tersebut, sedangkan nilai PDR menurun. Sehingga didapatkan hasil kinerja *protocol* OLSR lebih unggul daripada *protocol* AODV. Kelebihan dari penelitian ini adalah membandingkan *routing protocol* OLSR dan AODV serta menggunakan skenario persimpangan lalu lintas. Kekurangan dari penelitian ini adalah kecepatan yang digunakan rata-rata 40 Km/h dan hanya pada skenario persimpangan lalu lintas serta penggunaan jumlah node kurang variatif.

Pada penelitian yang diusulkan oleh peneliti adalah membandingkan performansi *routing protocol* OSLR dan AODV dalam lingkungan VANET dengan parameter QoS yang dilakukan analisa yaitu *Throughput, Packet Delivery Ratio* (PDR), *Packet Loss Ratio* (PLR), *Delay* dan *Jitter*. Perbedaan dari penelitian sebelumnya yaitu pada skenario penambahan jumlah node 30, 50 dan 80, dimana suatu jaringan VANET mengalami tingkat kepadatan node yang bertambah pada wilayah 1 km² dalam mentransmisikan paket data dari node sumber ke node tujuan berdasarkan rute yang dilalui. Untuk pergerakan semua node bersifat *random* dengan kecepatan bervariasi antara 0 m/s sampai 20 m/s. Dengan asumsi untuk pergerakan semua node terdapat node diam dan node yang bergerak dengan pergerakan bebas.

Sedangkan pada skenario peningkatan kecepatan node, dimana terjadi peningkatan kecepatan node pada jaringan VANET yang bertujuan untuk mengetahui tingkat keberhasilan node dalam menerima paket data berdasarkan kecepatan yang bervariasi. Untuk kecepatan yang digunakan 10 m/s, 15 m/s dan 20 m/s bergerak secara *random* dengan jumlah node 35 dan kecepatan tetap serta wilayah yang digunakan seluas 1 km². Dengan asumsi untuk pergerakan semua node sama atau tetap denga pergerakan bebas.

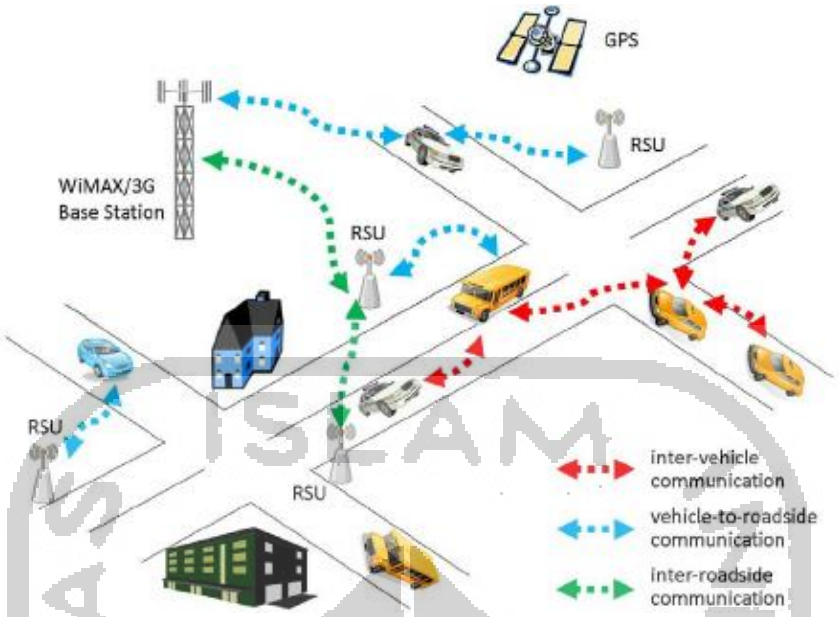
Adapun tabel perbandingan parameter dari penelitian sebelumnya dapat dilihat pada Tabel 2.1.

Tabel 2.1 Perbandingan Parameter Penelitian

Parameter	**Penelitian** (Raharjo, 2017)	**Penelitian** (Basil, Ismail, Altahrawi, Mahdi, & Ramli, 2017)	**Penelitian Ini**
Network Simulator	NS-2	NS-3	NS-3
Routing Protocol	AODV	OLSR, AODV	OLSR, AODV
Waktu Simulasi	360 detik	80 detik	100 detik
Area Simulasi	1,2 km² (grid), 1,2 km x 0,8 km (riil)	0,9 km²	1 km²
Jumlah Node	50, 100, 150, 200	18, 72	30, 50, 80 atau 35
Kecepatan	15 m/s	11,1 m/s	0-20 m/s *random* atau 10 m/s, 15 m/s, 20 m/s *constant*
Data *Type*	UDP	-	UDP
Packet Size	512 *bytes*	-	512 *bytes*
Transmission Range	50 m	-	300 m
Radio Propagation	*Two-Ray Ground*	-	*Threelogdistance*

2.2 *Vehicular Ad Hoc Network* (VANET)

Vehicular Ad Hoc Network (VANET) merupakan salah satu bagian dari teknologi *Intelligent Transportation System* (ITS) yang dikembangkan untuk mendukung sistem transportasi dengan memanfaatkan teknologi komunikasi jaringan nirkabel dan infrastruktur transportasi. Sebagai bagian dari ITS, komunikasi kendaraan antar kendaraan dan dengan keadaan sekitar dalam VANET dapat lebih efektif untuk menghindari kecelakaan dan kemacetan lalu lintas dari pada memecahkan masalah tersebut secara individual pada setiap kendaraan (Perdana, 2015). VANET merupakan adaptasi dan perkembangan dari teknologi *Mobile Ad Hoc Network* (MANET) yang membangun *ad hoc network* antara kendaraan sehingga dapat mendukung komunikasi *Vehicle to Vehicle* (V2V), *Vehicle to Infrastruktur* (V2I), *Vehicle to Roadside* (V2R), dan *Hybrid models* (Chaudhry, Seth, & Sharma, 2014). Dalam VANET kendaraan atau node dapat berlaku sebagai *router* atau *client* yang bergerak bebas sehingga dapat saling berkomunikasi membentuk jaringan *one-hop* atau *multi-hop* untuk mengirimkan data menuju node lain. Seperti terlihat pada Gambar 2.1 merupakan ilustrasi dari jaringan VANET.



Gambar 2.1 Ilustrasi Jaringan VANET

Sumber: Raharjo (2017)

Pada *Vehicle to Vehicle* (V2V) atau jaringan *ad hoc* antar kendaraan, seperti mobil, bus, truk dan lainnya yang dapat saling terhubung sehingga memungkinkan untuk berkomunikasi dengan mengirimkan data ke masing-masing kendaraan. Menggunakan komunikasi V2V, sebuah kendaraan dapat mendeteksi posisi dan pergerakan kendaraan lain karena kendaraan tersebut dilengkapi dengan antena, *chip computer* dan teknologi GPS. Dengan cara tersebut kendaraan dapat berbagi informasi mengenai posisi kendaraan, titik buta kendaraan, kecelakaan dan kondisi jalan. Dengan demikian kendaraan dapat mengantisipasi dan bereaksi terhadap situasi mengemudi yang berpotensi berbahaya serta akan memberi tahu pengemudi mengenai kondisi tersebut. Apablia pengemudi tidak menanggapi peringatan, kendaraan dapat bertindak sendiri dan berhenti pada titik aman untuk menghindari terjadi tabrakan (Profentzas, 2012).

Setiap node atau kendaraan tersebut pada jaringan VANET dilengkapi dengan sensor *On Board Units* (OBU) yang terdapat didalam kendaraan dan infrastruktur jalanan atau *Road Side Unit* (RSU). OBU tersebut digunakan untuk mengumpulkan data informasi seperti informasi *routing, traffic control centres, Global Positioning System* (GPS), *weather centres* dan lainnya yang dapat ditampilkan pada pengendara serta dapat di-*broadcast* ke kendaraan lain (Training, 2016). Sehingga dalam layanan kemanan pada VANET dapat membimbing pengendara untuk berkomunikasi dan mengatur pergerakan guna terhindar dari kecelakaan lalu lintas, kemacetan, kontrol kecepatan dan lainnya (Training, 2016).

Sehubungan dengan itu, kebutuhan akan keamanan pada kendaraan menjadi masalah penting yang akhirnya mengarah kepada keselamatan dalam berkendara (Chaudhry, Seth, & Sharma, 2014). Dengan penerapan teknologi VANET pada kendaraan memungkinkan untuk berkomunikasi satu sama lain yang dapat berbagi data dan menyesuaikan tindakan untuk menghindari kecelakaan (Chaudhry, Seth, & Sharma, 2014). Sehingga proses komunikasi antar kendaraan dalam pertukaran data harus efisien dan terjamin tingkat keamanan karena pesan yang ditransmisikan dapat berlangsung sangat cepat (Toulni & Nsiri, 2015). Karena pada dasarnya inti dari desain VANET yaitu untuk meningkatkan efisiensi keselamatan dan keamanan dalam berkendara, memberikan layanan informasi, atau kondisi ketidakpastian berkendara dan lalu lintas serta layanan non keamanan seperti layanan entertainment, video, audio dan lain-lain (Toulni & Nsiri, 2015).

2.2.1 Aplikasi VANET

Komunikasi VANET dapat mendukung komunikasi *Vehicle to Vehicle* (V2V), *Vehicle to Infrastruktur* (V2I), *Vehicle to Roadside* (V2R), dan *Hybrid models* (Chaudhry, Seth, & Sharma, 2014). Dalam komunikasi tersebut dapat digunakan untuk sejumlah aplikasi potensial dengan beberapa persyaratan yang beragam. Tiga kelas utama aplikasi yang memungkinkan diterapkan dalam VANET adalah berorientasi pada keselamatan, kenyamanan dan komersial (Kamini & Kumar, 2010). Aplikasi yang berorientasi pada keselamatan akan memantau jalan sekitar, pendekatan kendaraan, permukaan dan kurva jalan (Kamini & Kumar, 2010). Pada aplikasi yang berorientasi kenyamanan yaitu dari jenis manajemen lalu lintas jalan dan aplikasi yang berkaitan dengan komersial yaitu layanan streaming, video, dan audio. Menurut (Kamini & Kumar, 2010) ada beberapa hal yang dapat merepresentasikan aplikasi VANET yaitu:

- *Traffic Signal*

Traffic signal yaitu komunikasi dari lampu lalu lintas yang terhubung dalam jaringan VANET sehingga dapat menyiarkan pesan peringatan apabila mendeteksi kemacetan atau kendaraan tidak bergerak untuk pencarian dan perencanaan rute perjalanan.

- *Vision Enhancement*

Vision enhacement yaitu peningkatan penglihatan bagi pengemudi mengenai pandangan yang jelas tentang keberadaan kendaraan disekitarnya dan dalam kondisi rintangan, seperti, kabut tebal, hujan dan lainya.

- *Driver Assistance*

Driver assistance yaitu bantuan untuk pengemudi apabila terdapat kerusakan kendaraan dijalan sedangkan pada kendaraan tersebut tidak mempunyai suku cadang, peralatan otomotif atau mencari tempat servis kendaraan terdekat dengan mengirimkan pesan ke kendaraan lain untuk meminta bantuan.

- *Safety*

Safety yaitu layanan aplikasi keselamatan mengenai peringatan tabrakan, mendeteksi dan menghindari kendaraan yang tiba-tiba mengerem, berbelok arah secara mendadak, peringatan lajur berhenti dan peringatan kondisi jalan.

- *Emergency*

Emergency yaitu layanan penyebaran pesan darurat kepada kendaraan lain untuk memberikan otoritas penggunaan jalan, seperti mobil ambulan, mobil pemadam kebakaran dan lainnya.

2.2.2 Karakteristik VANET

Jaringan VANET memiliki beberapa karakterisktik yang membedakan dari jaringan *ad hoc* lainya, yaitu (Training, 2016):

- Topologi Dinamis

Topologi VANET yang sangat dinamis karena pergerakan kendaraan yang sangat cepat. Selain itu dengan pergerakan yang cepat saat bertukar informasi akan membuat putusnya koneksi secara berkala.

- Model Mobilitas Dan Prediksi

Model mobilitas node tergantung pada kondisi lalu lintas, struktur jalan, kecepatan kendaraan, perilaku pengemudi dalam berkendara, dan lainnya.

- *Hard Delay Constrains*

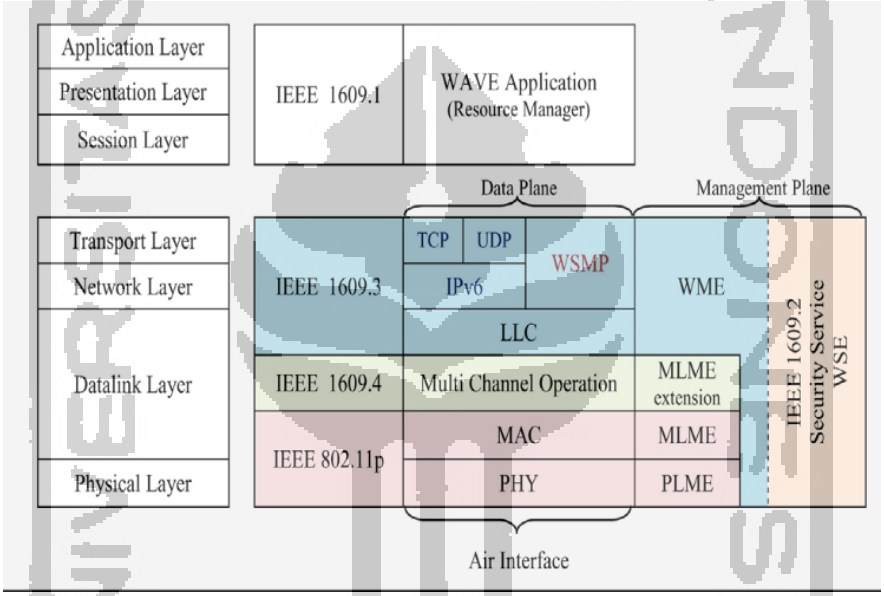
Hard delay constrains yaitu mengenai aspek keamanan dari aplikasi VANET terkait dengan kecelakaan, masalah rem, dan lainnya harus menjamin pengiriman pesan ke node lain harus relevan.

- Interaksi Dengan Sensor *On Board*

Interaksi dengan *on board* yaitu mengenai posisi terkini dan mengetahui pergerakan node dapat dikirim dengan sensor GPS. Hal ini akan membantu dalam komunikasi dan keputusan *routing* yang lebih efektif

2.3 IEEE 802.11p

IEEE 802.11p merupakan salah satu bagian *protocol* yang terdapat pada *Wireless Access in Vehiculer Environment* (WAVE) yang dikembagkan oleh *Institute of Electrical and Electronics Engineers* (IEEE). WAVE merupakan standar pertukaran komunikasi pesan antar kendaraan yang ditetapkan sebagai pondasi untuk pengembangan dan penerapan teknologi VANET dalam membangun suatu sistem transportasi berbasis informasi dan komunikasi secara terpadu (Afdhal, 2014). Beberapa *protocol* yang terdapat dalam arsitektur WAVE meliput: IEEE 802.11p, IEEE 1609.1, IEEE 1609.2, IEEE 1609.3, dan IEEE 1609.4, seperti terlihat pada Gambar 2.2 (Afdhal, 2014).



Gambar 2.2 Arsitektur dan Kerangka Kerja WAVE

Sumber: Afdhal (2014)

Keterangan:

IEEE 802.11p sebagai standard operasi *physical layer* dan *layer* MAC.

IEEE 1609.1 sebagai standard operasi jenis aplikasi pengatur/*remote* (OBU dan RSU).

IEEE 1609.2 sebagai standard operasi layanan keamanan.

IEEE 1609.3 sebagai standard operasi penggunaan *channel.*

IEEE 1609.4 sebagai standard operasi jenis pertukaran data.

Pada *protocol* IEEE 802.11p merupakan peningkatan dari *protocol* IEEE 802.11 yang tidak mampu menangani mobilitas pada VANET karena waktu komunikasi yang singkat diakibatkan oleh kecepatan kendaraan (Yusuf & Anggoro, 2017). Peningkatan *protocol* IEEE

802.11p terjadi pada *layer data link Medium Access Control* (MAC) dan *Physical layer* (PHY) karena *protokol* IEEE 802.11 standard MAC membutuhkan waktu yang lama dalam melakukan pemindaian kanal untuk *beacon* dari *Basic Service Set* (BSS) dan melakukan banyak *handshakes* dalam membangun komunikasi (Yusuf & Anggoro, 2017). Hal tersebut yang menjadikan IEEE 802.11p digunakan sebagai standard *protocol* untuk penyediaan jaringan VANET yang mendukung komunikasi *Vehicle to Vehicle* (V2V) dan *Vehicle to Infrastructure* (V2I) dengan menggunakan teknik komunikasi jarak pendek atau *Dedicated Short Range Communication* (DSRC) (Perdana, 2015).

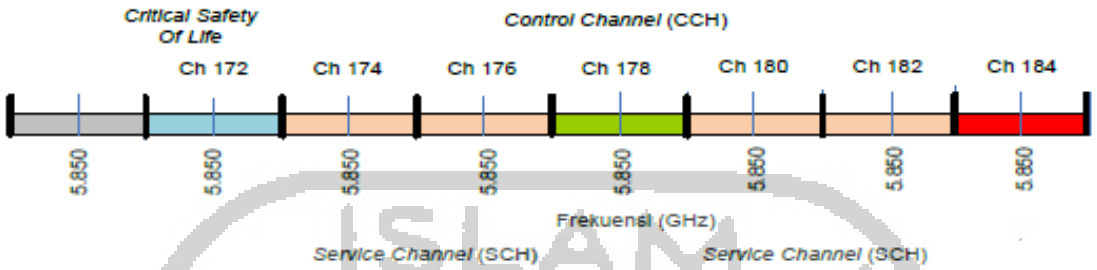
Untuk meningkatkan keamanan dan keselamatan dalam berkendara pada penggunaan spektrum DSRC, US *Federal Communication Commission* (FCC) mengalokasikan *bandwidth* 75 MHz dari spektrum DSRC yang beroperasi pada frekuensi 5.850 GHz – 5.925 GHz (Perdana, 2015). Menggunakan sistem modulasi saluran komunikasi *Orthogonal Frequency Division Multiplexing* (OFDM) dengan kecepatan transmisi data antara 6 Mbps – 27 Mbps yang dapat memberikan tingkat *Latency* rendah (Raharjo, 2017). Disamping itu, jarak komunikasi yang mendukung VANET antara 300 - 1000 m. Pada Tabel 2.2 diperlihatkan parameter standar teknologi DSRC khususnya pada VANET.

Tabel 2.2 Tabel Parameter Standar DSRC (Guo & Balon, 2006)

Parameter	**Nilai**
Spectrum	75 Mhz
Data Rate	6 Mbps – 27 Mbps
Interference Potential	Low
Coverage	*Overlapping communication zones*
Maximun Range	1000m
Minimum Separation	15.24 m
Channel Capacity	7 *channels*
Downlink Power	33 dBm
Uplink Power	33 dBm

Pada pengalokasian *bandwidth* terdiri dari 7 *channel* frekuensi yang berbeda masing-masing menggunakan *bandwidth* 10 Mhz, yaitu 1 *channel* CCH178 untuk *Control Channel* (CCH) yang digunakan sebagai layanan komunikasi sedangkan 6 *channel* lainnya dialokasikan untuk *Service Channel* (SCH) yang digunakan untuk penggunaan aplikasi *safety* dan *non safety* serta kedua *channel* yang berada diujung pita spektrum dicadangkan untuk aplikasi yang dapat

mencegah kecelakaan dan komunikasi keselamatan *public* dengan daya yang tinggi, seperti yang terlihat pada Gambar 2.3 (Perdana, 2015).



Gambar 2.3 Spektrum DSRC

Sumber: Perdana (2015)

Oleh karena itu, pengoperasian *protocol* yang terdapat dalam WAVE memungkinkan terjadi komunikasi pertukaran data antar kendaraan dengan kendaraan yang disebut *Vehicle to Vehicle* (V2V) serta antara kendaraan dengan infrastruktur atau *Vehicle to Infrastructure* (V2I) sehingga perpaduan dari jenis komunikasi tersebut dapat menggunakan jaringan nirkabel baik secara *ad hoc* ataupun infrastruktur (Afdhal, 2014).

2.4 *Routing Protocol*

Routing protocol merupakan suatu metode dalam menentukan rute terbaik dari *link* yang dilalui antar node sumber menuju node tujuan yang berhubungan. Sehingga dalam pemilihan rute terbaik dilakukan berdasarkan beberapa pertimbangan seperti jarak dan *bandwidth link* (Training, 2016). Disamping itu, proses *routing* biasanya akan mengarahkan pada penerusan paket berdasarkan *table routing* yang dapat mempertahankan catatan rute tujuan ke dalam jaringan (Islam & Bhuyan, 2015). Dalam menentukan sebuah *routing* harus mencakup beberapa hal yang perlu diperhatikan, yaitu: penentuan jalur terpendek yang akan dituju ke node tujuan harus efisien, selalu *up to date table routing* ketika terjadi perubahan topologi jaringan, dapat meminimalisir jumlah kontrol paket serta memiliki waktu konvergensi yang rendah (Chrisnamurti, 2018).

Routing dalam VANET memiliki tantangan sendiri, di antaranya yaitu: karena tingginya mobilitas node yang akan berpengaruh pada perubahan topologi jaringan yang lebih dinamis sehingga menyebabkan tingginya *Delay* dan rendahnya *Throughput* (Rezkinanda & Anggoro, 2016). Perubahan topologi yang lebih dinamis akan mempengaruhi transmisi pertukaran data yang mengakibatkan terputusnya rute dan menyebabkan *Delay* karena node keluar dari

jangkauan transmisi sinyal sehingga akan mencari rute baru untuk mentransmisikan data ke node tujuan (Rezkinanda & Anggoro, 2016) serta tingkat kepadatan node yang tidak dapat diprediksi. Hal tersebut dapat menyebabkan *Delay* tinggi dan banyaknya paket hilang yang terjadi pada VANET.

Selain itu masalah keamanan pada VANET menurut (Baqar, Aldabbas, Alwadan, Alfawair, & Janicke, 2014), ada beberapa mekanisme keamanan yang didefinisikan dalam X.800 (Stalling USA, 2005) yaitu: *Cryptography (Encipherment), Digital Signature, Access Control*, *Traffic Padding, Notarizzation dan Routing Control.* Berkaitan dengan mekanisme keamanan tersebut, yaitu salah satunya mengenai *Routing Control*. *Routing control* merupakan suatu mekanisme yang digunakan untuk memilih rute pengamanan khusus untuk data tertentu dan memungkinkan perubahan rute yang sesuai, terutama ketika terjadi ganggunan keamanan (Baqar, Aldabbas, Alwadan, Alfawair, & Janicke, 2014).

Sementara itu menurut (Kamini & Kumar, 2010), ada beberapa tantangan komunikasi dalam VANET yaitu:

- *Security*

Security yaitu menyangkut konten pesan keamanan dalam komunikasi kendaraan ke kendaraan. Isi dari pesan yang diterima harus diverifikasi dalam waktu singkat untuk dapat menggunakan informasi sesegera mungkin.

- *Authentication*

Authentication yaitu layanan yang berkaitan dengan otentifikasi dalam berkomunikasi antar kendaraan. Dengan memastikan bahwa pesan komunikasi yang terjadi antar kendaraan harus dikirimkan oleh pengirim yang sah.

- *Integrity*

Integrity yaitu layanan yang berkaitan integritas dengan stabilitas aliran pesan. Hal ini memastikan bahwa pesan yang diterima terkirim, tanpa modifikasi, penyisipan, pemesanan ulang atau *replays*.

- *Confidentiality*

Confidentiality yaitu layanan yang memberikan kerahasiaan konten dalam komunikasi. Hal ini akan menjamin privasi pengemudi terhadap pengamat yang tidak sah.

- *Accessibility*

Accessibility yaitu layanan yang memberikan ketersediaan akses terhadap komunikasi. Fitur penting dari keamanan VANET adalah *digital signature* yang digunakan sebagai *building*

block. Hal tersebut yang menjadi persyaratan dasar dalam komunikasi antar kendaraan atau dengan infrastruktur melalui otentifikasi menggunakan *digital signature*.

- *Scalability*

Scalability yaitu jumlah pengguna dan atau volume lalu lintas dapat ditingkatkan dengan penurunan kinerja yang cukup kecil atau bahkan pemadaman jaringan serta tanpa mengubah komponen sistem dan *protocol.*

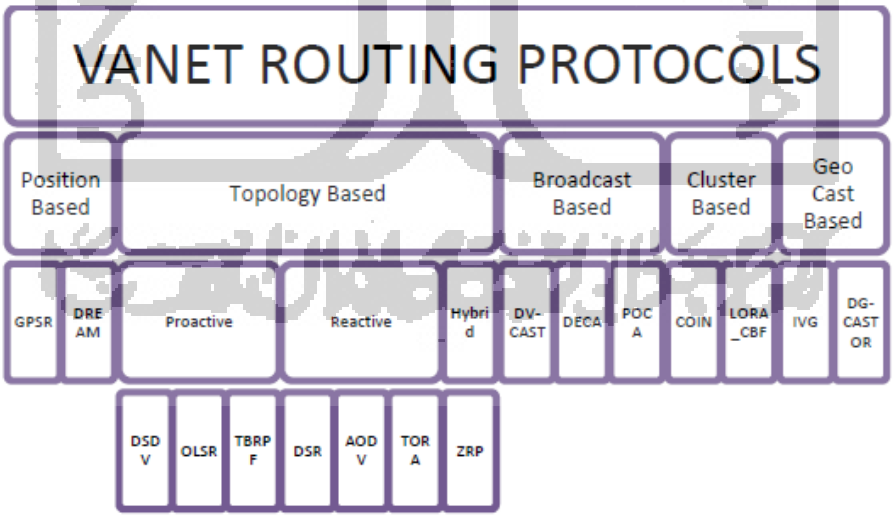
- *Reliability*

Reliability yaitu layanan yang menyangkut kehandalan terhadap komunikasi antar kendaraan. Karena waktu komunikasi yang singkat dalam VANET maka sebagian besar pesan yang dikirimkan akan berupa pesan *periodic broadcast* dengan memberitahu keadaan kendaraan di sekitarnya.

- *Media Access Control*

Media Access Control yaitu menyangkut perubahan yang harus dilakukan terhadap lapisan MAC dalam membuat jaringan VANET dalam skala besar. Tujuannya untuk mengakses media bersama menggunakan saluran nirkabel. Jika tidak ada metode yang digunakan untuk mengkoordinasikan pengiriman data, maka akan terjadi sejumlah tabrakan besar dan data yang dikirim akan hilang.

Berkaitan dengan *routing protocol*, pada jaringan VANET terdapat hirarki *routing protocol* yang dapat diklasifikasikan seperti terlihat pada Gambar 2.4.



Gambar 2.4 Hirarki *Routing Protocol*

Sumber: Chandel & Gupta (2014)

Oleh karena itu, peranan *routing* dalam sebuah jaringan merupakan hal yang sangat penting karena akan bertanggungjawab dalam memilih dan mempertahankan informasi rute serta menyalurkan paket data selama informasi rute yang telah dipilih. Disamping itu, *routing protocol* dalam VANET merupakan adaptasi dan pengembangan dari MANET yang memiliki karakteristik berbeda (Profentzas, 2012). Sehingga perlu untuk mengevaluasi dan menganalisa kinerja *routing protocol* yang terdapat dalam MANET untuk diterapkan pada VANET. Dengan menerapkan *routing protocol* yang handal dan tepat serta disesuaikan dengan kondisi lingkungan yang dinamis, transmisi pertukaran data antar kendaraan yang menyangkut layanan keselamatan dan keamanan pada VANET dapat dimininalisir akibat terputusnya rute.

2.4.1 *Protokol Optimized Link State Routing* (OLSR)

Optimized Link State Routing (OLSR) merupakan *routing protocol* yang termasuk dalam *proactive routing protocol*. *Proactive routing protocol* yaitu proses *routing* yang berusaha menyediakan informasi *routing* yang konsisten dan *up to date* pada setiap node sehingga node tersebut harus mempunyai satu atau lebih tabel untuk menyimpan informasi *routing* (Training, 2016). *Protocol* OLSR dirancang untuk jaringan *Mobile Ad Hoc Network* (MANET) dengan mengoptimalkan dari *protocol Link State Routing* (LSR) yang lama sehingga dapat menyediakan rute sesegera apabila diperlukan (Putra, Yulianto, & Herutomo, 2015). Penyediaan rute tersebut dengan teknik *multi-hop routing* sehingga dalam setiap node menggunakan informasi *routing* terbaru yang telah disimpan dalam tabel *routing* untuk mengirimkan sebuah paket informasi ke node tujuan sehingga apabila node bergerak atau berpindah maka akan tetap menerima informasi yang dikirimkan (Putra, Yulianto, & Herutomo, 2015).

Untuk mengurangi banyaknya *overhead* akibat dari informasi *routing* yang diperbarui secara berkala dalam tabel *routing*, *protocol* OLSR menggunakan mekanisme *Multi Point Relays* (MPR) (Huhtonen, 2004). Mekanisme dari MPR tersebut adalah dengan mengurangi pesan *broadcast* yang memiliki informasi sama dan mengontrol transmisi paket data yang terjadi. Sehingga dengan adanya MPR membuat paket OLSR yang diterima tidak akan langsung diteruskan ke node lain, tetapi hanya node yang dipilih sebagai MPR yang dapat meneruskan paket kontrol yang diterima (Ainurrachman, Bhawiyuga, & Ichsan, 2017). Dalam MPR terdapat node yang dipilih oleh beberapa node tetangga dengan tujuan bertanggungjawab ketika mendeklarasikan informasi *link state* dalam jaringan (Clausen & Jacquet, 2003). Berdasarkan kinerja *protocol* OLSR terdapat beberapa konsep kerja, yaitu: *link sensing,*

neighbor detection, MPR *selection*, pengiriman pesan *Topology Control*, dan *routing calculation* (Ainurrachman, Bhawiyuga, & Ichsan, 2017).

Berkaitan dengan node yang dipilih, *protocol* OLSR menggunakan dua jenis pesan kontrol yaitu *Hello* dan *Topology Control* (TC) (Huhtonen, 2004). Pesan *Hello* digunakan untuk menemukan informasi *link state* dengan mencari node tetangga satu *hop* dan tetangga dua *hop* melalui respon mereka dan memberitahu hanya tetangga satu *hop* apabila suatu node telah dipilih sebagai node MPR (Huhtonen, 2004). Sedangkan pesan *Topology Control* (TC) bersama node MPR digunakan untuk menyebarkan informasi node tetangga ke seluruh jaringan secara berkala (Huhtonen, 2004). Selain itu terdapat pesan *Multiple Interface Declaration* (MID) yang digunakan untuk menyebarkan informasi mengenai node yang menjalankan *protocol* OLSR menggunakan lebih dari satu *interface* dan pesan *Host Network Association* (HNA) yang digunakan oleh node yang bertindak sebagai *gateway* dari jaringan luar (Ainurrachman, Bhawiyuga, & Ichsan, 2017).

2.4.2 *Protokol Adhoc On-Demand Distance Vector* (AODV)

Ad hoc On-Demand Distance Vector (AODV) merupakan *routing protocol* yang termasuk dalam *reactive routing protocol*. *Reactive routing protocol* yaitu proses *routing* dibuat oleh node sumber apabila membutuhkan informasi *routing* ke node tujuan dengan melakukan *route discovery* yang diminta untuk membanjiri jaringan untuk mencari rute dan selesai ketika rute ditemukan (Training, 2016). *Protocol* AODV dirancang dengan menjaga *timer-based state* yang terdapat pada node sesuai dengan penggunaan tabel *routing* sehingga akan kadaluwarsa apabila tidak digunakan (Raharjo, 2017). Pada *protocol* AODV memiliki tahapan *routing* yang digunakan untuk menentukan dan memelihara rute yaitu *route discovery* dan *route maintenance*. *Route discovery* berupa *Route Request* (RREQ) serta *Route Reply* (RREP) sedangkan *route maintenance* berupa *Route Error* (RERR) (Raharjo, 2017). Dalam menjaga tabel *routing* pada *protocol* AODV, setiap node berisi informasi *field* yang berupa:

- *Destination* IP Address berisi alamat IP dari node tujuan untuk menentukan rute.
- *Destination sequence number* yang bekerjasama dalam menentukan rute.
- *Next hop* yaitu untuk meneruskan paket ke node tujuan.
- *Hop count* berisi jumlah *hop* dari alamat IP sumber ke alamat IP tujuan.
- *Lifetime* berisi waktu yang digunakan untuk node menerima RREP dari node tujuan.

- *Routing flags* berisi status dalam sebuah rute dapat berupa keadaan *up* (valid), *down* (tidak valid) atau dalam perbaikan.

Dalam mekanisme untuk menentukan rute dari node sumber ke node tujuan, *protocol* AODV menggunakan dua pesan yaitu mengirimkan pesan RREQ dan pesan RREP yang disimpan dalam tabel *routing* dengan satu *entry* untuk setiap tujuan (Perkins & Royer, 1999). Node sumber akan melakukan inisialisasi *route discovery process* untuk menemukan rute ke node tujuan apabila node sumber belum mempunyai rute yang benar dengan cara menyebarkan pesan RREQ menuju node tetangganya. Apabila node tersebut menerima pesan RREQ yang memiliki informasi rute menuju node tujuan maka akan mengirimkan pesan RREP kembali menuju node sumber (Rezkinanda & Anggoro, 2016).

Selain itu *protocol* AODV menggunakan *sequence number* yang digunakan untuk menjaga informasi rute terbaru mengenai *reverse path* atau rute balik pada setiap node tujuan agar tidak terjadi *routing loops* (Raharjo, 2017). Ketika node tujuan menerima pesan RREQ maka node tersebut akan membandingkan nilai *destination sequence number* yang dimiliki dengan nilai *destination sequence number* yang terdapat di pesan RREQ. Maka pesan RREP akan dikirimkan menuju node sumber apabila nilai *destination sequence number* yang terdapat pada node tujuan lebih besar atau sama dengan nilai yang ada di pesan RREQ, tetapi apabila lebih besar dilakukan dengan cara menyebarkan kembali ke node tetangganya (Rezkinanda & Anggoro, 2016). Sehingga node yang menerima RREP akan melakukan *update* informasi *route time out* yang telah dibuat dan akan dihapus ketika waktu *time out* telah habis. Serta apabila terjadi perubahan topologi yang mengakibatkan node tujuan tidak dapat menemukan informasi rute dalam tabel *routing*, maka node tersebut akan mengirimkan pesan RERR ke node tetangganya dan akan menghapus informasi rute yang mengalami *error* (Rezkinanda & Anggoro, 2016). Sehingga node sumber akan melakukan *route discovery* ulang untuk mencari informasi rute tersebut masih dibutuhkan.

2.5 Bahasa Pemrograman C++

Bahasa pemrograman C++ merupakan bahasa yang dikembangkan pada awal tahun 1980-an oleh Bjarne Stroustrup. Berawal dari mengembangkan bahasa C yang memiliki pemrograman yang prosedural sehingga dalam menyelesaikan suatu masalah dibagi dalam sub-sub masalah yang lebih kecil. Seiring dalam pengembangannya, ditemukan metode untuk menyelesaikan suatu masalah dengan mendefinisikan *class-class* yang merupakan *class* yang dibuat sebelumnya sebagai abstraksi dari objek-objek fisik dan memiliki konsep *Object*

Oriented Programming (OOP) yang dikenal sebagai bahasa C++. *Class-class* tersebut berisikan keadaan suatu objek yang anggotanya memiliki kemampuan dari objeknya, setelah beberapa *class* dibuat maka masalah dipecahkan menggunakan *class* untuk membantu dalam membuat dan mengelola program yang besar dan kompleks (Jiatmiko & Prayudi, 2015).

2.6 *Network Simulator 3*

Network simulator 3 (NS-3) merupakan simulator jaringan yang berfokus pada bidang penelitian dan pedidikan. Dibawah lisensi GNU GPLv2 perangkat lunak NS-3 memulai *project* pada Juli 2006 dan rilis pertama pada 30 Juni 2008 tersebut bersifat *open source* yang tersedia dalam beberapa lintas *platform* sehingga dapat digunakan secara umum untuk kepentingan penelitian, pengembangan dan penggunaan. NS-3 bertujuan untuk mengembangkan model simulasi jaringan dengan lingkungan yang luas, selaras dengan pengembangan kebutuhan simulasi jaringan yang modern. Pada insfrastruktur simulasi NS-3 juga mendukung pengembangan model simulasi jaringan mendekati kondisi yang realistis dan memiliki karakteristik sebagai emulator jaringan *wireless* ataupun *wired* (nsnam, 2011). Dengan dukungan modul *library* yang lengkap serta dukungan *protocol* jaringan yang populer, beberapa model radio, MAC dan lainnya, memungkinkan untuk pengembangan model jaringan yang modern. Selain itu penulisan baris kode pada NS-3 menggunakan bahasa pemrograman C++ dan Pyhton sehingga dapat disesuaikan dengan kebutuhan implemetasi pada simulasi jaringan (Islam & Bhuyan, 2015). Berikut merupakan model elemen yang terdapat dalam NS-3 yaitu (Henderson, Riley, Floyd, & Roy, 2019):

2.6.1 Node

Node dalam NS-3 merupakan istilah dari perangkat komputasi dasar atau *computer* yang terhubung pada suatu jaringan biasa disebut *host* atau *end system*. Pada NS-3, suatu perangkat jaringan sederhana adalah sebuah node. Kelas node tersebut menangani metode untuk mengelola representasi dari sebuah perangkat jaringan simulasi yang dapat dilakukan penambahan seperti aplikasi, *protocol internet* dan beberapa fungsi lainnya sehingga membentuk sistem yang bekerja secara penuh.

2.6.2 *Application*

Application dalam NS-3 merupakan suatu metode untuk meng-*generate* beberapa aktivitas atau paket data dijalankan pada saat simulasi. Kelas *application* tersebut memberikan

struktur untuk mengelola gambaran versi aplikasi *user-level* pada saat simulasi. Aplikasi tersebut berupa *UdpEchoClientApplication* dan *UdpEchoServerApplication*, suatu aplikasi *client* atau *server* yang digunakan untuk menghasilkan suatu paket dalam jaringan.

2.6.3 *Channel*

Channel dalam NS-3 merupakan suatu jenis saluran komunikasi yang berfungsi untuk menghubungkan sebuah node ke node lain atau sebuah objek kedalam jaringan. Kelas *channel* tersebut menyediakan metode untuk mengelola saluran komunikasi yang digunakan pada saat simulasi dapat berupa saluran *wired* atau *wireless.* Pada NS-3, beberapa *channel* yang digunakan antara lain: *CsmaChannel, PointToPointChannel* dan *WifiChannel.*

2.6.4 *Net Device*

Net device dalam NS-3 merupakan representasi dari sebuah kartu jaringan atau *Network Interface Card* (NIC) yang menghubungkan *device* ke dalam suatu jaringan. *Net device* tersebut dapat mewakili *software* ataupun *hardware* yang di-*install* dalam sebuah node untuk memungkinkan berkomunikasi antar node melalui *channel*. Pada NS-3, beberapa *net device* yang digunakan seperti: *CsmaNetDevice, PointToPointNetDevice,* dan *WifiNetDevice.*

2.6.5 *Topology Helper*

Topology helper dalam NS-3 merupakan suatu metode untuk mengatur banyak koneksi antar node, *NetDevice* dan *Channel* yang digunakan pada simulasi. *Topology helper* tersebut memudahkan konfigurasi dalam membuat simulasi jaringan dengan skala yang lebih besar, seperti: membuat *NetDevice,* menambahkan alamat MAC, meng-*install NetDevice* ke suatu *Channel*, konfigurasi *protocol routing* dan lain-lain.

2.7 *Quality of Service*

Quality of Service (QoS) merupakan suatu kemampuan dalam jaringan untuk mencapai kinerja yang maksimal dengan parameter-parameter tertentu. Dalam sebuah kinerja jaringan dapat terlihat kemampuan untuk mengatasi permasalahan seperti kepadatan trafik, waktu pengiriman yang besar, dan permasalahan lain agar suatu informasi dapat dikirimkan dari penerima ke penerima. Pada standar IEEE 802.11, parameter yang digunakan untuk mengukur kinerja jaringan berada pada MAC *Layer* (Kamarullah, Endroyono, & Wirawan, 2017). QoS merupakan salah satu bagian penting dalam VANET karena akan menentukan kualitas jaringan

yang diakibatkan oleh topologi jaringan yang berubah dengan mobilitas yang tinggi dan tersedianya informasi status perutean (Kamini & Kumar, 2010). Beberapa parameter yang dapat digunakan sebagai pengukur kinerja jaringan antara lain:

2.7.1 *Throuhgput*

Throughput merupakan rata-rata keberhasilan paket data yang dikirimkan dari node sumber ke node lain dalam satuan waktu pada suatu jaringan (Chrisnamurti, 2018). *Throughput* juga disebut sebagai *bandwidth* dalam kondisi yang sebenarnya, namun yang membedakan dengan *Throughput* adalah pada sifat transmisi datanya yang dinamis sedangkan *bandwidth* bersifat tetap (Jiatmiko & Prayudi, 2015). Oleh karena itu untuk menyimpulkan kecepatan jaringan, pengukuran *Throughput* dibutuhkan karena pada *Throughput* mengukur total kedatangan paket yang diterima dibagi dengan waktu yang dibutuhkan untuk mengirimkan paket ke tujuan sedangkan pengukuran *bandwidth* dianggap tidak cukup karena *bandwidth* hanya mentransmisikan data pada jaringan.

Throughtput dapat diukur dalam satuan *bit per second* (bps), *byte per second* (Bps) atau *packet per second* (pps). Adapun *throughput* dapat dihitung pada persamaan (2.1) (Jiatmiko & Prayudi, 2015).

$$Throughput = \frac{\text{ukuran paket yang diterima}}{\text{waktu pengiriman paket}} \qquad (2.1)$$

2.7.2 *Packet Loss Ratio* (PLR)

Packet Loss Ratio merupakan suatu kegagalan hilangnya paket data selama proses pengiriman dari node asal ke node tujuan (Jiatmiko & Prayudi, 2015). Kegagalan paket data tersebut dapat disebabkan oleh beberapa hal, di antaranya: terjadinya *overload* trafik dalam jaringan, tabrakan *(congestion)* dalam jaringan, *error* yang terjadi pada media fisik dan kegagalan pada sisi penerima karena *overflow* yang terjadi pada *buffer* (Chrisnamurti, 2018). Adapun *Packet Loss Ratio* dapat dihitung pada persamaan 2.2 (Jiatmiko & Prayudi, 2015).

$$Packet\ Loss\ Ratio = \frac{\text{paket yang dikirim} - \text{paket yang diterima}}{\text{paket yang dikirim}} . 100\% \qquad (2.2)$$

2.7.3 *Packet Delivery Ratio* (PDR)

Packet Delivery Ratio merupakan perbandingan dari jumlah paket data yang dikirimkan dengan jumlah paket yang diterima (Raharjo, 2017). Pada nilai rasio pengiriman paket ini

dipengaruhi dari beberapa faktor, seperti ukuran paket, jarak pengiriman, dan mobilitas dari node (Kamarullah, Endroyono, & Wirawan, 2017). Adapun *Packet Delivery Ratio* dihitung pada persamaan 2.3 (Jiatmiko & Prayudi, 2015).

$$Packet\ Delivery\ Ratio = \frac{\text{paket yang diterima}}{\text{paket yang dikirim}} . 100\% \quad (2.3)$$

2.7.4 *Delay*

Delay atau *end to end* merupakan waktu yang dibutuhkan untuk mengirimkan paket data dari node sumber menuju node tujuan (Rezkinanda & Anggoro, 2016). Apabila perhitungan waktu *Delay* yang dihasilkan rendah, maka kinerja *protocol* baik. Adapun waktu *Delay* dapat diketahui pada penghitungan 2.4.

$$Delay = \frac{\text{waktu data diterima} - \text{waktu data dikirim}}{\text{paket yang diterima}} \quad (2.4)$$

2.7.5 *Jitter*

Jitter merupakan variasi dari *Delay* yang dihasilkan akibat adanya selisih waktu atau interval pada saat pengiriman ke node sumber ke node tujuan yang diantrikan terlebih dahulu dalam *buffer* (Sidharta, 2012). Adapun *Jitter* dapat diketahui pada penghitungan 2.5 (Sidharta, 2012).

$$\text{Jitter} = \text{end to end delay}_n - \text{end to end delay}_{(n-1)} \quad (2.5)$$