SIMULASI SERANGAN *BLACK HOLE* PADA MANET *(MOBILE AD HOC NETWORK)*

Firman Ardiyansyah.[1], Ir. Wahyu Adi Priyono, M.T.[2], Rusmi Ambarwati, S.T., M.T. [2]
[1]Mahasiswa Teknik Elektro Univ. Brawijaya, [2]Dosen Teknik Elektro Univ. Brawijaya
Jurusan Teknik Elektro Fakultas Teknik Universitas Brawijaya
Jalan MT. Haryono 167, Malang 65145, Indonesia
E-mail: firman.ardiyansyah88@gmail.com

ABSTRAK

MANET *(Mobile Ad Hoc Network)* adalah kumpulan perangkat *mobile* yang digunakan untuk berkomunikasi secara nirkabel tanpa otoritas jaringan terpusat atau terstruktur. MANET memiliki beberapa karakteristik khusus seperti media yang terbuka, topologi yang dinamis, kurangnya pemantauan pusat, algoritma yang sederhana, dan tidak ada mekanisme pertahanan yang jelas. Dalam sistem jaringan yang terbuka, MANET mudah terkena berbagai jenis serangan, salah satunya adalah serangan *blackhole*. Serangan *black hole* adalah sejenis serangan *Denial of Service(DoS)* pada MANET dimana node berbahaya menyatakan kepada node sumber bahwa node tersebut memiliki rute terbaik untuk mencapai node tujuan. Masalah utama yang timbul pada saat terjadi serangan *black hole* adalah bagaimana untuk mendeteksi *black hole*, sambil menghindari sebanyak mungkin diagnosa yang salah, seperti RREP palsu. Dalam simulasi ini diusulkan mekanisme *watchdog* yang mengintegrasikan teknik dari *Bayesian filtering*. Mekanisme *watchdog* bertujuan memantau aktivitas node dalam jaringan untuk mendeteksi node bermasalah yang merugikan jaringan. Simulasi dilakukan menggunakan protocol routing AODV dengan dua skenario, yaitu simulasi serangan *black hole* dengan dan tanpa mekanisme *watchdog* menggunakan perangkat lunak NS-2. Hasil perbandingan performansi pada saat tidak menggunakan mekanisme *watchdog* dan dengan mekanisme watchdog diukur berdasarkan parameter *packet loss*, *delay*, dan *throughput*. nilai *packet loss* tanpa mekanisme watchdog sebesar nilai paket yang terkirim yaitu 100% untuk ukuran paket 1000 byte sementara pada saat penggunaan mekanisme *watchdog* nilai *packet loss* sebesar 0% untuk ukuran 1000 byte. Semakin kecil nilai *packet loss*, tingkat keberhasilan pencegahan serangan *black hole* dengan mekanisme *watchdog* semakin tinggi.

Kata Kunci : Serangan *Black Hole*, AODV, Mekanisme *watchdog*, NS2

ABSTRACT

MANET (Mobile Ad Hoc Network) is a collection of mobile devices that are used to communicate wirelessly without the authority of a centralized network or structured. MANET has some special characteristics such as an open media, dynamic topology, the lack of central monitoring, a simple algorithm, and no obvious defense mechanism. In the system of an open network, MANET is exposed to various types of attacks, one of which is a blackhole attack. Attacks black hole is a type of Denial of Service (DoS) on MANET where malicious nodes declare the source node that has the best route to the destination node. The main problem that arises in the event of black hole attacks is how to detect black holes, while avoiding as much as possible the wrong diagnosis, such as false RREP. In this simulation proposed a watchdog mechanism that integrates the techniques of Bayesian filtering. watchdog mechanism seeks to monitor the activity node in a network to detect problematic adverse node network. Simulations carried out using a routing protocol AODV with two scenarios, the simulated black hole attacks with and without using the software watchdog mechanism NS-2. Performance comparison results when not using the watchdog mechanism and the watchdog mechanisms measured by packet loss, delay, and throughput. the value of packet loss without a watchdog mechanism in the amount of packages sent 100% to the package size is 1000 bytes at a time when the use of the watchdog mechanism packet loss value of 0% for the 1000 bytes package size. The smaller the value of packet loss, the success rate of black hole attack prevention with higher watchdog mechanism.

Keywords: Black Hole Attack, AODV, watchdog mechanism, NS2

I. PENDAHULUAN

MANET *(Mobile Ad Hoc Network)* adalah kumpulan perangkat *mobile* yang digunakan untuk berkomunikasi secara nirkabel tanpa otoritas jaringan terpusat atau terstruktur. MANET memiliki beberapa karakteristik khusus seperti media yang terbuka, topologi yang dinamis, kurangnya pemantauan pusat, algoritma yang sederhana, dan tidak ada mekanisme pertahanan yang jelas. Dalam sistem jaringan yang terbuka, MANET mudah terkena berbagai jenis serangan, salah satunya adalah serangan *blackhole*. Serangan *black hole* adalah sejenis serangan *Denial of Service(DoS)* pada MANET dimana node berbahaya menyatakan kepada node sumber bahwa node tersebut memiliki rute terbaik untuk mencapai node tujuan selama proses pencarian rute. Masalah utama yang timbul pada saat terjadi serangan *black hole* adalah bagaimana untuk mendeteksi *black hole*, sambil menghindari sebanyak mungkin diagnosa yang salah, seperti RREP palsu.

Dalam permasalahan ini diusulkan mekanisme *Watchdog* yang mengintegrasikan teknik dari Bayesian filtering. Mekanisme *Watchdog* ini berperan sebagai *Intrusion Detection System (IDS)*, yang merupakan bagian

perangkat lunak yang mengumpulkan dan menganalisa lalu lintas jaringan untuk mendeteksi suatu serangan. Dalam konteks ini, sistem deteksi intrusi bertujuan memantau aktivitas node dalam jaringan untuk mendeteksi perilaku yang merugikan jaringan.

II. LANDASAN TEORI

Bab ini membahas mengenai teori penting yang menjadi acuan dalam skripsi ini. Bagian tersebut meliputi penjelasan tentang pengenalan MANET (*Mobile Ad Hoc Network*), *routing protocol*, AODV (*Ad Hoc On Demand Vector)*, Serangan *Blackhole*, mekanisme *watchdog*, NS 2, dan penelitian terkait.

2.1 Umum

2.1.1 MANET (*Mobile Ad Hoc Network*)

MANET merupakan suatu jaringan yang terdiri dari berbagai perangkat bergerak (*mobile)* yang tergabung menjadi satu tanpa adanya otoritas jaringan terpusat atau terstruktur *(ad hoc)*, sehingga membentuk suatu jaringan yang bersifat sementara (Irawan, Dedy; Roestam, Rusdianto, 2011). Dimana tiap perangkat yang ada memiliki tampilan nirkabel yang saling terhubung melalui gelombang radio. Perangkat - perangkat tersebut disebut *node*. Contoh nyata dari suatu node pada jaringan *ad hoc* adalah sebuah laptop dan *smartphone* yang saling terhubung secara langsung.

2.1.1 Protokol Routing AODV *(Ad Hoc On Demand Vector)*

AODV merupakan sebuah routing yang menyampaikan pesan antar komputer portable, dimana tiap komputer portabel atau biasa disebut node, menyampaikan pesan yang tidak bisa dikirim langsung, melalui node – node yang berada di sekitarnya untuk meneruskan pesan ke node tujuan dari node sumber.

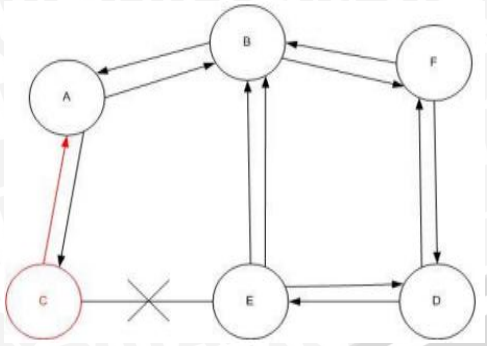
Proses pembentukan rute pada protokol routing AODV yaitu dengan menggunakan dua pesan yaitu dengan menggunakan *route request* (RREQ) dan *route reply* (RREP) (Perkins, Belding-Royer, & Das, 2003). Ketika node sumber menginginkan suatu rute menuju node tujuan tetapi belum mempunyai rute yang benar, maka node sumber akan menginisialisasi proses pencarian rute untuk menemukan rute ke *node* tujuan dengan langkah-langkah berikut :

1. Node sumber akan mem-*broadcast* paket RREQ menuju node terdekatnya. RREQ paket berisi *source address, destination address, hop counter, source* dan *destination sequence number,* dan *broadcast ID.* Nilai *Broadcast ID* akan bertambah satu setiap suatu node sumber mengirimkan RREQ yang baru dan digunakan sebagai identifikasi sebuah paket RREQ.
2. Jika node yang menerima RREQ memiliki informasi rute menuju node tujuan, maka node tersebut akan mengirim paket RREP kembali menuju node sumber. Tetapi jika tidak memiliki informasi rute maka node tersebut akan mem-*broadcast* ulang RREQ ke node terdekatnya setelah menambahkan nilai *hop counter*.
3. Node yang menerima RREQ dengan nilai *source address* dan *broadcast ID* yang sama dengan RREQ yang diterima sebelumnya akan membuang RREQ tersebut. *Source sequence number* digunakan oleh suatu node untuk memelihara informasi yang benar mengenai *reverse path* (jalur balik) menuju node sumber. Pada saat RREQ mengalir menuju node tujuan yang diinginkan, dia akan menciptakan *reverse path* menuju node sumber, setiap node akan membaca RREQ dan mengidentifikasi alamat dari node terdekat yang mengirim RREQ tersebut.
4. Ketika node tujuan atau node yang memiliki informasi rute menuju node tujuan menerima RREQ maka node tersebut akan membandingkan nilai *destination sequence number* yang dia miliki dengan nilai *destination sequence number* yang ada di RREQ.
5. Route replay (RREP) akan dikirim menuju node sumber apabila nilai *destination sequence number* yang ada di node lebih besar atau sama dengan nilai yang ada di RREQ , namun jika lebih besar maka akan di-*broadcast* kembali ke node terdekatnya.
6. Node perantara yang menerima RREP akan meng-*update* informasi *time out* (masa aktif rute) jalur yang telah diciptakan. Informasi rute dari node sumber ke node tujuan akan dihapus apabila waktu (*time out)* habis.

Di dalam AODV setiap node bertanggung jawab untuk memelihara informasi rute yang telah disimpan di dalam routing table-nya. Pada saat pengiriman data apabila terjadi perubahan topologi yang mengakibatkan suatu node tidak dapat dituju dengan menggunakan informasi rute yang ada di routing table, maka suatu node akan *route error packet (RRER)* ke node terdekatnya dan node terdekatnya akan mengirim kembali RRER demikian seterusnya hingga menuju node sumber .Setiap node yang memperoleh RRER ini akan menghapus informasi yang mengalami error di dalam routing table-nya. Kemudian node sumber akan melakukan proses pencarian rute kembali apabila rute tersebut masih diperlukan.

2.1.2 Serangan *Black Hole*

Serangan *Black Hole* merupakan serangan yang ada pada MANET. Serangan ini merupakan sebuah node yang mengirimkan RREP palsu pada node sumber yang telah menyebarkan RREQ untuk memulai pengiriman sebuah paket. *Node* ini akan memberi informasi baru mengenai rute baru yang lolos dari pengecekan tabel *routing*. Dengan cara ini, node tersebut dapat mengecoh node sumber agar menerima RREP yang dikirimkan. Sehingga paket yang terkirim akan dicegah dan proses pengiriman ke node tujuan mengalami gangguan.



Gambar 2.1 Serangan Black Hole

Gambar 2.1 memperlihatkan bagaimana proses serangan *Black Hole* berlangsung. Proses serangan *Black hole* diuraikan sebagai berikut:

- nodeA akan melakukan pengiriman paket ke nodeD dan menyebarkan proses penemuan rute pada semua node yang berada didekatnya.
- Jika *node* C adalah node berbahaya atau gangguan, maka node tersebut akan membentuk sebuat rute aktif menuju node tujuan dengan cepat setelah node tersebut menerima paket RREQ.
- Balasan RREP palsu akan dikirimkan oleh *node* C kepada *node* A sebelum node yang lain membalas. Sehingga *node*A akan mengira jika rute aktif tersebut adalah sebuah rute pengiriman yang telah selesai terbentuk.
- *Node* A tidak akan menghiraukan seluruh balasan dari *node* lain yang melakukan RREP padanya dan pengiriman paket akan mulai berlangsung pada *node* C. Karena hal inilah, paket akan mengalami kehilangan ataupun pengurangan informasi data(*Ullah, Irshad and Rehman, Shoaib Ur, 2010*).

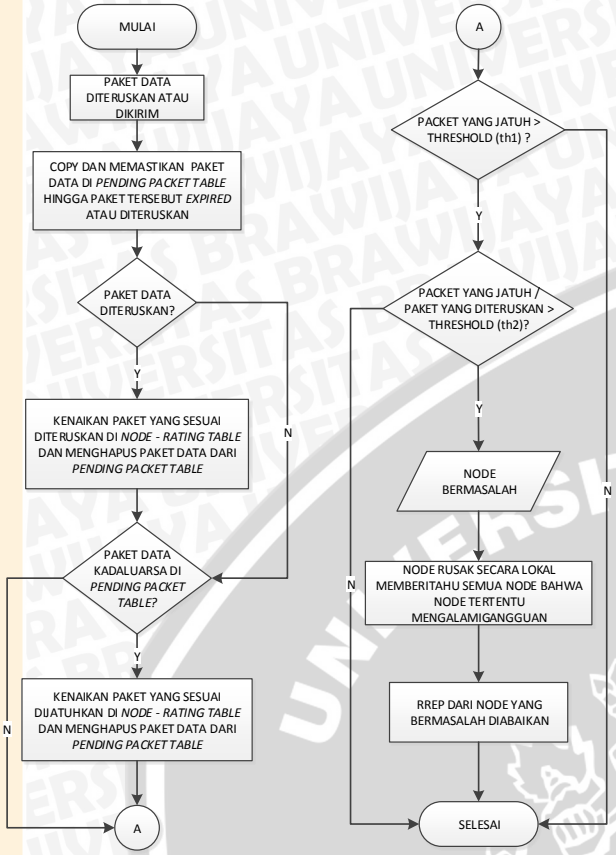
Terdapat dua tipe serangan *Black Hole* pada AODV, yaitu *Internal Serangan Black Hole* dan *External Serangan Black Hole*. *Internal Serangan Black Hole* adalah jenis serangan yang terjadi pada rute asal paket dan rute tujuan paket dikirim. Secepat setelah mendapatkan kesempatan, *node* penyerang akan membuat dirinya menjadi sebuah bagian dari aktif *router* data. Pada tahap ini *node* penyerang mampu untuk melakukan serangan dengan memulainya pada transmisi data. Ini termasuk serangan internal karena *node* tersebut berada dalam rute pengiriman data. Serangan internal lebih rentan dan sulit dicegah karena sulitnya mendeteksi *node* internal yang berbahaya. Sementara *External Serangan Black Hole* secara fisik berada di luar jaringan dan menolak akses menuju lalu lintas jaringan atau menciptakan kemacetan dalam jaringan atau dengan mengganggu seluruh jaringan. Serangan eksternal dapat menjadi jenis serangan internal jika mengambil alih kontrol dari *node* penyerang yang berada di bagian internal dan mengontrol untuk melakukan suatu serangan terhadap *node* lain yang ada pada MANET. Serangan *Black Hole* eksternal dapat dilihat dari poin – pon berikut ini :

a. Node penyerang mendeteksi rute yang aktif dan mencatat alamat tujuan.
b. Node penyerang mengirim rute paket balasan (RREP) termasuk alamat tujuan *field spoofed* menuju alamat tujuan yang tidak diketahui. Nilai hop menjadi lebih rendah dan pengurutan nomer menjadi lebih tinggi.
c. Node penyerang mengirim RREP ke node terdekat yang mana termasuk dalam rute aktif. Hal ini juga termasuk dalam pengiriman langsung menuju node sumber data yang ada.
d. RREP diterima oleh node terdekat yang ada kepada node penyerang yang menyesuaikan diri dengan rute berbeda yang dibentuk dari nodesumber.
e. Informasi baru diterima dalam rute yang memberi balasan akan mengijinkan node sumber untuk memperbarui tabel *routing.*
f. Rute baru dipilih oleh node sumber untuk seleksi data.
g. Node penyerang akan menjatuhkan seluruh data yang ada pada rute yang terbentuk.

2.1.3 Mekanisme *Watchdog*

Inti dari mekanisme *watchdog* adalah pengawasan tepat (*promiscuous monitoring).* Jika terdeteksi node berbahaya, node sumber akan memilih rute baru yang bebas dari node berbahaya dengan bantuan "*Path rater*". Mekanisme ini tidak akan menampilkan hasil dengan baik pada saat kondisi jaringan tidak menguntungkan dimana terdapat gangguan dari node berbahaya yang mengakibatkan data rusak saat probabilitas tinggi. Terdapat dua jenis pendekatan pada mekanisme *watchdog,* yaitu pendekatan pertama adalah deteksi serangan *End-to-End* menggunakan deteksi kesalahan. Kekurangannya adalah *Throughput* jaringan dapat berkurang bahkan saat tidak ada node berbahaya.

Dalam skema ini, node sumber dan node tujuan tidak mengetahui tentang lokasi node berbahaya. Yang kedua adalah pendekatan yang diambil secara umum untuk mengeksploitasi sifat penyiaran dari media nirkabel dimana sebuah node akan mengawasi arus pergerakan data pada suatu jaringan, node tersebut merupakan *watchdog.* Untuk pendekatan yang kedua ini memiliki tantangan utama, yaitu jaringan yang perlahan menghilang, terjadinya tabrakan pengirimandan gangguan lain yang dapat muncul tanpa diduga. Sehingga *watchdog* tidak dapat menangkap pengiriman yang berlangsung. Di sini, perilaku *next - hop* diukur dengan evaluasi catatan lokal terbagi menjadi 2, yaitu *ratio bytes and ratio packet* dan *forwarded by the next - hop neighbor.* Tingkat kepercayaan dari *node* adalah gabungan dari pengamatan lokal dan informasi yang disiarkan. Tingkat kepercayaan dimasukan pada RREQ tersebut kemudian rute dipilih dengan cara yang sama dengan AODV. Algoritma prinsip kerja mekanisme *watchdog* dapat dijelaskan pada Gambar 2.2 sebagai berikut:



Gambar 2.2 Diagram alur mekanisme *watchdog*

Mekanisme watchdog pada Gambar 2.2 dijelaskan sebagai berikut:

1. Data paket diteruskan atau dikirim kepada node tujuan.
2. Meng- *Copy* dan menjaga paket data di tabel*pending packet* hingga paket tersebut kadaluarsa atau diteruskan
3. Jika paket data diteruskan maka paket yang peningkatannya sesuai dengan tabel *node rating* diteruskan dan menghapus paket data dari tabel *pending packet*.
4. Jika paket data berakhir di tabel *pending packet*, maka paket yang peningkatannya sesuai dengan tabel *node rating* dihentikan dan menghapus paket data dari tabel *pending packet*.
5. Jika paket yang dihentikan lebih besar dari threshold (TH1) kemudian jika paket yang dihentikan ataupun paket yang diteruskan lebih besar dari threshold (Th2) *node* tersebut dianggap bermasalah *(misbehave)* kemudian memberi tahu semua *node* yang ada pada jaringan bahwa *node* tersebut bermasalah.
6. RREP yang datang dari *node* yang bermasalah tersebut diabaikan.

2.1.6 *Bayesian Watchdog*

Untuk mendeteksi node yang bermasalah, monitoring jaringan diperlukan. Setiap node harus mengawasi node terdekatnya, dan pengawas merupakan komponen populer dalam system deteksi gangguan didedikasikan untuk tugas ini. Masalah utama adalah bahwa pengawas ditandai dengan ketidakakuratan dan kecepatan deteksi rendah, pada dasarnya karena mobilitas dan sinyal gangguan. Hasil-hasil sebelumnya dari kelompok telah dievaluasi pengawas Bayesian lebih Ad-hoc On-demand Distance Vector (AODV) routing dalam MANETs. Pengawas Bayesian ini hasil dari agregasi filter Bayesian dengan implementasi pengawas standar. Pengawas standar hanya mendengar paket yang dikirimkan dan diterima oleh tetangganya, menghitung paket yang harus ditransmisikan secara berulang, dan komputasi kepercayaan untuk setiap tetangga sebagai rasio "paket yang dikirimkan" untuk "paket yang seharusnya dikirimkan kembali". Jika node mentransmisikan semua paket yang seharusnya telah ditransmisikan ulang, memiliki tingkat kepercayaan dari 1 jika node memiliki tingkat kepercayaan yang lebih rendah dari ambang toleransi dikonfigurasi, node yang ditandai sebagai node berbahaya.

Peran filter Bayesian di pengawas adalah untuk tingkat kemungkinan memperkirakan kondisi sistem. Dasar matematika dari bayesian filter berikut: pada waktu t, state diperkirakan oleh variabel θ acak, yang tidak diketahui, dan ketidakpastian ini dimodelkan dengan asumsi bahwa θ sendiri diambil sesuai dengan distribusi yang diperbarui seperti terjadi pengamatan yang baru. Hal ini biasa disebut kepercayaan atau Belt (θ). Untuk menggambarkan hal ini, mari kita asumsikan pada persamaan berikut:

$$Bel_t(\vartheta) = p(\vartheta|z_1, z_2, .., z_n, .., z_t) = Beta(\alpha_t, \beta_t, \vartheta)$$

Keterangan:

Error! Reference source not found.= tingkat kepercayaan

Error! Reference source not found. = variable acak

α dan β = keadaan sistem

Error! Reference source not found. = informasi

Dalam pendekatan ini, variabel θ acak milik interval [0,1]. Bayesian filtering bergantung pada distribusi Beta, yang cocok untuk memperkirakan keyakinan dalam interval ini, seperti yang ditunjukkan dalam ekspresi 1; α dan β mewakili keadaan sistem, dan mereka diperbarui sesuai dengan persamaan berikut:

$$\alpha_{t+1} = \alpha_t + Z_t; \beta_{t+1} = \beta_t + Z_t$$

Keterangan:

α dan β = keadaan sistem

Error! Reference source not found. = informasi pengawas lokal

Fungsi Beta hanya membutuhkan dua parameter yang terus diperbarui sebagai pengamatan yang dibuat atau dilaporkan. Dalam pendekatan ini, ZT pengamatan merupakan informasi dari pengawas lokal yang diperoleh dalam interval waktu [t, t + AT] tentang persentase paket

non-diteruskan. Pengawas bayesian menggunakan tiga parameter: dua parameter pertama adalah α dan β, yang ditangani ke fungsi Beta untuk mendapatkan estimasi jalur node. Dengan demikian, kita dapat mengatakan bahwa α dan β adalah representasi numerik dari reputasi node. Parameter ketiga adalah γ, yang mewakili devaluasi bahwa pengamatan harus berusajha keras untuk beradaptasi dengan perilaku pengawas untuk skenario yang terus berubah tanpa mempengaruhi node tertentu. Jadi itu adalah mekanisme untuk mengintegrasikan kembali node ke MANET jika mereka mengubah perilaku mereka ke yang lebih kooperatif.

2.1.7 NS2 *(Network Simulator 2)*

NS2 (*Network Simulator 2*) dikembangkan pertama kali di UCB (University of California Berkeley) yang didukung oleh DARPA. NS2 merupakan suatu *system* yang bekerja pada *system* Unix/Linux , NS2 juga dapat dijalankan dalam *system* Windows namun harus menggunakan Cygwin sebagai Linux *Enviroment*nya. NS2 dibangun dari 2 bahasa pemrograman yaitu C++, sebagai *library* yang berisi *event scheduler*, protokol , dan *network component* yang diimplementasikan pada simulasi oleh *user*. Otcl digunakan pada *script* simulasi yang ditulis oeh NS *user*. Otcl juga berperan sebagai *interpreter*. Bahasa C++ digunakan pada library karena C++ mampu mendukung *runtime* simulasi yang cepat, meskipun simulasi melibatkan simulasi jumlah paket dan sumber data dalam jumlah besar. Sedangkan bahasa Tcl memberikan respon *runtime* yang lebih lambat daripada C++, namun jika terdapat kesalahan , respon Tcl terhadap kesalahan *syntax* dan perubahan *script* berlangsung dengan cepat dan interaktif. Pengetahuan tentang komponen pembangun NS2 dan letaknya akan sangat berguna dalam membangun simulasi. *Transport agent* pada NS 2 di jaringan internet, terdapat 4 layer komunikasi TCP/IP. Yaitu layer aplikasi, transport, IP dan network. Lapisan transport merupakan layer komunikasi yang mengatur komunikasi data yang akan digunakan oleh lapisan aplikasi di atasnya. NS berada pada lapisan transport dengan objek simulasi yang bernama *transport agent*. Pada simulasi pengiriman data, *transport agent* tidak dapat bergerak sendiri. *Transport agent* membutuhkan lapisan aplikasi di atasnya yang berfungsi sebagai *traffic generator*. Kedudukan lapisan aplikasi NS 2 pada sistem dunia nyata, aplikasi terhubung dengan lapisan transport yang ada di bawahnya melalui sebuah *Aplication Program Interface* (API). Jenis API yang umum digunakan yaitu *socket*. Ada 2 tipe dasar aplikasi yang disimulasikan pada NS2, yaitu:

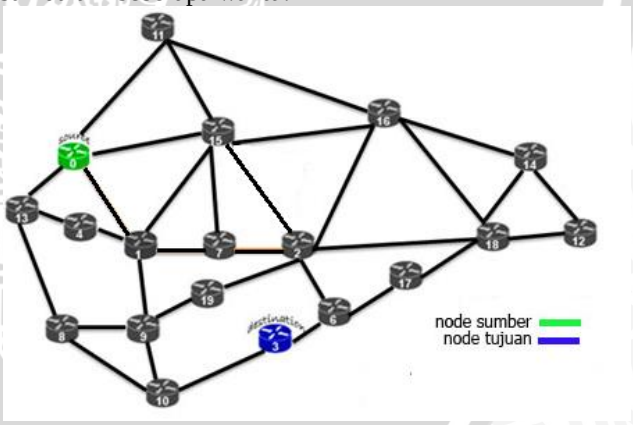
a. *Simulated Application*
Pada saat ini baru terdapat dua jenis aplikasi yang disimulasikan oleh NS yaitu :
a. FTP dibangun untuk mensimulasikan *bulk* data transfer.
b. Telnet masing-masing aplikasi diatur oleh *transport agent*. Jumlah paket yang ditransmisikan diatur oleh mekanisme *flow control* dan *congestion control* TCP.

b. *Generator Traffic*
Object generator traffic dibagi atas 4 type, yaitu:
a. Eksponensial, *generator traffic* ini membangkitkan *traffic* dengan *inter arrival time* antar paket sesuai dengan fungsi eksponensial.
b. Pareto, *generator traffic* ini membangkitkan *traffic* dengan *inter arrival time* antar paket sesuai dengan fungsi pareto.
c. CBR, fungsi ini membangkitkan data secara berkelanjutan dengan *bit rate* yang konstan.
d. Traffic race, generator ini membangkitkan traffic dari sebuah file trace.

Secara umum simulasi *routing* pada NS2 dilakukan melalui 3 blok fungsi, yaitu *routing agent* yang mempertukarkan paket *routing* antar node, *routing logic* yang menggunakan informasi yang dikumpulkan oleh *routing agent* untuk melakukan perhitungan rute dalam bentuk tabel *routing* dan peng-klasifikasi yang ada pada node.

2.2 Topologi

Topologi yang digunakan merupakan jenis topologi dinamis dimana pergerakan *routing protocol* AODV merupakan pergerakan secara acak dan berubah – ubah dalam beberapa waktu.

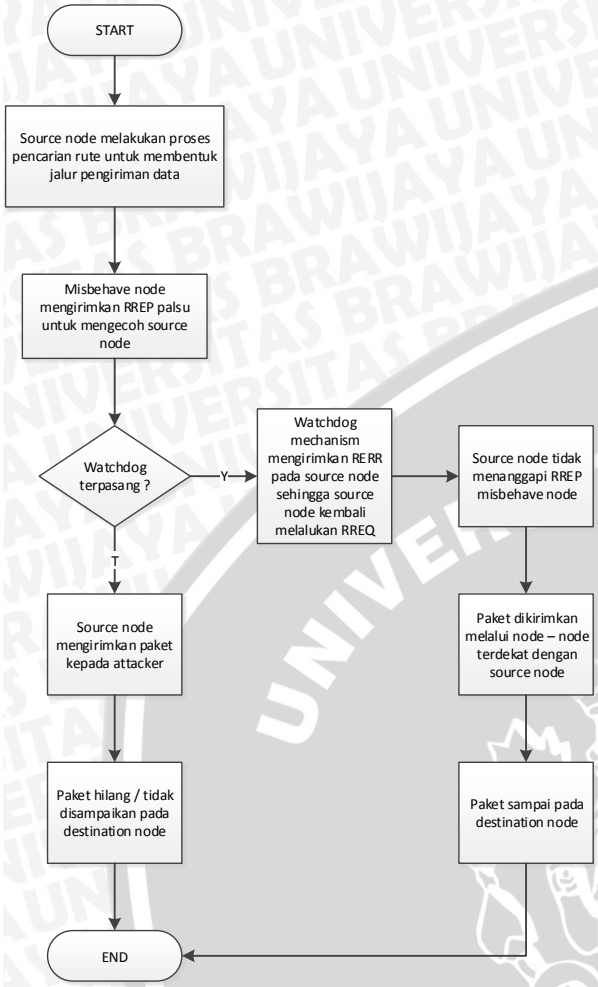


Gambar 2.3 Topologi dinamis dengan routing protokol AODV

III. METODE PENELITIAN

3.1 Diagram Alur Sistem

Adapun penjelasan mengenai diagram alur perancangan dapat dilihat pada Gambar 3.1 di bawah ini, sebuah diagram alur penerapan rancangan program yang akan diujikan.

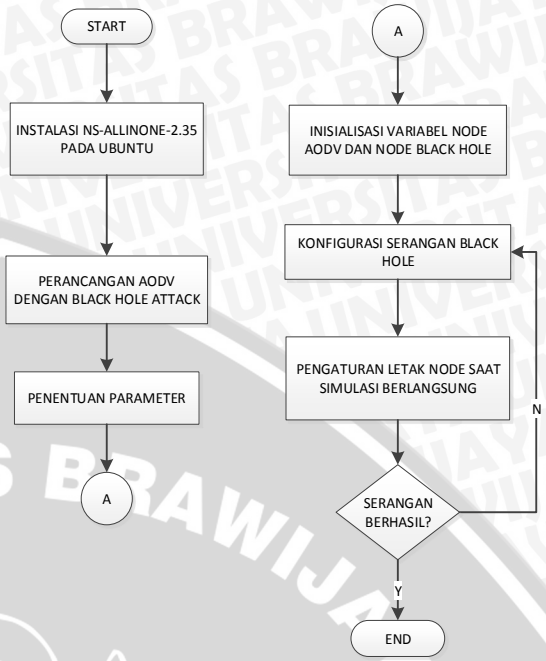


Gambar 3.1 Diagram alur rancangan program

Topologi yang digunakan merupakan jenis topologi dinamis dimana pergerakan *routing protocol* AODV merupakan pergerakan *routing* secara acak dan berubah – ubah dalam beberapa waktu. Topologi yang diimplementasikan terbagi dua topologi ialah topologi tanpa mekanisme *Watchdog* seperti pada Gambar 2.9 dan topologi dengan mekanisme Watchdog seperti pada Gambar 2.3 di BAB 2.

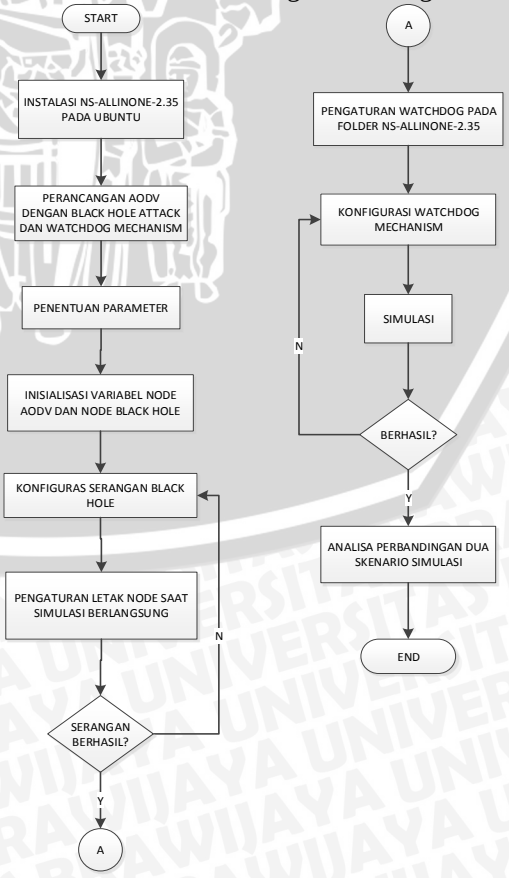
3.1.1 Instalasi Kebutuhan Simulasi

Pada bagian ini menjelaskan tentang desain instalasi program. Proses instalasi program berlangsung bertahap dan dilakukan secara terpisah dengan instalasi Serangan *Black Hole* (Gambar3.2) terlebih dahulu dilanjutkan dengan instalasi Mekanisme *watchdog* (Gambar 3.3).



Gambar 3.2 Diagram alur instalasi Serangan *Black Hole*.

Proses pemasangan serangan *Black Hole* pada Gambar 3.2 menggunakan sebuah berkas *patch* yang dipasang dalam folder ns-allinone-2.35. Selanjutnya dilakukan konfigurasi untuk penyesuaian informasi agar dapat membaca pergerakan data yang telah ditentukan. Begitu juga dengan pemasangan Mekanisme *watchdog* pada Gambar 3.3, sebuah berkas *patch* dalam folder ns-allinone-2.35 setelah konfigurasi Serangan *Black Hole*.



Gambar 3.3 Diagram alur instalasi Mekanisme *watchdog*

3.2 Pembahasan dan Hasil

Uji coba simulasi dilakukan pada *network simulator 2.35* yang telah dipasang pada Ubuntu 12.04 di virtualbox. Penerapan dua skenario dijelaskan sebagaia berikut:

a. Skenario 1 – Serangan *Black Hole*

Parameter :

val(nn)	20	val(x)	800
Rp	AODV	val(y)	541
val(ifqlen)	50	time finish	20.0

Keterangan:

Val(nn) : jumlah node yang ditentukan pada simulasi

Rp :karakteristik protokol routing

Val(x) dan val (y): luas bidang pada simulasi

Val(ifqlen) : variable maksimum pada file Tcl di simulasi

Time finish: waktu pengiriman paket

Prinsip Kerja :

Saat simulasi berlangsung, *node* sumber akan menyebarkan proses pencarian rute untuk memulai pengiriman paket. Sebuah *node* yang diidentifikasikan sebagai *Black Hole* akan melakukan RREP pada *node* sumber dan melakukan serangan dengan *drop packet*. Dari sini dapat dihitung jumlah paket yang hilang, waktu *delay* dan paket yang berhasil terkirim. Jika jumlah paket hilang lebih besar dibanding paket terkirim, maka serangan *Black Hole* berhasil dilakukan.

b. Skenario 2 – Mekanisme *watchdog*

Parameter :

val(nn)	20	val(x)	800
Rp	AODV	val(y)	541
val(ifqlen)	50	time finish	20.0

Keterangan:

Val(nn) : jumlah node yang ditentukan pada simulasi

Rp : karakteristik protokol routing

Val(x) dan val (y): luas bidang pada simulasi

Val(ifqlen) : variable maksimum pada file Tcl di simulasi

Time finish : waktu pengiriman paket

Prinsip Kerja :

Mekanisme *watchdog* yang telah terpasang pada NS2 akan melakukan kerjanya ketika serangan *Black Hole* terjadi. Hal ini dapat dilihat dari pergerakan proses pengiriman paket yang langsung mencari rute baru tanpa melewati *node* berbahaya. Saat jumlah paket yang terkirim akan lebih besar dari jumlah paket yang hilang. Ini dikarenakan *watchdog* telah menemukan keberadaan *node* berbahaya dan mengabarkan pada *node* lain untuk membuat rute baru tanpa melewati *node* berbahaya. Berdasarkan pengawasan Bayesian penelitian ini menerapkan sebuah pengawas Bayesian yang kolaboratif berdasarkan pada mekanisme pesan-lewat di setiap pengawas individu yang memungkinkan reputasi yang baik. Setiap node berjalan dan berkolaborasi dengan pengawas untuk mengumpulkan semua informasi reputasi agar mendapatkan nilai-nilai α dan β 'untuk setiap node tetangga. Hal yang mendasari pendekatan ini adalah bahwa jika pengawas Bayesian bekerja dengan baik untuk mendeteksi *Black Hole*, sekelompok gabungan pengawas Bayesian akan mampu melakukan deteksi lebih cepat dan lebih akurat. Demikian pula untuk pengawas bayesian, pengawas Bayesian kolaboratif hanya mendengar jaringan untuk mengumpulkan informasi tentang paket yang dikirim dan diterima. Selain itu, memperoleh α dan β nilai untuk seluruh lingkungan nya. Nilai-nilai ini sama dengan nilai yang diperoleh oleh pengawas bayesian dengan pengamatan yang sama, hal ini disebut "informasi tangan pertama" atau "reputasi langsung". Secara berkala, pengawas berbagi informasi dengan node terdekat, dan disebut "informasi tangan kedua" atau "reputasi tidak langsung". Dalam implementasi ini, reputasi tidak langsung dimodulasi menggunakan parameter δ. Setiap kali diperlukan, setiap node menjalankan penghitungan pengawas bayesian kolaboratif.

Penggunaan kelas layanan atau tipe komunikasi berupa CBR (*Constant Bit Rate*) yang bekerja sebagaimana layaknya, atau mengemulasikan sebuah saluran fisik, tidak membutuhkan pengecekan *error* atau kontrol aliran data dan tanpa gangguan *jitter* (variasi *delay* antar paket yang terjadi pada jaringan). Jika didapat perbedaan seperti yang diharapkan maka penelitian simulasi tugas akhir ini dinyatakan berhasil sesuai dengan konsep yang diterapkan sebelumnya.

3.2.1 Performansi

Parameter yang digunakan untuk perbandingan performansi adalah sebagai berikut :

a. Throughput

Throughput adalah laju data aktual per satuan waktu, bisa disebut juga sebagai *bandwith* dalam kondisi yang sebenarnya. Namun, *bandwith* lebih bersifat tetap sementara *throughput* sifatnya dinamis tergantung proses yang yang sedang terjadi dengan satuan yang digunakan adalah Bps (*Bits per second*). Rumus menghitung *throughput* yaitu:

$$Throughput = received\_Data * 8/DataTransmissionPeriod \quad (1)$$

b. Delay

Delay adalah jeda waktu antara paket pertama dikirim dengan paket yang diterima oleh tujuan. Rumus *delay* yaitu :

$$Delay = \left(\frac{waktu\ terima\ paket - waktu\ kirim\ paket}{paket\ yang\ diterima}\right) \quad (2)$$

c. Packet Loss

Packet Loss adalah banyaknya paket yang hilang selama proses pengiriman paket berlangsung. Rumus *packet loss* yaitu :

$$Packet\ Loss = \frac{Packet\ Drop}{Packet\ Sent} \times 100\% \quad (3)$$

3.2.2 Analisa Simulasi

Pada bagian ini menjelaskan bagaimana dua simulasi akan dilakukan, yaitu simulasi serangan *black hole* tanpa mekanisme *watchdog* dan simuasi serangan *black hole* dengan mekanisme *watchdog* dimana masing-masing dari kedua simulasi tersebut menggunakan 20 node dengan luas bidang simulasi sebesar 541x800 pixel. Dari kedua simulasi tersebut akan ditemukan nilai perbandingan sesuai dengan parameter yang nantinya dianalisa untuk penentuan keberhasilan mekanisme keamanan *watchdog* terhadap serangan *black hole* pada protokol *routing* AODV dalam MANET.

IV PEMBAHASAN DAN HASIL

4.1 Menjalankan Serangan Black Hole

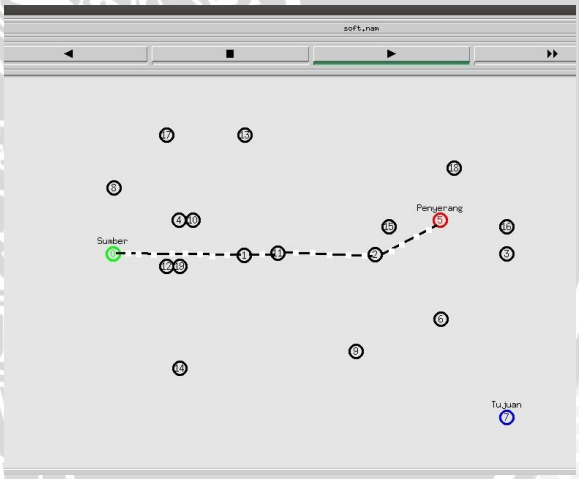
Penerapan serangan *Black Hole* dapat langsung dijalankan melalui terminal. Jumlah awal node yang digunakan sebagai uji coba keberhasilan konfigurasi sebanyak 7 *node*. Node sumber adalah node 0 berwarna hijau dan node tujuan adalah node 3 berwarna biru. Sementara node 5 diidentifikasikan sebagai node berbahaya. Node 0 melakukan proses pengiriman data yang tidak disampaikan oleh node *2*.

Proses pengiriman diatur dalam konfigurasi jenis komunikasi yang digunakan, yaitu CBR. Besar paket yang terkirim terbagi menjadi 5, yaitu 1000, 1500, 2000, 2500 dan 3000 bytes dengan melakukan 5 kali simulasi yang berbeda. Sementara besaran paket pada UDP bernilai tetap yaitu 1000 bytes.

```
set udp0 [new Agent/UDP]
$ns attach-agent $n0 $udp0
set null1 [new Agent/Null]
$ns attach-agent $n3 $null1
$ns connect $udp0 $null1
$udp0 set packetSize_ 1000

set cbr0 [new
Application/Traffic/CBR]
$cbr0 attach-agent $udp0
$cbr0 set packetSize_ 1000
$cbr0 set rate_ 0.1Mb
$cbr0 set random_ null
$ns at 0.0 "$cbr0 start"
$ns at 20.0 "$cbr0 stop"
```

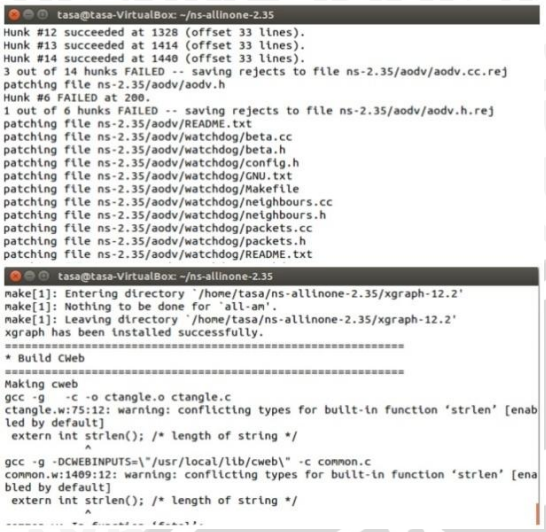
Pada gambar 4.1 memperlihatkan bagaimana simulasi proses serangan telah berhasil. Node 0 sebagai node sumber mengirimkan RREQ kepada seluruh node yang ada pada jaringan untuk proses penemuan rute dalam menemukan rute yang sesuai menuju node tujuan (node 7). Node 0 terkecoh oleh RREP palsu yang diberikan node berbahaya (node 5) kemudian melakukan pengiriman paket pada node 5 dan tidak menyampaikan paket pada node 7 sebagai node tujuan yang telah ditentukan.



Gambar 4.1 Proses serangan *Black Hole*

4.2 Konfigurasi Mekanisme watchdog

Tidak jauh berbeda dengan konfigurasi serangan *Black Hole*, konfigurasi Mekanisme *watchdog* sendiri menggunakan metode *pacthing* pada NS2.35. Proses *pacthing* merupakan proses penambahan fungsi library pada file NS2.35 agar bisa menyesuaikan kebutuhan informasi mengenai *watchdog*.

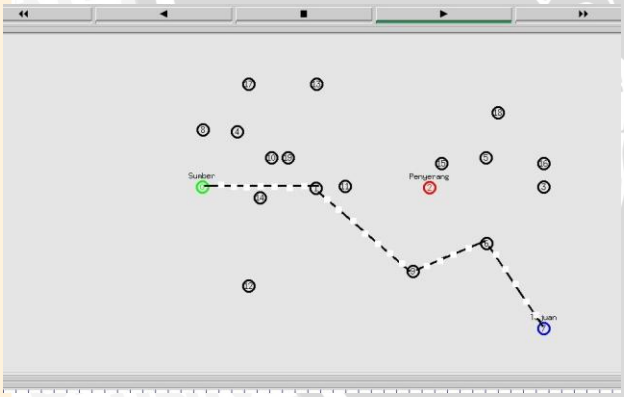


Gambar 4.2 Proses konfigurasi *Watchdog*

Berikut ini adalah code yang digunakan untuk konfigurasi *mekanisme watchdog* pada NS2 :

```
$ cd ns-allinone-2.35/
$ patch -p0 < watchdog-
bayesian2.0_ns235.patch
$ export CC=gcc-4.4 CXX=g++-4.4
&& ./install
```

Untuk menjalankan mekanisme *watchdog* dapat langsung melakukan pemanggilan program .tcl seperti sebelumnya. Perubahan langsung terlihat seperti pada gambar 4.6 di bawah ini.



Gambar 4.3 Simulasi dengan Mekanisme *watchdog*

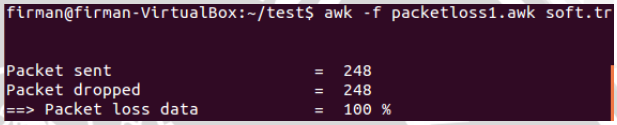
Proses akan langsung kembali normal sebelum serangan terjadi. Hal ini dikarenakan pendektesian node berbahaya oleh mekanisme *watchdog*. Node 0 dimana sebagai node sumber tidak akan menghiraukan RREP palsu yang dikirimkan oleh node berbahaya (node 2) dan langsung mencari node lain sebagai jalur yang dapat mengirimkan paket kepada node tujuan (node7).

4.3 Analisa Serangan *Black Hole* dan Keamanan *Watchdog*

Pada bagian ini menjelaskan analisa proses serangan dan keamanan yang telah dilakukan dengan membandingkan hasil dari kedua proses tersebut. Bandwidth yang digunakan sesuai dengan bandwidth yang telah disediakan oleh NS 2 dengan tipe wifi 802.11 yaitu sebesar 2Mbps.

4.3.1 Analisa Serangan *Black Hole*

Pembuktian serangan dilakukan dengan cara menjalankan file tcl melalui terminal. Node sumber menyebarkan proses penemuan rute untuk membentuk jalur pengiriman menuju node tujuan. Selanjutnya, node – node lain yang berada pada lingkup penyebaran akan memberi RREP terhadap RREQ dari node sumber. Bila dalam keadaan normal node sumber akan memilih node – node terdekat untuk membentuk suatu rute pengiriman, namun pada keadaan ini, dimana serangan telah diaktifkan, node sumber hanya akan menanggapi RREP dari node berbahaya yang telah melakukan proses RREP lebih cepat dibanding node normal lainnya pada node sumber.

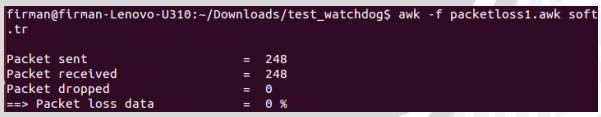


Gambar 4.4 Nilai *Packet Loss* saat serangan berlangsung

Sehingga tanpa disadari oleh node sumber akan mengirimkan paket pada serangan. Simulasi berlangsung seperti pada gambar 4.4 di atas yang telah disebutkan. Serangan dinyatakan berhasil karena nilai *Packet Loss* sebesar nilai paket yang dikirimkan. Ini berarti paket tidak diterima oleh node tujuan dan serangan telah terjadi selama proses pengiriman berlangsung.

4.3.2 Analisa Keamanan *Watchdog*

Sementara untuk pembuktian mekanisme *watchdog* dilakukan sama persis dengan simulasi saat serangan *Black Hole* terjadi. Yaitu dengan memanggil program .tcl pada terminal dan menjalankan simulasi seperti pada gambar 4.6 di atas. Node sumber tidak akan menghiraukan RREP yang dikirimkan oleh node berbahaya dan memberi akses kirim pada node lain yang dapat mengirimkan paket pada node tujuan. Hal ini dibuktikan dengan jumlah *Packet Loss* yang lebih kecil dibandingkan pada saat terjadinya serangan.



Gambar 4.5 Nilai *Packet Loss* saat mekanisme *watchdog* terpasang

4.4 Perbandingan Analisa Dua Simulasi

Perbandingan QoS dilakukan untuk mempermudah perubahan informasi selama serangan berlangsung. Dengan melakukan perbandingan ini akan dapat dianalisa pergerakan serangan *Black Hole*. Seperti yang telihat pada tabel 4.1 di bawah ini hasil nilai *Throughput, Delay* dan *Packet Loss*.

Tabel 4.1a Hasil Tanpa Mekanisme *watchdog*

Hasil Tanpa Mekanisme *watchdog*

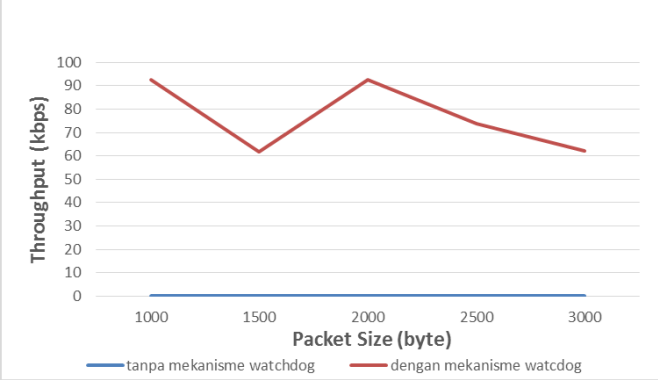
No.	Packet Size (byte)	Sent (packet)	Recv (packet)	Throughput (kbps)	Delay (s)	Packet Loss (%)
1	1000	248	0	0	0	100
2	1500	166	0	0	0	100
3	2000	248	0	0	0	100
4	2500	198	0	0	0	100
5	3000	166	0	0	0	100

Tabel 4.1b Hasil dengan Mekanisme *watchdog*

Hasil dengan Mekanisme watchdog

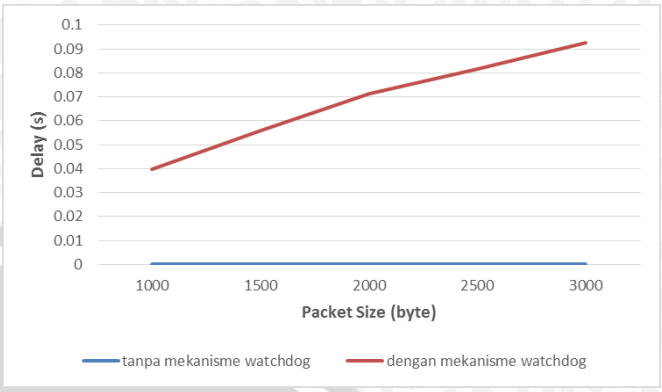
No.	Packet Size (byte)	Sent (packet)	Recv (packet)	Throughput (kbps)	Delay (s)	Packet Loss (%)
1	1000	248	248	92,5867	0,0398	0
2	1500	166	165	61,6	0,0559	0,6
3	2000	248	248	92,5867	0,0713	0
4	2500	198	198	73,92	0,0817	0
5	3000	166	166	61,9733	0,0926	0

Tabel 4.1a dan table 4.1b menjelaskan perbedaan besar ukuran paket yang dikirimkan pada saat simulasi berlangsung. Pada saat serangan *Black Hole* terjadi (table 4.1a), perbedaan terlihat jelas pada jumlah paket yang terkirim *(sent)* dan waktu jeda pengiriman (*Delay*) yang bernilai 0 karena paket tidak sampai kepada node tujuan. Jumlah paket yang terkirim berbeda – beda tergantung besar paket yang ditentukan. Sementara pada hasil dengan mekanisme *watchdog*, *delay* terjadi peningkatan nilai seiring bertambahnya besaran paket yang dikirimkan. Semakin besar ukuran paket, semakin tinggi *delay* yang terjadi.



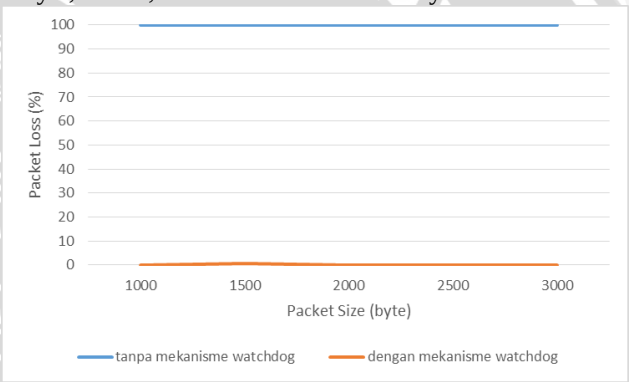
Gambar 4.6 Hasil *Throughput*

Gambar 4.9 menjelaskan hasil perbandingan nilai *throughput* antara serangan *black hole* dengan mekanisme *watchdog*. Pada saat serangan terjadi tanpa adanya pengawasan, *throughput* bernilai 0. Sementara pada saat *watchdog* dipasang, *throughput* bernilai 92,5867 kbps dengan 1000 byte, 61,6 kbps dengan 1500 byte, 92,5867 kbps dengan 2000 byte, 73,92 kbps dengan 2500 byte, dan 61,9733 kbps dengan 3000 byte.



Gambar 4.7 Hasil *Delay*

Untuk hasil perbandingan *delay* seperti pada Gambar 4.10, nilai *delay* pada saat serangan berlangsung bernilai 0 dibandingkan pada saat mekanisme *watchdog* telah terpasang, yaitu 0,0398 detik untuk 1000 byte, 0,0559 detik untuk 1500 byte, 0,0713 detik untuk 2000 byte, 0,0817 detik untuk 2500 byte, dan 0,0926 detik untuk 3000 byte.



Gambar 4.8 Hasil *Packet Loss*

Sedangkan untuk *packet loss*, nilai pada saat mekanisme *watchdog* terpasang lebih kecil dibandingkan sebelum terpasang mekanisme *watchdog.*

BAB V
KESIMPULAN DAN SARAN

5.1 Kesimpulan

Kesimpulan yang dapat diambil dari penelitian "*Simulasi Serangan Black Hole pada MANET (Mobile Ad Hoc Network)*" adalah sebagai berikut :

1. Serangan *Black Hole* dapat terjadi saat sebuah node mengirimkan RREP terhadap RREQ node sumber. Node berbahaya tersebut tidak akan mengirimkan paket pada *node tujuan.* Serangan *black hole* dapat terjadi dari dalam maupun luar lingkup membentukan jalur pengiriman paket.
2. Salah satu cara untuk menangani serangan *black hole* pada *routing protocol* AODV ialah dengan menggunakan mekanisme *watchdog* yang dapat memantau pergerakan pengiriman paket dan simulasi pada skripsi ini dinyatakan berhasil. Perbandingan performansi simulasi dengan dan tanpa mekanisme *watchdog* ialah sebagai berikut:

a) Hasil perbandingan pada saat tidak menggunakan mekanisme *watchdog* nilai *packet loss* sebesar nilai paket terkirim yaitu 100% untuk ukuran paket 1000 byte sementara pada saat penggunaan mekanisme *watchdog* nilai *packet loss* sebesar 0% untuk ukuran paket 1000 byte.
b) Nilai *throughput* saat tanpa mekanisme *watchdog* bernilai 0 kbps dengan ukuran paket 1000 byte karena paket tidak sampai ke tujuan, sedangkan nilai *throughput* dengan ukuran paket 1000 byte saat menggunakan mekanisme *watchdog* bernilai 92,5867 kbps.
c) Nilai delay saat tanpa mekanisme *watchdog* ialah 0 detik dengan ukuran paket 1000 byte, sedangkan saat menggunakan mekanisme *watchdog* sebesar 0,0398 detik. Semakin besar nilai *throughput* maka semakin kecil nilai *delay* saat paket dikirimkan.

3. Penerapan serangan *black hole* pada dunia nyata dapat berupa tidak sampainya *file* pada saat proses pengiriman ke tujuan. *File* tersebut akan membuat semua data baik gambar, suara atau film serta data hilang selama proses pengiriman. Sementara mekanisme *watchdog* merupakan mekanisme pengawasan yang berfungsi untuk mencegah gagalnya pengiriman *file*.

5.2 Saran

Adapun saran untuk pengembangan dan perbaikan penelitian selanjutnya mengenai "*Simulasi serangan Black Hole pada MANET (Mobile Ad Hoc Network)"* yaitu :

1. Dilakukannya pengembangan mekanisme *watchdog* dan mekanisme lainnya terhadap serangan *black hole*.

2. Dilakukannya percobaan jumlah *node* untuk mengetahui batas maksimum pada *routing protocol* dalam MANET, khusunya *routing protocol* AODV.

DAFTAR PUSTAKA

Anonim. (2010, January 25). *NS-2 Trace Formats*. Retrieved November 10, 2014, from NS-2: http://nsnam.isi.edu/nsnam/index.php/NS-2_Trace_Formats

Baumann, R. (2002). *AODV - Ad hoc On Demand Distance Vector Routing Protocol*. ETH Zürich: hypert.net.

Irawan, Dedy; Roestam, Rusdianto. (2011). Simulasi Model Jaringan *Mobile Ad Hoc* (MANET) dengan NS-3. *Konferensi Nasional Sistem dan Informatika*, KNS&I11-052.

K.A, Surana; S.B, Rathi; Mehatre, Snehal. (2012). Securing Black Hole Attack in Routing Protocol AODV in MANET with Watchdog Mechanism. *World Research Journal of Computer Architecture*, 19-23.

Jyoti; Kushwah, Rashmi. (2014). Performance Analysis of Black-Hole Attack in MANET. *International Journal of Computer Science and Information Technologies*, Volume 5

K.A, Surana; S.B, Rathi; Mehatre, Snehal. (2012). Securing Black Hole Attack in Routing Protocol AODV in MANET with Watchdog Mechanism. *World Research Journal of Computer Architecture*, Volume 1.

Lakhani, Kanika; bathla, Himani; Yadav, Rajesh. (2010). A Simulation Model to Secure the Routing Protocol AODV against Black-Hole Attack in MANET . *IJCSNS International Journal of Computer Science and Network Security*, Vol.10 No.5.

Nazaret, Ralph Nazaret; Erfianto, Bayu dan Yulianto, Fazmah Arif . (2011). Pengaruh Protokol Routing AODV dan TORA pada MANET terhadap Performansi Aplikasi VOIP. *Institut Teknologi Telkom.*

Outline, S. (2004). *Computer Networking (Jaringan Komputer).* Jakarta: Erlangga.

Pari, N. (2010). *Project Report Malicious Node Detection in MANET using Watchdog Mechanism.* MIT India.

Pari, N. (n.d.). *Project Report Malicious Node Detection in MANET Using Watchdog Mechanism.* MIT India.

Perkins, Charles E. ; Royer, Elizabeth M. ; Das, Samir R. and Marina, Mahesh K. (2001). Performance Comparison of Two On-Demand Routing Protocols for Ad Hoc Networks. *IEEE Personal Communications*, 1070-9916.

Sari, Riri Fitri ; Syarif, Abdusy dan Budiardjo, Bagio. (2008). Analisis Kinerja Protokol Routing *Ad Hoc On-Demand Distance Vector* (AODV) pada Jaringan *Ad Hoc Hybrid*: Perbandingan Hasil Simulasi dengan NS-2 dan Implementasi pada Testbed dengan PDA. *MAKARA, Teknologi*, 7-18.

Sidharta, Yonas dan Widjaja, Damar. (2013). Perbandingan Unjuk Kerja Protokol Routing *Ad Hoc On-Demand Distance Vector* (AODV) dan *Dynamic Source Routing* (DSR) pada Jaringan MANET. *Jurnal Teknologi*, 83 - 89.

Tahiliani, M. P. (2014, May 22). *Blackhole Attack in ns-2*. Retrieved October 29, 2014, from Mohit P. Tahiliani: http://mohittahiliani.blogspot.in/2014/05/blackhole-attack-in-ns-2.html

Talipov, E. (2009, October 24). *NS2: Adding Malicious Node to AODV*. Retrieved October 10, 2014, from elmurod.net: http://elmurod.net/en/index.php/archives/196

TS, P. (2013, March 25). *AWK Scripts for NS2 to process data from Trace Files* . Retrieved November 10, 2014, from Network Simulator: http://www.nsnam.com/2013/03/awk-scripts-for-ns2-to-process-data.html

TS, P. (2013, October 30). *Installing Network Simulator 2 (ns-2.35) in Ubuntu 13.10 (64 bit)* . Retrieved October 15, 2014, from Network Simulators: http://www.nsnam.com/2013/10/installing-network-simulator-2-ns-235.html

Ullah, Irshad and Rehman, Shoaib Ur. (2010). Analysis of Black Hole Attack on MANETs Using Different MANET Routing Protocols. *Blekinge Institute of Technology*, MEE 10:62.