



PENERAPAN PROTOKOL *ROUTING* PROAKTIF OLSR PADA JARINGAN *MOBILE AD HOC NETWORK* (MANET) MENGGUNAKAN RASPBERRY PI UNTUK *TRANSFER FILE*

Rico Dwi Kurniawan^{*1}, Muh. Yamin², LM. Fid Aksara³

^{*1,2,3}Jurusan Teknik Informatika, Fakultas Teknik, Universitas Halu Oleo, Kendari
e-mail: ^{*1}ricodwikurniawan@yandex.com, ²putra0683@gmail.com, ³kapasole@gmail.com

Abstrak

Pengguna jaringan nirkabel (*wireless*) semakin banyak digunakan dibandingkan jaringan kabel. Salah satu pengembangan jaringan *wireless* yaitu adanya teknologi bernama jaringan *Ad Hoc*. Jaringan *Ad Hoc* merupakan solusi dimana perangkat bergerak (*mobile*) seperti laptop, tablet, *smartphone*, dan sebagainya dapat saling berkomunikasi satu sama lain tanpa dukungan infrastruktur jaringan berupa *access point* atau sarana pendukung transmisi data. *Mobile Ad Hoc Network* (MANET) merupakan jaringan *wireless multihop* yang terdiri dari kumpulan *mobile node* yang bersifat dinamik.

Seiring dengan perkembangan teknologi, jaringan MANET mulai dikembangkan sebagai antisipasi jika infrastruktur pusat suatu jaringan tidak dapat digunakan. Pada jaringan ini tiap *node*-nya bersifat sebagai *router* yang berfungsi untuk menentukan rute yang terdekat antar *node* yang berkomunikasi. Tiap *node* pada jaringan MANET juga bersifat *mobile* sehingga memungkinkan untuk bebas bergerak, namun hal ini berdampak pada topologi yang terus berubah seiring dengan pergerakan *node*.

Pada pengujian yang dilakukan *file transfer* diterapkan pada skenario pengujian infrastruktur MANET dengan tiga *node* dengan dua kondisi, yaitu kondisi ketika tidak ada gangguan pada jaringan MANET dan kondisi ketika terjadi gangguan pada jaringan MANET. Pengujian *file transfer* pada jaringan MANET dapat bekerja sesuai dengan sifat dan kondisi jaringan yang dimiliki MANET, namun dari hasil analisis kecepatan *transfer file* akan menurun seiring dengan bertambahnya *node gateway* pada rute komunikasi antara *node server* dan *client*.

Kata Kunci—MANET, *File Transfer Protocol* (FTP), *Optimized Link Stated Routing* (OLSR)

Abstract

The wireless network's users are increasingly used compared to wired networks. One of the developments of wireless networks is the existence of a technology called Ad Hoc network. Ad Hoc network is a solution where mobile devices such as laptops, tablets, smartphones, etc. can communicate with each other without the support of network infrastructure in the form of access points or means of supporting data transmission. Mobile Ad Hoc Network (MANET) is a multi-hop wireless network consisting of a collection of dynamic mobile nodes.

Along with the development of technology, MANET began to be developed in anticipation if the central infrastructure of a network cannot be used. In this network, each node is a router that functions to determine the route that is closest to the nodes that communicate. Each node on the MANET network is also mobile so it is possible to be free to move, but this has an impact on the topology which keeps changing along with the movement of nodes.



In the test carried out file transfer is applied to the scenario of MANET infrastructure testing with three nodes with two conditions, namely the condition of the absence of interference with the MANET network and the condition when there is interference with the MANET network. file transfer testing on the MANET network can work in accordance with nature and network conditions that MANET has, but from the results of the analysis, the file transfer speed will decrease as the gateway nodes increase on the communication route between the node server and the client.

Keywords—MANET, File Transfer Protocol (FTP), Optimized Link Stated Routing (OLSR)

1. PENDAHULUAN

Perkembangan dunia telekomunikasi saat ini sangat pesat seiring dengan peningkatan kebutuhan layanan yang cepat dan efisien. Begitu juga dengan komunikasi data, mulai dari koneksi antar dua komputer hingga jaringan komputer. Jaringan komputer saat ini merupakan suatu layanan yang sangat dibutuhkan. Jaringan komputer mempunyai manfaat yang lebih dibandingkan dengan komputer yang berdiri sendiri. Jaringan komputer memungkinkan pemakaian secara bersama data, perangkat lunak, dan peralatan, sehingga kelompok kerja dapat berkomunikasi lebih efektif dan efisien. Dunia teknologi saat ini mengalami perkembangan yang cepat, terlebih lagi pada teknologi internet [1].

Salah satu alternatif solusi dari permasalahan tersebut adalah dengan mengimplementasikan *Wireless Distribution System* (WDS) pada perangkat *access point*. WDS merupakan sistem untuk mengembangkan jaringan internet nirkabel tanpa harus menggunakan kabel sebagai *backbone* untuk *access point* melainkan memanfaatkan jalur nirkabel dari *access point* tersebut. Analisis *Quality of Service* (QoS) WDS menggunakan parameter *Jitter*, *Throughput*, *Delay* dan *Packet Loss*. Berdasarkan analisis QoS tersebut maka akan diketahui perbandingan *access point* yang menggunakan *backbone* kabel sehingga dapat diketahui masing-masing keunggulan dan kekurangan penerapan metode WDS dan tanpa menggunakan WDS pada jaringan nirkabel [2].

Dalam suatu jaringan, konektivitas beberapa *node* dapat menghilang karena jarak yang terlalu jauh dan muncul *node* baru dalam satu waktu dikarenakan pergerakan *node-node* tersebut. Suatu bentuk teknik pemodelan yang digunakan untuk menggambarkan hal tersebut yaitu *Random Waypoint Mobility Model*. *Node* pada MANET tidak hanya berperan sebagai

pengirim atau penerima data saja, namun dapat juga difungsikan sebagai penghubung *node* yang lain. Untuk mengatur seluruh proses *routing* pada topologi MANET tidak memerlukan *router/node*, karena setiap *device* berfungsi sebagai *router* untuk menentukan arah yang akan ditentukan, sehingga pada proses komunikasi pada jaringan MANET sangat memerlukan protokol yang tepat dan cepat agar *node* dapat mengirimkan paket data yang dibutuhkan oleh jaringan MANET tersebut [3].

Seiring dengan perkembangan teknologi, jaringan, MANET mulai dikembangkan sebagai antisipasi jika infrastruktur pusat suatu jaringan tidak dapat digunakan. Melalui jaringan MANET, pengguna tidak memerlukan adanya infrastruktur pusat sebagai media penyalur jalur komunikasi karena MANET bersifat *stand-alone*. Pada jaringan ini tiap *node*-nya bersifat sebagai *router* yang berfungsi untuk menentukan rute yang terdekat antar *node* yang berkomunikasi. Tiap *node* pada jaringan MANET juga bersifat *mobile* sehingga memungkinkan untuk bebas bergerak, namun hal ini berdampak pada topologi yang terus berubah seiring dengan pergerakan *node*. Sampai saat ini MANET masih menjadi bahan penelitian untuk penerapannya pada berbagai kegiatan manusia.

2. METODE PENELITIAN

2.1 Jaringan Komputer

Jaringan komputer adalah sistem yang terdiri dari komputer - komputer serta piranti-piranti yang saling terhubung sebagai satu kesatuan. dengan dihubungkan piranti-piranti tersebut, alhasil dapat saling berbagi sumber daya satu piranti dengan piranti lainnya [4].

2.2 File Transfer Protocol (FTP)

FTP atau *File Transfer Protocol* merupakan protokol internet yang digunakan untuk urusan pengiriman data dalam jaringan

komputer, seperti *upload* dan *download file* yang dilakukan oleh *FTP client* dan *FTP server*. Layanan FTP bisa diatur menjadi *FTP public*, dimana semua orang bisa mengakses data-data yang ada di *server* FTP dengan mudah. Selain dapat diatur menjadi *FTP public*, layanan FTP ini juga bisa diatur agar tidak semua orang dapat mengakses data-data yang ada di *server*, jadi hanya pengguna terdaftar saja yang memiliki izin untuk mengakses data-data tersebut. FTP berkerja menggunakan salah satu protokol yang dapat diandalkan untuk urusan komunikasi data antara *client* dan *server*, yaitu protokol *TCP* (yang menggunakan *port* nomor 21). *Port* 21 ini digunakan untuk mengirimkan *command* (perintah). Oleh karena *port* 21 dimaksudkan khusus untuk mengirimkan *command*, maka *port* ini sering juga disebut dengan nama *command port*. Dengan adanya protokol ini, antara *client* dan *server* dapat melakukan sesi komunikasi sebelum pengiriman data berlangsung.

2.3 Protokol MANET

Terdapat berbagai jenis protokol *routing* untuk MANET yang secara keseluruhan dapat dibagi menjadi beberapa kelompok, yaitu :

a. Proactive Routing

Algoritma ini akan mengelola daftar tujuan dan rute terbaru masing-masing dengan cara mendistribusikan *routing table* ke seluruh jaringan, sehingga jalur lalu lintas (*traffic*) akan sering dilalui oleh *routing table* tersebut. Hal ini akan memperlambat aliran data jika terjadi restrukturisasi *routing table*.

Beberapa contoh algoritma *proactive routing* adalah :

1. Babel
2. B.A.T.M.A.N–*Better Approach to Mobile Ad Hoc Network*
3. DSDV–*Highly Dynamic Destination Sequenced Distance Vector routing protocol*
4. HSR–*Hierarchical State Routing Protocol*
5. IARP–*Intrazone Routing Protocol*
6. LCA–*Linked Cluster Architecture*
7. WAR–*Witness Aided Routing*
8. OLSR–*Optimized Link State Routing Protocol*

b. Reactive Routing

Tipe ini akan mencari rute (*on demand*) dengan cara membanjiri jaringan dengan

paket *router request*, sehingga dapat menyebabkan jaringan akan penuh (*clogging*). Beberapa contoh algoritma *reactive routing* adalah :

1. SENCAS
2. *Reliable Ad Hoc on Demand Distance Vector Routing Protocol*
3. *Ant-Based Routing Algorithm for Mobile Ad Hoc Network*
4. *Admission Control Enabled on Demand Routing (ACOR)*
5. *Ariadne*
6. *Associativity Based Routing*
7. *Ad Hoc on Demand Distance Vector (AODV)*
8. *Ad Hoc on Demand Multipath Distance Vector*
9. *Backup Source Routing*
10. *Dynamic Source Routing (DSR)*
11. *Flow State in the Dynamic Source Routing*
12. *Dynamic MANET On Demand Routing (DYMO)*

c. Flow Oriented Routing

Tipe protokol ini mencari rute dengan mengikuti aliran yang disediakan. Salah satu pilihan adalah dengan *unicast* secara terus-menerus ketika meneruskan data saat mempromosikan *link* baru. Beberapa kekurangan tipe protokol ini adalah membutuhkan waktu yang lama untuk mencari rute yang baru. Beberapa protokol yang memiliki tipe ini adalah :

1. *Interzone Routing Protocol (IERP)*
2. *Lightweight Underlay Network Ad Hoc Routing (LUNAR)*
3. *Signal Stability Routing (SSR)*

d. Hybrid Routing

Tipe protokol ini menggabungkan antara *proactive routing* dengan *reactive routing*. Protokol untuk tipe ini adalah :

1. *Hybrid Routing Protocol for Large Scale MANET (HRPLS)*
2. *Hybrid Wireless Mesh Protocol (HWMP)*
3. *Zone Routing Protocol (ZRP)*

2.4 Optimized Link Stated Routing (OLSR)

OLSR merupakan salah satu jenis dari *proactive routing protocol* yang biasa digunakan dalam jaringan *Ad Hoc*. Protokol ini melakukan pertukaran pesan secara periodik dalam rangka menjaga informasi topologi jaringan yang ada pada setiap *node*.

Protokol OLSR mewarisi sifat kestabilan dari *link state algorithm*. Berdasarkan sifat proaktifnya, protokol ini dapat menyediakan rute dengan segera apabila dibutuhkan. Dalam sebuah *link state protocol* yang murni, setiap *node* tetangga akan dideklarasikan dan dibanjiri dengan paket informasi yang akan memenuhi seluruh jaringan. OLSR merupakan sebuah optimasi dari *link state protocol* yang biasa digunakan dalam MANET [5].

Langkah pertama dari optimasi tersebut adalah mengurangi ukuran dari paket kontrol, daripada membanjiri paket kontrol tersebut pada setiap jalur, OLSR lebih memilih sejumlah jalur dengan *node* tetangga yang disebut dengan *multipoint relay selector*. Langkah kedua, OLSR meminimalisir pembanjiran paket kontrol pada jaringan dengan menggunakan *MPR* untuk menghantarkan paket-paket tersebut. Teknik ini akan mengurangi secara signifikan jumlah dari transmisi ulang yang akan membanjiri jaringan dengan prosedur *broadcast* [6].

Protokol OLSR dirancang untuk dapat bekerja pada kondisi yang terdistribusi atau selalu bergerak serta tidak memerlukan adanya pengaturan secara terpusat. Selain itu OLSR juga tidak memerlukan transmisi yang bagus dalam mengirimkan paket-paket kontrolnya. Setiap *node* mengirimkan paket kontrolnya masing-masing secara periodik sehingga dapat mentoleransi terjadinya *loss* dari beberapa paket pada saat-saat tertentu akibat dari tubrukan data ataupun akibat gangguan transmisi lainnya. Setiap paket kontrol yang dikirimkan akan diberikan *sequence number* (nomor urut) yang dapat menandakan tingkat baru tidak paket tersebut. OLSR menggunakan *multihop routing* dimana setiap *node* menggunakan informasi *routing* terbaru yang ada pada *node* tersebut dalam mengantarkan sebuah paket informasi. Sehingga, walaupun sebuah *node* bergerak ataupun berpindah tempat maka pesan yang dikirimkan padanya akan tetap dapat diterima [7].

2.5 Raspberry Pi

Raspberry Pi, sering disingkat dengan nama Raspi, adalah komputer papan tunggal (*Single-Board Circuit; SBC*) yang seukuran

dengan kartu kredit yang dapat digunakan untuk menjalankan program perkantoran, permainan komputer, dan sebagai pemutar media hingga video beresolusi tinggi. Raspberry Pi dikembangkan oleh yayasan nirlaba, Raspberry Pi Foundation, yang digawangi sejumlah pengembang dan ahli komputer dari Universitas Cambridge, Inggris. [8].

Raspberry Pi diawali dari keinginan untuk mencetak pemrogram generasi baru. Seperti disebutkan dalam situs resmi Raspberry Pi Foundation, waktu itu Eben Upton, Rob Mullins, Jack Lang, dan Alan Mycroft, dari Laboratorium Komputer Universitas Cambridge memiliki kekhawatiran melihat kian turunnya keahlian dan jumlah siswa yang hendak belajar ilmu komputer. Mereka lantas mendirikan yayasan Raspberry Pi bersama dengan Pete Lomas dan David Braben pada 2009.

Raspberry Pi memiliki dua model: model A dan model B. Secara umum Raspberry Pi Model B memiliki kapasitas penyimpanan RAM sebesar 512 MB. Perbedaan model A dan B terletak pada modul penyimpanan yang digunakan. Model A menggunakan penyimpanan sebesar 256 MB dan penyimpanan model B sebesar 512 MB. Selain itu, model B sudah dilengkapi dengan *port* Ethernet (untuk LAN) yang tidak terdapat di model A. Desain Raspberry Pi didasarkan pada SoC (*system-on-a-chip*) Broadcom BCM2835, yang telah menanamkan prosesor ARM1176JZF-S dengan 700 MHz, GPU VideoCore IV, dan RAM sebesar 256 MB (model B). Penyimpanan data tidak didesain untuk menggunakan cakram keras atau *solid-state drive*, melainkan mengandalkan kartu penyimpanan tipe SD untuk menjalankan sistem dan sebagai media penyimpanan jangka panjang [9].

3. HASIL DAN PEMBAHASAN

Untuk pengujian menggunakan 3 *node* yang telah terkonfigurasi pengalamatan IP-nya, kemudian OLSR sebagai protokol *routing* dijalankan pada tiap *node*. Fungsi protokol *routing* inilah yang menjadi ciri utama dari MANET, karena dengan adanya protokol *routing* maka tiap *node* MANET dapat melakukan *routing* untuk mendeteksi *node* MANET sekitar yang terdeteksi sinyalnya.

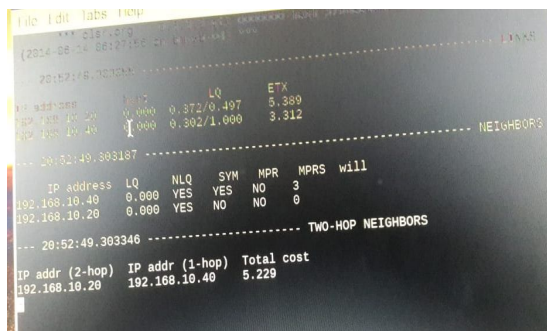
Saat beberapa *node* MANET ini terhubung maka terciptalah satu infrastruktur jaringan MANET.

Pada skenario pengujian MANET 3 *node*, penentuan *IP address* pada masing-masing *node* dilakukan. *Node 1* dengan *IP address* 192.168.10.1 dan *node 2* dengan *IP address* 192.168.10.20, serta *node 3* dengan alamat *IP address* 192.168.10.40, yang diletakkan dengan posisi ketiga *node* saling berjauhan, seperti yang terlihat pada Tabel 1.

Tabel 1 Alamat IP Setiap *Node*

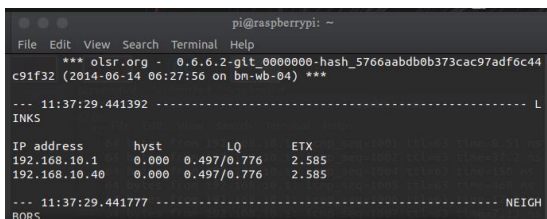
IP Node 1	IP Node 2	IP Node 3
192.168.10.1	192.168.10.20	192.168.10.40

Setelah itu, ketika selesai melakukan tes koneksi kami kemudian melakukan pengujian *protocol* OLSR dengan menjalankan OLSRD pada tiap *node* yang ditunjukkan oleh Gambar 1.



Gambar 1 *Output Normal Protocol OLSR* pada *Node 1*

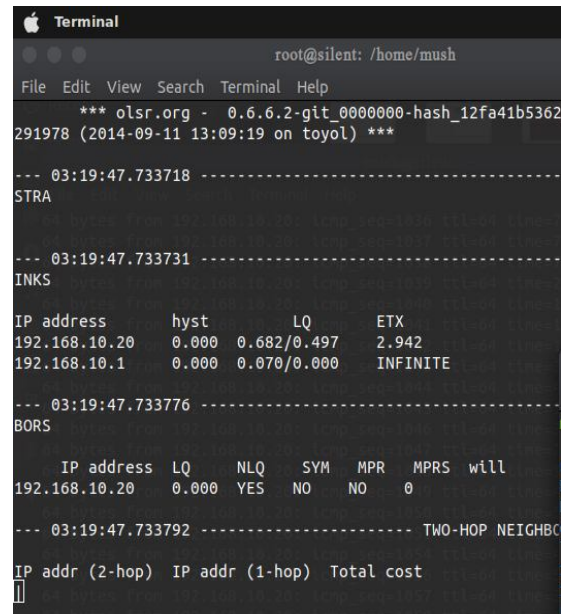
Gambar 2 menunjukkan bahwa *node 1* sudah mendeteksi IP *node 3* dan IP *node 2*.



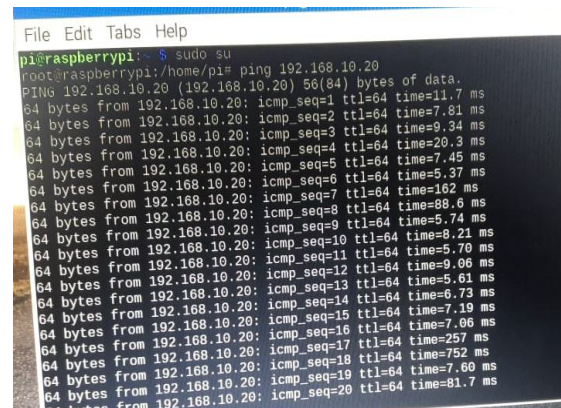
Gambar 2 *Output Normal Protocol OLSR* pada *Node 2*

Gambar 3 menunjukkan bahwa *node 2* sudah mendeteksi IP *node 1* dan IP *node 3*. Pada Gambar 3 juga menunjukkan bahwa *node 3* sudah mendeteksi IP *node 2* dan IP *node 1*. Hal ini menandakan bahwa *protocol* OLSR sudah berjalan semestinya pada tiap *node*. Gambar 4 adalah gambar hasil dari

perintah *test ping* dari masing-masing *node* yang saling terhubung pada jaringan MANET.

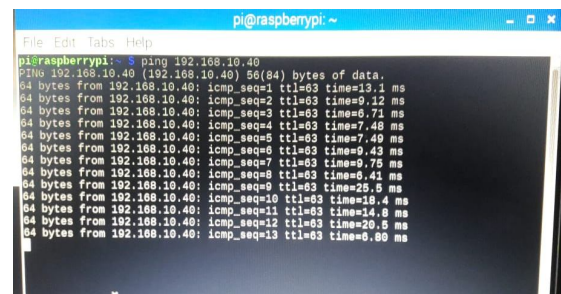


Gambar 3 *Output Normal Protocol OLSR* pada *Node 3*



Gambar 4 Tampilan *Test Ping* dari *Node 1* ke *Node 2*

Gambar 5 adalah hasil perintah *ping* yang menunjukkan bahwa *node 1* mendapat balasan dari *node 2*, hal ini membuktikan bahwa *node 1* tersambung dengan *node 2*.



Gambar 5 Tampilan *Test Ping* dari *Node 1* ke *Node 3*

Gambar 5 menunjukkan hasil perintah *ping* pada *command line* terminal pada *node 1*. Dari hasil tersebut terlihat bahwa kondisi *node 1* mempunyai jalur dengan *node 3* melalui *gateway node 2*. Hasil perintah *ping* yang menunjukkan bahwa *node A* mendapat balasan dari *node B*, hal ini membuktikan bahwa *node 1* tersambung dengan *node 3* karena bantuan *node gateway 2*. Jika pada kondisi ini *node 2* mengalami gangguan atau sengaja dinonaktifkan, sedangkan *node 1* dan *node 3* tetap statis di tempatnya maka jalur *routing* antara *node 1* dan *node 2* akan terputus dan menciptakan dua infrastruktur MANET kecil yang terdiri dari satu *node* yang masing-masing adalah *node A* dan *node C*. kemudian hasil *test ping* dari *node 3* ke *node 2* ditunjukkan Gambar 6.

```

mush@silent: ~
└─$ ping 192.168.10.20
PING 192.168.10.20 (192.168.10.20) 56(84) bytes of data:
64 bytes from 192.168.10.20: icmp_seq=1 ttl=64 time=1.93 ms
64 bytes from 192.168.10.20: icmp_seq=2 ttl=64 time=2.08 ms
64 bytes from 192.168.10.20: icmp_seq=3 ttl=64 time=4.83 ms
64 bytes from 192.168.10.20: icmp_seq=4 ttl=64 time=68.3 ms
64 bytes from 192.168.10.20: icmp_seq=5 ttl=64 time=153 ms

```

Gambar 6 Tampilan *Test Ping* dari *Node 3* ke *Node 2*

Gambar 7 adalah hasil perintah *ping* yang menunjukkan bahwa *node 3* mendapat balasan dari *node 2*, hal ini membuktikan bahwa *node 3* tersambung dengan *node 2*.

```

mush@silent: ~
└─$ ping 192.168.10.1
PING 192.168.10.1 (192.168.10.1) 56(84) bytes of data:
64 bytes from 192.168.10.1: icmp_seq=1 ttl=63 time=20.2 ms
64 bytes from 192.168.10.1: icmp_seq=2 ttl=63 time=24.1 ms
64 bytes from 192.168.10.1: icmp_seq=3 ttl=63 time=9.26 ms

```

Gambar 7 Tampilan *Test Ping* dari *Node 3* ke *Node 1*

Gambar 7 menunjukkan hasil perintah *ping* pada *command line* terminal pada *node 3*. Dari hasil tersebut terlihat bahwa kondisi *node 3* mempunyai jalur dengan *node 1* melalui *gateway node 2*. Hasil perintah *ping* yang menunjukkan bahwa *node 3* mendapat balasan dari *node 1*, hal ini membuktikan bahwa *node 3* tersambung dengan *node 1* karena bantuan *node gateway 2*. Jika pada kondisi ini *node 2* mengalami gangguan atau sengaja dinonaktifkan, sedangkan *node 3* dan *node 1* tetap statis di tempatnya maka jalur *routing* antara *node 3* dan *node 2* akan terputus dan menciptakan dua infrastruktur MANET kecil yang terdiri dari satu *node* yang masing-masing adalah *node 3* dan *node 1*.

Pada pengujian ini *file transfer* diterapkan pada skenario pengujian infrastruktur MANET dengan tiga *node* dengan dua kondisi, yaitu kondisi ketikan tidak ada gangguan pada jaringan MANET dan kondisi ketika terjadi gangguan pada jaringan MANET. Langkah awal yakni dengan mengaktifkan FTP *server* ProFTPD dengan melakukan perintah *service* ProFTPD seperti Gambar 8.

```

root@raspberrypi:/home/pi# service proftpd
Usage: /etc/init.d/proftpd {start|status|force-start|stop|force-stop|reload|restart|force-reload|check-config}
root@raspberrypi:/home/pi# service proftpd status
● proftpd.service - LSB: Starts ProFTPD daemon
   Loaded: loaded (/etc/init.d/proftpd; generated; vendor preset: enabled)
   Active: active (running) since Sun 2018-12-02 20:52:03 WITA; 1min 35s ago
     Docs: man:systemd-sys-generator(8)
   Process: 428 ExecStart=/etc/init.d/proftpd start (code=exited, status=0/SUCCESS)
   CGroup: /system.slice/proftpd.service
           └─553 proftpd: (accepting connections)
Dec 02 20:51:59 raspberrypi systemd[1]: Starting LSB: Starts ProFTPD daemon...
Dec 02 20:52:03 raspberrypi proftpd[428]: Starting ftp server: proftpd.
Dec 02 20:52:03 raspberrypi systemd[1]: Started LSB: Starts ProFTPD daemon.

```

Gambar 8 Tampilan Pengaktifan FTP *Server* ProFTPD

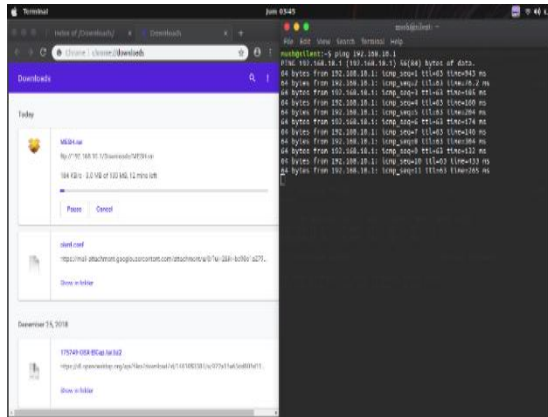
ProFTPD merupakan aplikasi FTP *server* yang digunakan dalam pengujian ini dan yang bertindak sebagai *server* yaitu *node 1* dengan IP *address* 192.168.10.1.

Berbeda dengan pengujian sebelumnya, pada pengujian ini *file transfer* oleh *node server* akan sampai pada *node client* melalui perantara *node gateway* yang menghubungkan jalur antara kedua *node* tersebut. Langkah yang dilakukan sama seperti pada pengujian sebelumnya yaitu setelah OLSR pada semua *node* diaktifkan, FTP *server* pada *node 1* dijalankan. *Node 3* sebagai *client* akan mengakses *file* pada FTP *server* *node 1*. Saat *client* sudah menentukan *file* yang dipilih, selanjutnya *file* tersebut akan diunduh dan proses *file transfer* dimulai. Gambar 9 menunjukkan tampilan *directory file server*.

Name	Size	Date Modified
2018-07-10-095052_1360x768_scrot.png	115 KB	7/10/18, 9:50:00 AM
2018-12-02-072921_1360x768_scrot.png	1.0 MB	12/2/18, 7:28:00 AM
2018-12-02-192547_1360x768_scrot.png	1.0 MB	12/2/18, 7:28:00 PM
C	2.7 MB	12/01/18, 8:00:00 AM
cam.jpg	2.7 MB	12/01/18, 8:00:00 AM
Desktop/		12/2/18, 10:34:00 AM
Documents/		12/2/18, 7:51:00 PM
Downloads/		12/2/18, 10:34:00 AM
Hydrol.inh/		7/8/18, 2:27:00 PM
Hydrol.inh.py	2.7 kB	6/29/18, 5:38:00 PM
Music/		11/29/17, 8:00:00 AM
olicon/files/		10/30/18, 0:14:00 PM
Pictures/		11/29/17, 8:00:00 AM
Public/		11/29/17, 8:00:00 AM
python_games/		11/29/17, 8:00:00 AM
RPI_Cam_Web_Interface-master/		12/1/18, 8:00:00 AM
Templates/		11/29/17, 8:00:00 AM
thinclient_drives/		12/1/18, 8:00:00 AM
Videos/		11/29/17, 8:00:00 AM

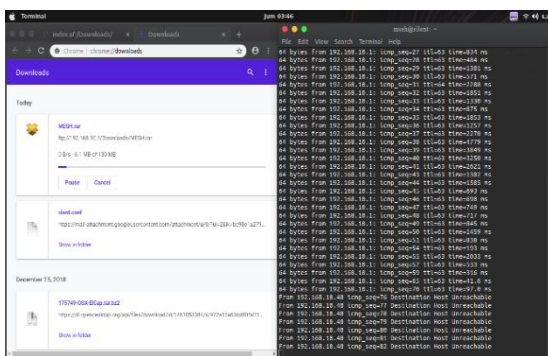
Gambar 9 Tampilan *Directory File* FTP *Server*

Gambar 9 merupakan tampilan *directory file FTP server* yang dimiliki oleh *node 1*. Cara mengaksesnya dengan membuka lewat aplikasi *browser* lalu tuliskan perintah *FTP://192.168.10.1* yang merupakan *IP address node 1* yang bertindak sebagai *server*. Gambar 10 menunjukkan tampilan *transfer file*.



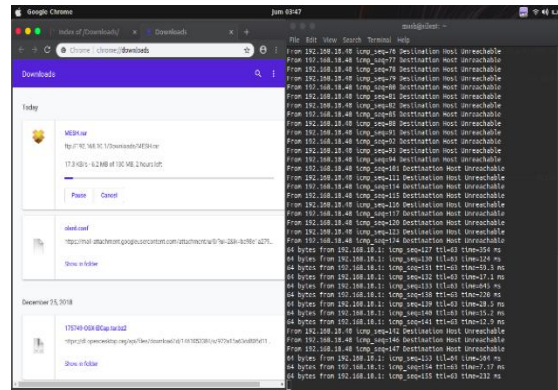
Gambar 10 Tampilan Proses *Transfer File* Tak Ada Gangguan

Dari hasil pengujian pada skenario ini dapat diambil kesimpulan bahwa *transfer file* yang terjadi pada *node client-server* berlangsung tanpa ada gangguan. *Node 3* bertindak sebagai *client* mengunduh salah satu *file* yang berada di *directory file server node 1* yang bertindak sebagai *server*. Gambar 11 menunjukkan tampilan proses *transfer file* saat ada gangguan.



Gambar 11 Tampilan Proses *Transfer File* Mengalami Gangguan

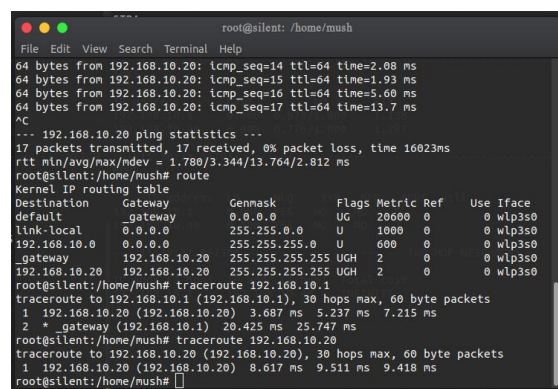
Namun pada saat proses *transfer file* sedang berjalan, terdapat gangguan pada jaringan MANET, yakni *node 3* berada di luar jangkauan *server* sehingga proses unduhan terhenti sementara ditandai dengan kecepatan *transfer file* menurun hingga berhenti sementara. Gambar 12 menunjukkan tampilan proses *transfer file* berlanjut.



Gambar 12 Tampilan Proses *Transfer File* Berlanjut

Setelah jaringan MANET kembali normal, antara *node 1* dan *node 3* kembali terhubung, dilakukan proses *transfer file* ulang, *file* yang di-*transfer* akan melanjutkan kembali proses *download* dari *server*.

Routing Table adalah tabel yang memuat seluruh informasi *IP address* dan *interface router* yang lain sehingga *router* yang satu dengan lainnya bisa berkomunikasi, sedangkan perintah *traceroute* digunakan untuk menunjukkan rute yang dilewati paket untuk mencapai tujuan. Ini dilakukan dengan mengirim pesan *Internet Control Message Protocol (ICMP) Echo Request* ke tujuan dengan nilai *Time to Live* yang semakin meningkat. Rute yang ditampilkan adalah daftar *interface router* (yang paling dekat dengan *host*) yang terdapat pada jalur antara *host* dan tujuan). Gambar 13 menunjukkan tampilan *Routing Table* dan *traceroute*.



Gambar 13 Tampilan Proses *Routing Table* dan *Traceroute* oleh *Node 3*

Gambar 13 menunjukkan hasil *routing table* yang dilakukan oleh *node 3* serta perintah *traceroute* ke *IP address node 2* yaitu 192.168.10.20 dan *traceroute* ke *IP address node 1* yaitu 192.168.10.1. *Traceroute* yang

dilakukan oleh *node* 3 ke IP address *node* 1 menunjukkan rute paket yang dilewati melalui *node* 2 dengan IP address 192.168.10.20 menuju ke *node* 1 dengan IP address 192.168.10.1. *Traceroute* yang dilakukan *node* 3 ke IP address *node* 2 menunjukkan rute paket langsung ke *node* 2 dengan IP address 192.168.10.20.

4. KESIMPULAN

Berdasarkan hasil implementasi dan pengujian penerapan jaringan MANET untuk *file transfer* maka dapat diambil kesimpulan sebagai berikut :

- a. Implementasi untuk membangun infrastruktur jaringan MANET pada *embedded system* Raspberry Pi dapat bekerja dengan baik namun harus disertai dengan beberapa alat tambahan agar MANET dapat berfungsi dengan kondisi spesifikasi dari Raspberry Pi.
- b. Pengujian fungsi kerja Raspberry Pi untuk melakukan *routing* pada jaringan MANET dapat bekerja sesuai dengan kondisi *routing* jaringan lokal pada umumnya.
- c. Kinerja Raspberry Pi pada pengujian letak dan pergerakan *node* bekerja sesuai dengan sifat *mobile* pada MANET, namun masih terbatas ruang geraknya dikarenakan secara *default* Raspberry Pi tidak menyediakan *built-in network card* sebagai media pemancar dan penyaring sinyal *WiFi*.
- d. Pengujian *file transfer* pada jaringan MANET dapat bekerja sesuai dengan sifat dan kondisi jaringan yang dimiliki MANET, namun dari hasil analisis kecepatan transfer *file* akan menurun seiring dengan bertambahnya *node gateway* pada rute komunikasi antara *node server* dan *client*.
- e. Jika terjadi gangguan pada proses *transfer file* berlangsung antara *node server* dan *client* maka proses *transfer file* akan terhenti, dan jika terkoneksi kembali antara *client* dan *server* maka proses *transfer file* akan berlanjut.

FTP sebagai media komunikasi *file transfer* skala besar (*multi client*) yang diterapkan pada jaringan MANET akan sangat berguna jika digunakan pada kondisi darurat dimana saat infrastruktur jaringan pusat mengalami kerusakan.

5. SARAN

Adapun saran dalam penelitian ini yaitu diharapkan ke depannya pengujian yang dilakukan dapat menggunakan peralatan yang lebih baik lagi sehingga dapat mencapai hasil yang maksimal. dikarenakan ketidakstabilan perangkat yang digunakan ketika melakukan pengujian.

DAFTAR PUSTAKA

- [1] A. Purnomo and H. Kurniawa, "Implementasi Protokol Routing Destination Sequenced Distance-Vector (DSDV) pada Jaringan Wireless Ns-2 Dengan Nam," 2017. [Online]. Available: <https://docplayer.info/30334364-Implementasi-protokol-routing-destination-sequenced-distance-vector-dsdv-pada-jaringan-wireless-ns-2-dengan-nam.html>. [Accessed: 07-May-2018].
- [2] A. Putra, F. A. Yulianto, and A. Herutomo, "Analisis Performansi dan Perbandingan Routing Protocol OLSR dan ZRP pada Vehicular Ad Hoc," *e-Proceeding Eng.*, Vol. 2, No. 2, pp. 6285–6292, 2015.
- [3] B. Benardi, "Analisa Unjuk Kerja Jaringan Nirkabel Ad Hoc dalam Beberapa Situasi yang Berbeda Ditinjau Dari Sudut Pandang Routing," Universitas Mercu Buana, 2009.
- [4] M. I. D. D. Perdana and R. M. Negara "Analisis Perbandingan Kinerja Protokol Routing DSDV dan OLSR untuk Perubahan Kecepatan Mobilitas pada Standar IEEE 802.11ah," *J. Infotel*, Vol. 8, No. 2, pp. 100–106, 2016.
- [5] S. D. Anggraini, K. Nugroho, and E. F. Cahyadi "Analisis Perbandingan Performansi Protokol Routing AODV dan DSR pada Mobile Ad Hoc Network (MANET)," *2nd Semin. Nas. IPTEK Terap.*, pp. 112–118, 2017.
- [6] Sukaridhoto, *Buku Jaringan Komputer I*. Surabaya: Politeknik Elektronika Negeri Surabaya (PENS), 2014.
- [7] Supriyanto, *Jaringan Dasar Jaringan*

- Dasar*. Malang: Kementerian Pendidikan & Kebudayaan, 2013.
- [8] Y. Ge, T. Kunz, and L. Lamont, "Quality of Service Routing in Ad Hoc Networks Using OLSR," in *Proceedings of the 36th Annual Hawaii International Conference on System Sciences, HICSS 2003*, 2003.
- [9] Z. A. Aris, A. Muttaqin, and W. Kurniawan, "Penerapan Teknologi Jaringan Mobile Ad Hoc Network (MANET) untuk File Transfer." [Online]. Available: https://www.academia.edu/7985470/Penerapan_Teknologi_Jaringan_Mobile_Ad_Hoc_Network_MANET_Untuk_File_Transfer. [Accessed: 07-Jan-2019].
-

