

Perbandingan Kinerja ION-DTN Dan IBR-DTN Menggunakan *Raspberry Pi* Sebagai Router Delay Tolerant Network

Ewi Ismaredah , Oktaf Brilliant Kharisma, Ahmad Faizal, Mulyono, Dian Mursyitah, Hasdi Radiles, Teguh Wibowo

Universitas Islam Negeri Sultan Syarif Riau

e-mail: ewi.ismaredah@uin-suska.ac.id, ahmad.faizal@uin-suska.ac.id, dmursyitah@uin-suska.ac.id

Abstrak

Pada saat ini Internet merupakan suatu kebutuhan. Hampir semua orang menggunakan Internet dalam kegiatan sehari-hari seperti mengirim file ke suatu tempat yang jauh untuk kepentingan pekerjaan. Namun tidak semua tempat memiliki akses yang memadai untuk akses Internet, sehingga pengiriman file terhambat. Terhambatnya pengiriman yang disebabkan oleh akses Internet yang tidak memadai dikarenakan protokol Internet yang umum digunakan seperti TCP/IP tidak dapat mengirimkan file dengan waktu tunda (delay) yang tinggi, tetapi, dengan adanya DTN (Delay Tolerant Network) Internet bukan lagi suatu hal yang tidak mungkin karena DTN merupakan protokol jaringan menyediakan konektivitas Internet yang kurang memadai. Karena DTN mempunyai kelebihan dapat mengirim file dalam kondisi intermitten connectivity dan non-intermitten connectivity dan mempunyai bundle pada layer sehingga saat koneksi putus dan tersambung kembali proses pengiriman file dapat dilanjutkan tanpa mengulang proses dari awal dan tanpa ada file yang di buang. *Raspberry Pi* merupakan device yang mempunyai bentuk cukup kecil. Sehingga dapat dibawa dengan mudah. Pengujian pengiriman file dilakukan dengan jeda. Pengujian pertama dilakukan dengan mengirim file dari host 1 ke host 2 melalui DTN-Mules dengan jumlah 1000 file, pengiriman kedua dengan jumlah 2000 file, pengiriman ketiga dengan jumlah 3000 file, pengiriman keempat dengan jumlah 4000 file dan pengiriman kelima dengan jumlah 5000 file. File tersebut sampai pada host 2 dengan persentase packet loss sebesar 0%, karena dalam praktiknya file sebelum dikirim telah digenerate MD5 sehingga menghasilkan MD5 hash. Dan pada saat sampai pada host 2, file di-generate ulang dan dibandingkan dengan MD5 sebelum dikirim, dan hasilnya 100% file yang diterima tidak ada yang hilang dan tidak ada file yang rusak. semua file yang dikirim diuji dengan software DTN yaitu IBR-DTN dan ION DTN.

Kata Kunci : Internet, DTN, ION-DTN, IBR-DTN, *Raspberry Pi*

Abstract

At this time the Internet is a necessity. Almost everyone uses the Internet in daily activities such as sending files to a some place for work purposes. However, not all places have adequate access to Internet access, so file transmission is hampered. Delivery delays caused by inadequate Internet access due to commonly used Internet protocols such as TCP / IP that cannot send files with a high delay, however, with the presence of the DTN (Delay Tolerant Network) the Internet is no longer impossible because DTN is a network protocol that provides inadequate Internet connectivity. because DTN has the advantage of being able to send files in intermitten connectivity and non-intermittent connectivity and has a bundle at the layer so that when the connection is lost and reconnected the file transfer process can be continued without repeating the process from the beginning and without having any files removed. *Raspberry Pi* is a device that has a fairly small shape. So it can be carried easily. File sending testing is done by pausing. The first test is done by sending files from host 1 to host 2 via DTN-Mules with a total of 1000 files, a second sending with 2000 files, a third sending with 3000 files, a fourth sending with 4000 files and a fifth sending with 5000 files. The file arrives at host 2 with a packet loss percentage of 0%, because in practice the file before it has been generated has been MD5 generated so that it produces MD5 hashes. And when it arrives at host 2, the file is re-generated and compared with MD5 before sending, and the result is that 100% of the files received are not missing and no damaged files. all files sent are tested with DTN software, i.e. IBR-DTN and ION DTN.

Keyword: Internet, DTN, ION-DTN, IBR-DTN, *Raspberry Pi*

1. Pendahuluan

Hampir semua orang menggunakan Internet dalam kegiatan sehari-hari seperti mengirim file ke suatu tempat yang jauh untuk kepentingan pekerjaan. Namun tidak semua tempat memiliki akses yang memadai untuk akses Internet, sehingga pengiriman file terhambat. Terhambatnya pengiriman yang disebabkan oleh akses Internet yang tidak memadai dikarenakan protokol Internet yang umum digunakan seperti TCP/IP tidak dapat mengirimkan file dengan waktu tunda (delay) yang tinggi.

Delay Tolerant Network (DTN) merupakan protokol jaringan yang menyediakan solusi untuk pengguna Internet yang memiliki konektivitas Internet yang kurang memadai. DTN tidak

mengutamakan unsur *quality of service* seperti *delay* yang rendah dan *packet loss* yang kecil, namun DTN mengutamakan pengiriman *file* tanpa menyebabkan kerusakan *file*. DTN terdiri dari *node* pengirim, *node*- DTN *router* dan *node* penerima. Pembangunan jaringan DTN yang mudah membuat jaringan DTN banyak dikembangkan oleh peneliti. (Fall dan Kevin, 2003).

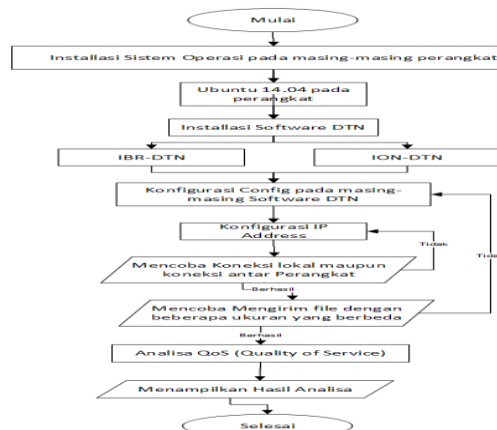
Pada penelitian sebelumnya tentang *Pengembangan Sistem Aplikasi Pengiriman Data Daerah Terpencil Berbasis Delay Tolerant Network* menjelaskan tentang pengembangan pengiriman data di daerah terpencil menggunakan jaringan DTN. Pengiriman dilakukan dengan metode *store and forward* menggunakan kurir digital berupa laptop sebagai DTN *router*. Hal ini memiliki kelemahan yaitu biaya *router* yang mahal karena *router* yang digunakan adalah laptop. Namun sistem ini sangat *userfriendly* untuk mengirim data dari Tepat pemungutan suara (TPS) (Siswanti, 2013).

Penelitian selanjutnya meneliti tentang *Implementasi Interplanetary Overlay Network Pada Delay Tolerant Network Sebagai Jaringan Transmisi Dalam Pencarian Korban Bencana Alam*. Penelitian ini menggunakan *Raspberry pi 3* dengan menggunakan *drone* sebagai DTN *mule* dan dua *client* menggunakan PC. Hal ini memudahkan dalam mentransmisi data dari *client* ke DTN-*Mule*. Dalam pengujian data yang dikirim tidak sepenuhnya terkirim namun *throughput* yang dihasilkan dari dua kali pengujian adalah dibawah *80 ms*. Dan dalam rancangan bangun jaringan DTN tersebut memerlukan biaya yang cukup mahal karena menggunakan *drone* dan *drone* ini memiliki batas jangkauan, sehingga tidak dapat menjangkau area yang sangat jauh. Namun kelebihan dari sistem ini adalah mudah dikembangkan dan dapat menjangkau area yang tidak dapat ditempuh menggunakan jalur darat. (Runi, 2017).

Dari beberapa penelitian diatas, untuk membangun sebuah DTN menggunakan komputer memakan biaya yang cukup mahal dan susunan yang kompleks memerlukan waktu yang lama dalam pembangunan sistem, pemeliharaan sistem dan perbaikan sistem apabila terjadi masalah. *Raspberry pi* merupakan komputer mini model terbaru yang memberikan solusi dari beberapa penelitian diatas. *Raspberry pi* memiliki harga yang murah dibandingkan komputer, memiliki ukuran yang kecil dan mudah dikembangkan seperti energi listrik yang dapat diubah menggunakan baterai sehingga apabila terjadi gangguan listrik maka sistem masih dapat berjalan. Maka dari pernyataan diatas, penulis ingin membuat jaringan DTN yang rendah konsumsi listrik, hemat biaya, susunan jaringan yang simpel sehingga mudah dalam perawatannya dan perbaikan apabila terjadi masalah pada *Raspberry pi*. Dalam menggunakan protokol DTN, semua proses dilakukan menggunakan *software* DTN. *Software* tersebut berguna untuk manajemen bundel yang akan dikirimkan kepada destinasi. Terdapat beberapa *software* yang digunakan untuk menjalankan protokol DTN diantaranya Nya ION-DTN dan IBR-DTN. Kedua *software* tersebut merupakan *software* yang umum digunakan untuk menjalankan protokol DTN. Masing masing mempunyai kelebihan dan kekurangan yang berbeda. Maka penulis mengambil judul "Perbandingan Kinerja ION-DTN dan IBR-DTN menggunakan *Raspberry Pi* sebagai *Router Delay Tolerant Network*"

2. Perancangan Alur Sistem

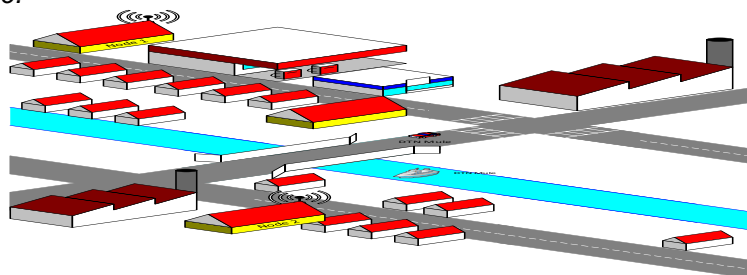
Perancangan alur sistem terdiri dari beberapa tahap yang dilalui dari awal pengiriman data sampai data itu sampai pada tujuan. Pada tahap awal adalah melakukan pemasangan sistem operasi pada masing-masing *node*, pada *node 1* dan *node 2* dipasang sistem operasi Linux Ubuntu 14.04 pada masing-masing perangkat. Setelah di *install* sistem operasi langkah selanjutnya adalah melakukan pemasangan *software* DTN dan melakukan *setting* pada masing-masing *file* config. Setelah itu melakukan konfigurasi *IP Address* dan melakukan pengujian koneksi antar *node*. Jika pada pengujian koneksi berhasil terhubung, langkah selanjutnya adalah melakukan *test* pengiriman *file* dan menghitung *Packet loss* dengan menggunakan *software* ION-DTN dan IBR-DTN. Setelah semua dilakukan maka akan mendapatkan hasil yang sesuai. Alur perancangan bisa dilihat pada bentuk *flowchart* pada Gambar berikut :



Gambar 1. Perancangan Alur Sistem

a. Perancangan Perangkat Keras

Pada Gambar 1 adalah perancangan perangkat keras pada sistem yang akan dibangun untuk pengiriman dari *Node 1* ke *Node 2* menggunakan DTN, dalam proses pengiriman dari *Node 1* ke *Node 2* menggunakan alat yang bergerak. dan alat ini dinamakan *DTN-Mule*. Di sini alat yang dipakai sebagai *DTN-Mule* adalah *Raspberry Pi*, *Raspberry Pi* sendiri adalah komputer mini atau minimum sistem yang dapat menjalankan pekerjaan kantor, memutar media dan mengolah program. Didalam proses ini *Node 1* akan mengumpulkan semua data yang akan di kirim ke *node 2*, *DTN-Mule* akan bergerak mendekati *Node 1* dan setelah mendekati dan *Node 1* terhubung maka *Node 1* akan melakukan pengiriman data ke *DTN-Mule*, setelah *DTN-Mule* selesai menerima data, *DTN-Mule* akan bergerak menuju atau mendekati *Node 2* untuk melakukan pengiriman data dari *Node 1*. Setelah *DTN-Mule* mendekati, maka otomatis *Node 2* akan melakukan penerimaan data dari *DTN-Mule*.



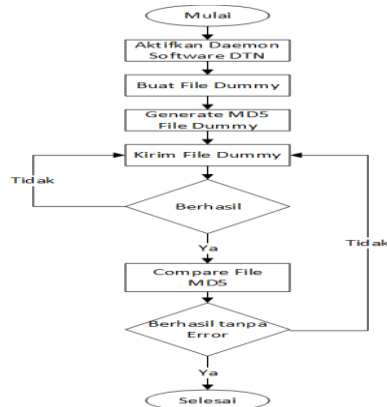
Gambar 2 Perancangan Perangkat Keras

Pada sistem yang akan dibangun ini yang menjadi sarana pengantaran data adalah alat transportasi, seperti kapal, pesawat, bus, mobil, motor dan sepeda.

b. Perancangan Perangkat Lunak

Pada penelitian ini metode yang digunakan adalah metode eksperimen. Langkah pertama dari membandingkan kinerja ION-DTN dan IBR-DTN adalah melakukan pengujian pengiriman *file*. Sebelum membangun jaringan DTN diperlukan rancang bangun jaringan TCP/IP sebagai rancangan awal komputer atau *DTN-Mule* dalam jaringan DTN. Tahap rancangan bangun jaringan TCP/IP terdiri dari Perancangan jaringan dan konfigurasi *IP address* pada *Node* dan *DTN-Mule*. Langkah kedua yaitu instalasi dan konfigurasi IBR-DTN dan ION-DTN. IBR-DTN dan ION-DTN merupakan *software* yang digunakan untuk menjalankan protokol DTN dan bersifat *Open Source*. Kedua *software* DTN ini menggunakan protokol TCP/IP untuk Index alamat protokol DTN. IBR-DTN dan ION-DTN berjalan di atas sistem operasi *Linux Ubuntu 14.04 LTS*. Untuk tahapan instalasi *software* IBR-DTN dan ION-DTN diunduh dari *repository* melalui terminal Ubuntu. Perintah untuk melakukan *download* dan *install software* DTN dapat dilakukan dengan memasukkan perintah melalui terminal Ubuntu. Cara penginstalan *software* IBR-DTN sedikit berbeda dengan cara penginstalan ION-DTN karena pada Ubuntu belum terdapat *repository* untuk IBR-DTN, maka dari itu sebelum meng-*install* IBR-DTN harus memasukkan *repository* melalui terminal. Setelah *software* DTN diunduh, *software* tersebut masih mempunyai konfigurasi yang standar dan perlu dikonfigurasi ulang. Konfigurasi ini berupa pemberian alamat protokol DTN pada *Node* dan

DTN-Mule. Langkah terakhir adalah pengukuran QoS. Pengukuran QoS berupa perhitungan *packet loss* saja karena pada dasarnya DTN merupakan manajemen koneksi yang tidak bergantung pada *delay*, *throughput* dan *jitter*. Pengukuran QoS dilakukan dengan cara mengirim *file*. *File* yang dikirim berupa *file dummy* yang akan dibuat secara acak dan memiliki ukuran yang berbeda serta banyak yang sudah ditentukan sebelumnya.

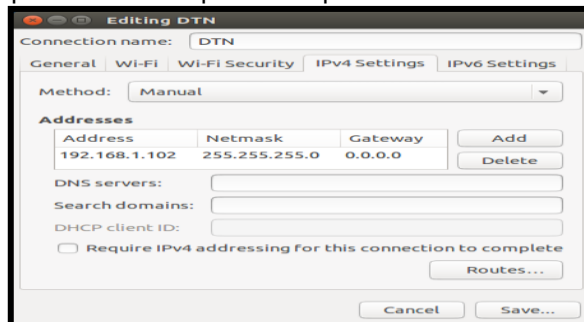


Gambar 3 Flowchart Perancangan Software

3.1 Rancang Bangun Jaringan TCP/IP

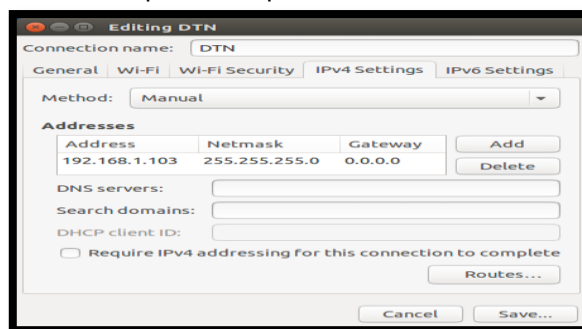
Tahapan pertama pada penelitian ini adalah rancang bangun jaringan TCP/IP. Pada dasarnya jaringan DTN menggunakan protokol TCP/IP sebagai *Index/Node* atau *DTN-Mule* dalam protokol jaringan DTN. Tahap rancang bangun jaringan TCP/IP merupakan menentukan rancangan jaringan yang digunakan dan konfigurasi *IP address* pada *Node* dan *DTN-Mule*.

Langkah berikutnya adalah konfigurasi *IP address* pada *Node* dan *DTN-Mule*. Konfigurasi *IP address* pada penelitian ini menggunakan IPv4. Langkah ini merupakan langkah *index/Node* dan *DTN-Mule* untuk protokol DTN, karena *software* BR-DTN dan ION-DTN menggunakan protokol TCP/IP sebagai *Index/Node* atau *DTN-Mule*. Pada *Node 1* *IP address* yang diberi adalah 192.168.1.102 dengan menggunakan SSID yaitu DTN. Untuk konfigurasi *IP address* pada *Node 1* dapat dilihat pada Gambar 3.6.



Gambar 3.6 Konfigurasi *IP address* Node 1

Pada komputer *mule* diberi *IP address* 192.168.1.103. Untuk konfigurasi *IP address* pada komputer *mule* dapat dilihat pada Gambar 3.7



Gambar 4 Konfigurasi *IP address* pada komputer *mule*

Pada komputer *mule* diberi IP address 192.168.1.104. Untuk konfigurasi IP address pada komputer *mule* dapat dilihat pada Gambar 5.



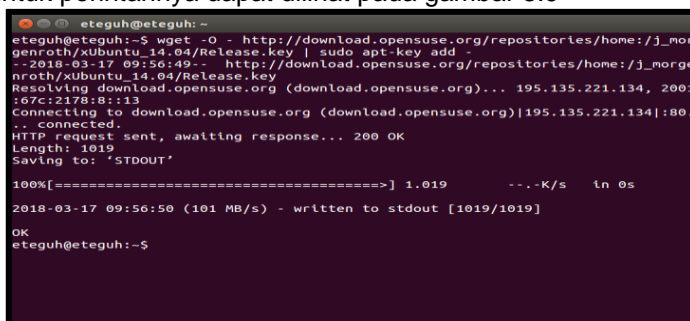
Gambar 5 Konfigurasi IP address Node 2

3.2 Instalasi dan Konfigurasi Jaringan DTN

Setelah IP address berhasil di konfigurasi pada *Node* dan *DTN-Mule*. Langkah selanjutnya adalah melakukan instalasi dan konfigurasi jaringan DTN. Untuk membangun jaringan DTN. Diperlukan *software* yang dapat menjalankan protokol jaringan DTN, *software* yang dipakai dalam penelitian ini adalah IBR-DTN dan ION-DTN. Kedua *software* ini bersifat *Open Source* dan dijalankan dalam sistem operasi Linux Ubuntu 14.04 LTS.

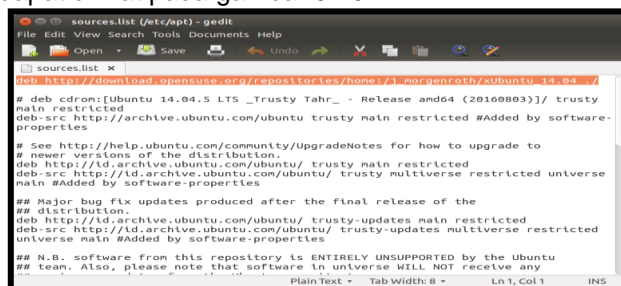
3.6.1 Instalasi dan Konfigurasi IBR-DTN

Dalam tahap instalasi IBR-DTN perlu diketahui bahwa IBR-DTN tidak ada dalam *repository* Ubuntu jadi perlu dimasukkan secara *manual repository* IBR-DTN pada *repository* Ubuntu yang ter-*install*. Langkah untuk memasukkan *repository* IBR-DTN pada *repository* Ubuntu dengan cara memasukkan perintah melalui *terminal*. Adapun perintah yang dimasukkan sebagai berikut
`wget -O http://download.opensuse.org/repositories/home:/j_morgenroth/xUbuntu_14.04/Release.key | sudo apt-key add -` untuk perintahnya dapat dilihat pada gambar 3.9



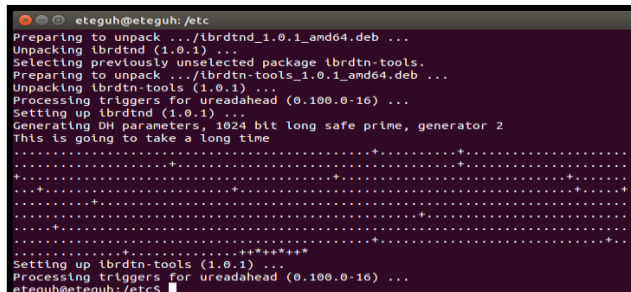
Gambar 3.9 Mengunduh key untuk konfigurasi IBR-DTN

Setelah mengunduh key untuk memasukkan *repository* Ubuntu langkah selanjutnya adalah memasukkan perintah untuk membuka *list repository* Ubuntu untuk memasukkan URL ke *list repository* Ubuntu. Perintah untuk membuka *list* Ubuntu adalah `sudo gedit /etc/apt/sources.list` setelah *list* Ubuntu terbuka, masukkan baris berikut ini pada *list repository* Ubuntu `deb http://download.opensuse.org/repositories/home:/j_morgenroth/xUbuntu_14.04 ./"` untuk *repository ubuntu* dapat dilihat pada gambar 3.10



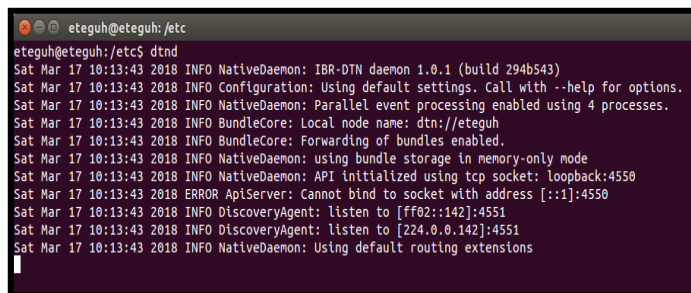
Gambar 3.10 memasukkan list ke repository Ubuntu.

Setelah tahap memasukkan list ke repository Ubuntu selesai tahap selanjutnya adalah melakukan update atau me-refresh repository dengan memasukkan perintah sebagai berikut "sudo apt-get update". Setelah selesai melakukan update langkah selanjutnya yaitu melakukan instalasi IBR-DTN. Meng-install IBR-DTN dapat dilakukan dengan cara memasukkan perintah berikut pada terminal Ubuntu. "sudo apt-get install ibrdtn ibrdtn-tools" untuk meng-install IBR-DTN dapat dilihat pada Gambar 3.11



Gambar 3.11 memasukkan perintah *install* IBR-DTN

Setelah selesai tahap penginstalan. Tahap selanjutnya adalah melakukan pengecekan apakah software IB-DTN sudah ter-install. Perintah yang kita masukkan pada terminal Ubuntu adalah "dtnd" perintah tersebut setelah diketik akan muncul semacam log bahwa IBR-DTN sudah ter-install atau belum. Jika perintah tersebut maka hasilnya bisa dilihat pada gambar 3.12



Gambar 3.12 Software IBR-DTN sudah ter-install dengan benar.

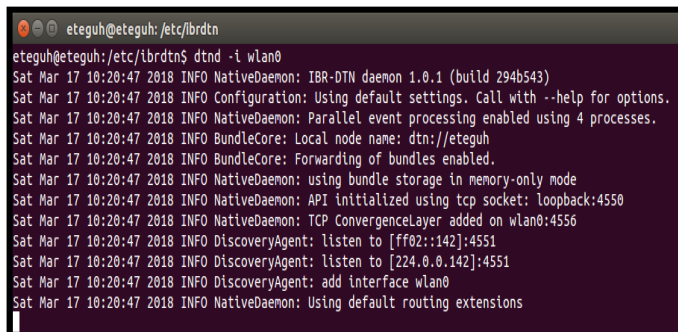
Setelah tahap pemasangan, tahap selanjutnya adalah melakukan konfigurasi *fileconfig* IBR-DTN yang secara default terdapat pada directory "/etc/ibrdtn/" dengan nama "ibrdtn.conf".

Untuk melakukan perubahan konfigurasi *fileconfig* default IBR-DTN masukkan perintah "sudo gedit ibrdtn.conf" pada directory "/etc/ibrdtn". Setelah memasukkan perintah akan keluar baris-baris konfigurasi IBR-DTN. Terdapat beberapa baris yang perlu diubah untuk menyesuaikan konfigurasi yang akan dijalankan. Baris konfigurasi yang perlu diubah adalah baris sebagai berikut:

```
local_uri = DTN://eteguh.dtn
logfile = /var/log/ibrdtn/ibrdtn.log
limit_blocksize = 1.3G
limit_foreign_blocksize = 500M
limit_lifetime = 604800
fragmentation = no
limit_payload = 500K
discovery_address = ff02::142 224.0.0.142
net_lan0_type = tcp
net_lan0_interface = wlan0
net_lan0_port = 4556
net_lan1_type = udp
net_lan1_interface = wlan0
net_lan1_port = 4556
routing = prophet
security_level = 0
```

Pada *fileconfig* IBR-DTN ganti baris default konfigurasi yang supaya sesuai dengan config diatas. Setelah diubah. Simpan dan jalankan IBR-DTN dengan memasukkan perintah

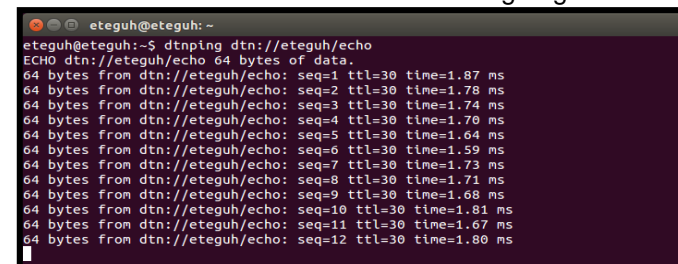
pada terminal Ubuntu. Adapun perintah yang dimasukkan adalah “dtnd -i wlan0 -c ibrdtn.conf”. Penjelasan dari perintah tersebut adalah “dtnd” digunakan untuk menjalankan *software* IBR-DTN, “-i wlan0” adalah parameter untuk menjalankan *Software* IBR-DTN menggunakan interface Wireless yang terpasang pada komputer, “-c ibrdtn.conf” adalah parameter yang digunakan untuk menjadikan ibrdtn.conf sebagai config yang dijalankan *software* IBR-DTN. Adapun perintah untuk menjalankan IBR-DTN dapat dilihat pada gambar 3.13



```
eteguh@eteguh:/etc/ibrdtnd
eteguh@eteguh:/etc/ibrdtnd$ dtnd -i wlan0
Sat Mar 17 10:20:47 2018 INFO NativeDaemon: IBR-DTN daemon 1.0.1 (build 294b543)
Sat Mar 17 10:20:47 2018 INFO Configuration: Using default settings. Call with --help for options.
Sat Mar 17 10:20:47 2018 INFO NativeDaemon: Parallel event processing enabled using 4 processes.
Sat Mar 17 10:20:47 2018 INFO BundleCore: Local node name: dtn://eteguh
Sat Mar 17 10:20:47 2018 INFO BundleCore: Forwarding of bundles enabled.
Sat Mar 17 10:20:47 2018 INFO NativeDaemon: using bundle storage in memory-only mode
Sat Mar 17 10:20:47 2018 INFO NativeDaemon: API initialized using tcp socket: loopback:4550
Sat Mar 17 10:20:47 2018 INFO NativeDaemon: TCP ConvergenceLayer added on wlan0:4556
Sat Mar 17 10:20:47 2018 INFO DiscoveryAgent: listen to [ff02::142]:4551
Sat Mar 17 10:20:47 2018 INFO DiscoveryAgent: listen to [224.0.0.142]:4551
Sat Mar 17 10:20:47 2018 INFO DiscoveryAgent: add interface wlan0
Sat Mar 17 10:20:47 2018 INFO NativeDaemon: Using default routing extensions
```

Gambar 3.13 menjalankan IBR-DTN pada terminal

Setelah *software*DTN berjalan dengan konfigurasi yang sudah diubah. Lakukan ping pada jaringan lokal. Masukkan perintah “DTNping DTN://eteguh/echo” untuk melakukan ping pada jaringan local. Jika berhasil maka akan muncul sesuai dengan gambar 3.14

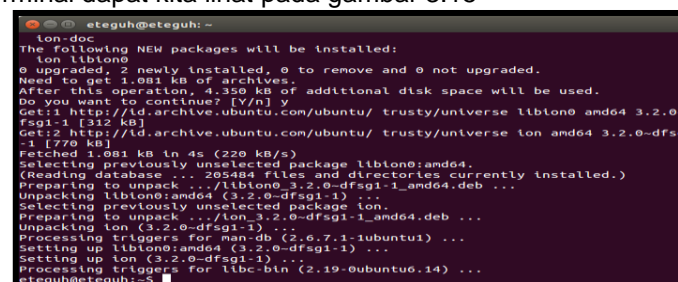


```
eteguh@eteguh:~$ dtncping dtn://eteguh/echo
ECHO dtn://eteguh/echo 64 bytes of data.
64 bytes from dtn://eteguh/echo: seq=1 ttl=30 time=1.87 ms
64 bytes from dtn://eteguh/echo: seq=2 ttl=30 time=1.78 ms
64 bytes from dtn://eteguh/echo: seq=3 ttl=30 time=1.74 ms
64 bytes from dtn://eteguh/echo: seq=4 ttl=30 time=1.70 ms
64 bytes from dtn://eteguh/echo: seq=5 ttl=30 time=1.64 ms
64 bytes from dtn://eteguh/echo: seq=6 ttl=30 time=1.59 ms
64 bytes from dtn://eteguh/echo: seq=7 ttl=30 time=1.73 ms
64 bytes from dtn://eteguh/echo: seq=8 ttl=30 time=1.71 ms
64 bytes from dtn://eteguh/echo: seq=9 ttl=30 time=1.68 ms
64 bytes from dtn://eteguh/echo: seq=10 ttl=30 time=1.81 ms
64 bytes from dtn://eteguh/echo: seq=11 ttl=30 time=1.67 ms
64 bytes from dtn://eteguh/echo: seq=12 ttl=30 time=1.80 ms
```

Gambar 3.14 Menjalankan perintah ping pada jaringan local

3.6.2 Instalasi dan Konfigurasi ION-DTN

ION-DTN merupakan *software* yang dapat menjalankan protokol DTN didalam sistem operasi Linux. ION-DTN ini menggunakan protokol TCP/IP sebagai Index komputer yang menjadi Node atau DTN-Mule pada protokol DTN. Untuk menginstall ION-DTN tidak perlu melakukan penambahan secara manual repository pada Ubuntu, karena pada dasarnya ION-DTN sudah terintegrasi pada repository Ubuntu. Untuk melakukan instalasi ION-DTN ketikkan perintah “sudo apt-get install ion” pada terminal Ubuntu. Setelah memasukkan perintah tersebut hasil pada terminal dapat kita lihat pada gambar 3.15

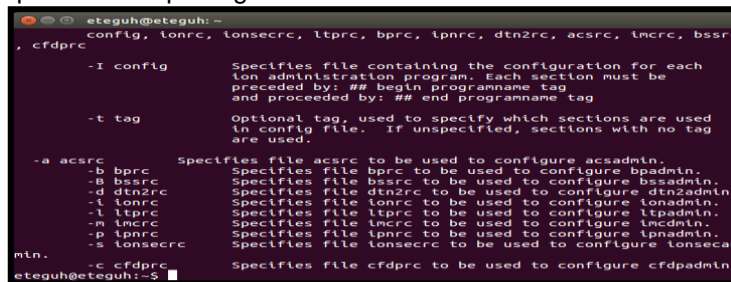


```
eteguh@eteguh:~$ sudo apt-get install ion
ion-doc
The following NEW packages will be installed:
ion liblbn0
0 upgraded, 2 newly installed, 0 to remove and 0 not upgraded.
Need to get 1.081 kB of archives.
After this operation, 4.350 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://id.archive.ubuntu.com/ubuntu/ trusty/universe liblbn0 amd64 3.2.0-dfsg1-1 [312 kB]
Get:2 http://id.archive.ubuntu.com/ubuntu/ trusty/universe ion amd64 3.2.0-dfsg1-1 [770 kB]
Fetched 1.081 kB in 4s (220 kB/s)
Selecting previously unselected package liblbn0:amd64.
(Reading database ... 265884 files and directories currently installed.)
Preparing to unpack .../liblbn0_3.2.0-dfsg1-1_amd64.deb ...
Unpacking liblbn0:amd64 (3.2.0-dfsg1-1) ...
Selecting previously unselected package ion.
Preparing to unpack .../ion_3.2.0-dfsg1-1_amd64.deb ...
Unpacking ion (3.2.0-dfsg1-1) ...
Processing triggers for man-db (2.6.7.1-1ubuntu1) ...
Setting up liblbn0:amd64 (3.2.0-dfsg1-1) ...
Setting up ion (3.2.0-dfsg1-1) ...
Processing triggers for libc-bin (2.19-0ubuntu6.14) ...
eteguh@eteguh:~$
```

Gambar 3.15 Instalasi ION-DTN pada terminal Ubuntu

Setelah tahap *install*, lakukan pemeriksaan apakah *software*ION-DTN sudah ter-*install* atau belum. Lakukan perintah “ionstart” pada terminal Ubuntu. Jika sudah ter-*install* dengan benar maka akan keluar beberapa parameter yang terdapat pada *software*ION-DTN. Karena parameter yang muncul berasal dari *list* parameter yang terdapat pada ION-DTN. Parameter tersebut berguna sebagai format standart penulisan parameter pada Terminal Linux. Jika

penulisan benar maka perintah akan di eksekusi dengan baik. Parameter yang dihasilkan oleh *software*ION-DTN dapat kita lihat pada gambar 3.16



```
etegeh@eteguh:~$ ionstart -h
config, lonrc, lonsecrc, ltrc, bsrc, ipnrc, dtn2rc, acsrc, lncrc, bsrc,
, cfdprc

-I config      Specifies file containing the configuration for each
              ion administration program. Each section must be
              preceded by: ## begin programname tag
              and proceeded by: ## end programname tag

-t tag        Optional tag, used to specify which sections are used
              in config file. If unspecified, sections with no tag
              are used.

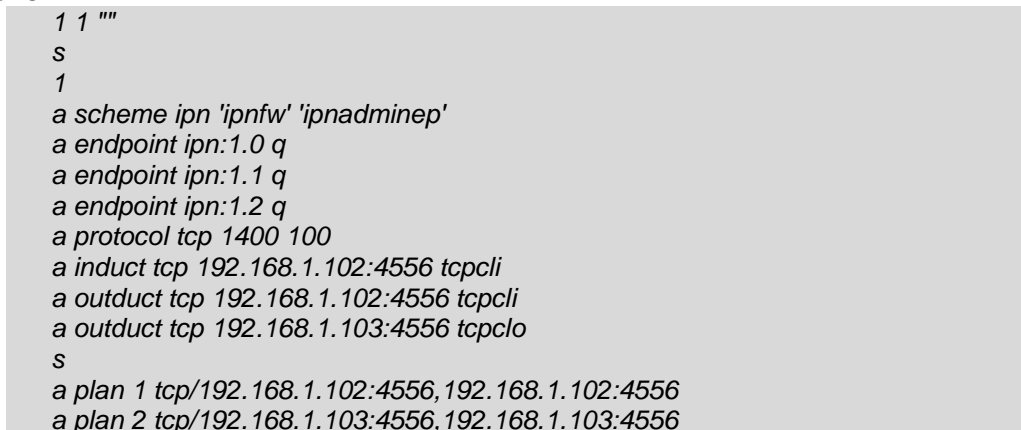
-a acsrc      Specifies file acsrc to be used to configure acsadmin.
-b bsrc       Specifies file bsrc to be used to configure bpsadmin.
-d dtn2rc     Specifies file dtn2rc to be used to configure dtn2admin.
-i ionrc      Specifies file ionrc to be used to configure ionadmin.
-l ltrc       Specifies file ltrc to be used to configure ltradmin.
-m lncrc      Specifies file lncrc to be used to configure lncadmin.
-p ipnrc      Specifies file ipnrc to be used to configure ipnadmin.
-s lonsecrc   Specifies file lonsecrc to be used to configure lonsecad
min.
-c cfdprc     Specifies file cfdprc to be used to configure cfdpadmin.

etegeh@eteguh:~$
```

Gambar 3.16 Parameter ION-DTN

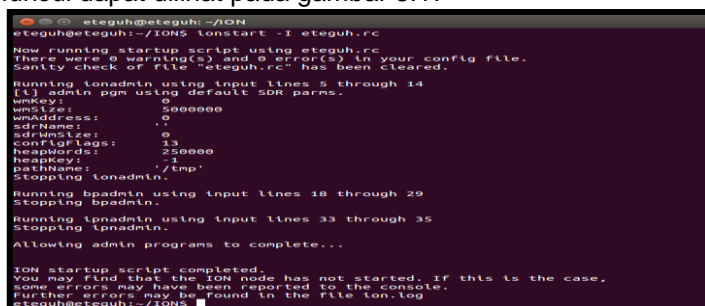
Setelah *software*ION-DTN sudah dipastikan ter-install maka langkah selanjutnya adalah melakukan konfigurasi untuk menjalankan *software*ION-DTN. Pada dasarnya *software*ION-DTN tidak mempunyai *directory default* untuk menaruh *configfile* nya, jadi *config*-nya harus dibuat secara manual tetapi tetap mempunyai aturan dalam membuat *fileconfig*-nya. Adapun *config* yang akan dibuat adalah dengan cara memasukkan beberapa perintah pada sebuah *software*editor pada Ubuntu yaitu menggunakan *SoftwareGedit*.

Untuk membuat *config*ION-DTN buka *software* Gedit yang ada pada Ubuntu dan masukkan *list*perintahkonfigurasi dan lakukan penyimpanan*file* konfigurasinya pada *directory* yang mudah dijangkau. Simpan dengan menggunakan nama dan ekstensi *.rc* karena ekstensi tersebut merupakan ekstensi *default* yang digunakan untuk mengidentifikasi *config* pada *software* ION-DTN .



```
1 1 ""
S
1
a scheme ipn 'ipnfw' 'ipnadminep'
a endpoint ipn:1.0 q
a endpoint ipn:1.1 q
a endpoint ipn:1.2 q
a protocol tcp 1400 100
a induct tcp 192.168.1.102:4556 tcpcli
a outduct tcp 192.168.1.102:4556 tcpcli
a outduct tcp 192.168.1.103:4556 tcpcli
S
a plan 1 tcp/192.168.1.102:4556,192.168.1.102:4556
a plan 2 tcp/192.168.1.103:4556,192.168.1.103:4556
```

Setelah selesai membuat *config* untuk *software*ION-DTN lakukan perintah untuk menjalankan *software*ION-DTN menggunakan *config* yang sudah dibuat pada *directory* tempat penyimpanan tadi. Jalankan *software*ION-DTN menggunakan perintah "ionstart -I ionconfig.rc". maka akan muncul log yang menandakan *software*ION-DTN sudah berjalan. Log yang muncul menampilkan bahwa ION-DTN berjalan menggunakan *config* yang sudah pilih dan jika terjadi error atau peringatan maka akan ditampilkan juga pada log tersebut. BeberapaLog yang muncul dapat dilihat pada gambar 3.17



```
etegeh@eteguh:~$ ionstart -I etegeh.rc
etegeh@eteguh:~/IONS$ ionstart -I etegeh.rc
Now running startup script using etegeh.rc
There were 0 warning(s) and 0 error(s) in your config file.
Sanity check of file "etegeh.rc" has been cleared.

Running ionadmin using input lines 5 through 14
[!] admin pgn using default SDK params.
wmKey: 0
wmName: 500000
wmAddress: 0
adrWmsSize: 0
configTags: 13
heapWords: 250000
heapKey: 1
batchName: '/tmp'
Stopping ionadmin.

Running bpsadmin using input lines 18 through 19
Stopping bpsadmin.

Running ipnadmin using input lines 33 through 35
Stopping ipnadmin.

Allowing admin programs to complete...

ION startup script completed.
You may find that the ION node has not started. If this is the case,
some errors may have been reported to the console.
Further errors may be found in the file ion.log
etegeh@eteguh:~/IONS$
```

3. Analisa dan Hasil Analisa Perbandingan ION-DTN dan IBR-DTN

Dari hasil pengujian yang telah dilakukan maka dapat disimpulkan bahwa *Raspberry Pi* layak digunakan sebagai *router* DTN karena semua *file* dari pengujian 1000 *file* hingga 5000 *file* yang dikirim dari *host* 1 ke *host* 2 melalui *Router Delay Tolerant Network* yang menggunakan *Raspberry Pi* mampu menangani dengan baik semua *file* sampai pada destinasi tanpa adanya *packet* yang hilang atau rusak. *packet* yang utuh dapat dilihat pada masing-masing pengujian dengan memeriksa struktur *file* MD5 yang dibuat pada *host* 1 dan diterima serta dibandingkan pada *host* 2. Tidak adanya *error* pada *file* yang diterima menunjukkan bahwa *packet* yang diterima dalam keadaan utuh.

Software IBR-DTN lebih baik dari pada ION-DTN karena dalam hal kecepatan mengirim *file* IBR-DTN lebih cepat dalam mengirim *file*. Berbeda dengan ION-DTN dalam mengirim *file* termasuk dalam kategori lambat. Kelebihan lain yang dimiliki IBR-DTN adalah bisa mengirim banyak *file* dalam sekali kirim, tetapi ION-DTN hanya dapat mengirim 1 *file* dalam sekali pengiriman. Sehingga perlu dikompres jika ingin mengirim banyak *file* dalam 1 pengiriman.

4. Kesimpulan

Dari hasil pengujian dan analisa hasil pada penelitian ini, kesimpulan yang dapat diambil adalah sebagai berikut :

1. *RaspberryPi* layak digunakan sebagai *router* DTN karena semua *file* yang di buat dapat *handle* dengan baik.
2. Dari kedua *software* penguji yaitu IBR-DTN dan ION DTN yang masing-masing memiliki kelebihan dan kekurangan masing-masing.
3. Dari segi kelengkapan fitur, *software* IBR-DTN mempunyai fitur *standart* dan pada *software* ION-DTN mempunyai fitur yang lebih lengkap dibandingkan IBR-DTN.
4. Dari segi banyaknya *file* yang bisa dikirim dalam sekali kirim, IBR-DTN dapat mengirim *file* banyak dalam sekali pengiriman, sedangkan ION-DTN hanya dapat mengirim satu *file* sekali pengiriman.
5. Dari segi kecepatan pengiriman *file*, IBR-DTN lebih cepat proses pengirimannya dari pada ION-DTN
6. Dari segi kemudahan dalam menggunakan IBR-DTN lebih mudah digunakan, sedangkan ION-DTN

Daftar Pustaka

- Siswanti, Sri Desy, Pengembangan Sistem Aplikasi Pengiriman Data Daerah Terpencil Berbasis DelayTolerant Network , Vol.8, No.2, Hal 238-253, 2013
- Naisuty, Meisuchi, Wibowo, A. U. A., Suhatman, Rahmat. Analisa Kinerja Protokol TCP/IP dan DTN Pada Jaringan Multi Jalur, Politeknik Caltex Riau, Vol.1, 2012
- Siti N, Runi. Rumani M., Setianingsih, Casi. Implementasi InterplanetaryOverlay Network Pada DelayTolerant Network Sebagai Jaringan Transmisi Dalam Pencarian Korban Bencana Alam". Vol.4, No.3, Hal 4023, Desember 2017
- Janitra., Wibowo, A. U. A., Rika Perdana Sari3. Implementasi Aplikasi Pengiriman File pada Protokol DTN Berbasis Web . Vol. 3, No. 1, Tahun 2014
- Warthman, Forrest. DelayTolerant Network (DTNs) : A Tutorial v.3.2. Hal.15, 2015
- Setyo, Galih Putro Dwi. Perancangan Dan Emulasi Protokol RoutingEpidemic Dan Static Pada Jaringan Dtn . Vol.4, No.1, Hal 378, April 2017
- Ningsih, Kurnia, Y., Dkk, Analisis QualityOf Service (QoS) pada Simulasi Jaringan Multiprotocol label Switching Virtual Private Network (MplsVpn), JETri, vol.3, no.2, pp.33-48, 2004
- Telecommunications and Internet Protocol Harmonization Over Networks (TIPHON); General aspectsofQualityof Service (QoS), v2.1.1, Juni 1999
- Wardhana, Anggita Nindya Wisnu., Yamin, Muh., Aksara, LM Fid. Analisis QualityOf Service (Qos) Jaringan Internet Berbasis Wireless Lan Pada Layanan Indihome, Vol.3, No.2, Jul-Des 2017, pp. 49-58