

Aplikasi Algoritma Kruskal dalam Sistem Pengunduhan File dengan Metode BitTorrent

Ryan Daniel 13518130
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
13518130@std.stei.itb.ac.id

Abstrak—Tidak jarang kita menemukan orang yang menggunakan aplikasi BitTorrent. Beberapa orang berkata karena kecepatan unduhnya yang luar biasa. Pemanfaatan sistem *peer-to-peer* yang dapat mempercepat kecepatan unduh hingga puluhan bahkan ratusan kali. File bajakan dapat diakses dengan mudah menggunakan BitTorrent. Namun, tidak sedikit juga yang kontra terhadap BitTorrent karena memiliki beberapa bahaya, seperti *Malware* dan keamanan data. Seberapa efisienkah BitTorrent yang diciptakan oleh Bram Cohen ini sehingga banyak orang yang meletakkan perangkatnya dalam risiko yang tidak kecil karena menggunakan BitTorrent? Penulis berharap dengan makalah ini, penulis dapat menjawab pertanyaan di atas.

Keywords—BitTorrent, kruskal, *peer-to-peer*, pohon merentang.

I. PENDAHULUAN

Dewasa ini, dengan internet, kita dapat melakukan melakukan banyak aktivitas. Mulai dari browsing, bermain game, *chatting*, dan lain-lain. Menurut penulis, ada dua aktivitas yang cukup krusial, yang tidak mungkin lepas dari kehidupan sehari-hari para pengguna internet, yakni pengunduhan file. Sangat sulit membayangkan internet tanpa kedua aktivitas tersebut. Bahkan aktivitas-aktivitas di atas, secara tidak langsung, melakukan kedua aktivitas ini.

Menurut referensi [1], pengunduhan yang merupakan kata bentukan dari unduh, berasal dari kata bahasa Inggris *download* dan *upload*. Unduh berarti memindahkan dari tempat yang lebih tinggi.

Karena aktivitas ini sangat sering dilakukan baik secara langsung maupun tidak langsung, faktor kecepatan merupakan salah satu faktor dalam aktivitas pengunduhan. Menurut referensi [2], paket internet dengan kecepatan 20 Mbps memiliki harga Rp630.000,00. Paket Internet dengan kecepatan 30 Mbps memiliki tambahan harga Rp190.000,00 dari paket 20 Mbps. Paket Internet dengan kecepatan 40 Mbps memiliki tambahan harga Rp365.000,00 dari paket 30 Mbps. Terlihat harga penambahan setiap 10 Mbps selalu dua kali lipat sebelumnya dan pada kondisi ideal, kecepatan internet berbanding lurus dengan kecepatan unduh, dapat disimpulkan bahwa kecepatan merupakan faktor utama yang menentukan kualitas suatu aktivitas pengunduhan.

Selain kecepatan internet, ada faktor lain yang mempengaruhi kecepatan unduh, yakni sistem unduh yang digunakan. Jika menganggap *PC* sebagai simpul dan hubungan antar-*PC* sebagai

sisi, akan didapatkan sebuah graf. Dengan membuat pohon merentang minimum, maka kita dapat membuat jaringan yang lebih efisien. Oleh karena itu, penulis akan membahas bagaimana BitTorrent memanfaatkan pohon merentang minimum.

II. LANDASAN TEORI

A. Graf

Graf merupakan sebuah pasangan himpunan (V, E) , dengan notasi $G = (V, E)$. V adalah himpunan tidak kosong dari simpul(*node*) dan E adalah himpunan sisi(*edge*) yang menghubungkan simpul.

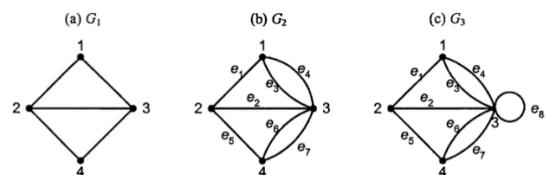
Berdasarkan ada/ tidaknya gelang atau sisi ganda, graf dapat dibagi menjadi dua jenis:

1. Graf sederhana

Graf yang tidak mengandung gelang atau sisi ganda, contohnya G_1 pada gambar 1.

2. Graf ganda

Graf yang mengandung gelang atau sisi ganda, contohnya G_2 pada gambar 1.



Gambar 8.3 Tiga buah graf (a) graf sederhana, (b) graf ganda, dan (c) graf semu

Gambar 1 Ilustrasi jenis-jenis graf

(Sumber: Matematika Diskrit, ed. 3, halaman 369)

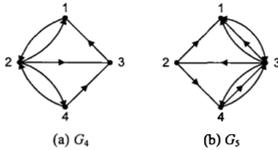
Berdasarkan orientasi arah sisi, graf dibagi menjadi dua jenis:

1. Graf tak-berarah

Graf yang sisinya tidak memiliki orientasi arah. Pada graf tak-berarah, (u,v) dan (v,u) adalah sisi yang sama. Contohnya G_4 pada gambar 2.

2. Graf berarah

Graf yang sisinya memiliki orientasi arah. Pada graf berarah, (u,v) dan (v,u) adalah sisi yang berbeda. Contohnya G_5 pada gambar 2.

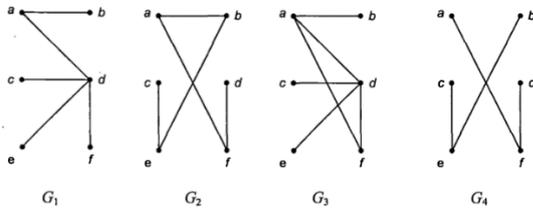


Gambar 8.4 (a) graf berarah, (b) graf-ganda berarah

Gambar 2 Ilustrasi jenis-jenis graf
(Sumber: Matematika Diskrit, ed. 3, halaman 369)

B. Pohon

Pohon merupakan graf sederhana dan tak-terhubung yang tidak mengandung sirkuit.



Gambar 9.1 G_1 dan G_2 adalah pohon, sedangkan G_3 dan G_4 bukan pohon

Gambar 3 Ilustrasi pohon dan bukan pohon
(Sumber: Matematika Diskrit, ed. 3, halaman 457)

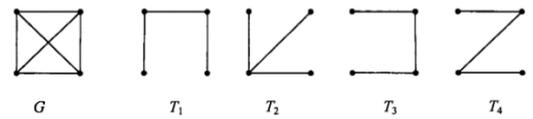
Sifat-sifat pohon:

Misalkan $G = (V, E)$ adalah graf tak-berarah sederhana dan jumlah simpulnya n . Maka pernyataan di bawah ini adalah benar:

- G adalah pohon.
- Setiap pasang simpul di dalam G terhubung dengan lintasan tunggal.
- G adalah terhubung dan memiliki $m = n-1$ buah sisi
- G tidak mengandung sirkuit dan memiliki $m = n-1$ buah sisi
- G tidak mengandung sirkuit dan penambahan satu sisi pada graf akan membuat hanya satu sirkuit
- G terhubung dan semua sisinya adalah jembatan.

C. Pohon Merentang

Misalkan $G = (V, E)$ adalah graf tak-berarah terhubung yang bukan pohon. G dapat diubah menjadi pohon $T = (V, E_1)$ dengan cara memutuskan sirkuit-sirkuit yang ada karena salah satu ciri-ciri pohon adalah tidak memiliki sirkuit. Pertama, pilih sebuah sirkuit, lalu hapus satu buah sisi dari sirkuit ini. G harus tetap terhubung dan jumlah sirkuitnya berkurang satu. Bila proses ini dilakukan berulang-ulang sampai semua sirkuit di G hilang, maka G menjadi sebuah pohon T , yang dinamakan pohon merentang (spanning tree). Disebut pohon merentang karena semua simpul pada pohon T sama dengan semua simpul pada graf G , dan sisi-sisi pada pohon T sisi-sisi pada graf G .

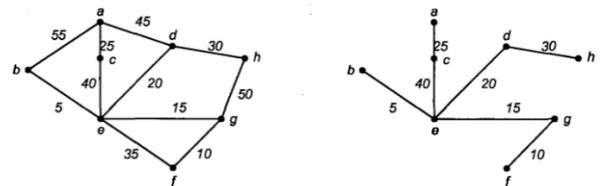


Gambar 9.4 Graf lengkap G dan empat buah pohon merentangnya, T_1 , T_2 , T_3 , dan T_4

Gambar 3 Ilustrasi graf dan pohon-pohonnya
(Sumber: Matematika Diskrit, ed. 3, halaman 447)

D. Pohon Merentang Minimum

Jika G adalah graf berbobot, maka bobot pohon merentang T dari G didefinisikan sebagai jumlah bobot semua sisi di T . Pohon merentang yang berbeda mempunyai bobot yang berbeda pula, tergantung dengan bobot pada sisi-sisi pohon tersebut. Di antara semua pohon merentang di G , pohon merentang yang berbobot minimum dinamakan pohon merentang minimum. Hal ini sangat penting karena memiliki aplikasi yang cukup banyak.



Gambar 9.6 (a) Graf yang menyatakan jaringan jalur rel kereta api. Bobot pada tiap sisi menyatakan panjang rel kereta api ($\times 100$ km)
(b) Pohon merentang yang mempunyai jumlah jarak minimum

Gambar 4 Graf berbobot dan pohon merentang minimumnya
(Sumber: Matematika Diskrit, ed. 3, halaman 463)

E. Algoritma Kruskal

Algoritma ini ditemukan oleh Joseph Kruskal pada 1956 yang ia tulis pada papernya yang berjudul *On the shortest spanning subtree of a graph and the traveling salesman problem*.

Pada algoritma Kruskal, kita harus mengambil sisi-sisi pada graf terlebih dahulu. Lalu mengurutkan graf tersebut dari bobot paling kecil hingga paling besar. Setelah itu, bentuk sebuah pohon dengan menggunakan sisi mulai dari bobot paling kecil. Bila ada pohon yang tidak saling berhubungan, maka dibiarkan saja karena saat akhir proses setiap pohon harus disatukan menjadi hanya satu pohon aja. Misalkan pohon T . Bila sisi A ditambahkan ke T dan membentuk sebuah sirkuit, maka sisi A tidak dapat ditambahkan ke T karena ciri-ciri sebuah pohon adalah tidak memiliki sirkuit.

```

procedure Kruskal(input G : graf, output T : pohon)
{ Membentuk pohon merentang minimum T dari graf terhubung G.
  Masukan: graf-berbobot terhubung G = (V, E), yang mana |V| = n
  Keluaran: pohon rentang minimum T = (V, E')
}

Deklarasi
i, p, q, u, v : integer

Algoritma
{ Asumsi: sisi-sisi dari graf sudah diurut menaik berdasarkan bobotnya }
T ← {}
while jumlah sisi T < n - 1 do
  e ← sisi di dalam E yang bobotnya terkecil
  E ← E - {e}      { e sudah dipilih, jadi buang e dari E }
  if e tidak membentuk siklus di T then
    T ← T ∪ {e}   { masukkan e ke dalam T yang sudah terbentuk }
  endif
endwhile

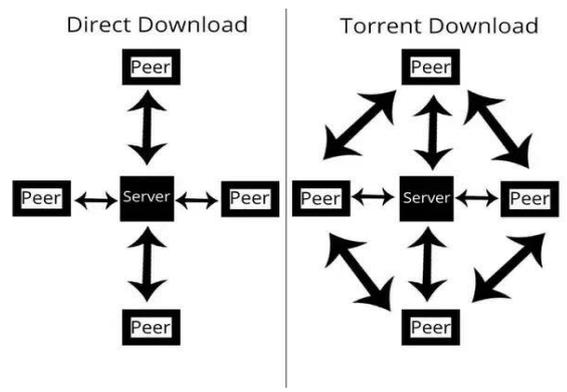
```

Gambar 5 Algoritma Kruskal dalam Notasi Algoritmik (Sumber: Matematika Diskrit, ed. 3, halaman 468)

F. BitTorrent

BitTorrent adalah sebuah sistem pengunduhan file *peer-to-peer* yang biasanya digunakan untuk mengunduh file yang berukuran besar. *Peer-to-peer* adalah suatu model komunikasi dua arah antar pengguna PC melalui jaringan komputer atau Internet tanpa melalui sebuah server. BitTorrent tidak membutuhkan *bandwith* yang besar karena cara kerjanya yang unik. BitTorrent akan memecah file yang akan diunduh ke beberapa bagian kecil.

Perbedaan dengan sistem unduh biasa adalah BitTorrent tidak membutuhkan server sehingga kecepatannya jauh lebih lambat.



Gambar 6 Ilustrasi Sistem Langsung dan Torrent Sumber: <https://www.quora.com/What-is-the-difference-between-a-normal-download-and-a-torrent-download>

Istilah-istilah umum dalam BitTorrent:

1. *Availability*
Angka dari ketersediaan file yang ingin diunduh, nilainya bisa kurang dari satu atau lebih. Semakin besar maka semakin baik dan lebih besar kemungkinan tingkat keberhasilan pengunduhan.
2. *Seeder*
Suatu komputer yang sudah selesai melakukan pengunduhan dan siap melakukan pengunggahan. Secara umum semakin banyak seeder maka semakin cepat downloadnya, dan file tersebut secara utuh sudah tersebar dengan baik.
3. *Peer/Leecher*

Orang yang sedang mendownload torrent dan masih belum selesai, kalau mereka sudah berhasil mengunduh filenya sampai lengkap maka akan berubah menjadi seeder diatas.

4. *Swarm*
Sebuah grup kolektif yang berisi Peer dan Seeder yang dihubungkan oleh suatu file Torrent.

III. ANALISIS DAN PEMBAHASAN

A. Model Permasalahan

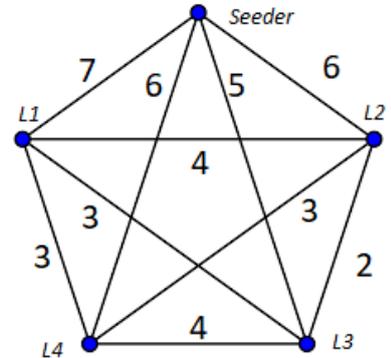
Penulis akan memodelkan masalah sebagai berikut:

1. *Seeder* dan *Leecher* sebagai simpul.
 2. Hubungan antar *Seeder/ Leecher* sebagai sisi.
 3. *Swarm* sebagai graf.
- Sehingga saat *Seeder* akan mengunduh, *Leecher* akan mencari dan membentuk jaringan paling efisien. Dalam kasus ini, *Leecher* akan menggunakan algoritma Kruskal.

Beberapa batasan masalah dalam memodelkan masalah:

1. Hubungan antar *Seeder/ Leecher* melambangkan waktu yang harus ditempuh.
2. Waktu yang harus ditempuh dianggap selalu dalam kondisi ideal sehingga konstan.
3. Hanya terdapat 1 buah *Seeder* dalam sebuah *Swarm*. Dalam kenyataannya, suatu PC dapat menjadi *Seeder* dan *Leecher* bersamaan. Contoh kasusnya adalah bila suatu *Leecher* mengunduh file yang berbeda dari dua *Leecher* yang berbeda.
4. *Seeder* dalam suatu *Swarm* dianggap memiliki kapasitas yang cukup sehingga kondisi pengunduhan dalam kondisi ideal.
5. Karena menurut referensi [6], file dipecah-pecah bila menggunakan Torrent maka *weight* antar-*Leecher* pasti lebih kecil dibandingkan *weight Leecher-Seeder*.
6. *Swarm* awal menggunakan graf lengkap sehingga akan dibuat model permasalahan dengan K3, K4, K5, dan K6.

Metode permasalahan dengan K5:

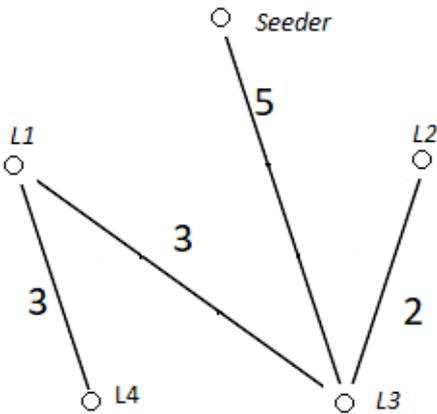


Gambar 7 K5 Sumber: Penulis

Tabel sisi-sisi berdasarkan bobot:

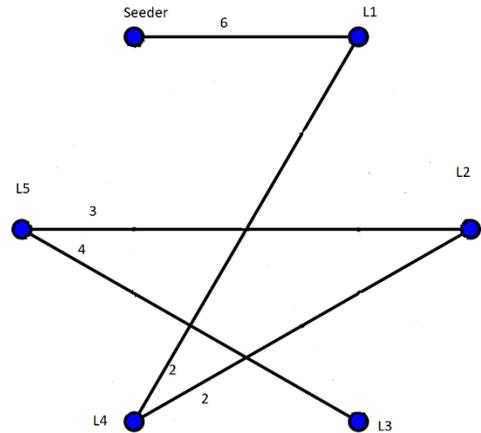
L2-L3	L1-L3	L1-L4	L2-L4	L1-L2
2	3	3	3	4
			TM	

L3	Seeder	Seeder	Seeder	Seeder
-L4	-L3	-L4	-L2	-L1
4	5	6	6	7
	TM	TM	TM	TM



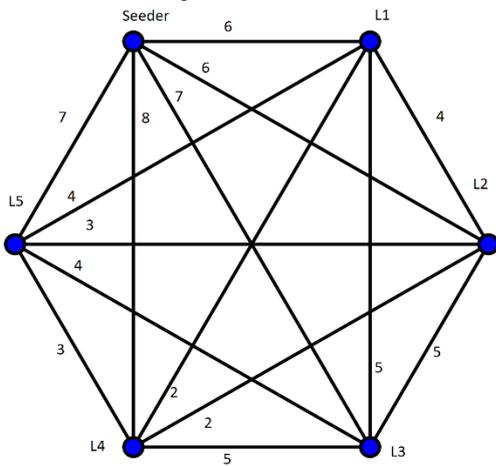
Gambar 8 Pohon Merentang Minimum K5
Sumber: Penulis

-L1	-L2	-L5	-L3	-L4
6	6	7	7	8
	TM	TM	TM	TM



Gambar 10 Pohon Merentang Minimum K6
Sumber: Penulis

Metode permasalahan dengan K6:



Gambar 9 K6
Sumber: Penulis

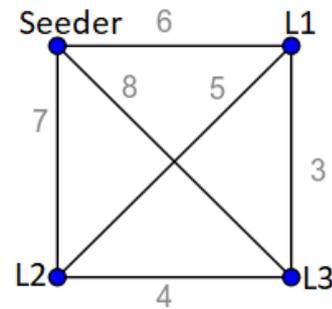
Tabel sisi-sisi berdasarkan bobot:

L1-L4	L2-L4	L2-L5	L4-L5	L1-L5
2	2	3	3	4
			TM	TM

L3-L5	L1-L2	L1-L3	L2-L3	L3-L4
4	4	5	5	5
	TM	TM	TM	TM

Seeder	Seeder	Seeder	Seeder	Seeder
--------	--------	--------	--------	--------

Metode permasalahan dengan K4:

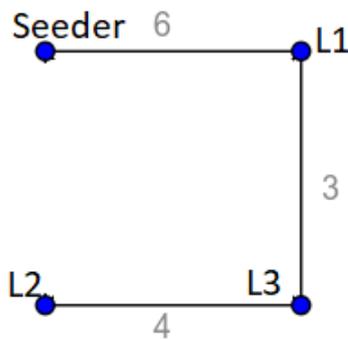


Gambar 11 K4
Sumber: Penulis

Tabel sisi-sisi berdasarkan bobot:

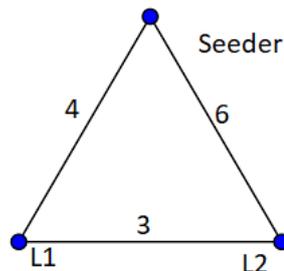
L1-L3	L2-L3	L1-L2	Seeder-L1
3	4	5	6
	TM	TM	

Seeder-L2	Seeder-L3
7	8
TM	TM



Gambar 12 Pohon Merentang Minimum K4
Sumber: Penulis

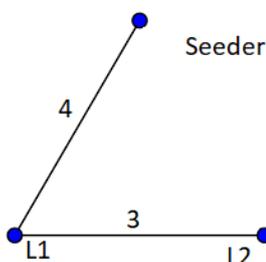
Metode permasalahan dengan K3:



Gambar 13 K3
Sumber: Penulis

Tabel sisi-sisi berdasarkan bobot:

L1-L2	Seeder-L1	Seeder-L2
3	4	6



Gambar 14 Pohon Merentang Minimum K3
Sumber: Penulis

Berdasarkan gambar-gambar dan tabel-tabel di atas maka kita dapat membandingkan durasi unduh dengan metode langsung dan BitTorrent. Untuk metode langsung, jalur pengunduhan semua berasal dari *Seeder*. Berikut ada perbandingannya:

1. Model 1:
 - a. Langsung: 28
 - b. BitTorrent: 13
2. Model 2:
 - a. Langsung: 34
 - b. BitTorrent: 17
3. Model 3:
 - a. Langsung: 21

- b. BitTorrent: 13
4. Model 4:
 - a. Langsung: 7
 - b. BitTorrent: 10

Kekurangan algoritma Kruskal adalah saat jumlah sisi sangat banyak karena pada algoritma ini, semua sisi di iterasi. Namun kelebihan algoritma Kruskal adalah saat jumlah simpul sangat banyak karena algoritma Kruskal tidak berorientasi pada simpul.

Dalam kenyataannya, *peer-to-peer* tidak sesederhana ini. Ada algoritma-algoritma lain yang mempengaruhi kecepatan transfer data. Misalnya, BitTorrent yang kurang stabil jika jumlah *Seeder* sedikit atau jika *bandwidth* yang dimiliki cukup besar. Karena jika *bandwidth* besar, maka sistem langsung memiliki kecepatan yang lebih stabil dan cepat dibandingkan dengan sistem *peer-to-peer*.

Namun, dalam keadaan ideal, algoritma Kruskal sangat menghemat waktu yang dibutuhkan. Karena dengan algoritma Kruskal, sisi yang memiliki bobot berat tidak akan dilalui dan sirkuit tidak akan terbentuk. Sehingga, transfer data dapat bergerak lebih cepat.

Sayangnya, kelebihan BitTorrent harus dibayar dengan risiko yang cukup tinggi. Contohnya *malware*. *Malware* yang dapat merusak perangkat. Hal ini dapat terjadi karena kita tidak mengetahui apakah *Seeder* secara sengaja menyusupi *Malware* atau virus lainnya ke dalam file yang akan diunduh. Dengan melakukan *sharing* file *peer-to-peer*, Torrent akan membuka jalur *gateway* komputer penyimpanan data tersebut. Data-data yang berada dalam komputer kita, terutama yang berada satu dengan satu folder dengan file unggahan BitTorrent

V. KESIMPULAN

1. Algoritma Kruskal terbukti memberikan efisiensi waktu. Namun, pada dunia nyata, dibutuhkan suatu algoritma lain karena keadaan *Swarm* tidak se-ideal yang diberikan pada model di atas. Pertama, Setiap *Leecher* belum tentu terhubung dengan *Leecher* lain. Kedua, ada kemungkinan *Leecher* juga merupakan *Seeder*. Ketiga, ada kondisi durasi antar-*Leecher* lebih lambat dibandingkan *Leecher-Seeder*.
2. BitTorrent memiliki kecepatan yang lebih cepat dibandingkan dengan sistem langsung jika dalam keadaan ideal. Keadaan ideal seperti *speed* yang stabil, *seeder* yang cukup, dan sebagainya. Selain itu, semakin besar *Swarm*/ file yang kita unduh, semakin signifikan efisiensi waktu.
3. Dengan adanya kelebihan dan kekurangan pada BitTorrent, pengguna BitTorrent harus berhati-hati dan sadar akan setiap tindakan yang dilakukan.

REFERENCES

- [1] <http://badanbahasa.kemdikbud.go.id/lamanbahasa/content/asal-kata-mengunduh-dan-mengunggah>. Pada 5/12/2019 15.37.
- [2] <https://www.aturduit.com/internet/indihome/paket-internet-premium>. Pada 5/12/2019 15.46.
- [3] Munir, Rinaldi. Matematika Diskrit, Bandung: Informatika, 2012, edisi ketiga.
- [4] <https://www.geeksforgeeks.org/kruskals-minimum-spanning-tree-algorithm-greedy-algo-2>. Pada 5/12/2019 17.45.

- [5] http://mathshistory.st-andrews.ac.uk/Biographies/Kruskal_Joseph.html
Pada 5/12/2019 18.01.
- [6] <https://www.nesabamedia.com/pengertian-torrent-dan-bittorrent/>. Pada
5/12/2019 18.15.
- [7] <https://idcloudhost.com/kamus-hosting/peer-to-peer/>. Pada 5/12/2019
19.10.
- [8] <https://utekno.com/daftar-istilah-torrent-6933>. Pada 5/12/2019 20.15.
- [9] <http://help.bittorrent.com/customer/portal/articles/179175-glossary>. Pada
5/12/2019 21.10.
- [10] <https://jalantikus.com/tips/bahaya-download-torrent/>. Pada 5/12/2019
22.20.

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 3 Desember 2017

A handwritten signature in black ink, appearing to read 'RD Daniel', written on a light-colored background.

Ryan Daniel-13518130